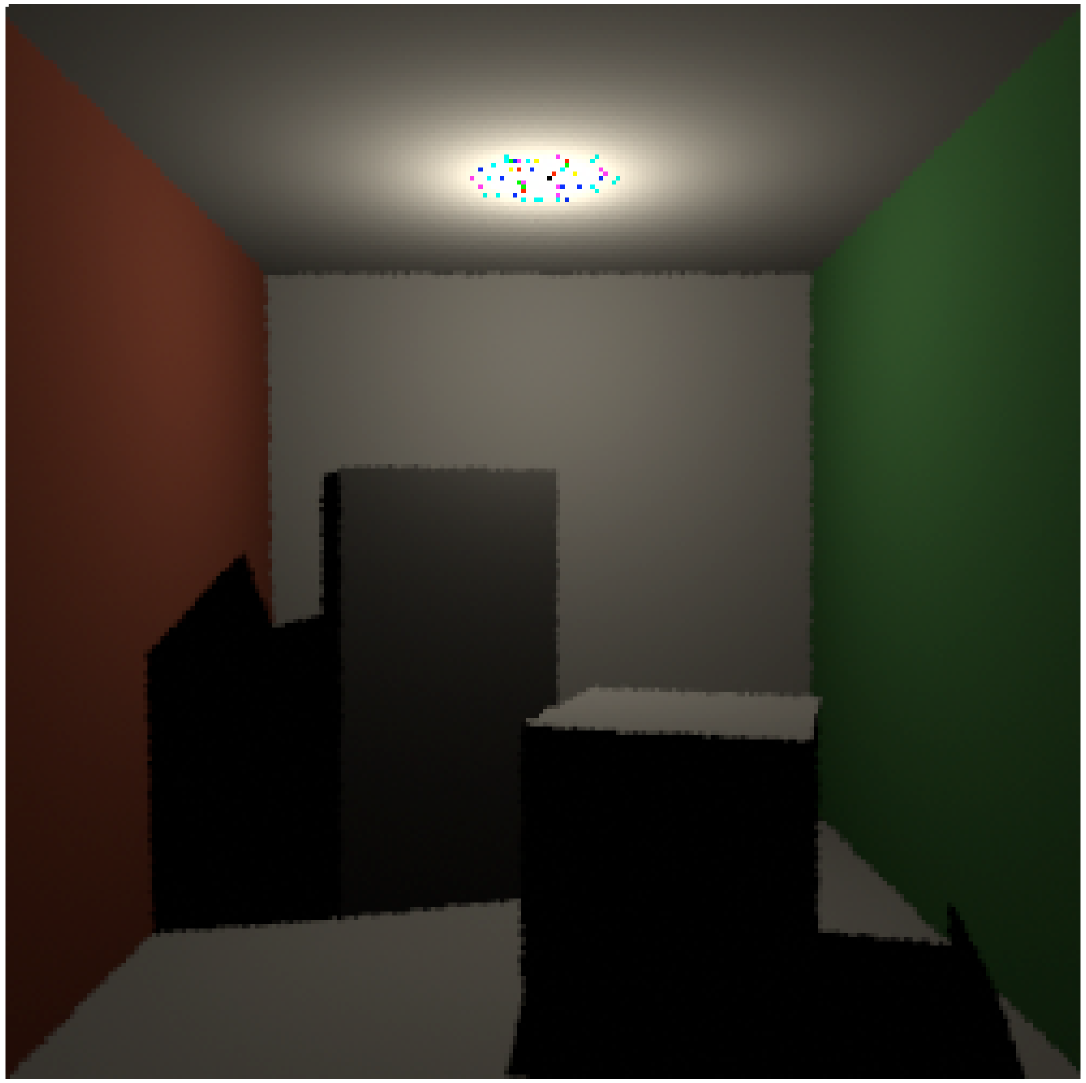
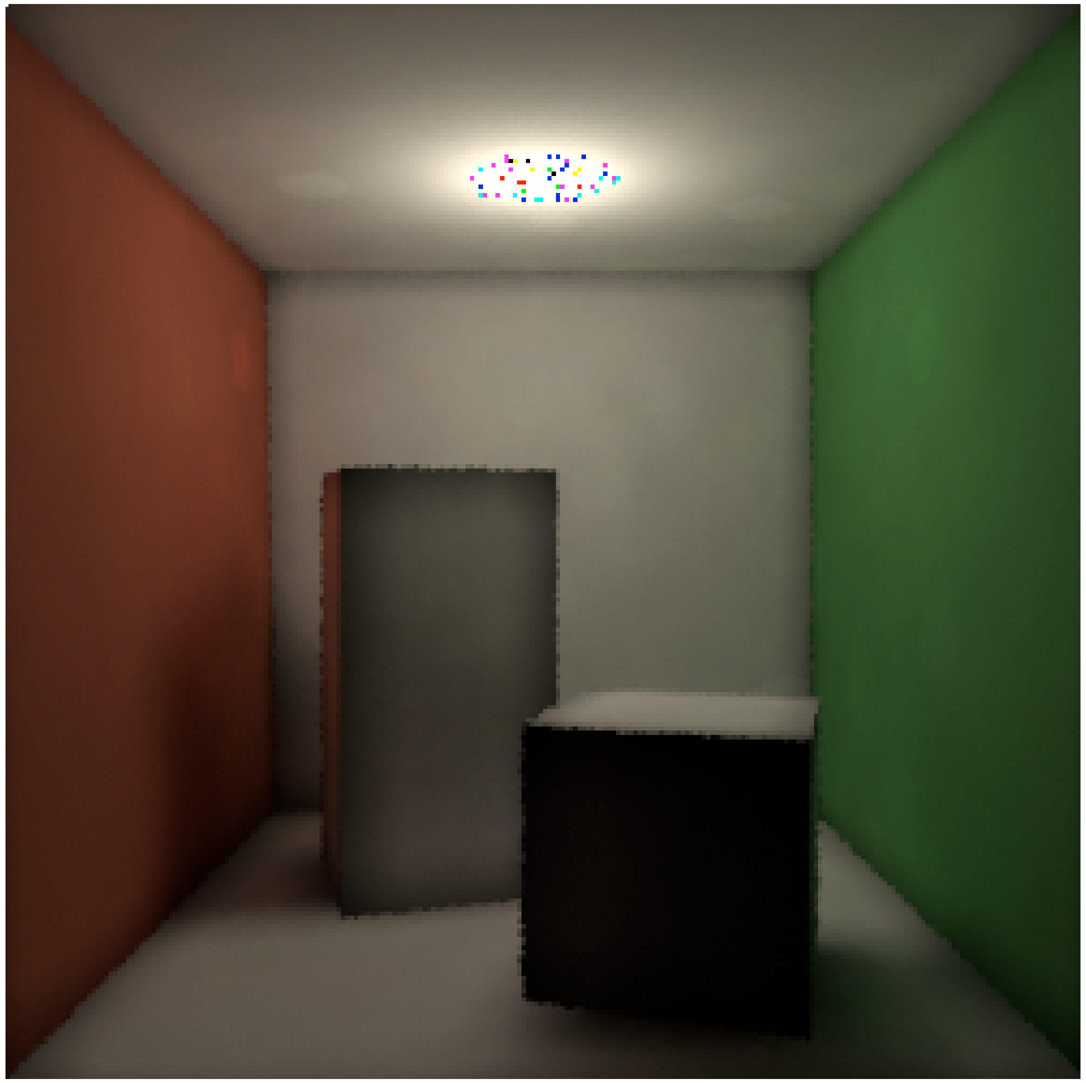
Monte Carlo III: Solving The Rendering Equation

cs348b

Matt Pharr

Local vs. Global Illumination





Overview

- Path tracing
  - Partitioning the rendering equation
  - MC estimates of path contributions
  - Efficiency: path re-use, termination
- Bidirectional path tracing
  - More robust sampling of path space
  - Multiple importance sampling
- Biased methods
  - Light ray tracing / splatting
  - Photon mapping

