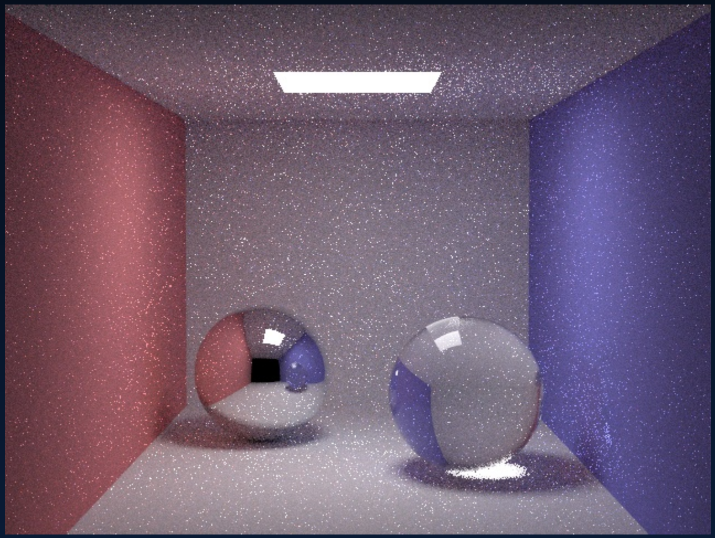
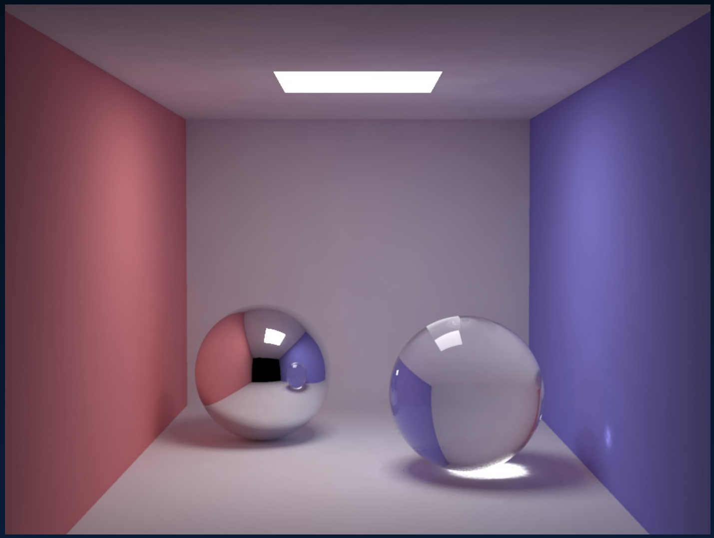
Biased Monte Carlo Ray Tracing

Filtering, Irradiance Caching, and Photon Mapping

 → 

Henrik Wann Jensen

Stanford University

May 23, 2002

Unbiased and Consistent

Unbiased estimator:

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

Consistent estimator:

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

Unbiased and Consistent

Unbiased estimator:

$$\frac{1}{N}\sum_{i=1}^{N} f(\xi_i)$$

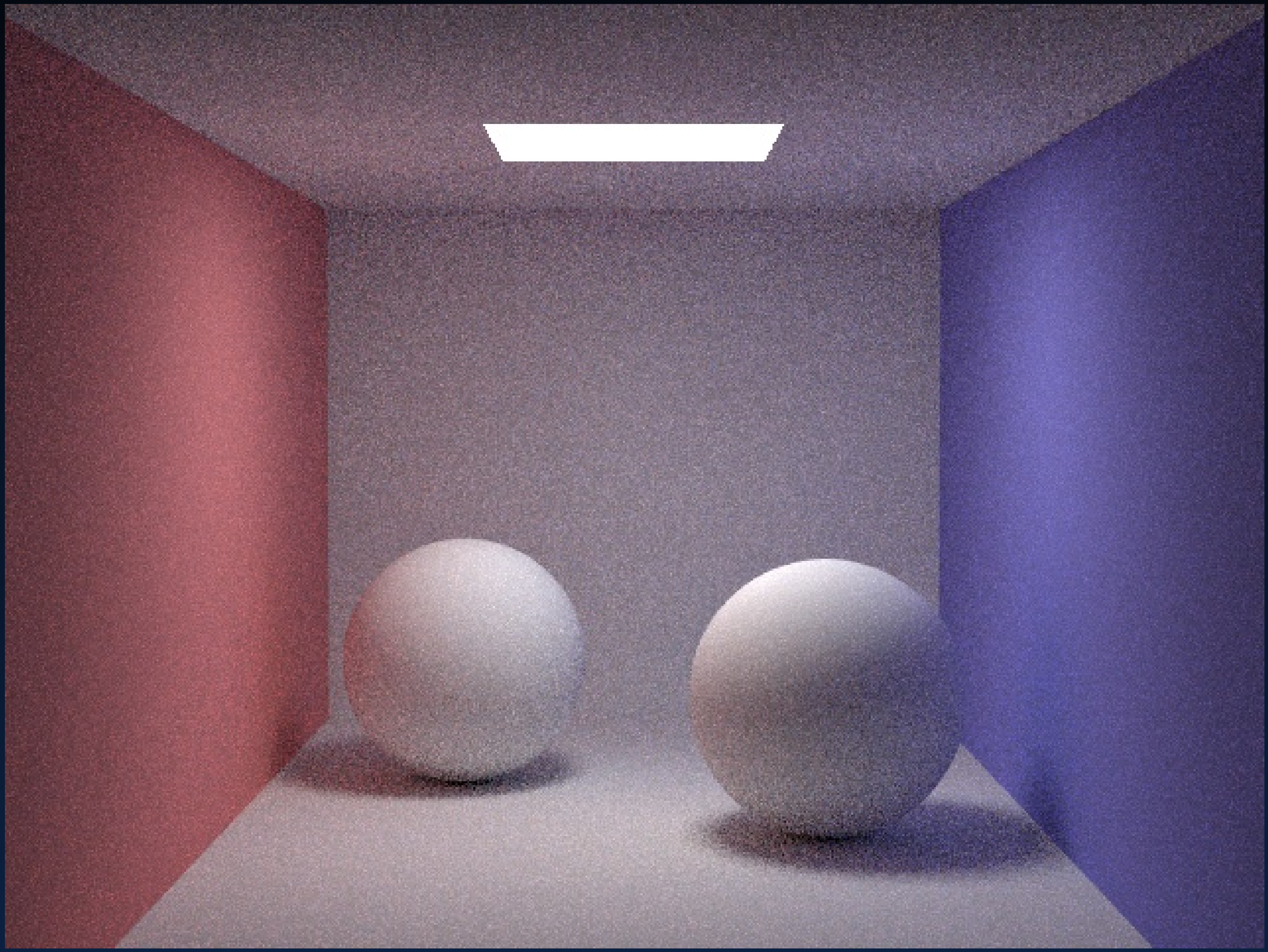
Consistent estimator:

$$\frac{1}{N+1}\sum_{i=1}^{N} f(\xi_i)$$

Unbiased Methods

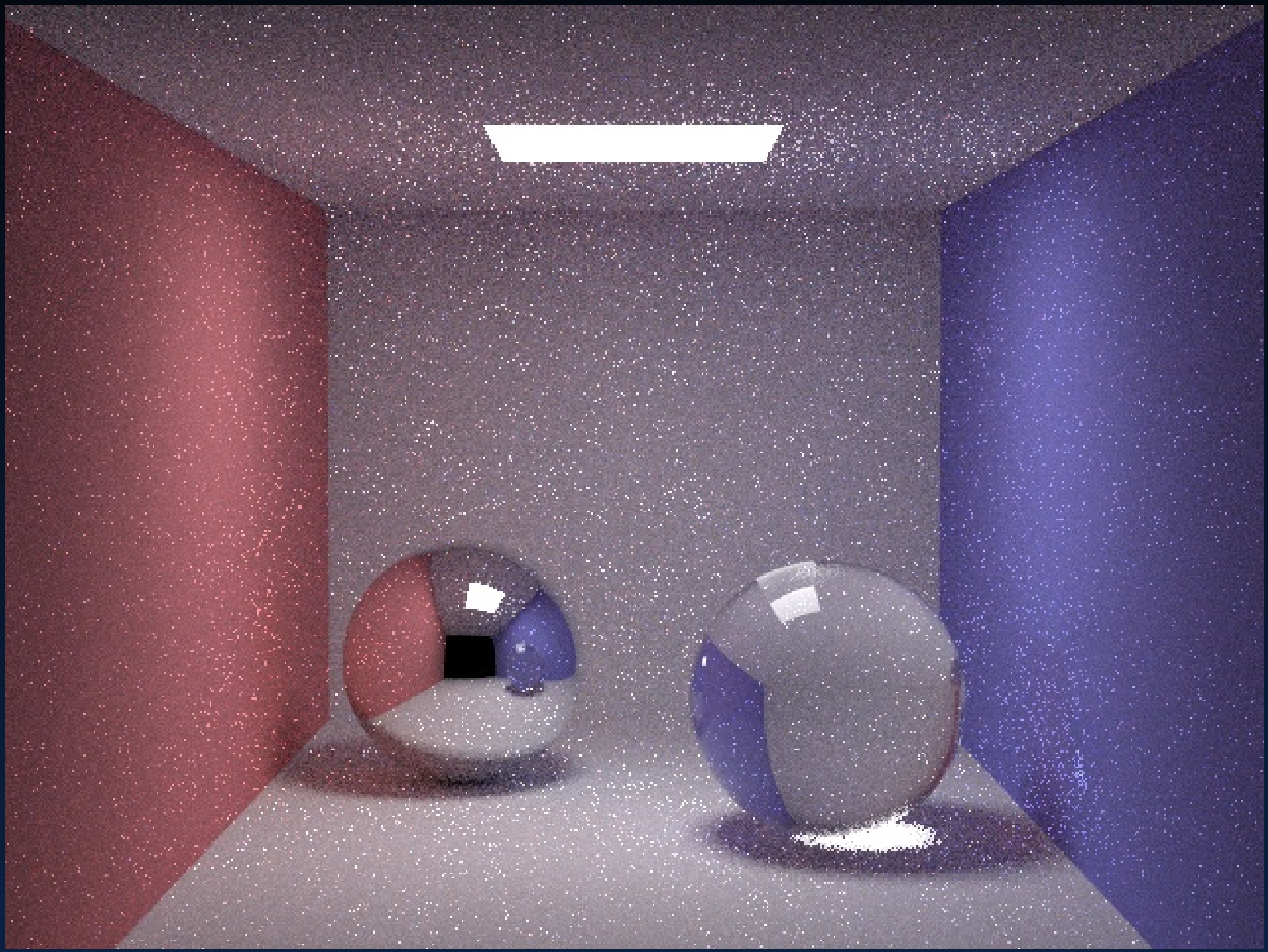
- Variance (noise) is the only error
- This error can be analyzed using the variance (i.e. 95% of samples are within 2% of the correct result)

Path Tracing (Unbiased)



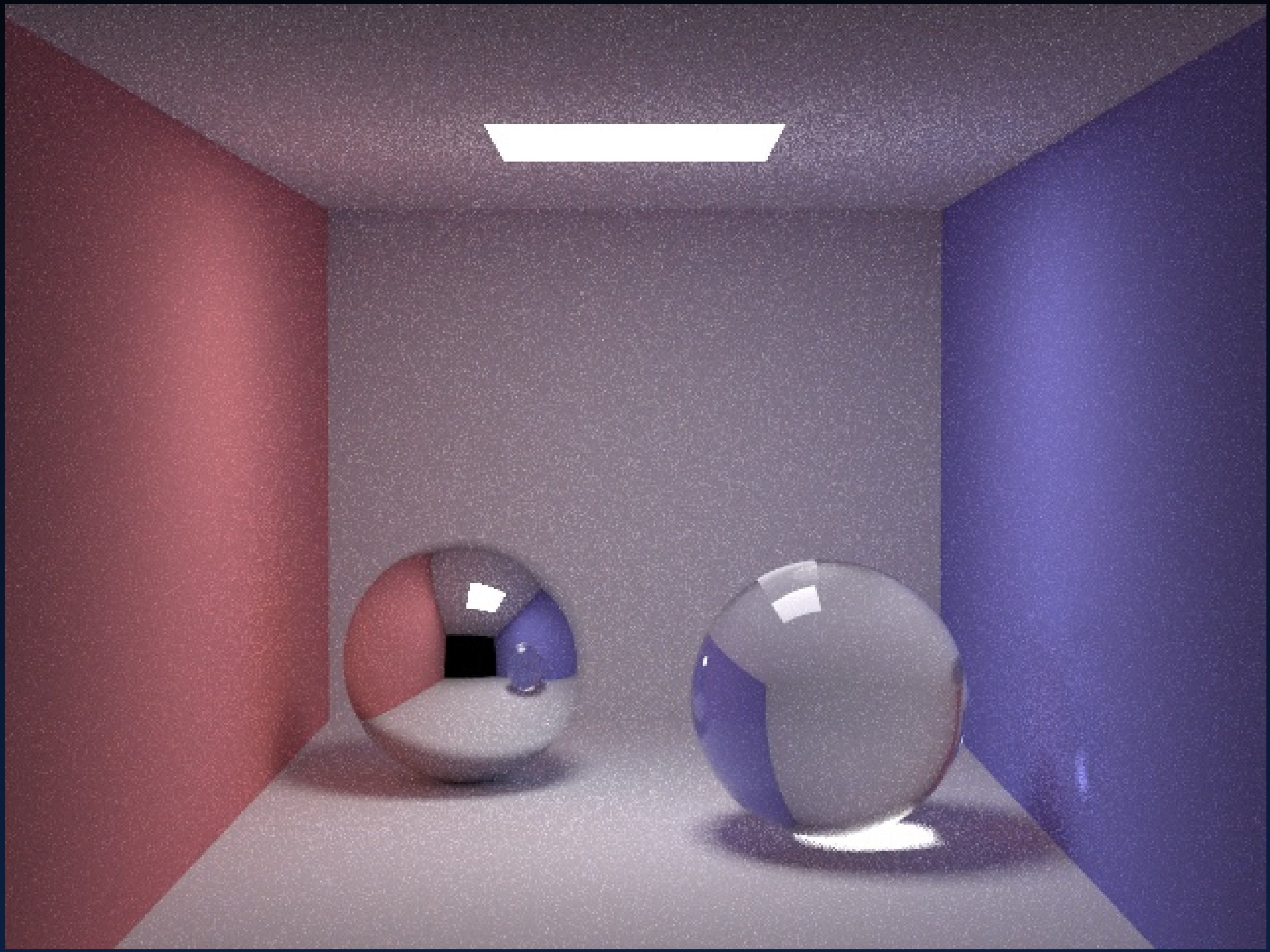
10 paths/pixel

Path Tracing (Unbiased)



10 paths/pixel

Path Tracing (Unbiased)



100 paths/pixel

How Can We Remove This Noise

The World is Diffuse!

Arnold Rendering

The World is Diffuse!

Arnold Rendering

The World is Diffuse!

Arnold Rendering

Noise Reduction/Removal

- More samples (slow convergence, $\sigma \propto 1/\sqrt{N}$)

Noise Reduction/Removal

- More samples (slow convergence, $\sigma \propto 1/\sqrt{N}$)
- Better sampling (stratified, importance, qmc etc.)

Noise Reduction/Removal

- More samples (slow convergence, $\sigma \propto 1/\sqrt{N}$)
- Better sampling (stratified, importance, qmc etc.)
- Adaptive sampling

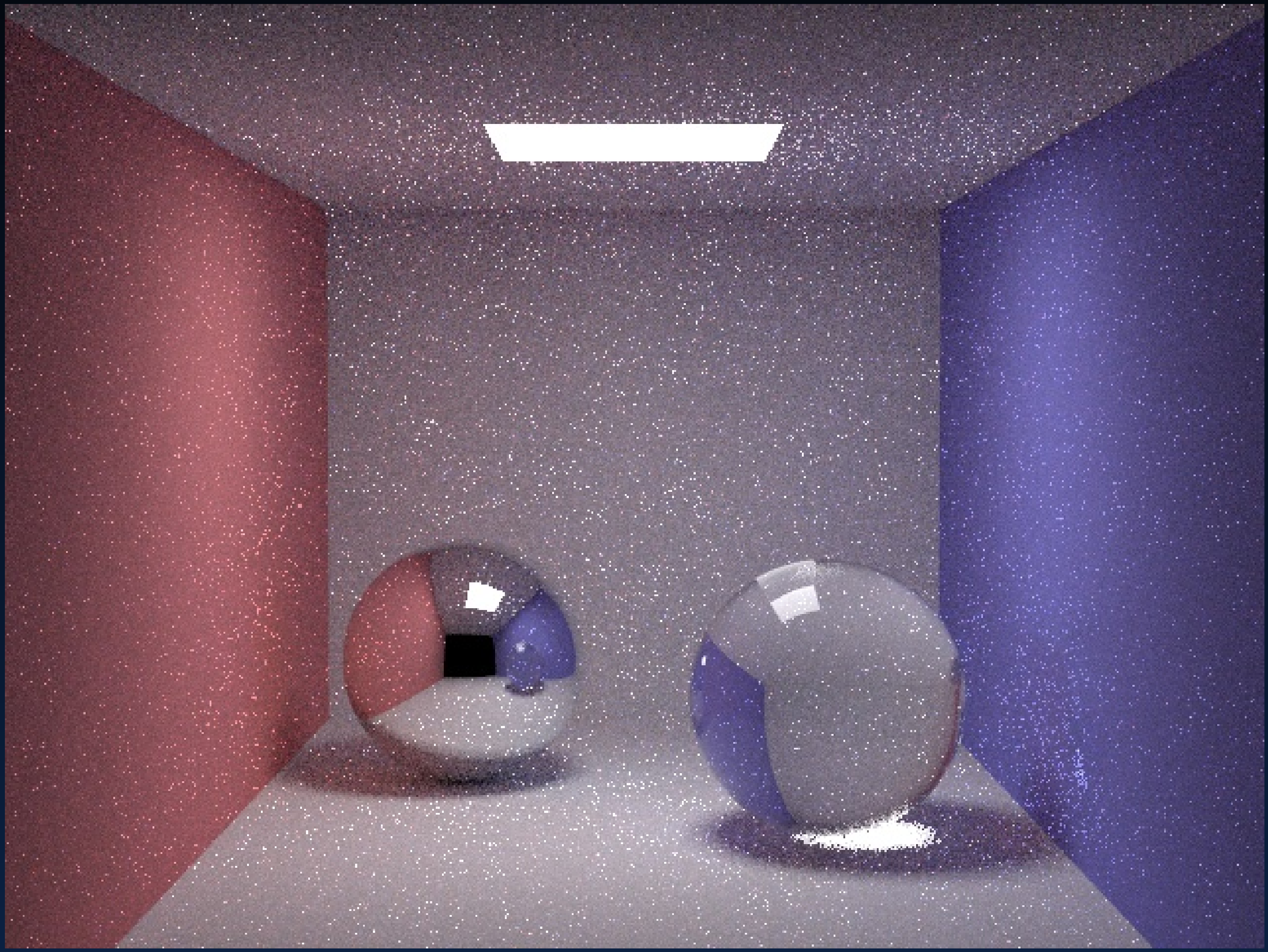
Noise Reduction/Removal

- More samples (slow convergence, $\sigma \propto 1/\sqrt{N}$)
- Better sampling (stratified, importance, qmc etc.)
- Adaptive sampling
- Filtering

Noise Reduction/Removal

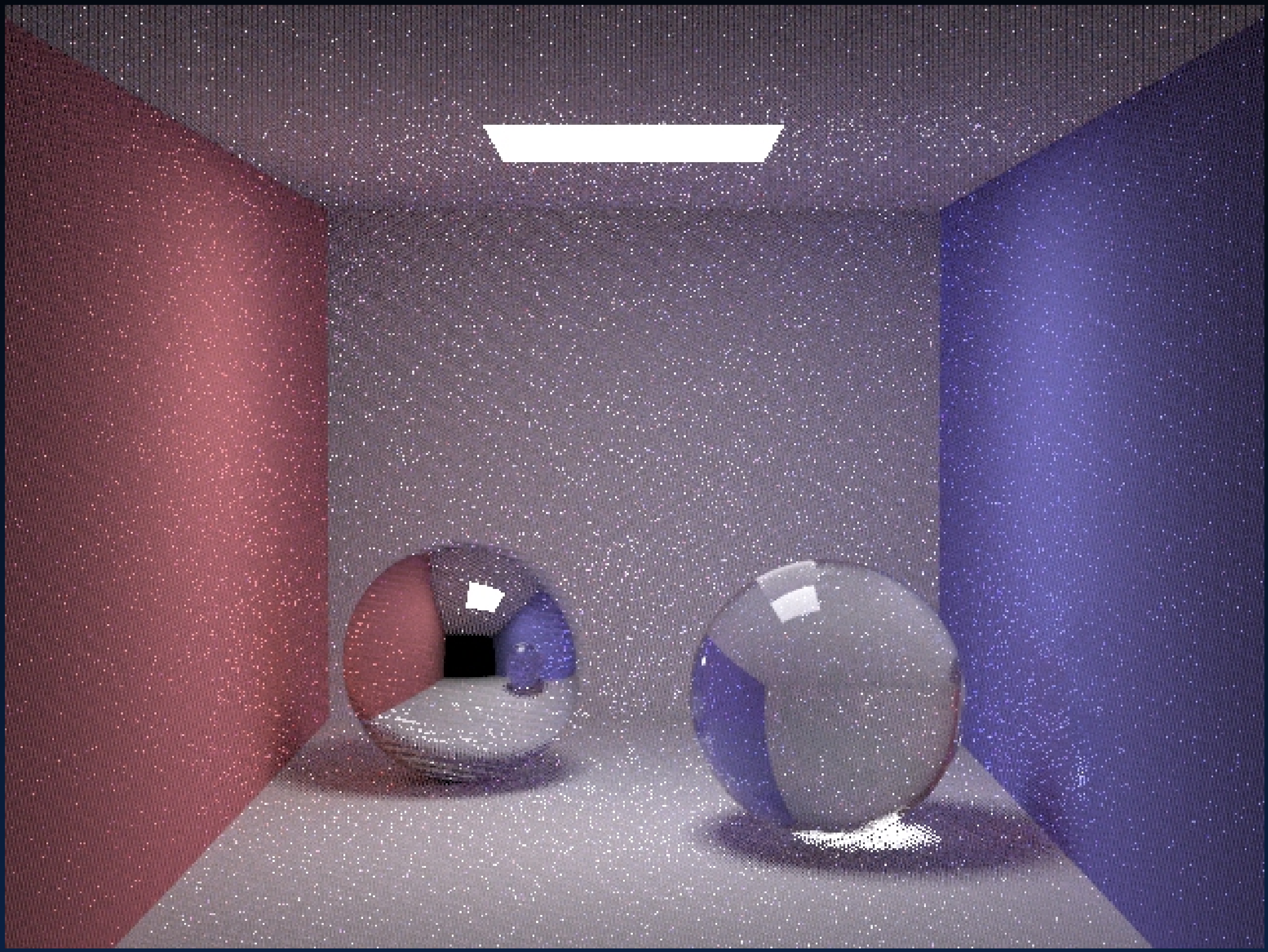
- More samples (slow convergence, $\sigma \propto 1/\sqrt{N}$)
- Better sampling (stratified, importance, qmc etc.)
- Adaptive sampling
- Filtering
- Caching and interpolation

