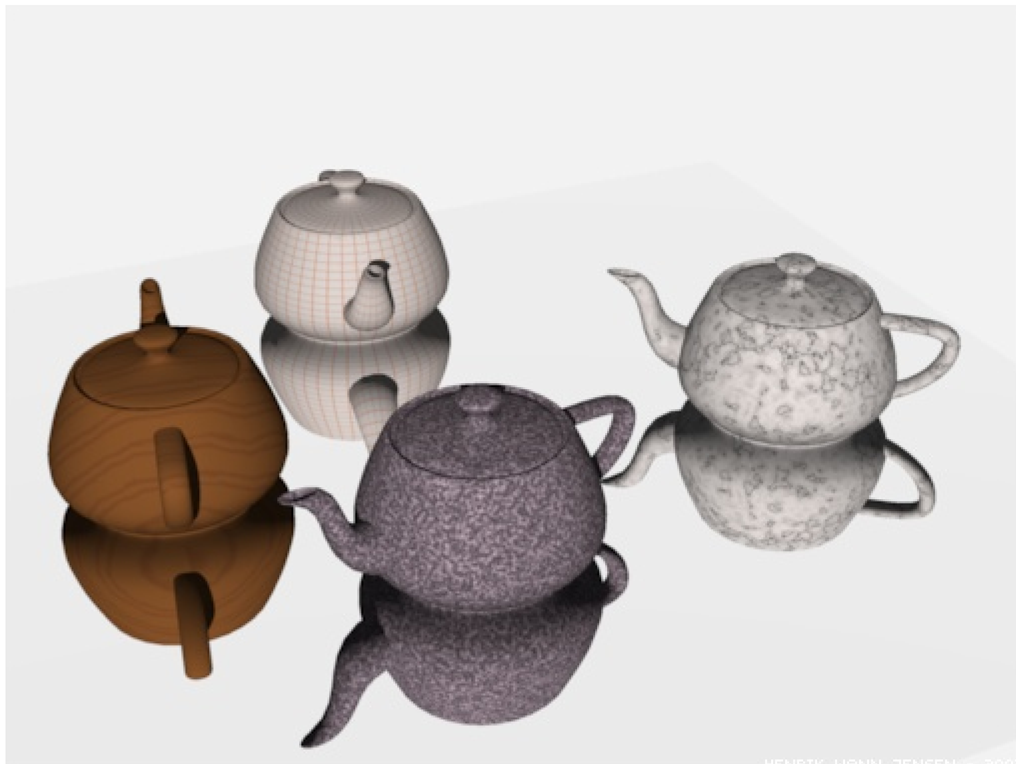
CSE168
Computer Graphics II, Rendering



Spring 2006

Matthias Zwicker

Final project

Render an awe-inspiring image

- Soft shadows
- Path tracing
- Photon mapping
- Illumination from an environment map
- Subsurface scattering
- Motion blur, depth of field
- Procedural modeling, smoke, water

Final project

Previous years

- See web pages http://graphics.ucsd.edu/courses/rendering/2003/, /2004, /2005

Final project

Organization

- Work out project description, due next Wednesday May 24
- Groups up to 3 students possible, but need to define individual contributions explicitly for individual grading
- Project presentation Monday June 12, 7pm

Final project

Prizes

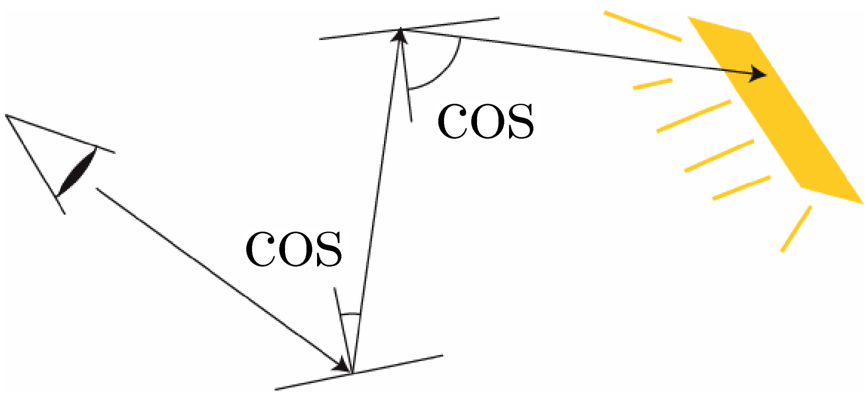
- First prize: $1000 contribution to a trip to SIGGRAPH
- Grand prize: $100 gift certificate for books
- Honorable mention: $50 gift certificate for books

Last time

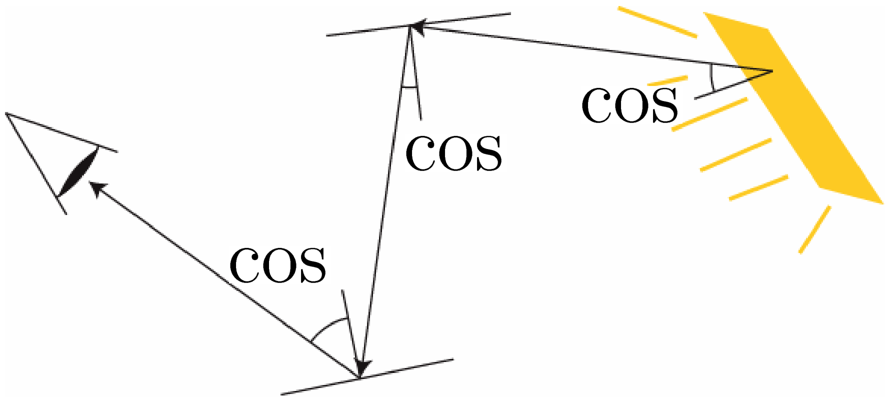
- Explanation of cosine terms in photon transport
- Photon mapping rendering algorithms

Cosine terms

Path tracing



Photon tracing



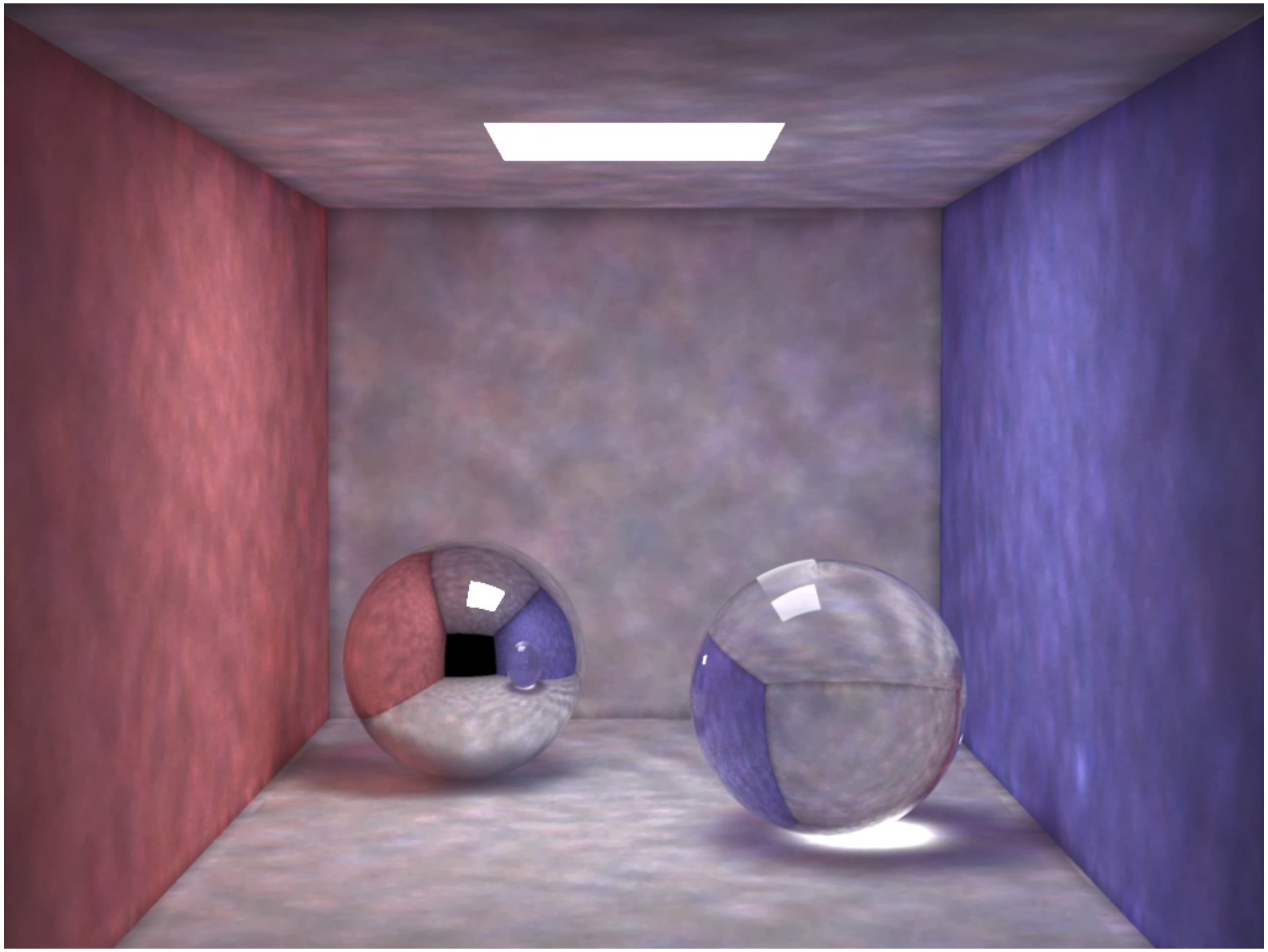
Photon mapping algorithm

Recap

- Emit and transport photons
- Build data structure for fast access
- Use stored photons to estimate reflected radiance

