CS482: Acceleration Methods for MC Ray Tracing:

Sung-Eui Yoon
(윤성의)

http://sglab.kaist.ac.kr/~sungeui/IC

Class Objectives

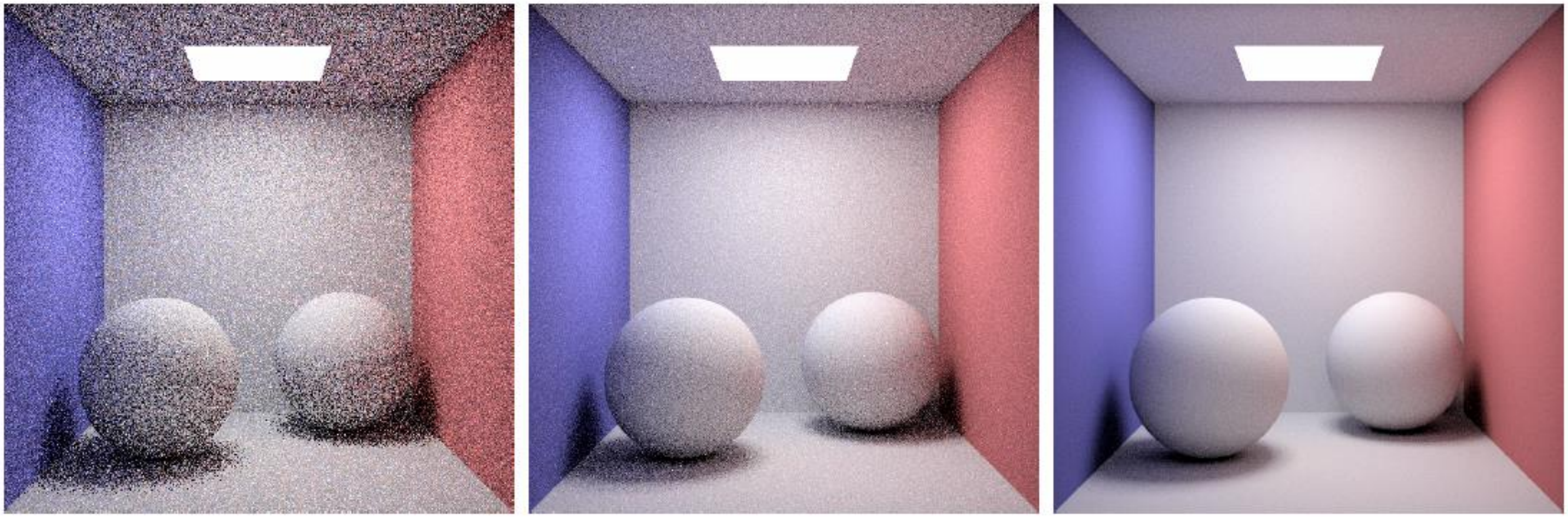
- **Discuss acceleration methods for GI**
  - **Importance sampling, bidirectional path tracing, and Metropolis**
- **Study biased techniques**
  - **Irradiance caching and photon mapping**

- **Last time:**
  - **Path tracing, a basic structure of Monte Carlo ray tracing including Russian roulette**

Algorithm so far: Path tracing

- **Shoot primary rays through each pixel**
- **Shoot indirect rays, sampled over hemisphere**
  - **Path tracing shoots only 1 indirect ray**
- **Terminate recursion using Russian Roulette**

Path Tracing



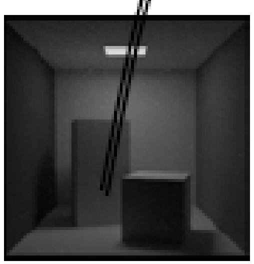
1 spp (samples per pixel) | 4 spp | 16 spp

- **Pixel sampling + light source sampling folded into one method**

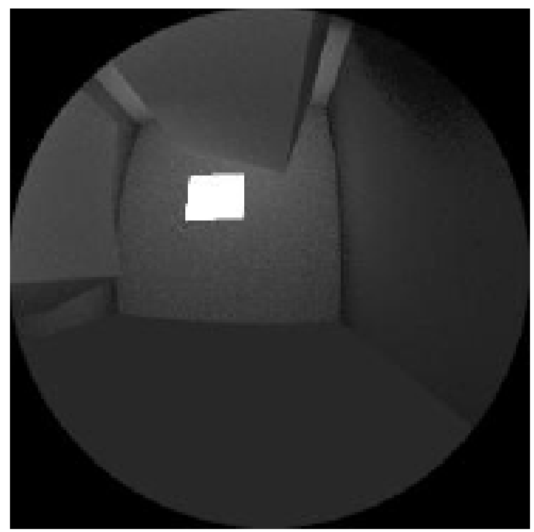
Importance Sampling

$$L(x \rightarrow \Theta) = L_e(x \rightarrow \Theta) + \int_{\Omega_x} f_r(\Psi \leftrightarrow \Theta) \cdot L(x \leftarrow \Psi) \cdot \cos(\Psi, n_x) \cdot d\omega_\Psi$$

Radiance from light sources + radiance from other surfaces



$$= L_e + \int_{\Omega_x}$$

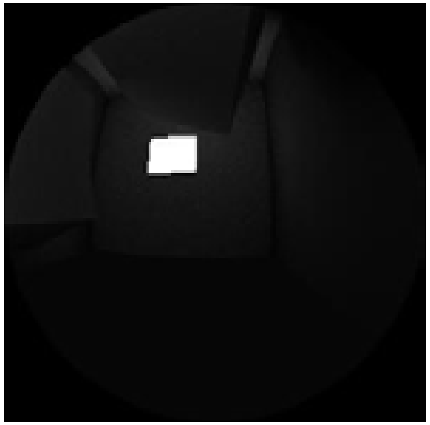


$$\cdot f_r \cdot \cos$$

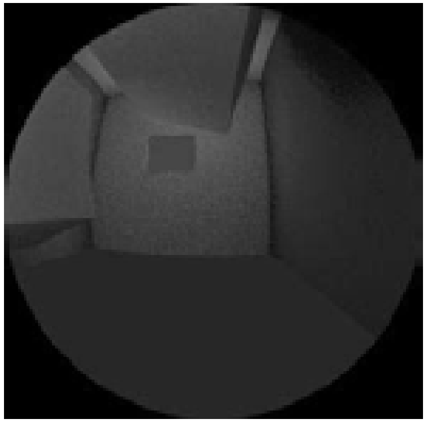
Importance Sampling

$$L(x \rightarrow \Theta) = L_e + L_{direct} + L_{indirect}$$

$$= L_e + \int_{\Omega_x}$$



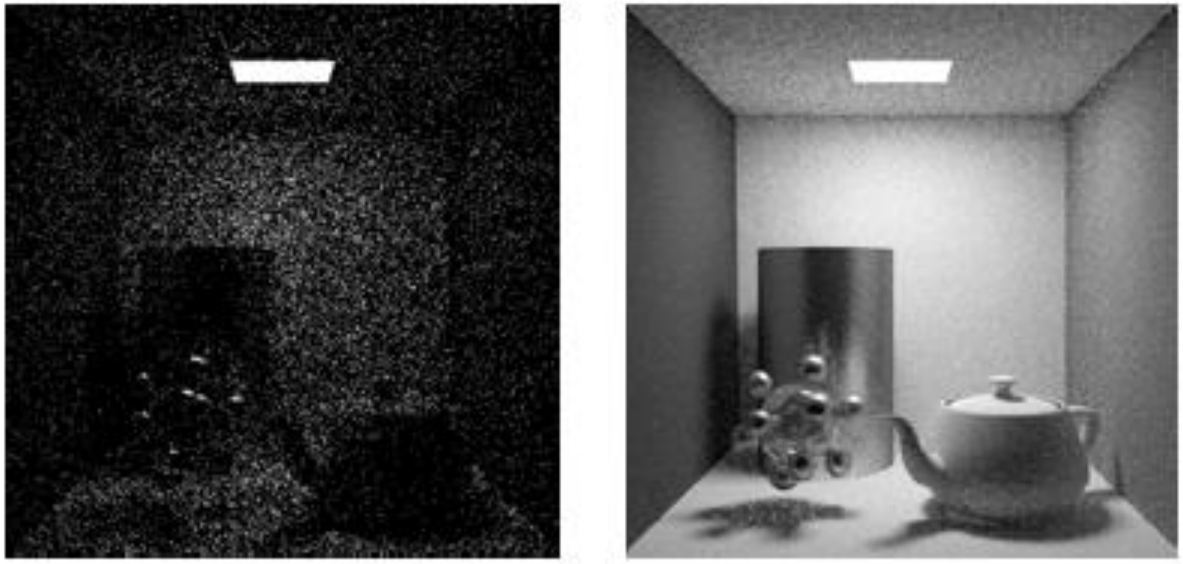
$$\cdot f_r \cdot \cos + \int_{\Omega_x}$$



