The Rendering Equation

Computer Graphics
CMU 15-462/15-662

Recap: Incident vs. Exitant Radiance

INCIDENT

EXITANT

In both cases: intensity of illumination is highly dependent on *direction* (not just location in space or moment in time).

Recap: Radiance and Irradiance

irradiance

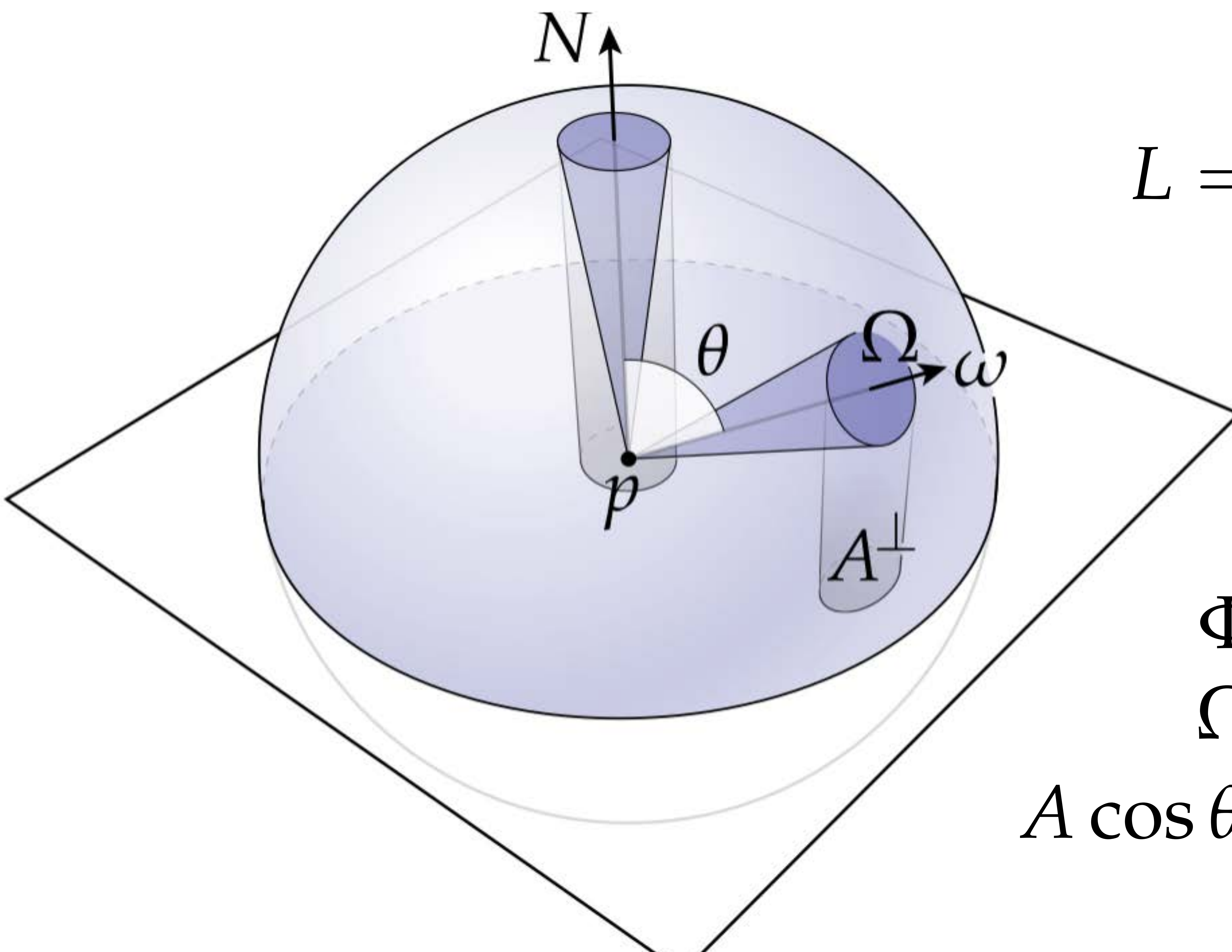
radiance in direction ω

$$E = \int_{H^2} L(\omega) \cos\theta \, d\omega$$

angle between ω and normal

Recap: What is *radiance?*

- **Radiance at point p in direction N is radiant energy ("#hits") per unit time, per solid angle, per unit area perpendicular to N.**



$$L = \frac{\partial^2 \Phi}{\partial \Omega \, \partial A \cos\theta}$$

Φ — **radiant flux**

Ω — **solid angle**

$A \cos\theta$ — **projected area***

***Confusing point: *this* cosine has to do w/ parameterization of sphere, not Lambert's law**

Aside: A Tale of Two Cosines

- **Confusing point first time you study photorealistic rendering: "cos θ" shows up for two completely unrelated reasons**

