

# Stochastic Ray-Tracing

basic path tracing and photon mapping

Haben Michael, Statistics Department

August 10, 2016

# Outline

Introduction

Path tracing

Photon mapping

- ▶ The rendering equation is a physical model of the light at a point on a surface

$$L_o(x, \omega, \lambda, t) = L_e(x, \omega, \lambda, t) + \int_{\Omega} f_r(x, \omega', \omega, \lambda, t) L_i(x, \omega', \lambda, t) (-\omega' \cdot n) d\omega'$$

- ▶ This is hard to solve

- ▶ approximate solutions are the goal of realistic rendering programs
- ▶ two stochastic approaches:
  - ▶ path tracing
  - ▶ photon mapping

# Outline

Introduction

**Path tracing**

Photon mapping

- Issue a ray as in a recursive ray-tracer but randomly choose new direction to reflect when recursing

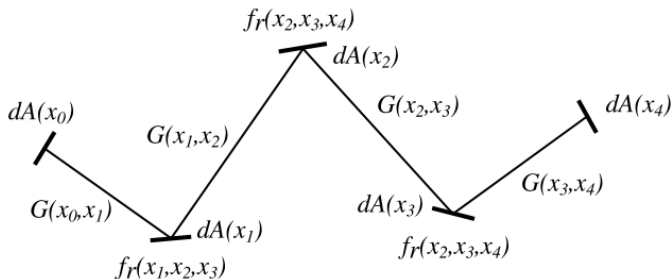


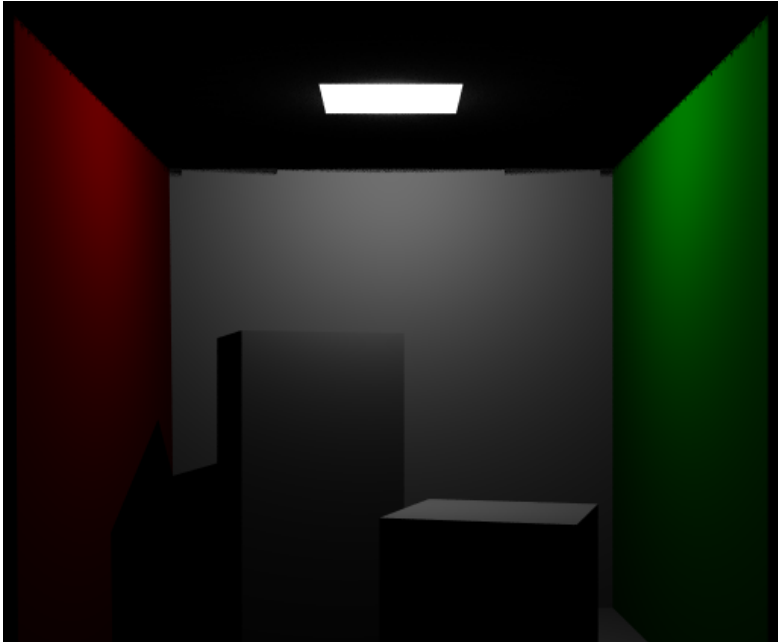
Figure 7.3: A path from point  $\vec{x}_1$  to  $\vec{x}_5$ .

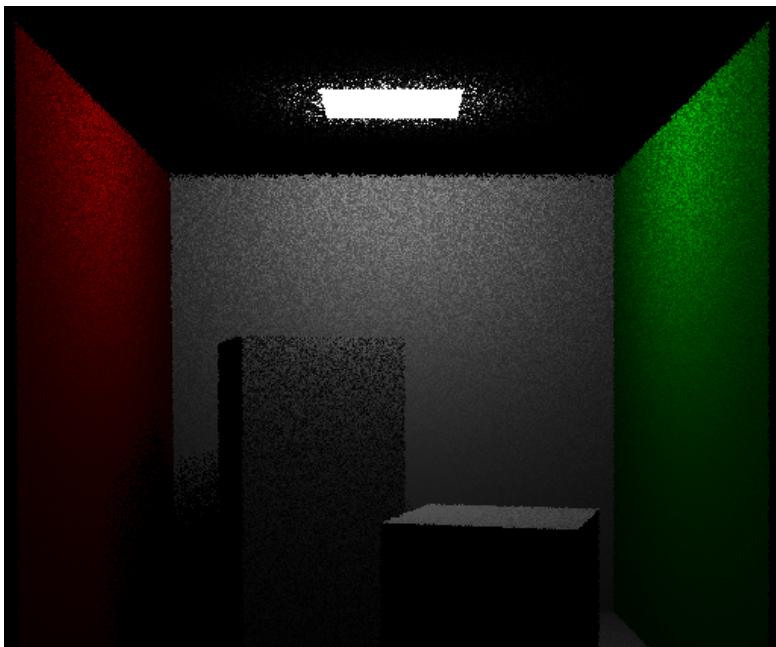
- i.e., don't just choose specular or refractive directions