$$\cdot f_r \cdot \cos$$

- So … sample direct and indirect with separate MC integration

Comparison



From kavita's slides

- With and without considering direct illumination
  - 16 samples / pixel

Rays per pixel

1 sample/
pixel

4 samples/
pixel

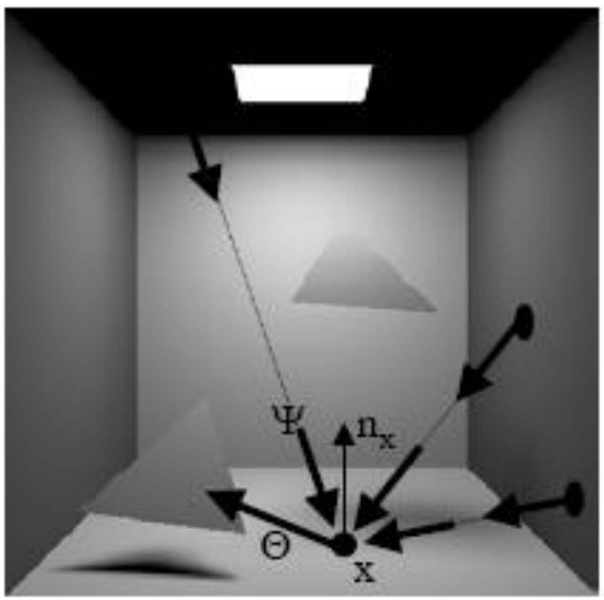
16 samples/
pixel

256 samples/
pixel

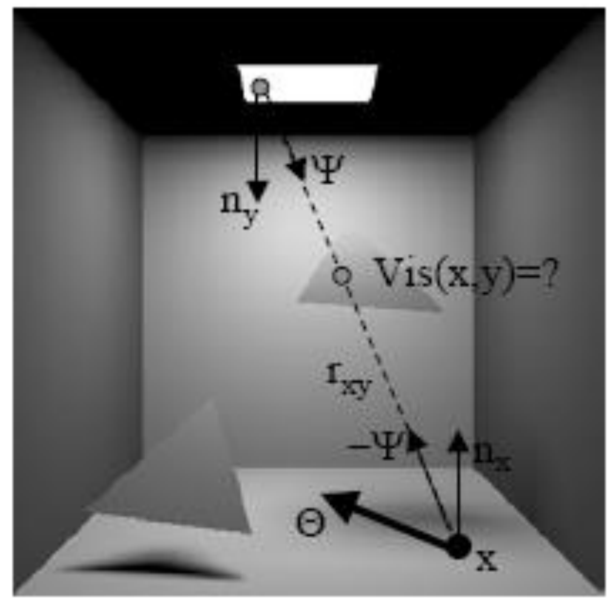
Direct Illumination

$$L(x \to \Theta) = \int_{A_{source}} f_r(x, -\Psi \leftrightarrow \Theta) \cdot L(y \to \Psi) \cdot G(x,y) \cdot dA_y$$

$$G(x,y) = \frac{\cos(n_x, \Theta)\cos(n_y, \Psi) Vis(x,y)}{r_{xy}^2}$$



hemisphere integration



area integration

Estimator for direct lighting

- Pick a point on the light's surface with pdf

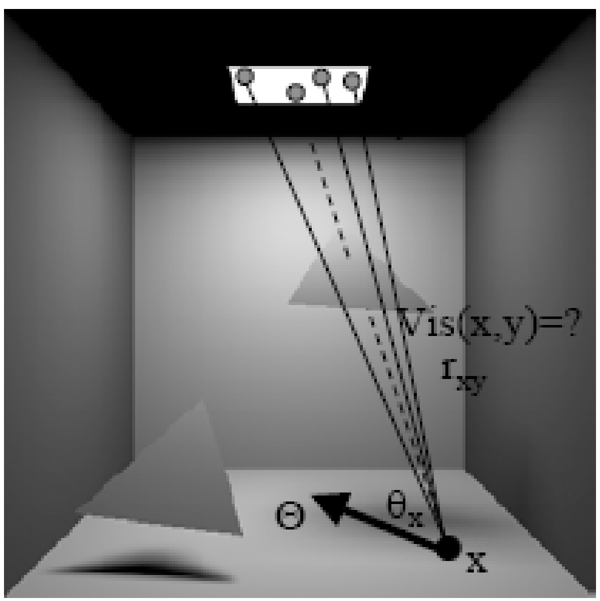
$$p(y)$$

- For N samples, direct light at point x is:

$$E(x) = \frac{1}{N}\sum_{i=1}^{N}\frac{f_r L_{source}\frac{\cos\theta_x \cos\theta_{\bar{y}_i}}{r^2_{x\bar{y}_i}} Vis(x,\bar{y}_i)}{p(\bar{y}_i)}$$

Generating direct paths

- Pick surface points y_i on light source
- Evaluate direct illumination integral



$$\langle L(x \to \Theta) \rangle = \frac{1}{N} \sum_{i=1}^{N} \frac{f_r(\ldots)L(\ldots)G(x, y_i)}{p(y_i)}$$

PDF for sampling light

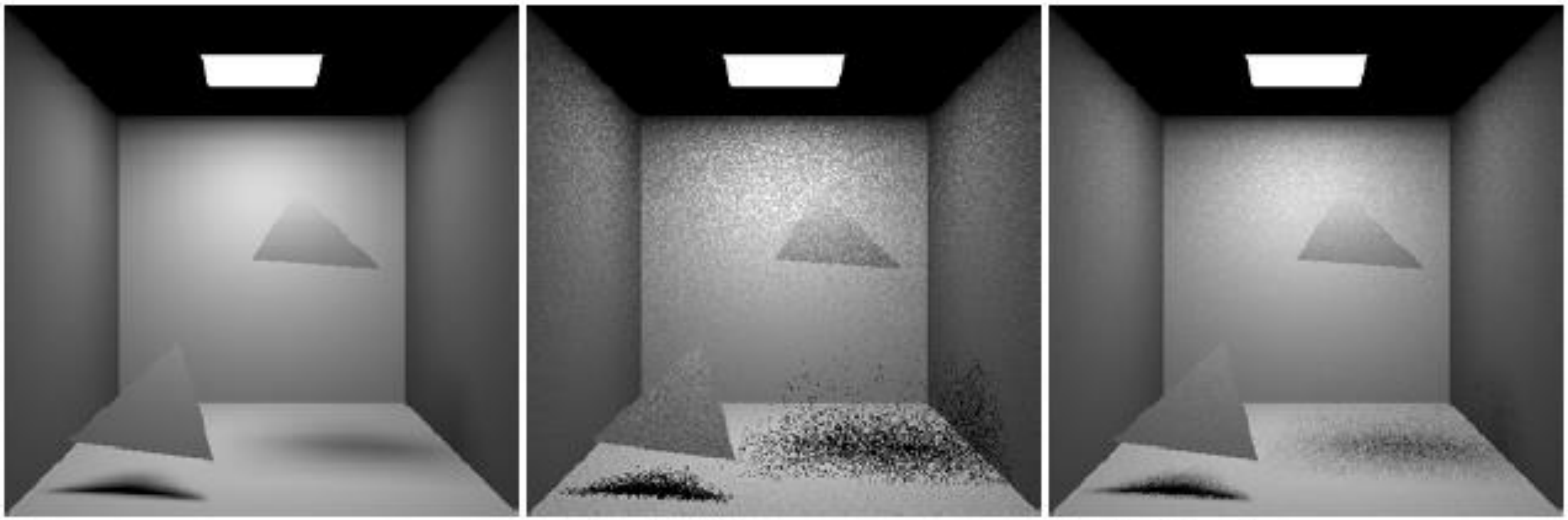
- Uniform
$$p(y) = \frac{1}{Area_{source}}$$

- Pick a point uniformly over light's area
  - Can stratify samples

- Estimator:

$$E(x) = \frac{Area_{source}}{N} \sum_{i=1}^{N} f_r L_{source} \frac{\cos\theta_x \cos\theta_{\bar{y}_i}}{r_{x\bar{y}_i}^2} Vis(x, \bar{y}_i)$$

More points ...

