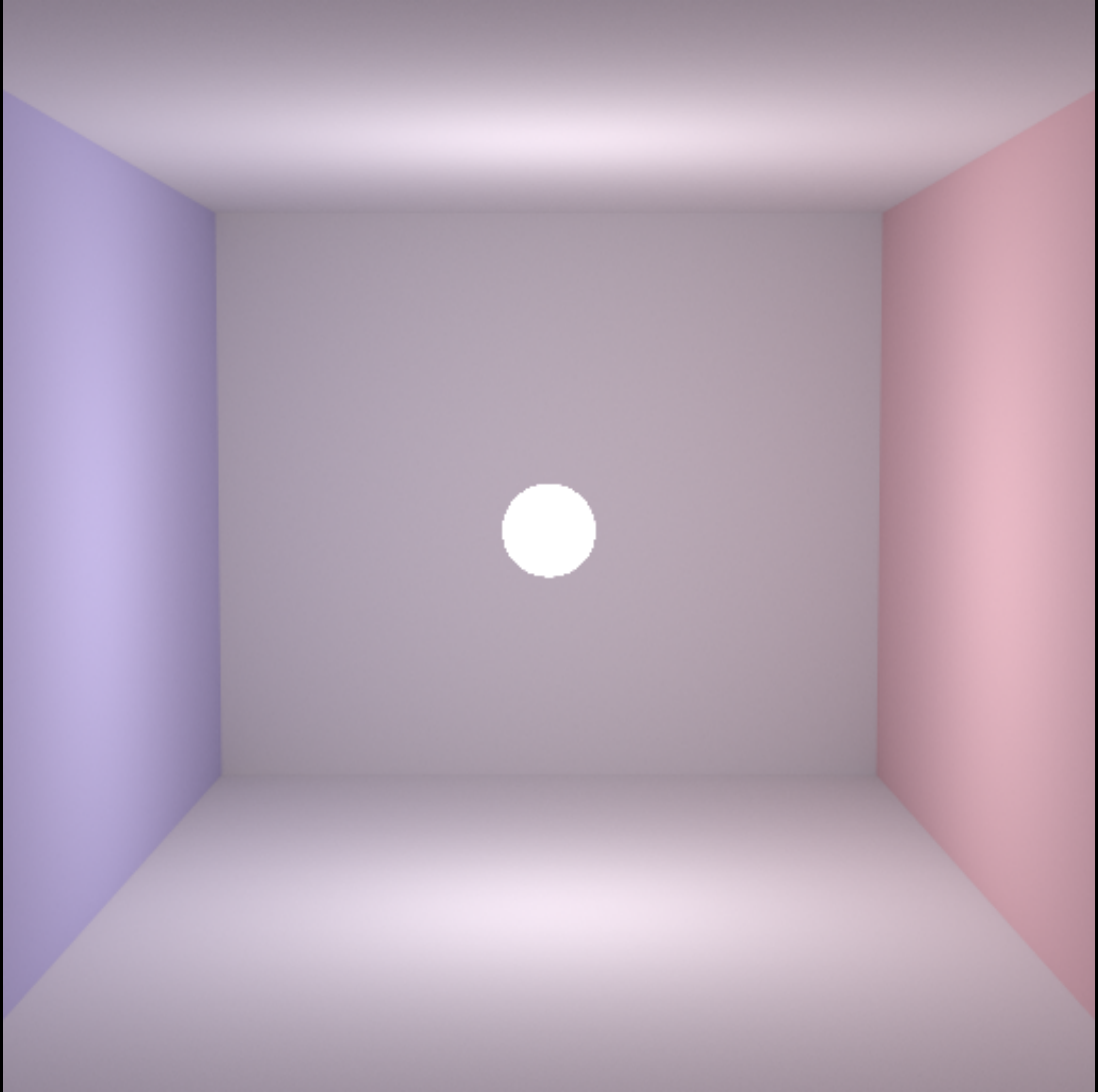
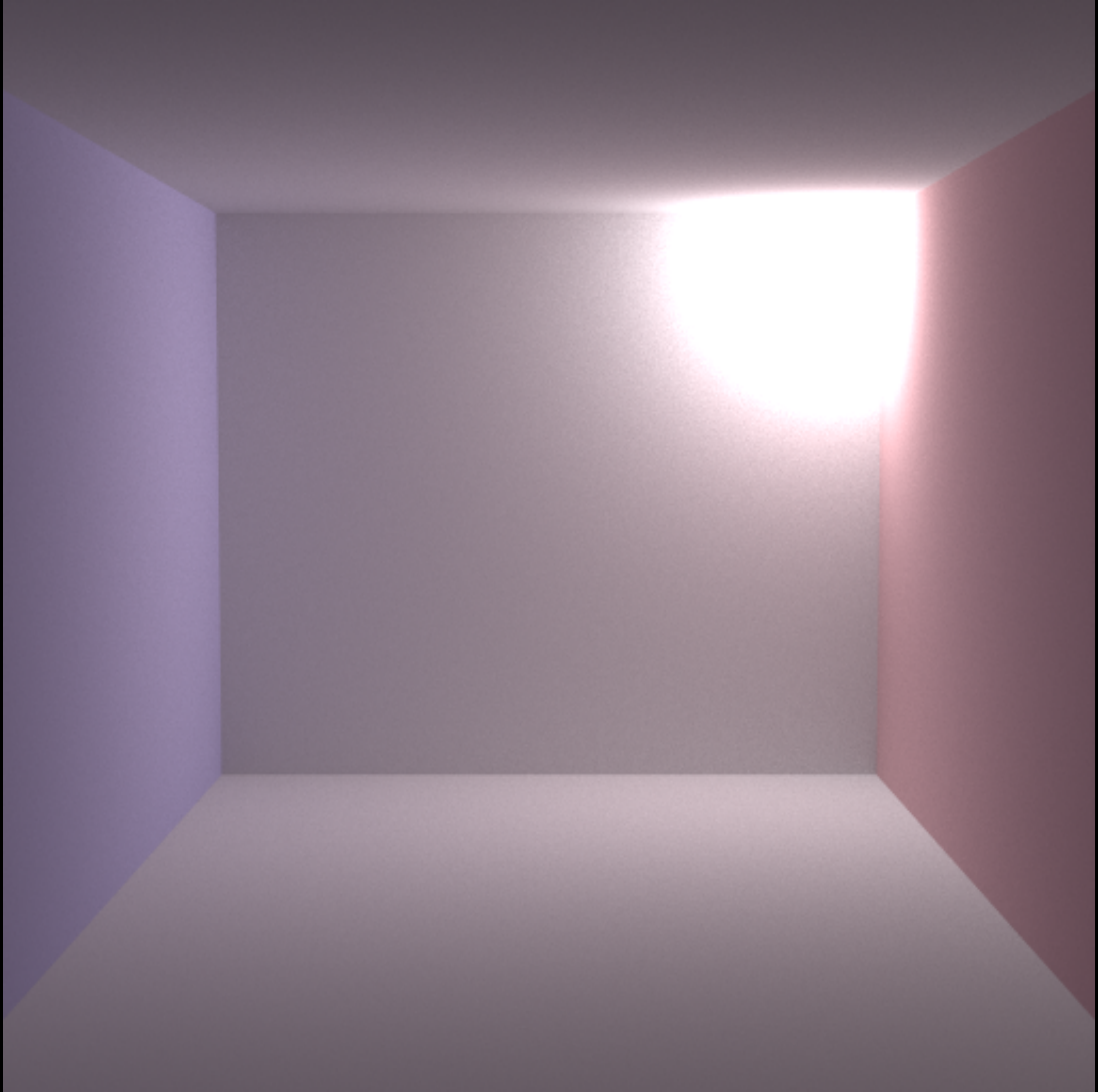
Bidirectional Light Transport

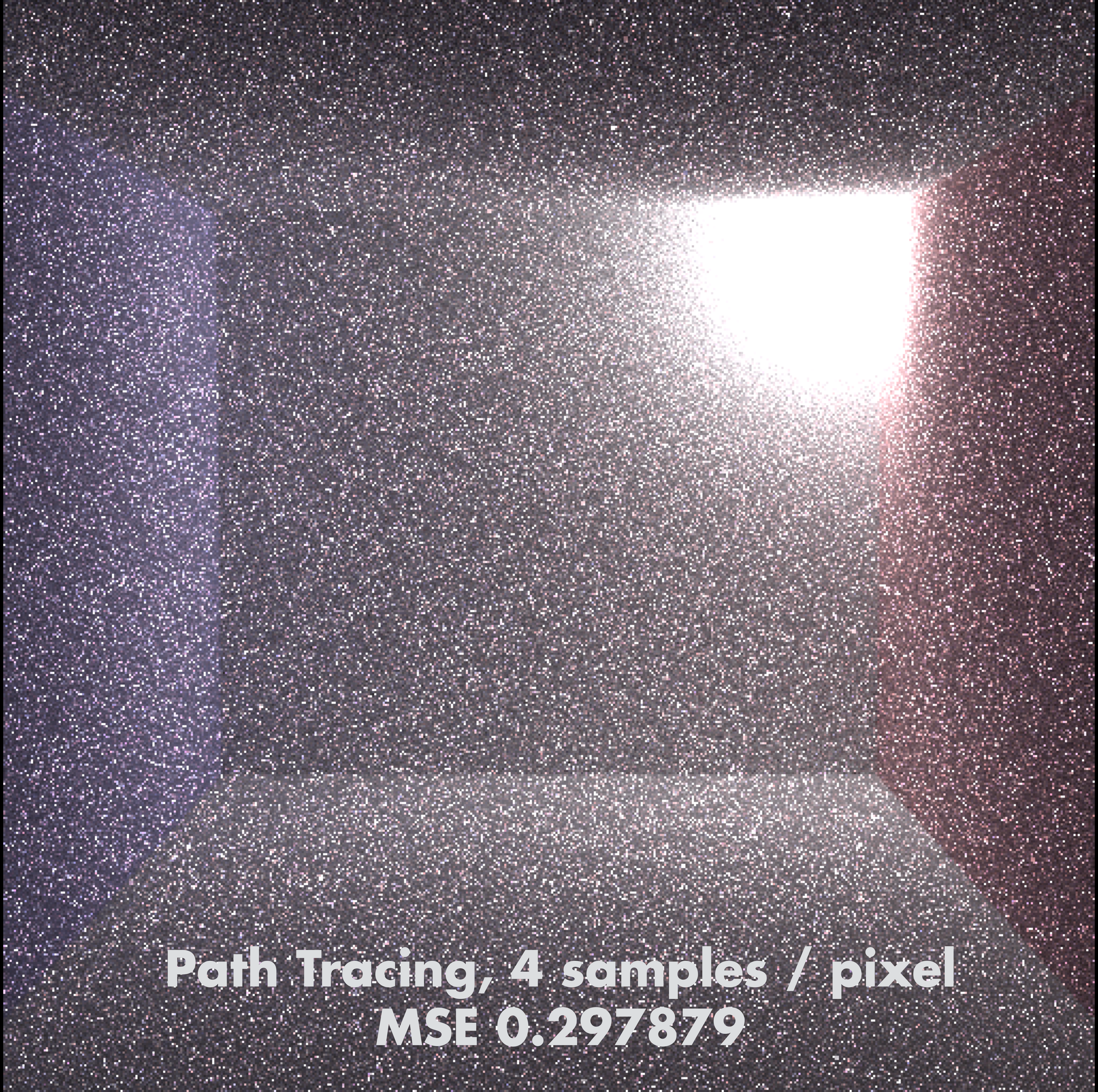
Today

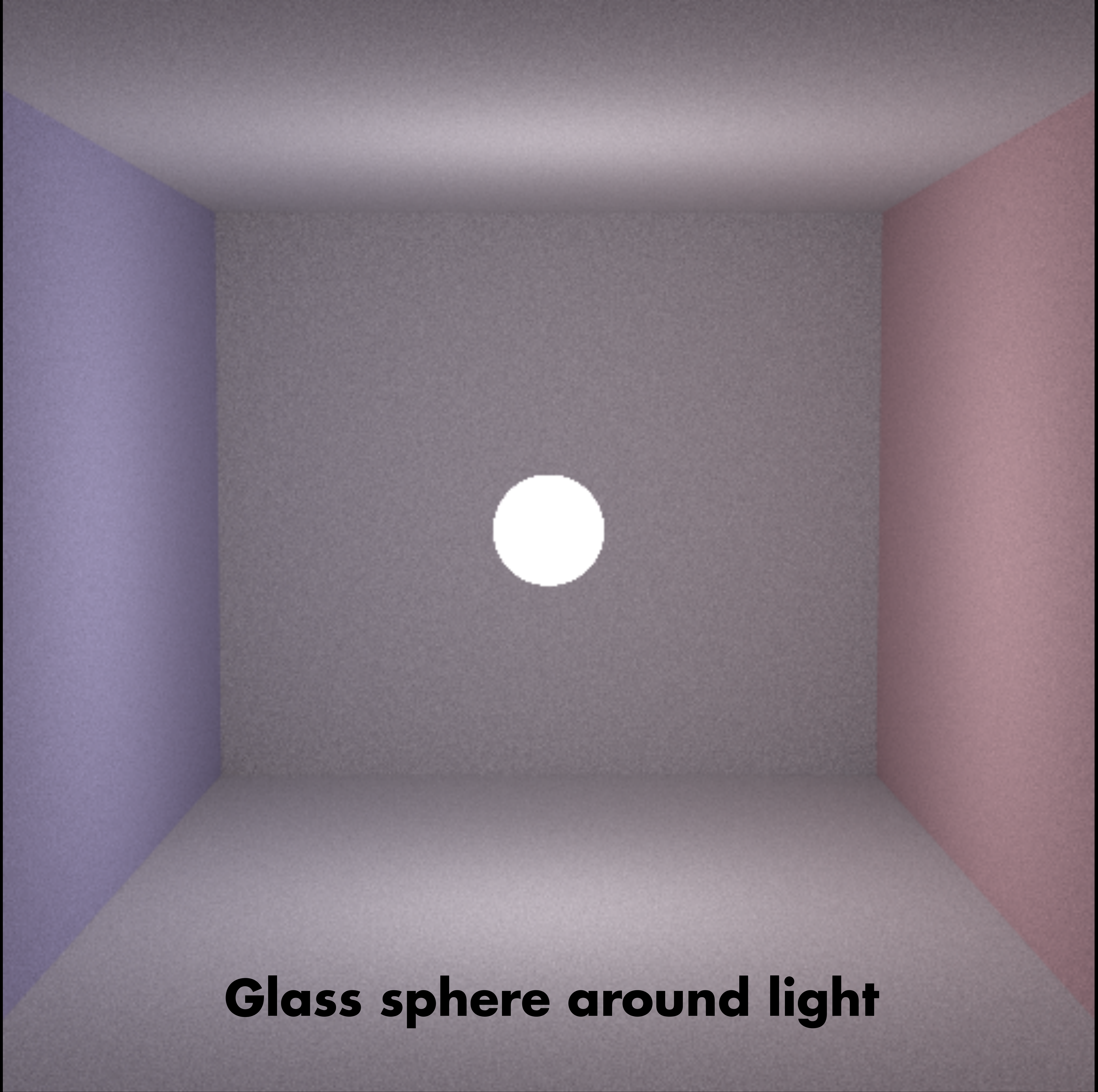
- Path tracing challenges
- Path integral form of the rendering equation
- Light tracing
- Bidirectional path tracing
- Photon mapping