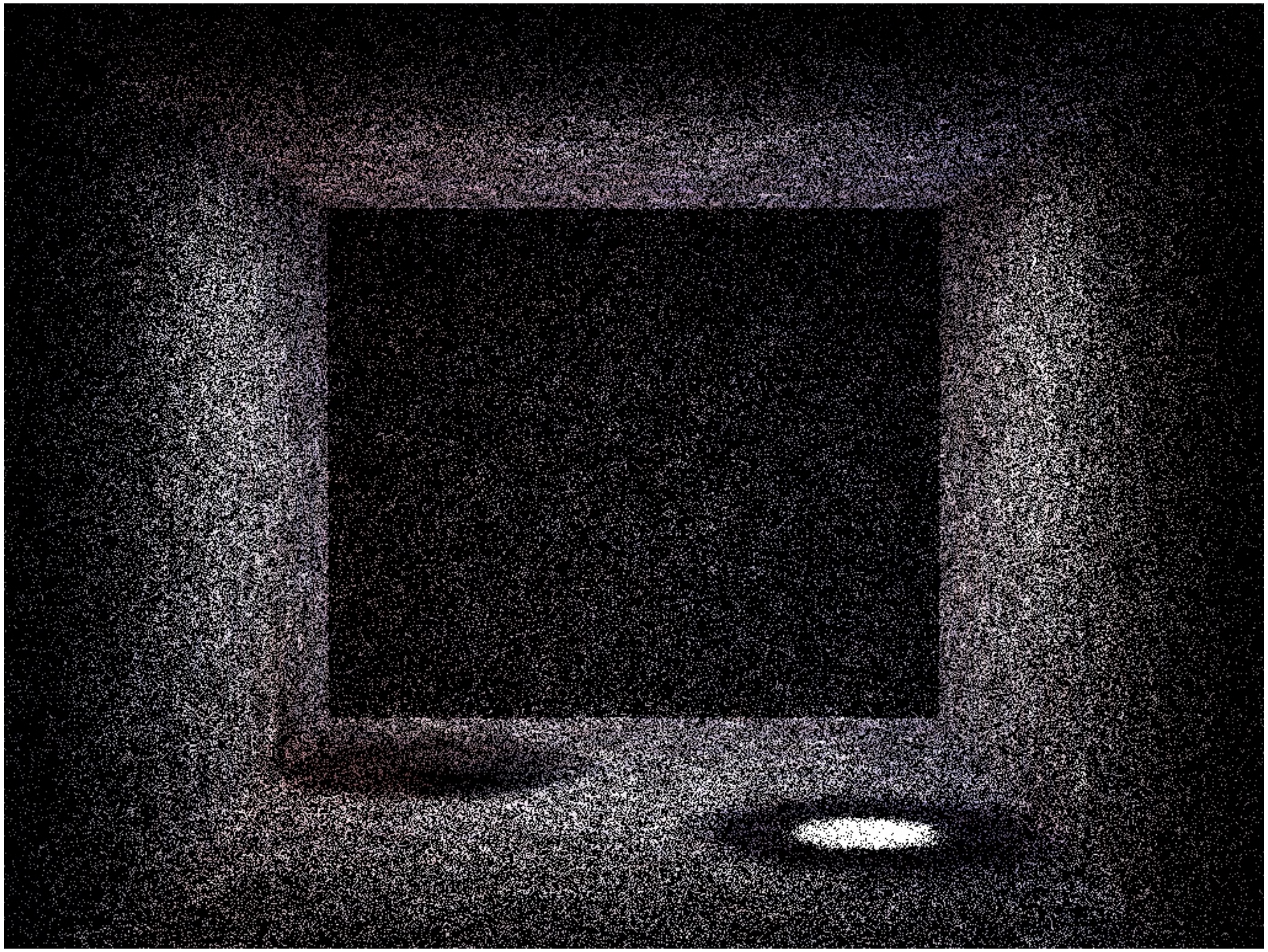
Global illumination

100000 photons, 50 photons in radiance estimate



[Wann Jensen]

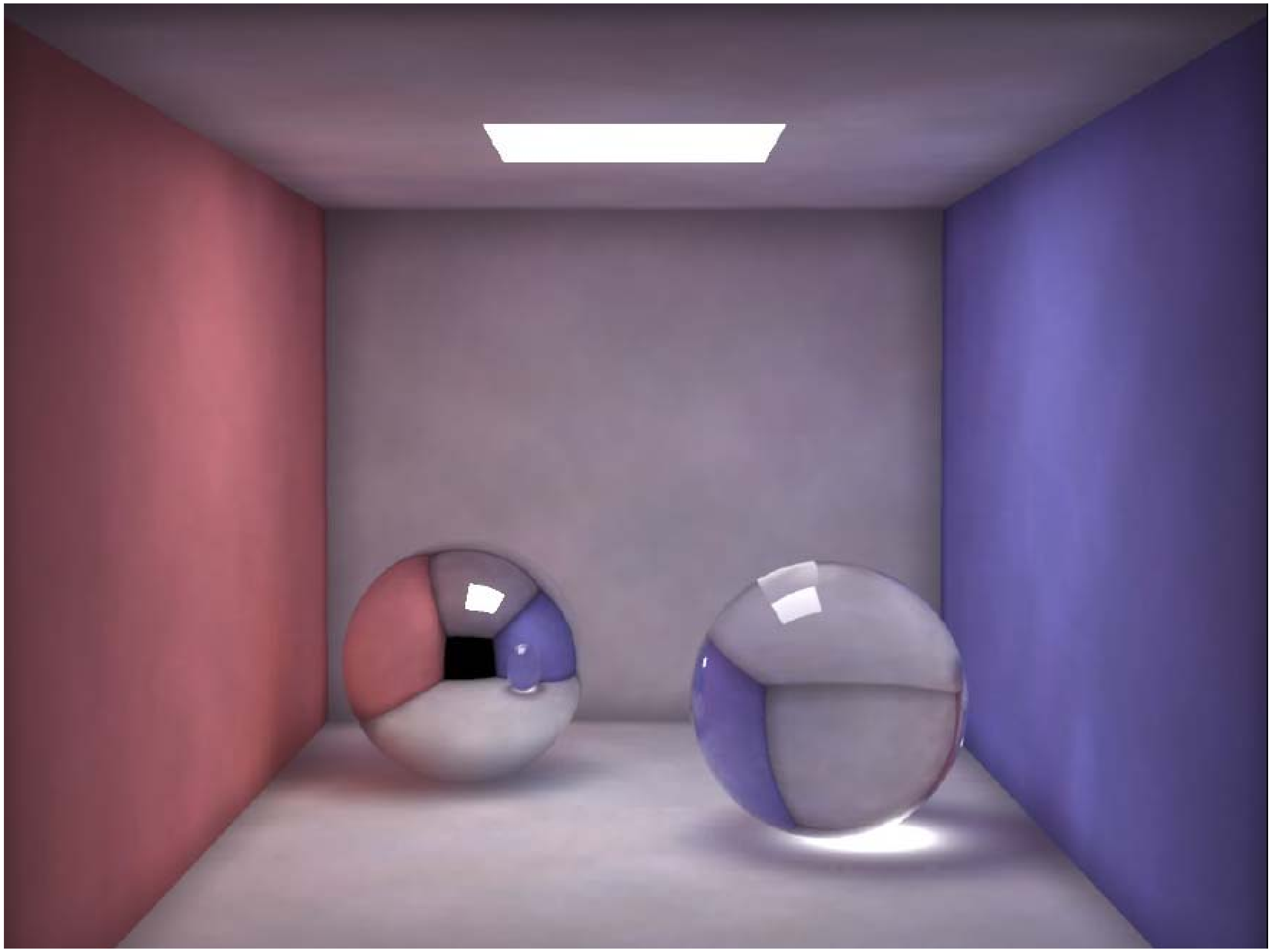
Visualization of the photons



[Wann Jensen]

Global illumination

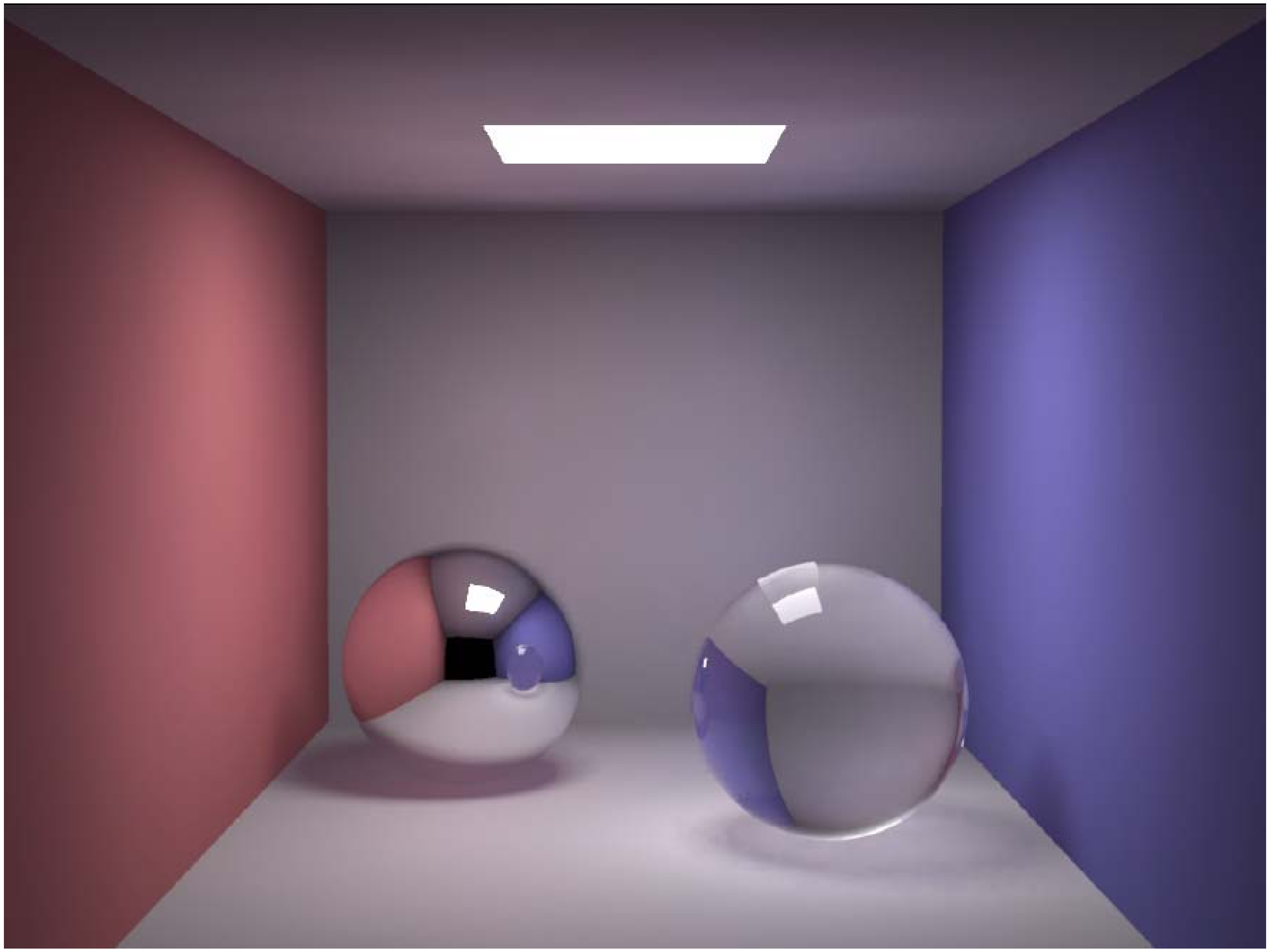
500000 photons, 500 photons in radiance estimate