Stratified Sampling



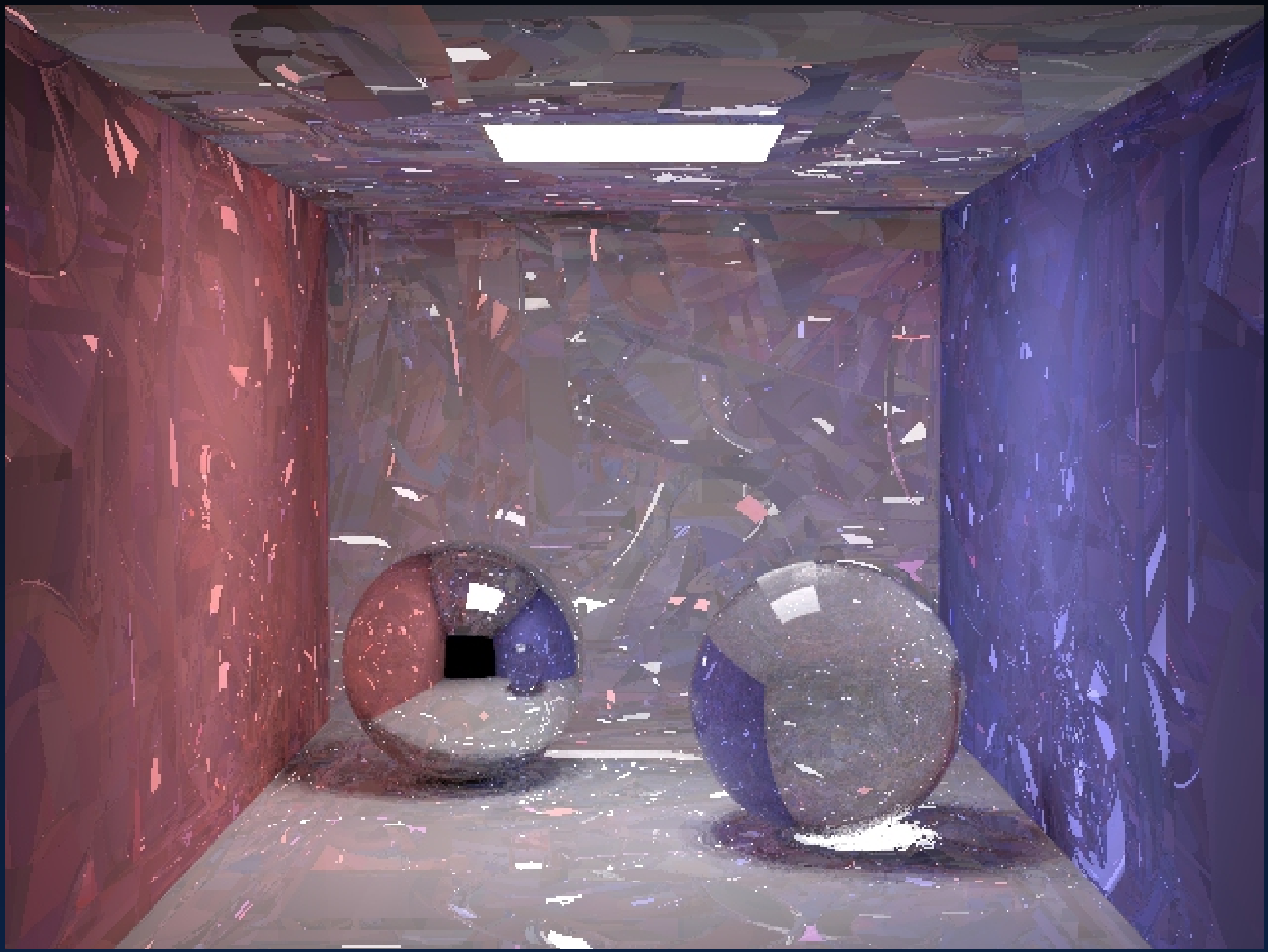
Latin Hypercube: 10 paths/pixel

Quasi Monte-Carlo



Halton-Sequence: 10 paths/pixel

Fixed (Random) Sequence



10 paths/pixel

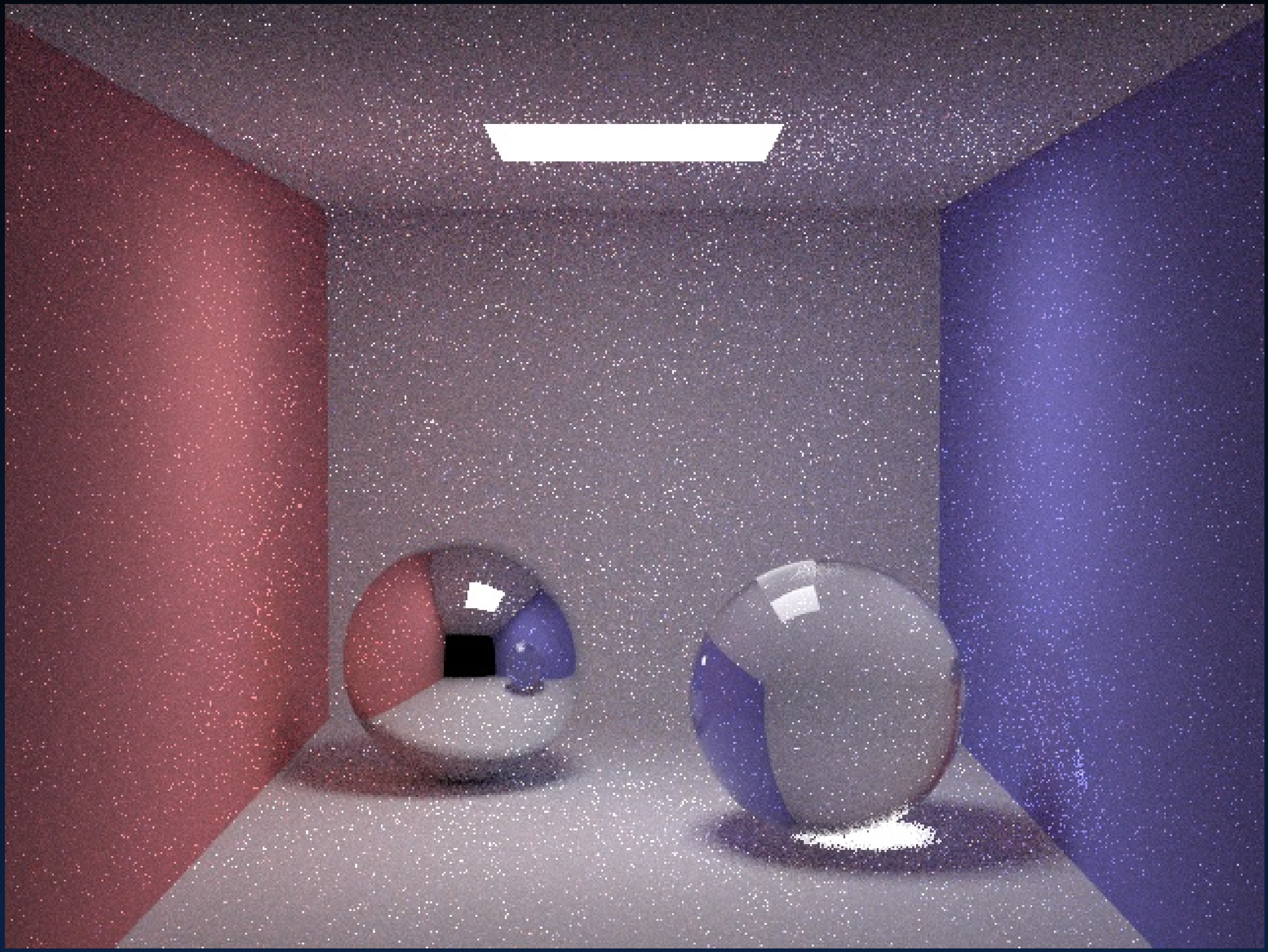
Filtering: Idea

- Noise is high frequency

Filtering: Idea

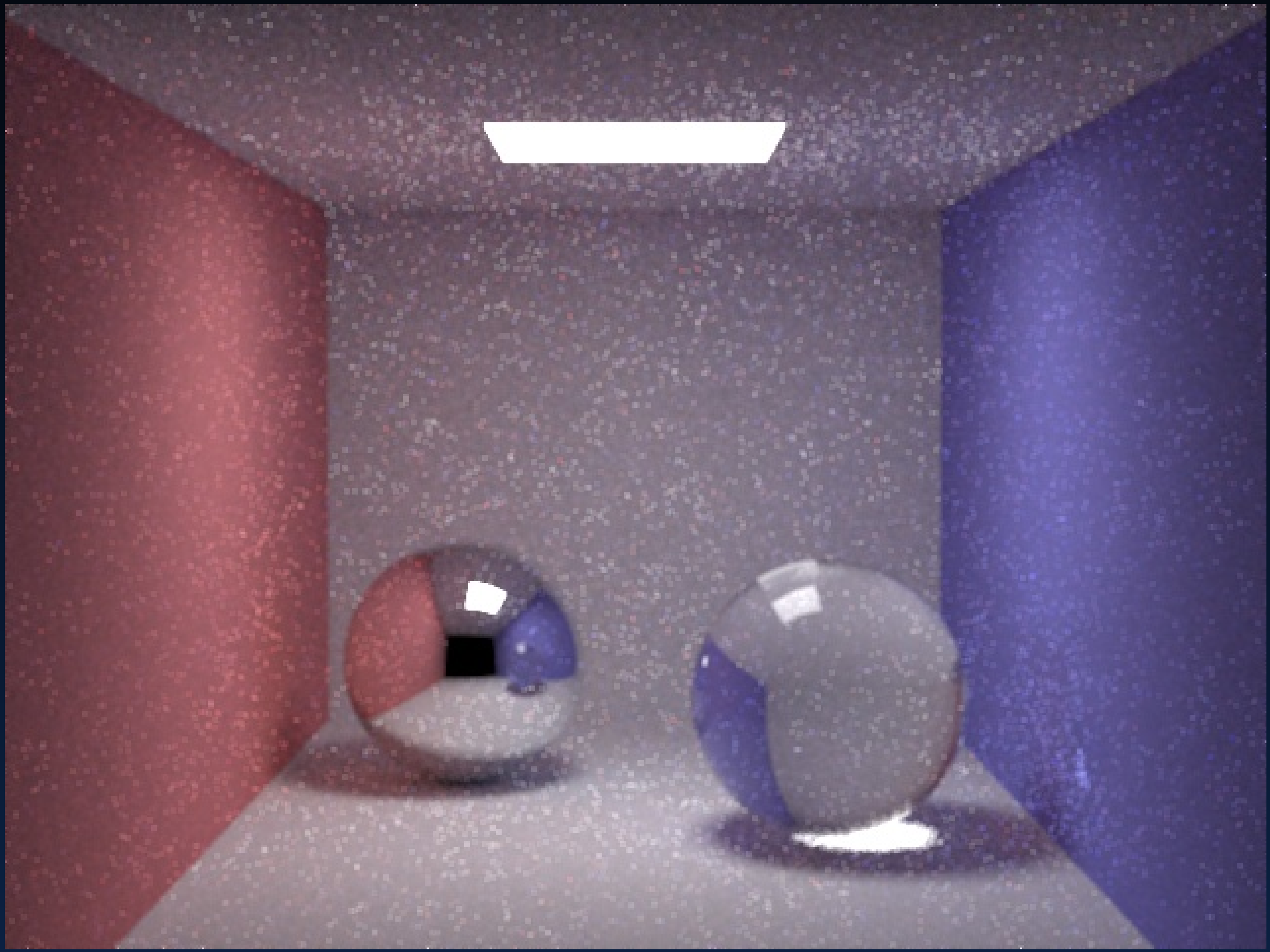
- Noise is high frequency
- Remove high frequency content

Unfiltered Image



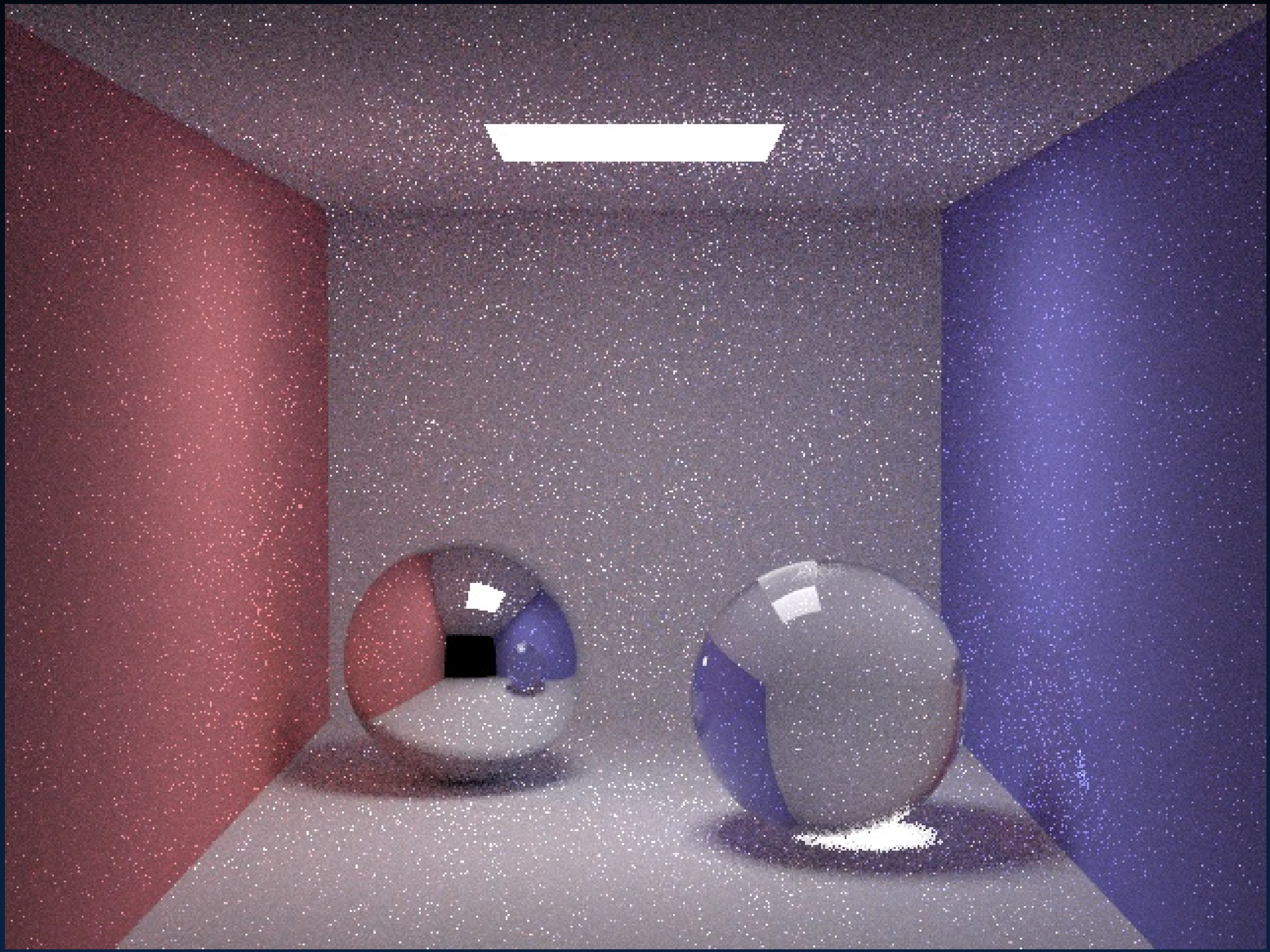
10 paths/pixel

3x3 Lowpass Filter



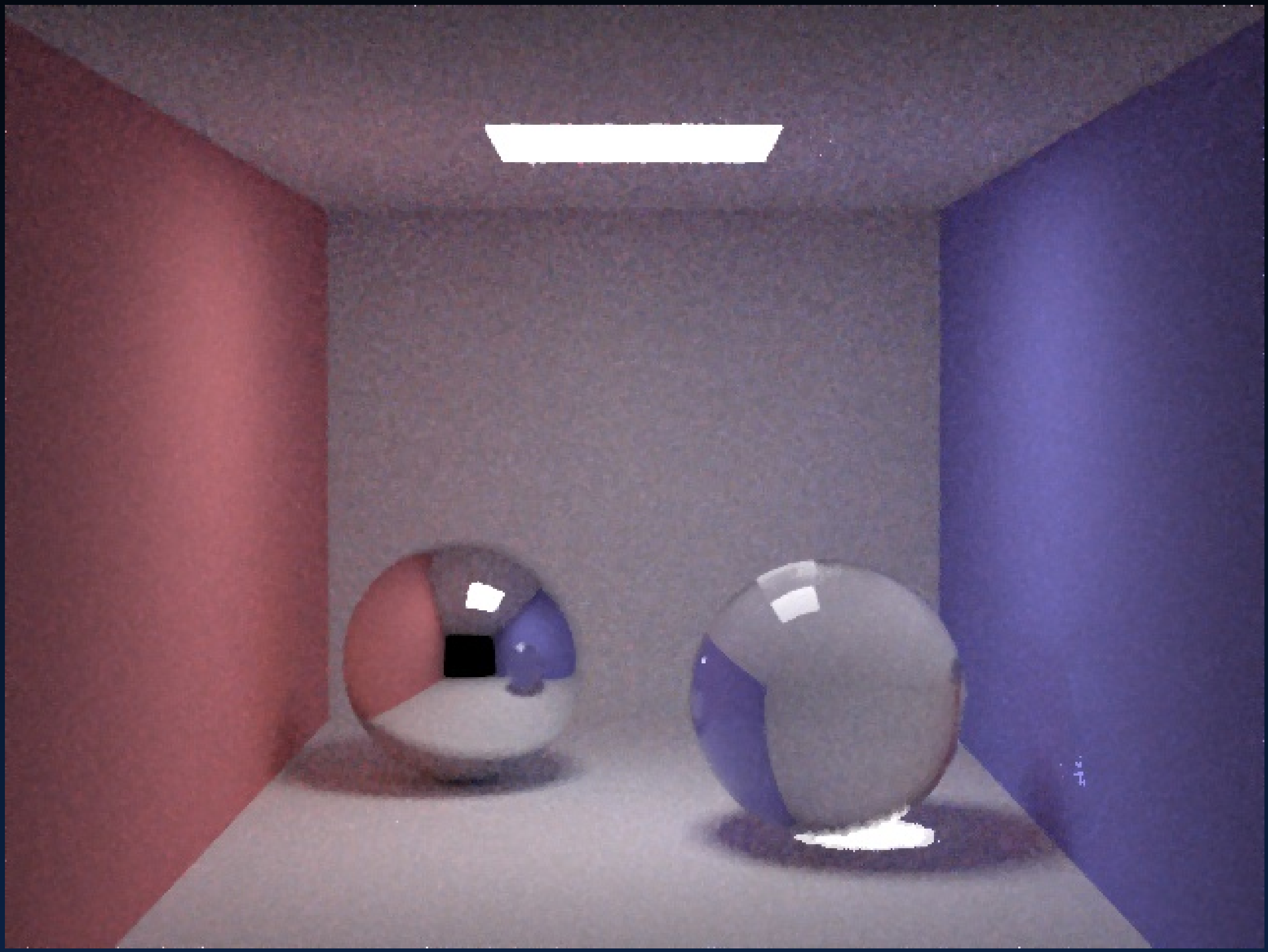
10 paths/pixel

Unfiltered Image



10 paths/pixel

3x3 Median Filter



10 paths/pixel

Energy Preserving Filters

Energy Preserving Filters

- Distribute noisy energy over several pixels

Energy Preserving Filters

- Distribute noisy energy over several pixels
- Adaptive filter width

  [Rushmeier and Ward 94]
- Diffusion style filters

  [McCool99]
- Splatting style filters

  [Suykens and Willems 00]

Problems With Filtering

- Everything is filtered (blurred)
  - Textures
  - Highlights
  - Caustics
  - . . .