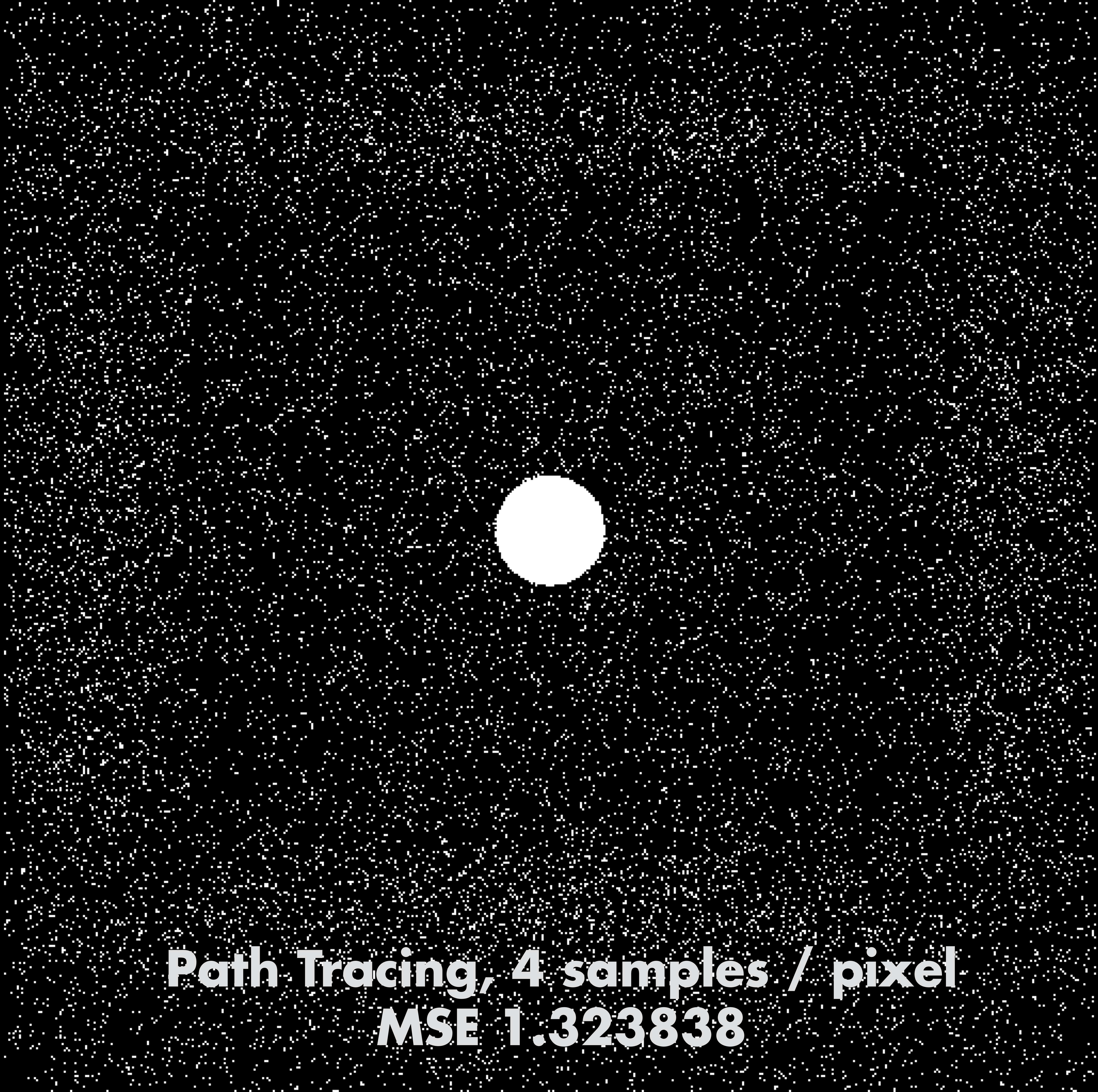

Path Tracing, 4 samples / pixel
MSE 0.050341



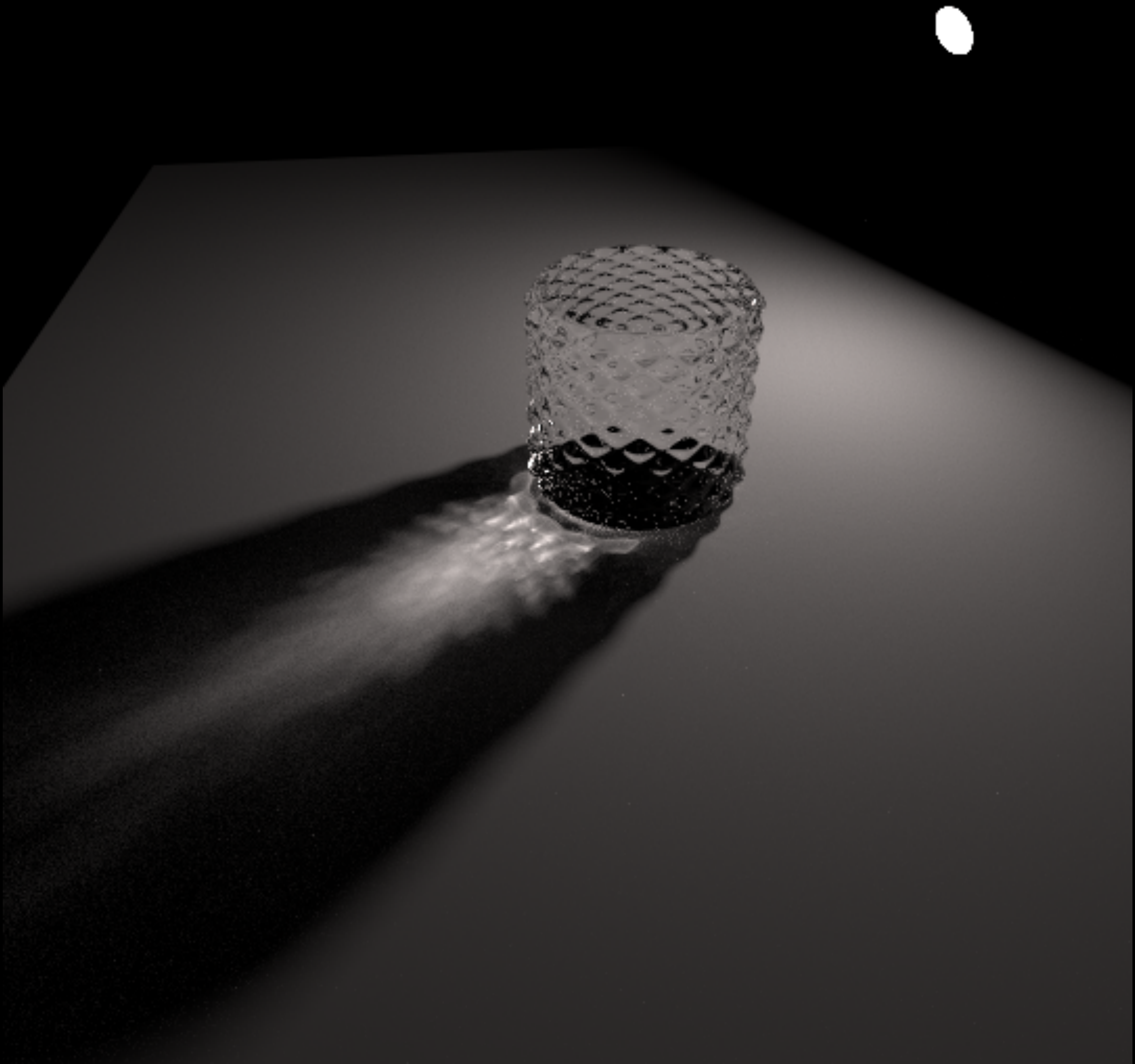

Path Tracing, 4 samples / pixel
MSE 0.297879

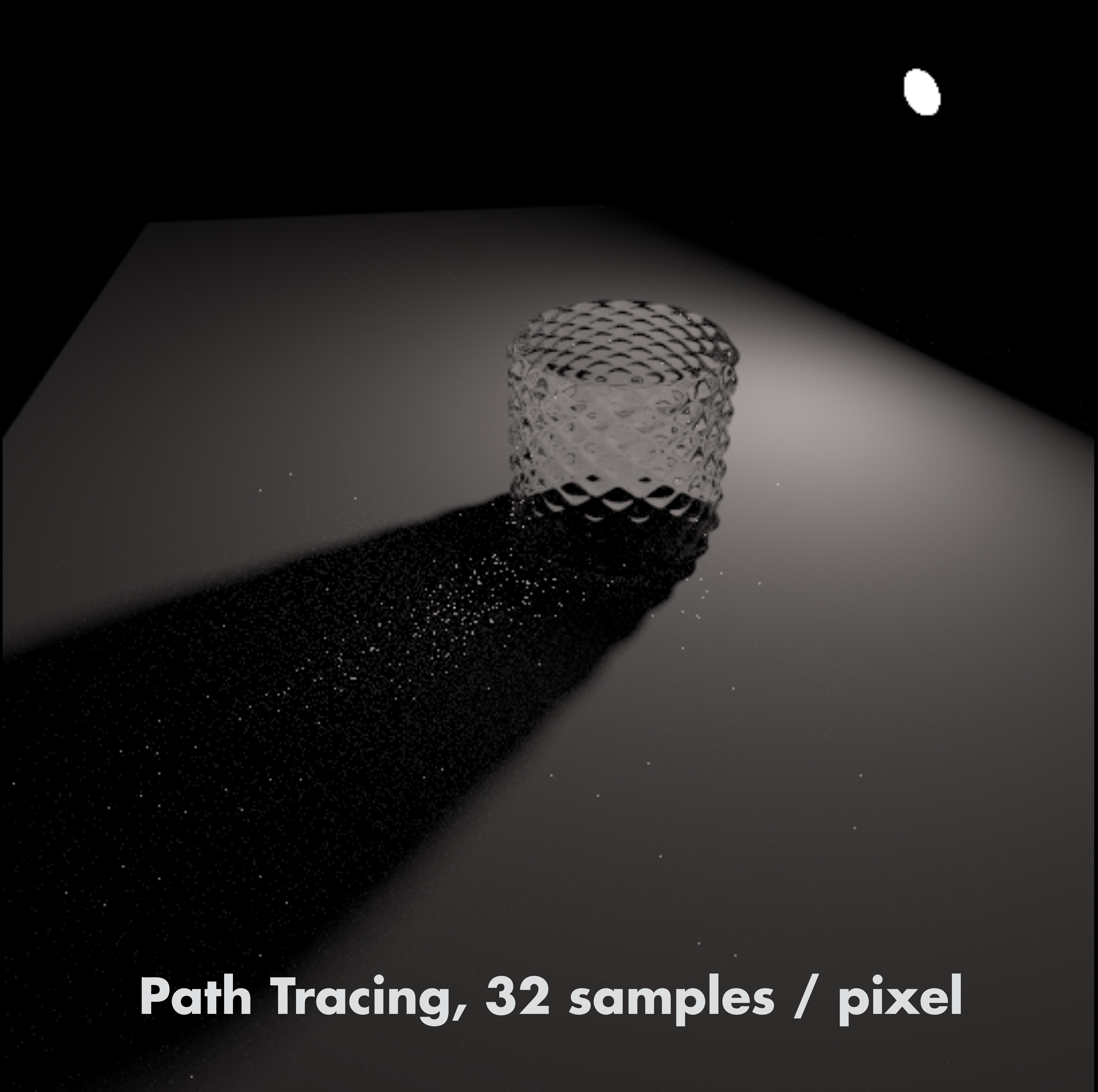


Glass sphere around light



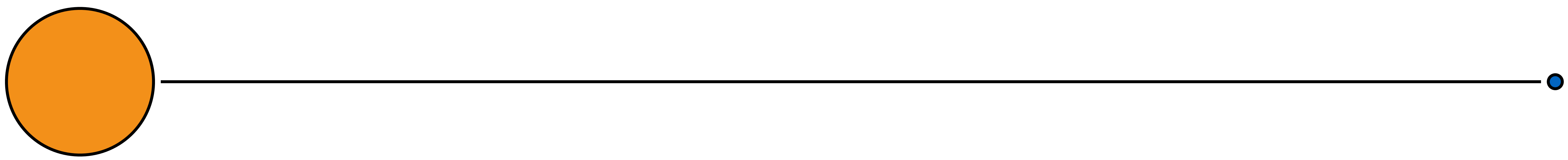
Path Tracing, 4 samples / pixel
MSE 1.323838





Path Tracing, 32 samples / pixel

Sun vs. Earth Solid Angles



r=695.7M m   149.6B m   r=6.371M m

	Solid Angle	% of Sphere
Sun f/Earth	6.87e-5 sr	0.00547%
Earth f/sun	5.70e-9 sr	0.0000000454%
1 m^2 on Earth f/sun	4.47e-17 sr	0.0000000000000003555%

Rendering Equation: Directional Form

$$L(\mathrm{p}, \omega) = L_e(\mathrm{p}, \omega) + \int_{H^2} f_r(\mathrm{p}, \omega_i \to \omega) L(tr(\mathrm{p}, \omega_i), -\omega_i) \cos\theta_i \, \mathrm{d}\omega_i$$

Can equivalently write as an integral over surface area of objects in the scene by applying

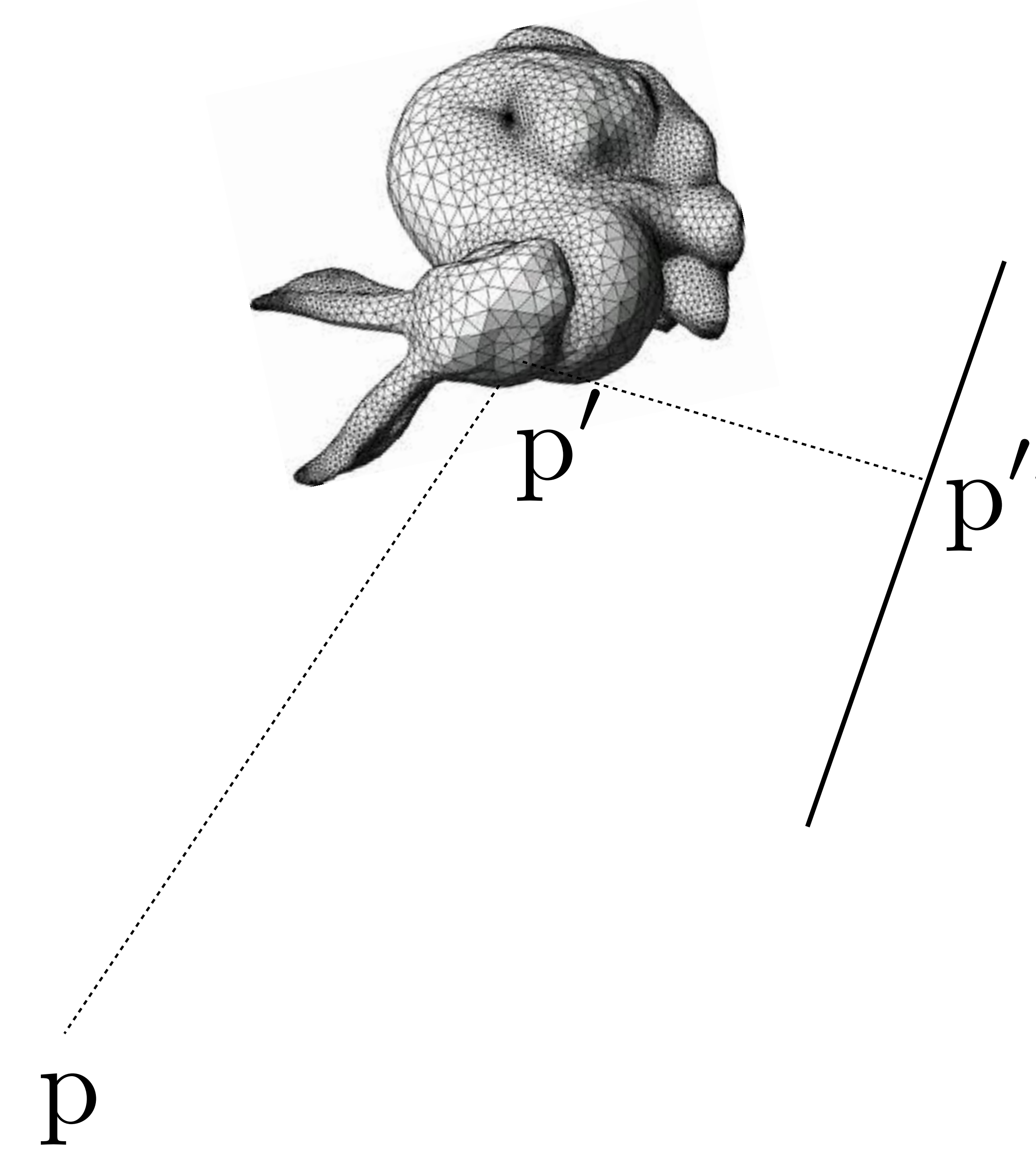
$$\mathrm{d}\omega = \frac{\cos\theta}{r^2} \mathrm{d}A$$

Rendering Equation: Area Form

$$L(\mathrm{p}' \to \mathrm{p}) = L_e(\mathrm{p}' \to \mathrm{p}) + \int_A f_r(\mathrm{p}'' \to \mathrm{p}' \to \mathrm{p})\, L(\mathrm{p}'' \to \mathrm{p}')\, G(\mathrm{p}'' \leftrightarrow \mathrm{p}')\, \mathrm{dp}''$$

Geometric coupling term

$$G(\mathrm{p}'' \leftrightarrow \mathrm{p}') = \frac{\cos\theta' \cos\theta''\, V(\mathrm{p}'' \leftrightarrow \mathrm{p}')}{||\mathrm{p}'' - \mathrm{p}'||^2}$$



Binary visibility function

$V(\mathrm{p} \leftrightarrow \mathrm{p}')$

Recursive Expansion

$$L(\mathrm{p}' \to \mathrm{p}) = L_e(\mathrm{p}' \to \mathrm{p}) + \int_A f_r(\mathrm{p}'' \to \mathrm{p}' \to \mathrm{p})\, L(\mathrm{p}'' \to \mathrm{p}')\, G(\mathrm{p}'' \leftrightarrow \mathrm{p}')\, \mathrm{d}\mathrm{p}''$$

$$L(\mathrm{p}' \to \mathrm{p}) = L_e(\mathrm{p}' \to \mathrm{p}) + \int_A f_r(\mathrm{p}'' \to \mathrm{p}' \to \mathrm{p}) \left[L_e(\mathrm{p}'' \to \mathrm{p}') + \int_A f_r(\mathrm{p}''' \to \mathrm{p}'' \to \mathrm{p}')\, L(\mathrm{p}''' \to \mathrm{p}'')\, G(\mathrm{p}''' \leftrightarrow \mathrm{p}'')\, \mathrm{d}\mathrm{p}''' \right] G(\mathrm{p}'' \leftrightarrow \mathrm{p}')\, \mathrm{d}\mathrm{p}''$$

Recursive Expansion

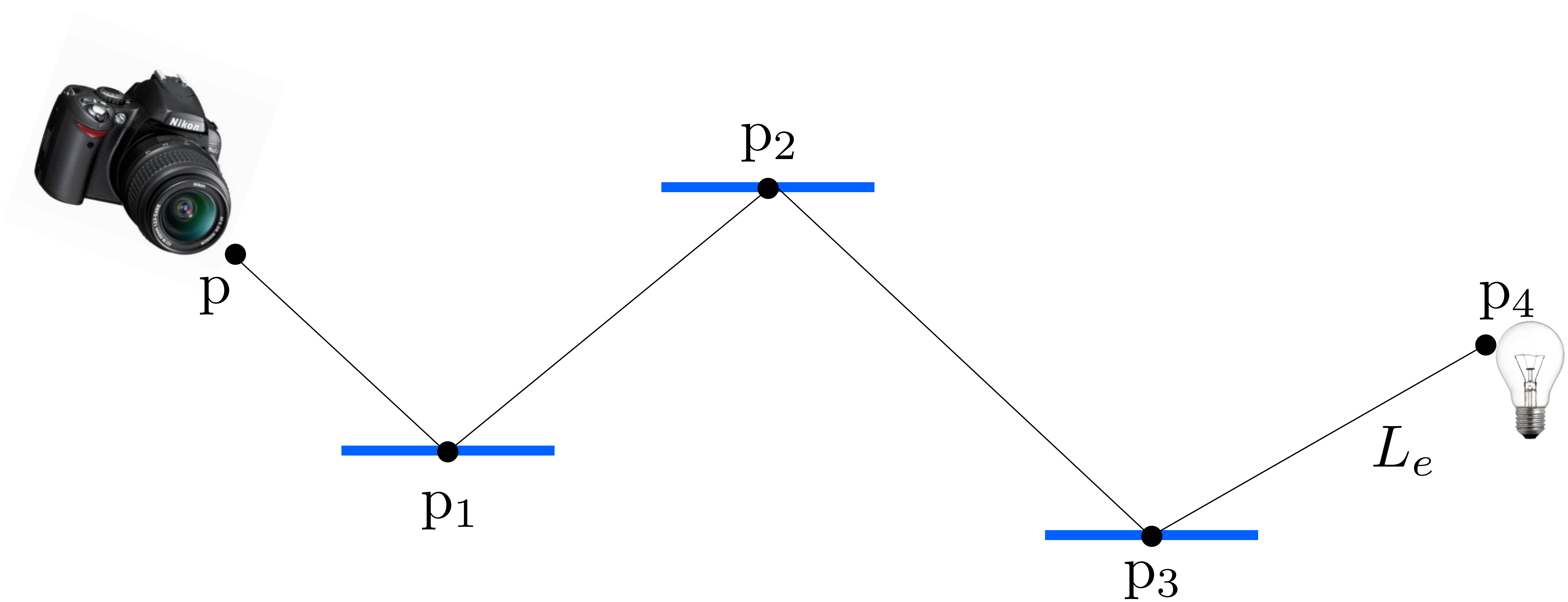
$$L(\mathrm{p}' \to \mathrm{p}) = L_e(\mathrm{p}' \to \mathrm{p}) +$$
$$\int_A f_r(\mathrm{p}'' \to \mathrm{p}' \to \mathrm{p})$$
$$\left[L_e(\mathrm{p}'' \to \mathrm{p}') + \int_A f_r(\mathrm{p}''' \to \mathrm{p}'' \to \mathrm{p}')\, L(\mathrm{p}''' \to \mathrm{p}'')\, G(\mathrm{p}''' \leftrightarrow \mathrm{p}'')\, \mathrm{dp}''' \right]$$
$$G(\mathrm{p}'' \leftrightarrow \mathrm{p}')\, \mathrm{dp}''$$

$$L(\mathrm{p}' \to \mathrm{p}) = L_e(\mathrm{p}' \to \mathrm{p}) +$$
$$\int_A f_r(\mathrm{p}'' \to \mathrm{p}' \to \mathrm{p})\, L_e(\mathrm{p}'' \to \mathrm{p}')\, G(\mathrm{p}'' \leftrightarrow \mathrm{p}')\, \mathrm{dp}'' +$$
$$\int_A \int_A f_r(\mathrm{p}'' \to \mathrm{p}' \to \mathrm{p})\, G(\mathrm{p}'' \leftrightarrow \mathrm{p}')\, f_r(\mathrm{p}''' \to \mathrm{p}'' \to \mathrm{p}')\, G(\mathrm{p}''' \leftrightarrow \mathrm{p}'')\, L(\mathrm{p}''' \to \mathrm{p}'')\, \mathrm{dp}'' \mathrm{dp}'''$$