LAMBERT'S LAW

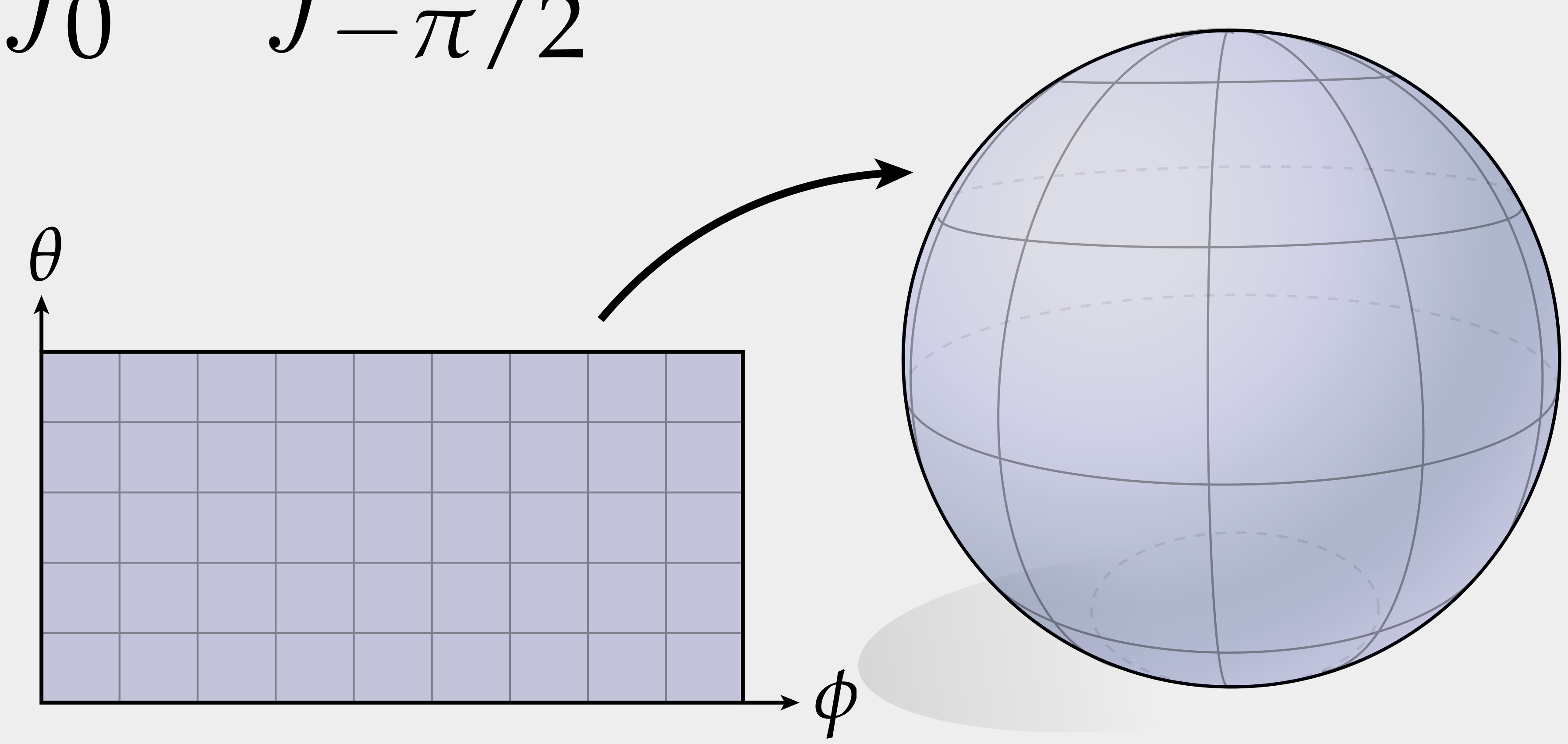
$$A = A' \cos\theta$$

SPHERICAL INTEGRALS

$$\int_{S^2} f\, dA$$

$$=$$

$$\int_0^{2\pi} \int_{-\pi/2}^{\pi/2} f(\theta,\phi)\ \cos\theta\, d\theta\, d\phi$$



Question du jour:

How do we use all this stuff to generate images?

