Global Illumination
– The Game of Light Transport

Jian Huang
Looking Back

• Ray-tracing and radiosity both computes global illumination
• Is there a more general methodology?
• It’s a game of light transport.
Radiance

- Radiance (L): for a point in 3D space, L is the light flux per unit projected area per unit solid angle, measured in W/(sr-m^2)

  - sr – steradian: unit of solid angle
    - A cone that covers r^2 area on the radius-r hemisphere
    - A total of 2π sr on a hemisphere Ω.

- power density/solid angle

- The fundamental radiometric quantity

\[ P = \int \int L(x \to \Theta) \cdot \cos \theta \cdot d\omega_\Theta \cdot dA \]

\[ \Omega = \frac{A}{r^2} \]

\( L(x \to \Theta) \): radiance leaving point \( x \) in direction \( \Theta \)

\( L(x \leftarrow \Theta) \): radiance arriving at point \( x \) from direction \( \Theta \)
Irradiance and Radiosity

• Irradiance (E)
  – Integration of incoming radiance over all directions, measured in W/m²
  – Incident radiant power (Watt) on per unit projected surface area

• Radiance distribution is generally discontinuous, irradiance distribution is generally continuous, due to the integration
  – ‘shooting’, distribute radiance from a surface
  – ‘gathering’, integrating irradiance and accumulate light flux on surface

• Radiosity (B) is
  – Exitant radiant power (Watt) on per unit projected surface area, measured in W/m² as well
Relationships among the Radiometric Units

Flux: \( \Phi(x \rightarrow \Theta) \)

Irradiance: \( E(x \leftarrow \Theta) = \frac{d\Phi(x \leftarrow \Theta)}{dA^\perp} \)

Radiant exitance or radiosity: \( B(x \rightarrow \Theta) = \frac{d\Phi(x \rightarrow \Theta)}{dA^\perp} \)

Radiance: \( L(x \rightarrow \Theta) = \frac{d^2\Phi(x \rightarrow \Theta)}{d\omega dA^\perp} = \frac{d^2\Phi(x \rightarrow \Theta)}{d\omega dA \cos \theta} \)

\[
\Phi = \int_A \int_\Omega L(x \rightarrow \Theta) \cos \theta d\omega_\Theta dA_x
\]

\[
E(x) = \int_\Omega L(x \leftarrow \Theta) \cos \theta d\omega_\Theta
\]

\[
B(x) = \int_\Omega L(x \rightarrow \Theta) \cos \theta d\omega_\Theta
\]
Path Notation

- A non-mathematical way to categorize the behavior of global illumination algorithm
  - Diffuse to diffuse transfer
  - Specular to diffuse transfer
  - Diffuse to specular transfer
  - Specular to specular transfer

- Heckbert’s string notation (1990): as light ray travels from source (L) to eye (E):
  - LDDE, LDSE+LDDE, LSSE+LDSE, LSDE, LSSDE
BRDF

• Materials interact with light in different ways, and different materials have different appearances given the same lighting conditions.
• The reflectance properties of a surface are described by a reflectance function, which models the interaction of light reflecting at a surface.
• The bi-directional reflectance distribution function (BRDF) is the most general expression of reflectance of a material.
• The BRDF is defined as the ratio between differential radiance reflected in an exitant direction, and incident irradiance through a differential solid angle:

\[
 f_r(x, \Theta_i \rightarrow \Theta_r) = \frac{dL(x \rightarrow \Theta_r)}{dE(x \leftarrow \Theta_i)} = \frac{dL(x \rightarrow \Theta_r)}{L(x \leftarrow \Theta_i \cos \theta_i d\omega_{\Theta_i}}
\]
BRDF

• The geometry of BRDF
BRDF properties

- Positive, and variable in regard to wave-length
- Reciprocity: the value of the BRDF will remain unchanged if the incident and exitant directions are interchanged.
  \[ f_r(x, \Theta_i \rightarrow \Theta_r) = f_r(x, \Theta_r \rightarrow \Theta_i) \]
- Generally, the BRDF is anisotropic.
- BRDF behaves as a linear function with respect to all incident directions.
  \[ L(x \rightarrow \Theta_r) = \int_{\Omega_x} f_r(x, \Theta \leftrightarrow \Theta_r) L(x \leftarrow \Theta) \cos(n_x, \Theta) d\omega_\Theta \]
BRDF Examples

