Advanced Computer Graphics
CSE 190 [Winter 2016], Lecture 11
Ravi Ramamoorthi
http://www.cs.ucsd.edu/~ravir

To Do
- Assignment 2 due Feb 19
  - Should already be well on way.
  - Contact us for difficulties etc.
- This lecture on rendering, rendering equation. Pretty advanced theoretical material. Don’t worry if a bit lost; not directly required on the homeworks.

Course Outline
- 3D Graphics Pipeline
- Rendering (Creating, shading images from geometry, lighting, materials)

Course Outline
- 3D Graphics Pipeline
- Rendering (Creating, shading images from geometry, lighting, materials)
  - Modeling (Creating 3D Geometry)

Unit 1: Foundations of Signal and Image Processing
Understanding the way 2D images are formed and displayed, the important concepts and algorithms, and to build an image processing utility like Photoshop
Weeks 1 – 3: Assignment 1

Unit 2: Meshes, Modeling
Weeks 3 – 5: Assignment 2

Unit 3: Advanced Rendering
Weeks 6 – 7, 8-9: (Final Project)

Unit 4: Animation, Imaging
Weeks 7-8, 9-10: (Final Project)

Illumination Models
- Local Illumination
  - Light directly from light sources to surface
  - No shadows (cast shadows are a global effect)

Global Illumination: multiple bounces (indirect light)
- Hard and soft shadows
- Reflections/refractions (already seen in ray tracing)
- Diffuse and glossy interreflections (radiosity, caustics)

Diffuse Interreflection
Diffuse interreflection, color bleeding [Cornell Box]

Some images courtesy Henrik Wann Jensen
Overview of lecture

- **Theory** for all global illumination methods (ray tracing, path tracing, radiosity)
- We derive **Rendering Equation** [Kajiya 86]
  - Major theoretical development in field
  - Unifying framework for all global illumination
  - Discuss existing approaches as special cases

Fairly theoretical lecture (but important). Not well covered in textbooks (though see Eric Veach’s thesis). See reading if you are interested.

Outline

- **Reflectance Equation** (review)
- **Global Illumination**
  - **Rendering Equation**
  - As a general Integral Equation and Operator
  - Approximations (Ray Tracing, Radiosity)
  - Surface Parameterization (Standard Form)

Reflection Equation

\[
L_r(x, \omega_r) = L_e(x, \omega_r) + \sum L_i(x, \omega_i) f(x, \omega_i, \omega_r) c(\omega_i, \omega_r)
\]

Reflected Light (Output Image)  Emission  Incident Light (from light source)  BRDF  Cosine of Incident angle

Sum over all light sources

\[
L_r(x, \omega_r) = L_e(x, \omega_r) + \sum L_i(x, \omega_i) f(x, \omega_i, \omega_r) c(\omega_i, \omega_r)
\]

Reflected Light (Output Image)  Emission  Incident Light (from light source)  BRDF  Cosine of Incident angle
Reflection Equation

$\omega_i \cdot r \omega_x(x, \omega_i) \cos \theta \cdot d\omega$

Reflected Light (Output Image)  Emission  Incident Light (from light source)  BRDF  Cosine of Incidence Angle

Replace sum with integral

$L_r(x, \omega_i) = L_e(x, \omega_i) + \int \omega_i \cdot f(x, \omega_i, \omega_i) \cos \theta \cdot d\omega$

Environment Maps

- Light as a function of direction, from entire environment
- Captured by photographing a chrome steel or mirror sphere
- Accurate only for one point, but distant lighting same at other scene locations (typically use only one env. map)

Environment Maps

- Environment maps widely used as lighting representation
- Many modern methods deal with offline and real-time rendering with environment maps
- Image-based complex lighting + complex BRDFs

The Challenge

$L_r(x, \omega_i) = L_e(x, \omega_i) + \int \omega_i \cdot f(x, \omega_i, \omega_i) \cos \theta \cdot d\omega$