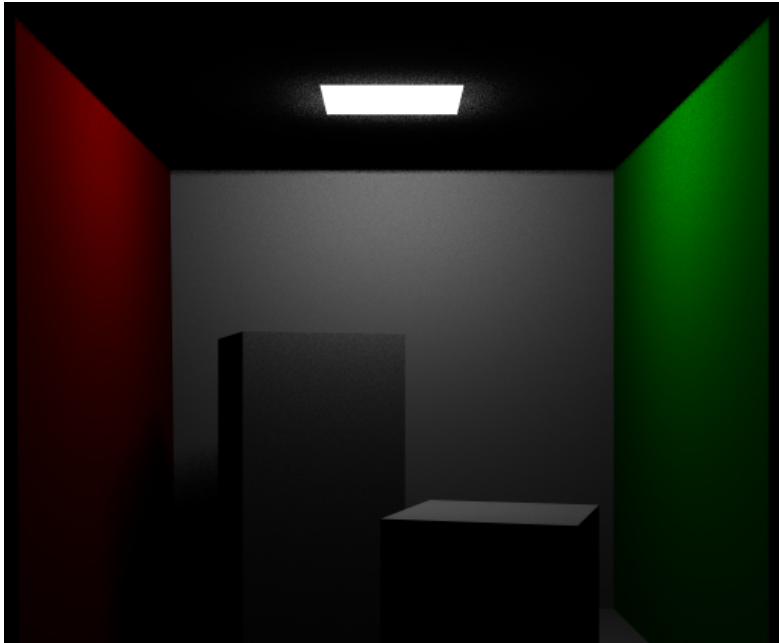
- ▶ advantage: “global illumination”
- ▶ treat light emitted from a lamp source and light emitted from a surface uniformly
- ▶ our approximation from hw4: ambient term