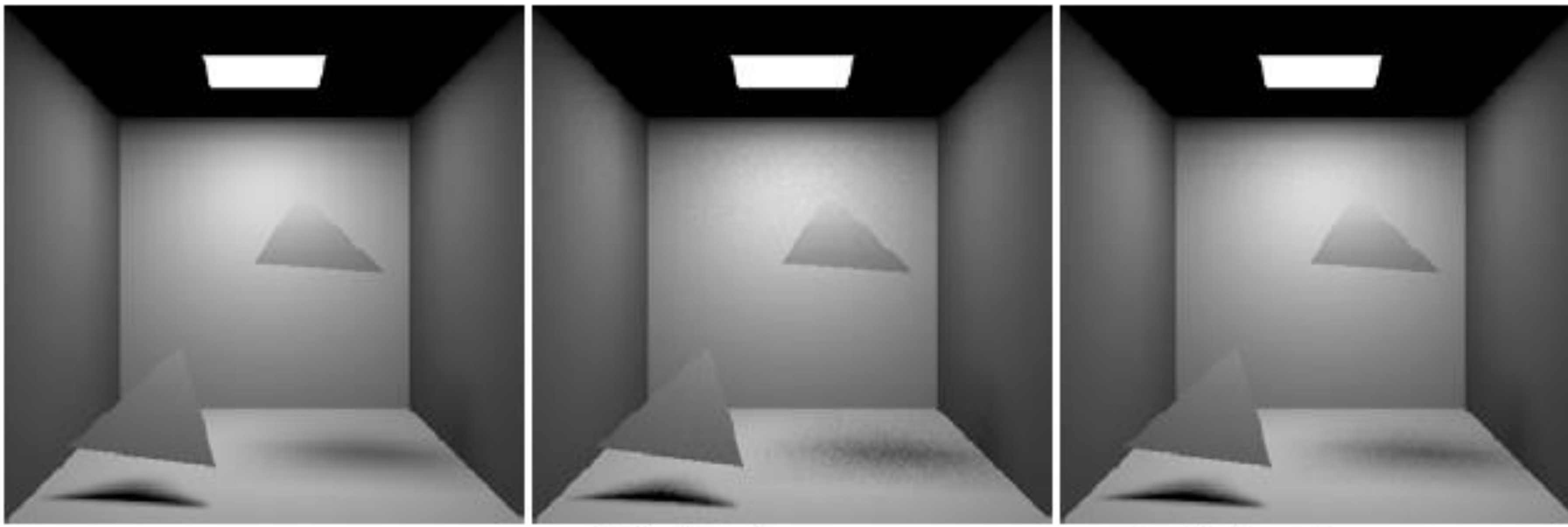


1 shadow ray

9 shadow rays

$$E(x) = \frac{Area_{source}}{N} \sum_{i=1}^{N} f_r L_{source} \frac{\cos\theta_x \cos\theta_{\bar{y}_i}}{r^2_{x\bar{y}_i}} Vis(x, \bar{y}_i)$$

Even more points ...

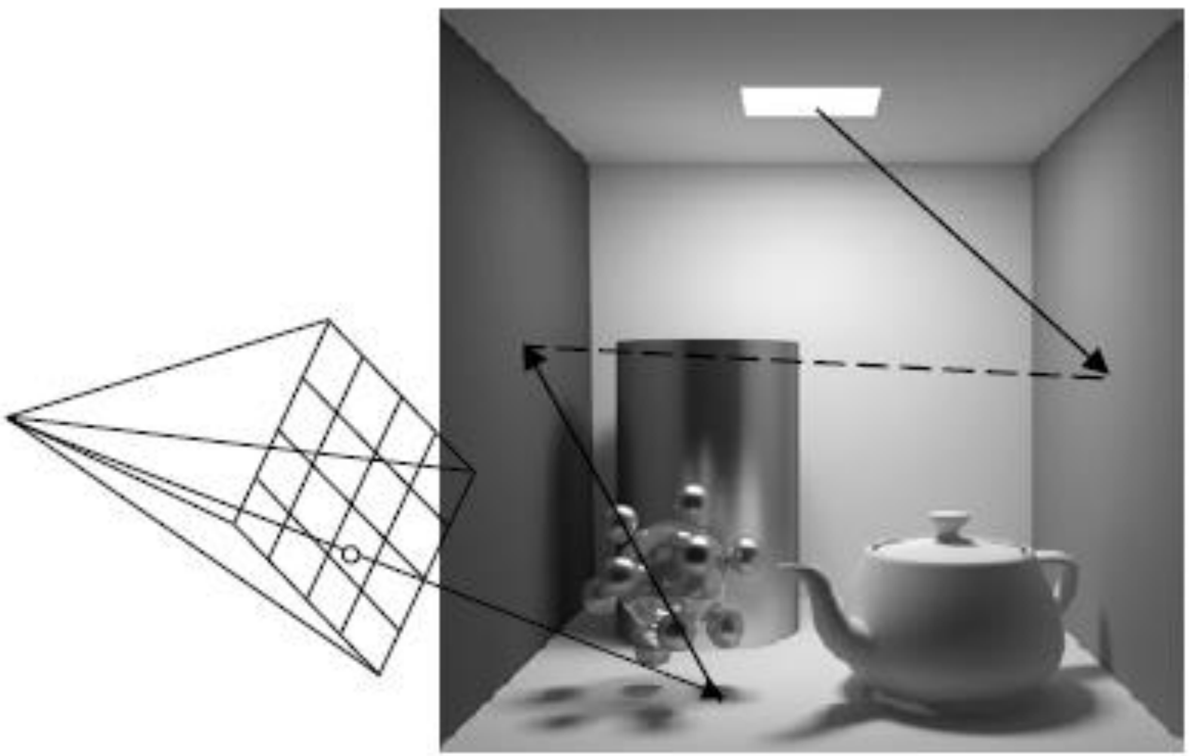


36 shadow rays 100 shadow rays

$$E(x) = \frac{Area_{source}}{N} \sum_{i=1}^{N} f_r L_{source} \frac{\cos\theta_x \cos\theta_{\bar{y}_i}}{r_{x\bar{y}_i}^2} Vis(x, \bar{y}_i)$$