[Wann Jensen]

Photons for indirect illumination

10000 photons, 500 photons in radiance estimate



[Wann Jensen]

Splitting up the reflection equation

- Reflection equation

$$L_o(\mathbf{x}, \omega_o) = \int_{\mathcal{H}^2} f(\mathbf{x}, \omega_o, \omega_i) L_i(\mathbf{x}, \omega_i) \cos\theta_i d\omega_i$$

- Split up BRDF

$$f(\mathbf{x}, \omega_o, \omega_i) = f_s(\mathbf{x}, \omega_o, \omega_i) + f_d(\mathbf{x}, \omega_o, \omega_i)$$

- Split up incoming radiance

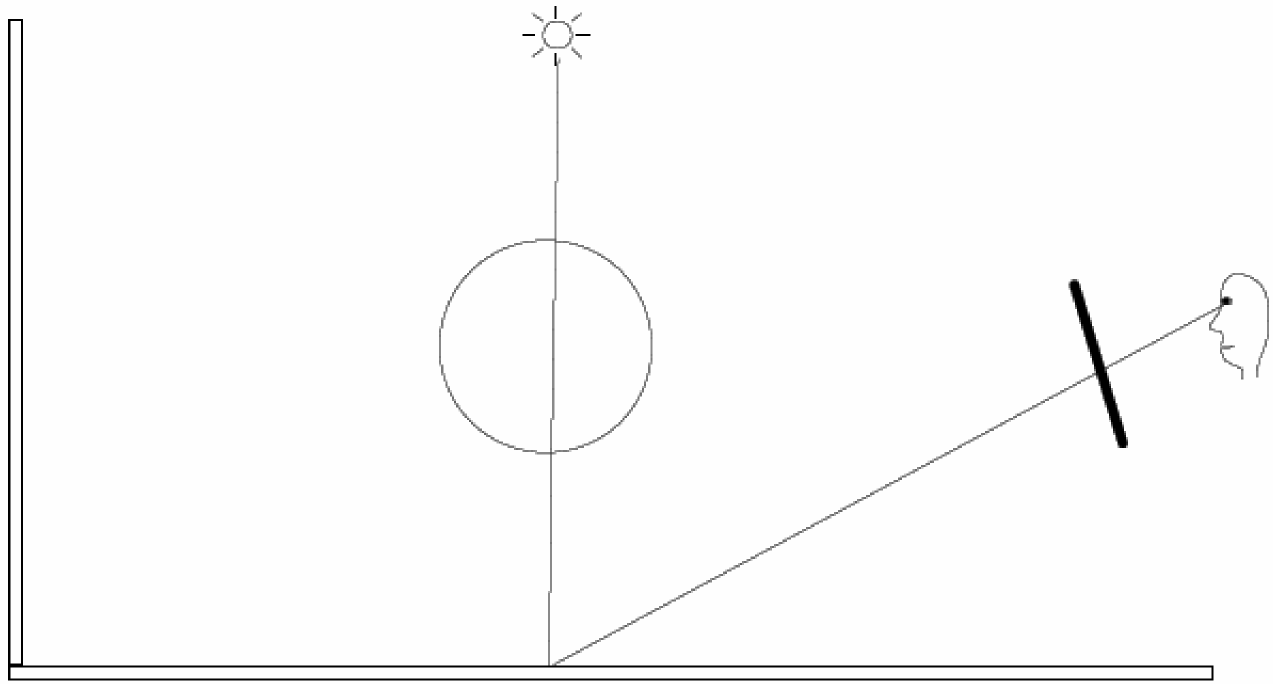
$$L_i(\mathbf{x}, \omega_i) = L_{i,l}(\mathbf{x}, \omega_i) + L_{i,c}(\mathbf{x}, \omega_i) + L_{i,d}(\mathbf{x}, \omega_i)$$

Photon maps

- Three photon maps, one for each illumination term
  - Direct
  - Indirect diffuse
  - Caustic (indirect specular)
- Finite state machine to keep track of photon type

Direct illumination

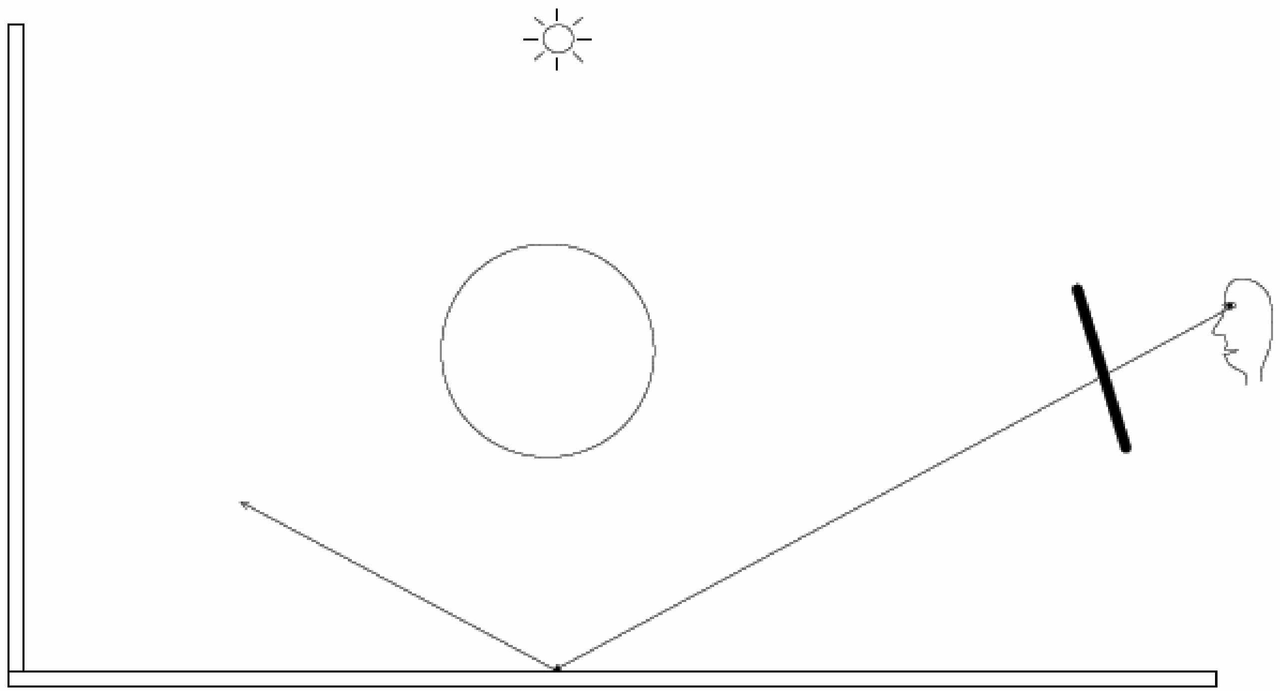
- Sample light sources



[Wann Jensen]

Specular reflection

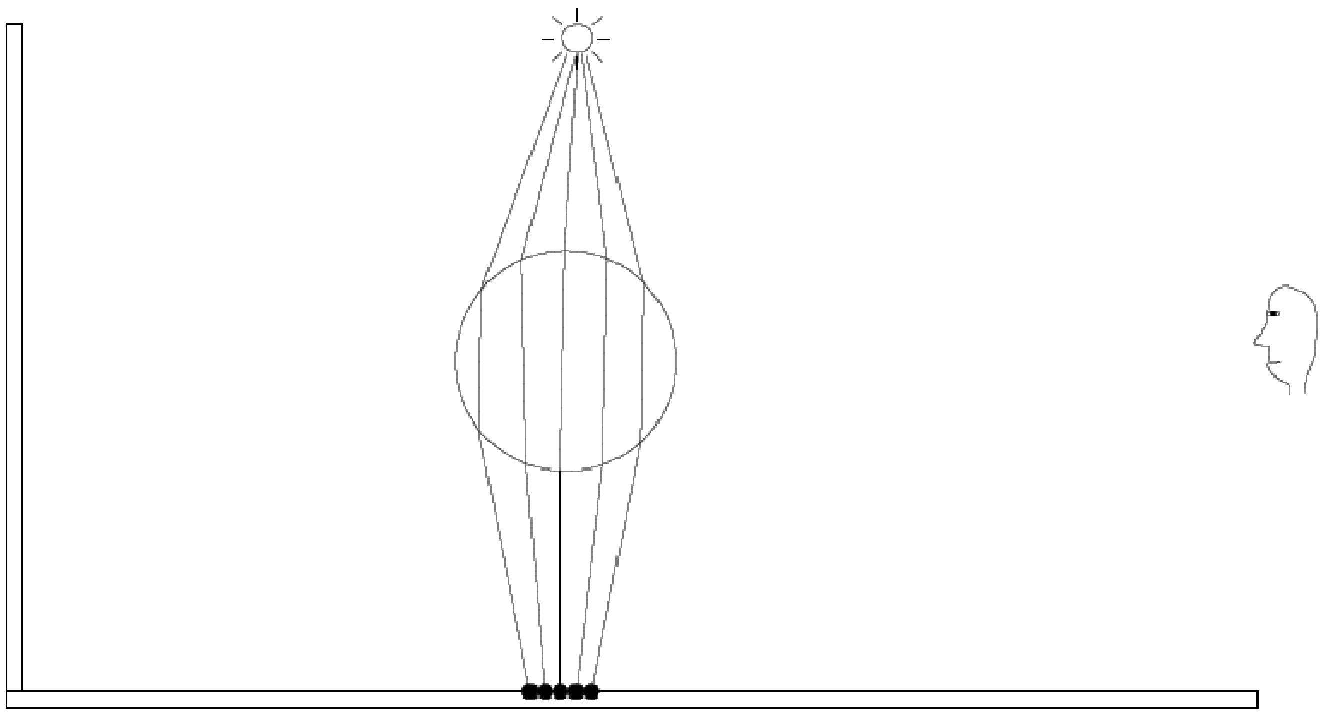
- Path tracing



[Wann Jensen]