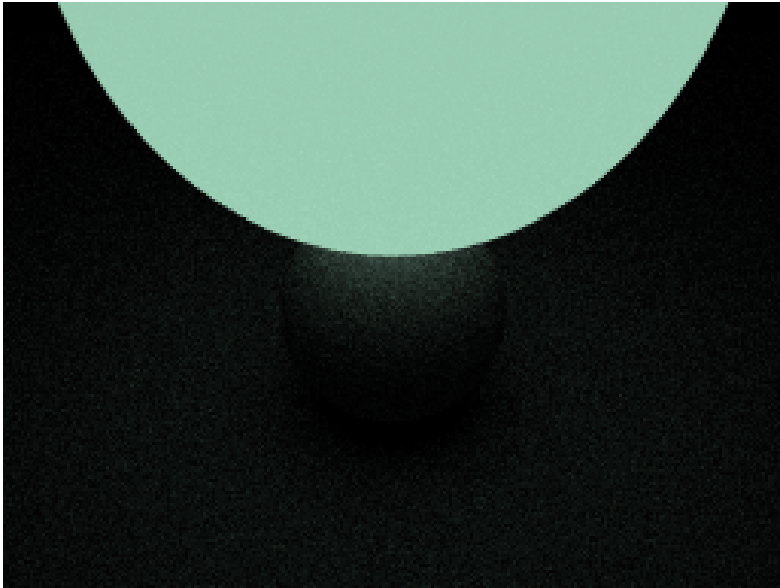




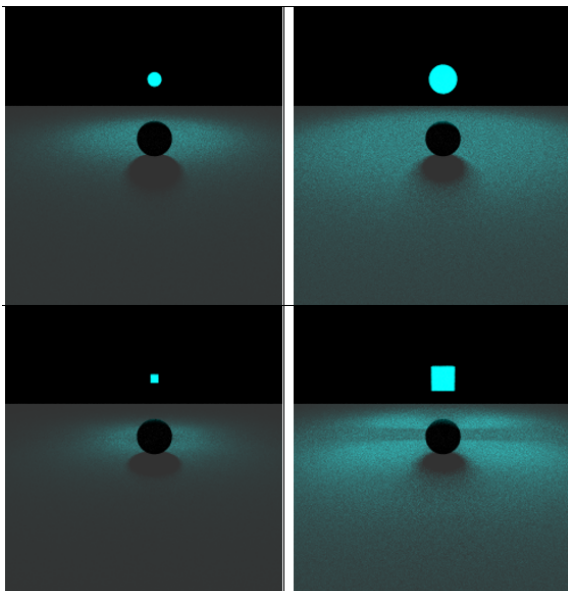




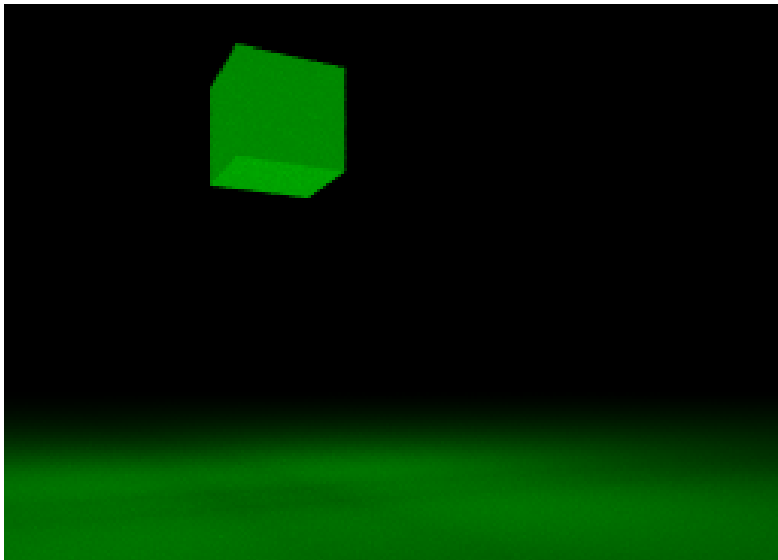
- ▶ Must be judicious, can't sample all directions



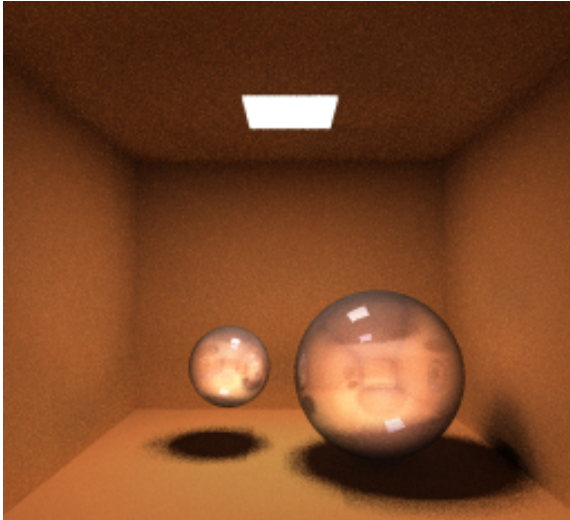
- ▶ accelerate convergence by sampling directly from light sources



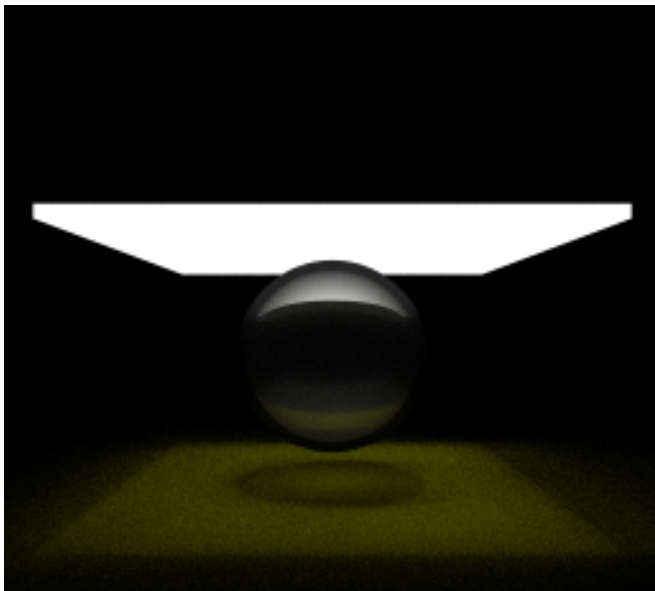
- ▶ now must figure out the proper pdf to sample from light sources