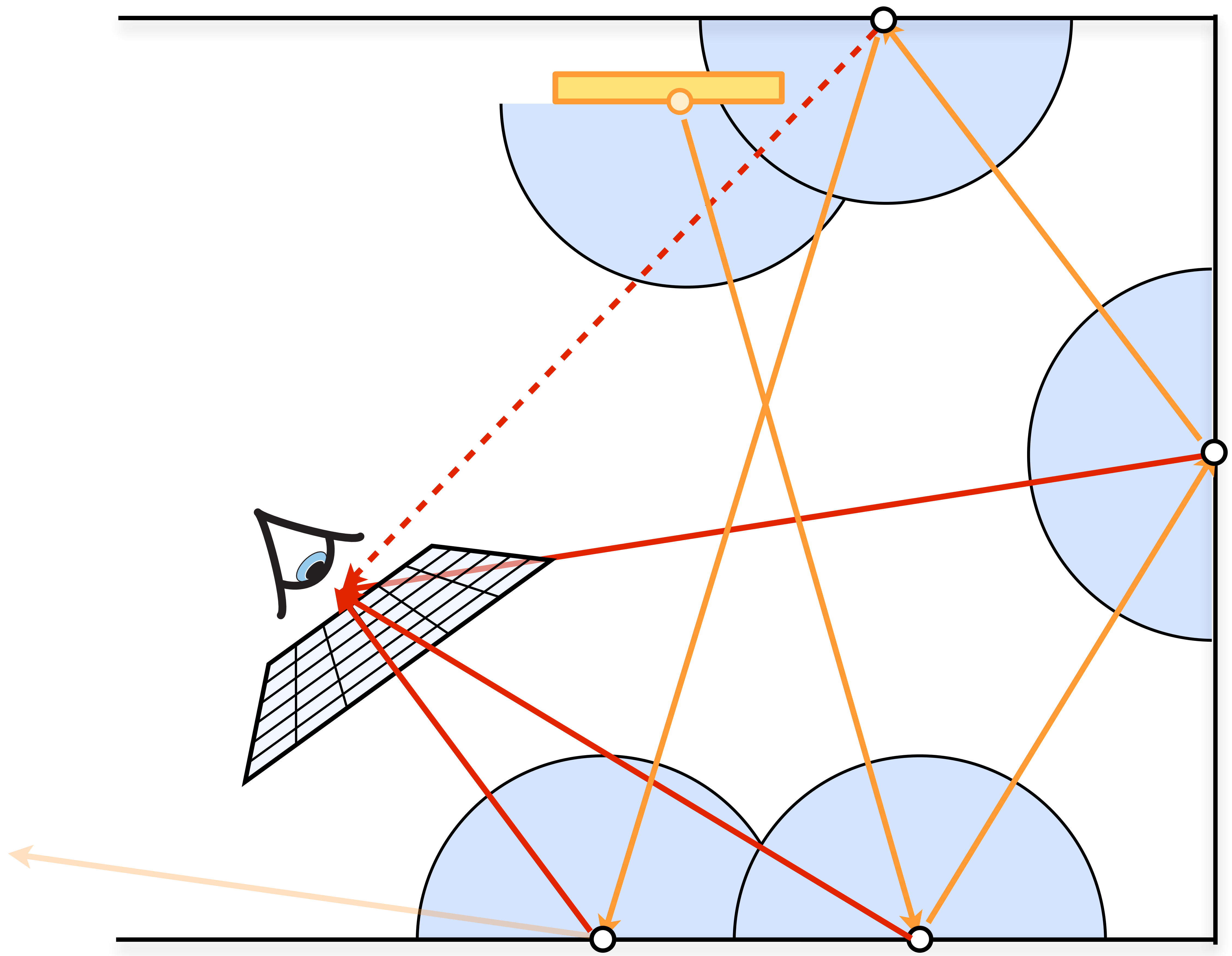
Rendering Equation: Sum Over Paths

$$L(\mathrm{p}_1 \to \mathrm{p}) = L_e(\mathrm{p}_1 \to \mathrm{p}) + \sum_i \left[\int_{A^i} T_i(\bar{\mathrm{p}})\, L_e(\mathrm{p}_{i+1} \to \mathrm{p}_i)\, d\mathrm{p}_{i+1} \dots d\mathrm{p}_2 \right]$$

$$T_n(\bar{\mathrm{p}}) = \prod_i^n f_r(\mathrm{p}_{i+1} \to \mathrm{p}_i \to \mathrm{p}_{i-1})\, G(\mathrm{p}_{i+1} \leftrightarrow \mathrm{p}_i)$$