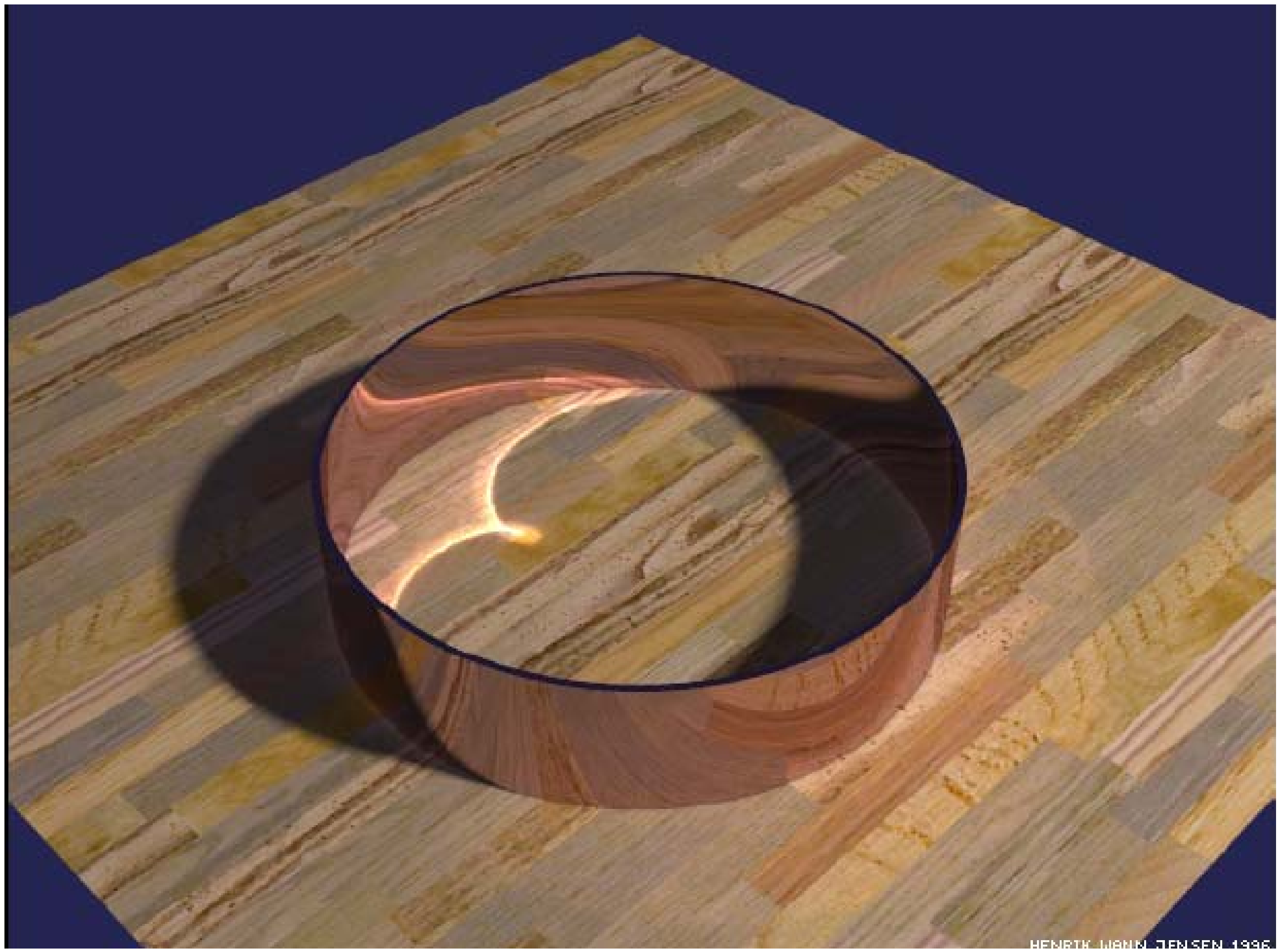
Caustics

- Radiance estimation using caustic photon map



[Wann Jensen]

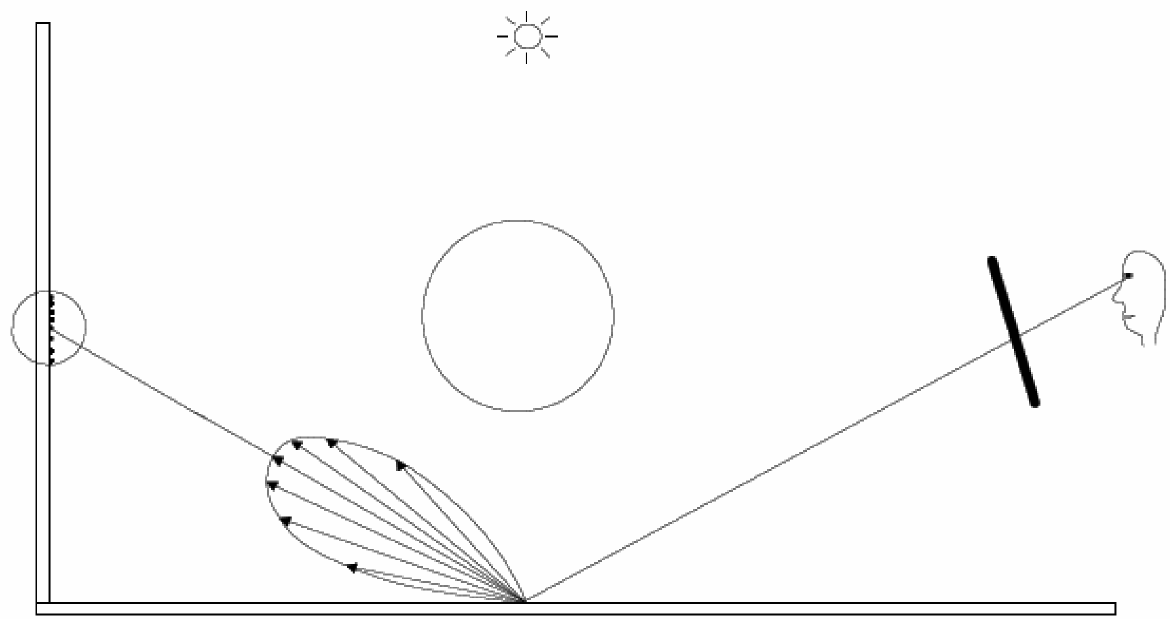
Reflections inside a ring



50000 photons, 50 photons for radiance estimation

Diffuse indirect illumination

- Path tracing, final gathering using all three photon maps, radiance estimation using indirect photons



Final gathering

[Wann Jensen]

