Biased Monte Carlo Ray Tracing: Irradiance Caching and Photon Maps

Henrik Wann Jensen

Stanford University

May 16, 2000

Unbiased and consistent Monte Carlo methods

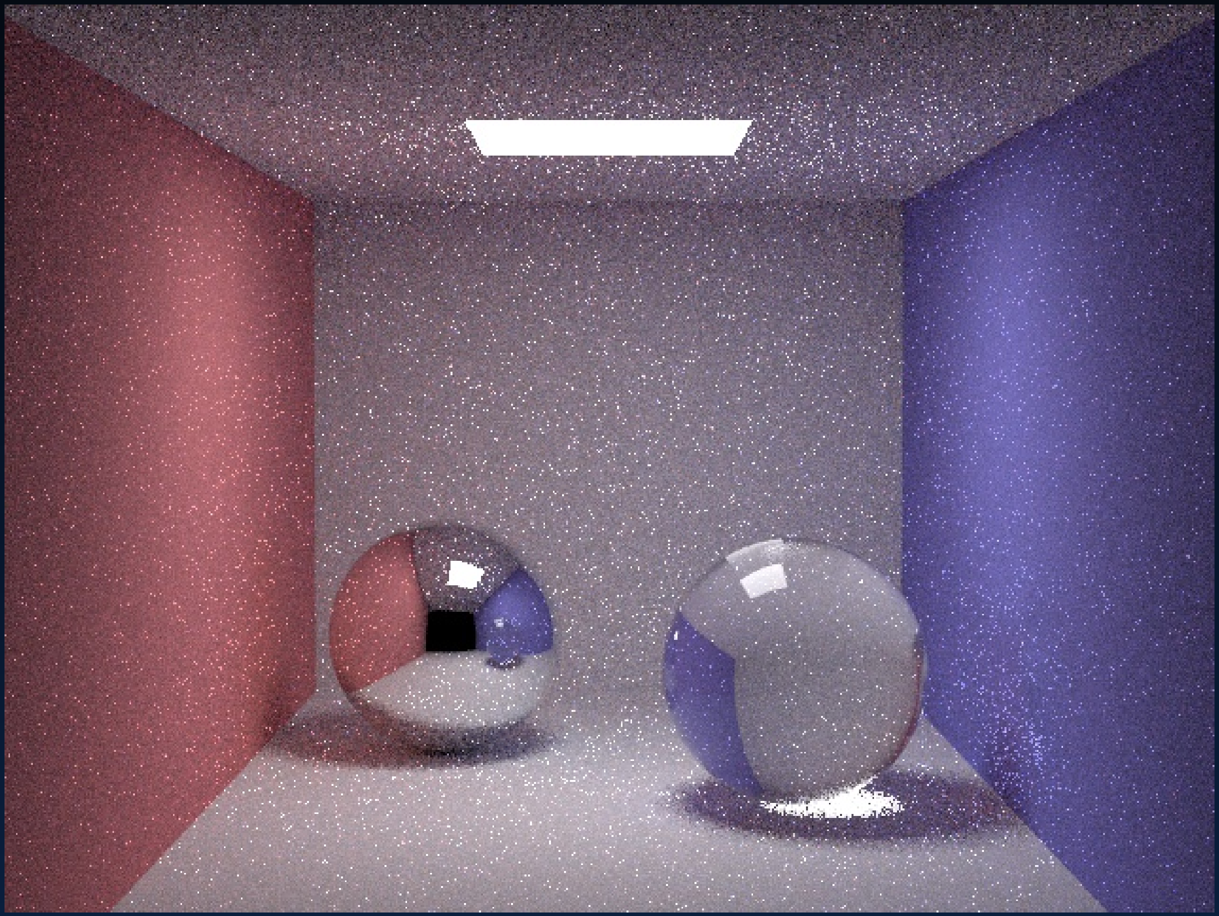
Unbiased estimator:

$$E\{X\} = \int \ldots$$

Consistent estimator:

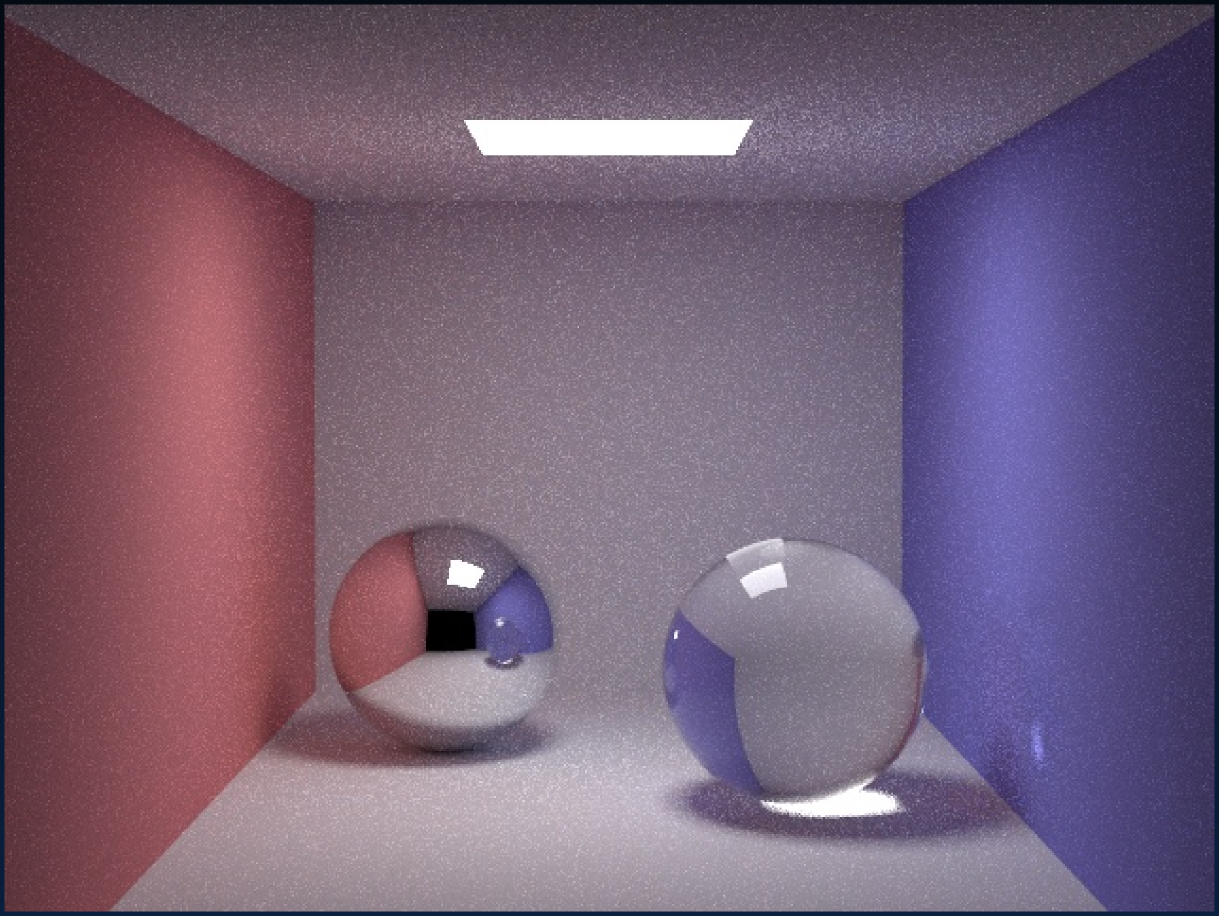
$$\lim_{N \to \infty} E\{X\} \to \int \ldots$$

Path tracing (unbiased)



10 rays/pixel

Path tracing (unbiased)