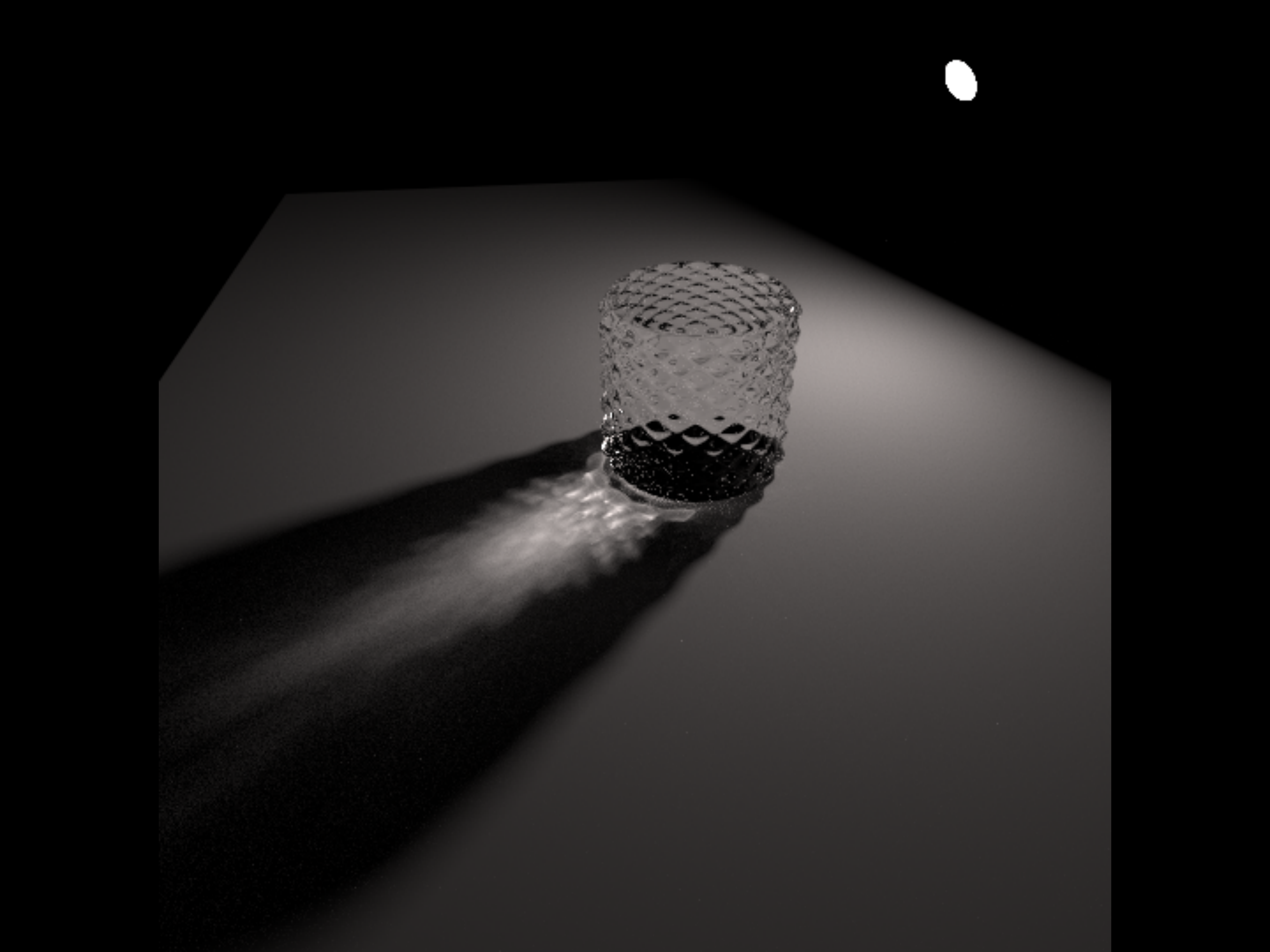


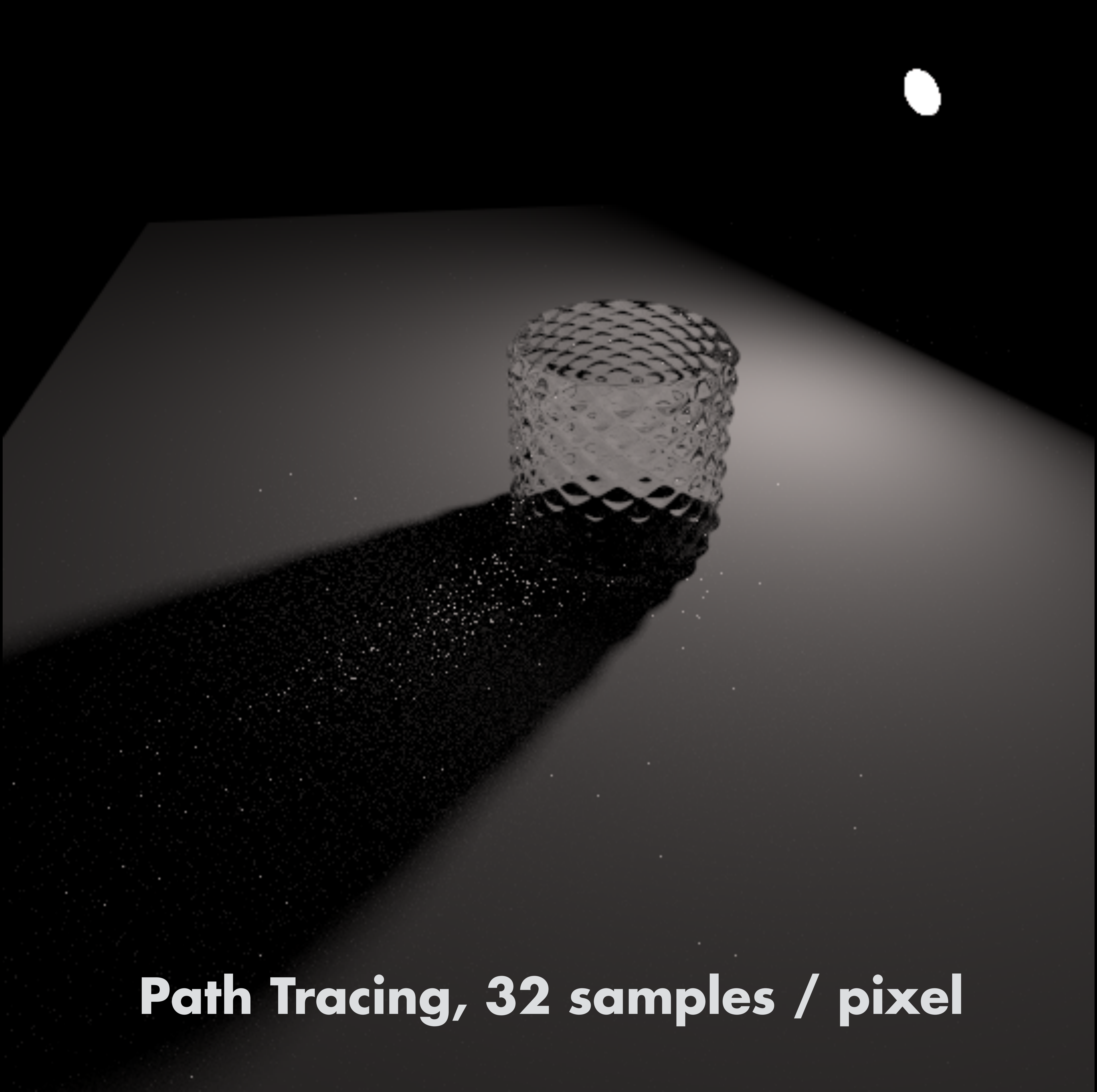
Light Tracing



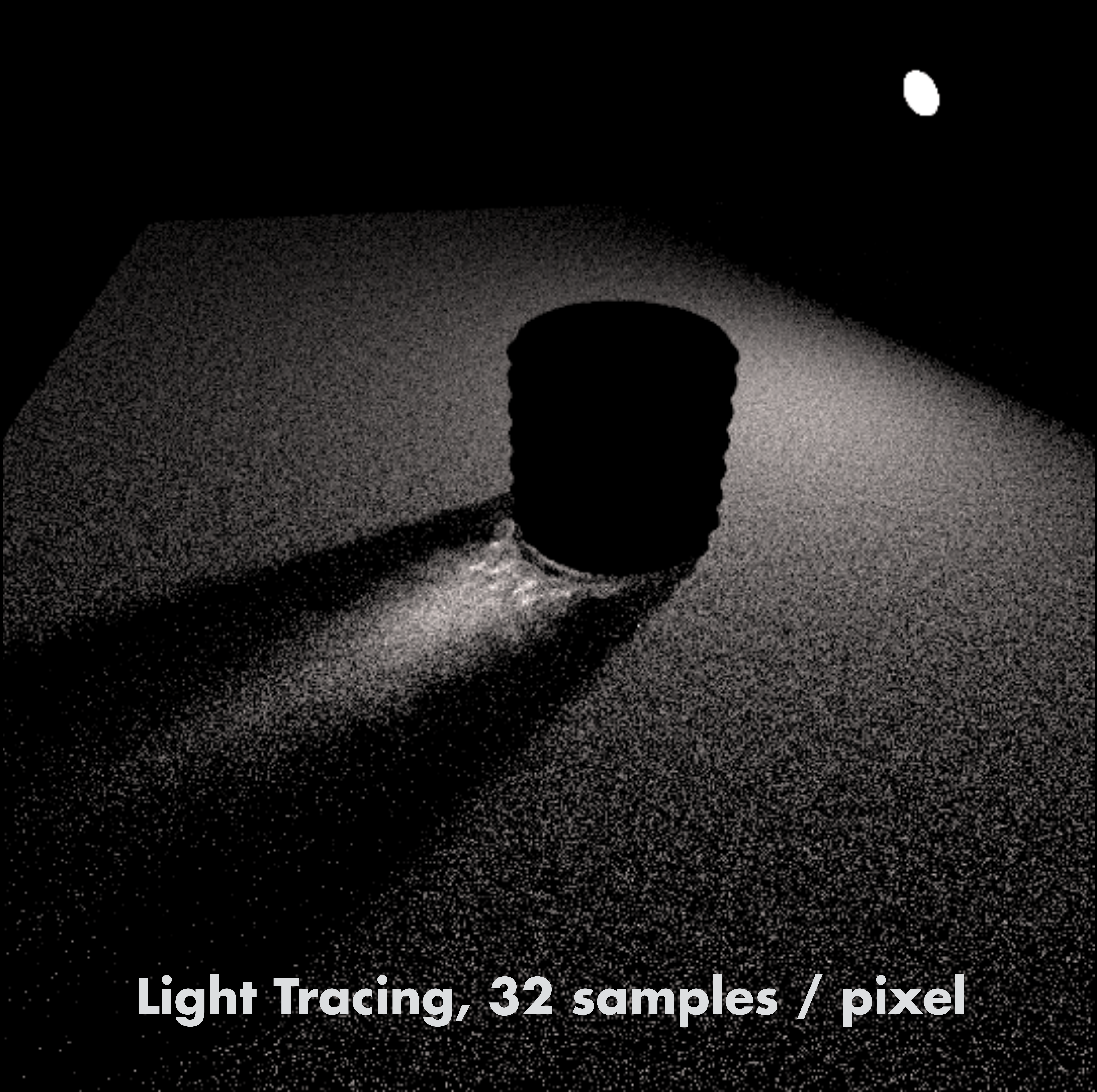
Splat to the image at each vertex

Wojciech Jarosz





Path Tracing, 32 samples / pixel



Light Tracing, 32 samples / pixel

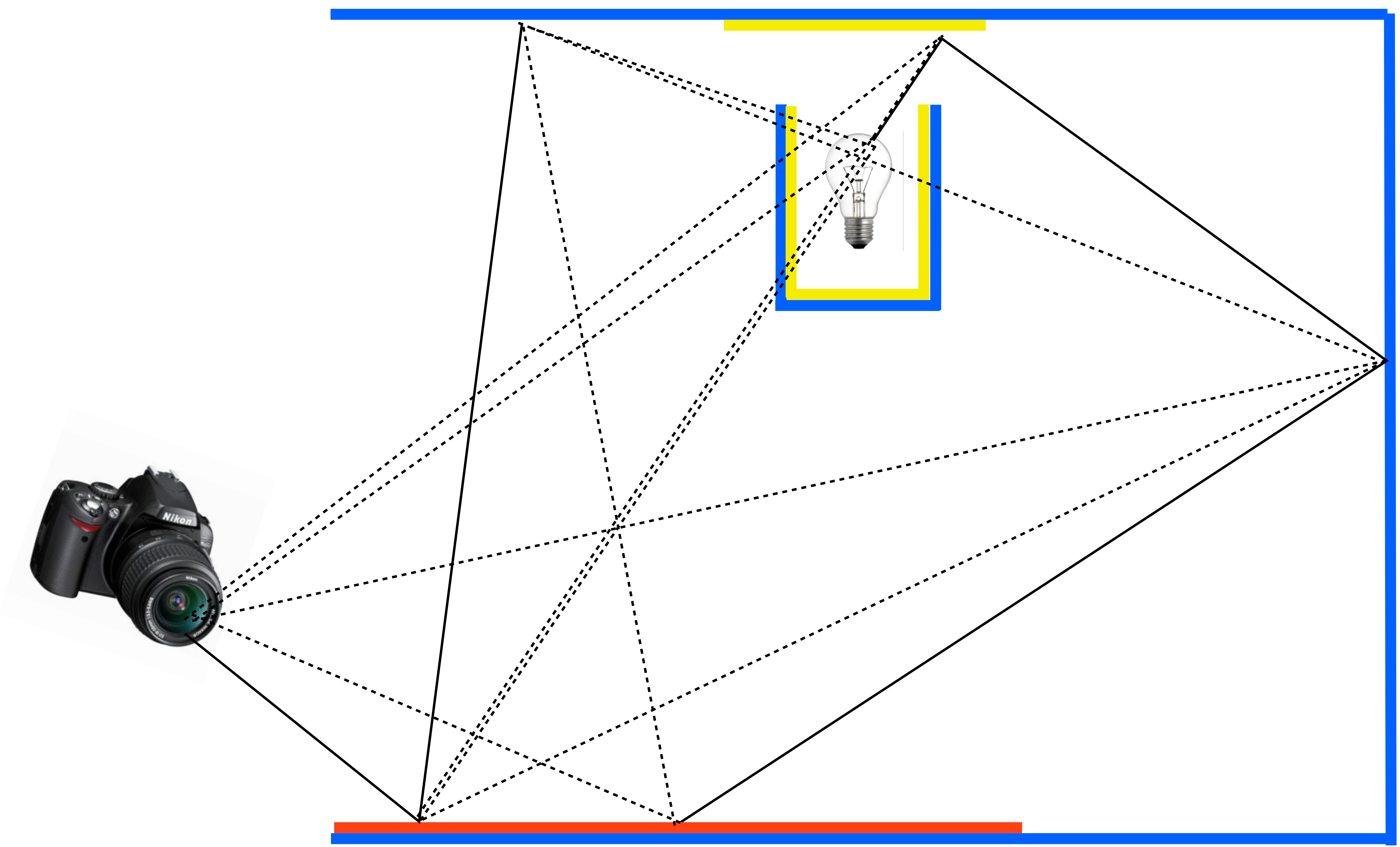
Bidirectional Path Tracing (BDPT)

Start paths from camera and lights

- Connect path vertices with shadow rays

Can handle difficult light sampling situations more robustly

Bidirectional Path Tracing (BDPT)

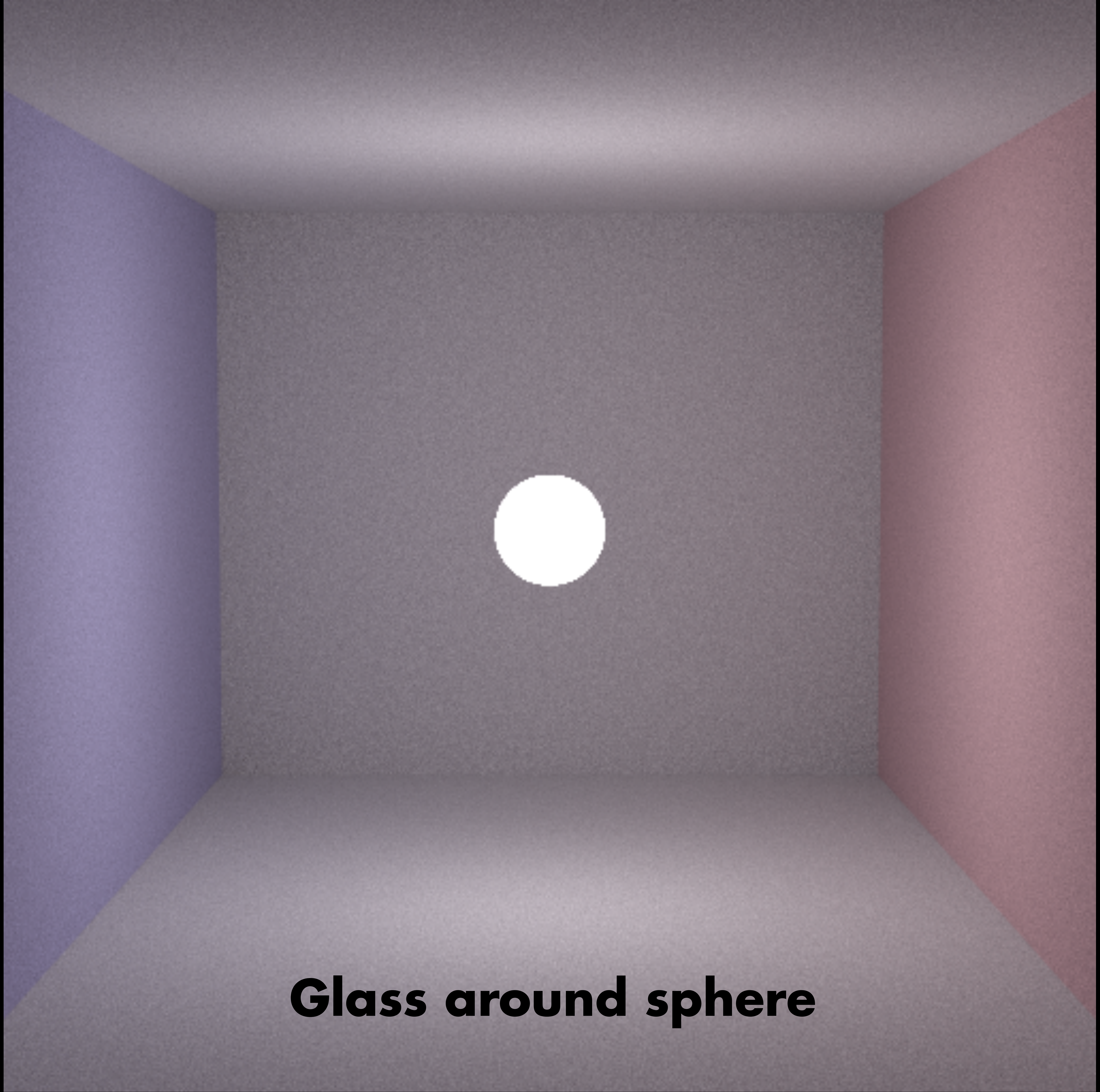


Bidirectional Path Tracing (BDPT)

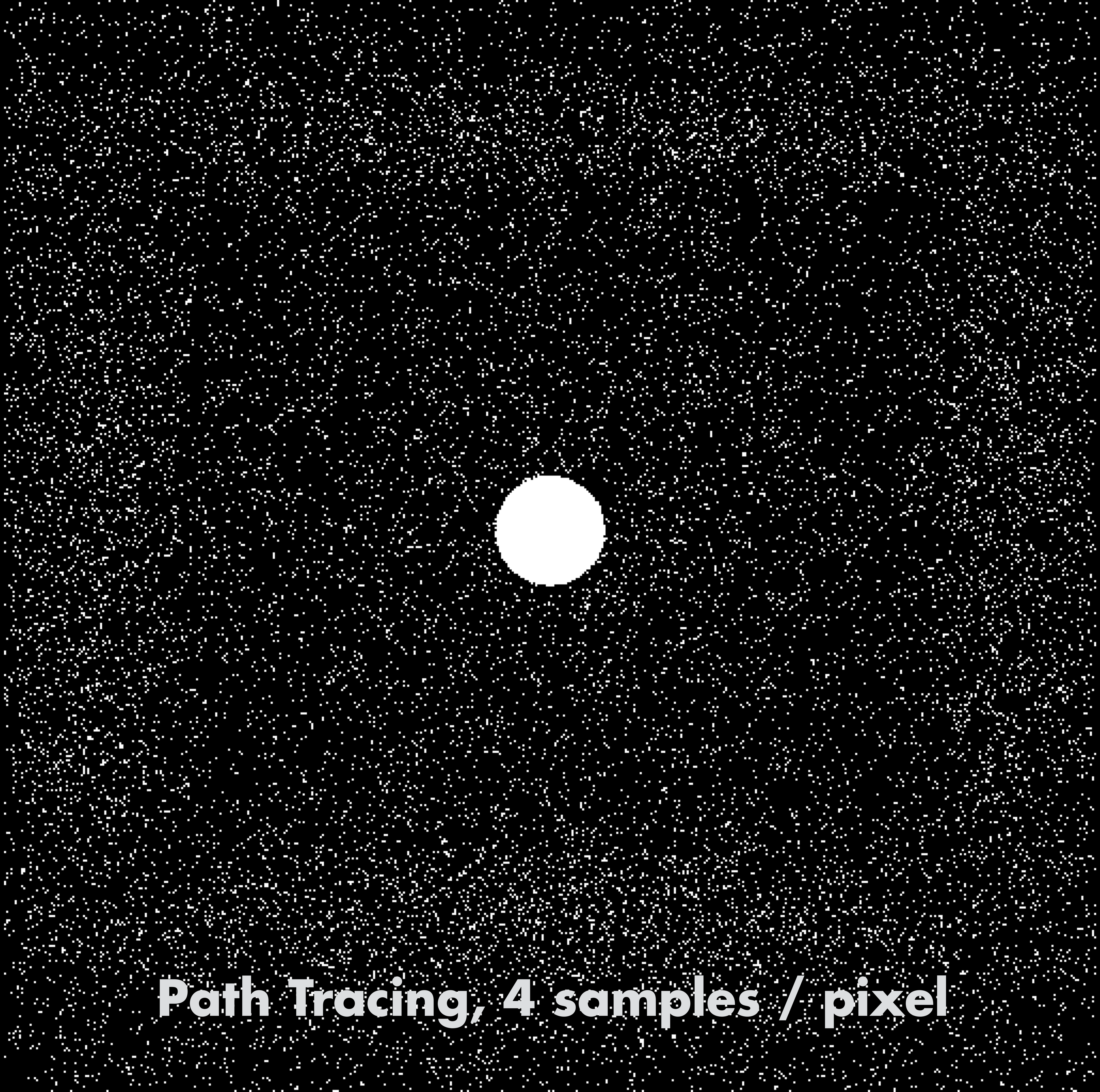
Provides a family of approaches for generating paths of length n

- **1 camera, n-1 light**
- **2 camera, n-2 light**
- **...**

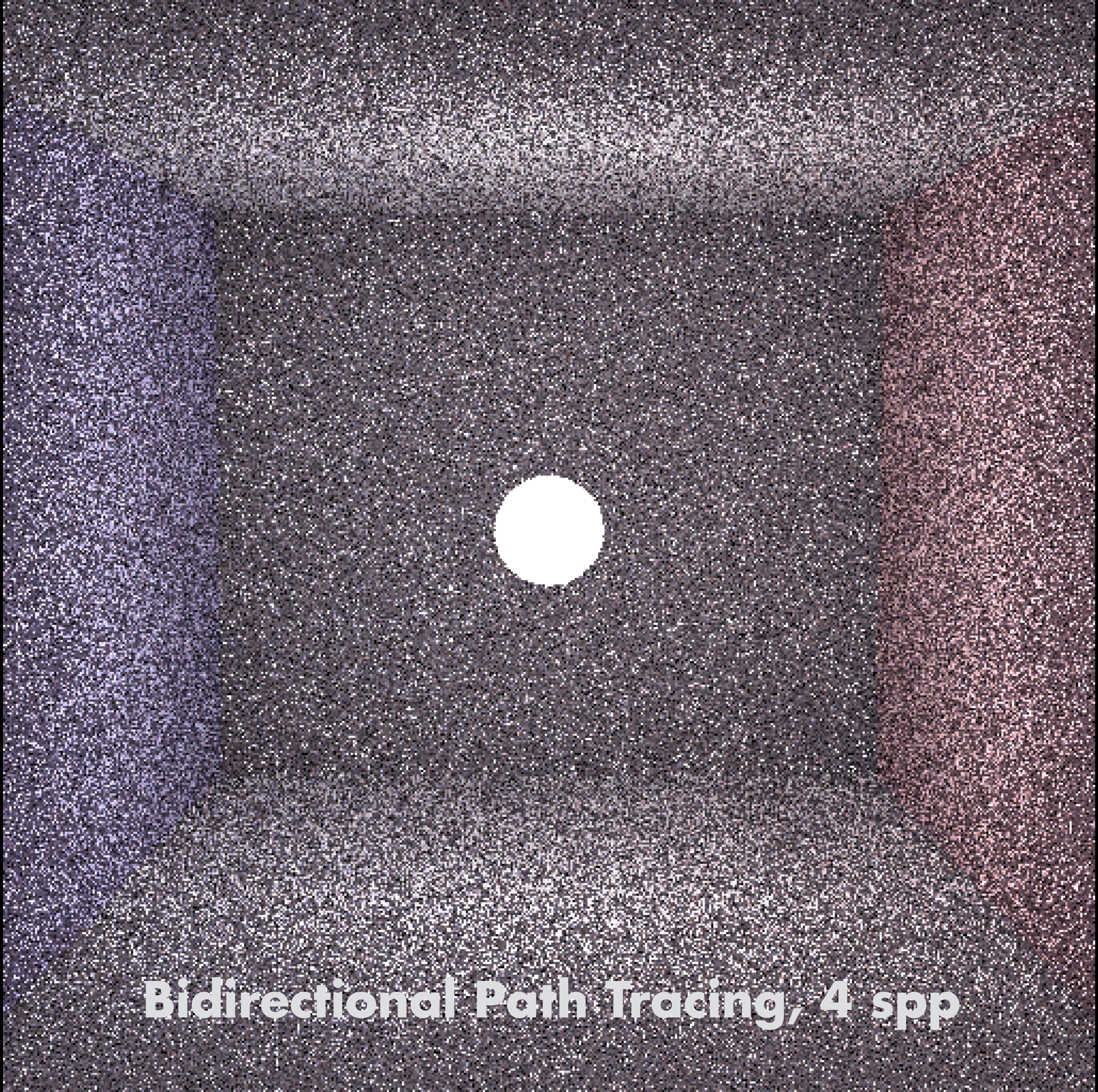
Key: apply multiple importance sampling to reweighs path contributions