Partitioning the Rendering Equation

$$L(x,\omega) = L_e(x,\omega) + \int_\Omega f(\omega_i \rightarrow \omega) L(x,\omega_i) d\omega_i$$

$$= L_e(x,\omega) + \int f_1(\omega_i \rightarrow \omega) L_1(x,\omega_i)$$
$$+ \int f_2(\omega_i \rightarrow \omega) L_1(x,\omega_i) + \int f_1(\omega_i \rightarrow \omega) L_2(x,\omega_i)$$
$$+ \int f_2(\omega_i \rightarrow \omega) L_2(x,\omega_i)$$

$$f = f_1 + f_2, L = L_1 + L_2$$

Why Partition?

- Take advantage of known structure
  - Direct vs. indirect light
  - Specular vs. diffuse BRDF
  - ...
- Can apply different solution techniques to different terms

Path Tracing (Kajiya)

- Based on natural recursive expansion of rendering equation

$$L = L_e + \int f L_i \quad \longrightarrow \quad L = L_e + SL$$

$$L = L_e + S(L_e + S(L_e + S(\ldots$$

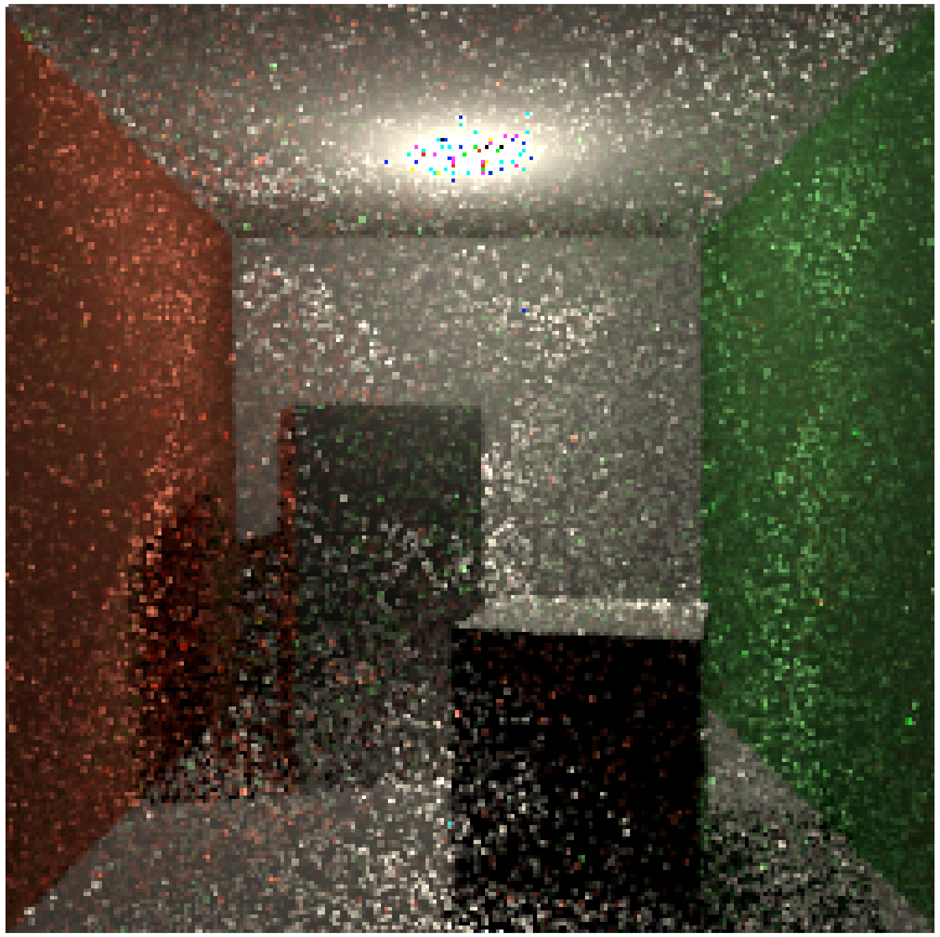
$$L = L_e + SL_e + SSL_e + SSSL_e + \ldots$$

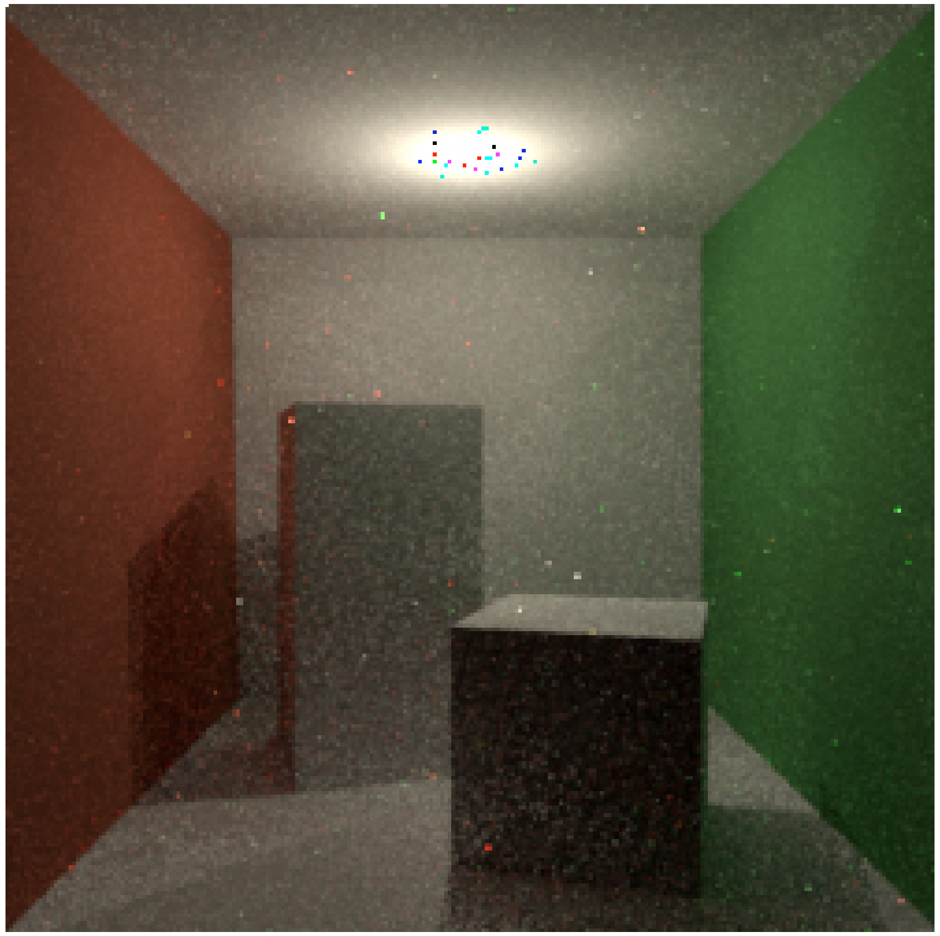
Path Tracing (Kajiya)