The Rendering Equation

- **Core functionality of photorealistic renderer is to estimate radiance at a given point p, in a given direction ω_o**
- **Summed up by the *rendering equation* (Kajiya):**

$$L_o(\mathbf{p}, \omega_o) = L_e(\mathbf{p}, \omega_o) + \int_{\mathcal{H}^2} f_r(\mathbf{p}, \omega_i \rightarrow \omega_o) L_i(\mathbf{p}, \omega_i) \cos\theta \, d\omega_i$$

outgoing/observed radiance

emitted radiance (e.g., light source)

angle between incoming direction and normal

incoming radiance

point of interest

direction of interest

all directions in hemisphere

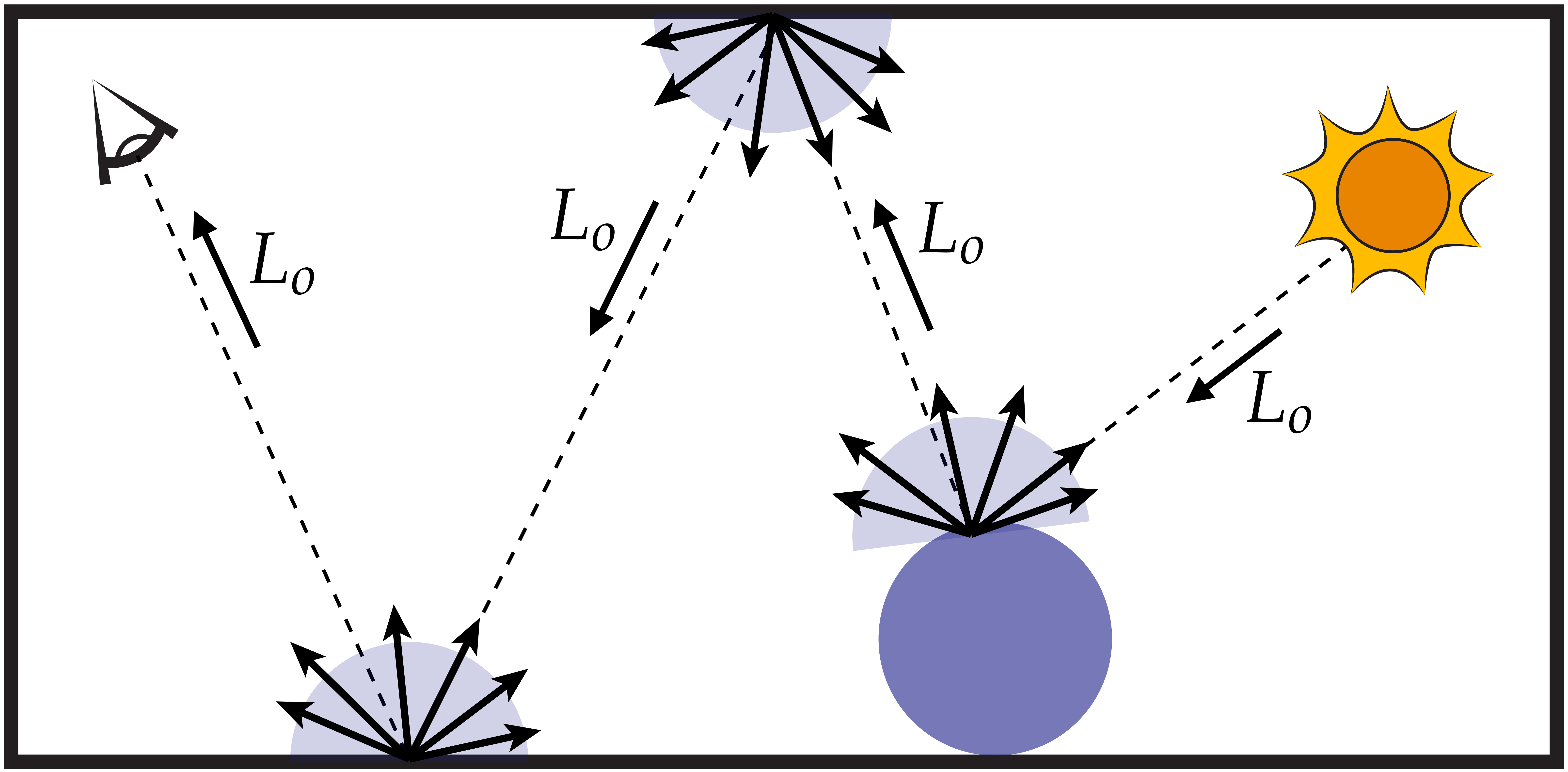
scattering function

incoming direction

Key challenge: to evaluate incoming radiance, we have to compute yet another integral. I.e., rendering equation is *recursive*.