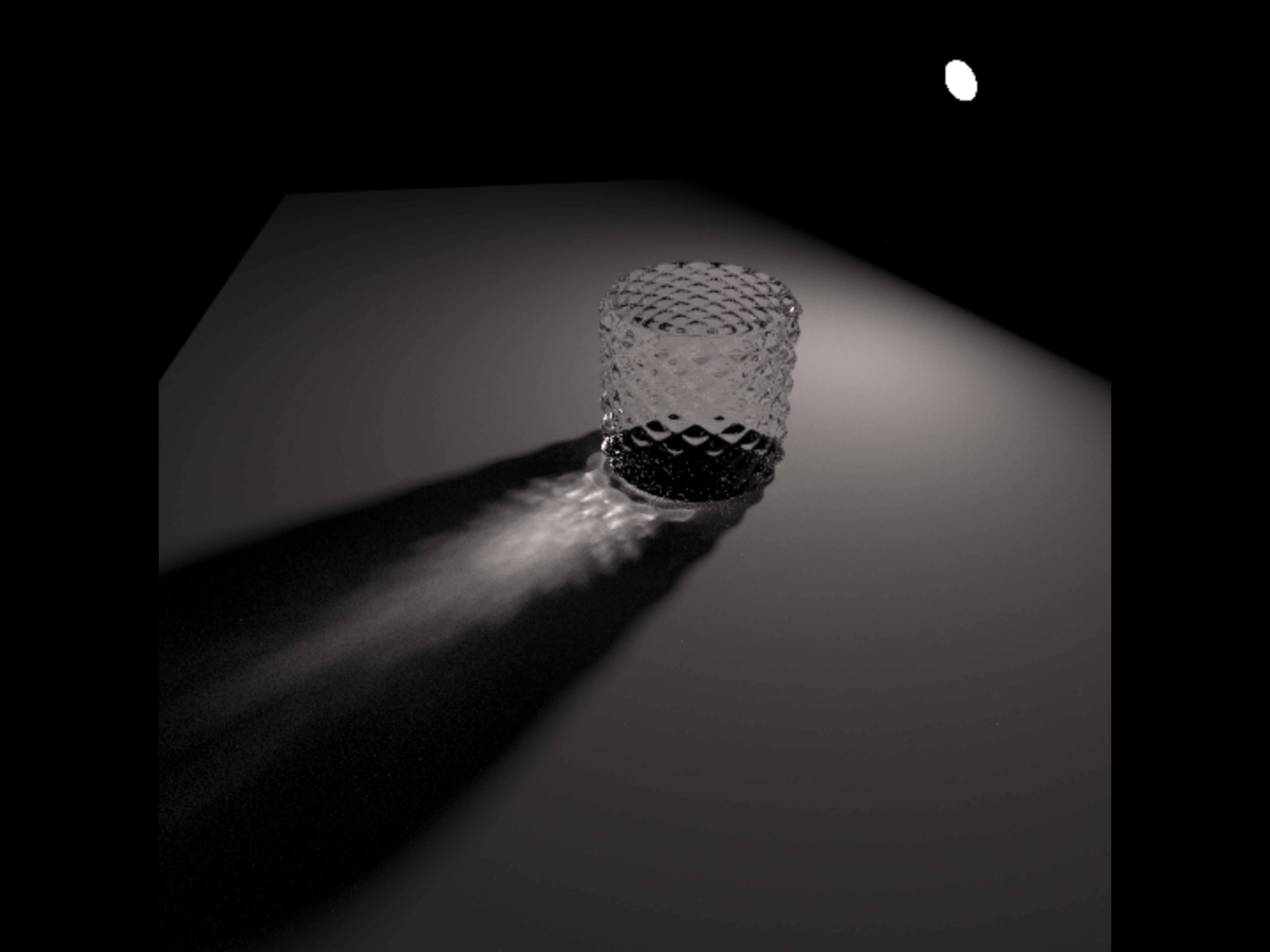


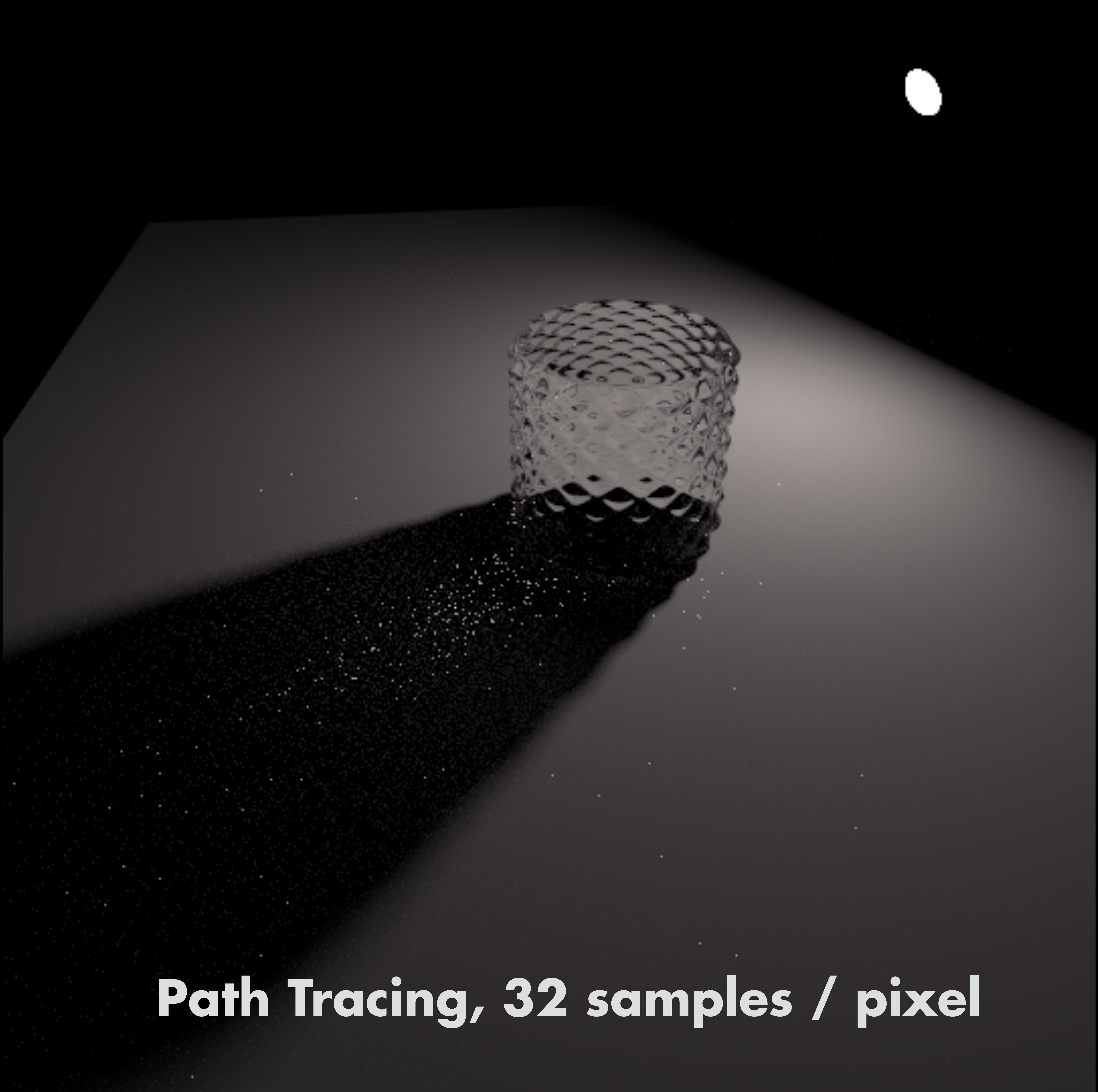
Glass around sphere



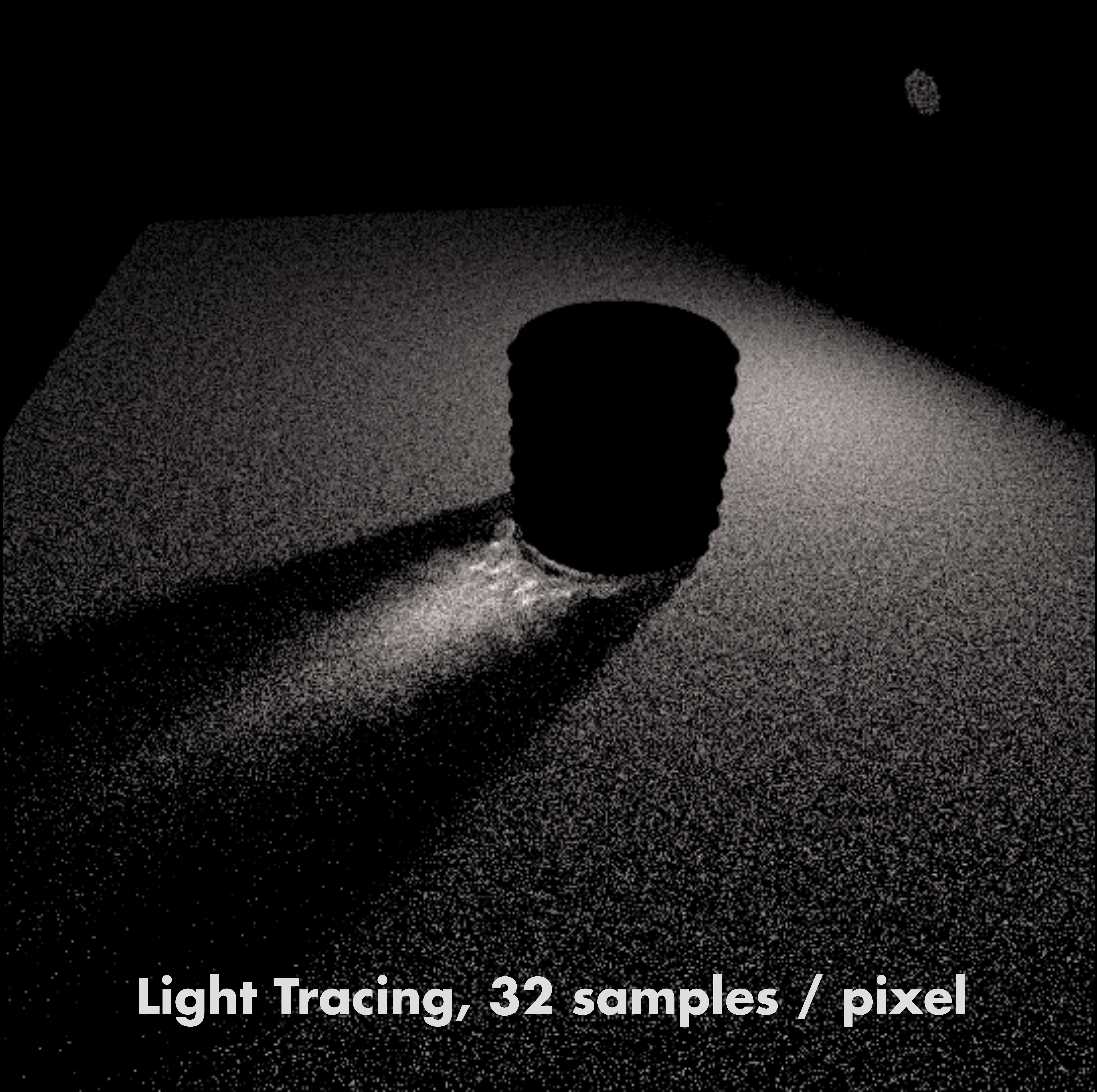
Path Tracing, 4 samples / pixel


Bidirectional Path Tracing, 4 spp

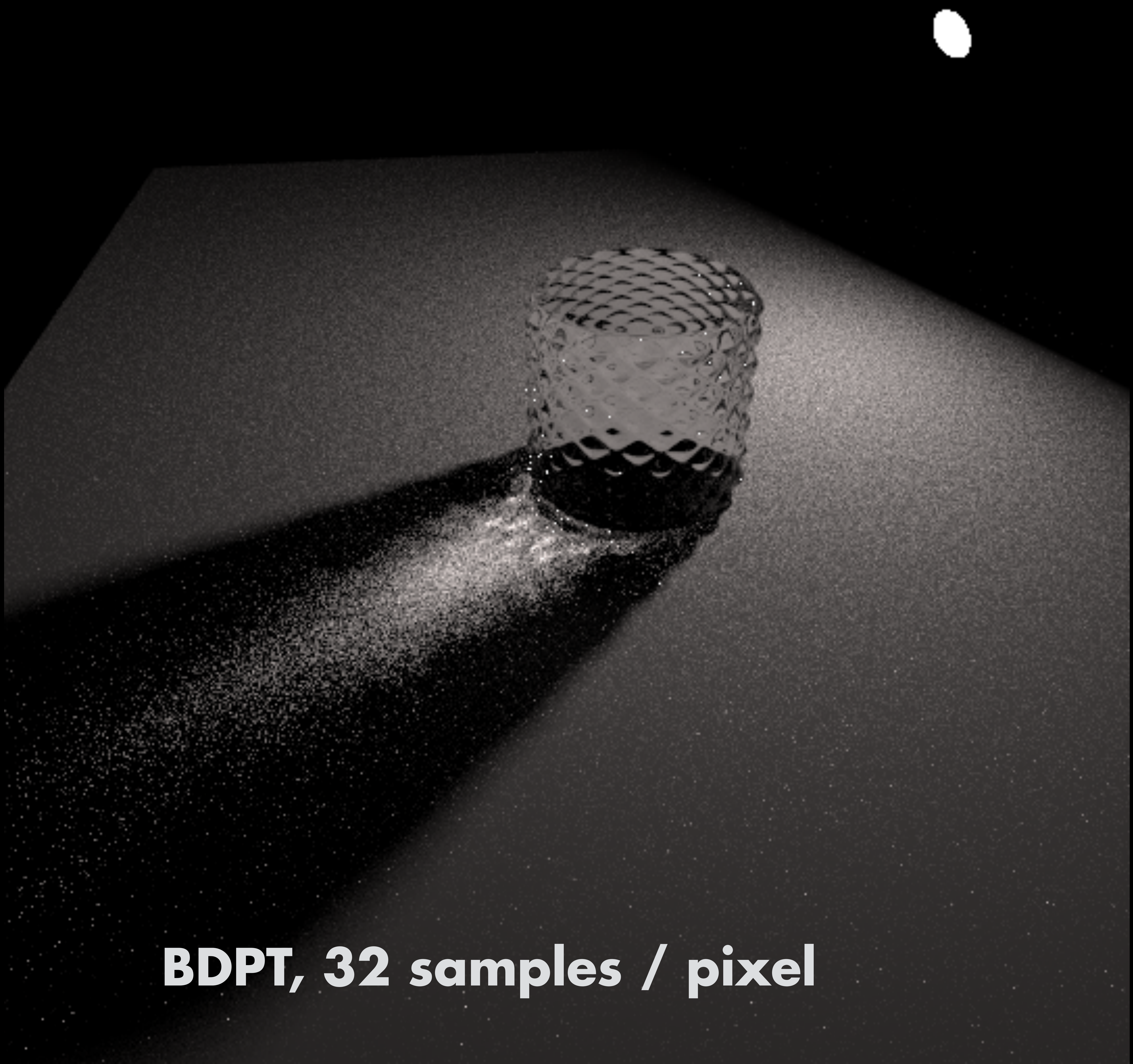




Path Tracing, 32 samples / pixel



Light Tracing, 32 samples / pixel


BDPT, 32 samples / pixel

Bidirectional Path Tracing

Path Tracing

[Veach and Guibas 1995]