• Diffuse surface (Lambertian)
  \[ f_r(x, \Theta_i \rightarrow \Theta_r) = \frac{\rho_d}{\pi} \]
  \( \rho_d \) varies from 0 to 1

• Perfect specular surface
  – BRDF is non-zero in only one exitant direction

• Glossy surfaces (non ideally specular)
  – Difficult to model analytically

• Transparent surfaces
  – Need to model the full sphere (hemi-sphere is not enough)
  – BRDF is not usually enough, need BSSRDF (bi-directional sub-surface scattering reflectance distribution function)
  – The transparent side can be diffuse, specular or glossy
Reflectance

• 3 forms
The Rendering Equation

- Proposed by Jim Kajiya in his SIGGRAPH’1986 paper
  - Light transport equation in a general form
  - Describes not only diffuse surfaces, but also ones with complex reflective properties
  - Goal of computer graphics: solution of the rendering equation!
  - Looks simple and natural, but really is too complex to be solved exactly; various techniques to find approximate solutions are used
The Rendering Equation

- $I(x,x') = $ intensity passing from $x'$ to $x$
- $g(x,x') = $ geometry term ($1$, or $1/r^2$, if $x$ visible from $x'$, $0$ otherwise)
- $\varepsilon(x,x') = $ intensity emitted from $x'$ in the direction of $x$
- $\rho(x,x',x'') = $ scattering term for $x'$ (fraction of intensity arriving at $x'$ from the direction of $x''$ scattered in the direction of $x$)
- $S = $ union of all surfaces

\[ I(x,x') = g(x,x') \left[ \varepsilon(x,x') + \int_S \rho(x,x',x'') I(x',x'') dx'' \right] \]
Linear Operator

• Define a linear operator, $M$.

$$M(I)(x, x') = \int_{S} \rho(x, x', x'') I(x', x'')$$

• The rendering equation:

$$I = g\epsilon + gM(I)$$

• How to solve it?
Neumann Series Solution

• Start with an initial guess $I_0$
• Compute a better solution
  \[ I_1 = g\epsilon + gM(I_0) \]
• Compute an even better solution
  \[ I_2 = g\epsilon + gM(I_1) = g\epsilon + gMg\epsilon + gMgM(I_0) \]
• Then, \[ I = g\epsilon + gMg\epsilon + gMgMg\epsilon + gMgMgMg\epsilon + \ldots \]
• In practice one needs to truncate it somewhere
Examples

• No shading/illumination, just draw surfaces as emitting themselves:
  \[ I = g\varepsilon \]

• Direct illumination, no shadows:
  \[ I = g\varepsilon + gM\varepsilon \]

• Direct illumination with shadows:
  \[ I = g\varepsilon + gMg\varepsilon \]
Implications

- How successful is a global illumination algorithm?
  - The first term is simple, just visibility
  - How an algorithm handles the remaining terms and the recursion?
  - How does it handle the combinations of diffuse and specular reflectivity
- The rendering equation is a view-independent statement of the problem
- How are the radiosity algorithm and the ray-tracing algorithm?
Monte Carlo Techniques in Global Illumination

• Monte Carlo is a general class of estimation method based on statistical sampling
  – The most famous example: to estimate $\pi$

• Monte Carlo techniques are commonly used to solve integrals with no analytical or numerical solution
  – The rendering equation has one such integral
Basic Monte Carlo Integration

- Suppose we want to numerically integrate a function over an integration domain $D$ (of dimension $d$), i.e., we want to compute the value of the integral $I$:

$$I = \int_D f(x) \, dx$$

$$D = [\alpha_1 \ldots \beta_1] \times [\alpha_2 \ldots \beta_2] \times \ldots \times [\alpha_d \ldots \beta_d] \quad (\alpha_i, \beta_i \in \mathbb{R})$$

- Common deterministic approach: construct a number of sample points, and use the function values at those points to compute an estimate of $I$.