- Partition the integrand
  - Separate BRDF into specular/non-specular
  - Separate incoming light into direct/indirect
- No branching of path; one shadow ray, one BRDF ray
  - Be careful to not double count illumination
  - Discrete PDF over lights
- Re-use prefix of path vertices for path with one more vertex
  - Correlation vs. efficiency

Path Tracing

- 1 vs 36 paths per pixel





The Rendering Equation as a Sum over Paths

- Better formulation for thinking about light transport

$$L(x,\omega) = L_e(x,\omega) + \sum_i \int f_i(x, x_1, \ldots, x_i) dA(x_1) dA(x_2) \ldots$$

$$f_i(x, x_1, \ldots, x_i) =$$
$$L_e(x_i \rightarrow x_{i-1}) G(x_i, x_{i-1}) f(x_i \rightarrow x_{i-1} \rightarrow x_{i-2}) \cdots$$
$$G(x_1, x_1) f(x_3 \rightarrow x_2 \rightarrow x_1) G(x_1, x) W_e(x_1 \rightarrow x)$$

Implications

- Don't need to follow paths forward from camera
- Great flexibility in sampling path vertices; doesn't even need to be done sequentially
- Can have much lower variance than path tracing

Sampling Path Vertices

- Can uniformly sample over area on surfaces
- Importance sampling based on important surfaces
- More commonly, incremental path sampling
  - Importance sample BRDF at each step
  - Solid angle to area density conversion

$$p_A(x) = p_\Omega(\omega) \frac{|\cos\theta|}{r^2}$$

Bidirectional Path Tracing

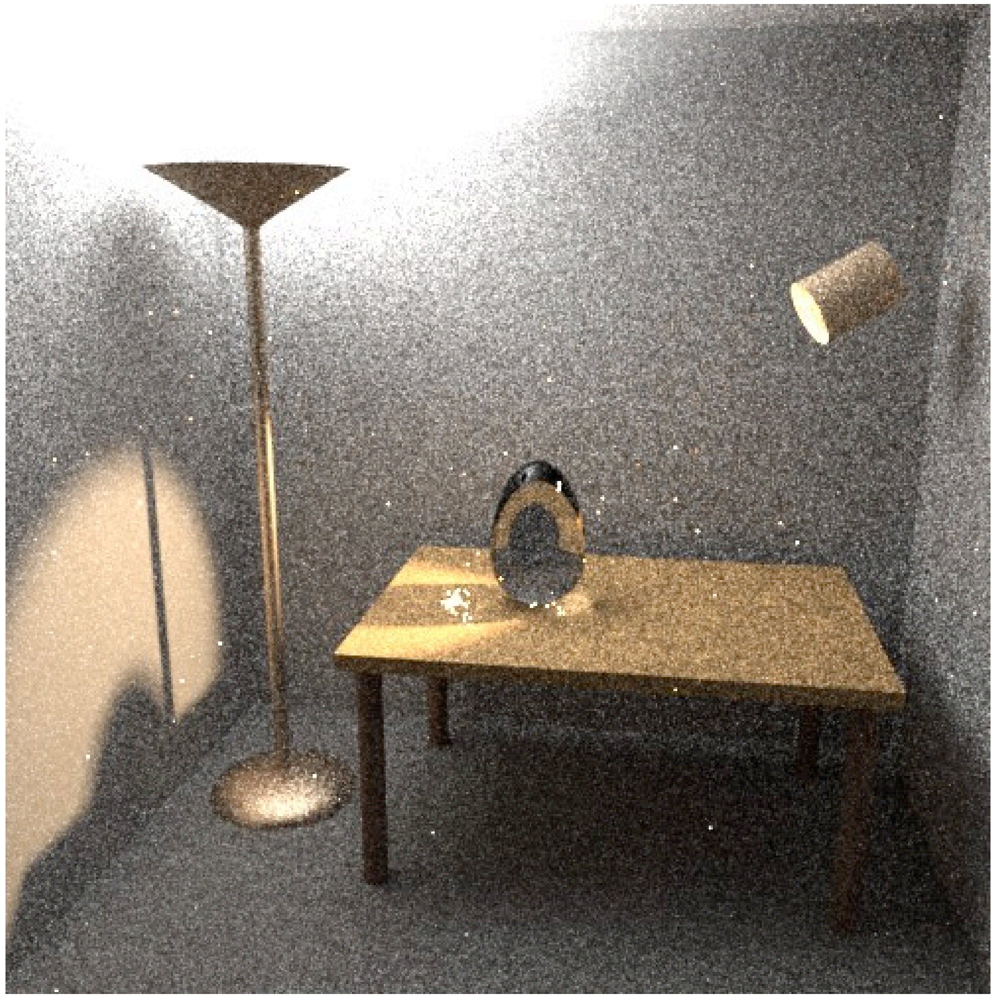
- Handle tricky lighting situations with both types of paths
- Generate path from eye, path from light
- Connect vertices with shadow rays
- Multiple importance sampling to compute each path's contribution