Path Pyramid

light ⟶ ⟵ eye

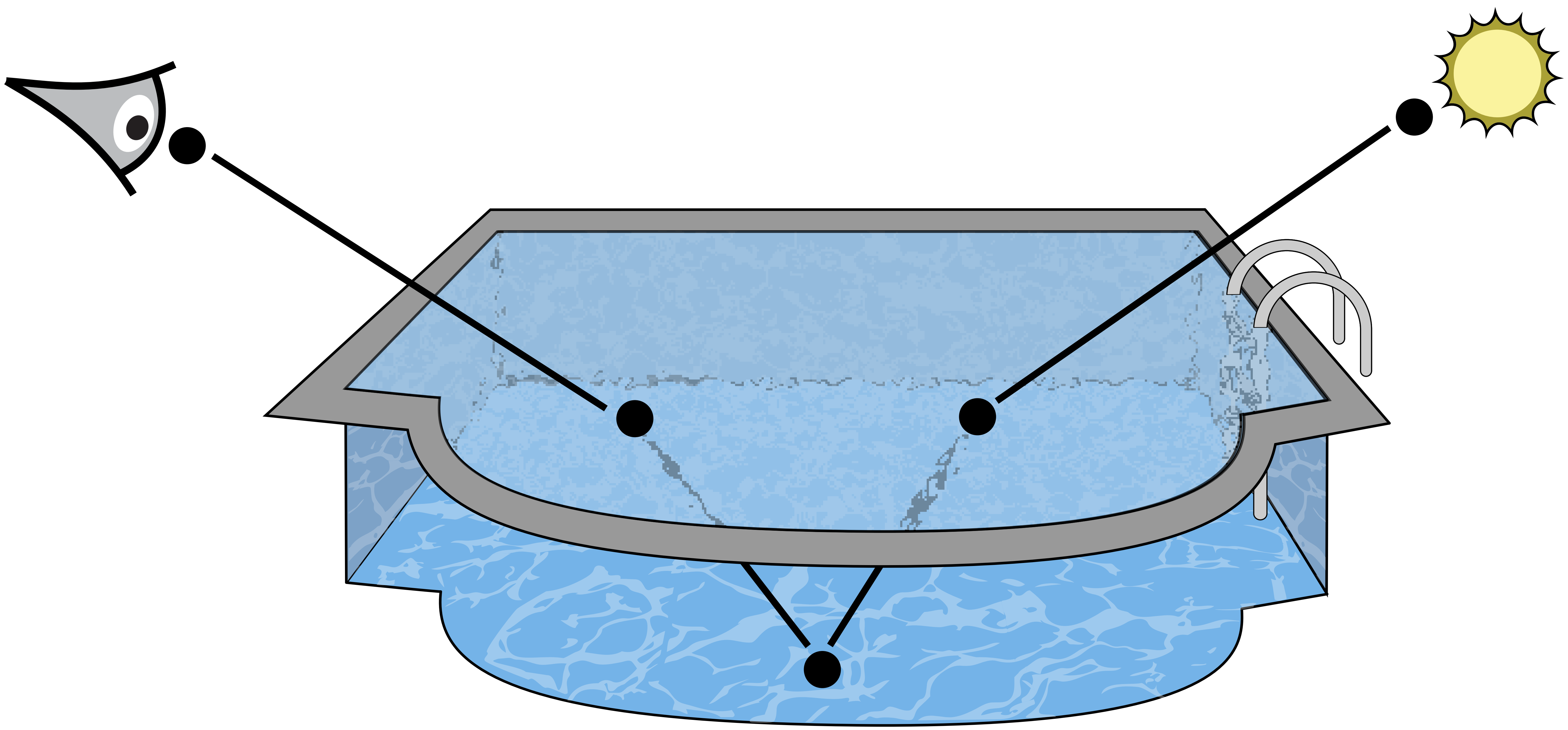
Challenges Remain

Challenges Remain

Reference **Bidirectional PT**

Wojciech Jarosz

Challenges Remain



Wojciech Jarosz

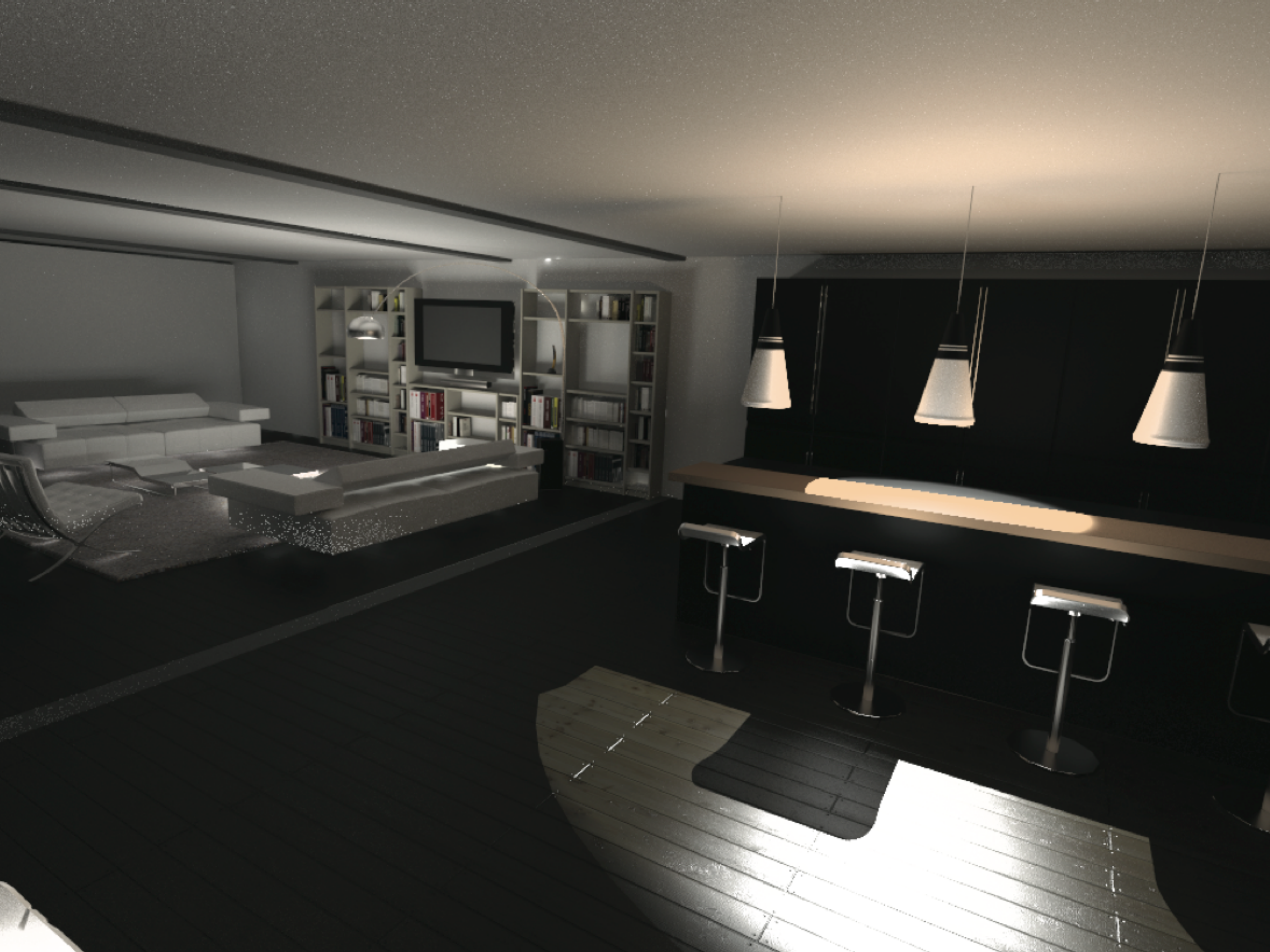
Photon Mapping

Trace particles from lights

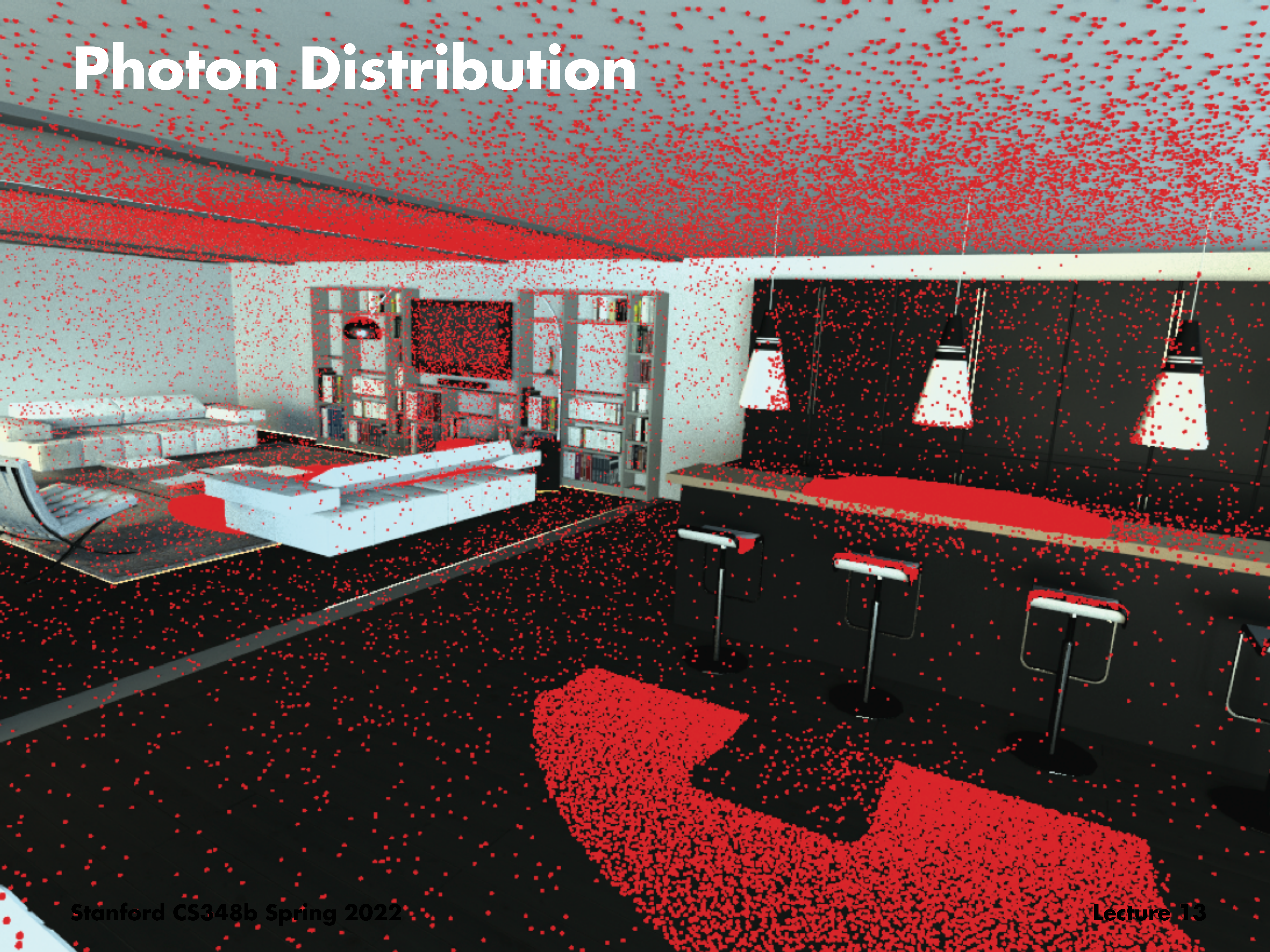
Interpolate *nearby* samples when computing reflected radiance

Key ideas:

- **Particle histories give scene radiance distribution**
- **Illumination information close to the point being shaded is generally applicable**



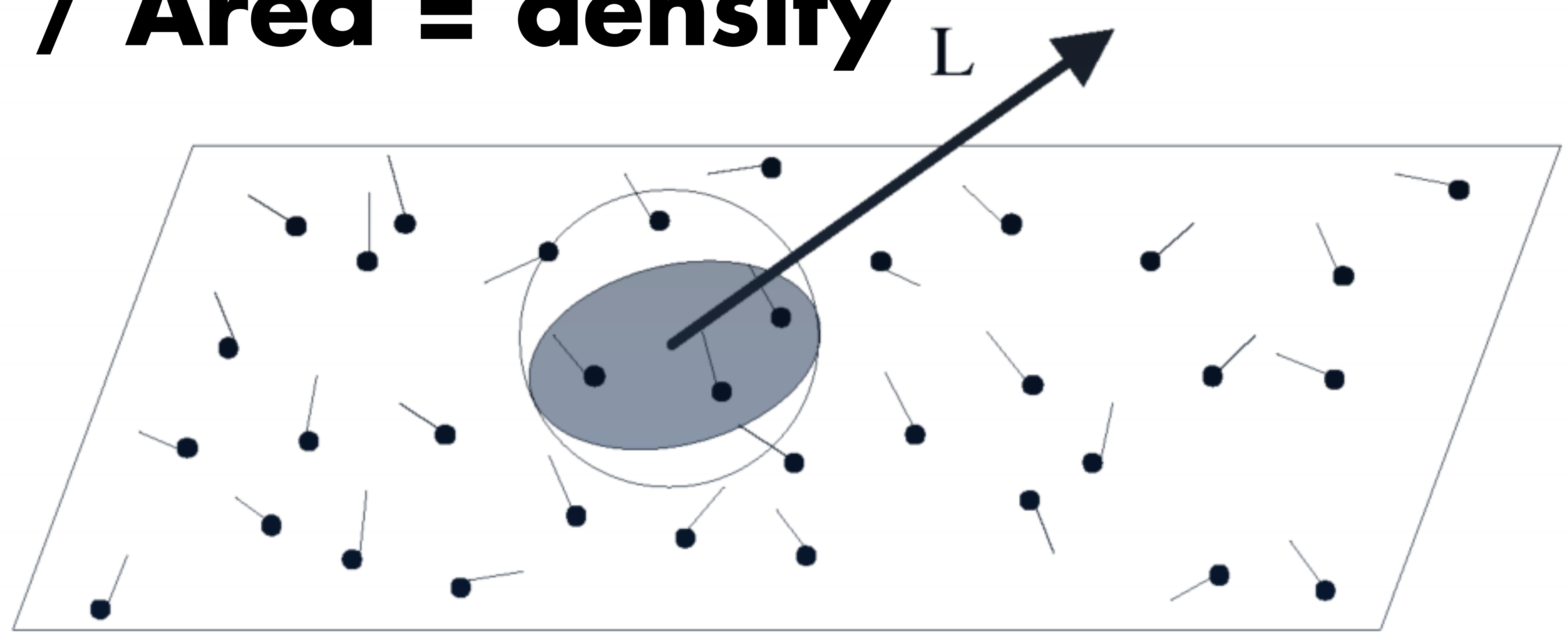
Photon Distribution



Radiance Estimate

Density estimate:

- Take a small spherical neighborhood containing k photons
- Count the power of the photons
- Compute the enclosed surface area
- Power / Area = density



Radiance Estimate

$$
\begin{aligned}
L(\mathrm{p},\omega_o) &= \int_{H^2} f_r(\omega_i \to \omega_o)\, L_i(p,\omega_i)\, \cos\theta_i \mathrm{d}\omega_i \\
&= \int_{H^2} f_r(\omega_i \to \omega_o)\, \frac{\mathrm{d}^2\Phi(\mathrm{p},\omega_i)}{\mathrm{d}\omega_i\, \cos\theta_i\, \mathrm{d}A}\, \cos\theta_i\, \mathrm{d}\omega_i \\
&= \int_{H^2} f_r(\omega_i \to \omega_o)\, \frac{\mathrm{d}^2\Phi(\mathrm{p},\omega_i)}{\mathrm{d}A} \\
&\approx \sum_i^N f_r(\omega[i] \to \omega_o) \frac{\Delta\Phi(\mathrm{p},\omega[i])}{\pi r^2}
\end{aligned}
$$

Unbiased vs. Consistent Estimators

Unbiased: $$E[X] = \int \cdots$$

- **Example:** $$\frac{1}{N}\sum_{i}^{N} f(x_i)$$

Consistent: $$\lim_{N\to\infty} E[X] = \int \cdots$$

- **Example:** $$\frac{1}{N+1}\sum_{i}^{N} f(x_i)$$

Unbiased vs. Consistent Estimators

Graphics interpretation:

- Consistent: image approaches correct solution as some parameter is increased
- Unbiased: produces correct answer on average

Value of biased estimators:

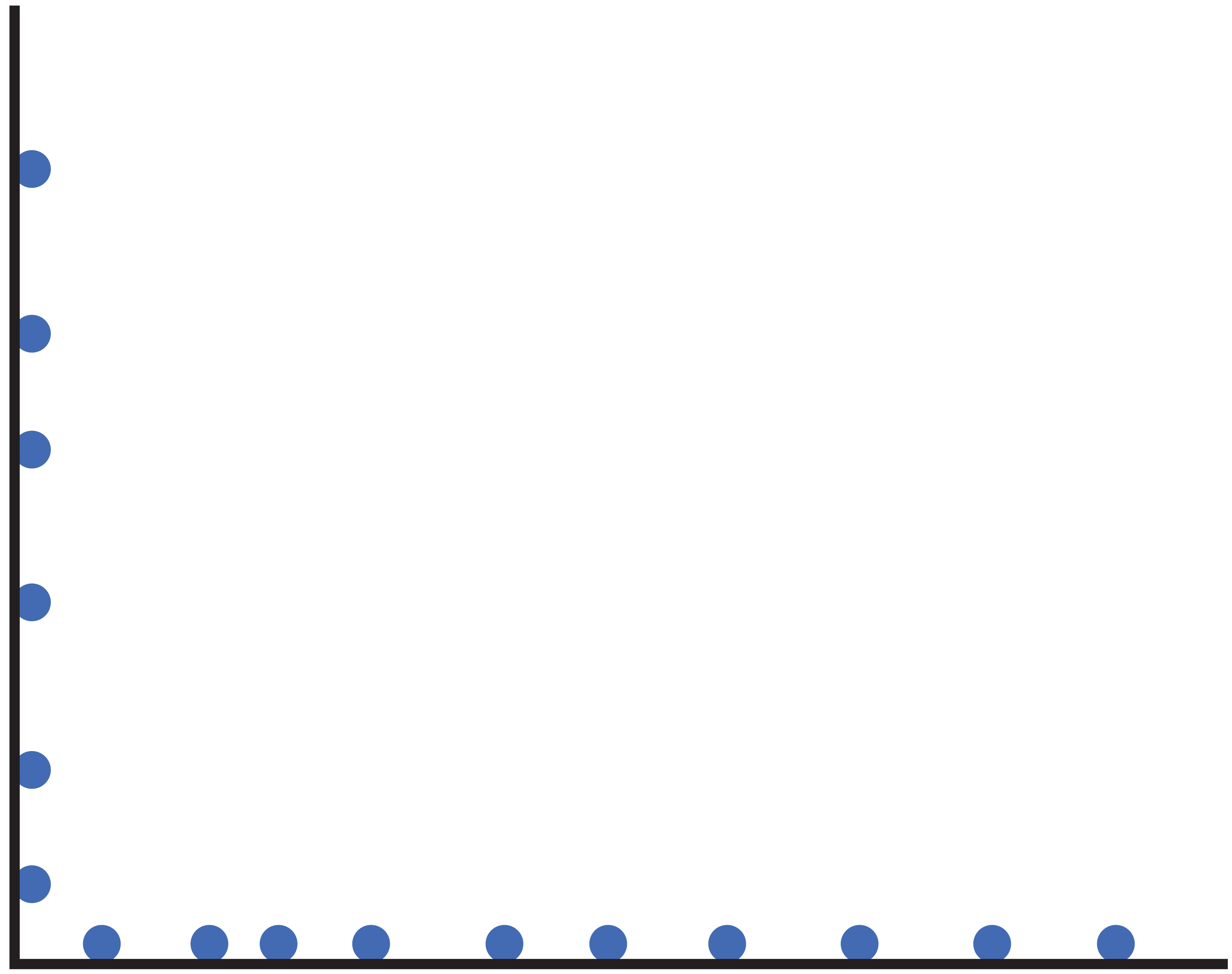
- May have lower variance
- May look better (less noise)

Radiance Estimate

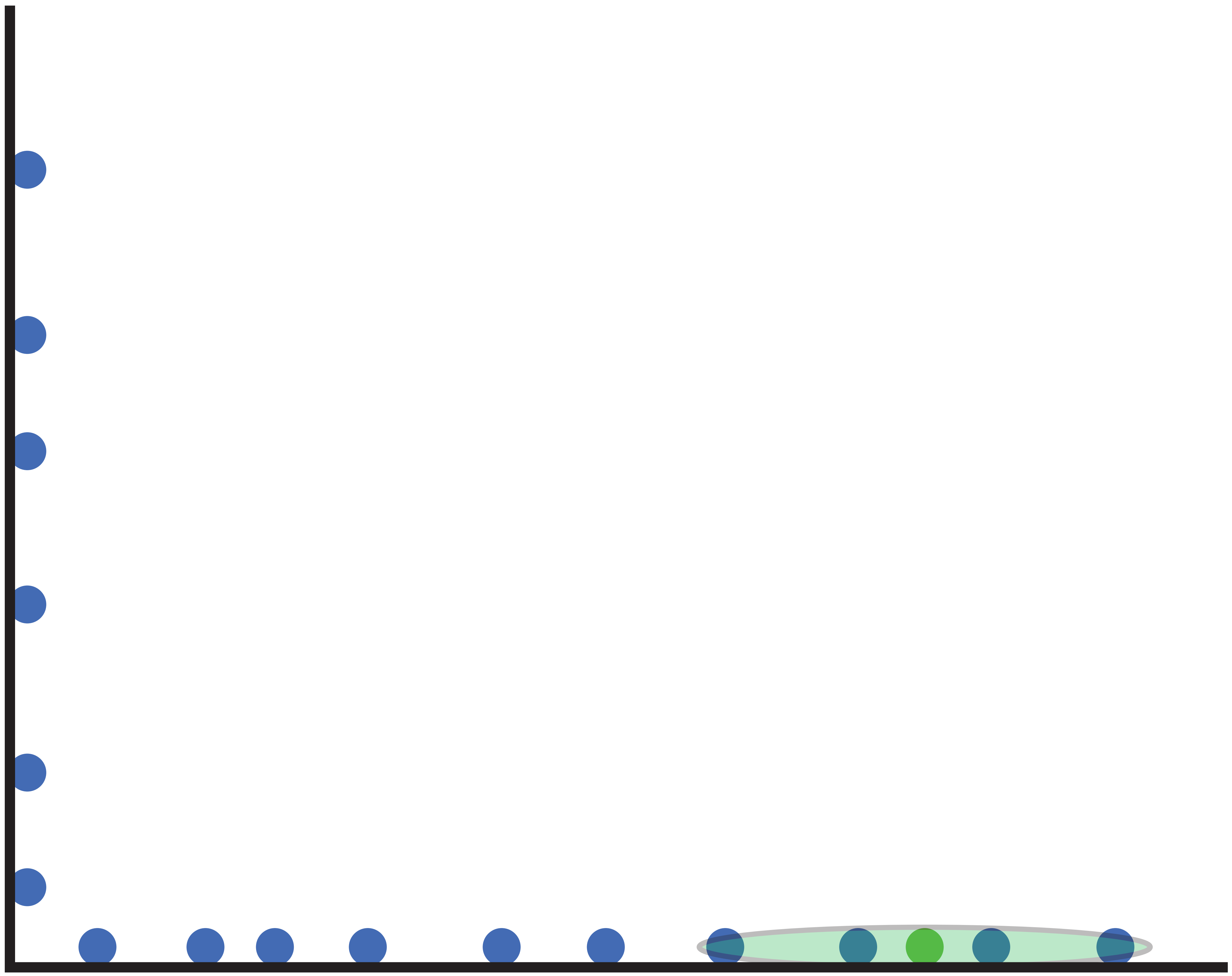
$$
\begin{aligned}
L(\mathrm{p}, \omega_o) &= \int_{H^2} f_r(\omega_i \to \omega_o)\, L_i(p, \omega_i)\ \cos\theta_i \mathrm{d}\omega_i \\
&= \int_{H^2} f_r(\omega_i \to \omega_o)\, \frac{\mathrm{d}^2\Phi(\mathrm{p}, \omega_i)}{\mathrm{d}\omega_i\ \cos\theta_i\, \mathrm{d}A}\ \cos\theta_i\, \mathrm{d}\omega_i \\
&= \int_{H^2} f_r(\omega_i \to \omega_o)\, \frac{\mathrm{d}^2\Phi(\mathrm{p}, \omega_i)}{\mathrm{d}A} \\
&\approx \sum_i^N f_r(\omega[i] \to \omega_o) \frac{\Delta\Phi(\mathrm{p}, \omega[i])}{\pi r^2}
\end{aligned}
$$