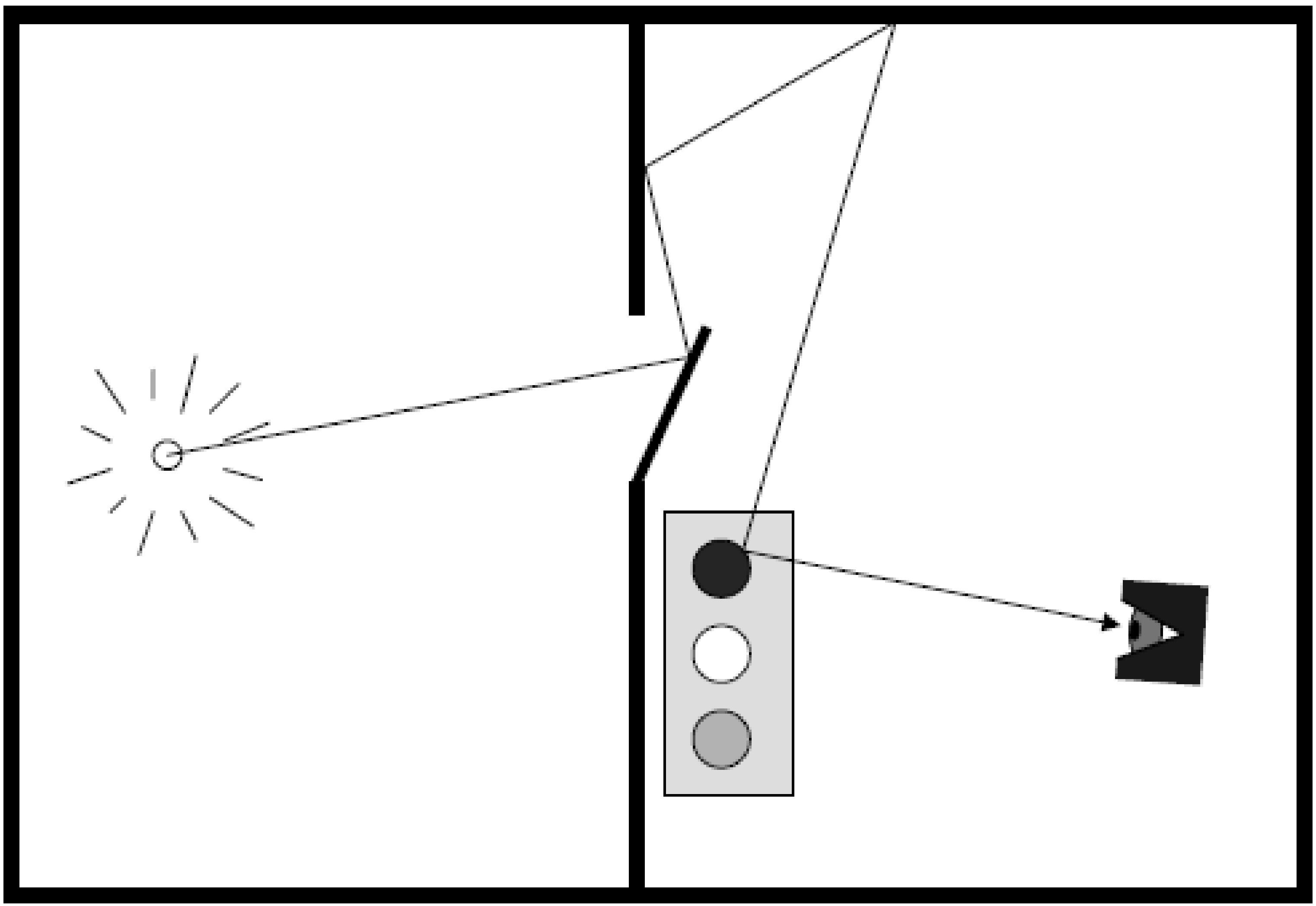
Bidirectional Path Tracing

- Or paths generated from both camera and source at the same time ...!