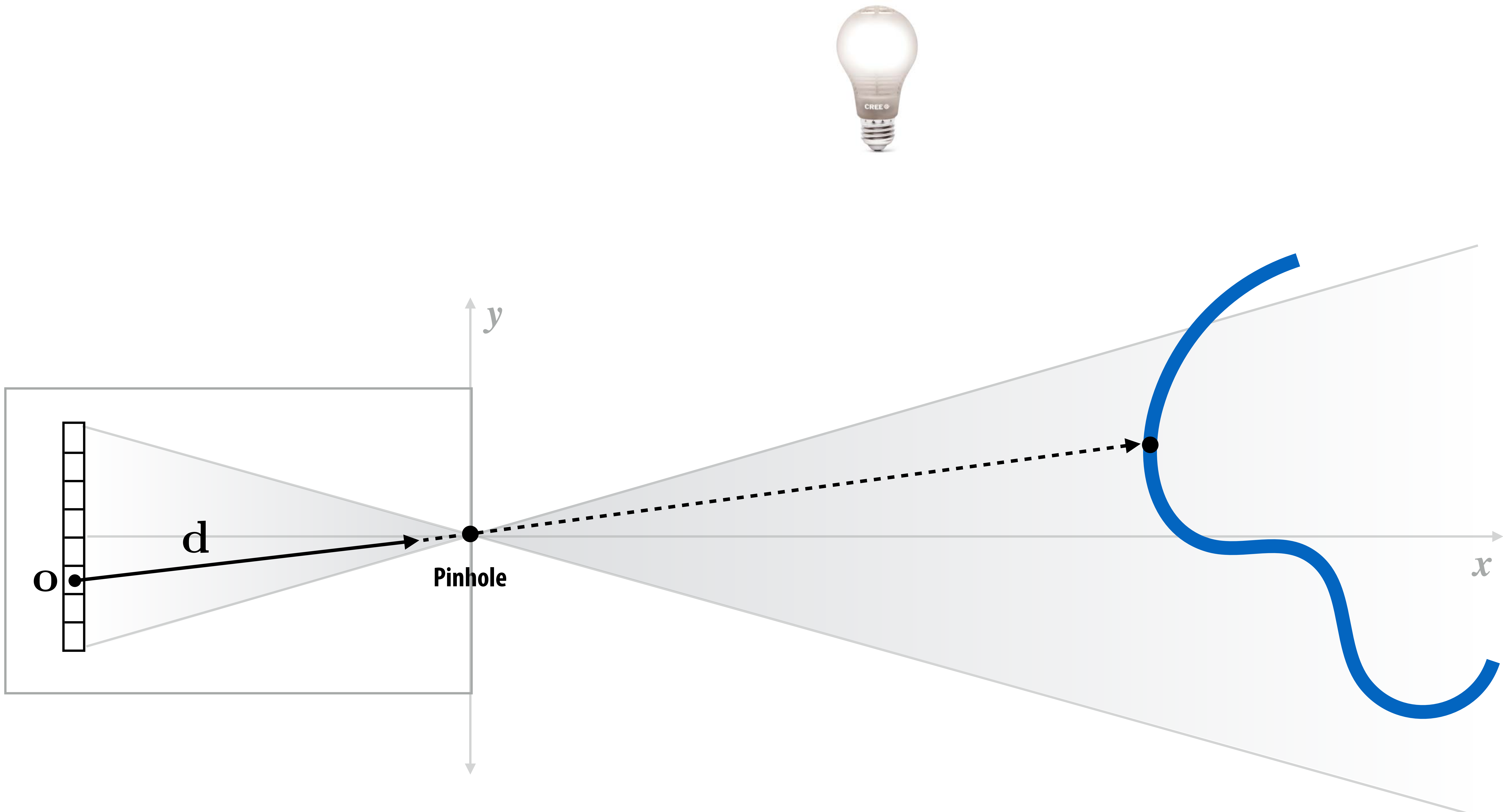
Recursive Raytracing

- **Basic strategy: recursively evaluate rendering equation!**