Biased, but consistent

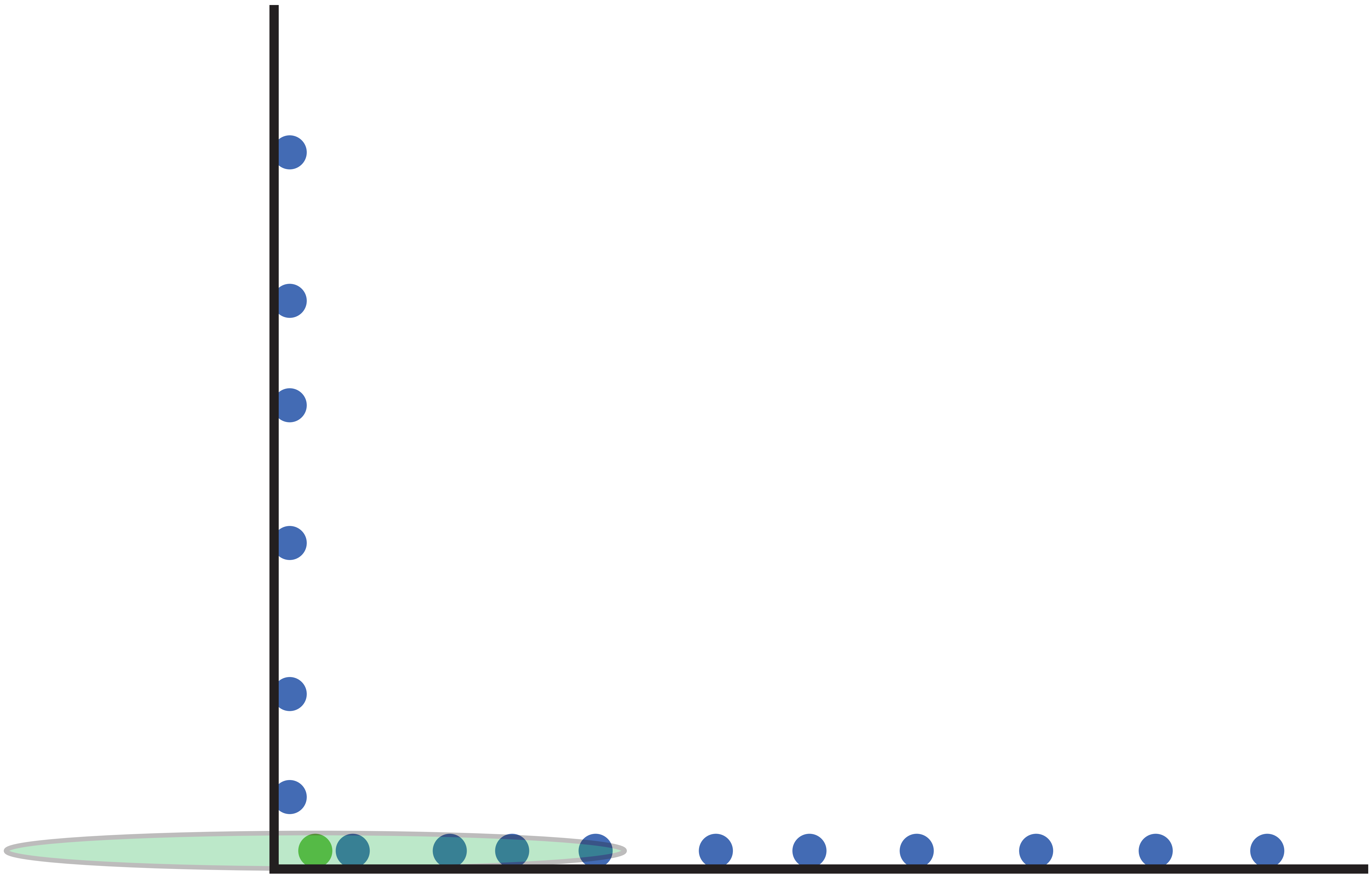
Density Estimate Challenges



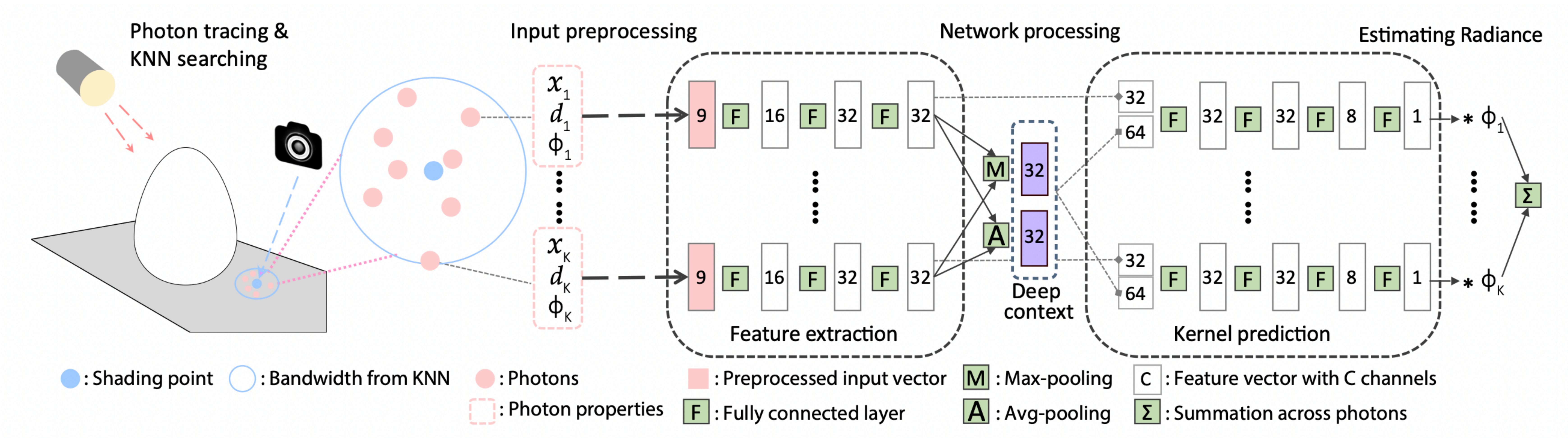
Density Estimate Challenges



Density Estimate Challenges



Density Estimate Challenges



Zhu et al., 2020. *Deep Kernel Destiny Estimation for Photon Mapping.*

Standard Photon Mapping

- Store all photons in memory
- Trace path from camera, lookup photons at each path vertex ("pull")
- Memory required $\propto$ # photons

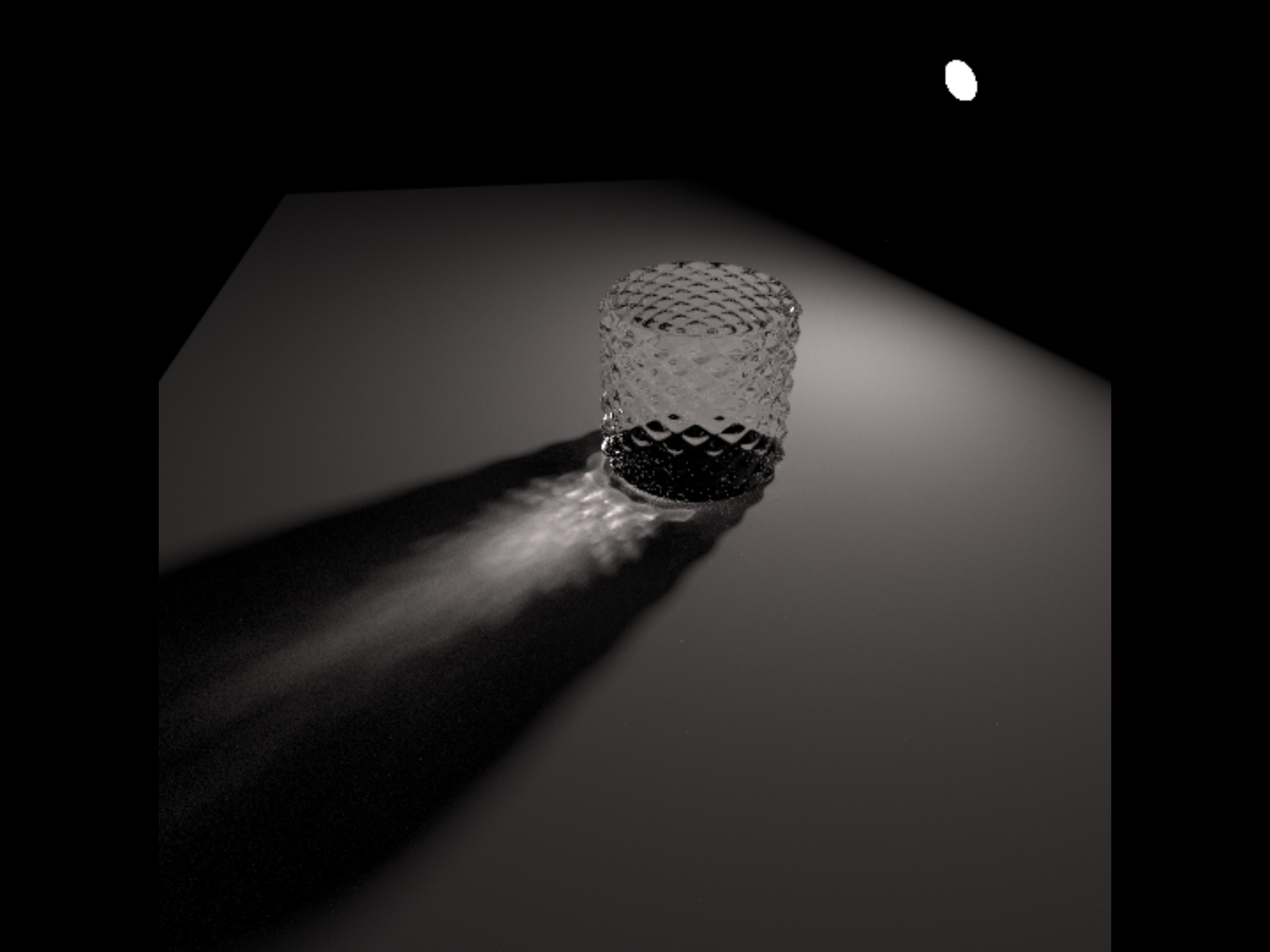
Progressive Photon Mapping

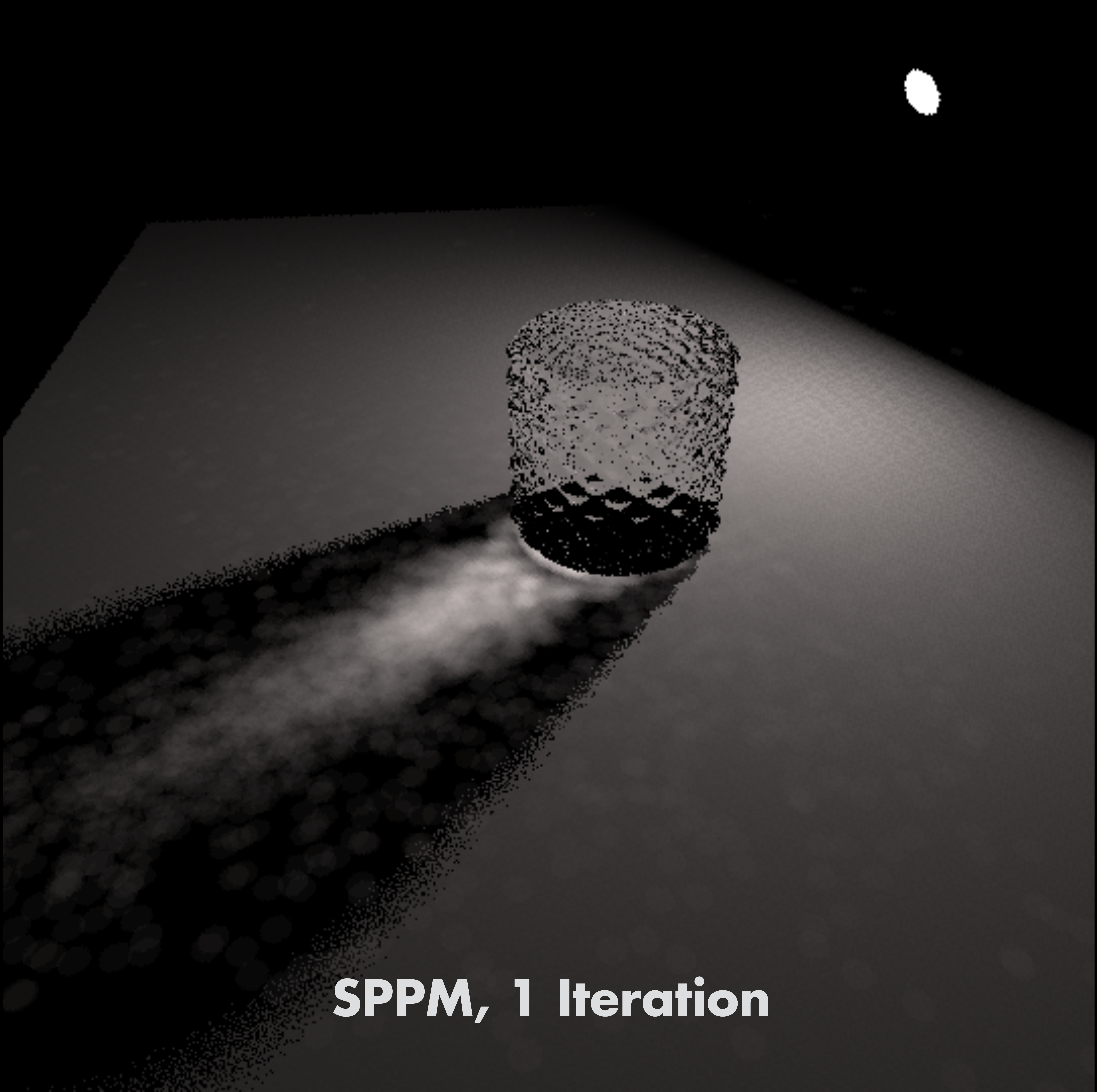
- Store visible points in memory
- Trace one photon at a time, splat to relevant points ("push")
- Memory required $\propto$ # visible points

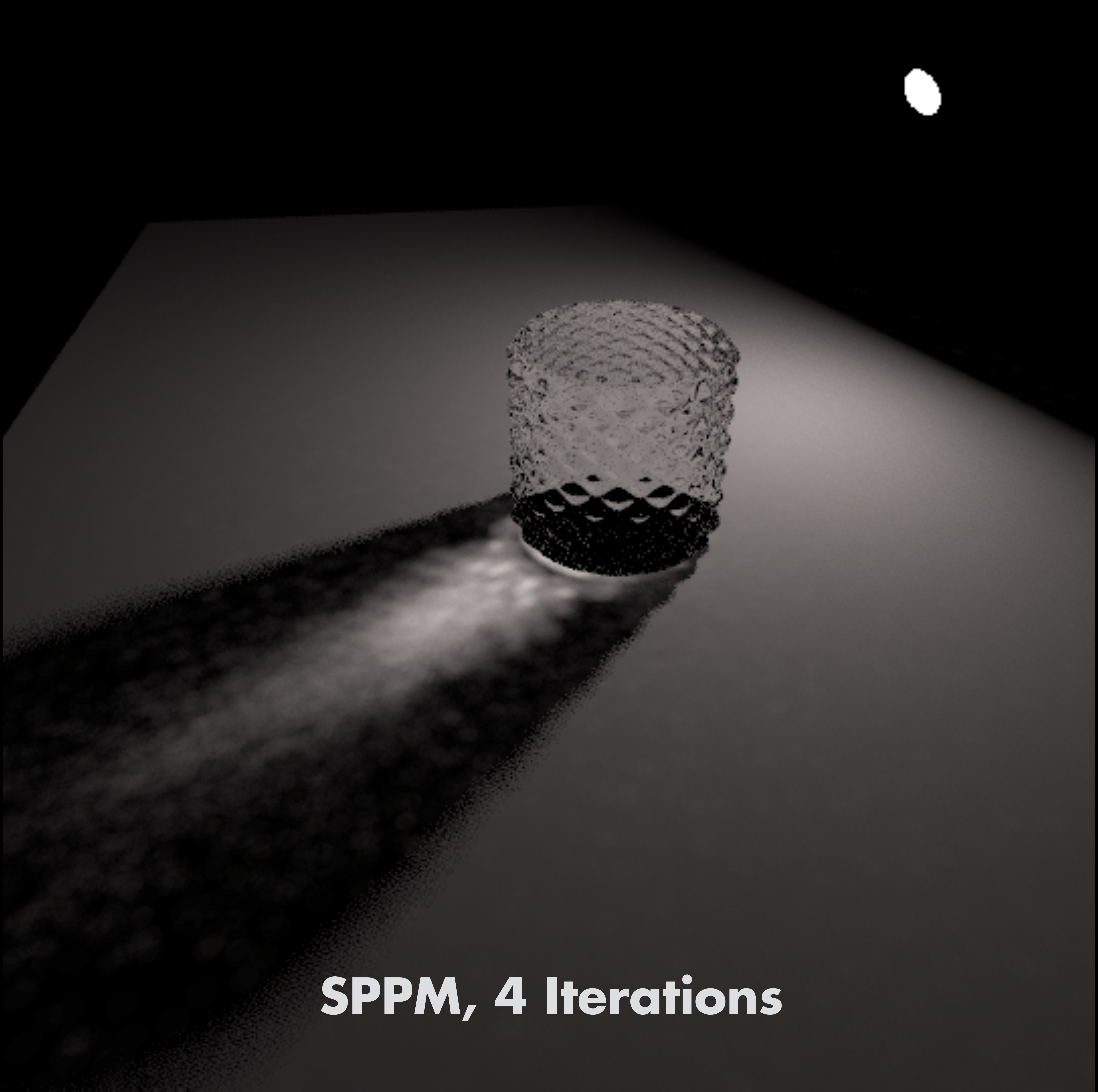
Progressive Photon Mapping

1. Trace rays from camera, find a VisiblePoint at each pixel
2. Store VisiblePoints in a 3D spatial structure that allows fast lookups
3. Trace photons from lights; at intersections search for nearby VisiblePoints and contribute illumination
4. Decrease acceptance radius as more photons contribute