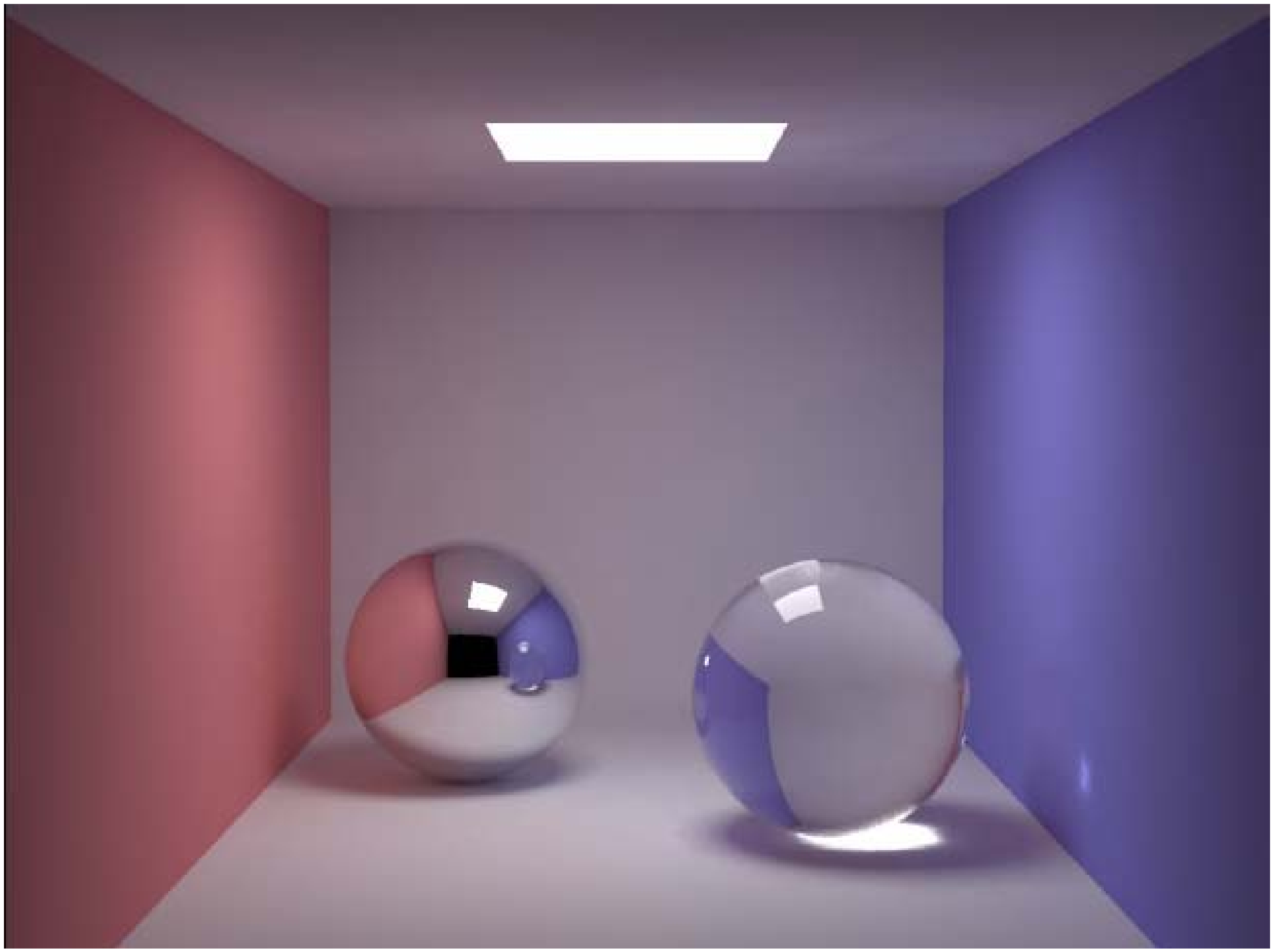
Today

- Irradiance caching
- Radiosity

Diffuse indirect illumination

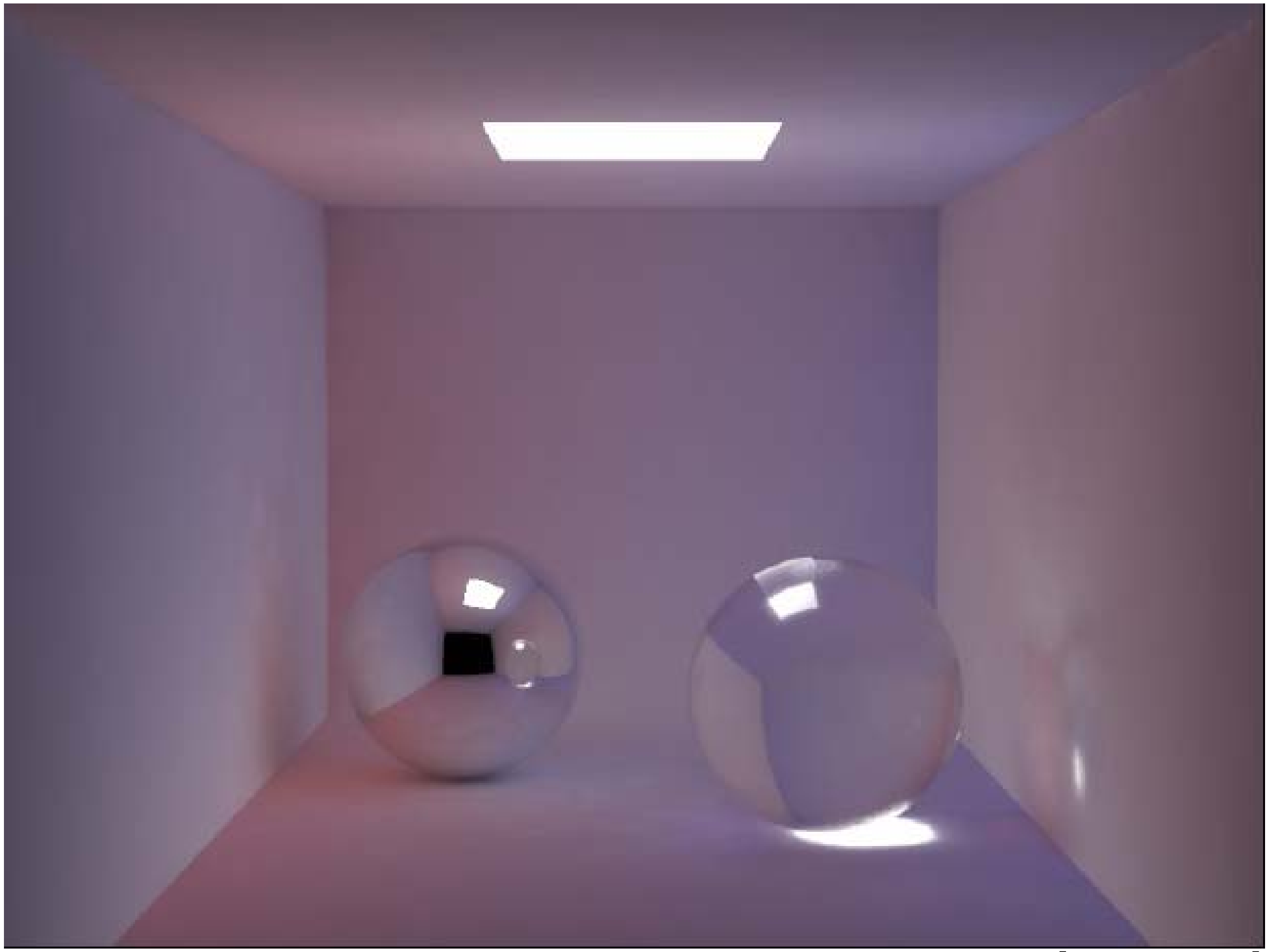
- Path tracing: slow
- Final gathering using all three photon maps: slow
- Radiance estimation using indirect photons: inaccurate for reasonable number of photons
- Other ideas?

Global illumination



[Wann Jensen]

Indirect irradiance



[Wann Jensen]

Indirect irradiance

[Humphreys, Pharr]

Indirect irradiance

- Changes very smoothly on diffuse surfaces, except for caustics

Irradiance caching

- "A ray tracing solution for diffuse interreflection", Ward, Rubinstein, Clear, SIGGRAPH 88
- Assume diffuse surfaces
- Cache irradiance samples instead of incident radiance as in photon mapping
- Interpolate cached samples
- Compute new samples only if interpolation fails

Irradiance caching algorithm

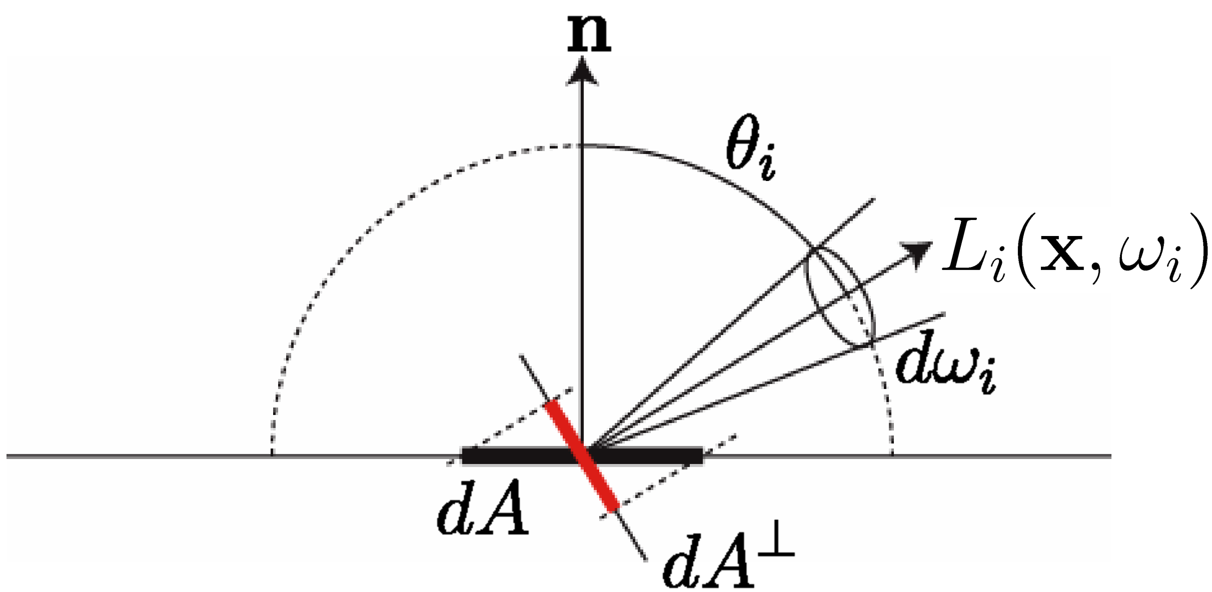
Three components

- Irradiance sampling
- Irradiance caching
- Irradiance interpolation

- Similar to photon mapping, but all steps are performed in main rendering pass

Irradiance sampling

$$E(\mathbf{x}) \;=\; \int_{\mathcal{H}^2} L_i(\mathbf{x},\omega_i)\cos\theta_i d\omega_i$$



Irradiance sampling

$$\begin{aligned} E(\mathbf{x}) &= \int_{\mathcal{H}^2} L_i(\mathbf{x}, \omega_i) \cos\theta_i d\omega_i \\ &= \int_0^{2\pi} \int_0^{\pi/2} L_i(\mathbf{x}, \theta_i, \phi) \cos\theta_i \sin\theta d\theta d\phi \end{aligned}$$

Irradiance sampling

$$\begin{aligned} E(\mathbf{x}) &= \int_{\mathcal{H}^2} L_i(\mathbf{x}, \omega_i) \cos\theta_i d\omega_i \\ &= \int_0^{2\pi} \int_0^{\pi/2} L_i(\mathbf{x}, \theta_i, \phi) \cos\theta_i \sin\theta d\theta d\phi \\ &\approx \frac{\pi}{TP} \sum_{t=1}^{T} \sum_{p=1}^{P} L_i(\theta_t, \phi_p) \end{aligned}$$

- Stratified sampling of the hemisphere
- Subdivision *T*,*P*, uniform random variables $\xi, \ \psi$

$$\theta_t = \sin^{-1}\left(\sqrt{\frac{t-\xi}{T}}\right), \text{ and } \phi_p = 2\pi \frac{p-\psi}{P}$$

Irradiance sampling

- Compute $L_i(\mathbf{x}, \omega_i)$ using path tracing or photon gathering at the first hit point
- For good quality expect tracing 200-5000 paths
- Costly, but we will do this only at few locations in the image

Irradiance sampling

- Assign a range for each sample, within which it can be used for interpolation
- Where irradiance changes quickly, range should be small
- Where irradiance changes slowly, range should be large
- Rate of change of irradiance depends on distance to visible surfaces

Irradiance sampling

[Wojciech Jarosz]

Irradiance sampling

Harmonic mean heuristics

- The range is given by a radius

$$r_j = \frac{N}{\sum_{i=1}^{N} 1/d_i}$$

  where N is the number of paths, d_i is the distance to the first intersection along the path
- Average would weight infinite distances too heavily

Irradiance caching

Irradiance sample

```
struct irradiance_sample {
  vector3 E  // irradiance
  vector3 n  // normal
  vector3 p  // position
  float r    // range
}
```