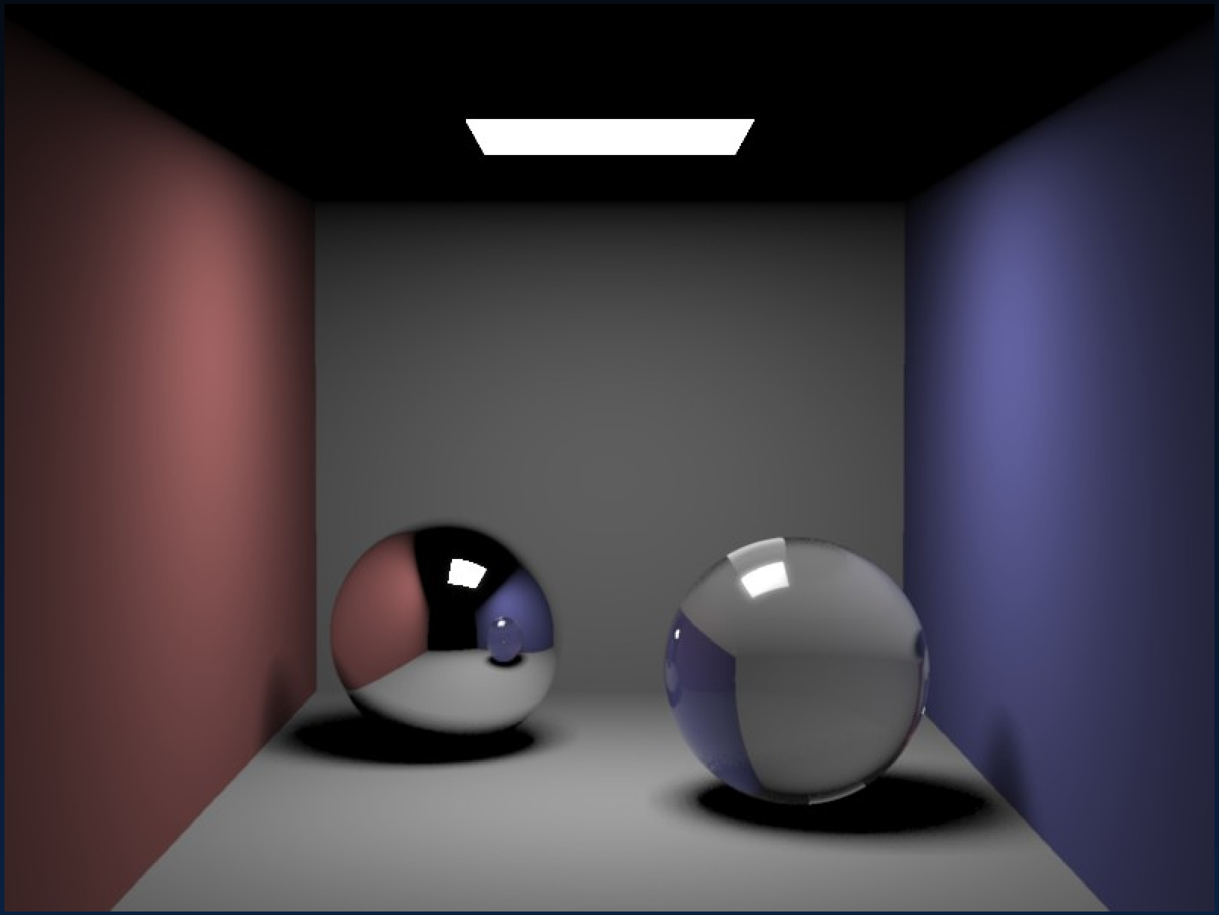
100 rays/pixel

Two consistent techniques

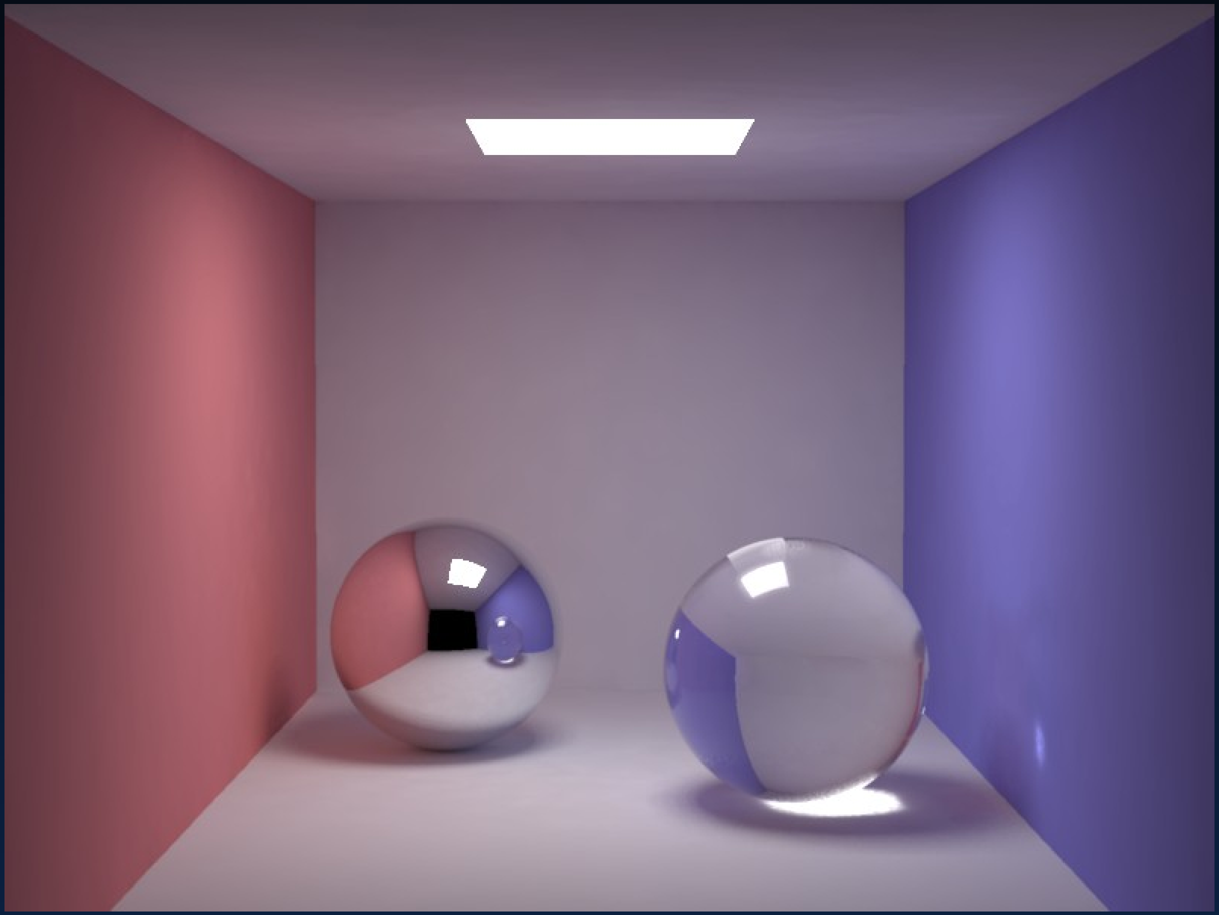
Irradiance caching : Compute irradiance at selected points and interpolate.

Photon maps : Render using flux approximation.

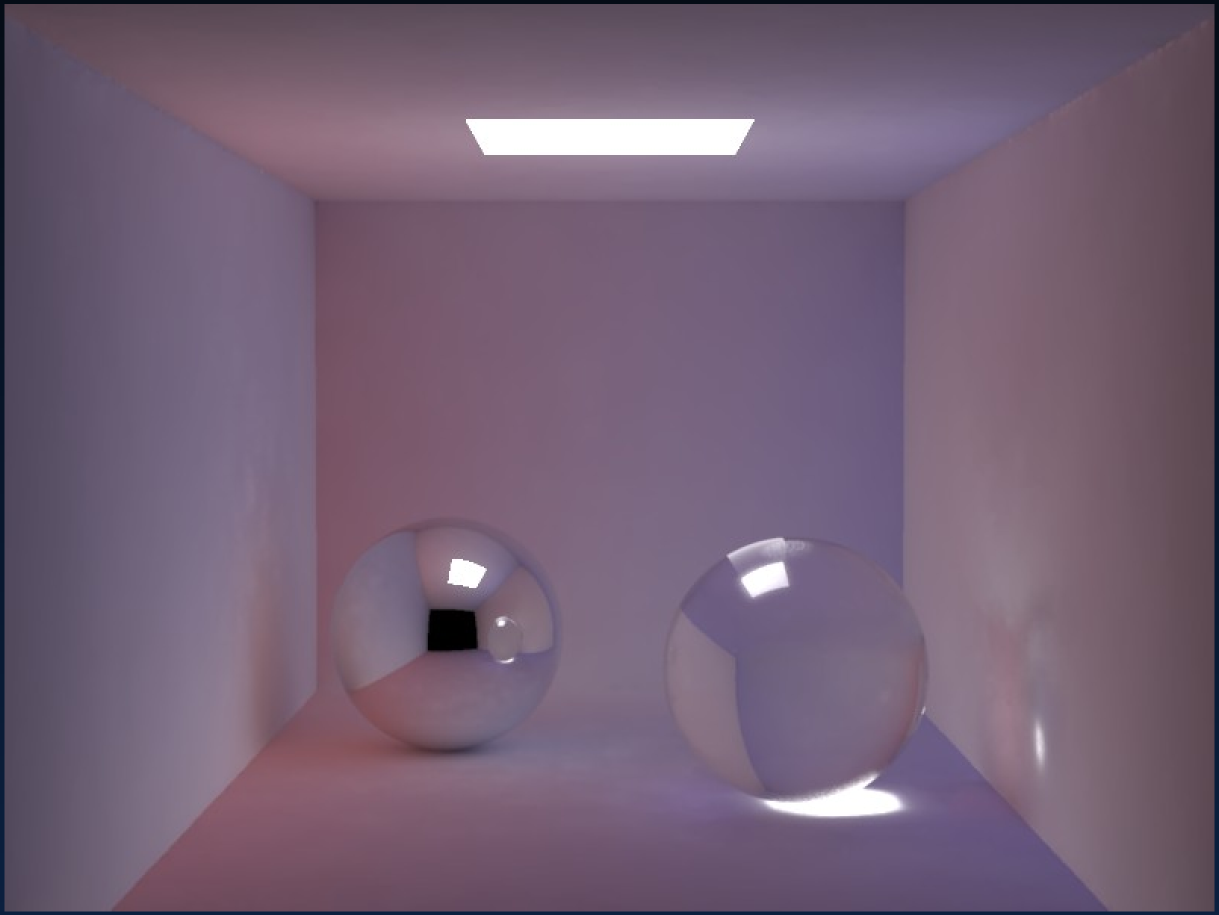
Cornell box: direct illlumination



Cornell box: global illlumination



Cornell box: irradiance



Irradiance caching: idea

Greg Ward, Francis Rubinstein and Robert Clear:
"A Ray Tracing Solution for Diffuse Interreflection".
Proceedings of SIGGRAPH 1988.

Idea: Irradiance changes slowly $\rightarrow$ interpolate.

Irradiance sampling

$$E(x) \;=\; \int_0^{2\pi} \int_0^{\pi/2} L'(x, \theta, \phi) \cos\theta \sin\theta \, d\theta \, d\phi$$

Irradiance sampling

$$E(x) = \int_0^{2\pi} \int_0^{\pi/2} L'(x, \theta, \phi) \cos\theta \sin\theta \, d\theta \, d\phi$$
$$\approx \frac{\pi}{TP} \sum_{t=1}^{T} \sum_{p=1}^{P} L'(\theta_t, \phi_p)$$

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

Irradiance change

$$\epsilon(x) \leq \underbrace{\left|\frac{\partial E}{\partial x}(x - x_0)}_{\text{position}} + \underbrace{\frac{\partial E}{\partial \theta}(\theta - \theta_0)\right|}_{\text{rotation}}$$

Irradiance change

$$\epsilon(x) \leq \left|\frac{\partial E}{\partial x}(x - x_0) + \frac{\partial E}{\partial \theta}(\theta - \theta_0)\right|$$

position rotation

$$\leq E_0 \left(\frac{4}{\pi}\frac{||x - x_0||}{x_{avg}} + \sqrt{2 - 2\vec{N}(x)\cdot\vec{N}(x_0)}\right)$$