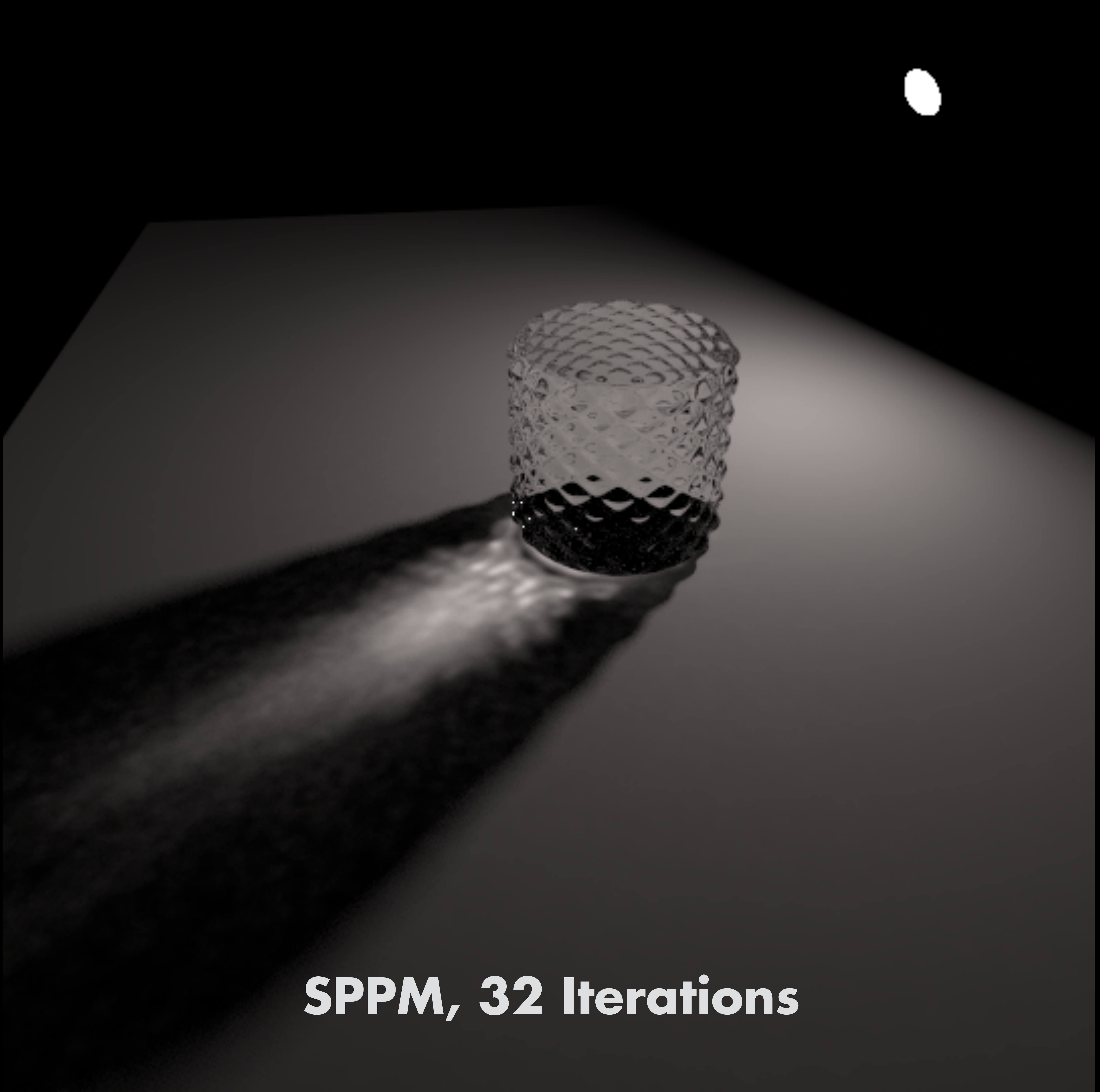
```
struct VisiblePoint {
    Point3f p;
    Vector3f wo;
    const BSDF *bsdf;
    Spectrum beta;
};
```



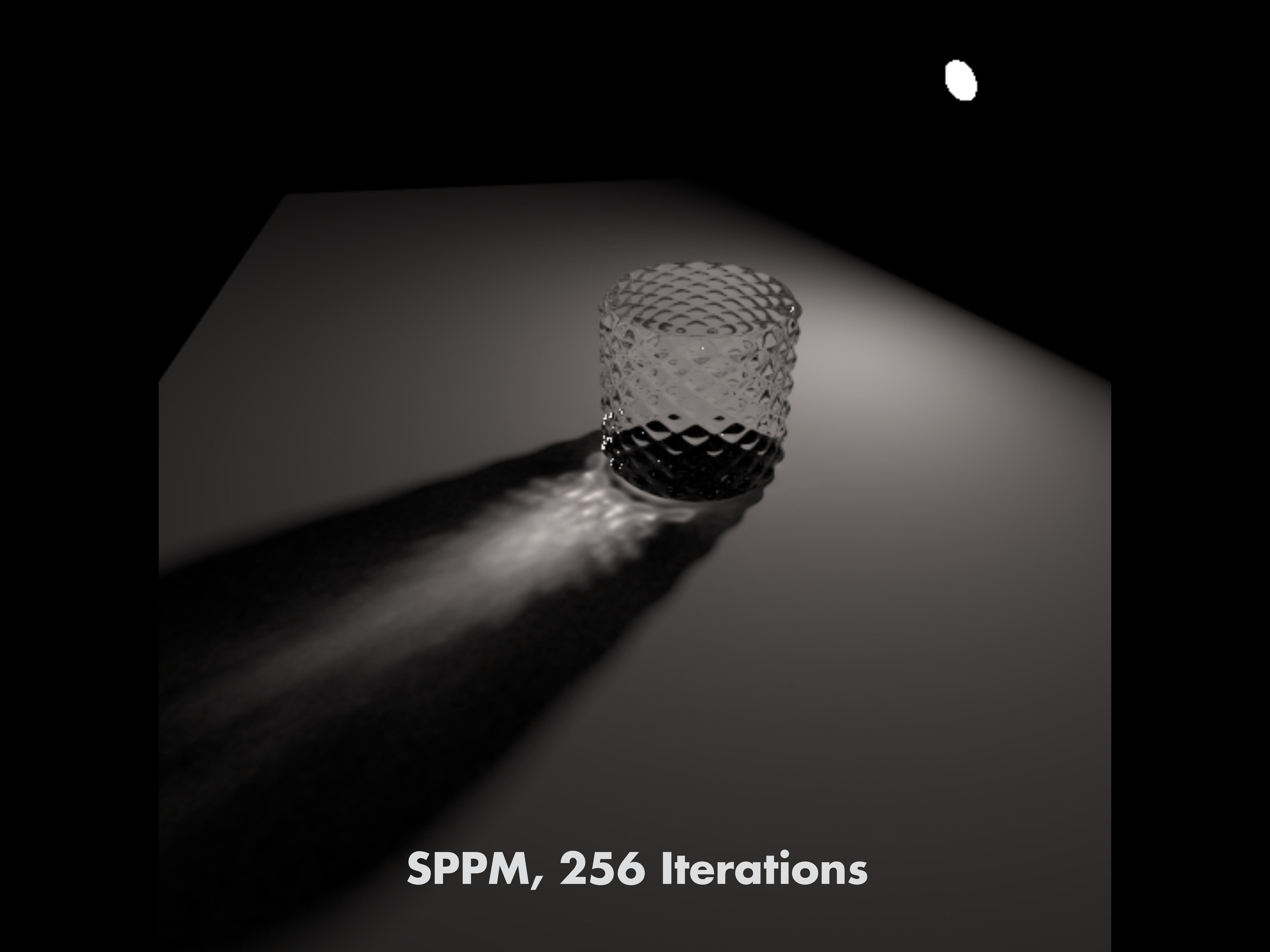

SPPM, 1 Iteration



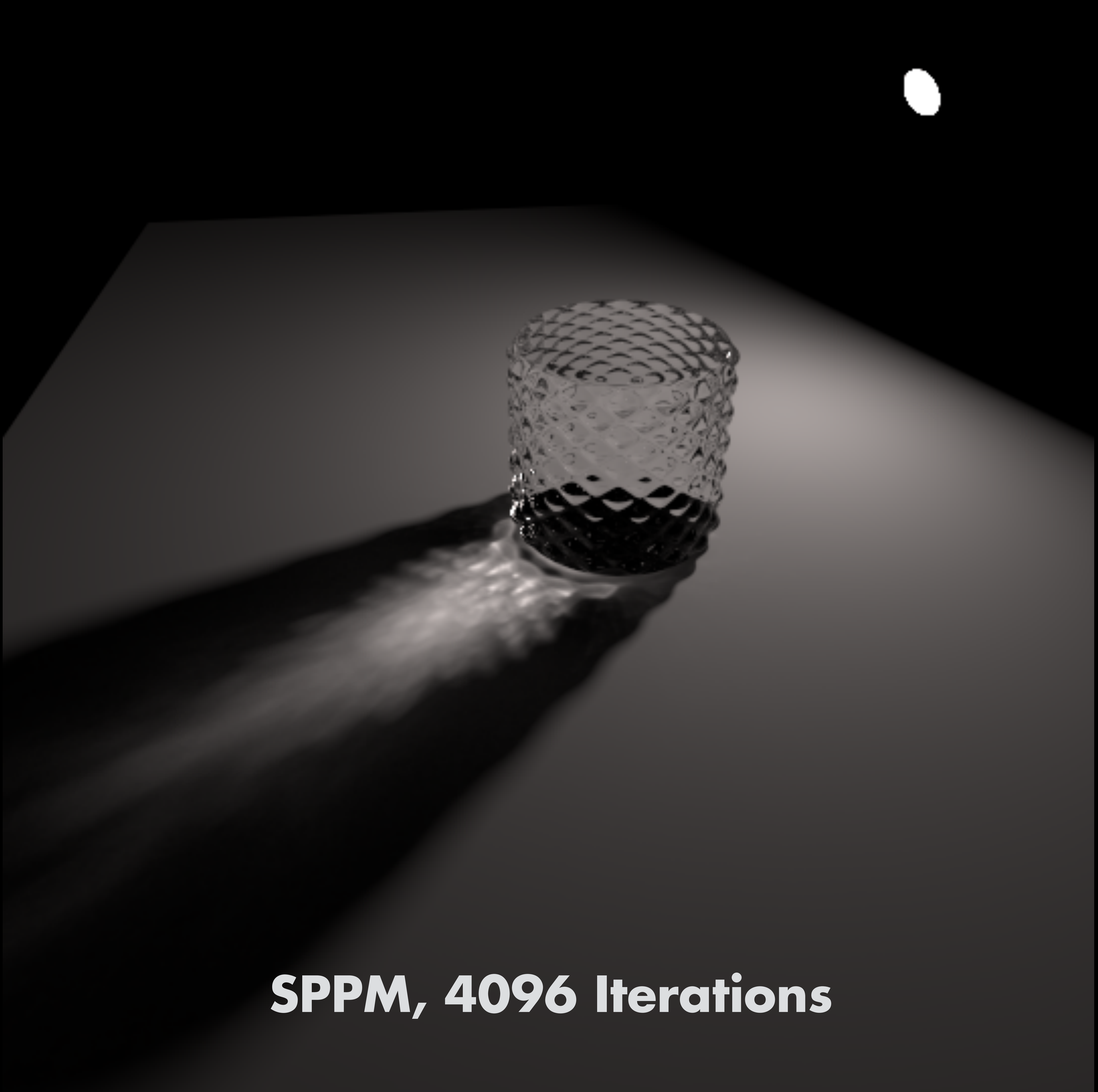
SPPM, 4 Iterations

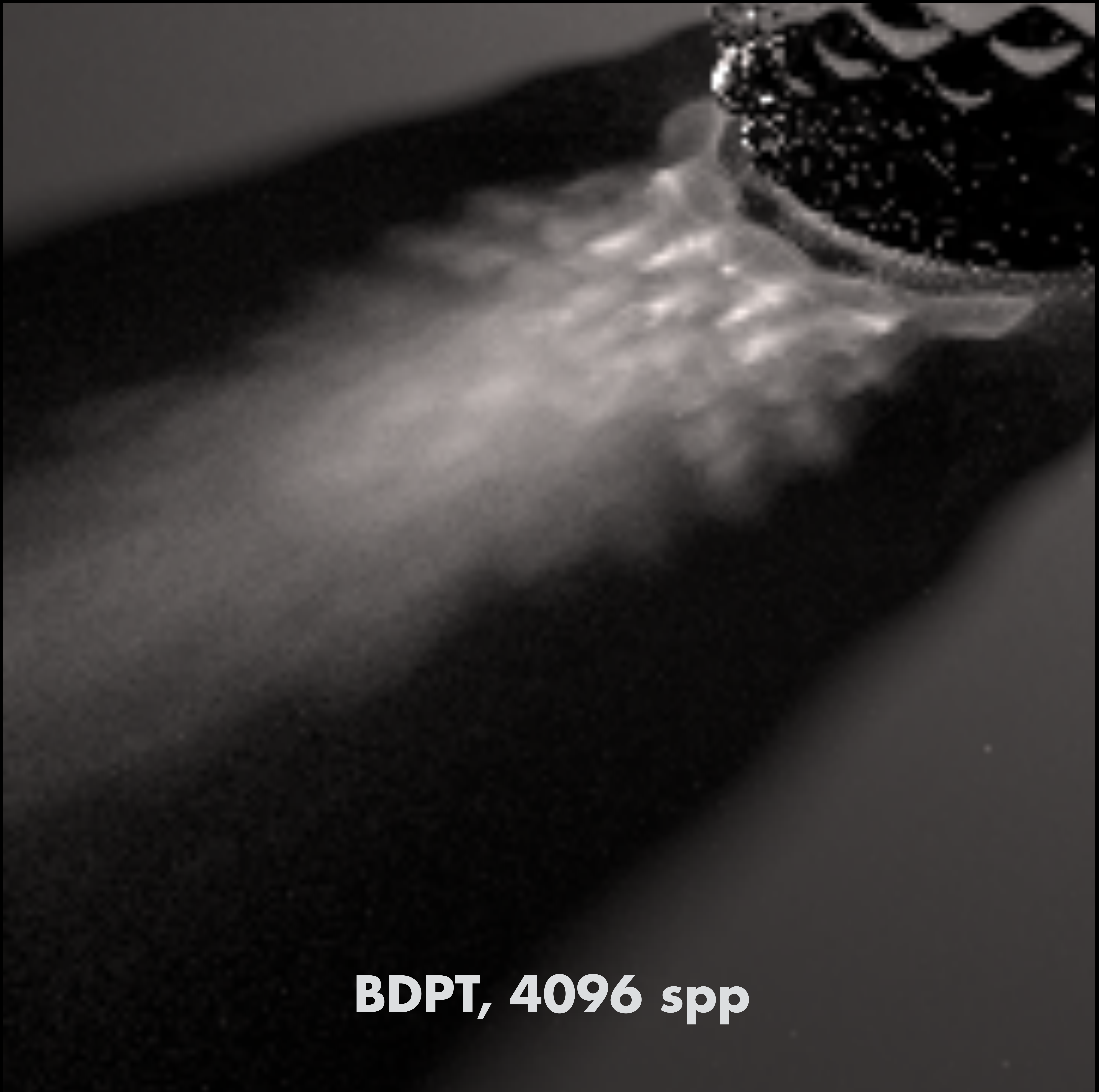


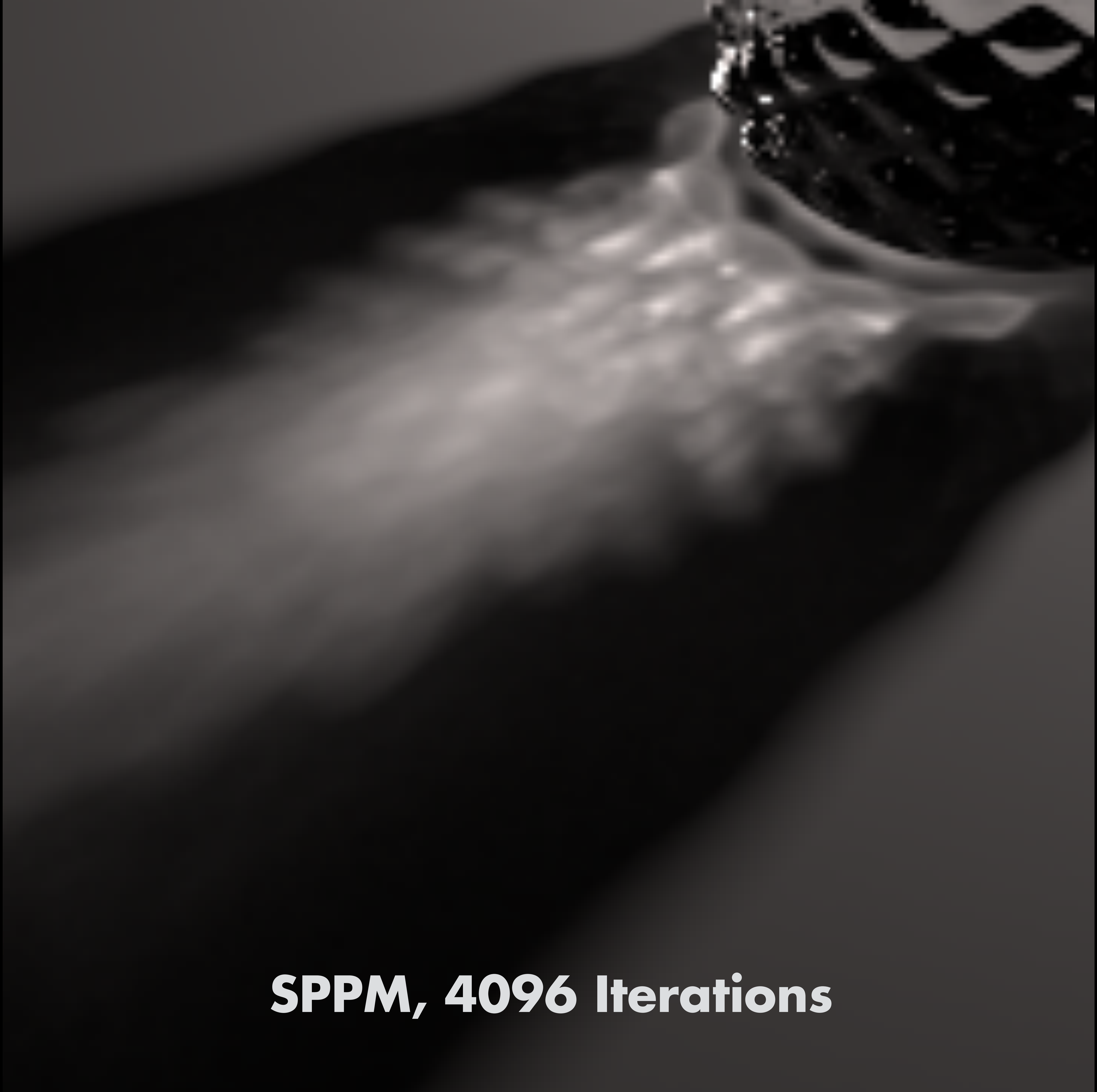
SPPM, 32 Iterations



SPPM, 256 Iterations


SPPM, 4096 Iterations


BDPT, 4096 spp



SPPM, 4096 Iterations