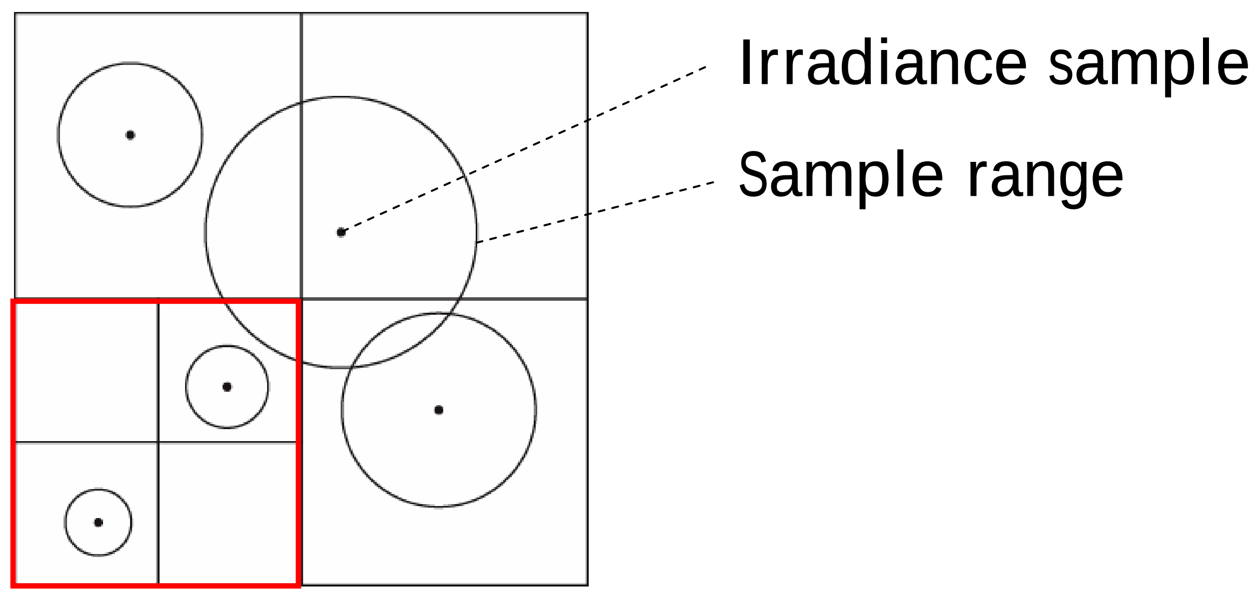
Irradiance caching

- Store samples in octree
- Add sample to each cell that it overlaps
- Adaptively subdivide octree such that each cell has limited number of samples

Irradiance caching

Octree example

- Adaptive subdivision such that each cell contains <3 samples



Irradiance interpolation

- Need to determine which samples should be used for interpolation

Error estimate

- Given a point on the surface $\mathbf{x}$ with normal $\mathbf{n}(\mathbf{x})$, estimate the difference to the irradiance cached at $\mathbf{x}_j$

Irradiance interpolation

Error estimate

- Ad-hoc estimate for error of sample j

difference in position

difference in orientation

$$\epsilon_j(\mathbf{x}) \leq E_j \left(\frac{4}{\pi} \frac{\|\mathbf{x} - \mathbf{x}_j\|}{r_j} + \sqrt{2 - 2\mathbf{n}(\mathbf{x}) \cdot \mathbf{n}(\mathbf{x}_j)} \right)$$

Irradiance interpolation

- Interpolation weights

$$w_j(\mathbf{x}) = \frac{1}{\frac{\|\mathbf{x}-\mathbf{x}_j\|}{r_j} + \sqrt{1 - \mathbf{n}(\mathbf{x}) \cdot \mathbf{n}(\mathbf{x}_j)}} \approx \epsilon_j(\mathbf{x})$$

- Interpolated irradiance

$$E(\mathbf{x}) = \frac{\sum_i w_i(\mathbf{x}) E(\mathbf{x}_i)}{\sum_i w_i(\mathbf{x})}$$

Irradiance caching algorithm

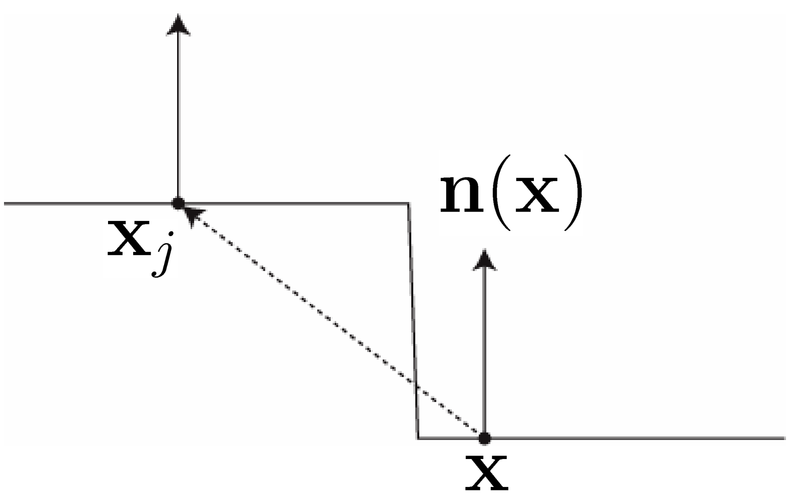
```
W = 0
for( all irradiance samples j in octree cell
     overlapping with x ) {
  compute weight w_j
  if( <sample is valid> ) {
    W += w_j; wE += w_j*E[j]
  }
}

if( W > 0 ) {
  return wE/W
} else {
  return( compute new irradiance sample )
}
```

Irradiance caching algorithm

```
<sample is valid> =

dist( x – x[j] ) < r[j] // within range
&& w_j > 1/a            // sufficient weight
&& dot( x[j] - x, n(x) ) < 0
                        // x[j] is behind x
```



Sample at $\mathbf{x}_j$ is invalid

Non-diffuse surfaces

Approximation

$$
\begin{aligned}
L_o(\mathbf{x}, \omega_o) &= \int_{\mathcal{H}^2} f(\mathbf{x}, \omega_o, \omega_i) L_i(\mathbf{x}, \omega_i) \cos\theta_i d\omega_i \\
&\approx \left(\int_{\mathcal{H}^2} f(\mathbf{x}, \omega_o, \omega_i) d\omega_i \right) \left(\int_{\mathcal{H}^2} L_i(\mathbf{x}, \omega_i) \cos\theta_i d\omega_i \right)
\end{aligned}
$$

Non-diffuse surfaces

Approximation

$$\begin{aligned} L_o(\mathbf{x},\omega_o) &= \int_{\mathcal{H}^2} f(\mathbf{x},\omega_o,\omega_i) L_i(\mathbf{x},\omega_i)\cos\theta_i d\omega_i \\ &\approx \left(\int_{\mathcal{H}^2} f(\mathbf{x},\omega_o,\omega_i) d\omega_i\right) \left(\int_{\mathcal{H}^2} L_i(\mathbf{x},\omega_i)\cos\theta_i d\omega_i\right) \\ &= \frac{1}{2}\rho_{hd}(\omega_o) E(\mathbf{x}) \end{aligned}$$

Hemispherical directional reflectance $\rho_{hd}(\omega_o)$

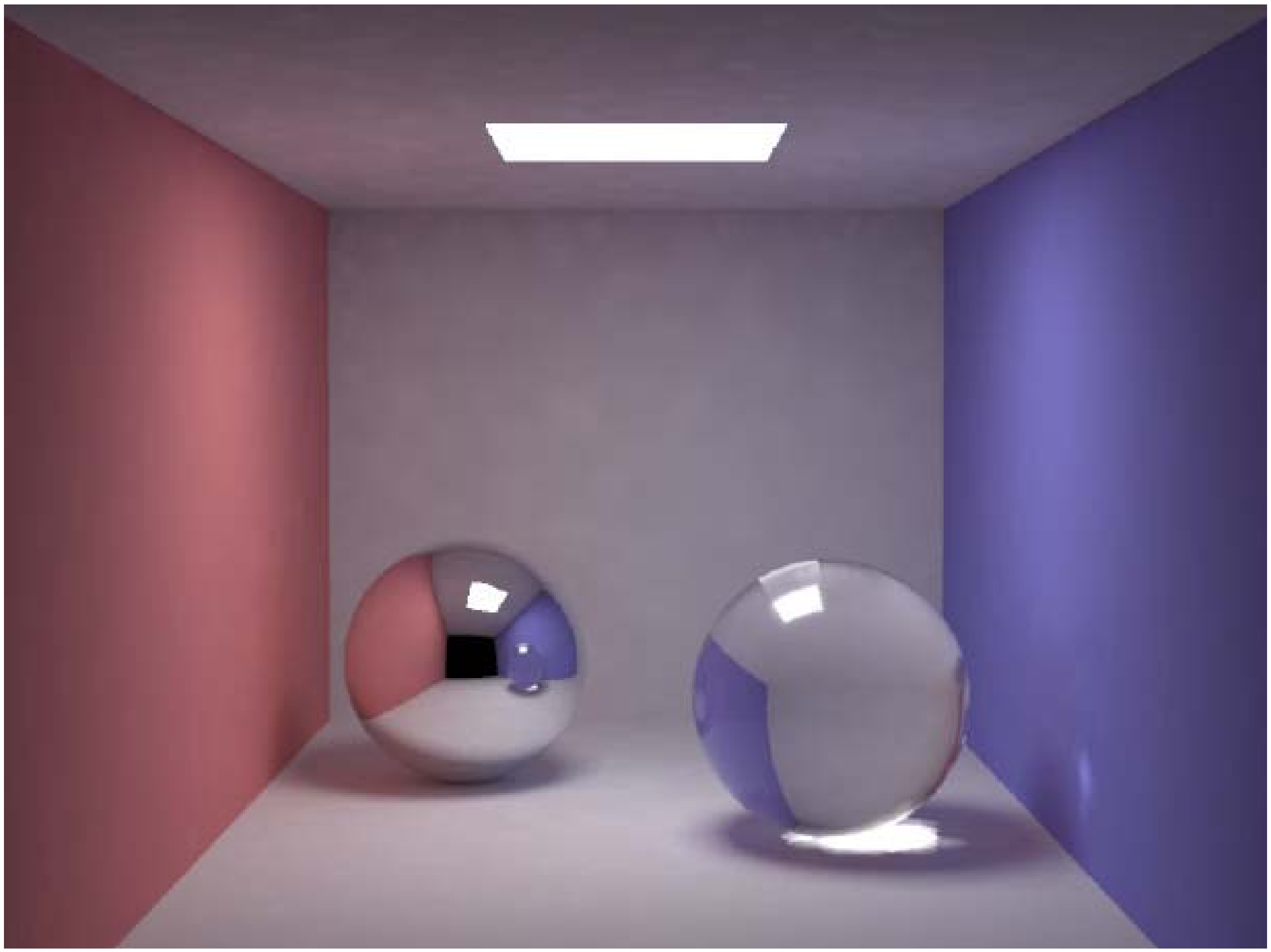
Photon mapping and irradiance caching

- Caustics break assumptions of irradiance caching
- Exclude caustic paths from irradiance sampling