position rotation

Irradiance interpolation

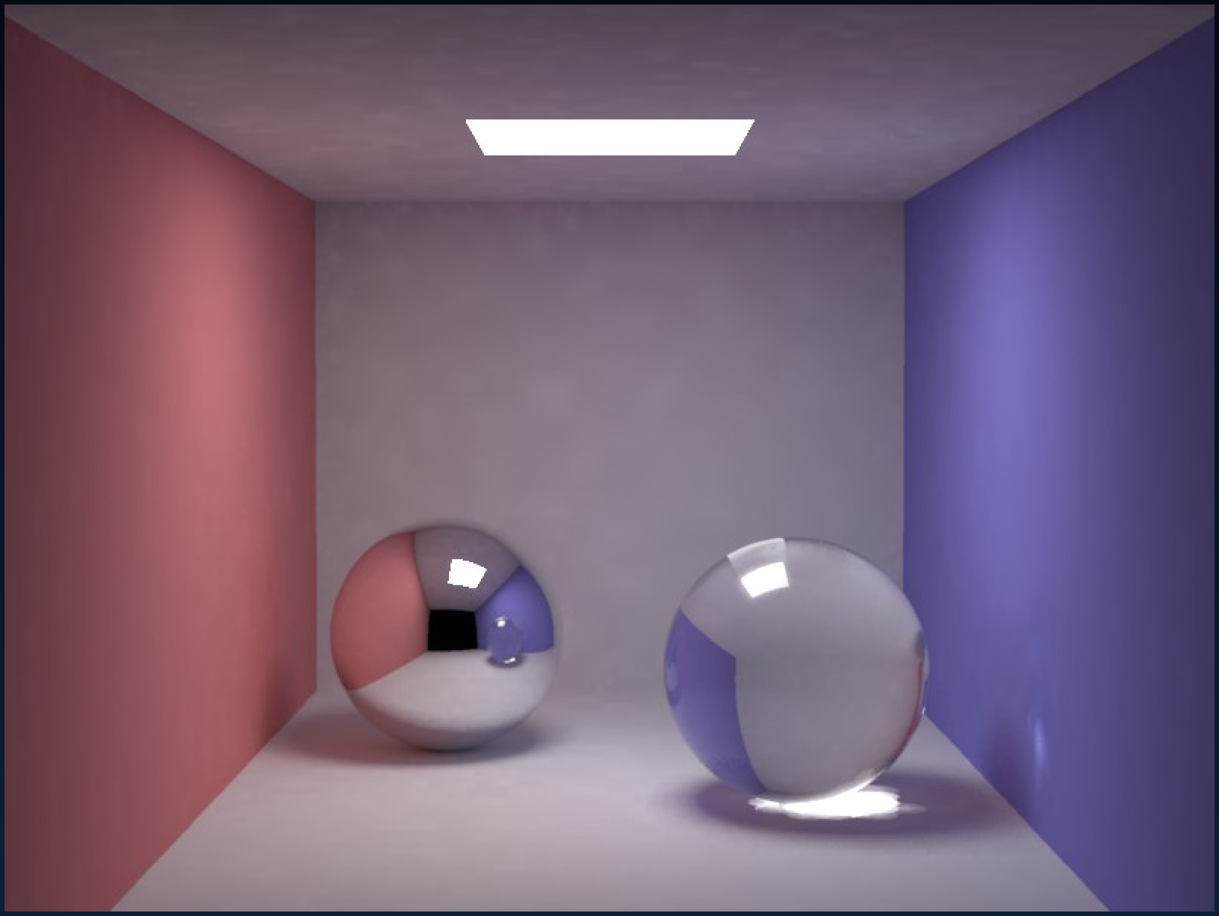
$$w(x) = \frac{1}{\epsilon(x)} \approx \frac{1}{\frac{||x-x_0||}{x_{avg}} + \sqrt{1 - \vec{N}(x) \cdot \vec{N}(x_0)}}$$

$$E_i(x) = \frac{\sum_i w_i(x) E(x_i)}{\sum_i w_i(x)}$$

Irradiance caching algorithm

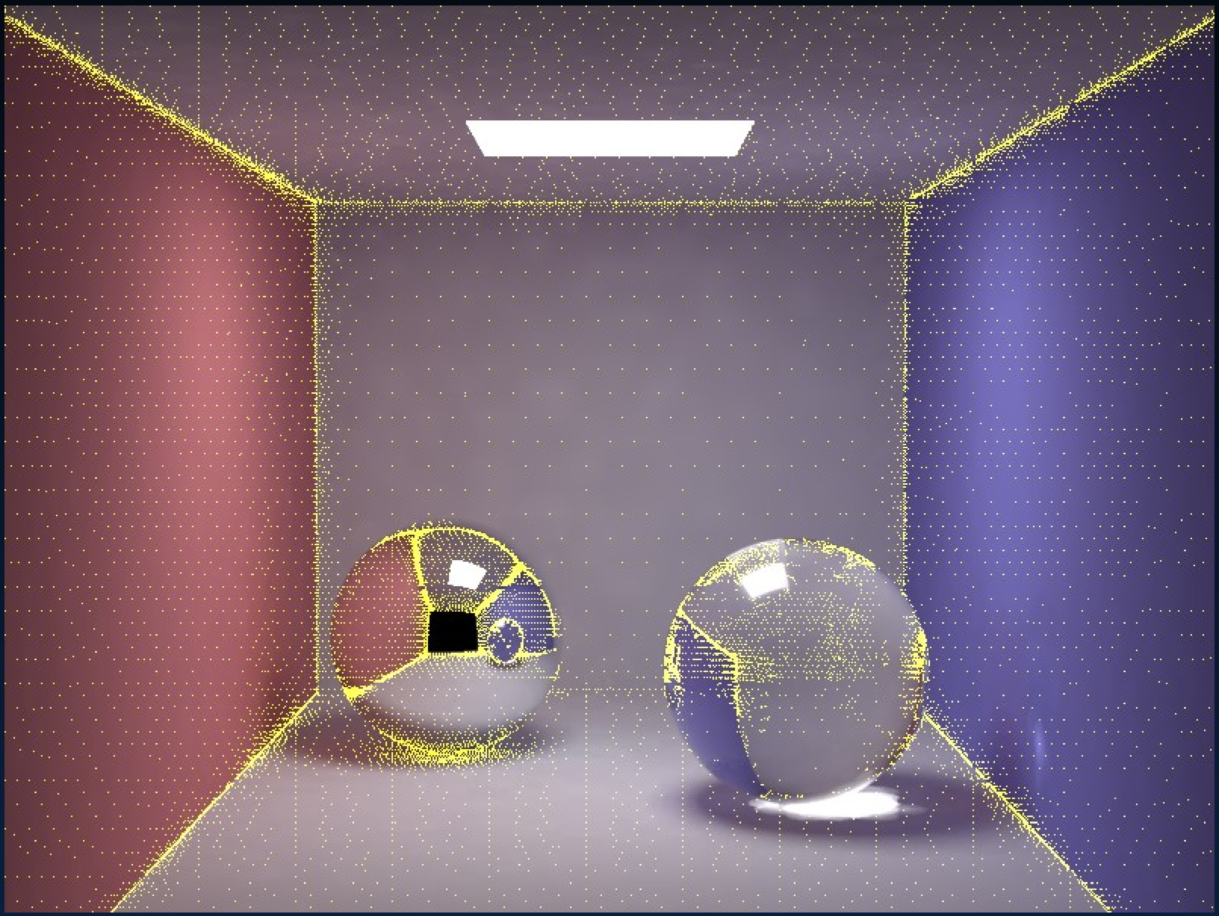
```
Find all irradiance samples with w(x) > q

if (samples found)
  interpolate
else
  compute new irradiance sample
```