Caching Techniques

Caching Techniques

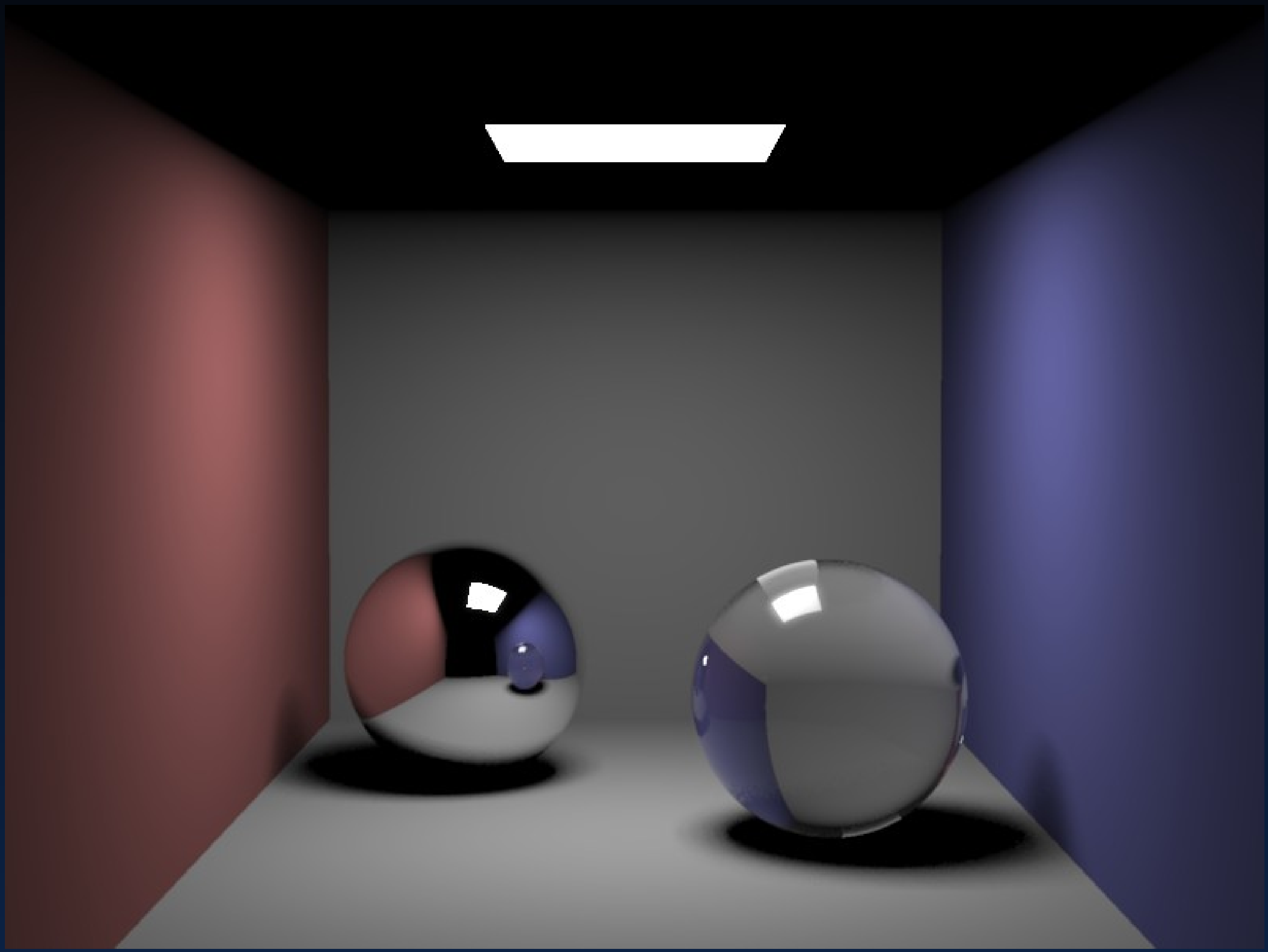
Irradiance caching :

Compute irradiance at selected points and interpolate.

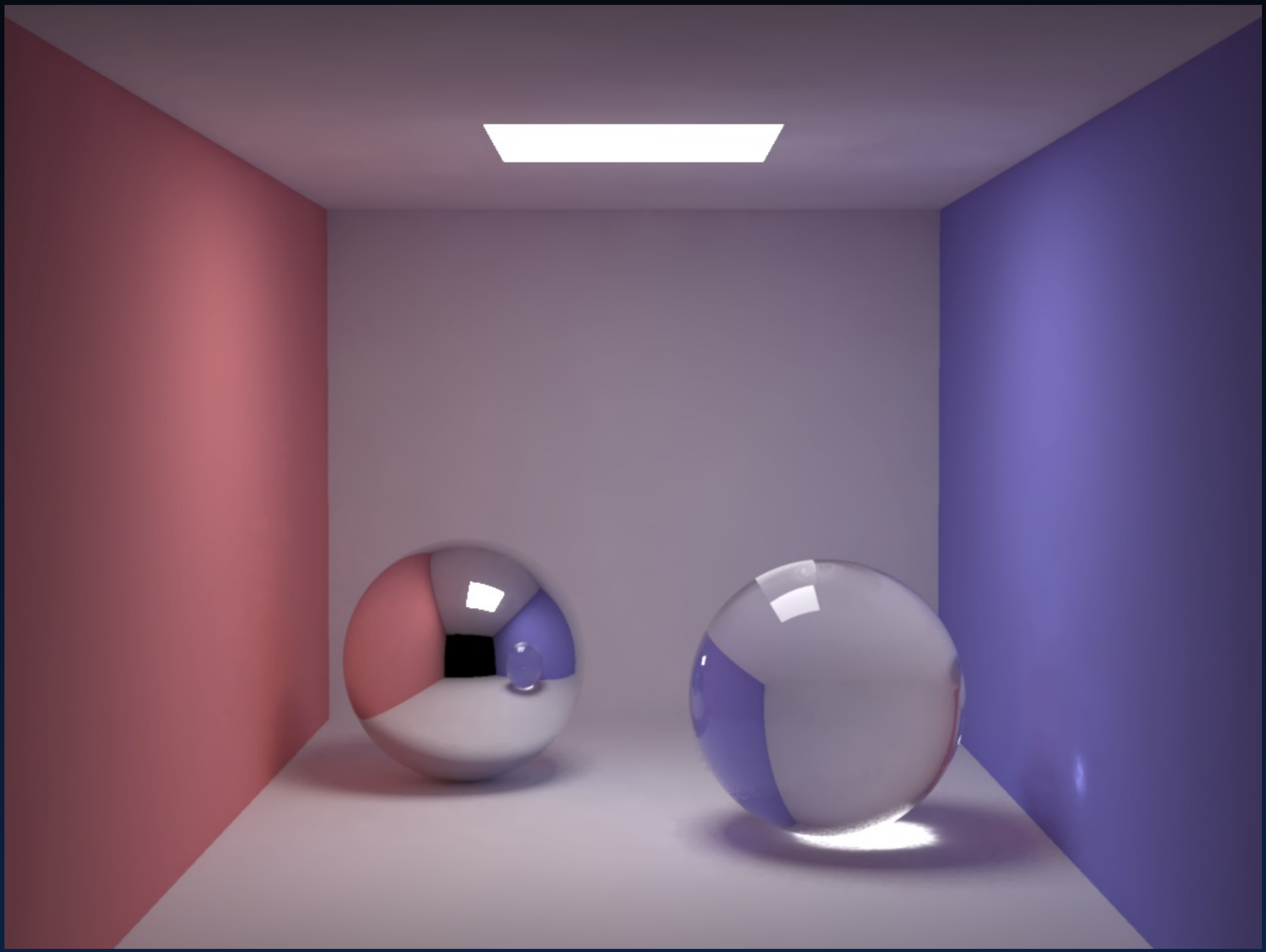
Photon mapping :

Trace “photons” from the lights and store them in a photon map, that can be used during rendering.

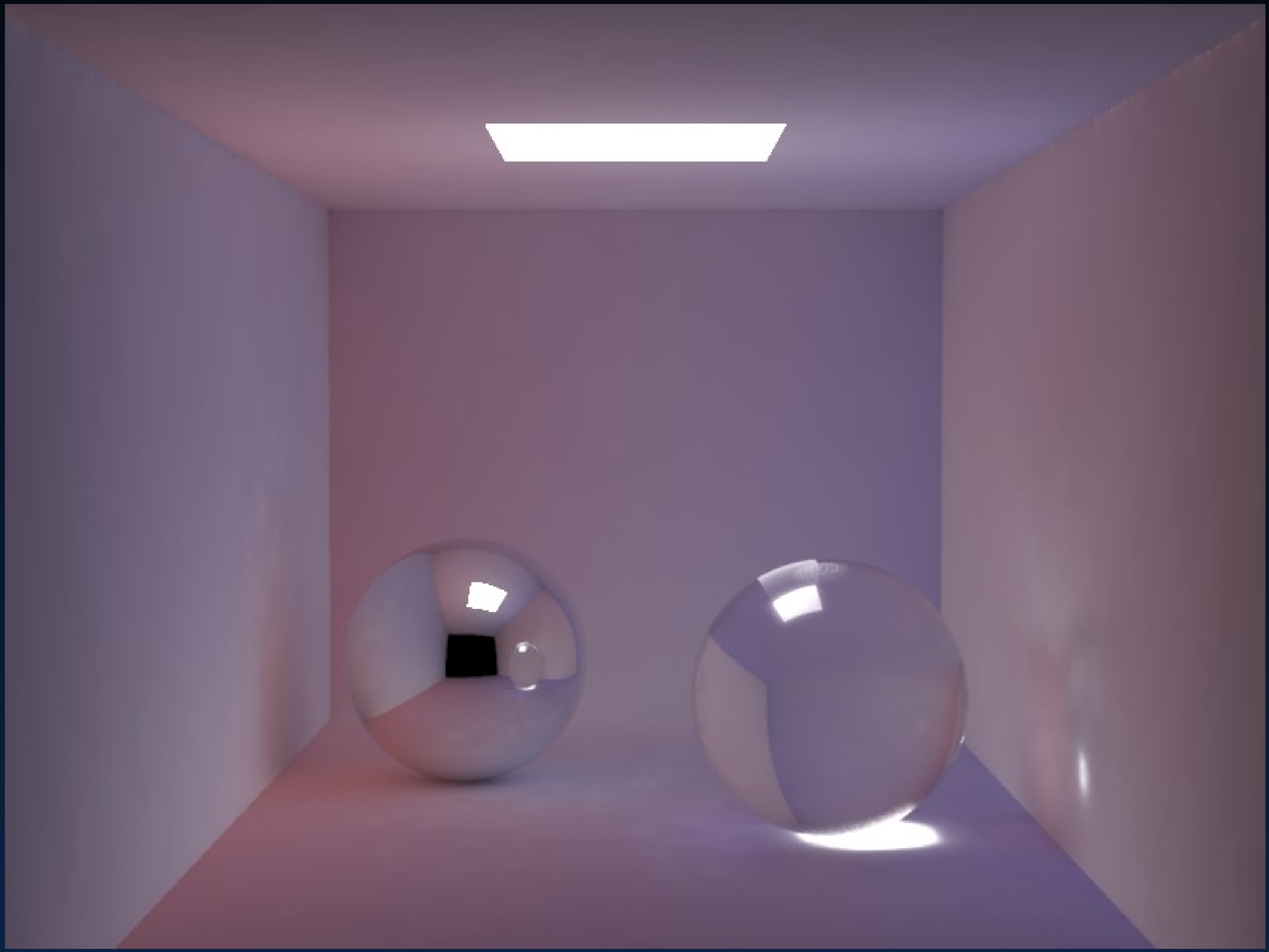
Box: Direct Illlumination



Box: Global Illlumination



Box: Indirect Irradiance



Irradiance Caching: Idea

"A Ray Tracing Solution for Diffuse Interreflection".
Greg Ward, Francis Rubinstein and Robert Clear:
Proc. SIGGRAPH'88.

Idea: Irradiance changes slowly $\rightarrow$ interpolate.

Irradiance Sampling

$$E(x) \;=\; \int_{2\pi} L'(x,\omega')\cos\theta\,d\omega'$$

Irradiance Sampling

$$
\begin{aligned}
E(x) &= \int_{2\pi} L'(x,\omega') \cos\theta \, d\omega' \\
&= \int_0^{2\pi} \int_0^{\pi/2} L'(x,\theta,\phi) \cos\theta \sin\theta \, d\theta \, d\phi
\end{aligned}
$$

Irradiance Sampling

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

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

Irradiance Change

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

Irradiance Change

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

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

Irradiance Interpolation

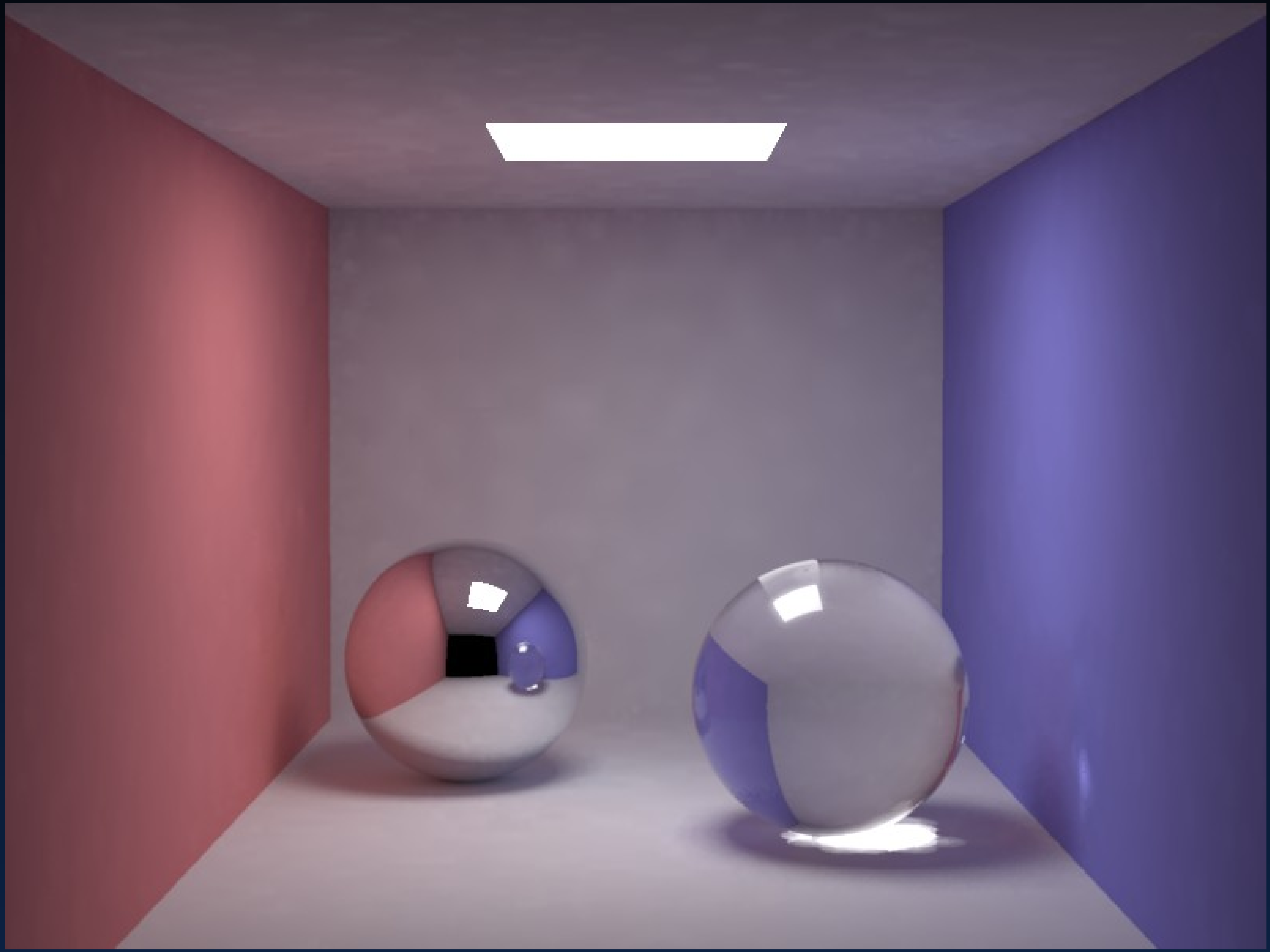
$$w(x) = \frac{1}{\epsilon(x)} \approx \frac{1}{\frac{||x - x_0||}{x_{avg}} + \sqrt{1 - \vec{N}(x) \cdot \vec{N}(x_0)}}$$