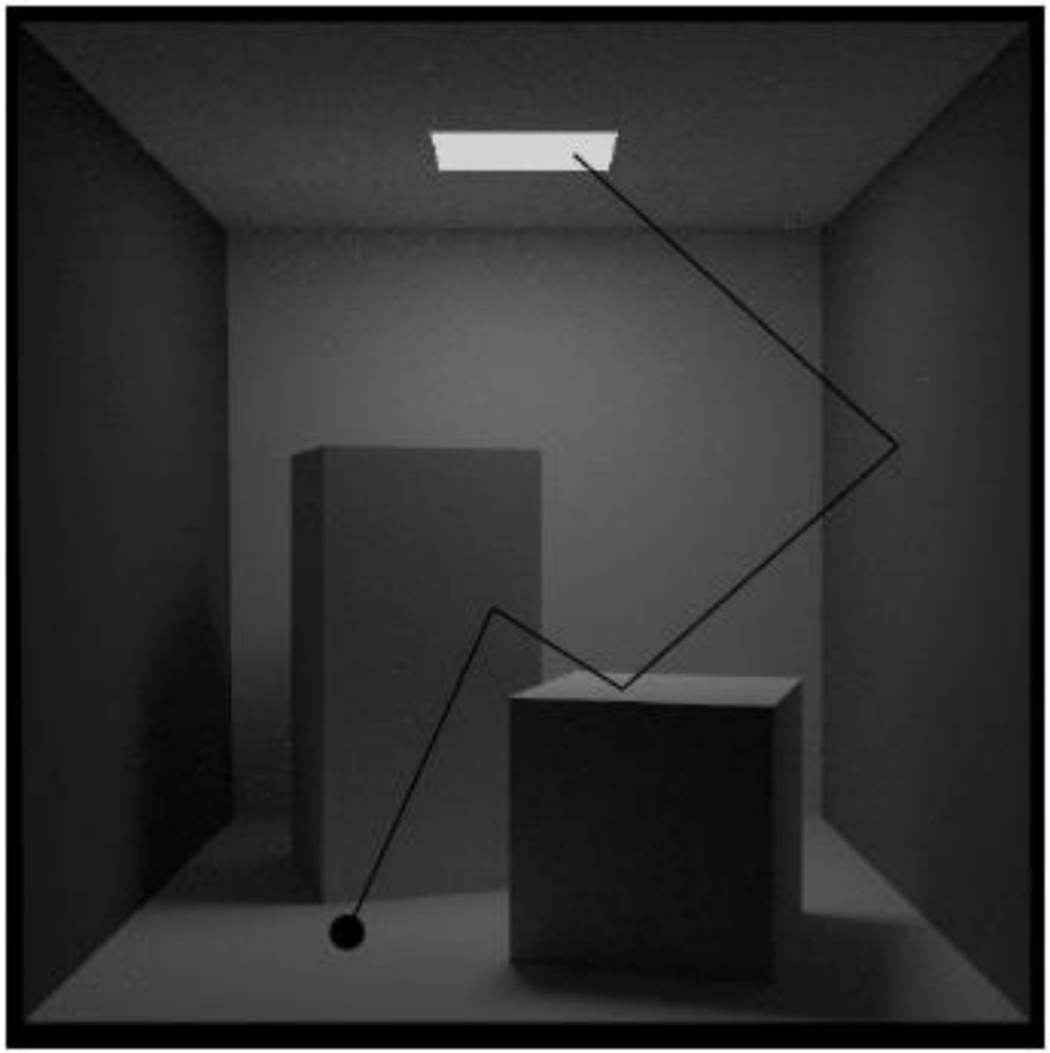


- Connect endpoints to compute final contribution

Metropolis

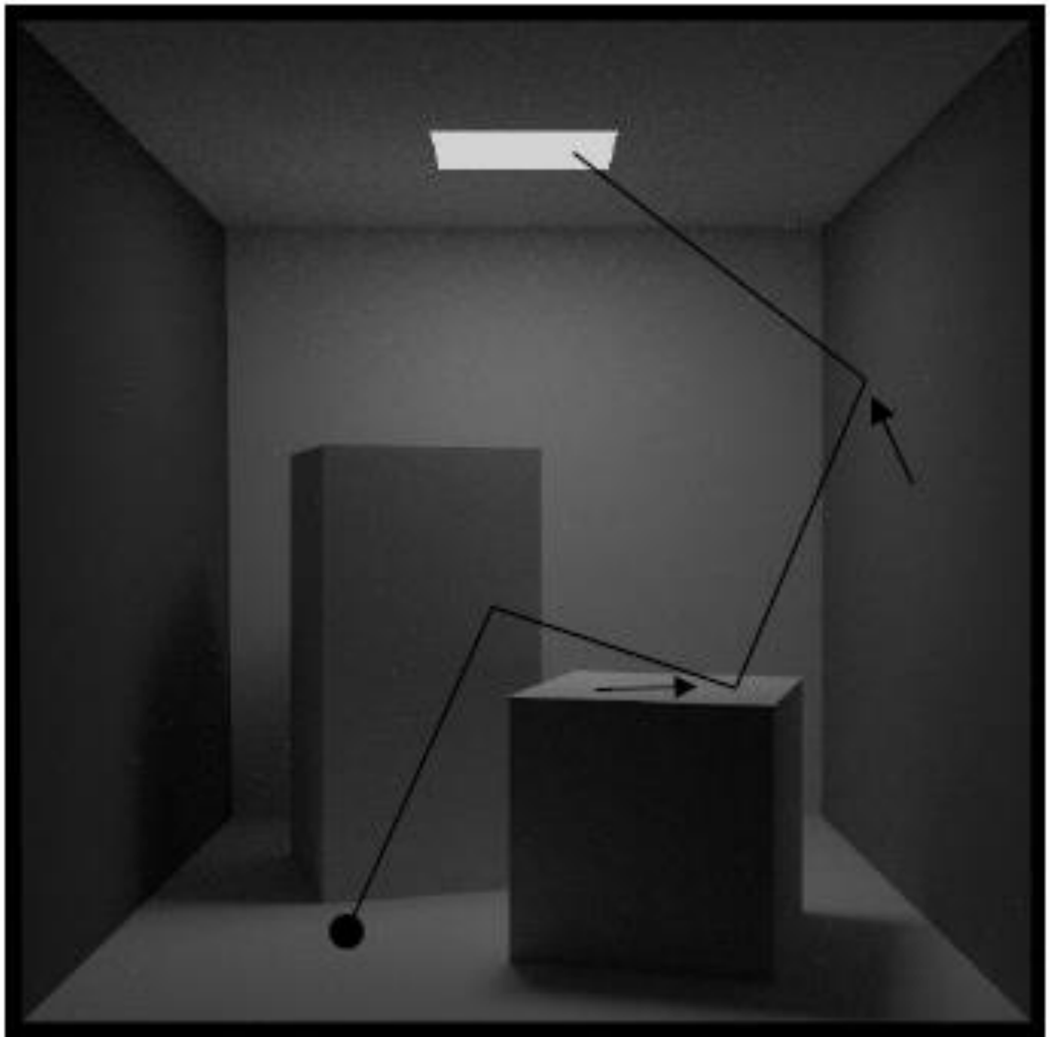


Metropolis



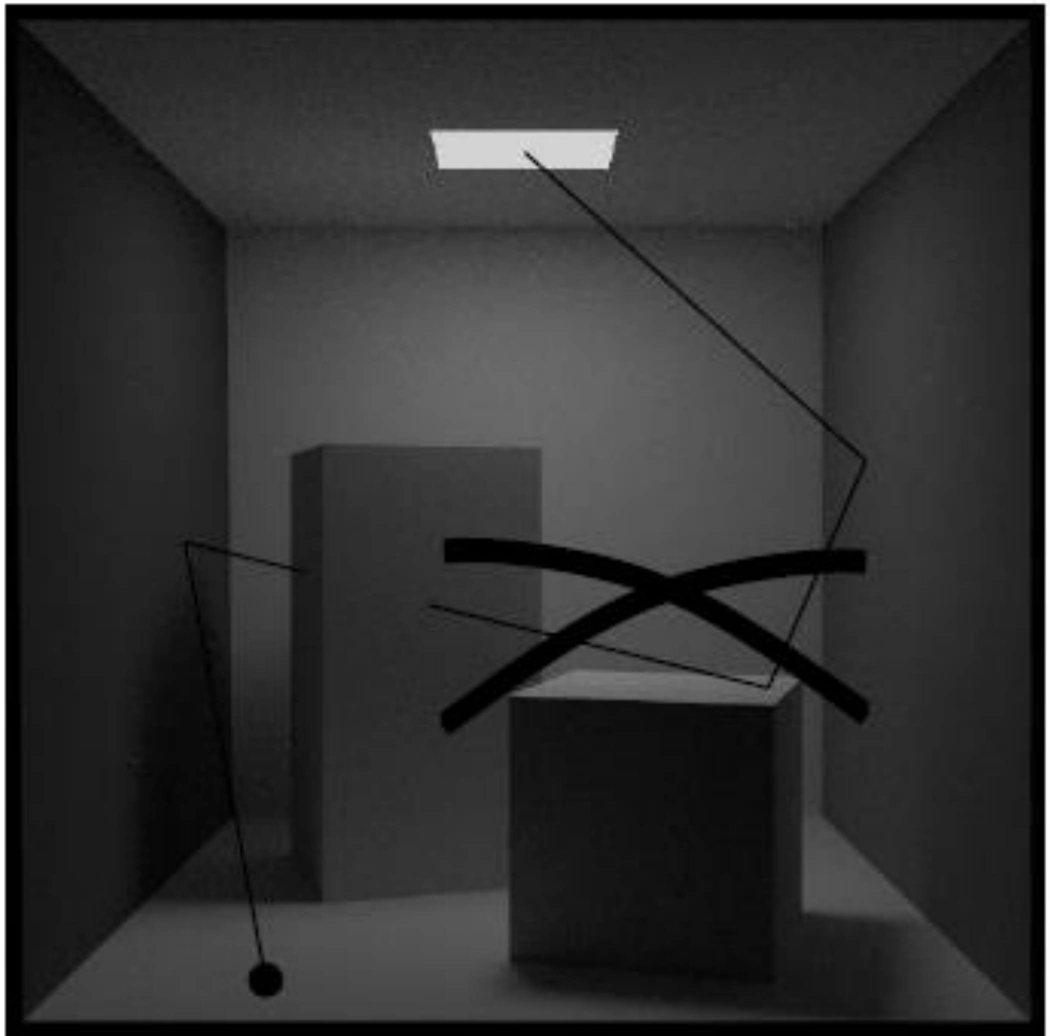
valid path

Metropolis



small perturbations

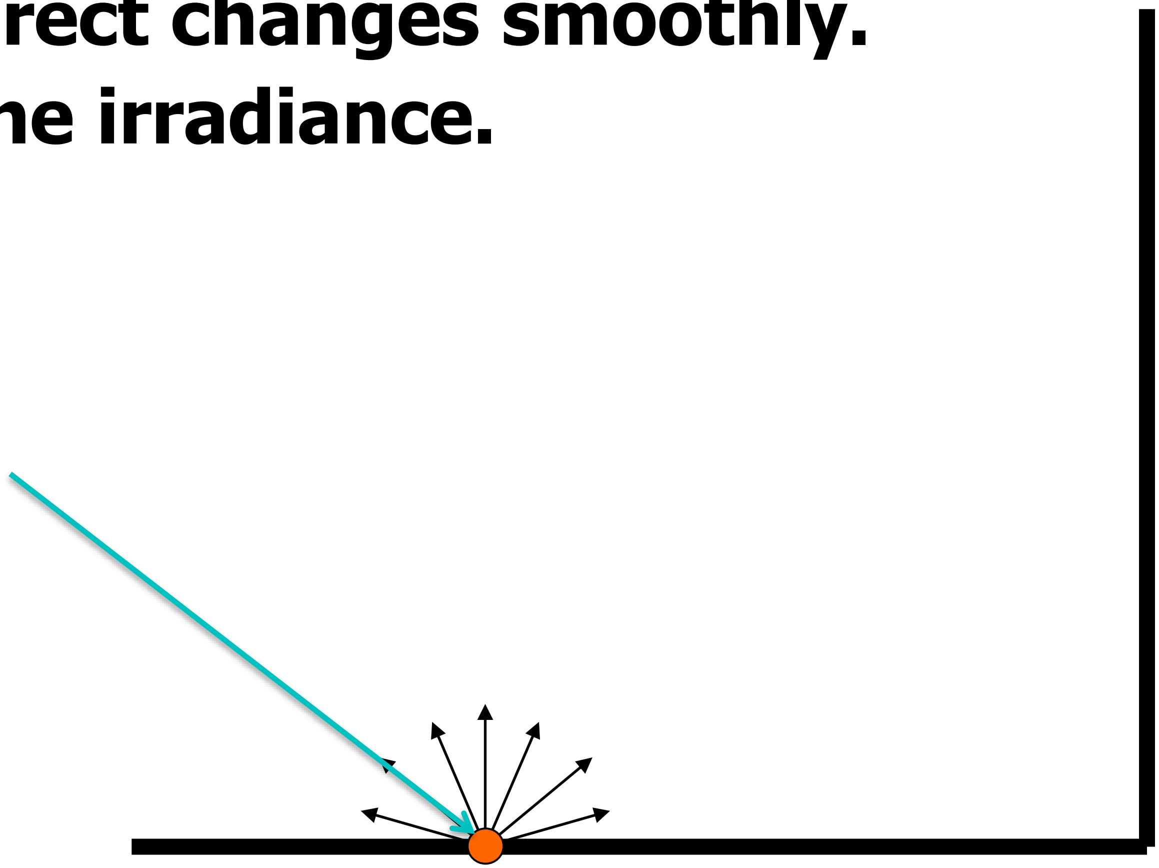
Metropolis



Accept mutations based on energy transport