Cornell box: irradiance gradients



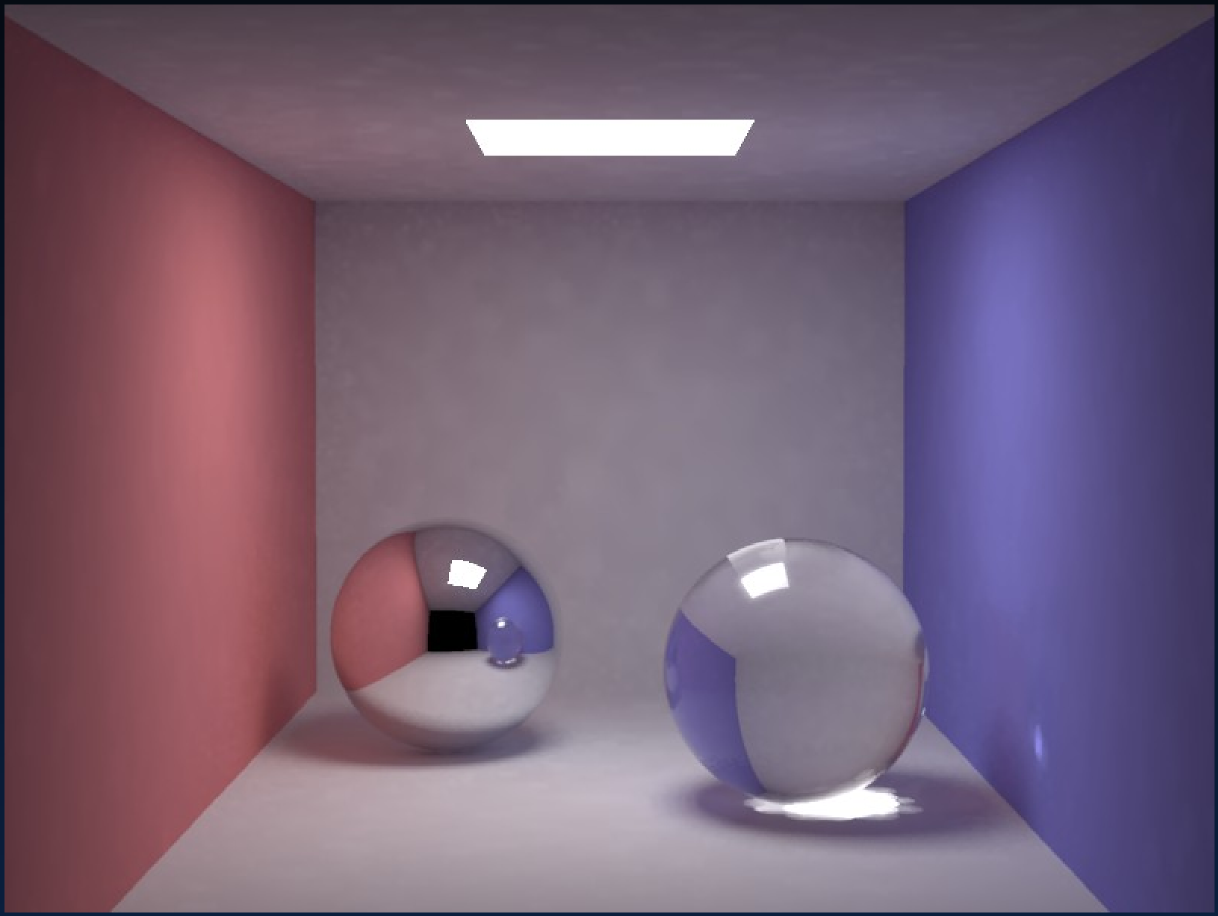
1000 sample rays, $w > 10$

Cornell box: irradiance cache positions



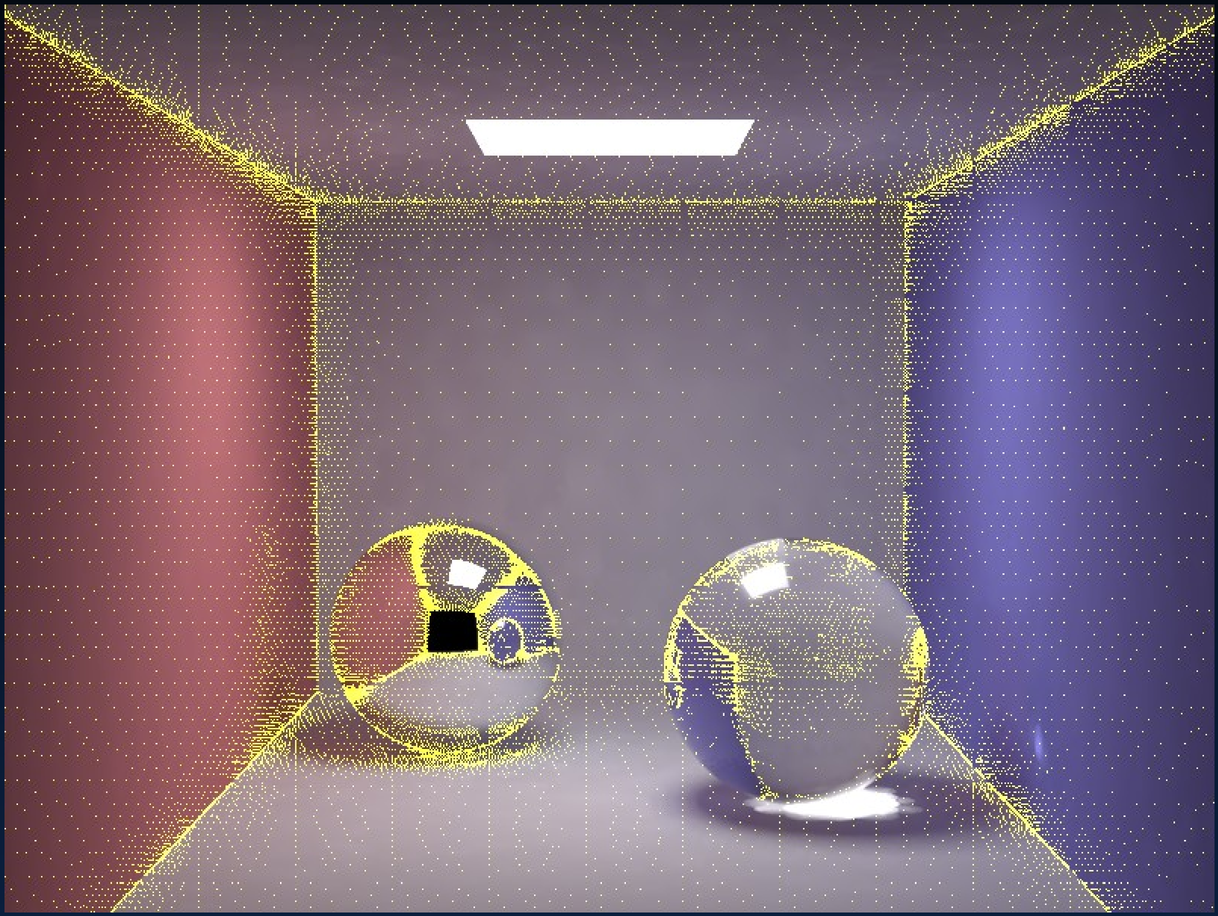
1000 sample rays, $w > 10$

Cornell box: irradiance gradients



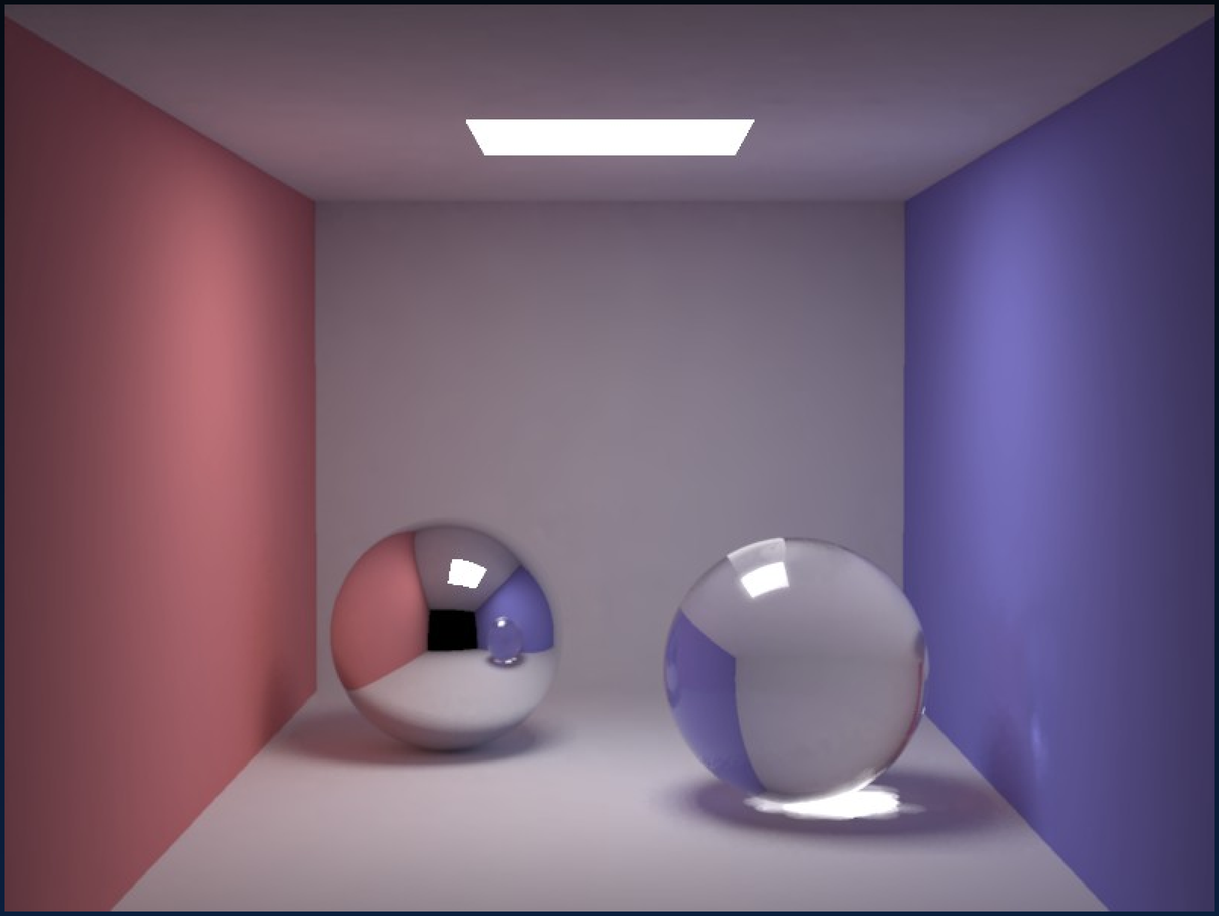
1000 sample rays, $w > 20$

Cornell box: irradiance cache positions



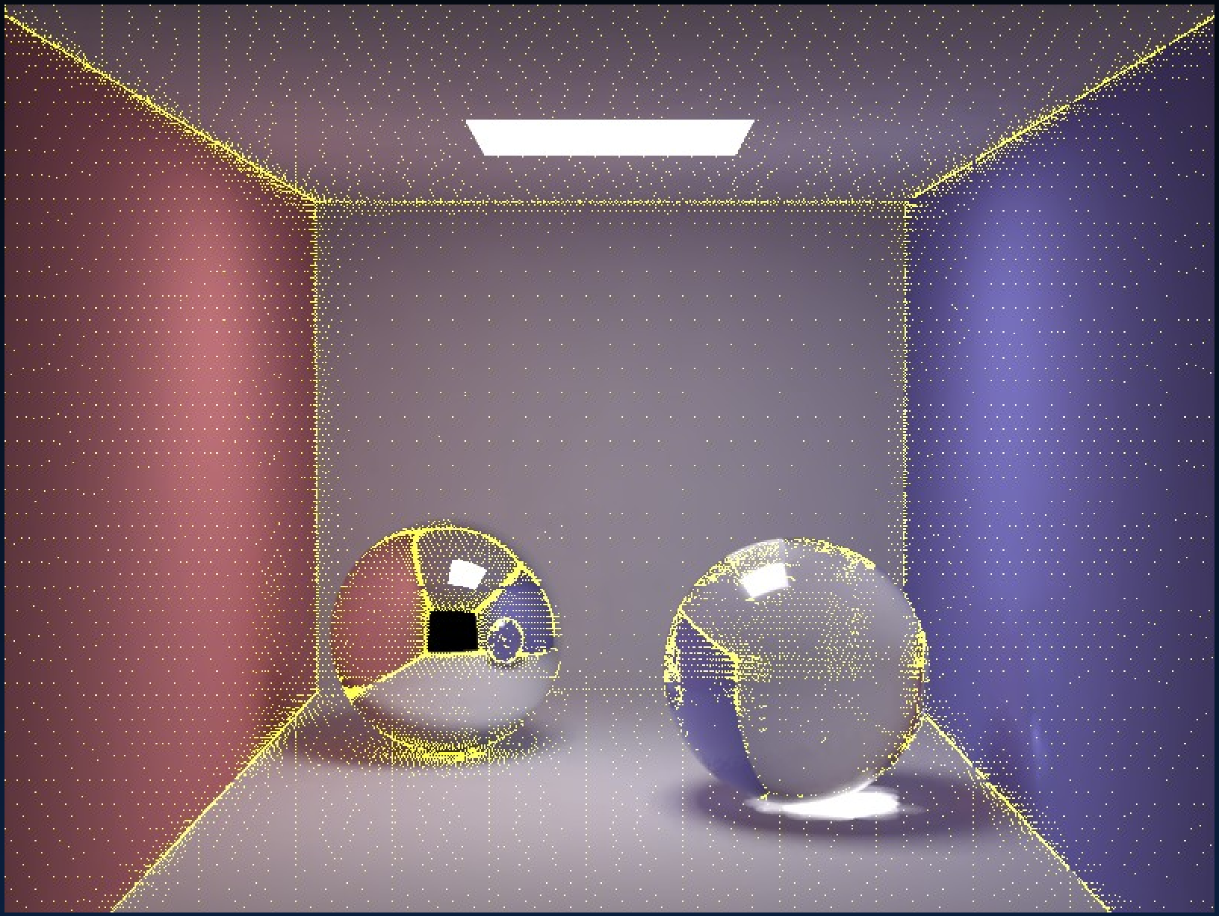
1000 sample rays, $w > 20$

Cornell box: irradiance gradients



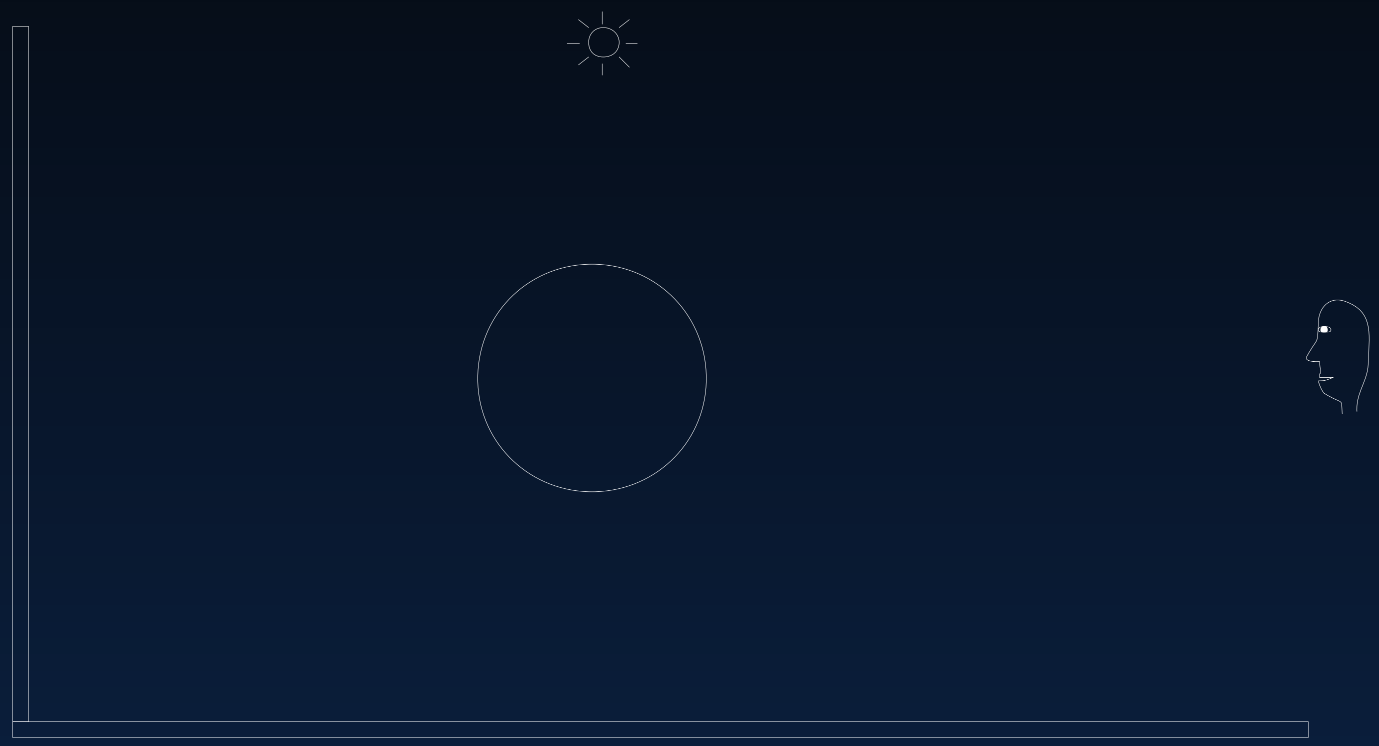
5000 sample rays, $w > 10$

Cornell box: irradiance cache positions



5000 sample rays, $w > 10$

A simple test scene



Photon Mapping

Two-pass method:

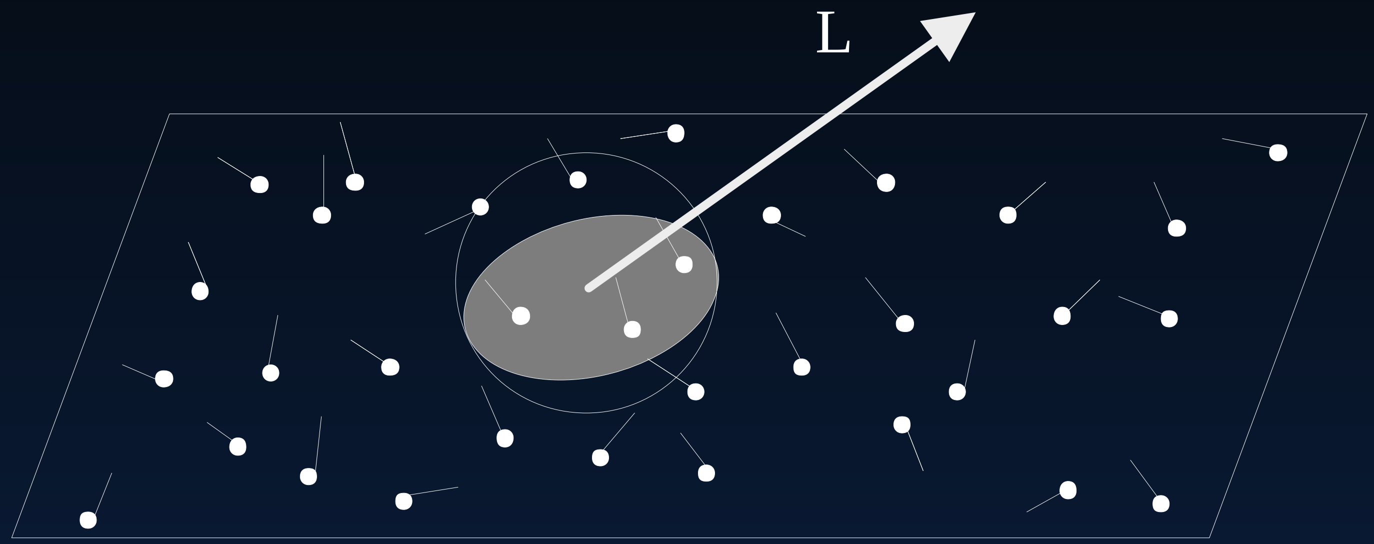
Pass 1 : Build a *photon map* using photon tracing

Pass 2 : Render the image using the photon map

Building the Photon Map: Photon Tracing

Photons

Radiance estimate



Radiance estimate

$$L(x,\vec{\omega}) \;=\; \int_\Omega f_r(x,\vec{\omega}',\vec{\omega})L'(x,\vec{\omega}')\cos\theta'\,d\omega$$

Radiance estimate

$$
\begin{aligned}