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

Irradiance Caching Algorithm

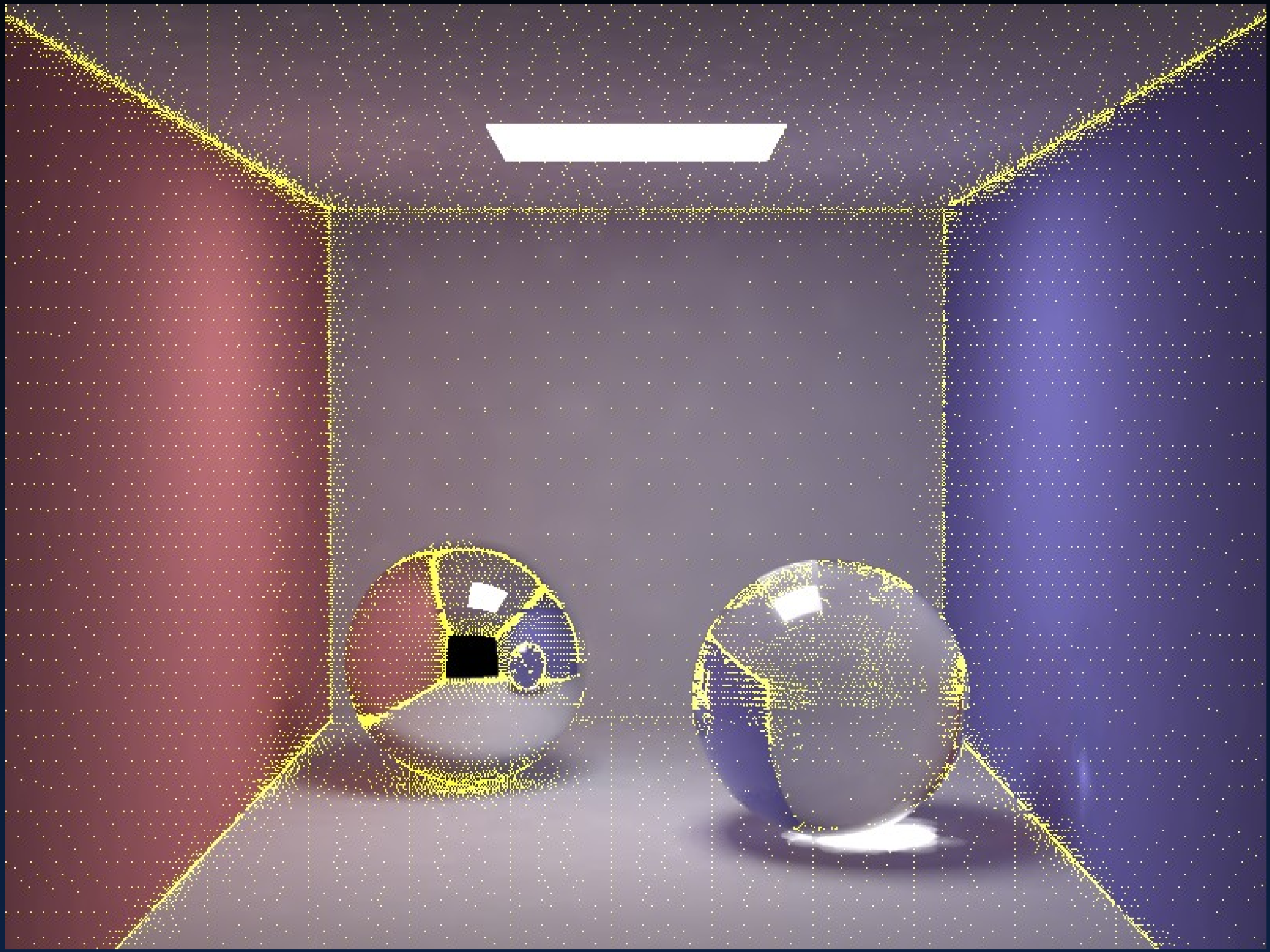
```
Find all irradiance samples with w(x) > q

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

Box: Irradiance Caching



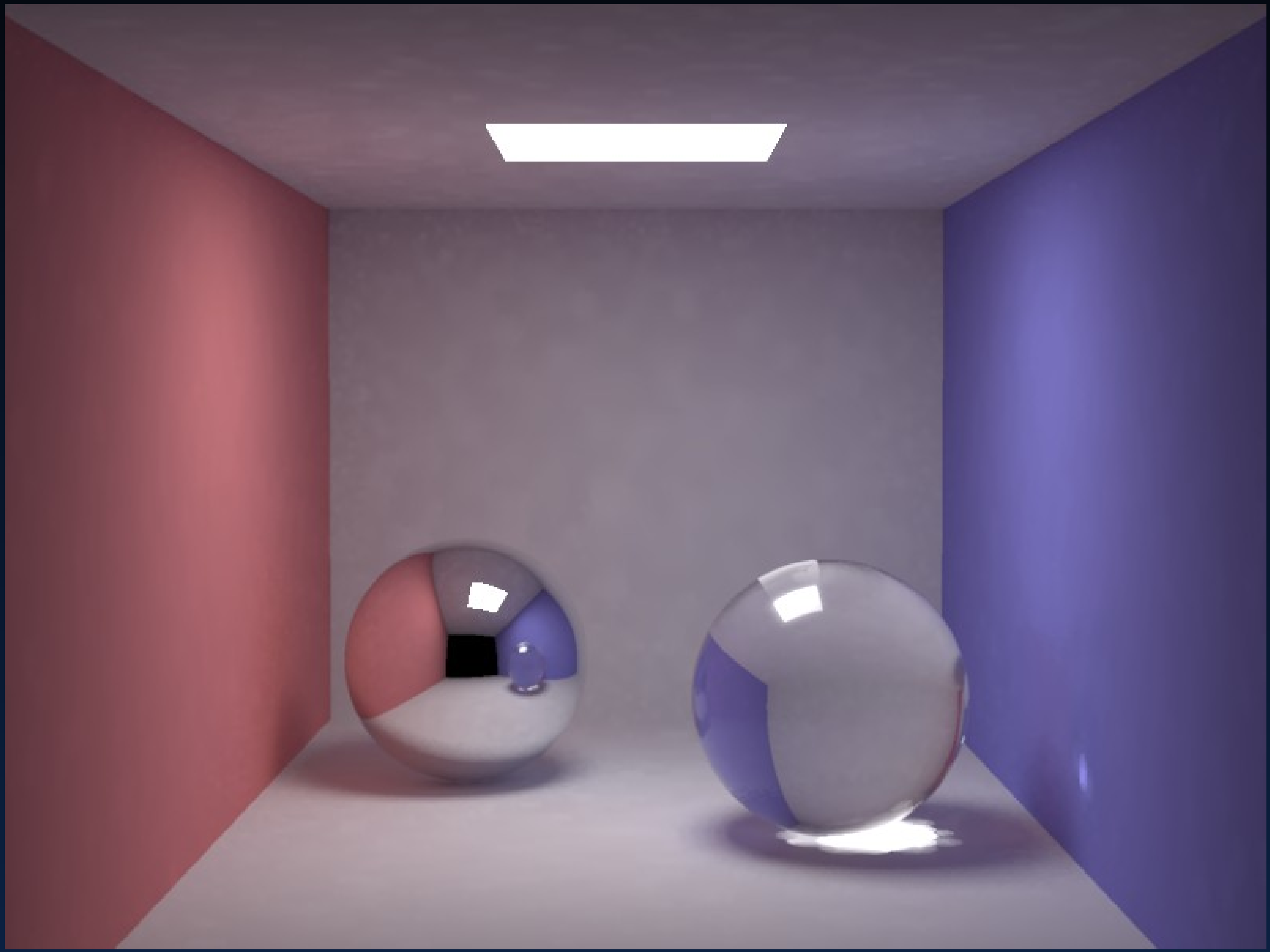
1000 sample rays, w $>$ 10

Box: Irradiance Cache Positions



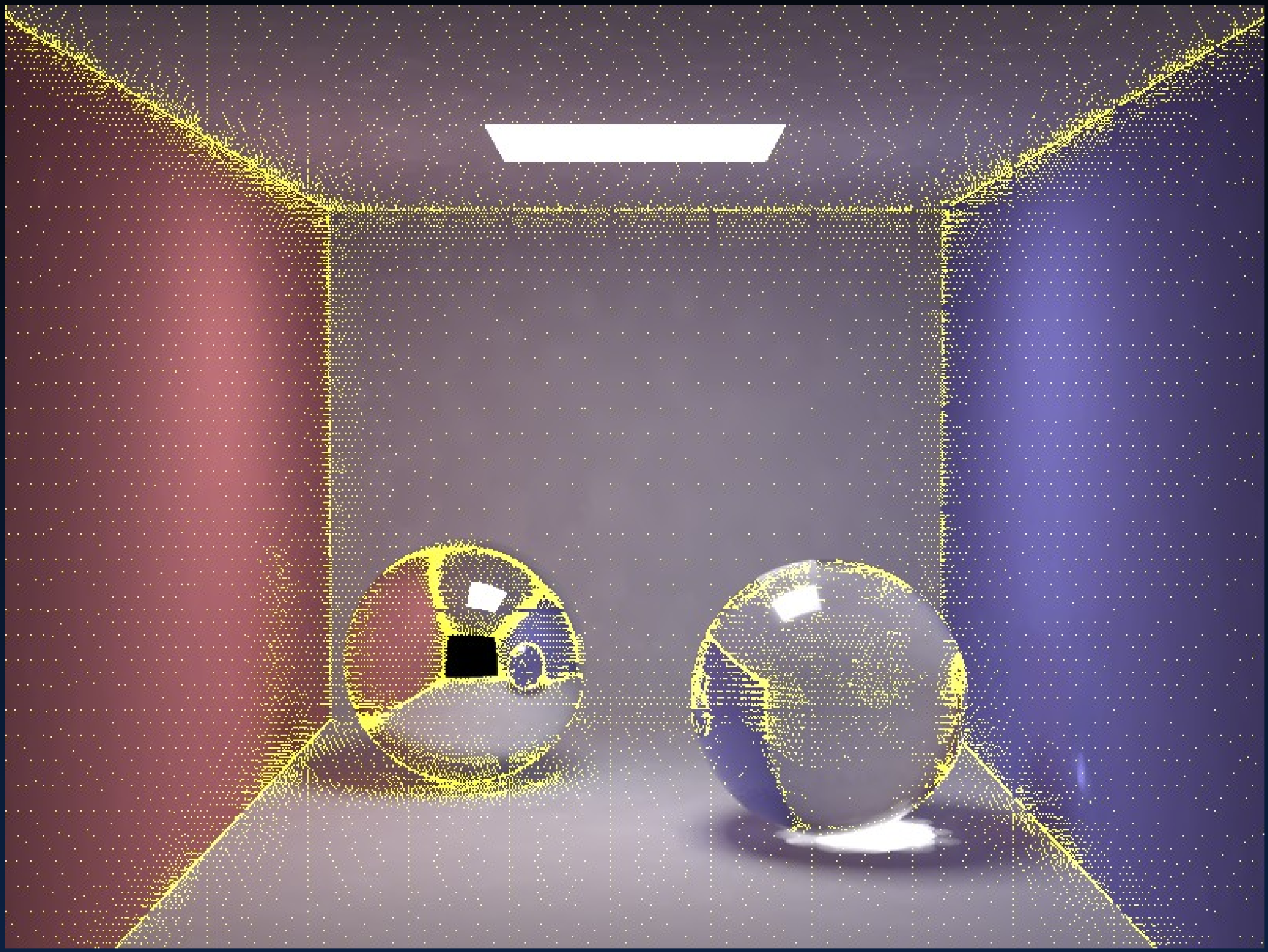
1000 sample rays, w > 10

Box: Irradiance Caching



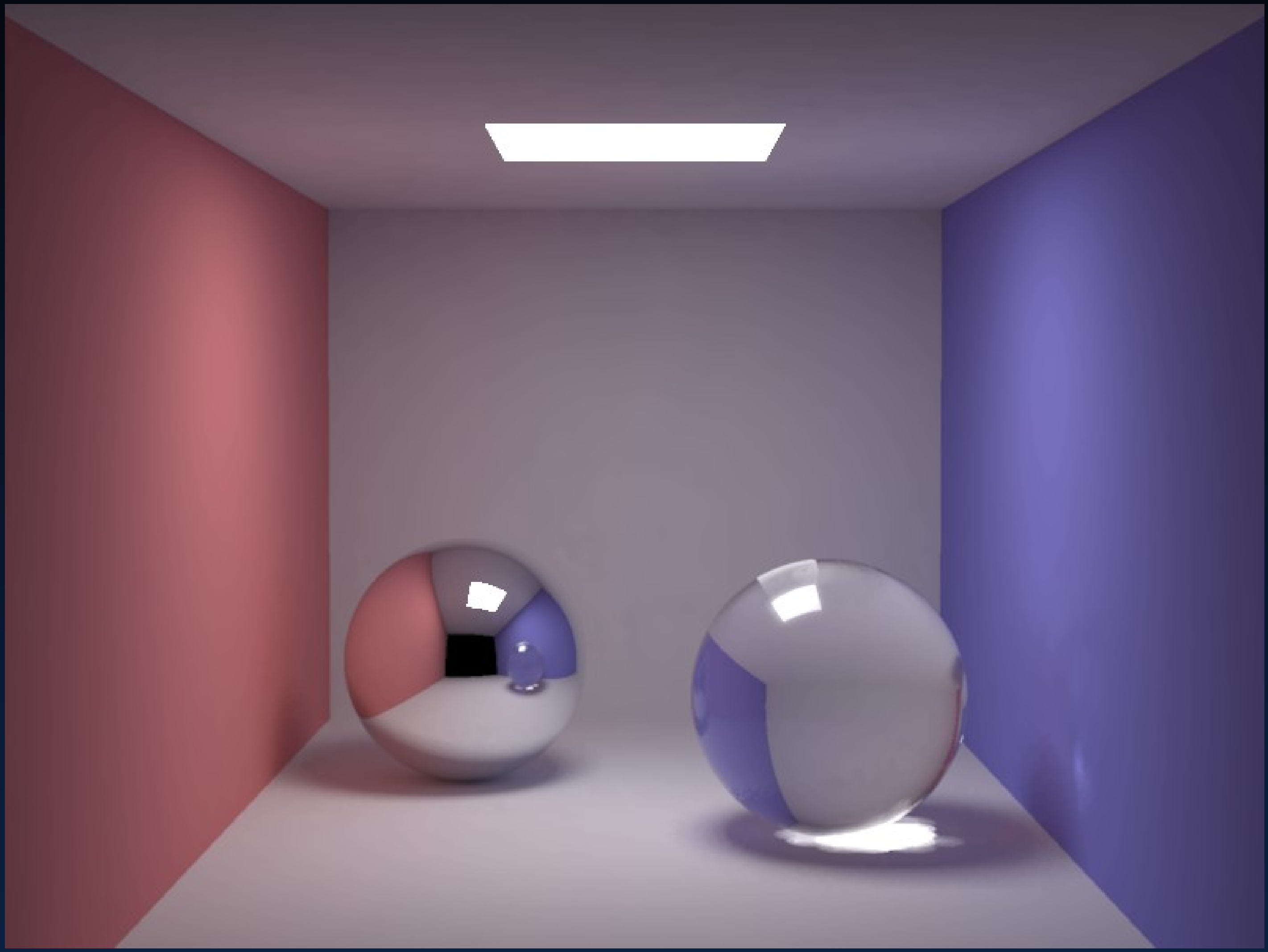
1000 sample rays, $w > 20$

Box: Irradiance Cache Positions



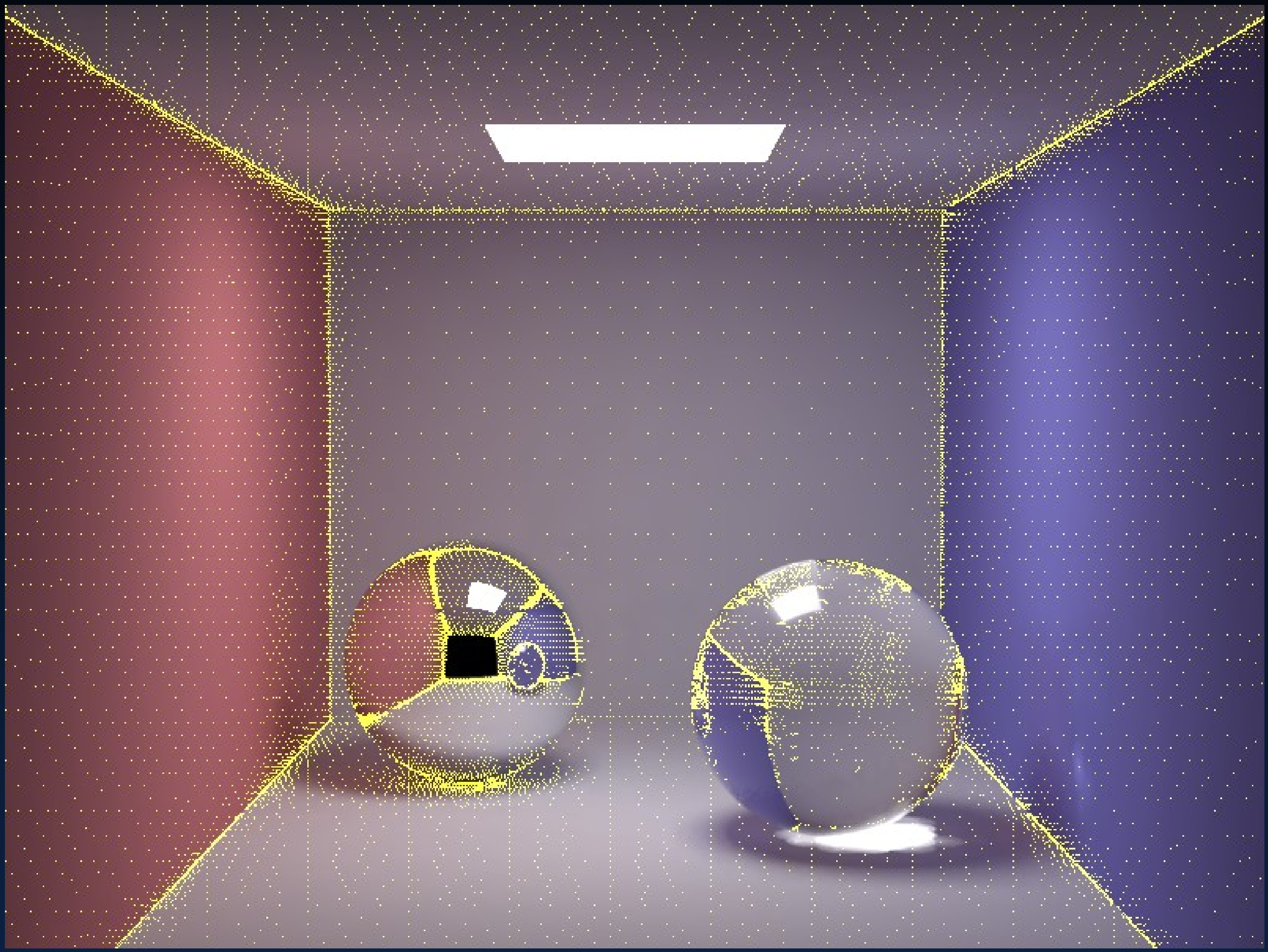
1000 sample rays, w $>$ 20

Box: Irradiance Caching



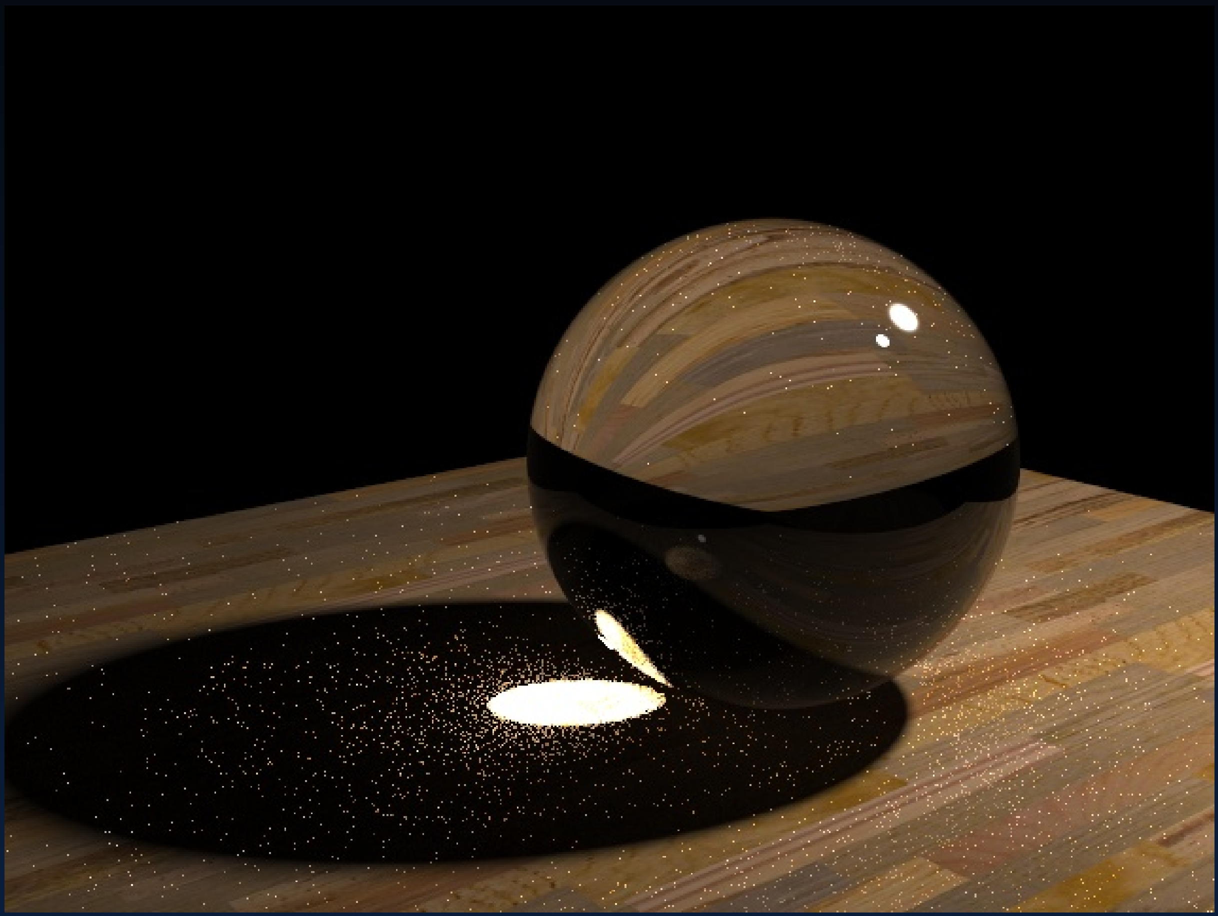
5000 sample rays, w>10

Box: Irradiance Cache Positions



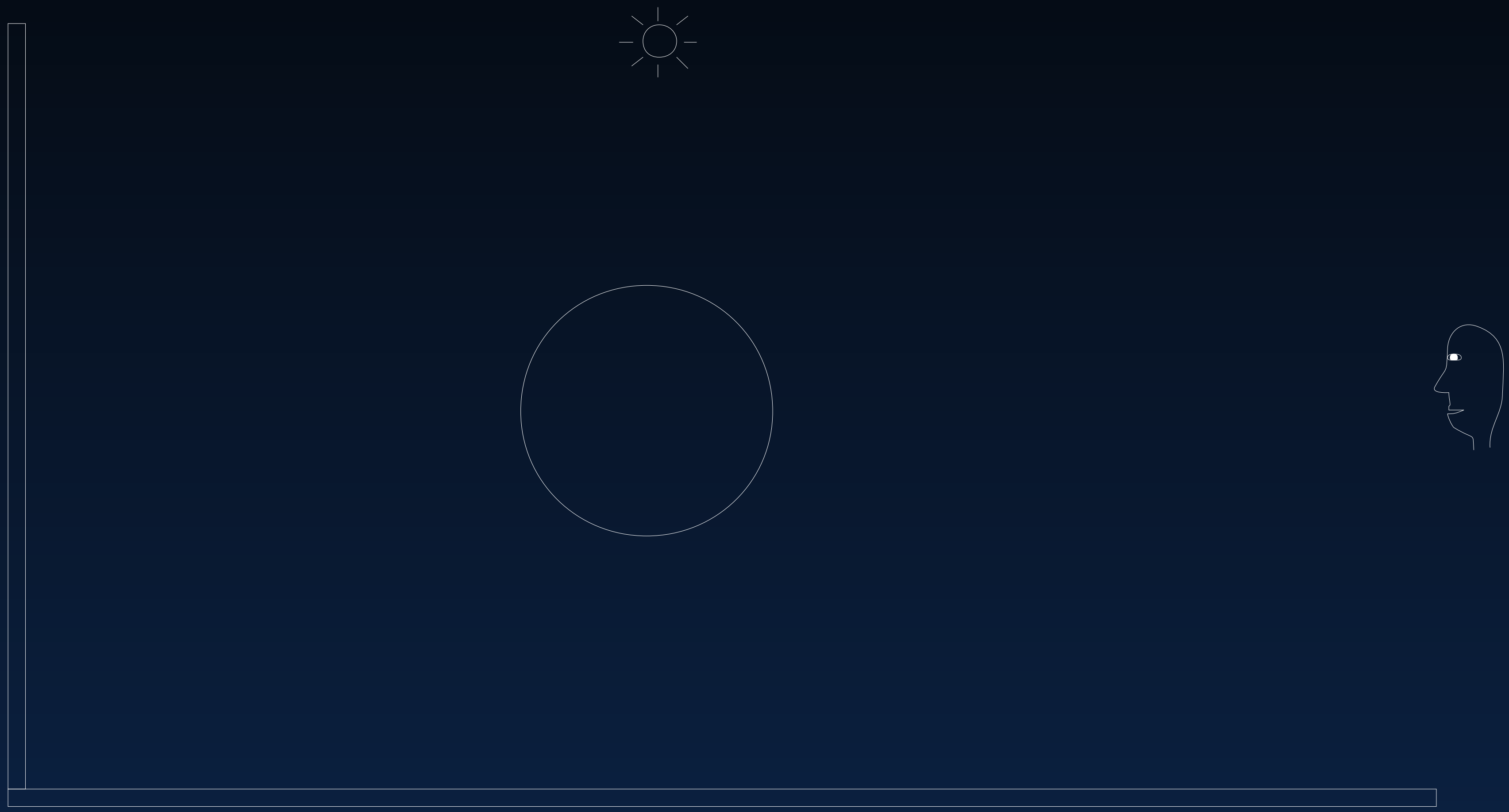
5000 sample rays, $w > 10$

Caustics

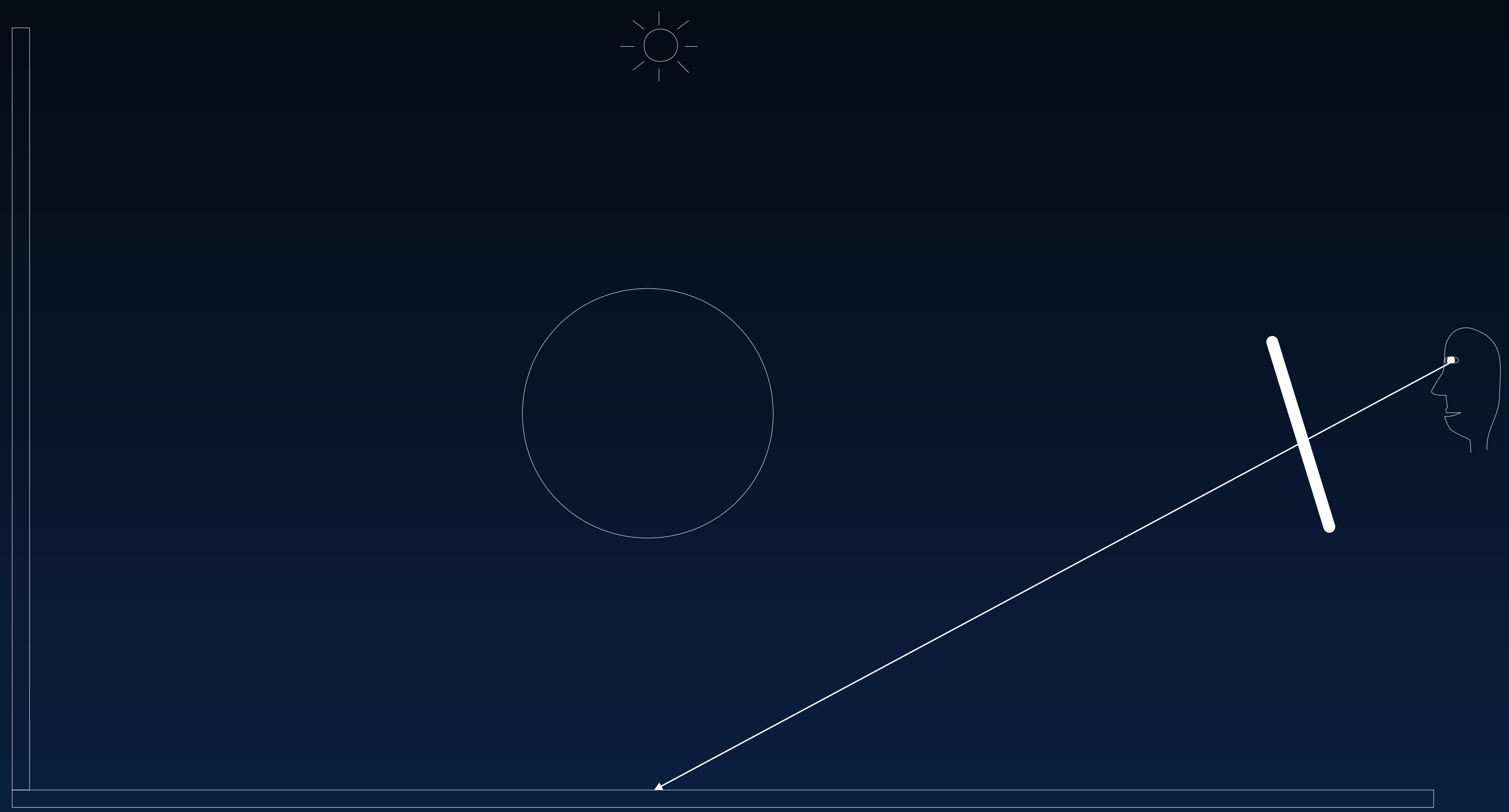


Pathtracing – 1000 paths/pixel

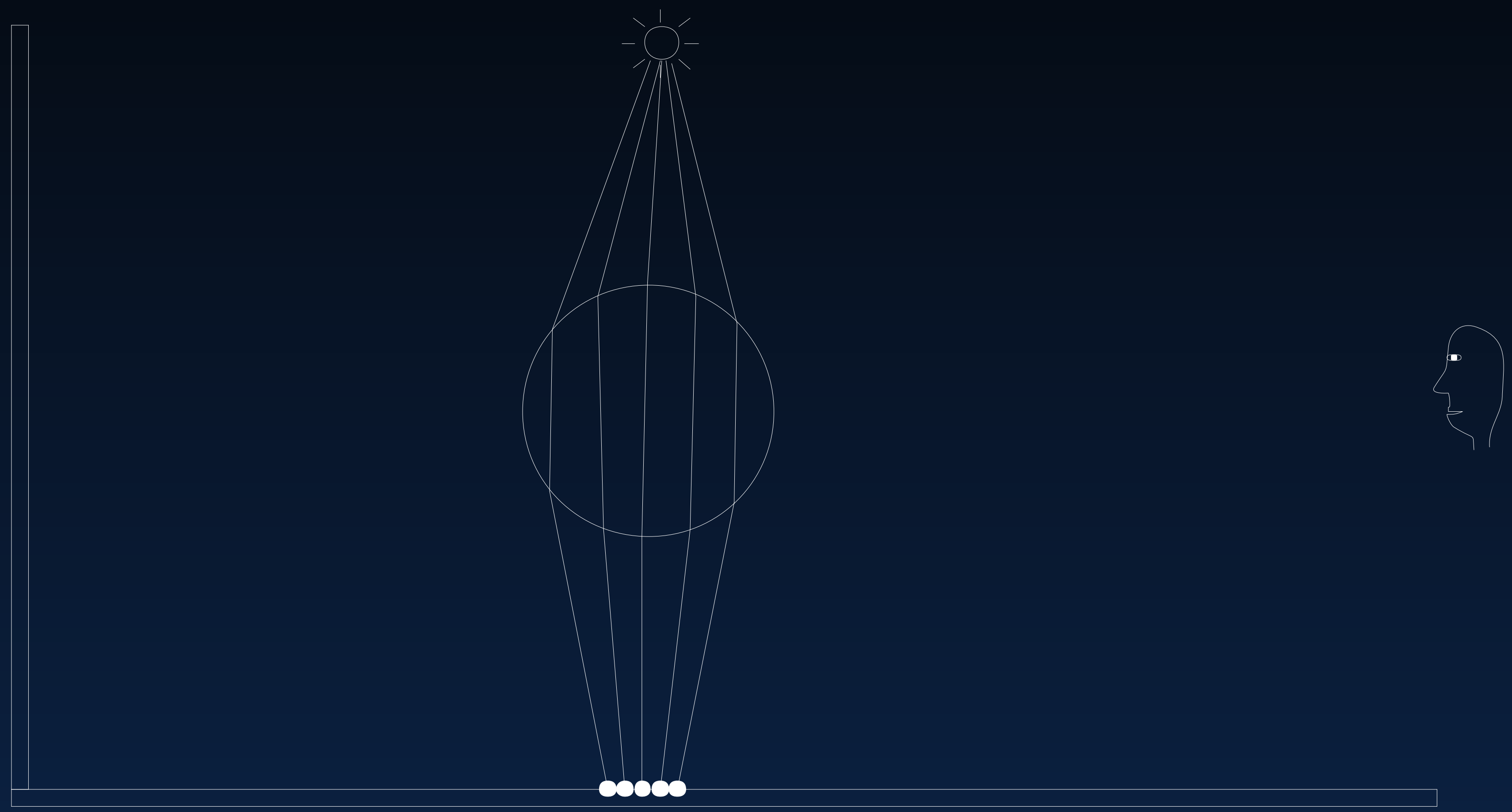
A simple test scene



Rendering



Photon Tracing



Photons

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

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

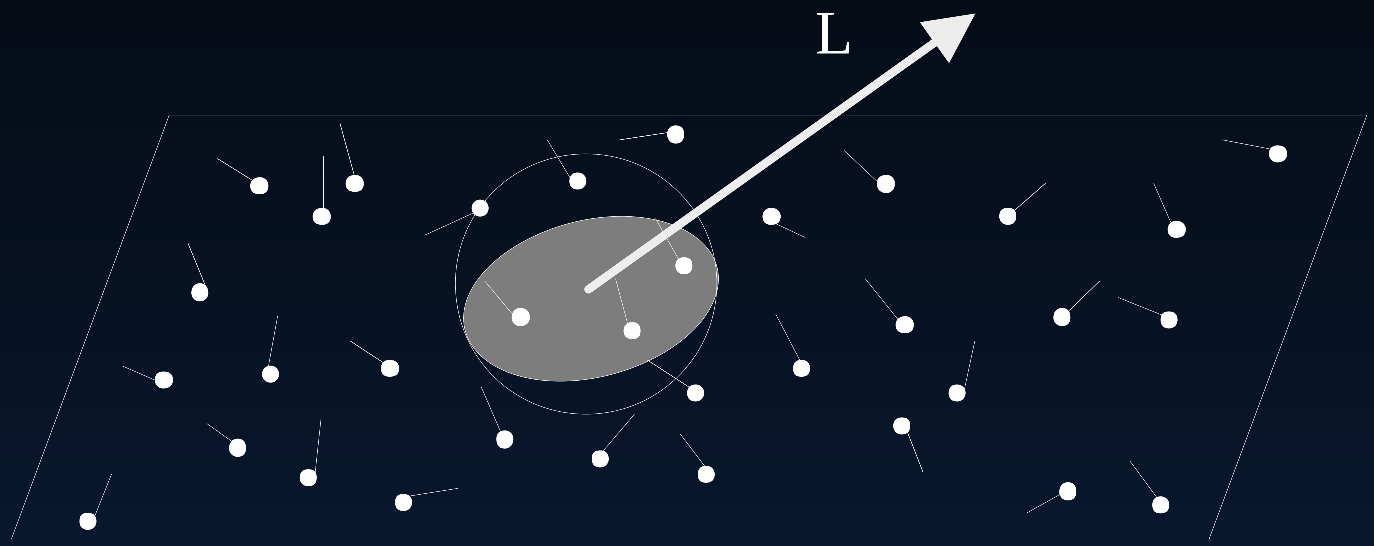