Biased Methods: Irradiance Caching

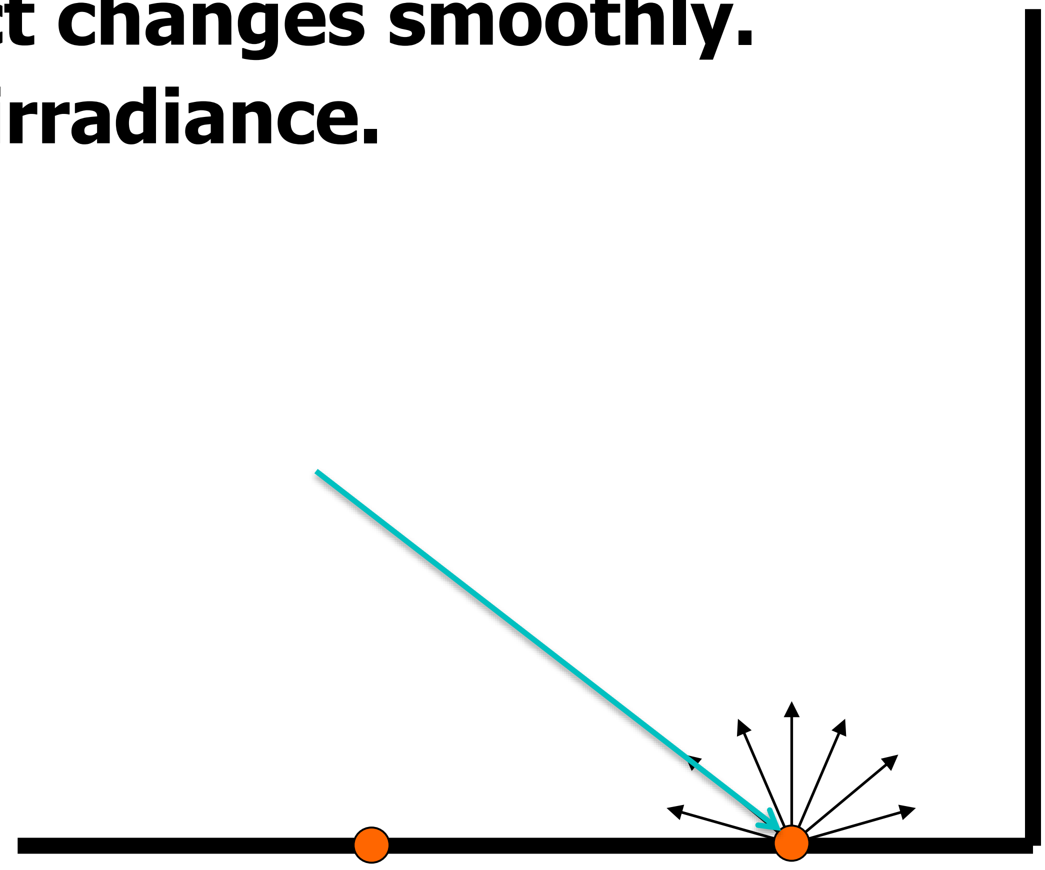
- Indirect changes smoothly.
- Cache irradiance.



From Wang's slides

Irradiance Caching

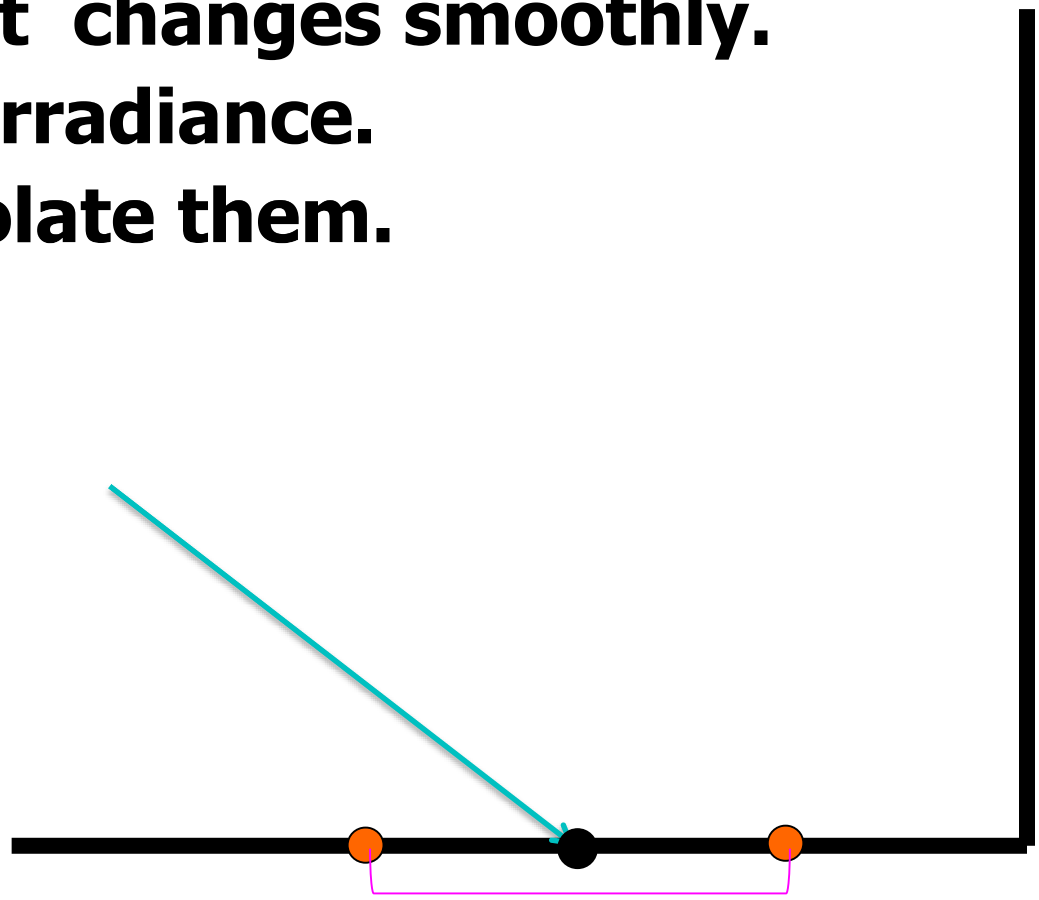
- Indirect changes smoothly.
- Cache irradiance.



From Wang's slides

Irradiance Caching

- Indirect changes smoothly.
- Cache irradiance.
- Interpolate them.