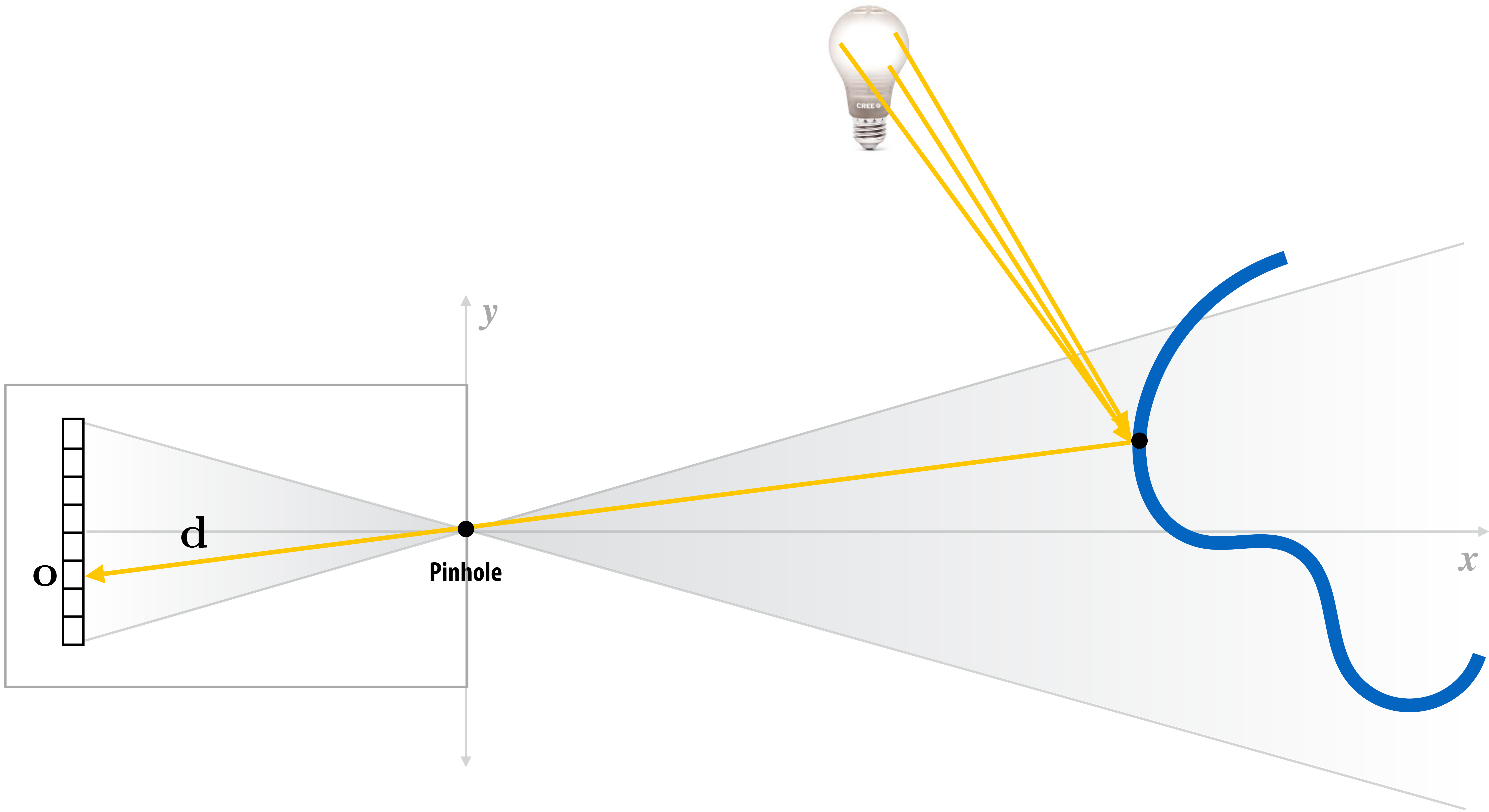
(*This* is why you're writing a ray tracer—rasterizer isn't enough!)