- ▶ problem with this approach: transmissive medium
- ▶ one solution is to sample in the forward direction out of light sources



- ▶ another weakness of path tracing is caustics





# Outline

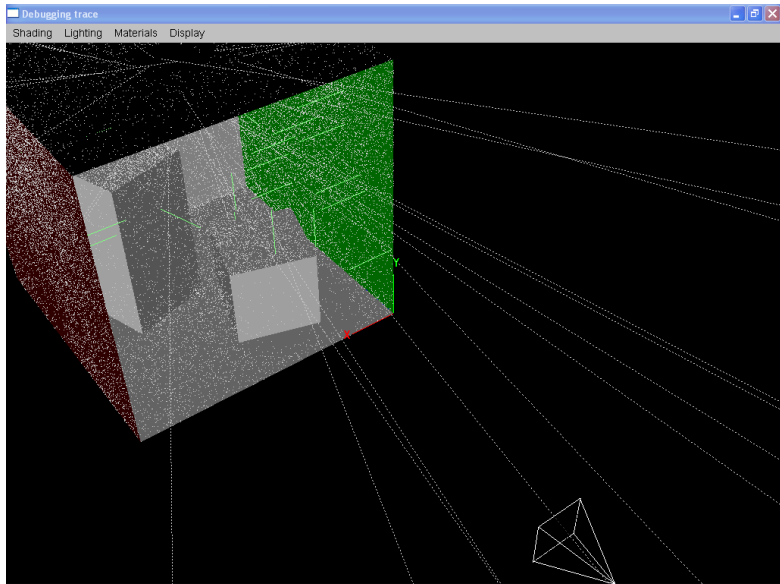
Introduction

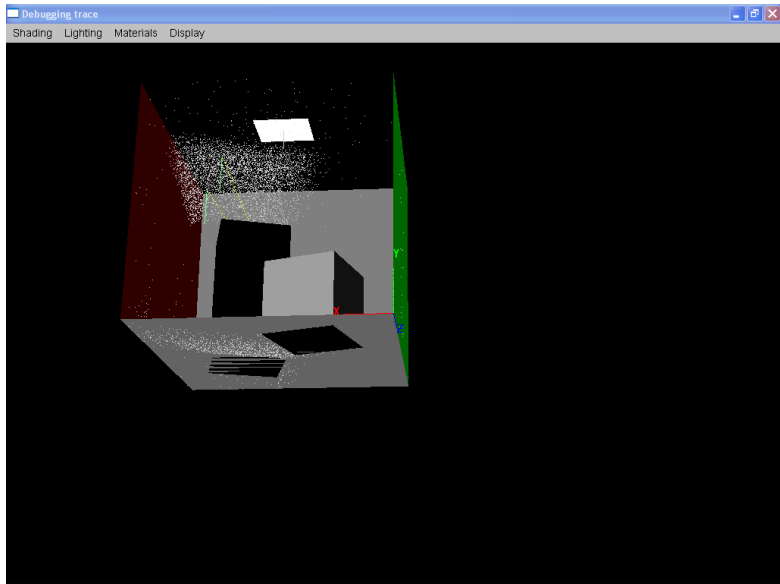
Path tracing

Photon mapping

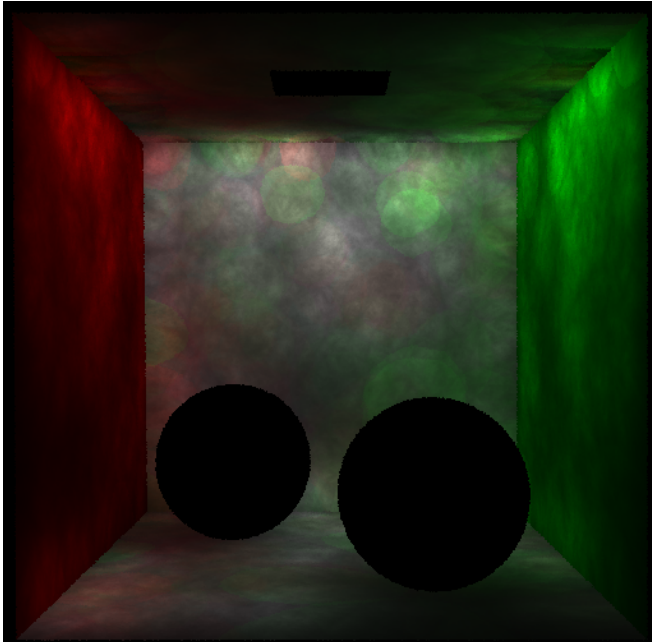
- ▶ issue a “photon” originating from a random point on the light surface in a random direction