Radiance Estimate

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

Radiance Estimate

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

Radiance Estimate

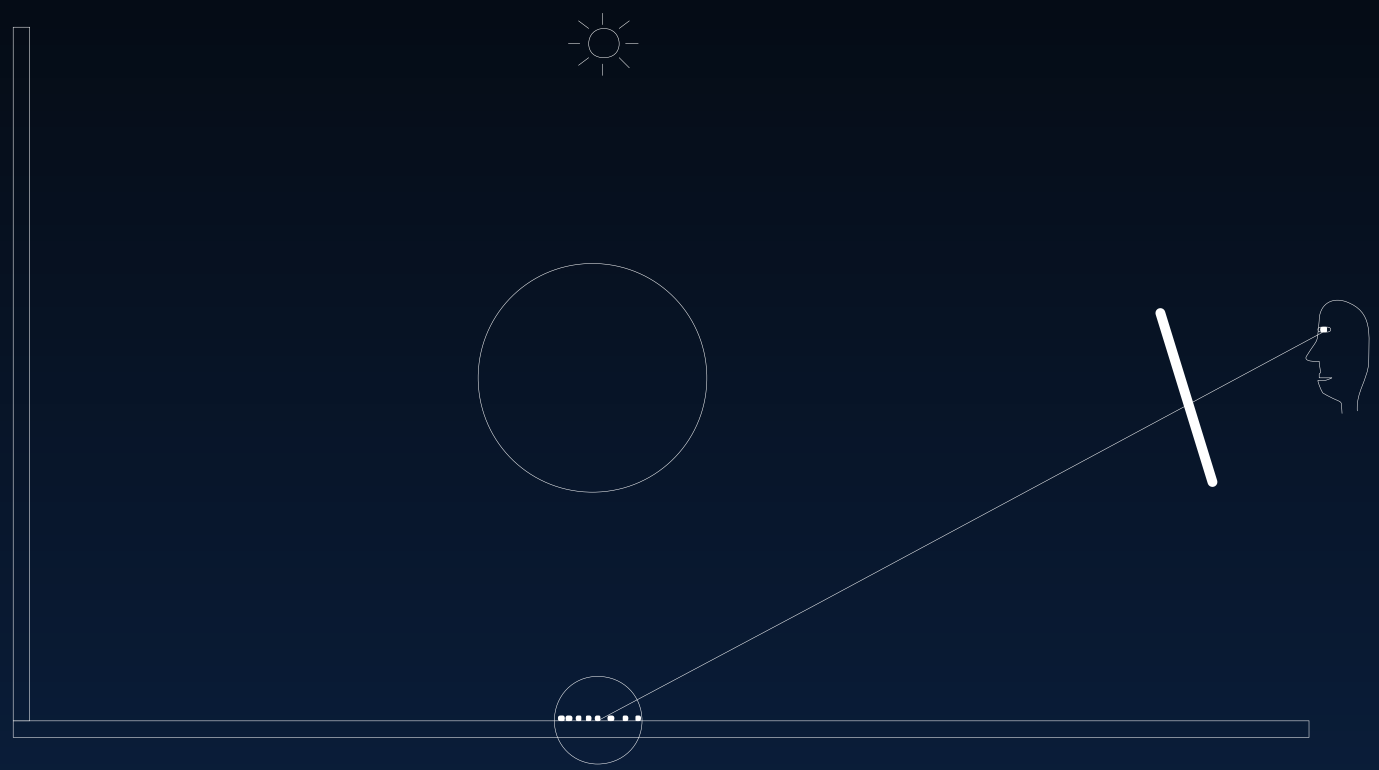


The photon map datastructure

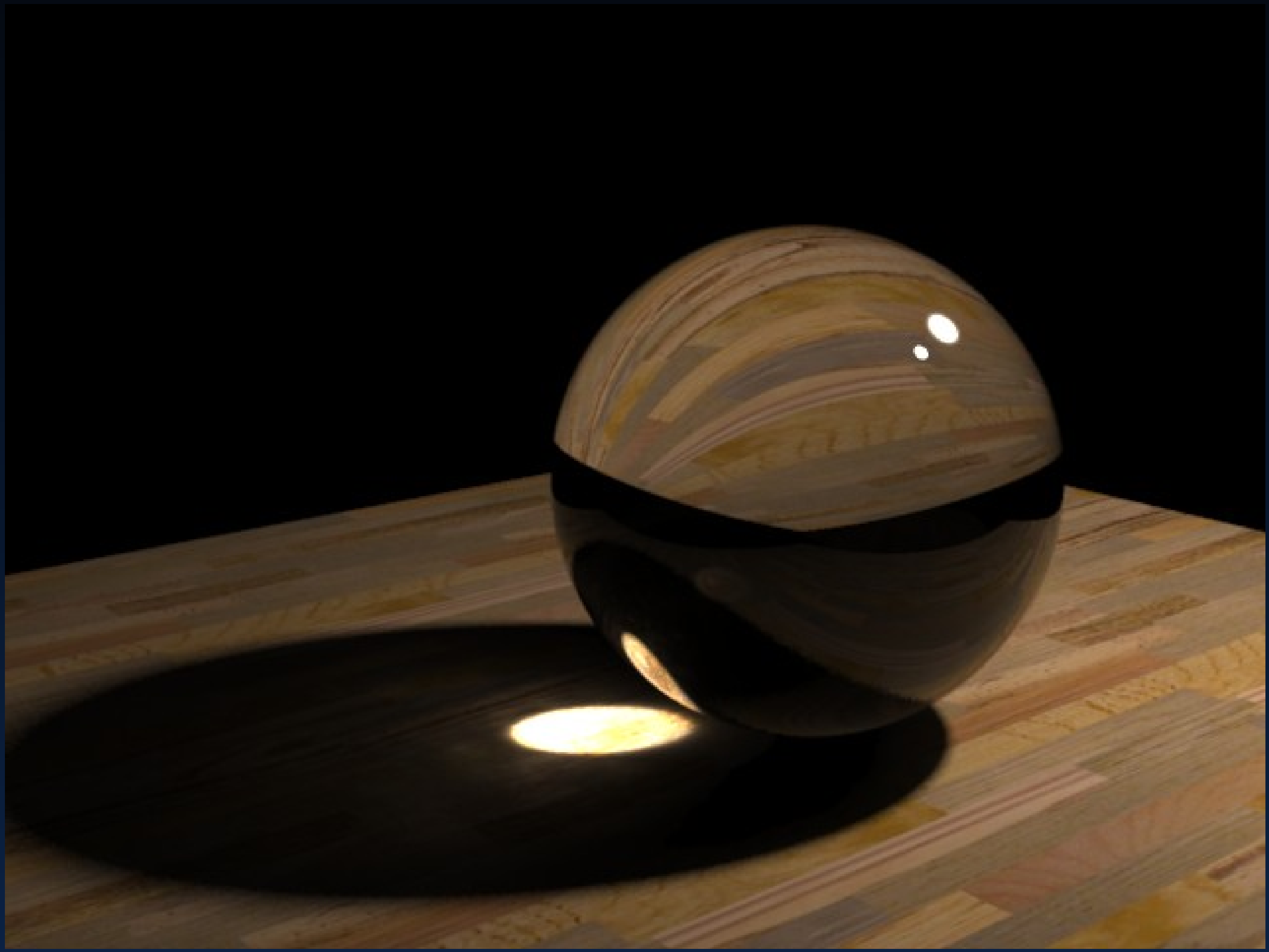
The photons are stored in a left balanced kd-tree

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

Rendering: Caustics

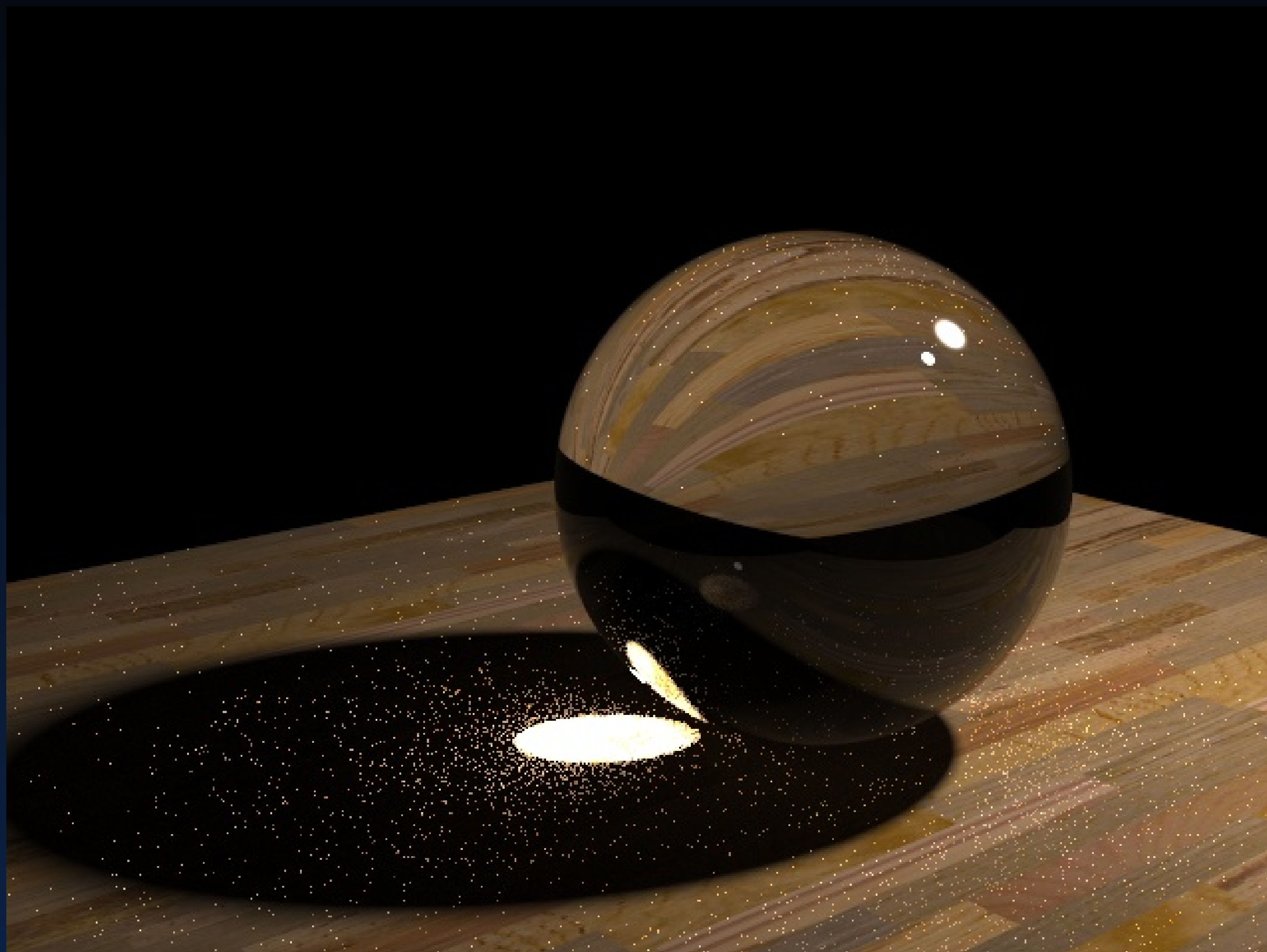


Caustic from a Glass Sphere



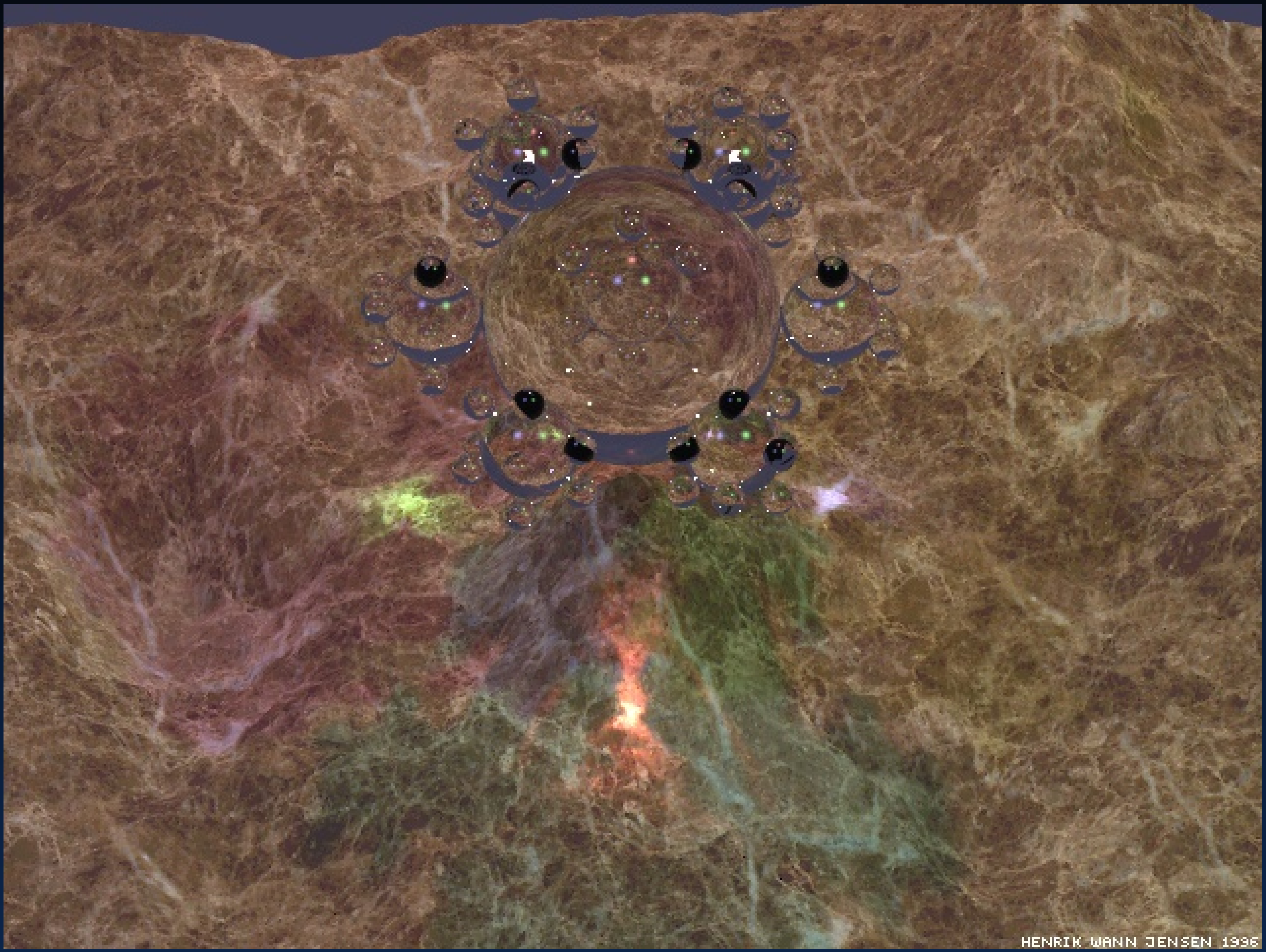
Photon Mapping: 10000 photons / 50 photons in radiance estimate

Caustic from a Glass Sphere

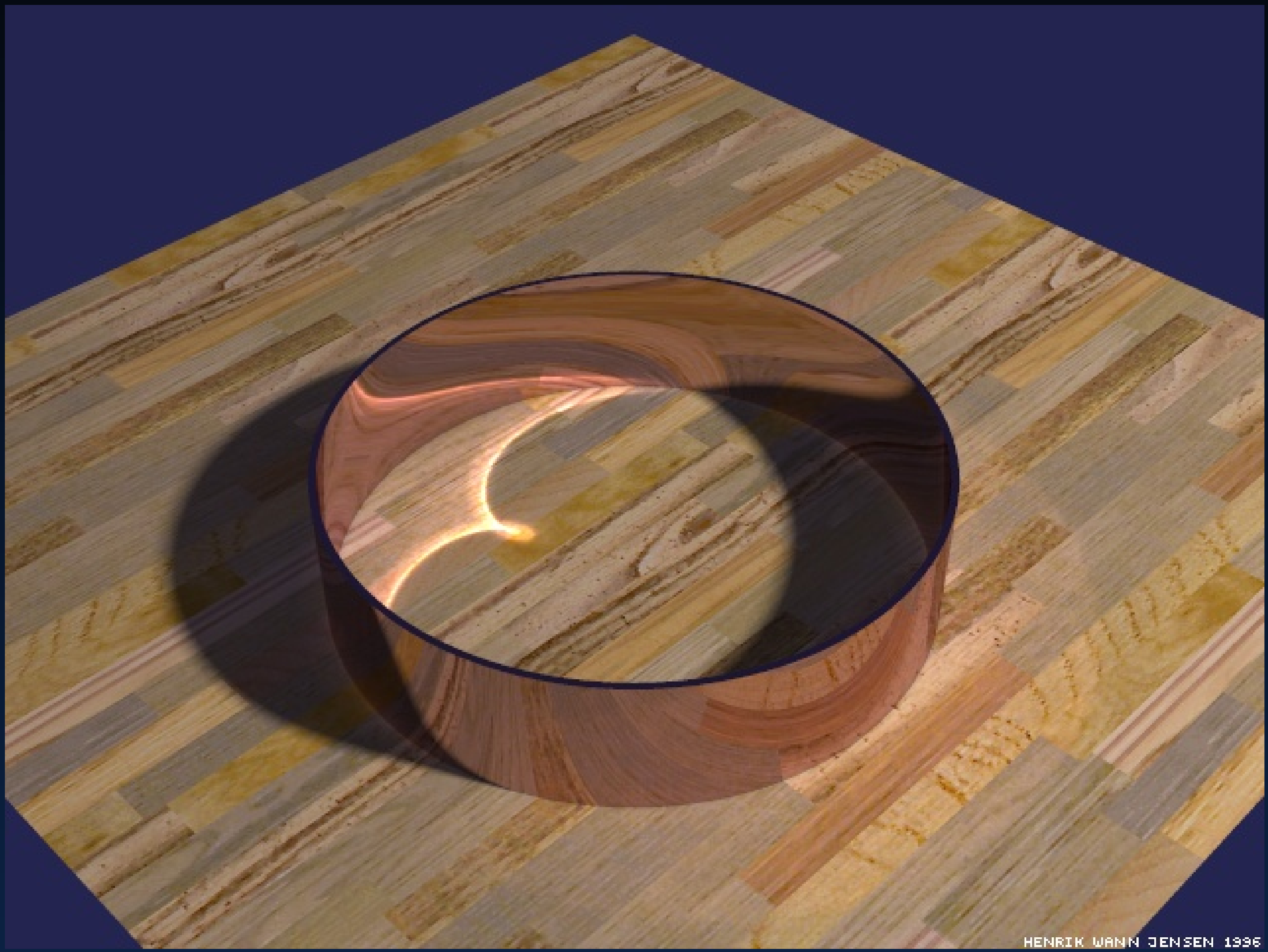


Path Tracing: 1000 paths/pixel

Spereflake Caustic



Reflection Inside A Metal Ring

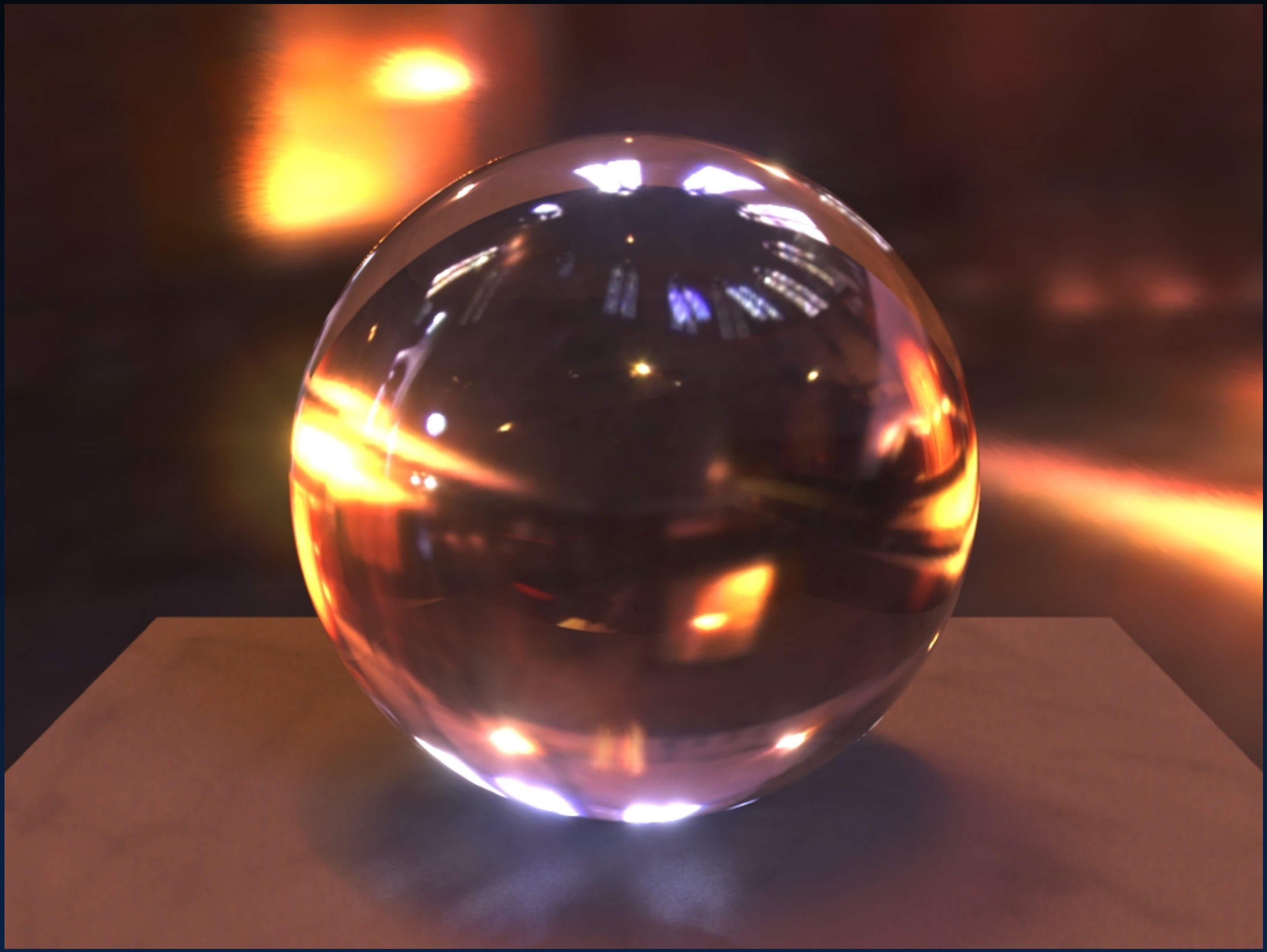


50000 photons / 50 photons in radiance estimate

Caustics On Glossy Surfaces

340000 photons / $\approx$ 100 photons in radiance estimate

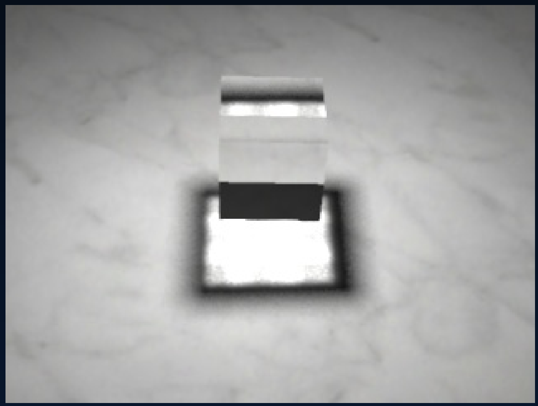
HDR environment illumination



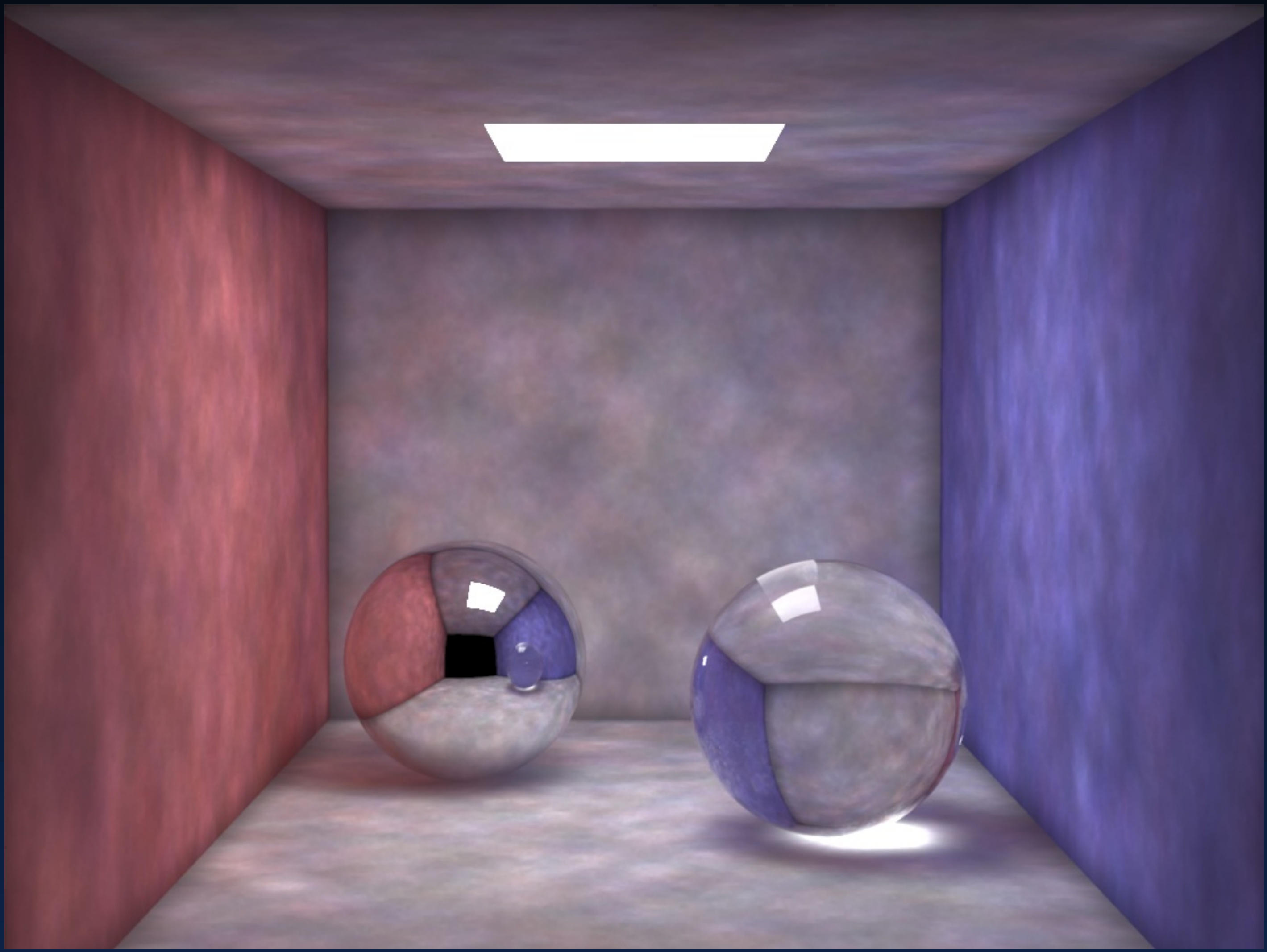
Using lightprobe from www.debevec.org

Cognac Glass

Cube Caustic