Renderer measures radiance along a ray

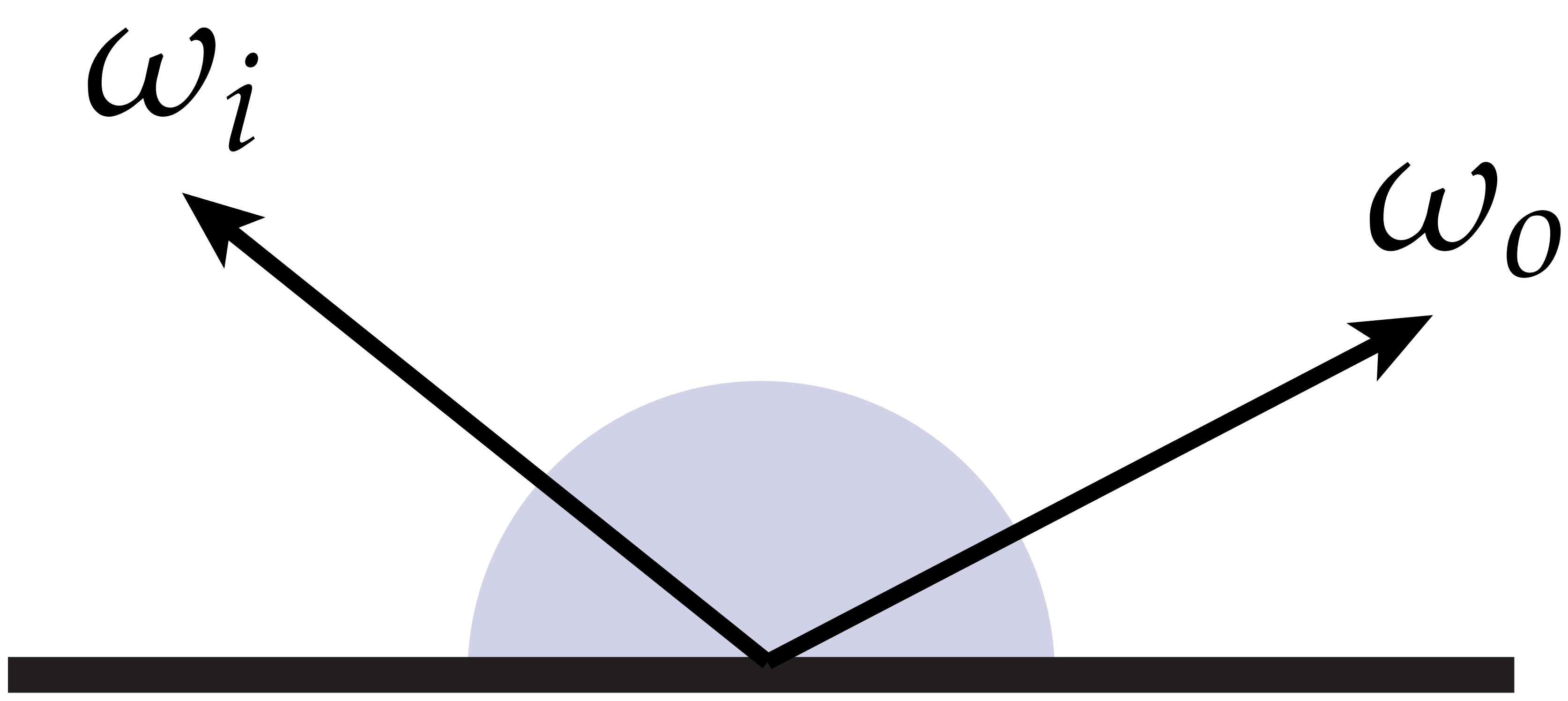


At each "bounce," want to measure radiance traveling in the direction opposite the ray direction.

Renderer measures radiance along a ray



Radiance entering camera in direction *d* = light from scene light sources that is reflected off surface in direction *d*.