L(x,\vec{\omega}) &= \int_{\Omega} f_r(x,\vec{\omega}',\vec{\omega}) L'(x,\vec{\omega}') \cos\theta' \, d\omega \\
&= \int_{\Omega} f_r(x,\vec{\omega}',\vec{\omega}) \frac{d\Phi^2(x,\vec{\omega}')}{d\omega \, dA} \cos\theta' d\omega \\
&= \int_{\Omega} f_r(x,\vec{\omega}',\vec{\omega}) \frac{d\Phi^2(x,\vec{\omega}')}{dA} \cos\theta'
\end{aligned}
$$

Radiance estimate

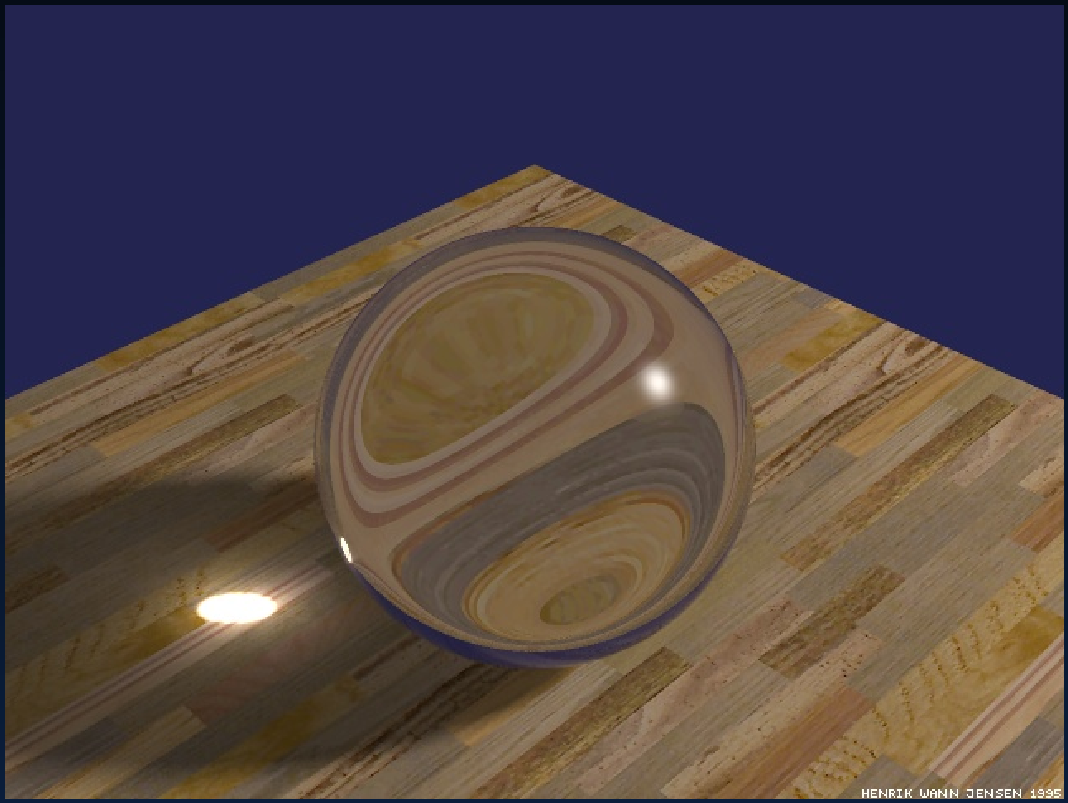
$$
\begin{aligned}
L(x,\vec{\omega}) &= \int_\Omega f_r(x,\vec{\omega}',\vec{\omega}) L'(x,\vec{\omega}') \cos\theta'\, d\omega \\
&= \int_\Omega f_r(x,\vec{\omega}',\vec{\omega}) \frac{d\Phi^2(x,\vec{\omega}')}{d\omega\, dA} \cos\theta' d\omega \\
&= \int_\Omega f_r(x,\vec{\omega}',\vec{\omega}) \frac{d\Phi^2(x,\vec{\omega}')}{dA} \cos\theta' \\
&\approx \sum_{p=1}^{n} f_r(x,\vec{\omega}'_p,\vec{\omega}) \frac{\Delta\Phi_p(x,\vec{\omega}'_p)}{\pi r^2}
\end{aligned}
$$

The photon map datastructure

The photons are stored in a balanced kd-tree

```
struct photon = {
  float position[3];
  rgbe power;           // power packed as 4 bytes
  char phi, theta;      // incoming direction
  short flags;
}
```