- Computing reflectance equation requires knowing the incoming radiance from surfaces
- But determining incoming radiance requires knowing the reflected radiance from surfaces

Rendering Equation (Kajiya 86)

$L_r(x, \omega_i) = L_e(x, \omega_i) + \int \omega_i \cdot f(x, \omega_i, \omega_i) \cos \theta \cdot d\omega$

Figure 8: A sample image. All objects are rendered gray. Color on the objects is coming from the green glass table and white lighting from the lamp.
Rendering Equation as Integral Equation

\[ L(x, \omega_r) = L(x, \omega_i) + \int_{S} L(x', \omega_r) \cdot \frac{\cos \theta_i}{\pi} \, dv \]

Is a Fredholm Integral Equation of second kind [extensively studied numerically] with canonical form

\[ I(u) = e(u) + \int K(u, v) \, dv \]

Kernel of equation

Linear Operator Equation

\[ I(u) = e(u) + \int K(u, v) \, dv \]

Kernel of equation

Light Transport Operator

\[ L = E + KL \]

Can be discretized to a simple matrix equation [or system of simultaneous linear equations] 

(L, E are vectors, K is the light transport matrix)

Ray Tracing and extensions

* General class numerical Monte Carlo methods
* Approximate set of all paths of light in scene

\[ L = E + KL \]

\[ IL - KL = E \]

\[ (I - K)L = E \]

Binomial Theorem

\[ L = (I - K)^{-1}E \]

\[ L = (I + K + K^2 + K^3 + \ldots)E \]

\[ L = E + KE + K^2E + K^3E + \ldots \]

Outline

* Reflectance Equation (review)
* Global Illumination
* Rendering Equation
* As a general Integral Equation and Operator
* Approximations (Ray Tracing, Radiosity)
* Surface Parameterization (Standard Form)
Linear Operator Theory

- Linear operators act on functions like matrices act on vectors or discrete representations
  \[ h(u) = (M \circ f)(u) \]
  where \( M \) is a linear operator, \( f \) and \( h \) are functions of \( u \)
- Basic linearity relations hold
  \[ M(af + bg) = a(Mf) + b(Mg) \]
- Examples include integration and differentiation
  \[ (K \circ f)(u) = \int k(u,v)f(v)dv \]
  \[ (D \circ f)(u) = \frac{df}{du} \]

Kernel of equation

Light Transport Operator

\[ L = E + KL \]

Linear Operator Equation

\[ L(u) = \theta(u) + \int (K(u,v)dv \]

Can also be discretized to simple matrix equation

[Solving the Rendering Equation]

- Too hard for analytic solution, numerical methods
- Approximations, that compute different terms, accuracies of the rendering equation
- Two basic approaches are ray tracing, radiosity. More formally, Monte Carlo and Finite Element
- Monte Carlo techniques sample light paths, form statistical estimate (example, path tracing)
- Finite Element methods discretize to matrix equation

Solving the Rendering Equation

\[ L = E + KE + K^2E + K^3E + \ldots \]

Ray Tracing

- Emission directly from light sources
- Direct Illumination on surfaces
- Global Illumination (One bounce indirect) [Mirrors, Refraction]
- (Two bounce indirect) [Caustics etc]
Ray Tracing

\[ L = E + KE + K^2E + K^3E + \ldots \]

- Emission directly from light sources
- Direct Illumination on surfaces
- Global Illumination (One bounce indirect) [Mirrors, Refraction]
- (Two bounce indirect) [Caustics etc]

OpenGL Shading

Outline

- Reflectance Equation (review)
- Global Illumination
- Rendering Equation
- As a general Integral Equation and Operator
- Approximations (Ray Tracing, Radiosity)
- Surface Parameterization (Standard Form)

Change of Variables

\[ L_i(x, \omega_i) = L_i(x, \omega_i) + \int \frac{dA}{|x-x'|} \cos \theta \, d\omega \]

Change of Variables

\[ L_i(x, \omega_i) = L_i(x, \omega_i) + \int \frac{dA}{|x-x'|} \cos \theta \, d\omega \]