Path Pyramid

$\longrightarrow$ l $\longleftarrow$ e

Bidirectional Path Tracing

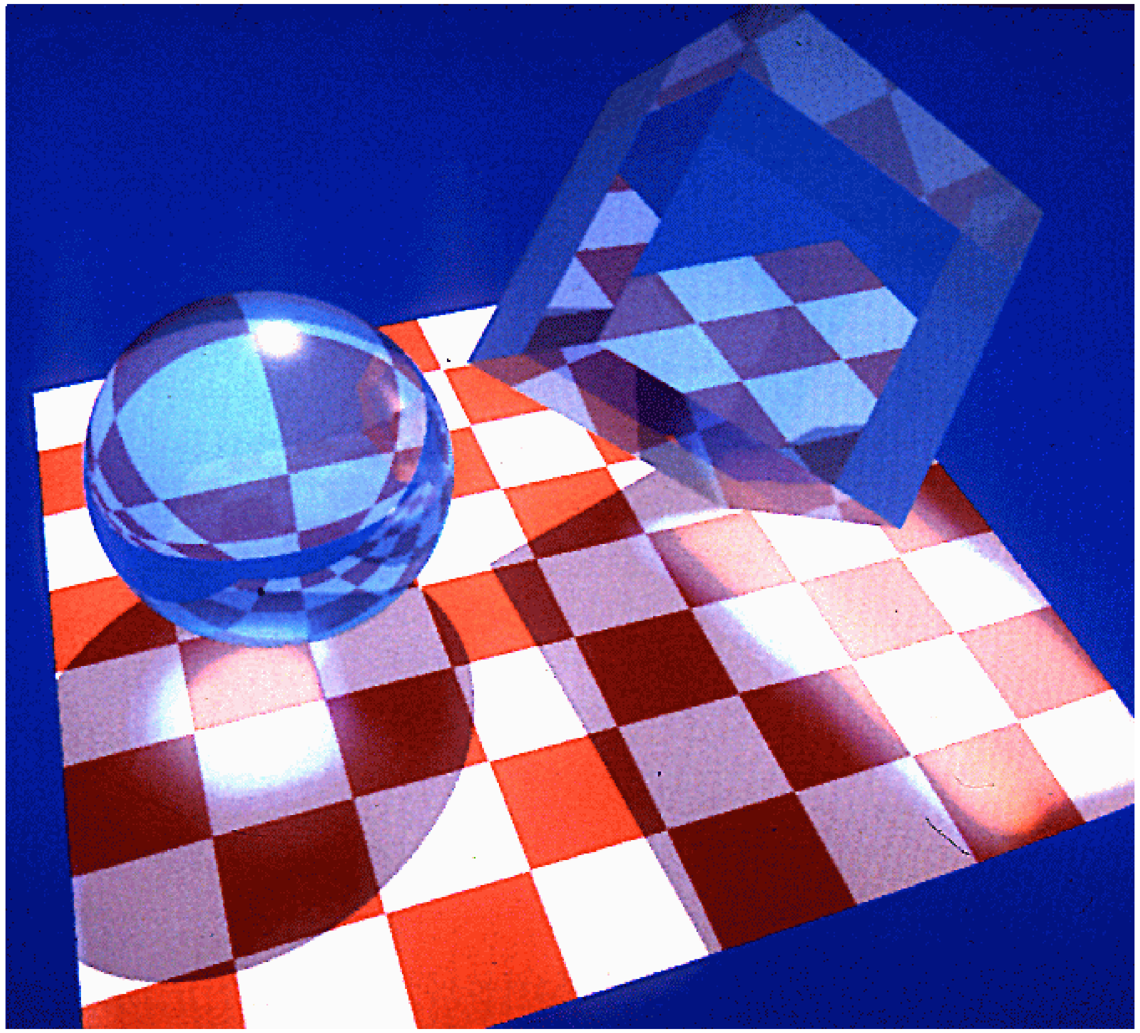
Bidirectional
Path Tracing



Path Tracing

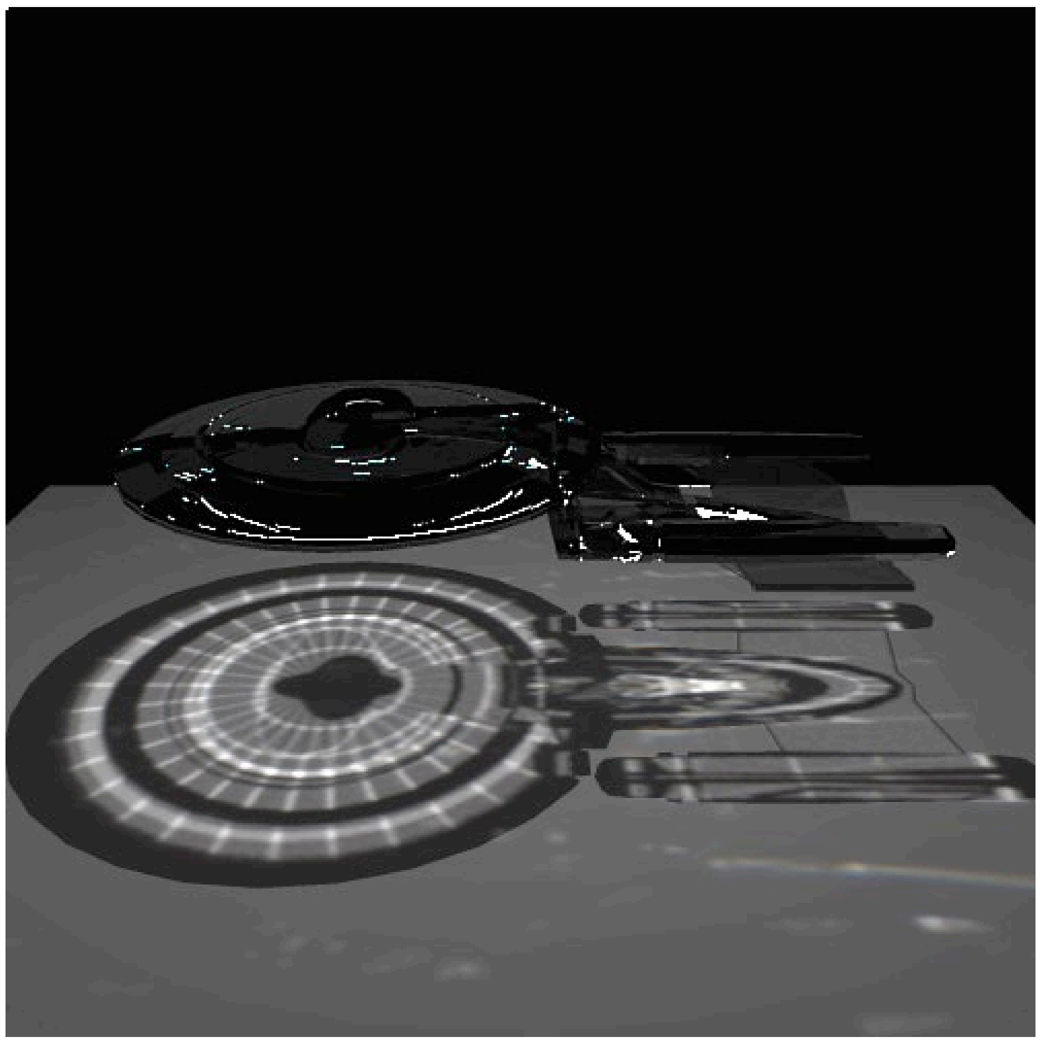
Veach & Guibas

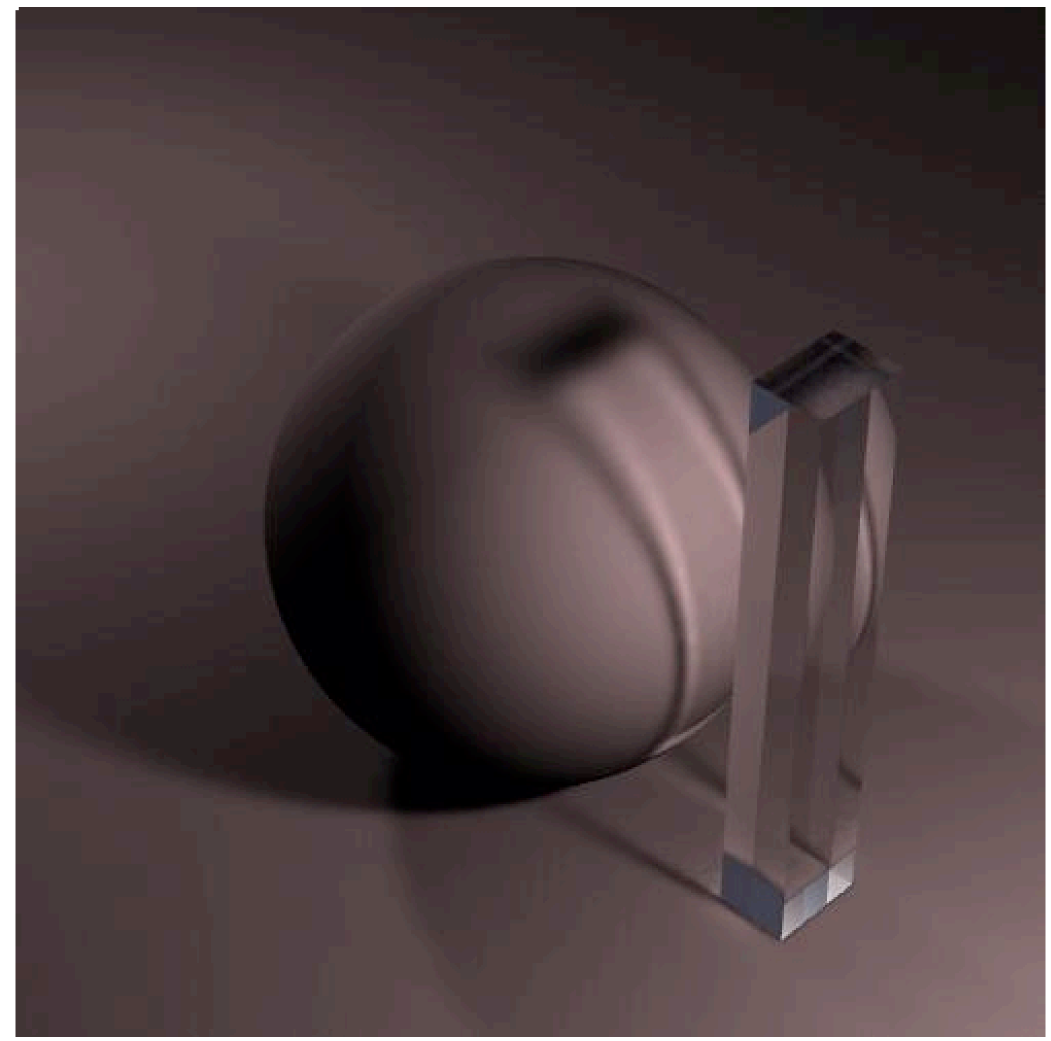
Tracing Paths from Lights



Arvo, 1986

Tracing Paths from Lights





Steve Collins, 1995

Basic Photon Mapping

- Trace paths from lights
- Store samples in kd-tree
  - Fast lookup of nearest photons