Caustic from a glass sphere

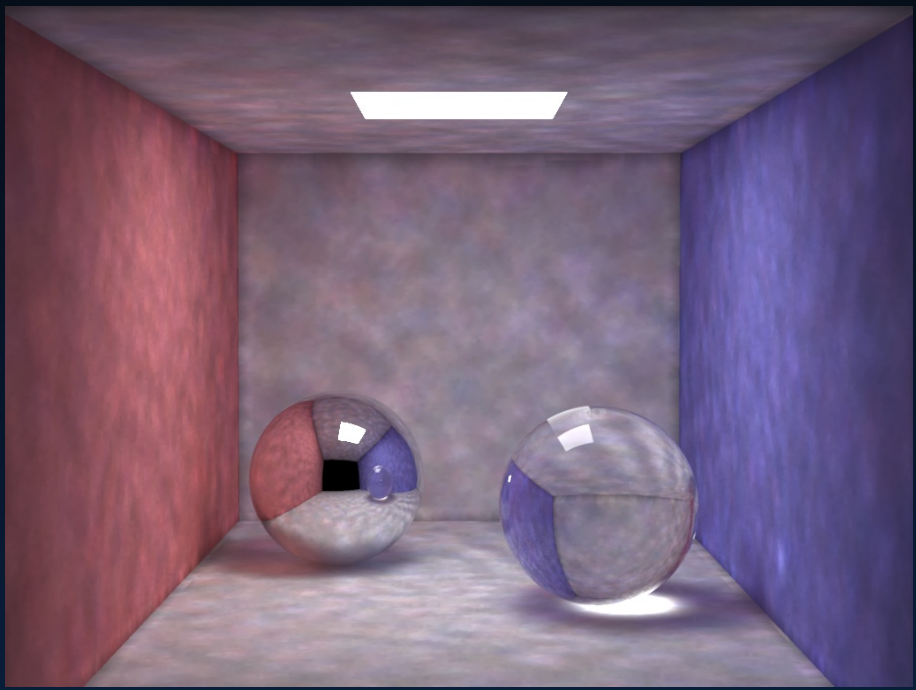


30000 photons / 50 photons in radiance estimate

Caustic on a glossy surface

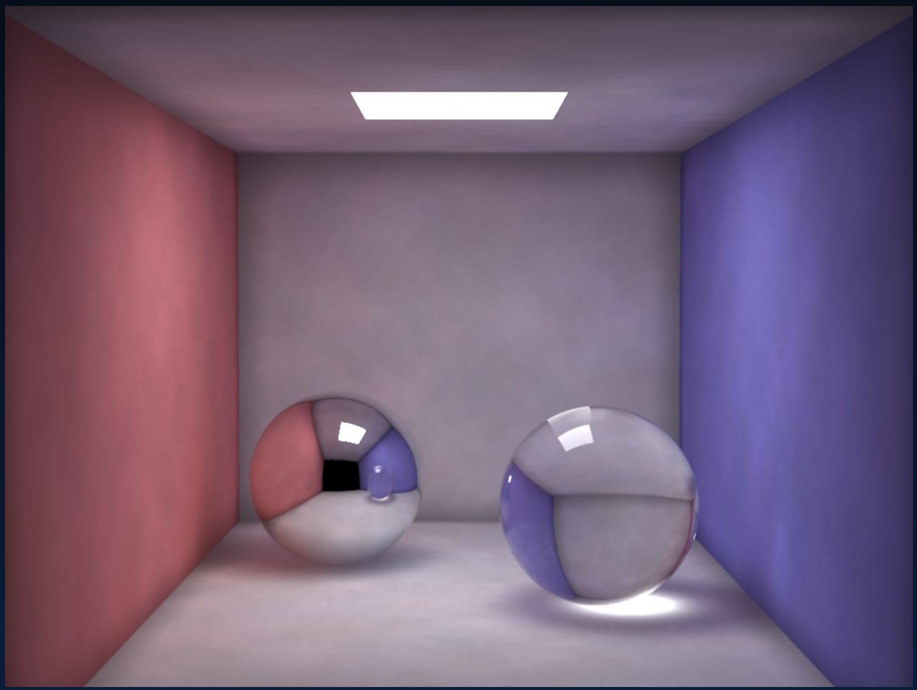
340000 photons / ≈ 100 photons in radiance estimate

Direct visualization of the radiance estimate



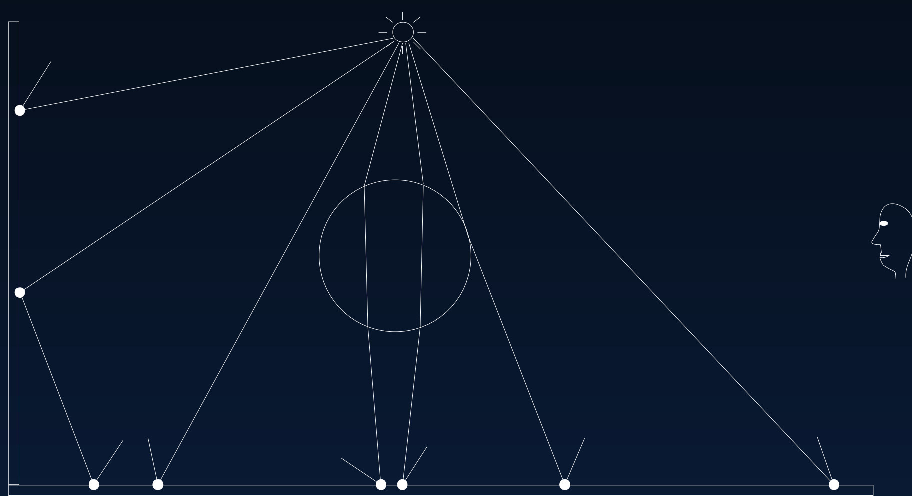
200000 photons / 50 photons in radiance estimate