- ▶ store intensity of the photon at each point of intersection, build up a “photon map”
- ▶ a kd-tree, decoupled from scene geometry

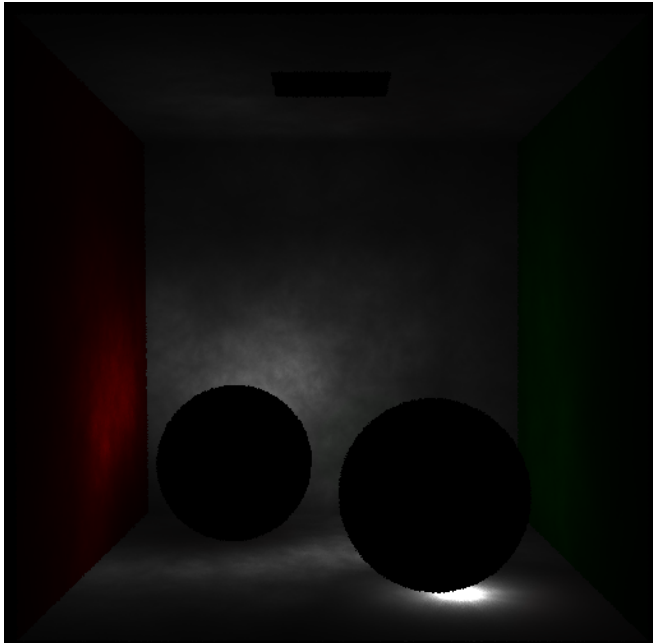
- ▶ actually build 2 maps
  - ▶ global map
  - ▶ caustic map—when the ray has been reflected/transmitted



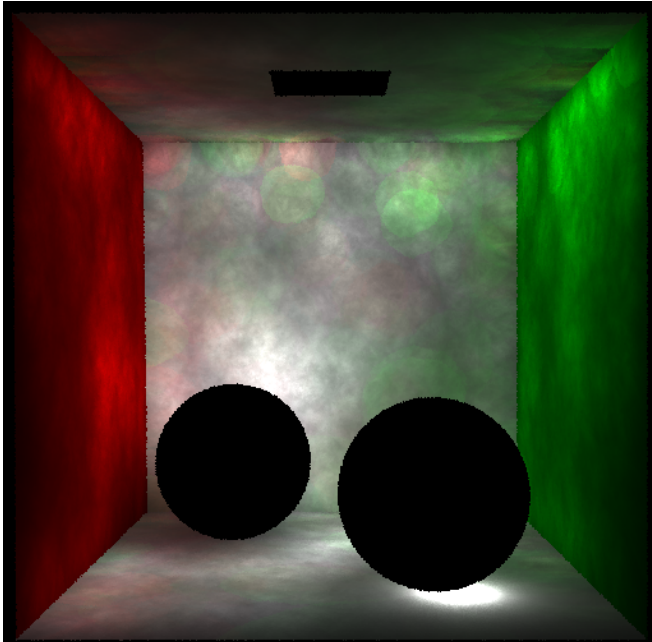


estimate radiance at a point using k-nearest neighbors with the photon maps









render scene by issuing a ray as in a standard ray tracer, but use the photon map estimate at the point as the radiance estimate