- Monte Carlo integration basically uses the same approach, but uses a stochastic process to generate the sample points. And would like to generate $N$ sample points distributed uniformly over $D$. 
Basic Monte Carlo Integration

- The mean of the evaluated function values at each randomly generated sample point multiplied by the area of the integration domain, provides an unbiased estimator for $I$:

$$
\langle I \rangle = \left( \frac{1}{N} \sum_{i=1}^{N} f(x_i) \right) \cdot \prod_{i=1}^{d} (\beta_i - \alpha_i)
$$

- Monte Carlo methods provides an un-biased estimator
- The variance reduces as $N$ increases
- Usually, given the same $N$, deterministic approach produces less error than Monte Carlo methods
When to Use Monte Carlo?

- High dimension integration – the sample points needed in deterministic approach exponential increase

- Complex integrand: practically can’t tell the error bound for deterministic approaches

- Monte Carlo is always un-biased, and for rendering purpose, it converts errors into noise!!
Two Types of Monte Carlo

• Monte Carlo integration methods can roughly be subdivided in two categories:
  – those that have no information about the function to be integrated: ‘blind Monte Carlo’
  – those that do have some kind of information available about the function: ‘informed Monte Carlo’

• Intuitively, one expects that informed Monte Carlo methods to produce more accurate results as opposed to blind Monte Carlo methods.

• The basic Monte Carlo integration is a blind Monte Carlo method
Importance Sampling

• An informed Monte Carlo
• Importance sampling uses a non-uniform probability function, $pdf(x)$, for generating samples.
  – By choosing the probability function $pdf(x)$ wisely on the basis of some knowledge of the function to be integrated, we can often reduce the variance
  – Can prove: if can get the $pdf(x)$ to match the exact shape of the function to be integrated, $f(x)$, the variance of the integration estimation is 0.
• Practically, can use a sample table to generate a ‘good’ pdf.
• Intuitively, want to send more rays into the more detailed areas in space
Stratified Sampling

• Importance sampling (probability) using a limited number of samples, which is the case for graphics rendering, does not have a guarantee.

• Stratified sampling address this further: the basic idea of stratified sampling is to split up the integration domain in $m$ disjunct subdomains (also called strata), and evaluate the integral in each of the subdomains separately with one or more samples.

• More precisely:

$$
\int_{0}^{1} f(x) dx = \int_{0}^{\alpha_1} f(x) dx + \int_{\alpha_1}^{\alpha_2} f(x) dx + \ldots + \int_{\alpha_{m-2}}^{\alpha_{m-1}} f(x) dx + \int_{\alpha_{m-1}}^{1} f(x) dx
$$
More On Ray-Tracing

• Already discussed recursive ray-tracing!

• Improvements to ray-tracing!
  – Area sampling variations to address aliasing
    • Cone tracing (only talk about this)
    • Beam tracing
    • Pencil tracing

• Distributed ray-tracing!
Cone Tracing (1984)

• Generalize linear rays into cones
• One cone is fired from eye into each pixel
  – Have a wide angle to encompass the pixel
• The cone is intersected with objects in its path
• Reflection and refraction are modeled as spherical mirrors and lenses
  – Use the curvature of the object intersecting that cone
  – Broaden the reflected and refracted cones to simulate further scattering
• Shadow: proportion of the shadow cone that remains un-blocked
Distributed Ray-Tracing

• Another way to address aliasing
• By Cook, Porter, and Carpenter in 1984.
• A stochastic approach to supersampling that trades objectionable aliasing artifacts for the less offensive artifacts of noise
• ‘Distributed’: rays are stochastically distributed to sample the quantities
• This method was covered during our recursive ray tracing lecture as extension to correct aliasing
Sampling Other Dimensions

- Other than stochastic spatial sampling for anti-aliasing, can sample in other dimensions
  - Motion blur (distribute rays in time)
  - Depth of field (distribute rays over the area of the camera lens)
  - Rough surfaces: blurred specular reflections and translucent refraction (distribute rays according to specular reflection and transmission functions)
  - Soft shadow: distribute shadow feeler rays over the solid angle span by the area light source

- In all cases, use stochastic sampling to perturb rays