- Lookup nearby samples when shading, estimate illumination with density estimate

- Key idea: particle histories give approximation of scene radiance distribution

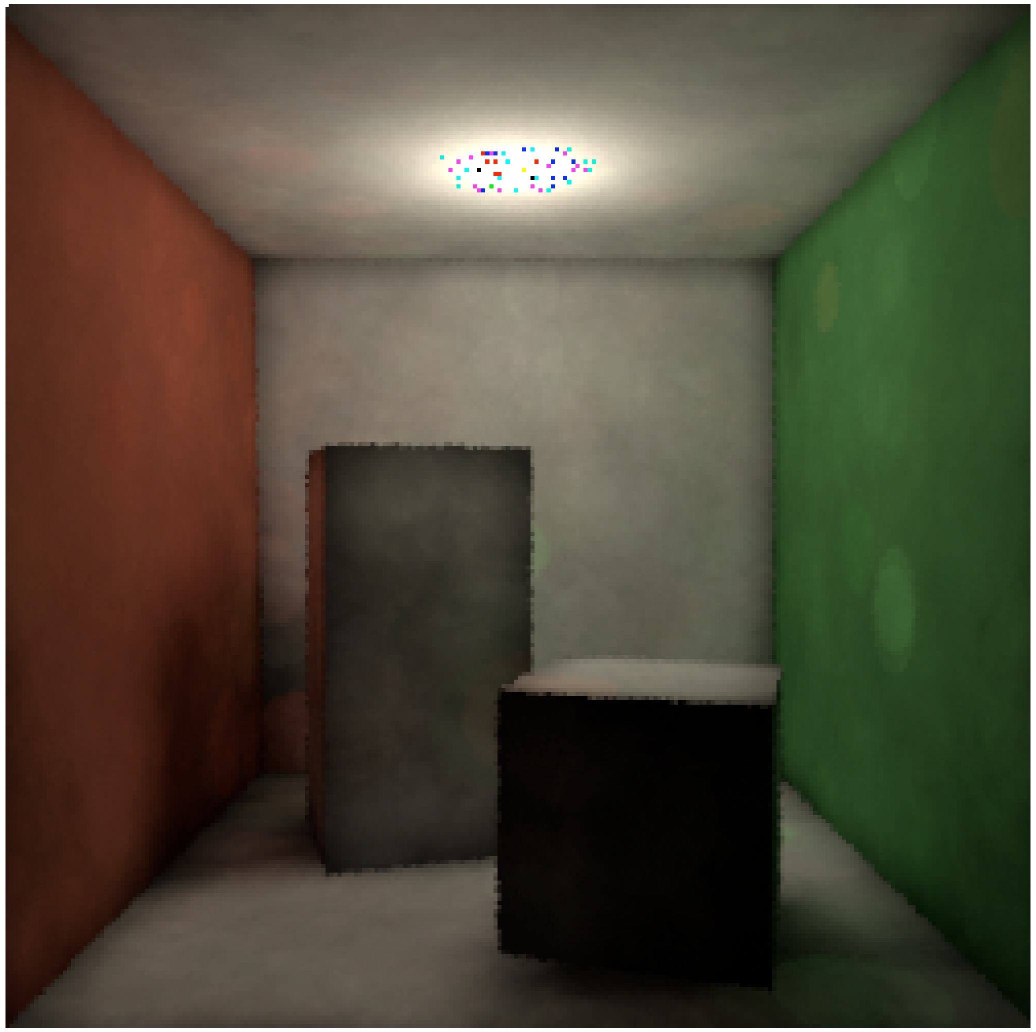
Photon Map-Based Reflection

$$L(x, w) = \int_\Omega f(\omega_i \to \omega) L_i(x, \omega_i) \cos\theta_i d\omega_i =$$

$$\int_\Omega f(\omega_i \to \omega) \frac{d\Phi(x, \omega_i)}{dA d\omega_i} d\omega_i \approx \sum^N f(\omega[i] \to \omega) \frac{\Delta\Phi(x, \omega[i])}{\pi r^2}$$