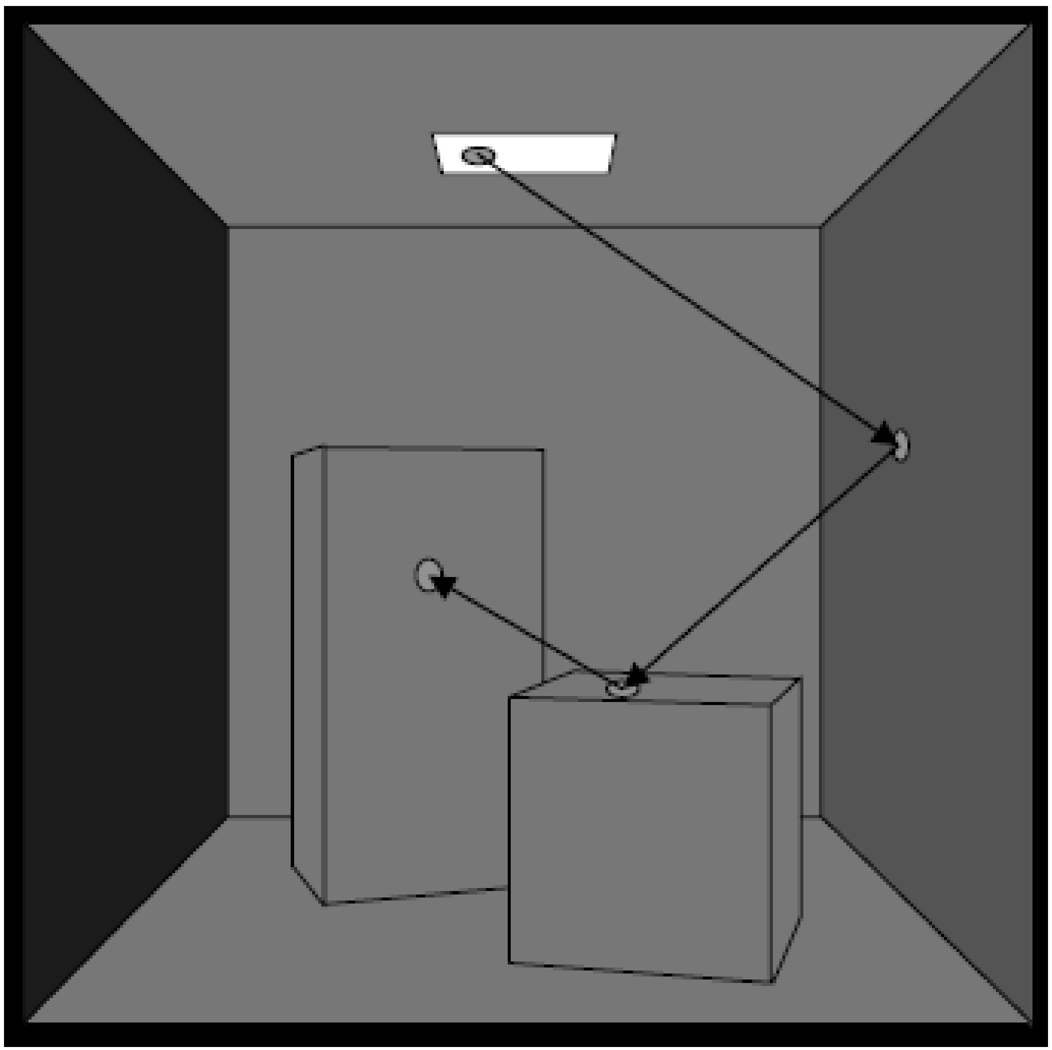


From Wang's slides

Biased Method: Photon Mapping

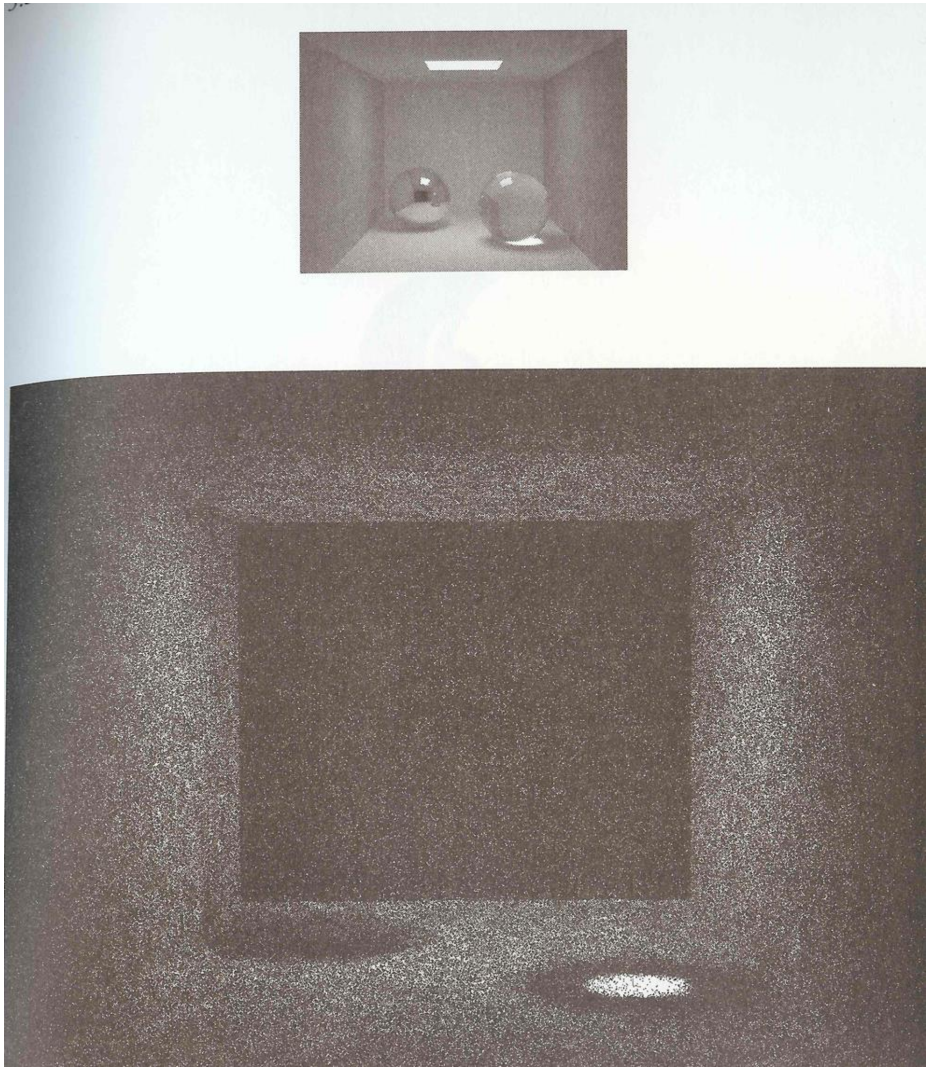
- 2 passes:
  - Shoot “photons” (light-rays) and record any hit-points
  - Shoot viewing rays and collect information from stored photons