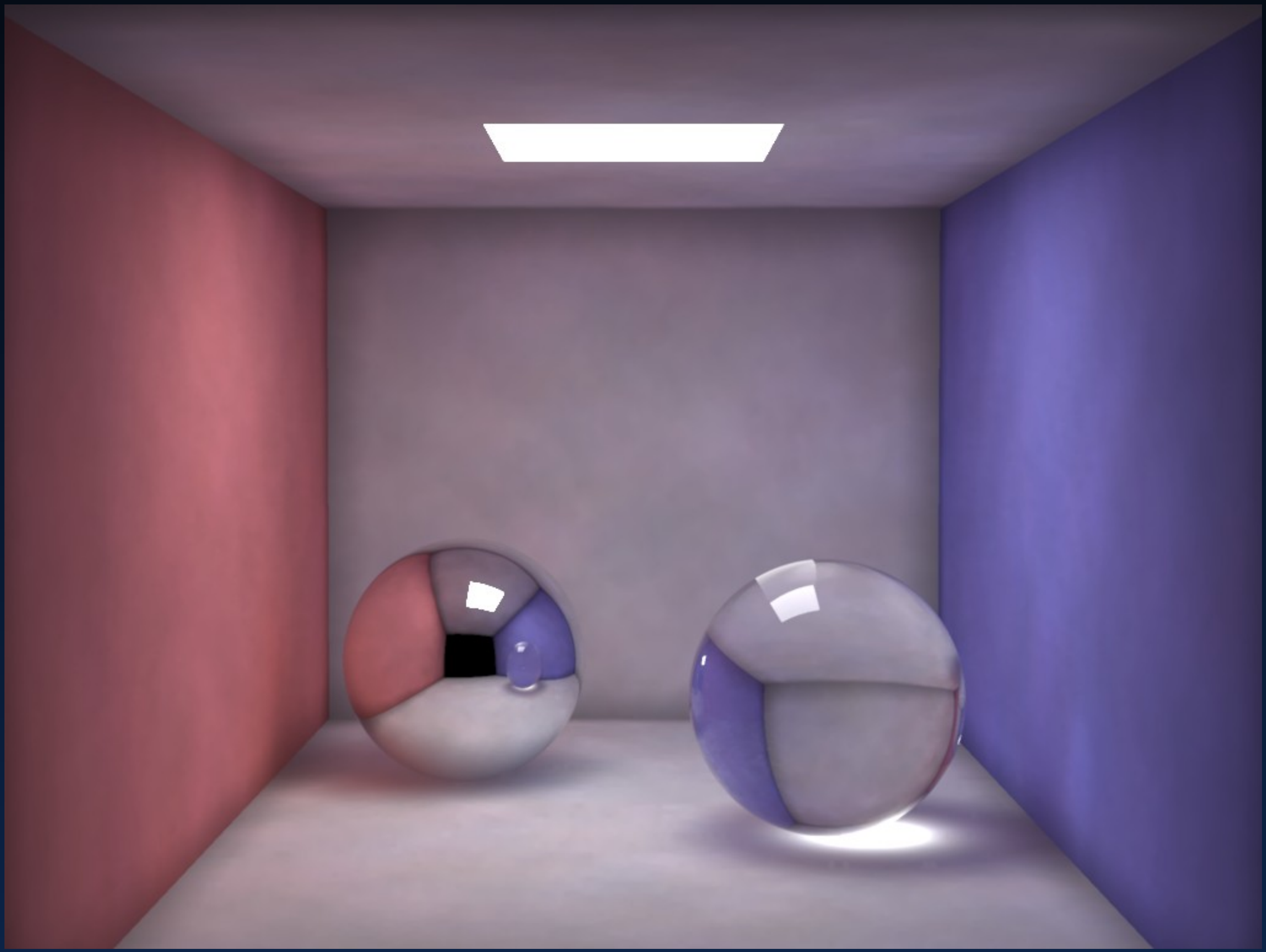


Global Illumination



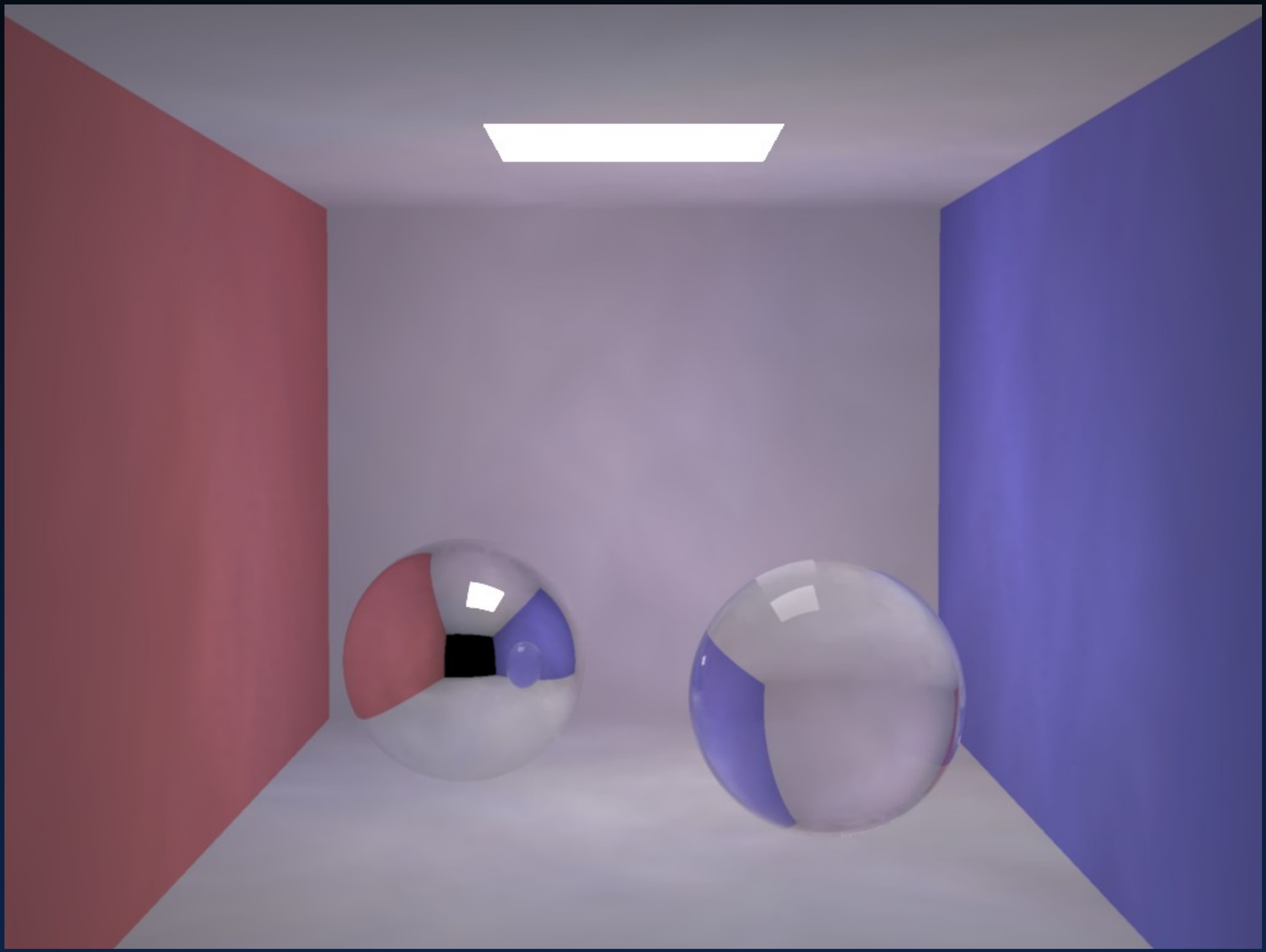
100000 photons / 50 photons in radiance estimate

Global Illumination



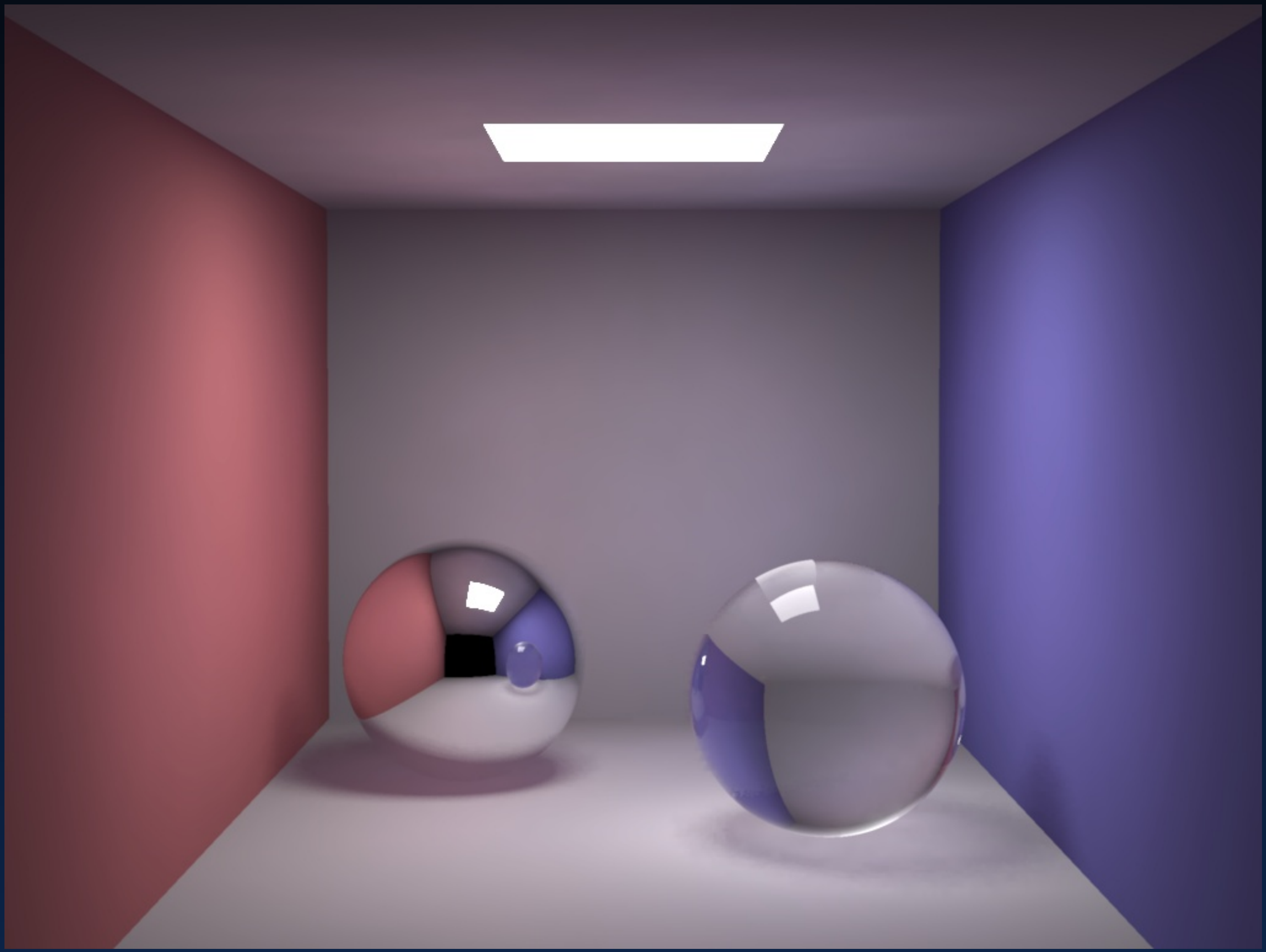
500000 photons / 500 photons in radiance estimate

Fast estimate



200 photons / 50 photons in radiance estimate

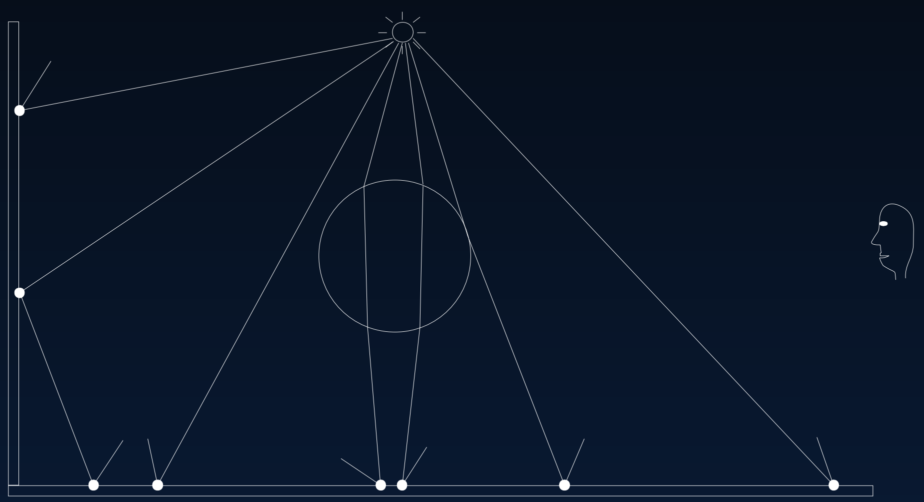
Indirect illumination



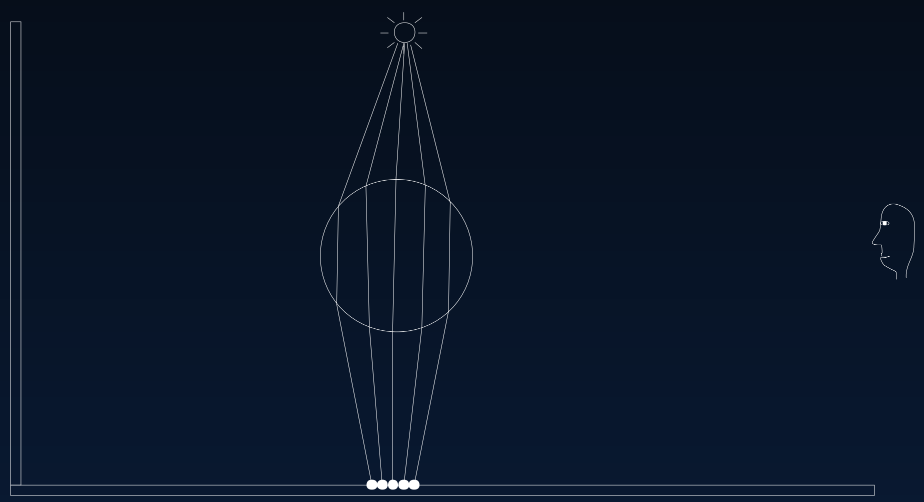
10000 photons / 500 photons in radiance estimate

Global Illumination

Global Illumination



global photon map



caustics photon map

Photon tracing

- Photon emission
- Photon scattering
- Photon storing

Photon emission

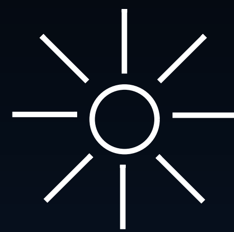
Given Φ Watt lightbulb.
Emit N photons.
Each photon has the power $\frac{\Phi}{N}$ Watt.

- Photon power depends on the number of emitted photons. Not on the number of photons in the photon map.

What is a photon?

- Flux (power) - not radiance!
- Collection of physical photons
  - A fraction of the light source power
  - Several wavelengths combined into one entity

Diffuse point light



Generate random direction
Emit photon in that direction