Advanced technique

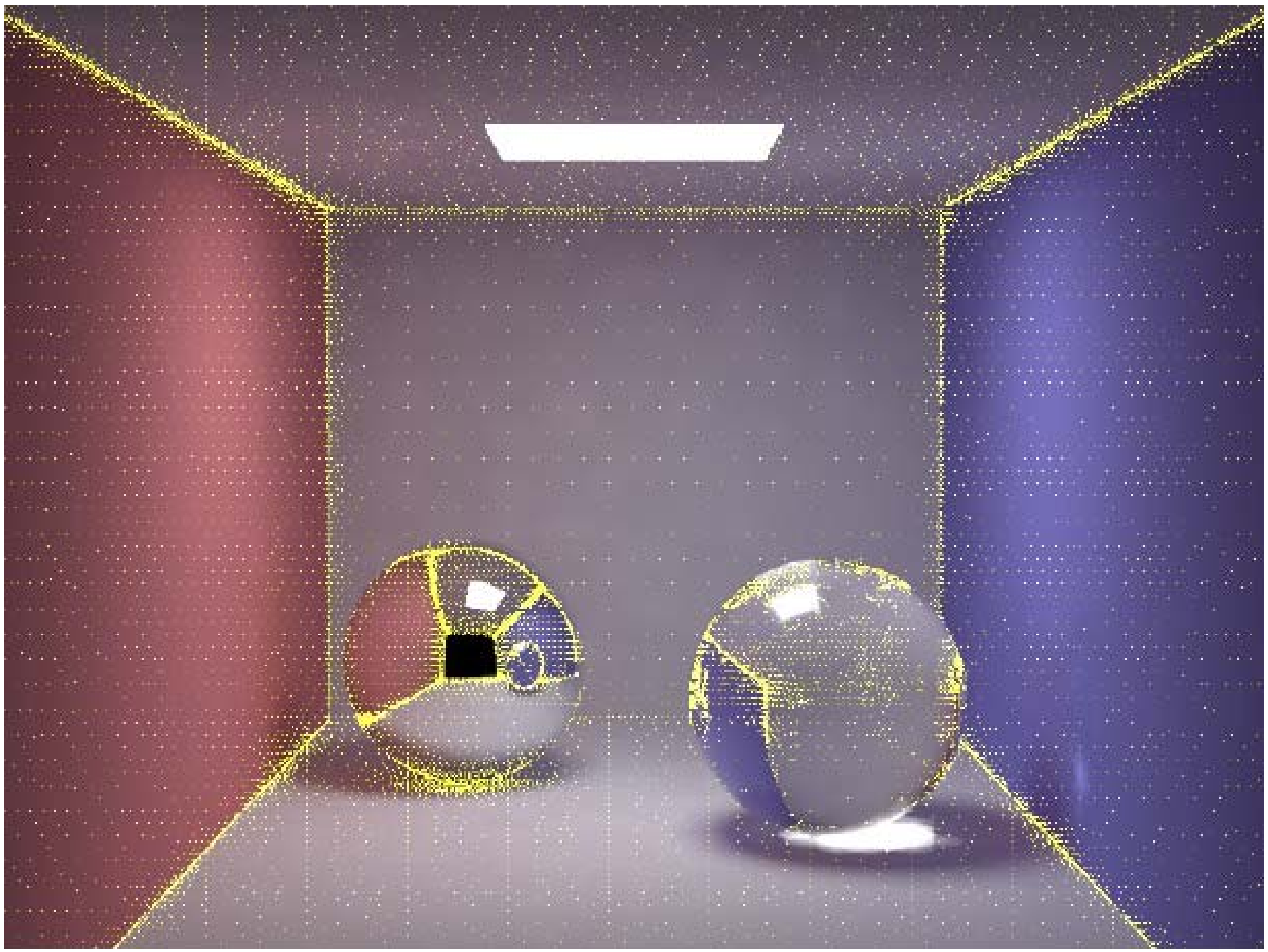
- Use photon map for importance sampling during path tracing

Irradiance caching examples



1000 sample rays, w>10 [Wann Jensen]