Pass 1: shoot photons



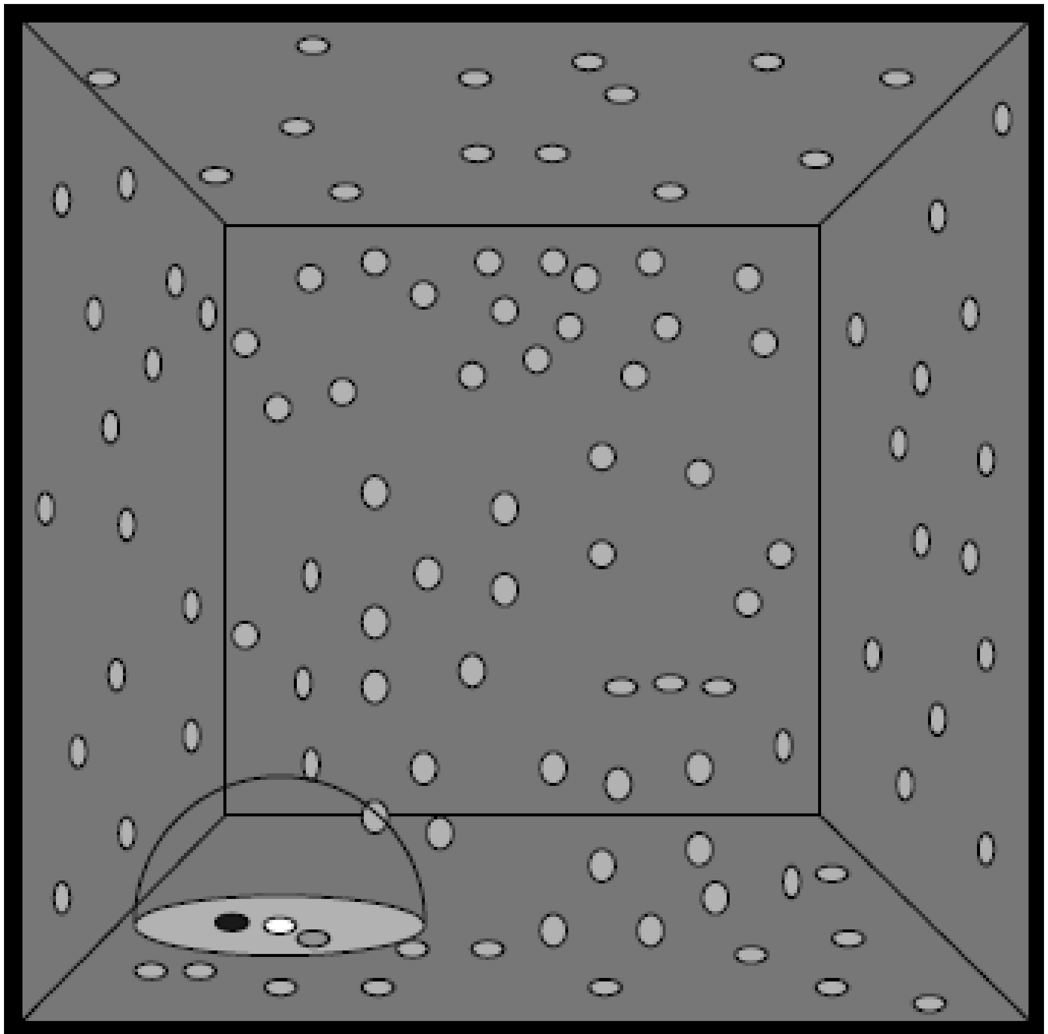
- Light path generated using MC techniques and Russian Roulette
- Store:
  - position
  - incoming direction
  - color
  - ...

Stored Photons



Generate a few hundreds of thousands of photons

Pass 2: viewing ray



- Search for N closest photons (+check normal)
- Assume these photons hit the point we're interested in
- Compute average radiance

Result

350K photons for the caustic map

Result

350K photons for the caustic map

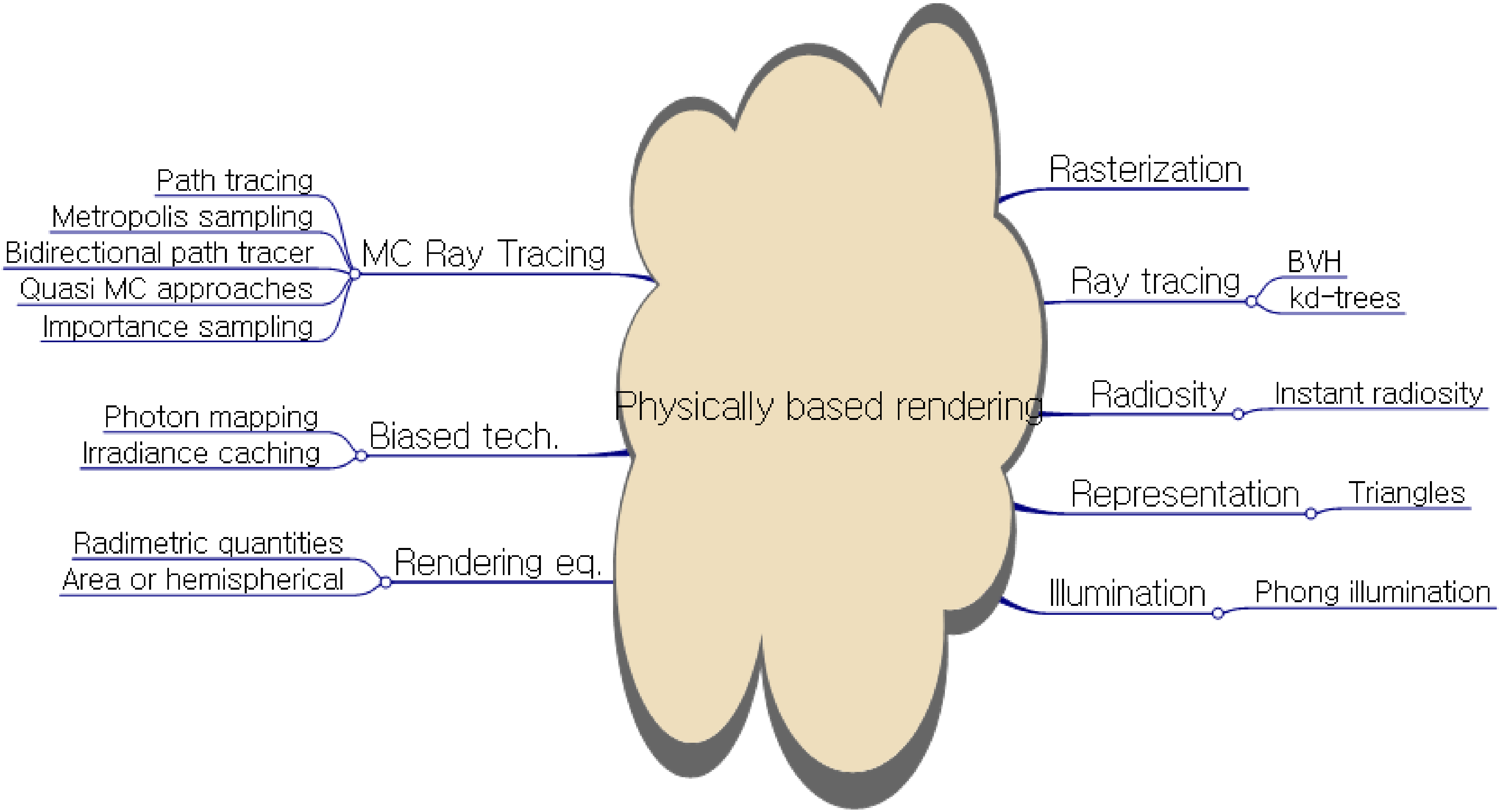
Class Objectives were:

- **Discuss acceleration methods for GI**
  - **Importance sampling, bidirectional path tracing, and metropolis**
- **Study biased techniques**
  - **Irradiance caching and photon mapping**

Summary

- Two basic building blocks
- Radiometry
- Rendering equation
- MC integration
- MC ray tracing
  - Unbiased methods
  - Biased methods

Summary


Physically based rendering
MC Ray Tracing
Path tracing
Metropolis sampling
Bidirectional path tracer
Quasi MC approaches
Importance sampling
Biased tech.
Photon mapping
Irradiance caching
Rendering eq.
Radimetric quantities
Area or hemispherical
Rasterization
Ray tracing
BVH
kd-trees
Radiosity
Instant radiosity
Representation
Triangles
Illumination
Phong illumination

Next Time…

- Recent techniques

Homework

- **Go over the next lecture slides before the class**
- **Watch 2 SIG/CVPR/ISMAR videos and submit your summaries every Mon. class**
  - **Just one paragraph for each summary**
  - **Any top-tier conf (e.g., ICRA) is okay**

Example:

Title: XXX XXXX XXXX

Abstract: this video is about accelerating the performance of ray tracing. To achieve its goal, they design a new technique for reordering rays, since by doing so, they can improve the ray coherence and thus improve the overall performance.