Successive Approximation

Rendering Equation

<table>
<thead>
<tr>
<th>Output Image</th>
<th>Emission Reflected Light</th>
</tr>
</thead>
<tbody>
<tr>
<td>UNKNOWN</td>
<td>KNOWN</td>
</tr>
<tr>
<td>KNOWN</td>
<td>UNKNOWN</td>
</tr>
<tr>
<td>KNOWN</td>
<td>KNOWN</td>
</tr>
</tbody>
</table>

Integrand over angles sometimes insufficient. Write integral in terms of surface radiance only (change of variables)

\[ L_i(x, \omega_i) = L_i(x, \omega_i) + \int \frac{dA}{|x-x'|} \cos \theta \, d\omega \]

\[ G(x, x') = \frac{\cos \theta \, \cos \theta}{|x-x'|^2} \]
Rendering Equation: Standard Form

\[ L_i(x, o_i) = L_i(x, o_i) + \int \int \int L_i(x', -\omega) f(x, o_i, o_i) \cos \theta d\omega d\Omega d\omega' \]

Integral over angles sometimes insufficient. Write integral in terms of surface radiances only (change of variables)

\[ L_i(x, o_i) = L_i(x, o_i) + \int \int L_i(x', -\omega) f(x, o_i, o_i) \cos \theta d\omega d\Omega d\omega' \]

Domain integral awkward. Introduce binary visibility fn \( V(x, x') \)

\[ \int \int \int \int f(x, o_i, o_i) G(x, x') V(x, x') d\Omega d\omega d\Omega' d\omega' \]

Same as equation 2.52 Cohen Wallace. It swaps primed and unprimed, omits angular args of BRDF, - sign.

Radiosity Equation

\[ L_i(x, o_i) = L_i(x, o_i) + \int \int L_i(x', -\omega) f(x, o_i, o_i) G(x, x') V(x, x') d\Omega d\omega d\Omega' d\omega' \]

Drop angular dependence (diffuse Lambertian surfaces)

\[ L_i(x) = L_i(x) + \int L_i(x') G(x, x') V(x, x') d\Omega d\omega d\Omega' d\omega' \]

Change variables to radiosity \( B(x) \) and albedo \( \rho \)

\[ B(x) = E(x) + \rho(x) \int B(x') G(x, x') V(x, x') d\Omega d\omega d\Omega' d\omega' \]

Expects conservation of light energy at all points in space

Discretization and Form Factors

\[ B_i = E_i + \rho \sum_j B_j F_{j-i} \frac{A_j}{A_i} \]

F is the form factor. It is dimensionless and is the fraction of energy leaving the entirety of patch \( j \) (multiply by area of \( j \) to get total energy) that arrives anywhere in the entirety of patch \( i \) (divide by area of \( i \) to get energy per unit area or radiosity).

Form Factors

\[ A_{F_{i-j}} = A_j F_{j-i} = \frac{1}{\pi} \int G(x, x') V(x, x') d\Omega dA \]

\[ G(x, x') = G(x', x) = \frac{\cos \theta \cos \theta'}{|x - x'|} \]

Matrix Equation

\[ B_j = E_j + \rho \sum_j B_j F_{j-i} \frac{A_j}{A_i} \]

\[ A_{F_{i-j}} = A_j F_{j-i} = \frac{1}{\pi} \int G(x, x') V(x, x') d\Omega dA \]

\[ B_i = E_i + \rho \sum_j B_j F_{j-i} \]

\[ B_i - \rho \sum_j B_j F_{j-i} = E_i \]

\[ \sum_j M_j B_j = E_j \quad MB = E \quad M_j = I_j - \rho_j F_{j-i} \]

Summary

- **Theory** for all global illumination methods (ray tracing, path tracing, radiosity)
- We derive Rendering Equation [Kajiya 86]
  - Major theoretical development in field
  - Unifying framework for all global illumination
- Discuss existing approaches as special cases