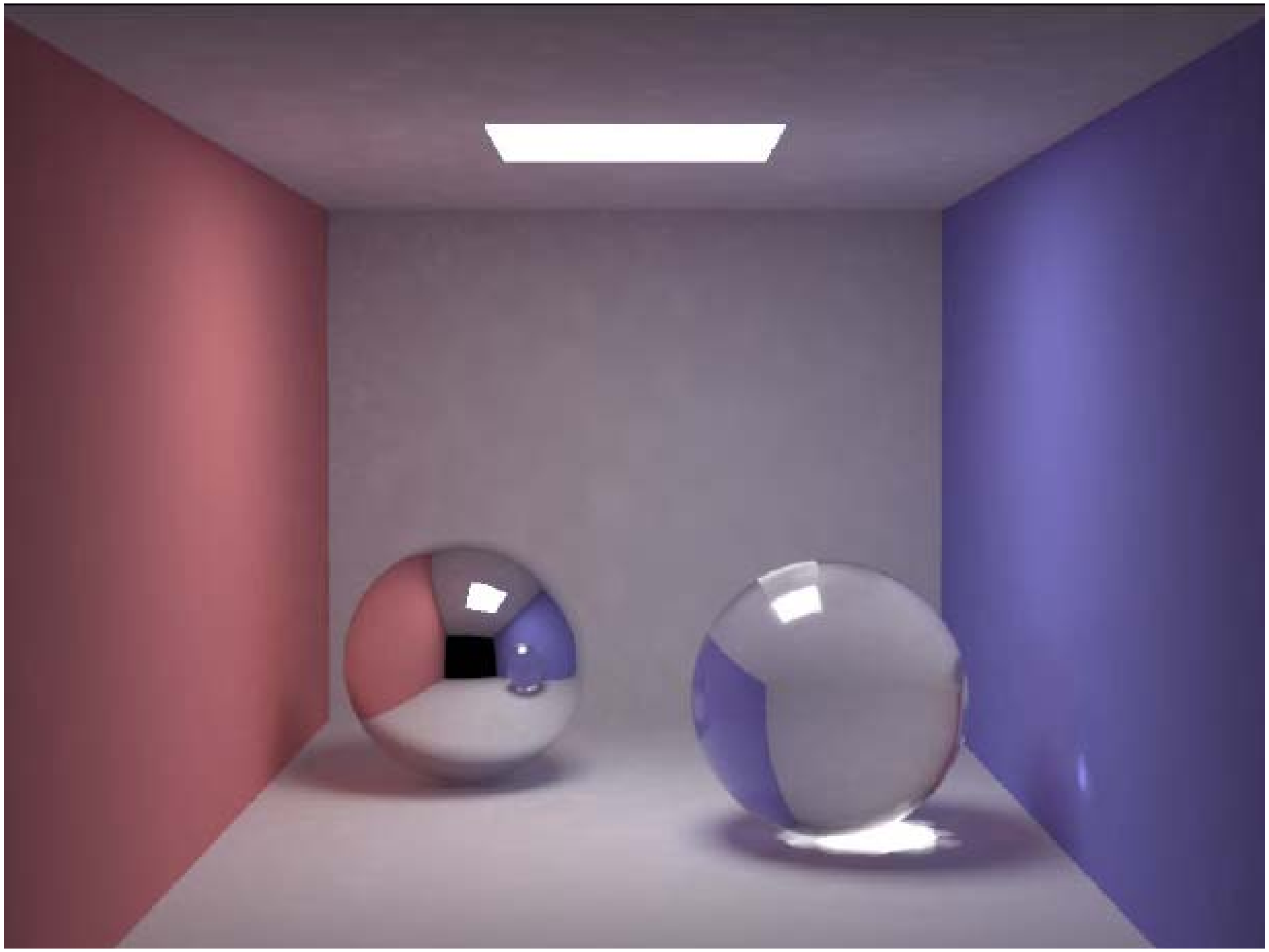
Irradiance caching examples



1000 sample rays, w>10

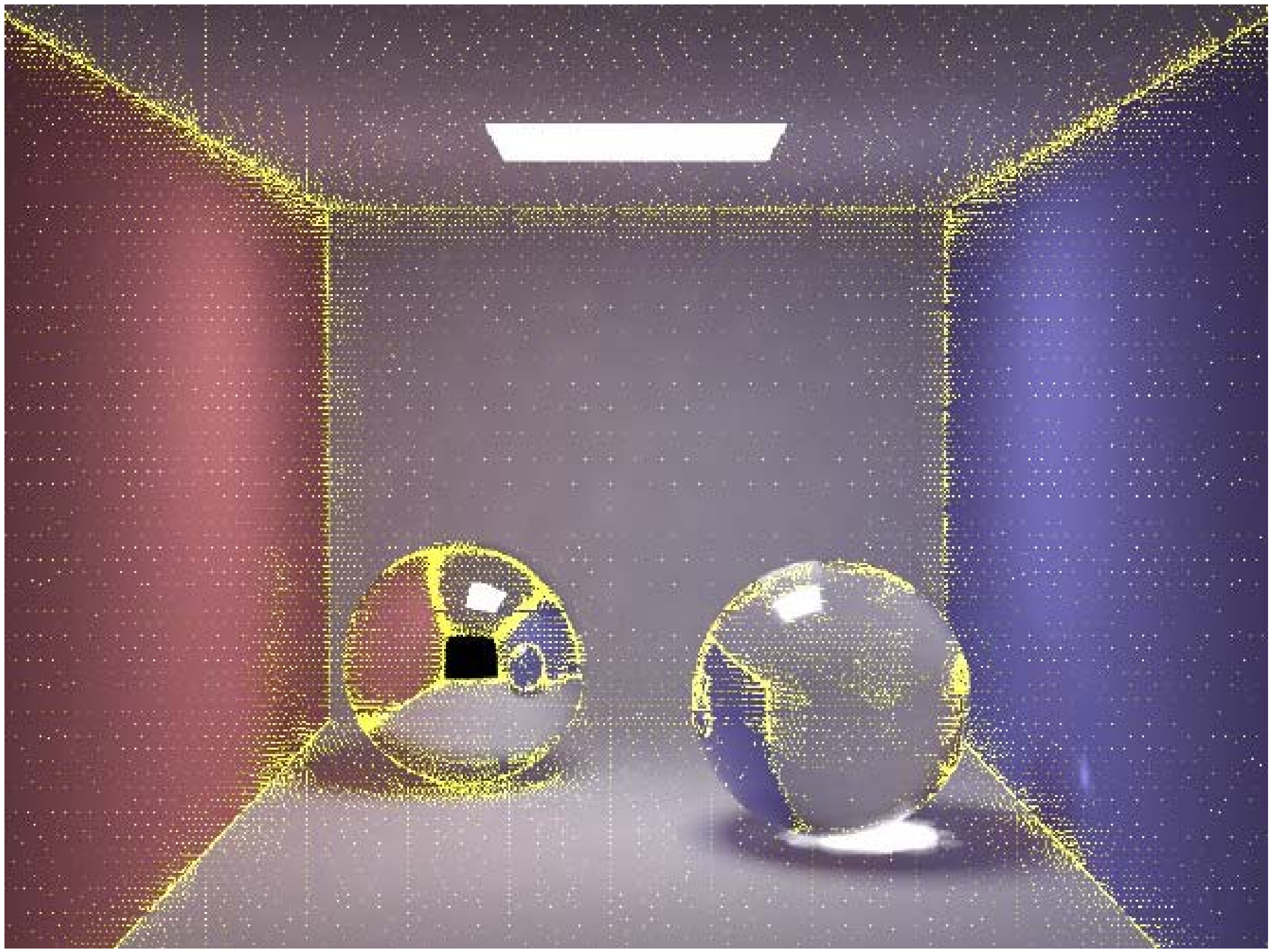
[Wann Jensen]

Irradiance caching examples



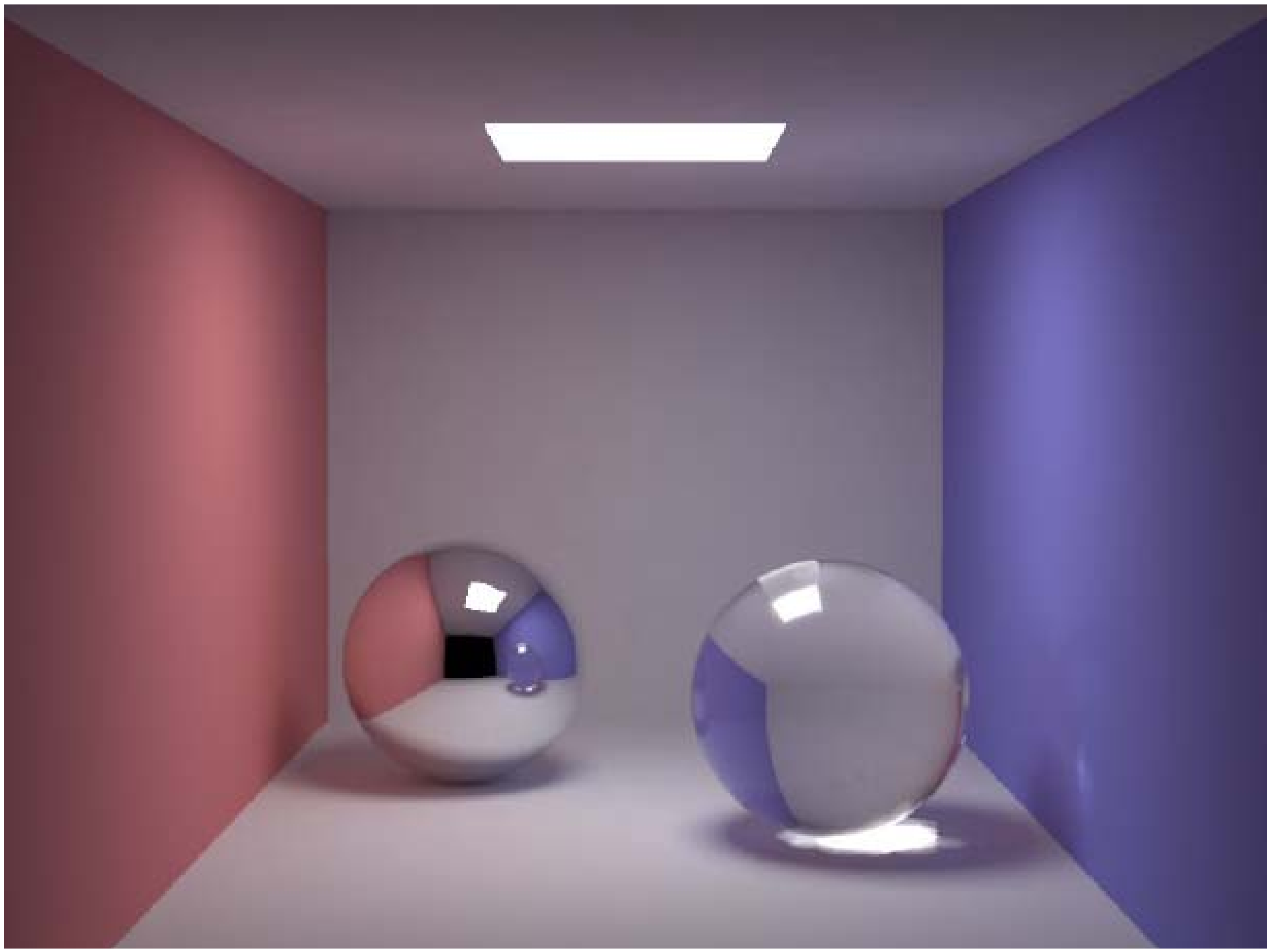
1000 sample rays, w>20 [Wann Jensen]

Irradiance caching examples



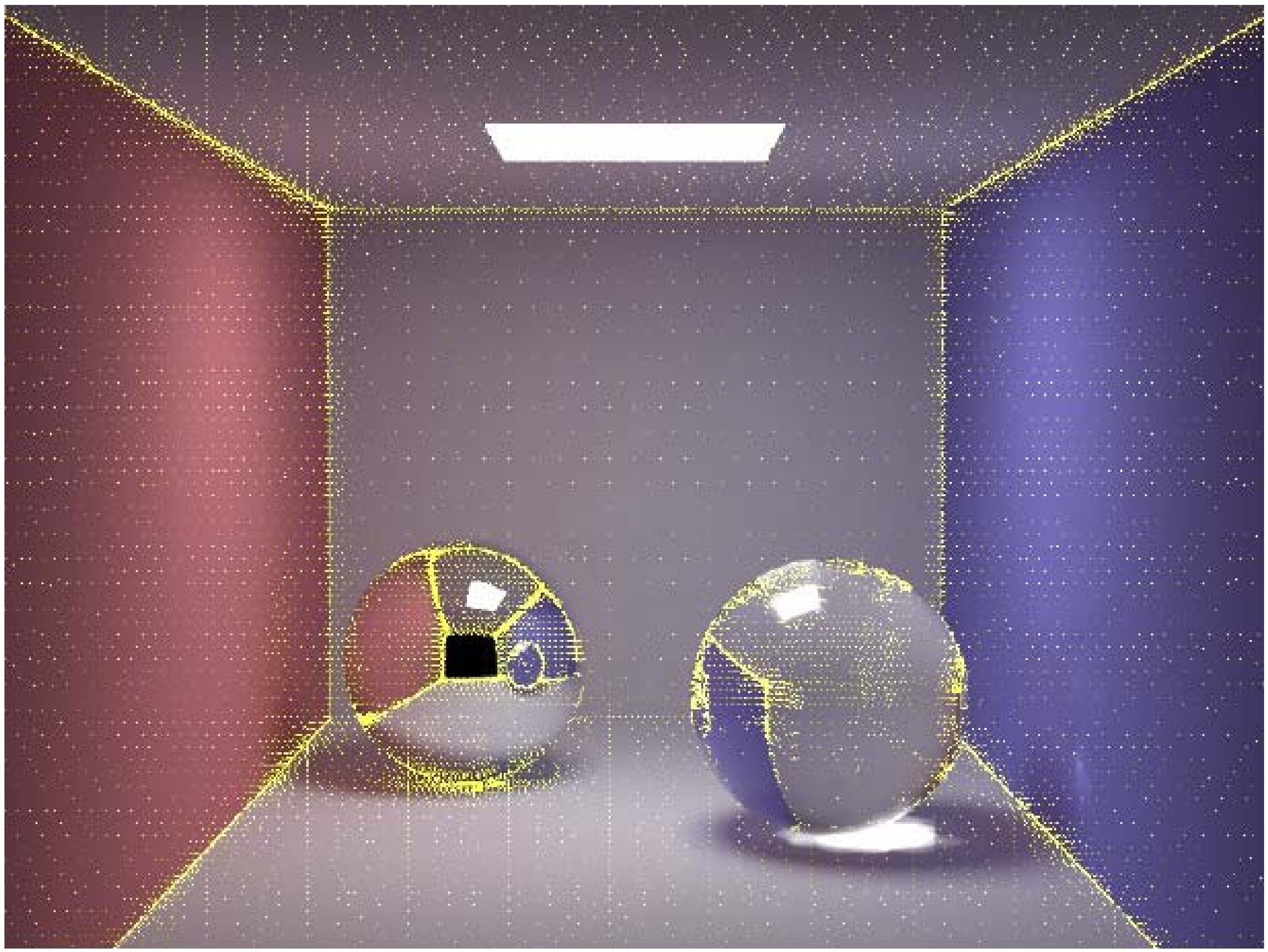
1000 sample rays, w>20 [Wann Jensen]

Irradiance caching examples



5000 sample rays, w>10 [Wann Jensen]

Irradiance caching examples



5000 sample rays, w>10

[Wann Jensen]

Irradiance caching

[Humphreys, Pharr]

Irradiance caching example

[Humphreys, Pharr]

Path tracing

[Humphreys, Pharr]

Approx. same amount of computation

Next time

- Midterm