```
// Find random direction
do {
  x = 2.0*random()-1.0;
  y = 2.0*random()-1.0;
  z = 2.0*random()-1.0;
} while ( (x*x + y*y + z*z) > 1.0 );
```

Example: Diffuse square light

- Generate random position p on square
- Generate diffuse direction d
- Emit photon from p in direction d

```
// Generate diffuse direction
u = random();
v = 2*π*random();
d = vector( cos(v)√u, sin(v)√u, √(1-u) );
```

Surface interactions

The photon is

- Stored (at diffuse surfaces) and
- Absorbed (A) or
- Reflected (R) or
- Transmitted (T)

$$A + R + T = 1.0$$

Photon scattering

The simple way:

Given incoming photon with power Φ_p

Reflect photon with the power $R * \Phi_p$

Transmit photon with the power $T * \Phi_p$

Photon scattering

The simple way:

Given incoming photon with power Φ_p

Reflect photon with the power $R * \Phi_p$

Transmit photon with the power $T * \Phi_p$

- Risk: Too many low-powered photons - wasteful!
- When do we stop (systematic bias)?
- Photons with similar power is a good thing.

Russian Roulette

- Statistical technique
- Known from Monte Carlo particle physics
- Introduced to graphics by Arvo and Kirk in 1990

Russian Roulette

Probability of termination: p

Russian Roulette

Probability of termination: p

$$E\{X\}$$

Russian Roulette

Probability of termination: p

$$E\{X\} = p \cdot 0$$

Russian Roulette

Probability of termination: p

$$E\{X\} = p \cdot 0 + (1 - p)$$

Russian Roulette

Probability of termination: p

$$E\{X\} = p \cdot 0 + (1-p) \cdot \frac{E\{X\}}{1-p}$$

Russian Roulette

Probability of termination: p

$$E\{X\} = p \cdot 0 + (1-p) \cdot \frac{E\{X\}}{1-p} = E\{X\}$$

Russian Roulette

Probability of termination: p

$$E\{X\} = p \cdot 0 + (1-p) \cdot \frac{E\{X\}}{1-p} = E\{X\}$$

Terminate un-important photons and still get the correct result.

Russian Roulette Example

Surface reflectance: R = 0.5
Incoming photon: Φ_p = 2 W