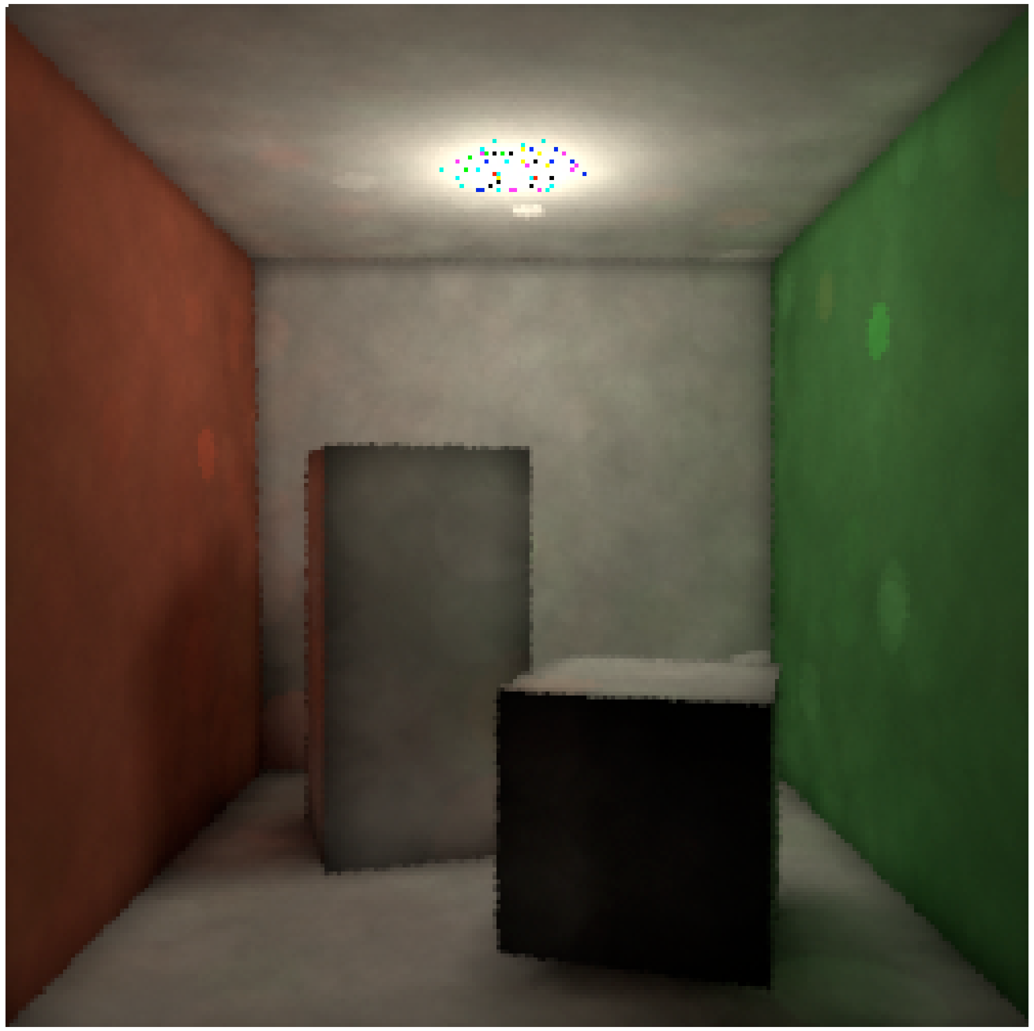
Photon Mapping

- 50k photons



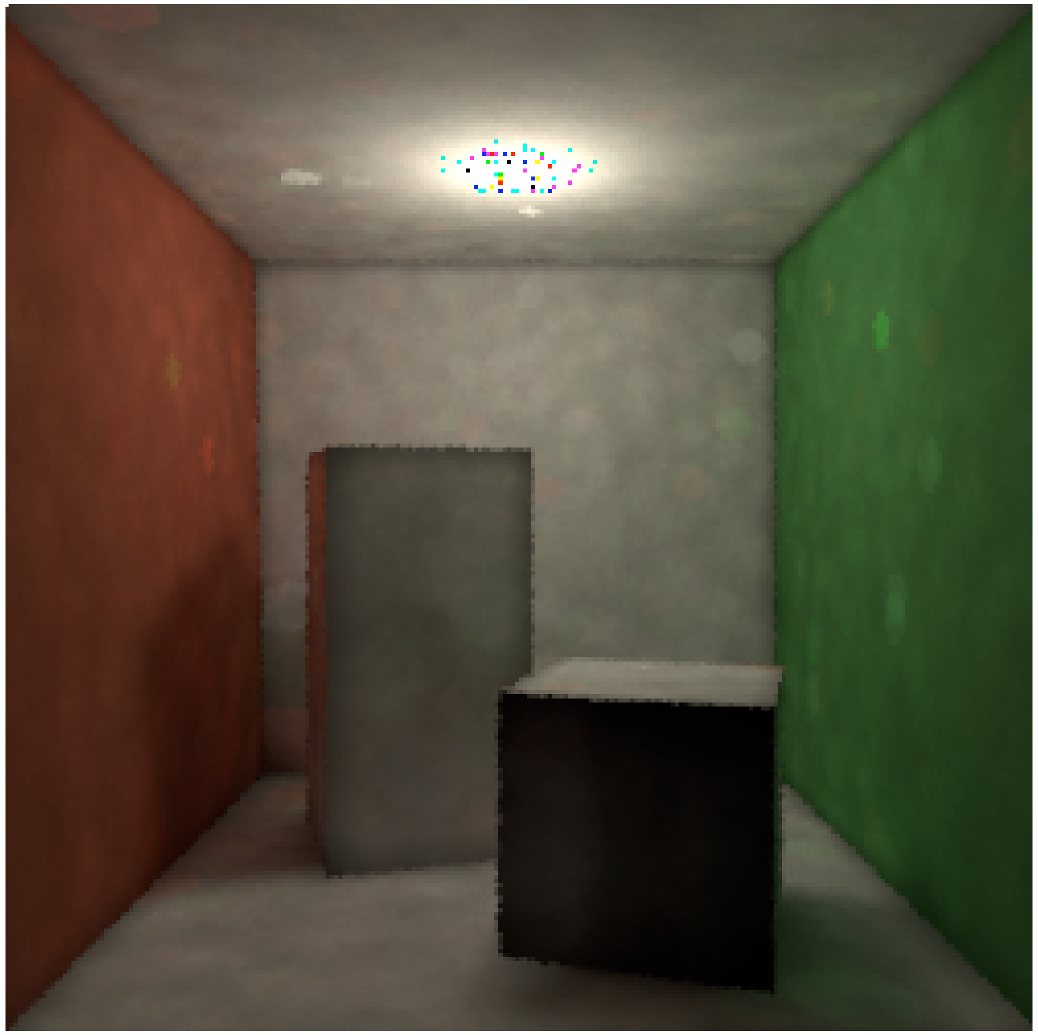
Photon Mapping

- 100k photons



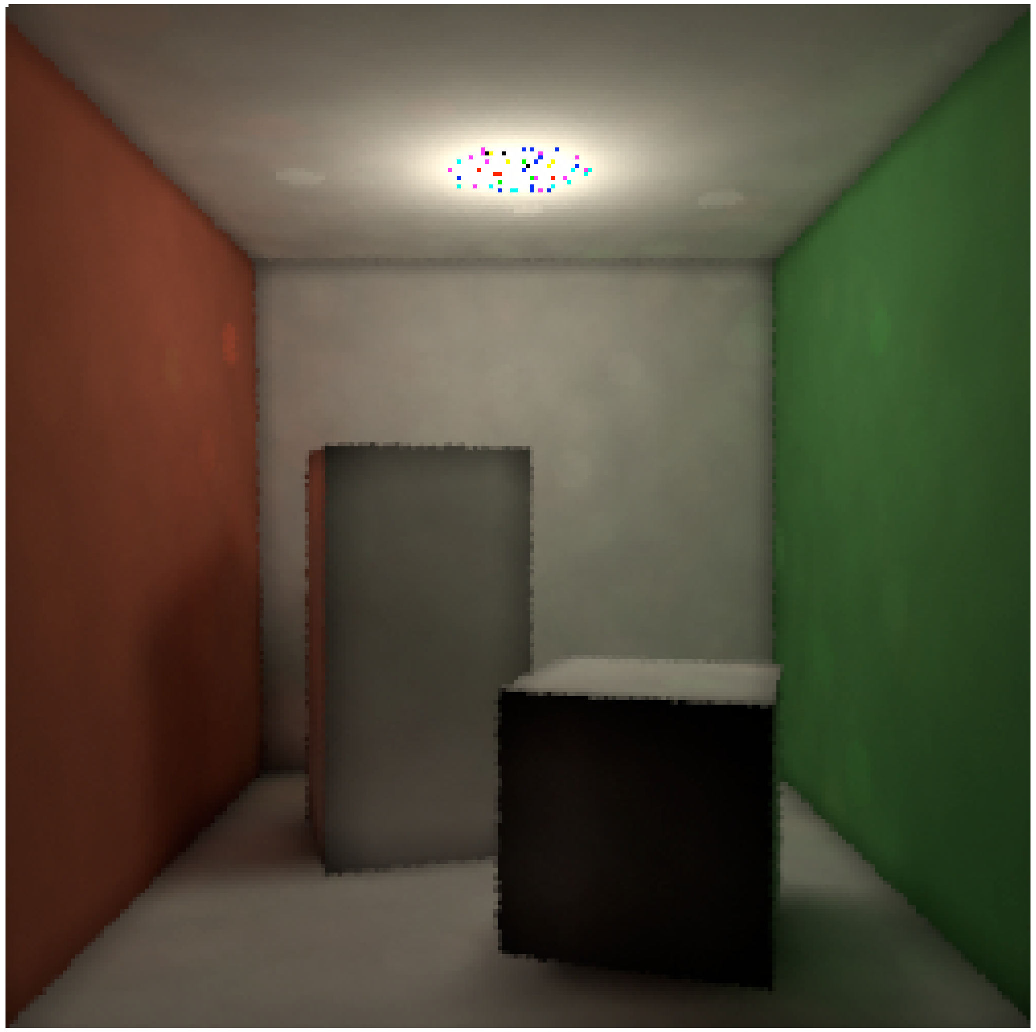
Photon Mapping

- 200k photons



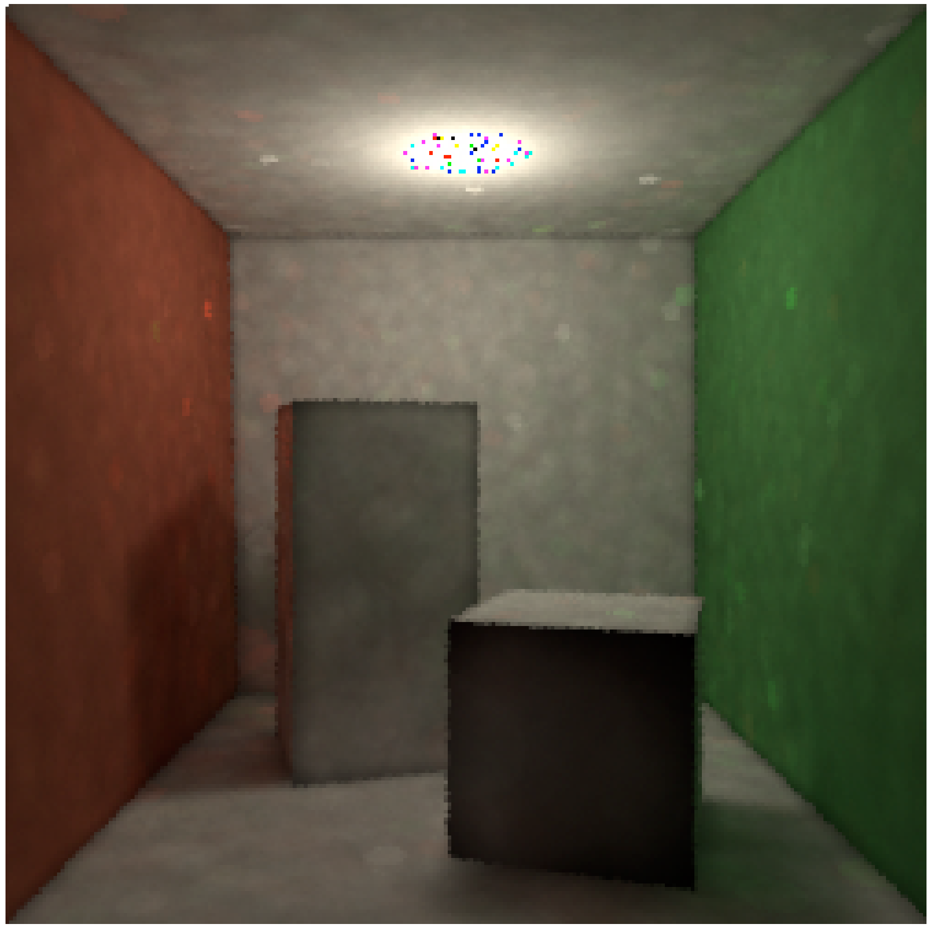
Photon Mapping

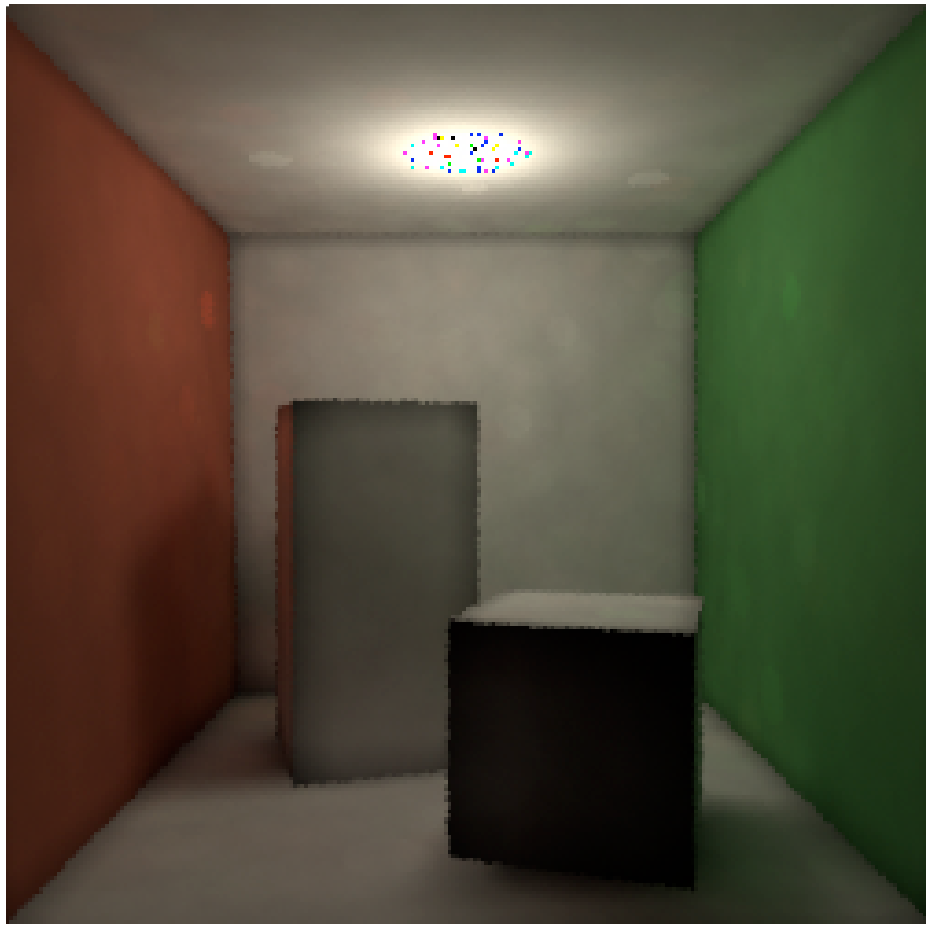