Direct visualization of the radiance estimate

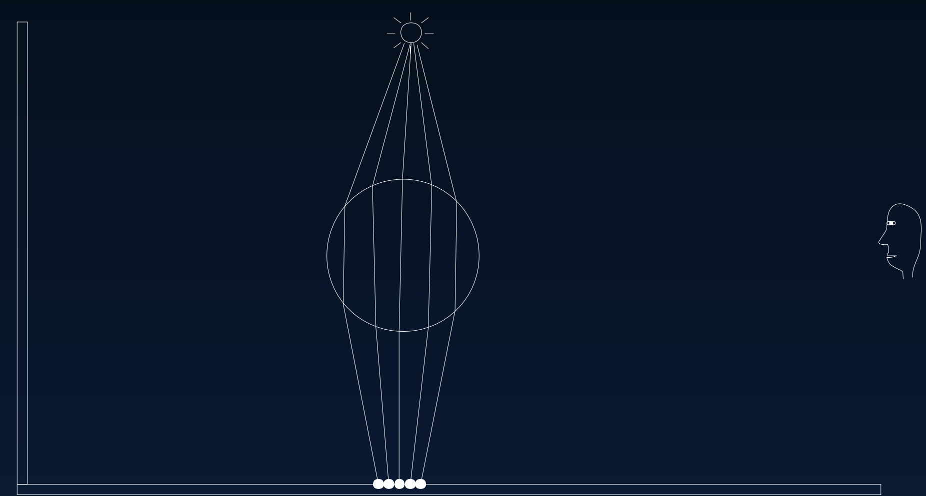


200000 photons / 500 photons in radiance estimate

Two photon maps

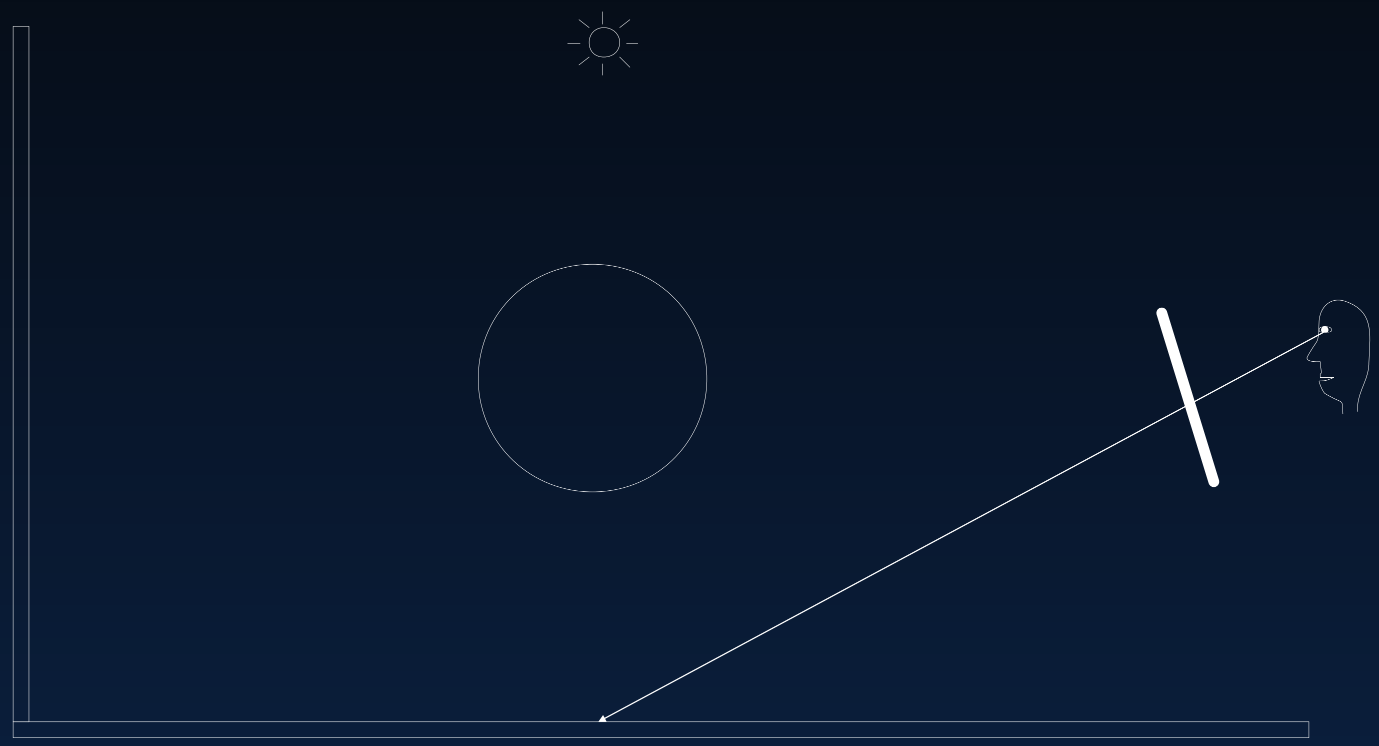


global photon map



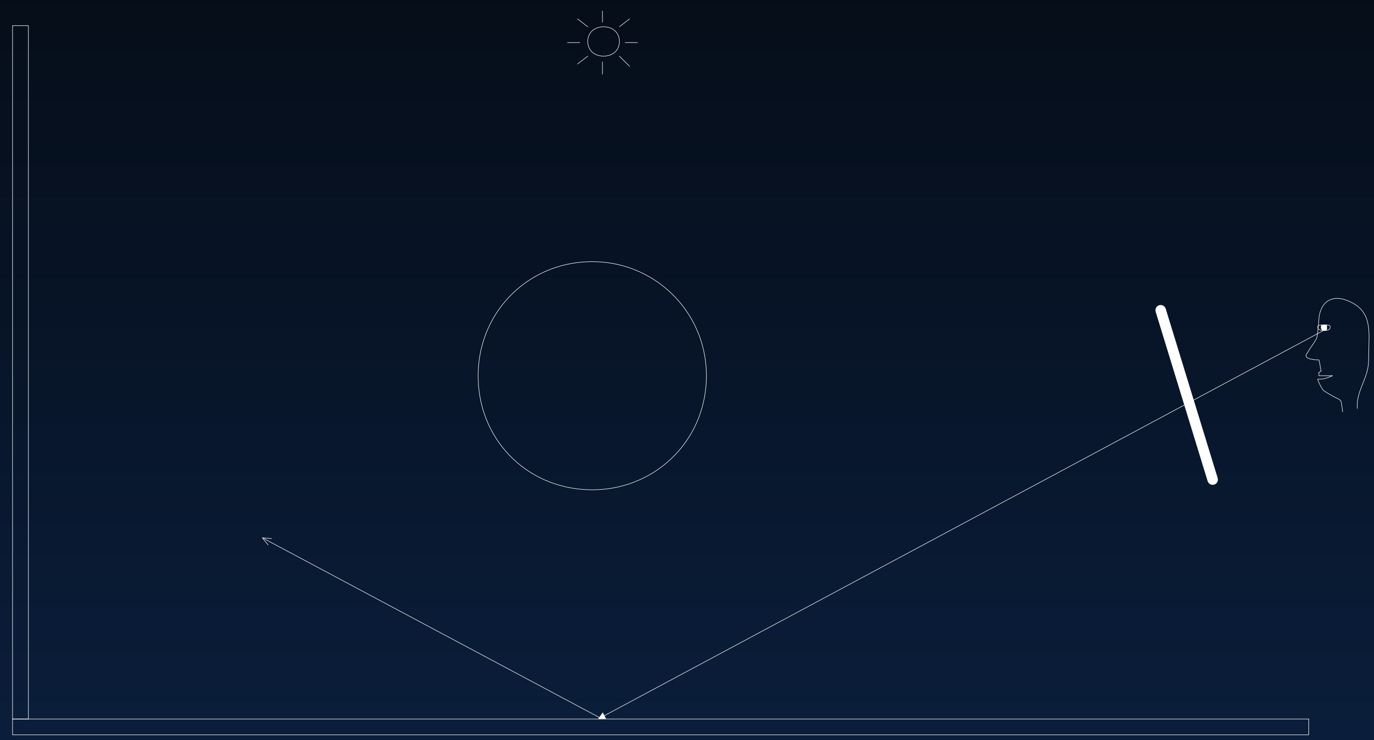
caustics photon map

Rendering

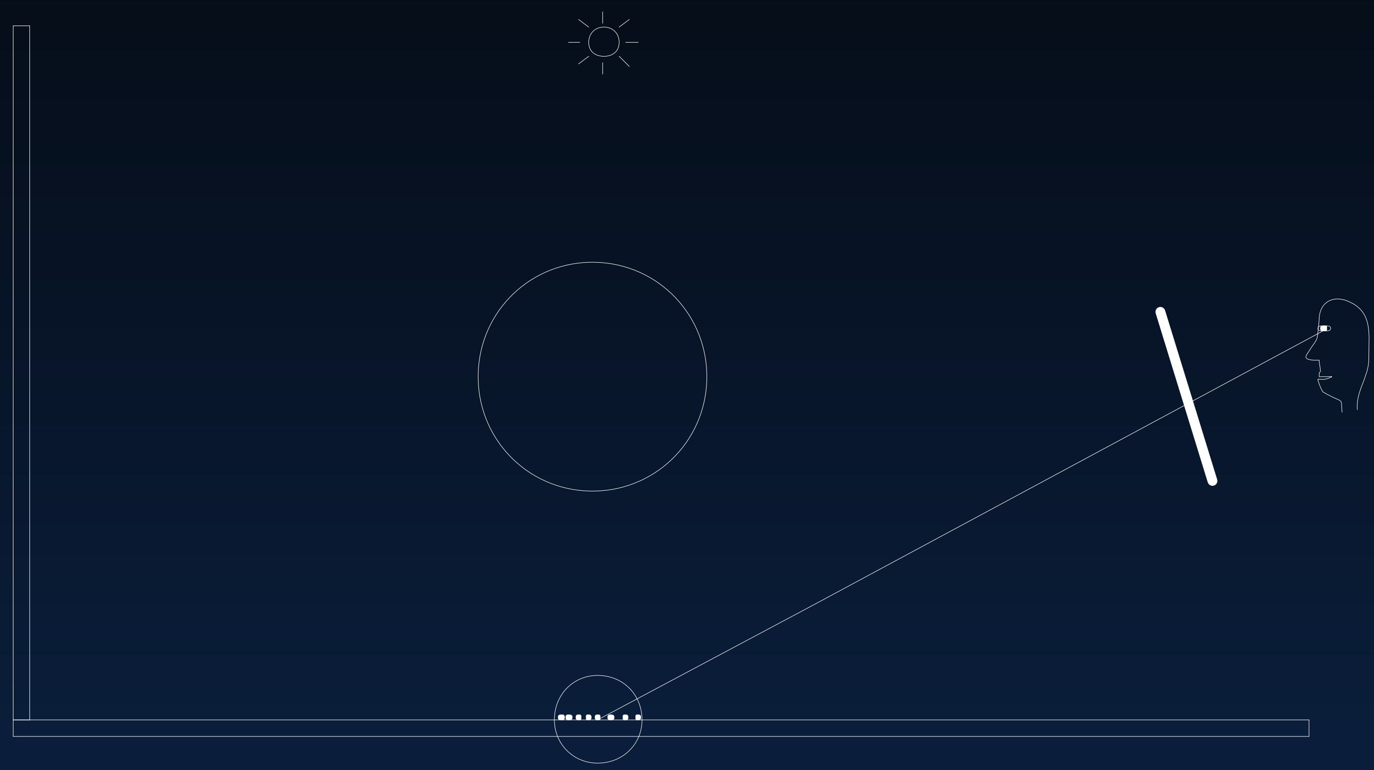


Rendering: direct illumination

Rendering: specular reflection



Rendering: caustics



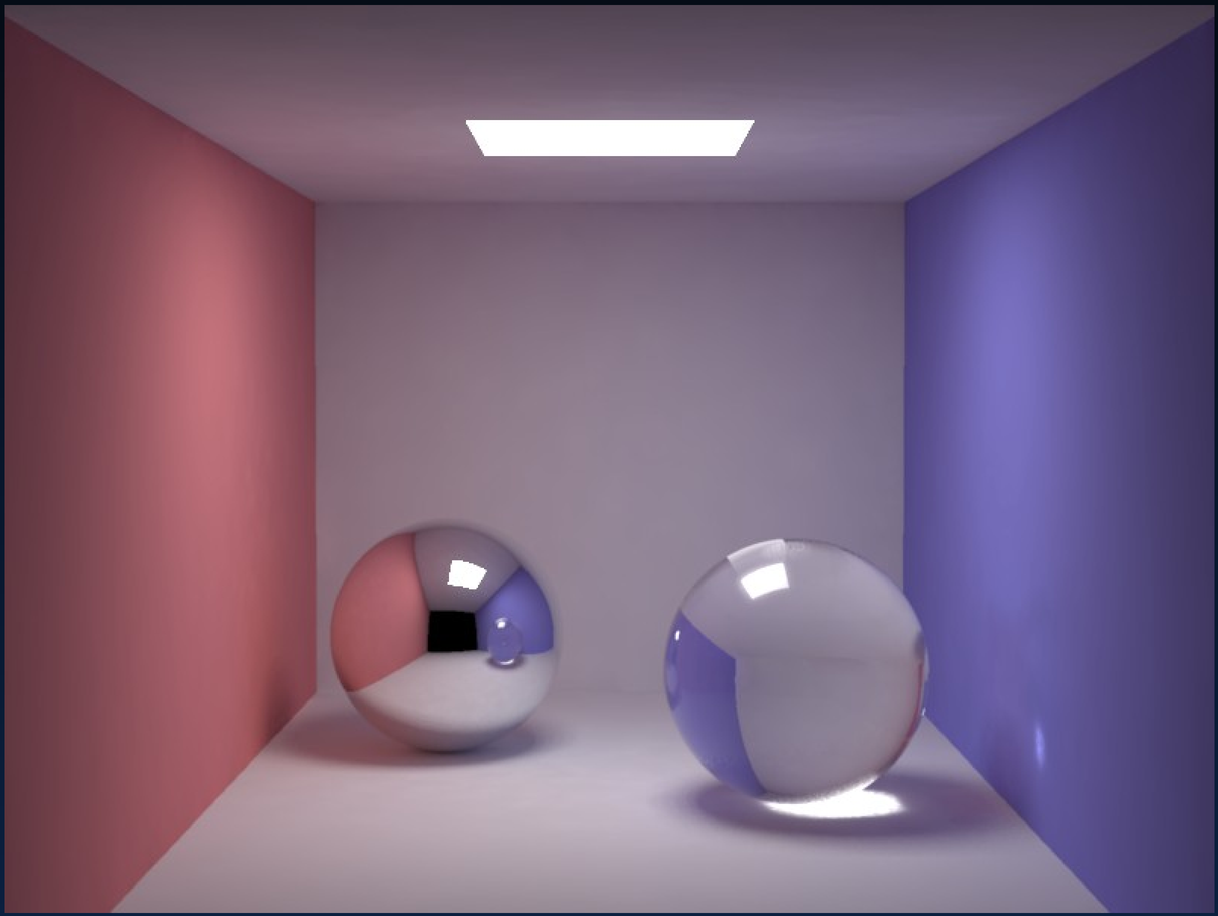
Rendering: indirect illumination

Rendering Equation Solution

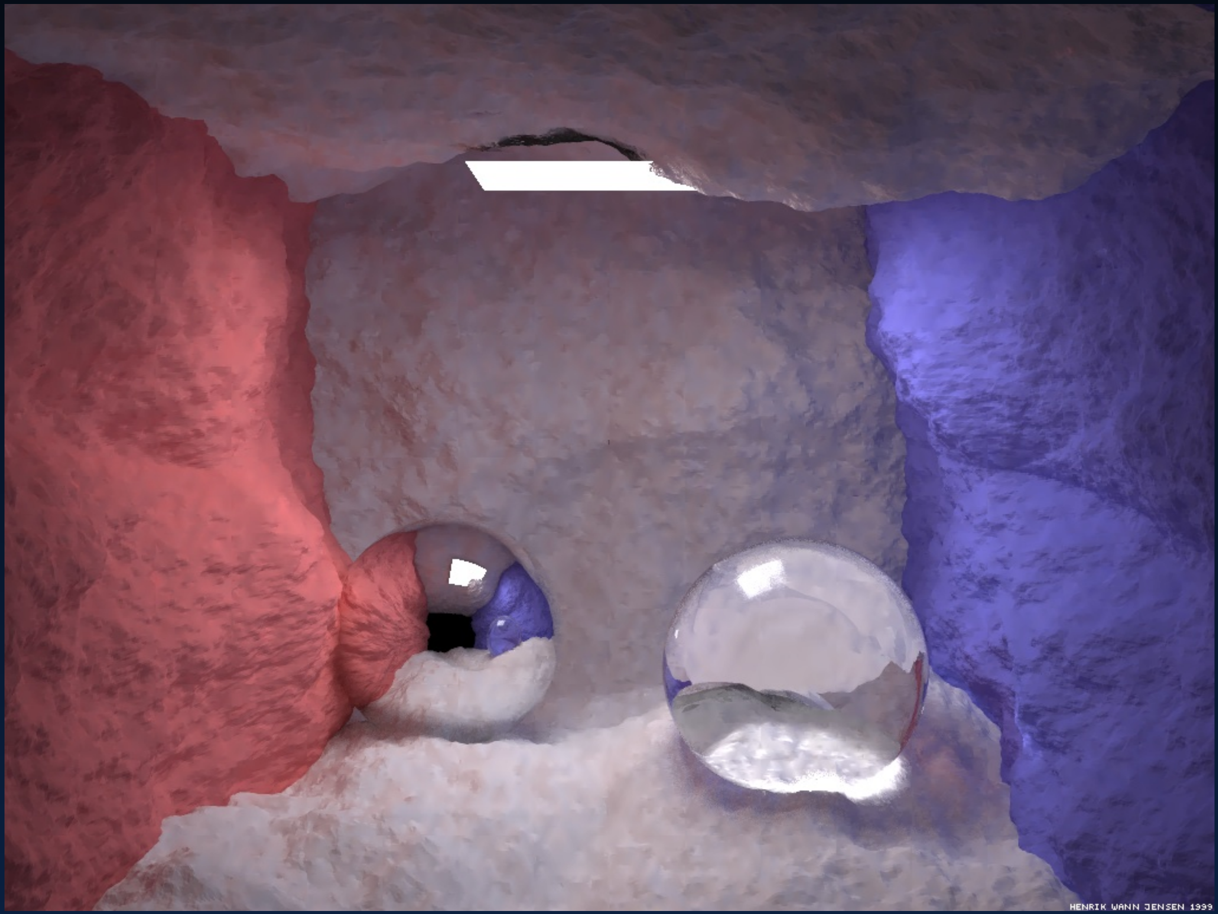
$$
\begin{aligned}
L_r(x,\vec{\omega}) &= \int_{\Omega_x} f_r(x,\vec{\omega}',\vec{\omega}) L_i(x,\vec{\omega}') \cos\theta_i \, d\omega_i' \\
&= \int_{\Omega_x} f_r(x,\vec{\omega}',\vec{\omega}) L_{i,l}(x,\vec{\omega}') \cos\theta_i \, d\omega_i' + \\