```
r = random();
if ( r < 0.5 )
  reflect photon with power 2 W
else
  photon is absorbed
```

Russian Roulette Intuition

Surface reflectance: R = 0.5
200 incoming photons with power: Φ_p = 2 Watt

Reflect 100 photons with power 2 Watt instead of 200 photons with power 1 Watt.

Russian Roulette

- Very important!
- Use to eliminate un-important photons
- Gives photons with similar power :)

Sampling a BRDF

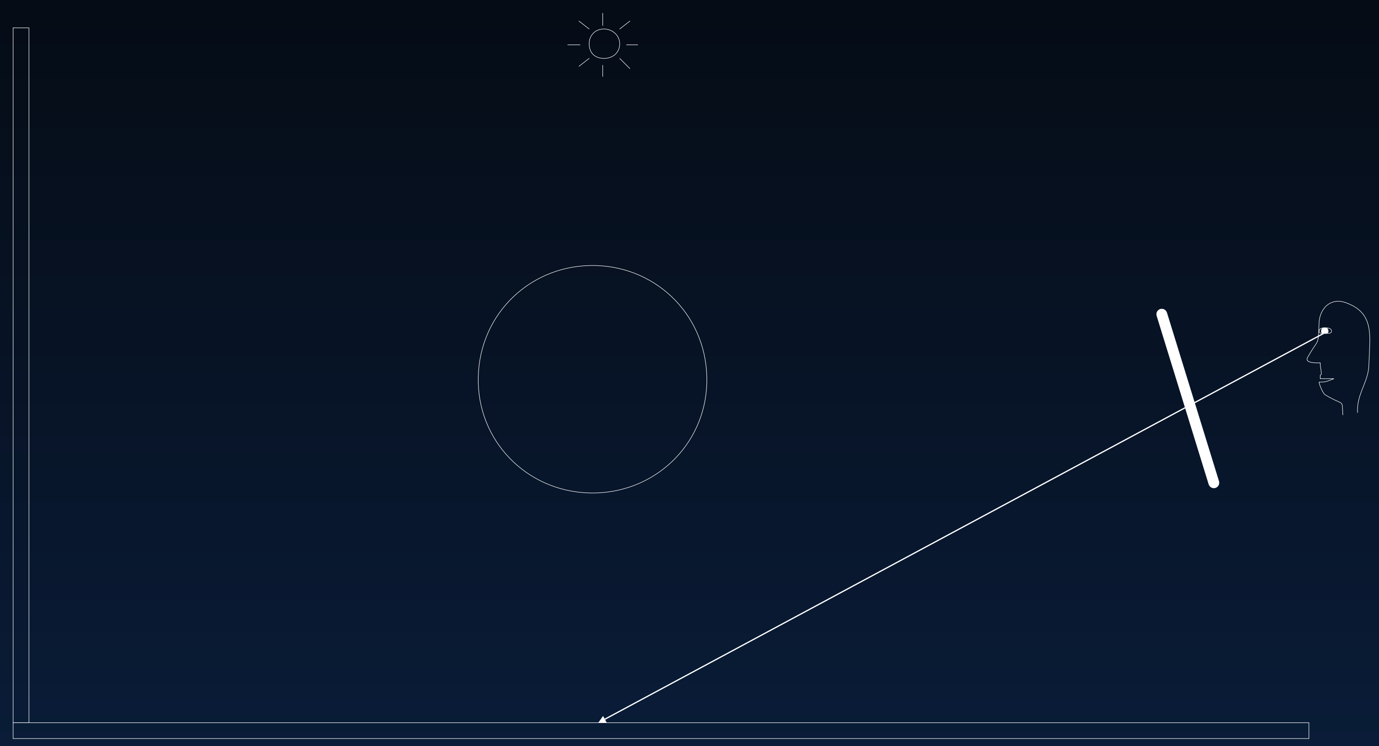
$$f_r(x, \vec{\omega}_i, \vec{\omega}_o) = w_1 f_{r,1}(x, \vec{\omega}_i, \vec{\omega}_o) + w_2 f_{r,2}(x, \vec{\omega}_i, \vec{\omega}_o)$$

Sampling a BRDF

$$f_r(x, \vec{\omega}_i, \vec{\omega}_o) = w_1 \cdot f_{r,d} + w_2 \cdot f_{r,s}$$

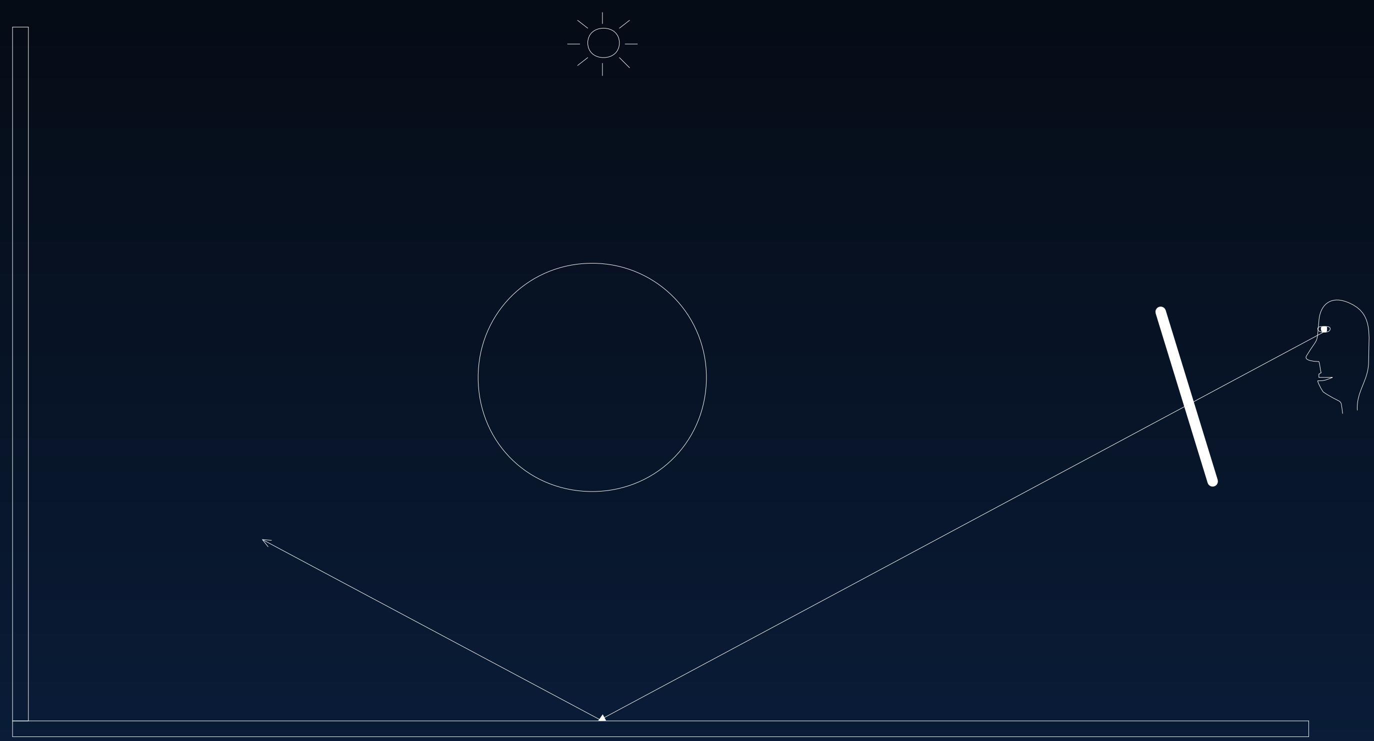
```
r = random()·(w_1 + w_2);
if ( r < w_1 )
  reflect diffuse photon
else
  reflect specular
```

Rendering

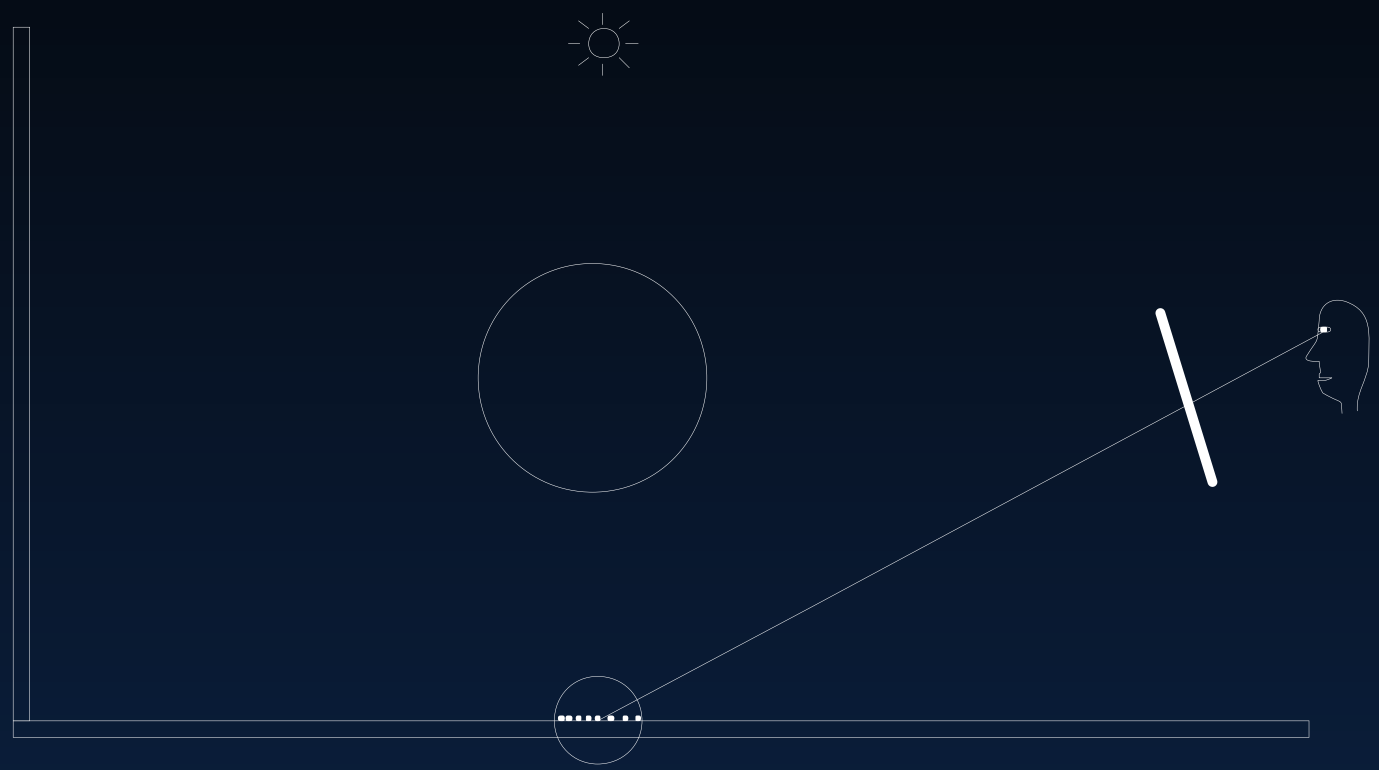


Direct Illumination

Specular Reflection



Caustics



Indirect Illumination

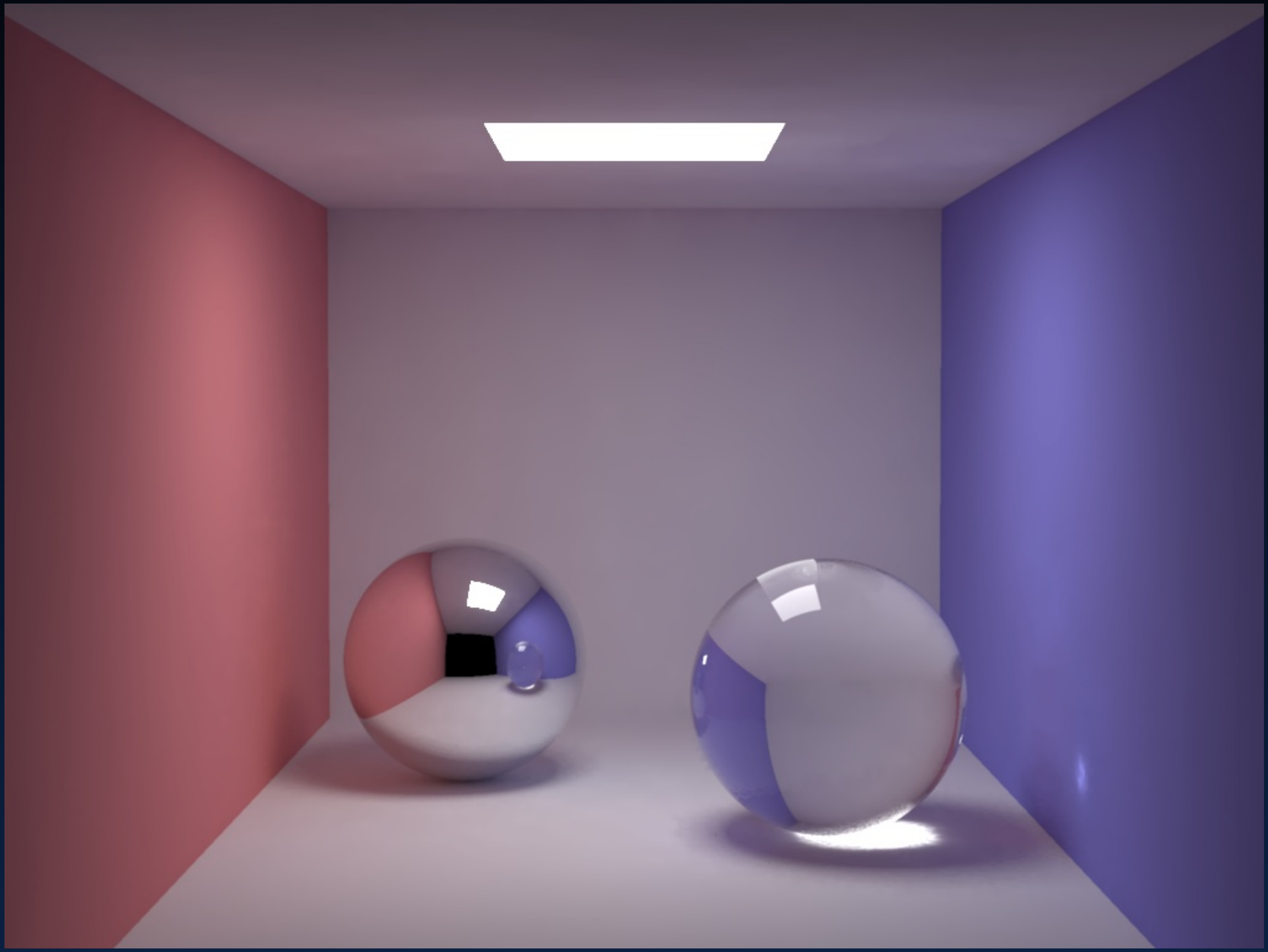
Rendering Equation Solution

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

Features

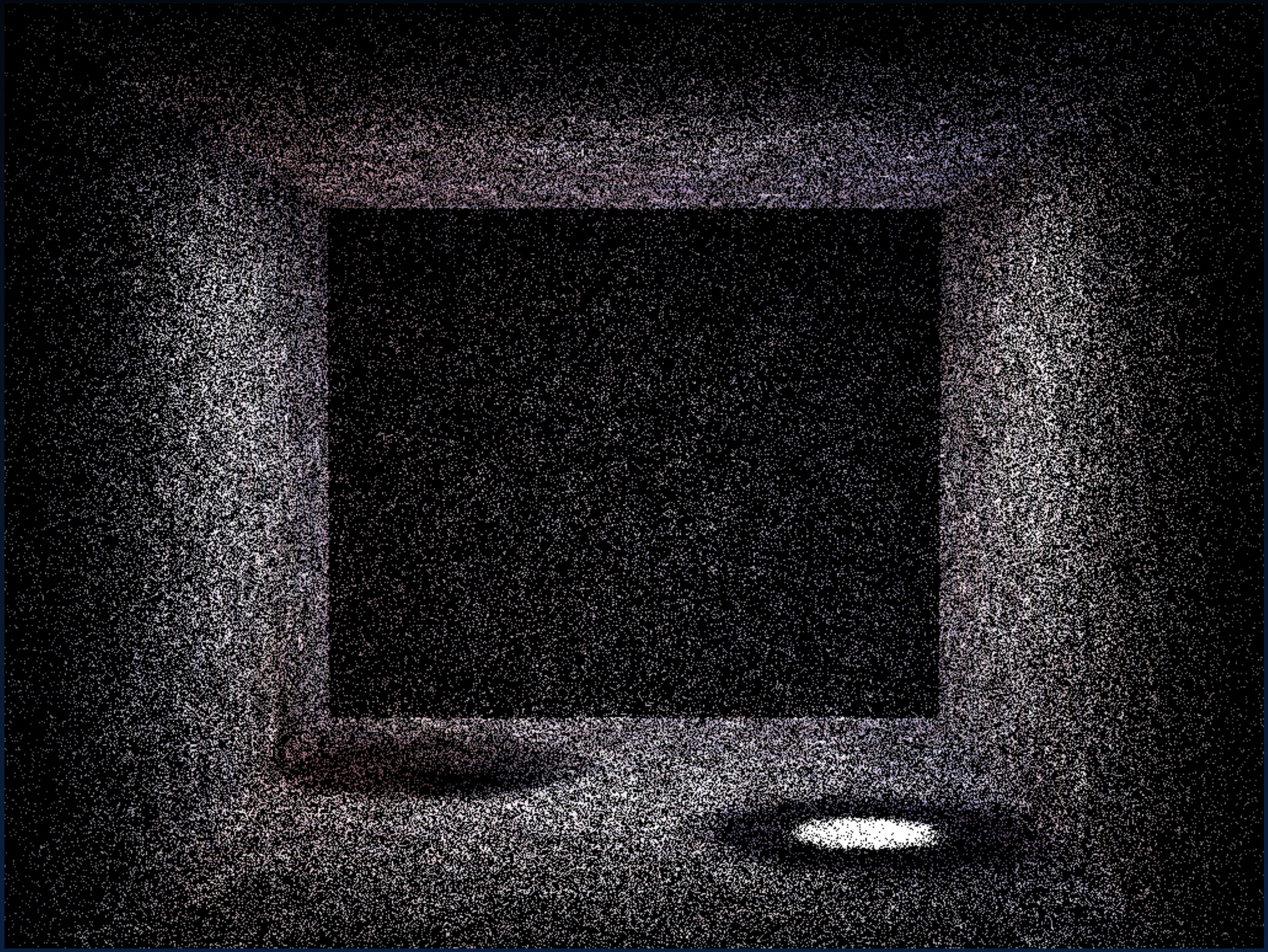
- Photon tracing is unbiased
  - Radiance estimate is biased but consistent
  - The reconstruction error is local
- Illumination representation is decoupled from the geometry

Box



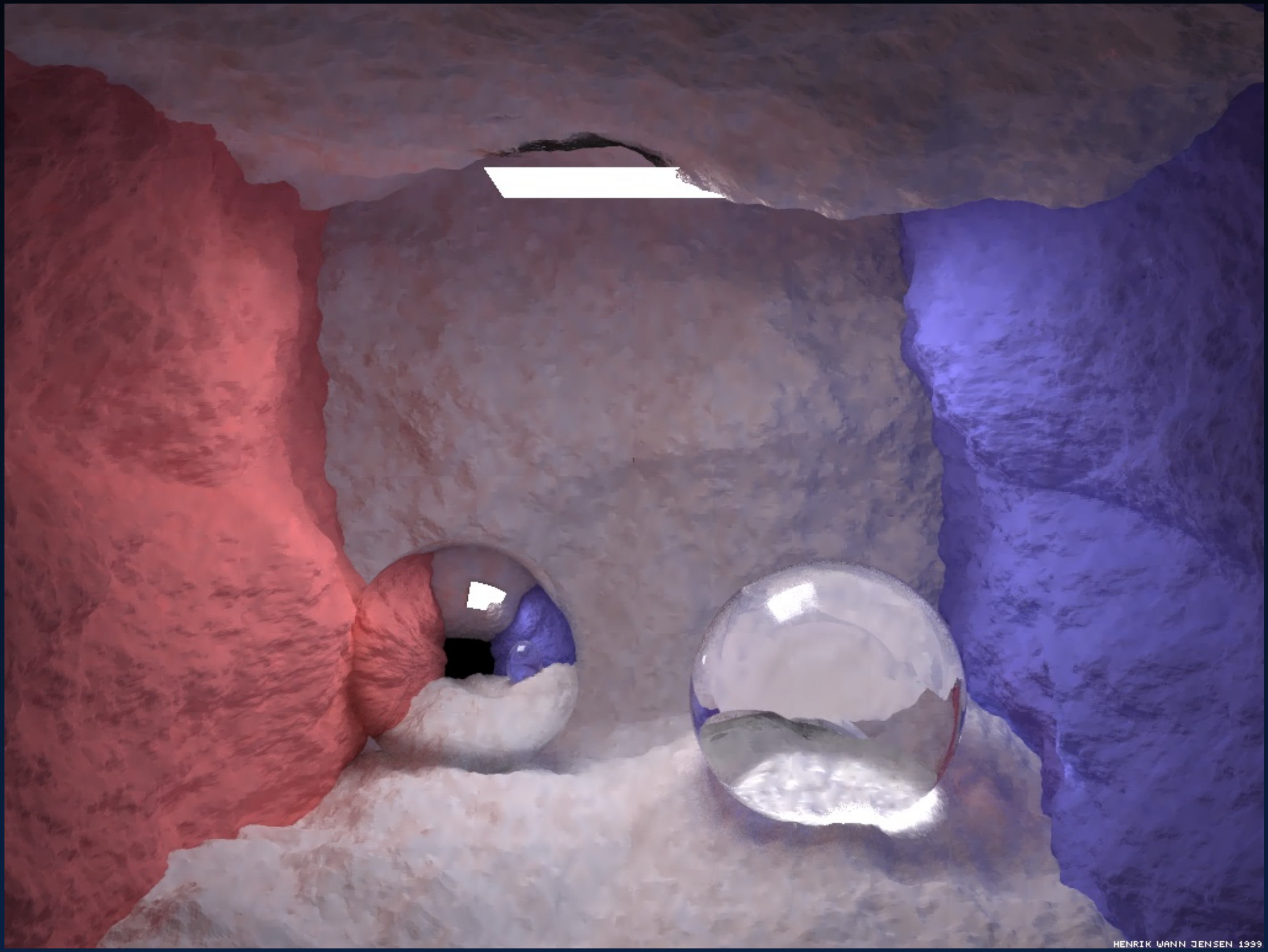
200000 global photons, 50000 caustic photons

Box: Global Photons



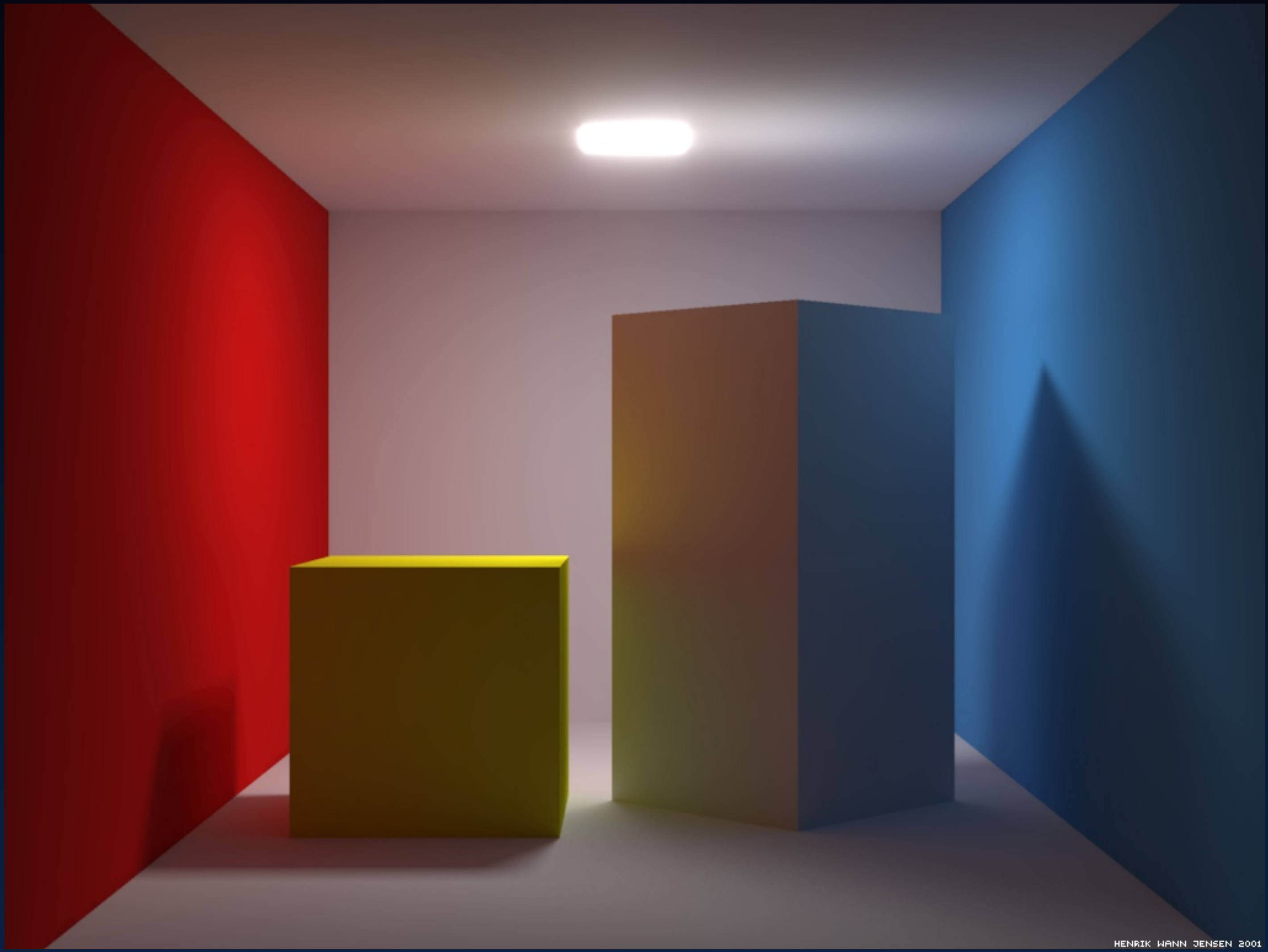
200000 global photons

Fractal Box



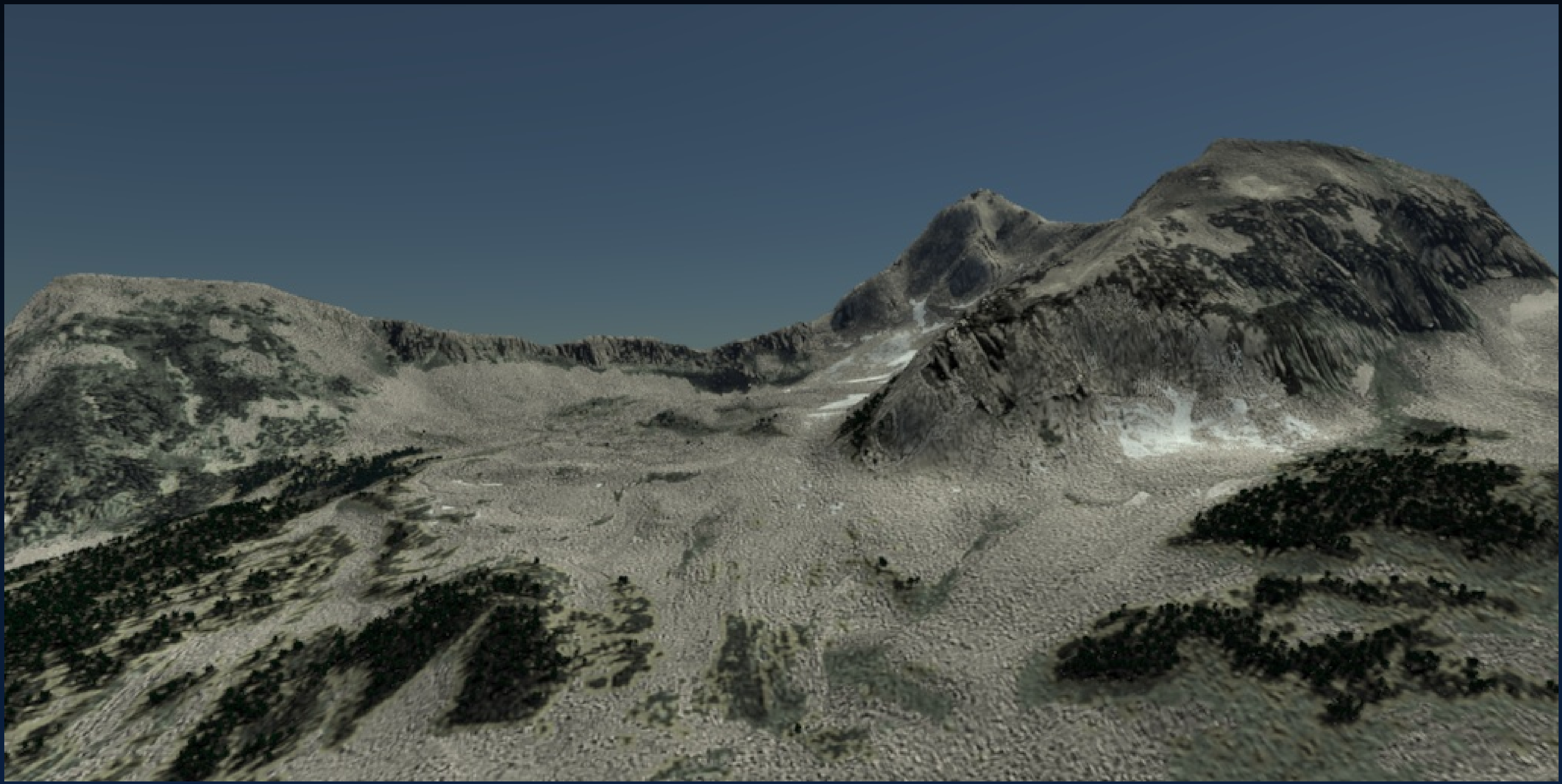
200000 global photons, 50000 caustic photons

Cornell Box



Indirect Illumination

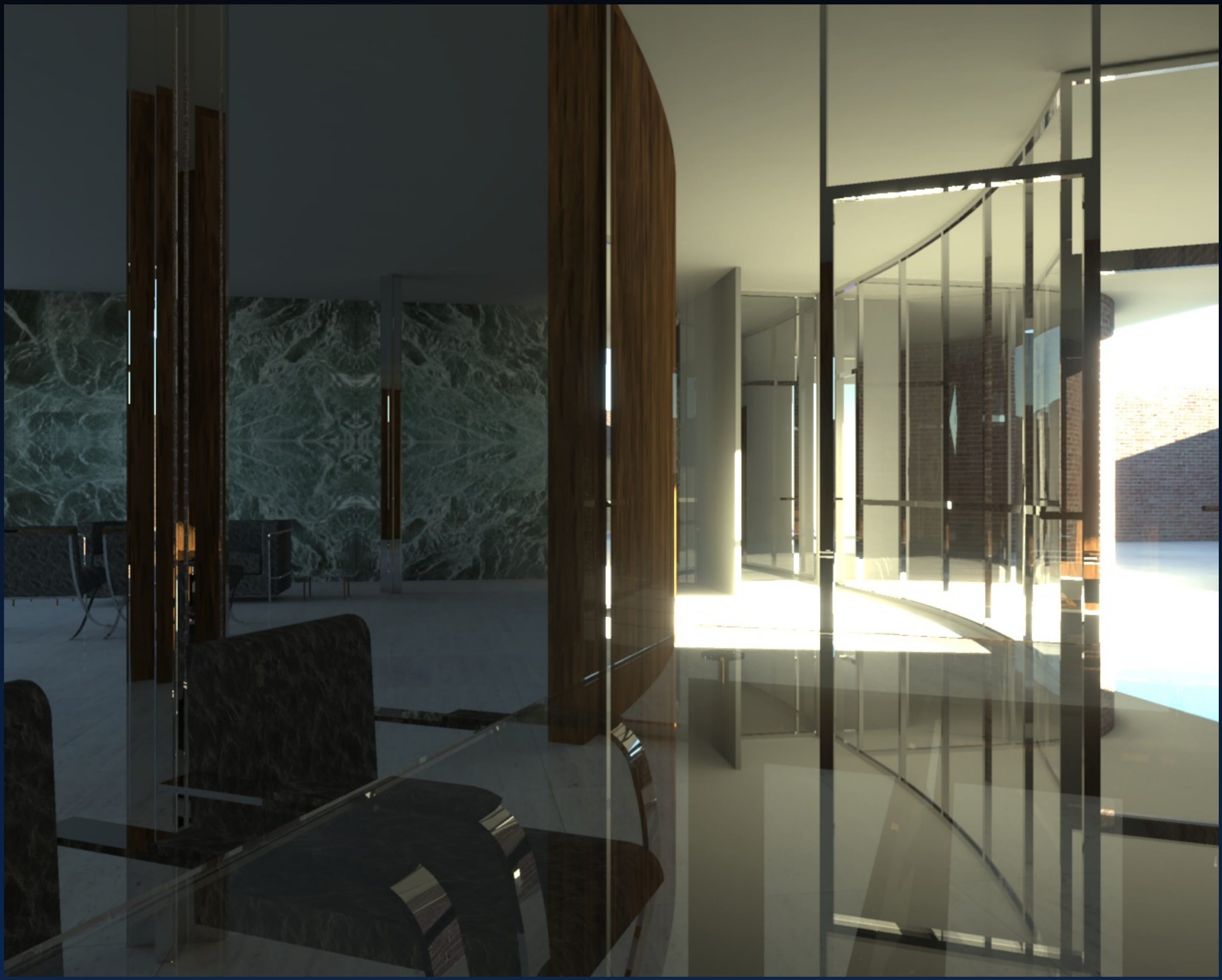
Little Matterhorn



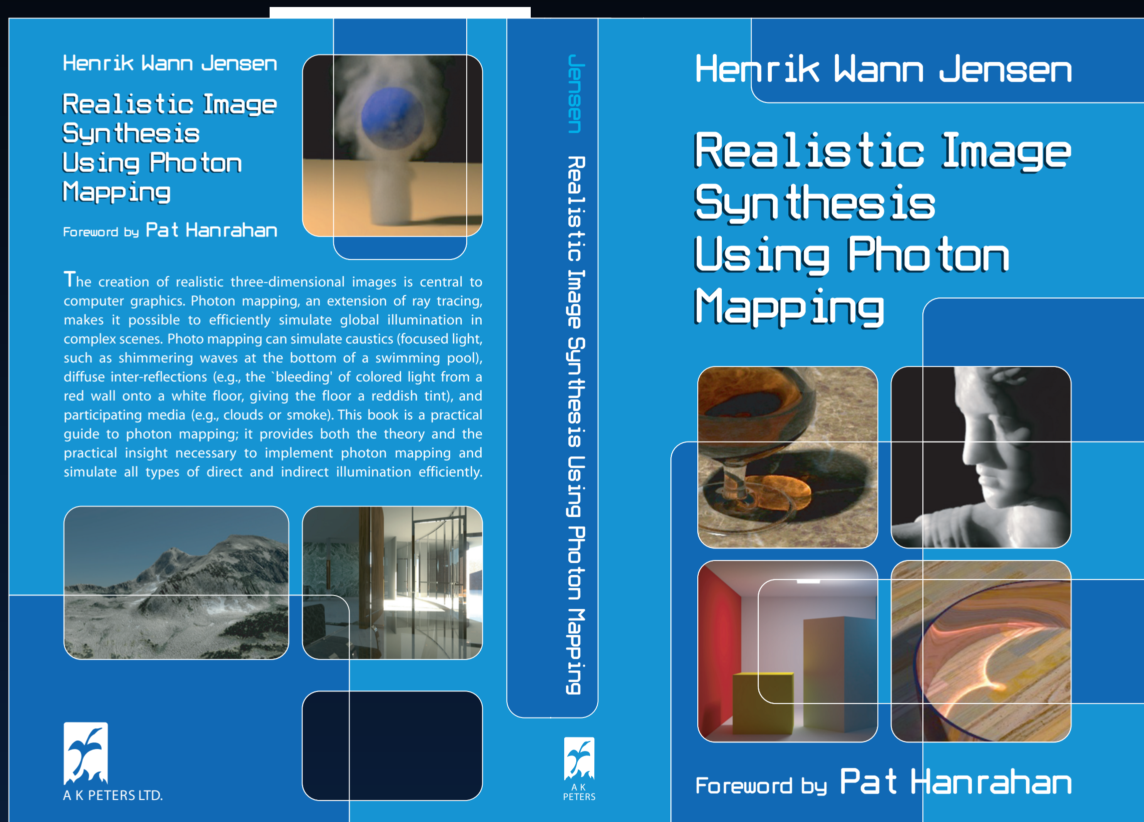
Mies house (swimmingpool)

Mies house (3pm)

Mies house (6pm)



More Information



http://graphics.stanford.edu/~henrik
henrik@graphics.stanford.edu