- 500k photons



Photon Mapping

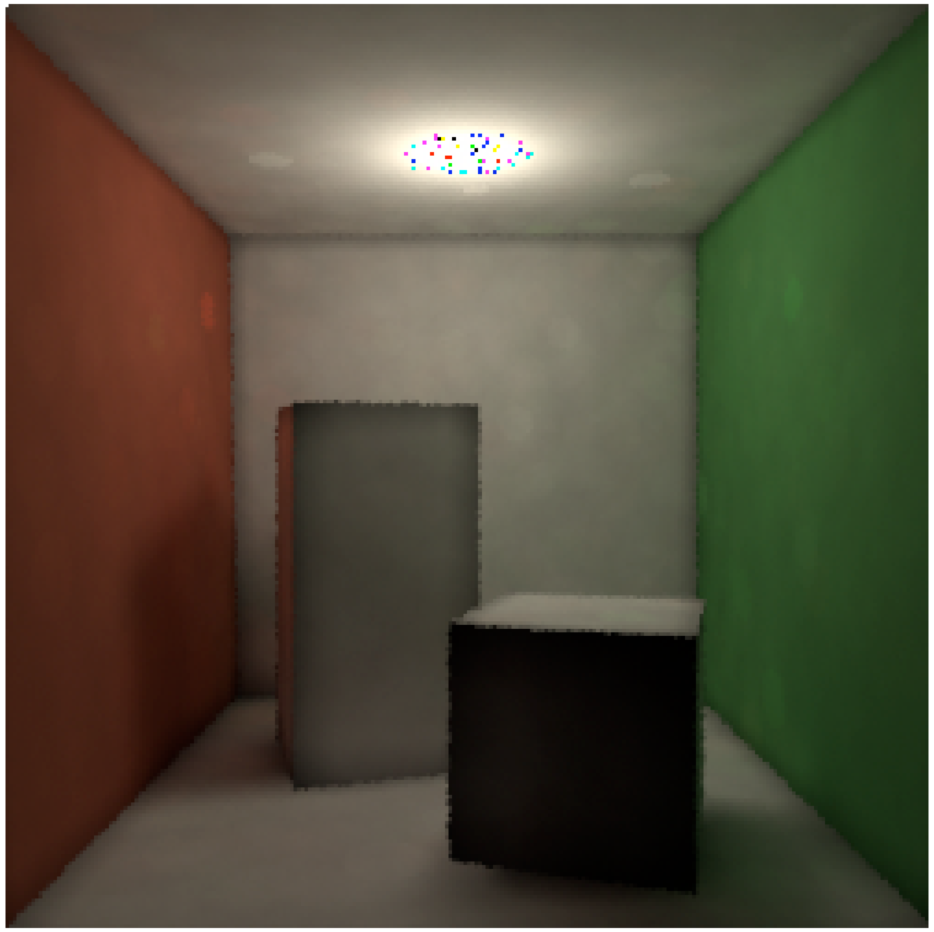
- 500k, use 125 vs use 500

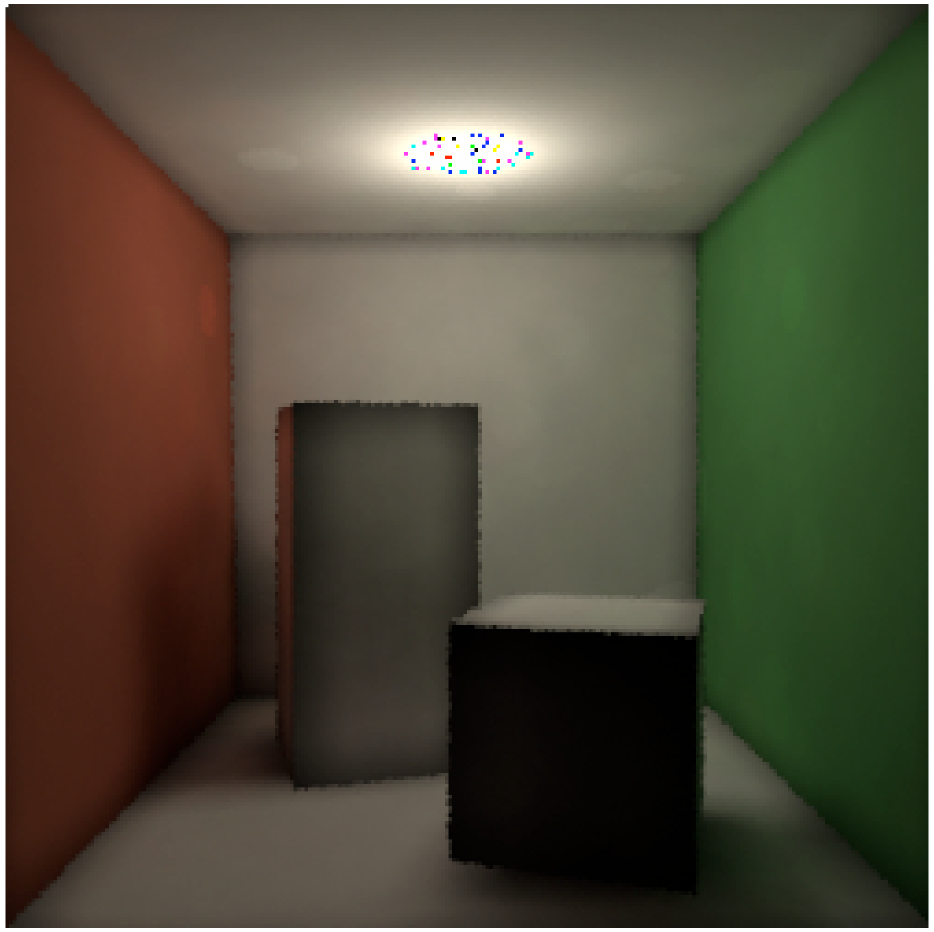




Photon Mapping

- 500k, use 500 vs use 1000





Improved Photon Mapping

- Partitioning
  - BRDF into diffuse + specular
  - Illumination into direct, indirect, and caustic
- Each term can be estimated two ways:
  - Accurately: Unbiased MC
  - Efficiently: Photon map lookup

Photon Mapping

- Biased method: using photons from nearby points to estimate illumination introduces error
- Consistent: error tends to decrease as number of photons increases