&\quad \int_{\Omega_x} f_{r,s}(x,\vec{\omega}',\vec{\omega}) (L_{i,c}(x,\vec{\omega}') + L_{i,d}(x,\vec{\omega}')) \cos\theta_i \, d\omega_i' + \\
&\quad \int_{\Omega_x} f_{r,d}(x,\vec{\omega}',\vec{\omega}) L_{i,c}(x,\vec{\omega}') \cos\theta_i \, d\omega_i' + \\
&\quad \int_{\Omega_x} f_{r,d}(x,\vec{\omega}',\vec{\omega}) L_{i,d}(x,\vec{\omega}') \cos\theta_i \, d\omega_i' \, .
\end{aligned}
$$

Rendering Equation Solution

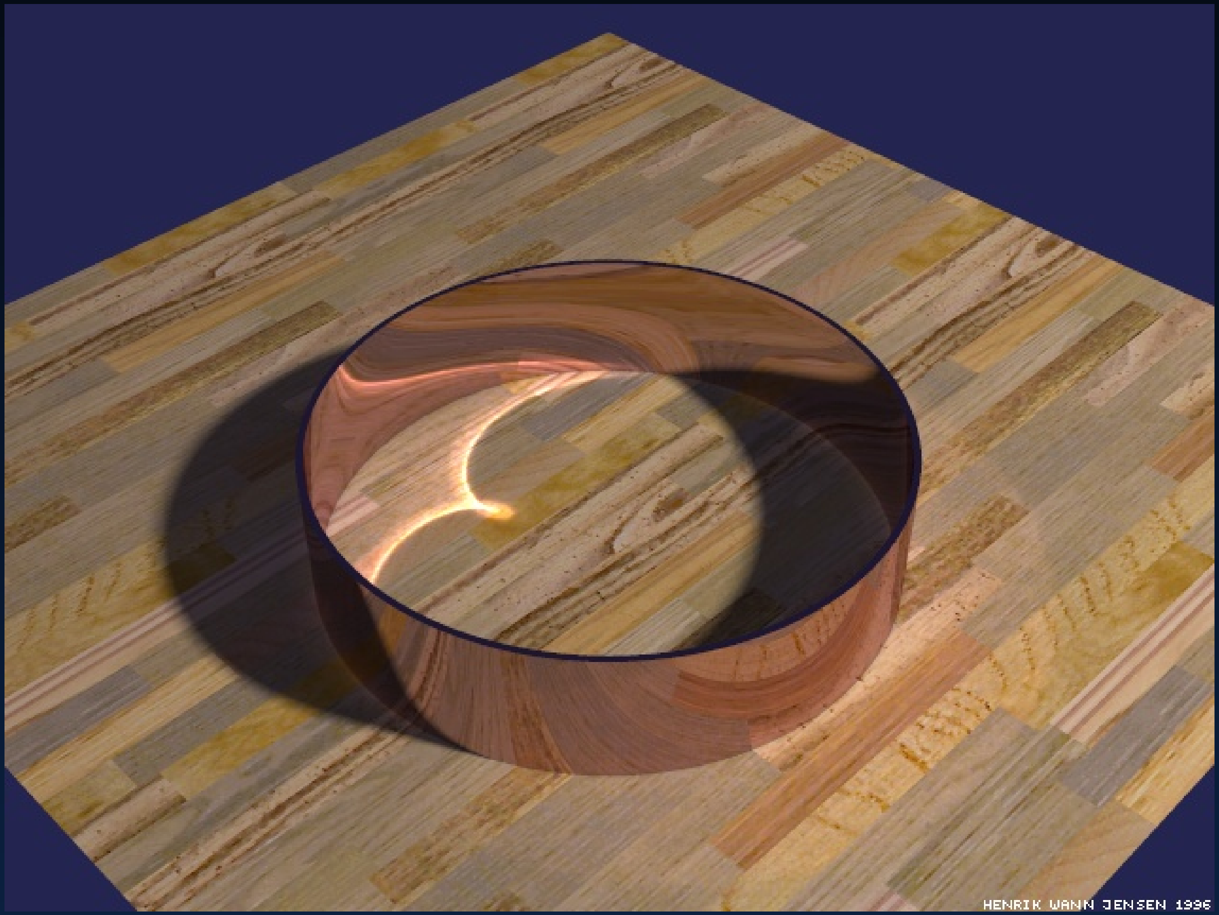
Cornell box



Fractal Cornell box

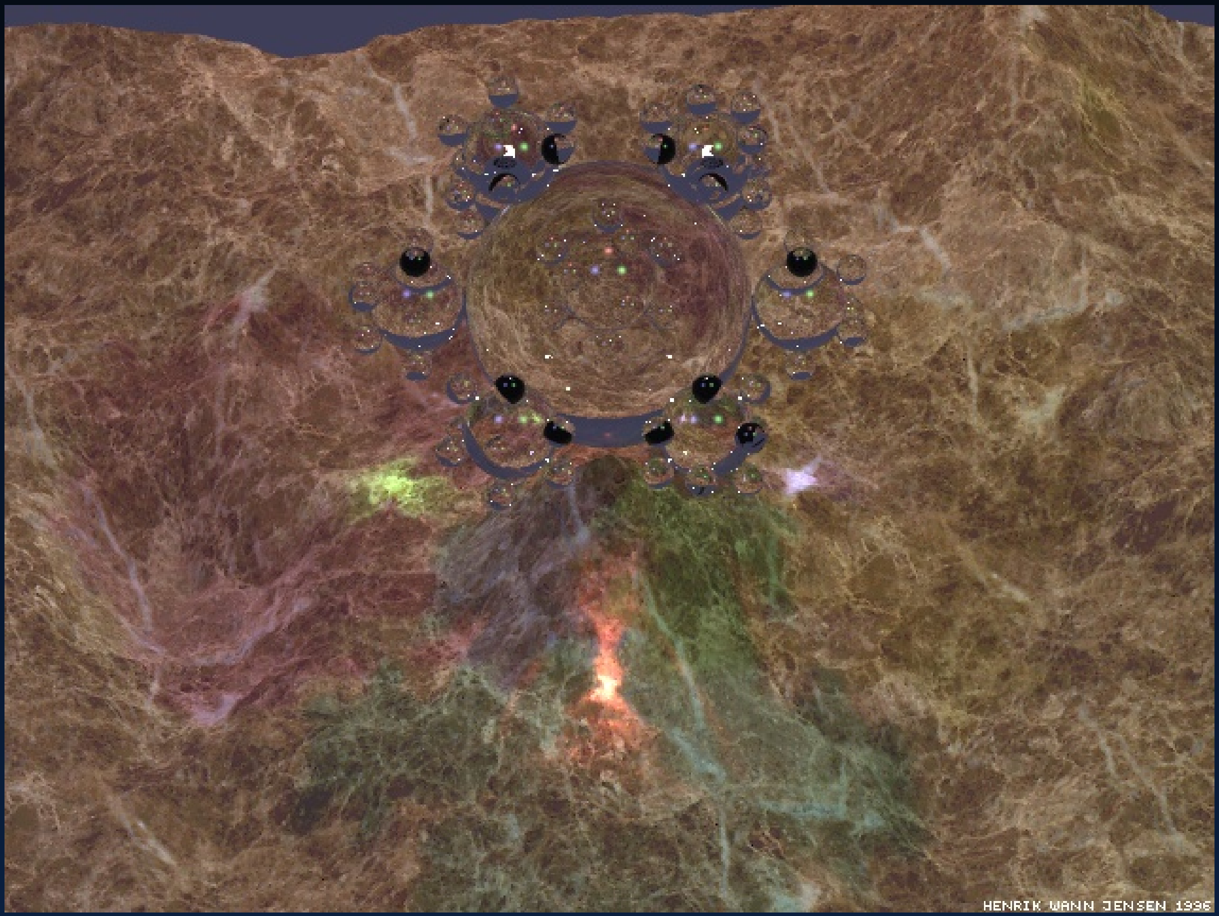


Metalring caustic

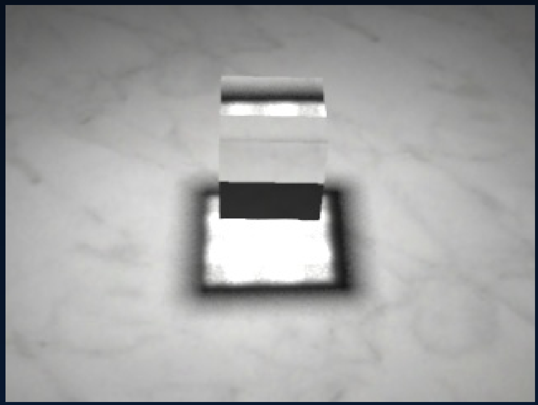


Cognac glass

Sphereflake caustic



Cube caustic



Mies house (swimmingpool)

Mies house (2pm)

Mies house (7pm)

David (subsurface scattering)

Diana the Huntress

Diana the Huntress: subsurface scattering