How does *reflection* of light affect the outgoing radiance?

$$L_o(\mathbf{p},\omega_o) = \int_{\mathcal{H}^2} f_r(\mathbf{p},\omega_i,\omega_o)\, L_i(\mathbf{p},\omega_i) \cos\theta \,\mathrm{d}\,\omega_i$$

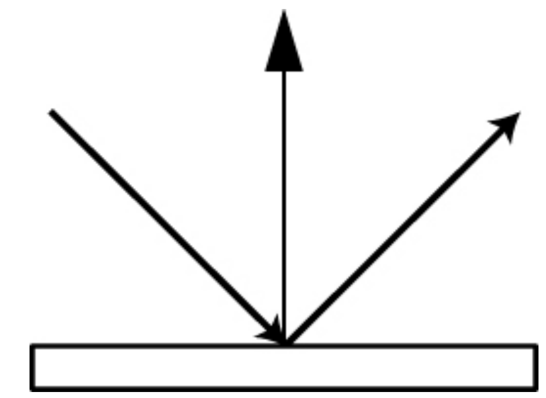
Reflection models

- *Reflection* is the process by which light incident on a surface interacts with the surface such that it leaves on the incident (same) side without change in frequency
- Choice of reflection function determines surface appearance

Some basic reflection functions

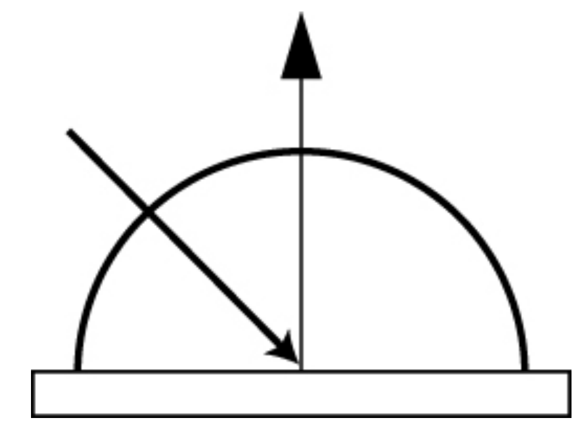
- **Ideal specular**

  Perfect mirror



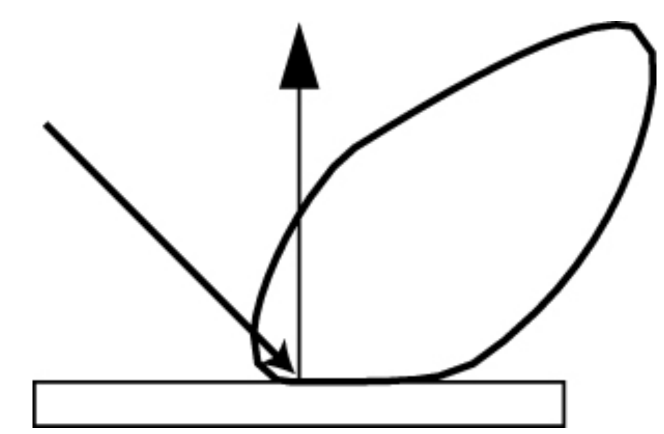
- **Ideal diffuse**

  Uniform reflection in all directions



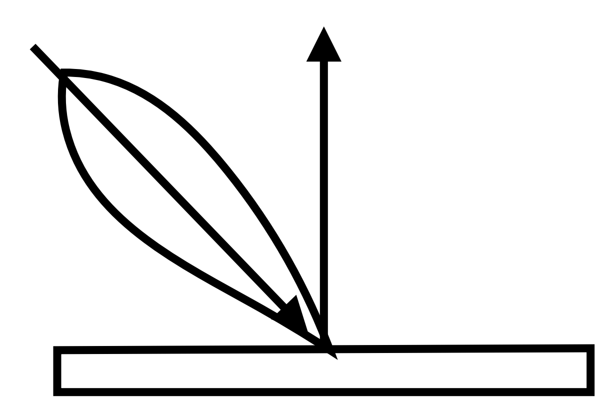

- **Glossy specular**

  Majority of light distributed in reflection direction



- **Retro-reflective**

  Reflects light back toward source




Diagrams illustrate how incoming light energy from given direction is reflected in various directions.

Materials: diffuse

Materials: plastic

Materials: red semi-gloss paint

Materials: Ford mystic lacquer paint

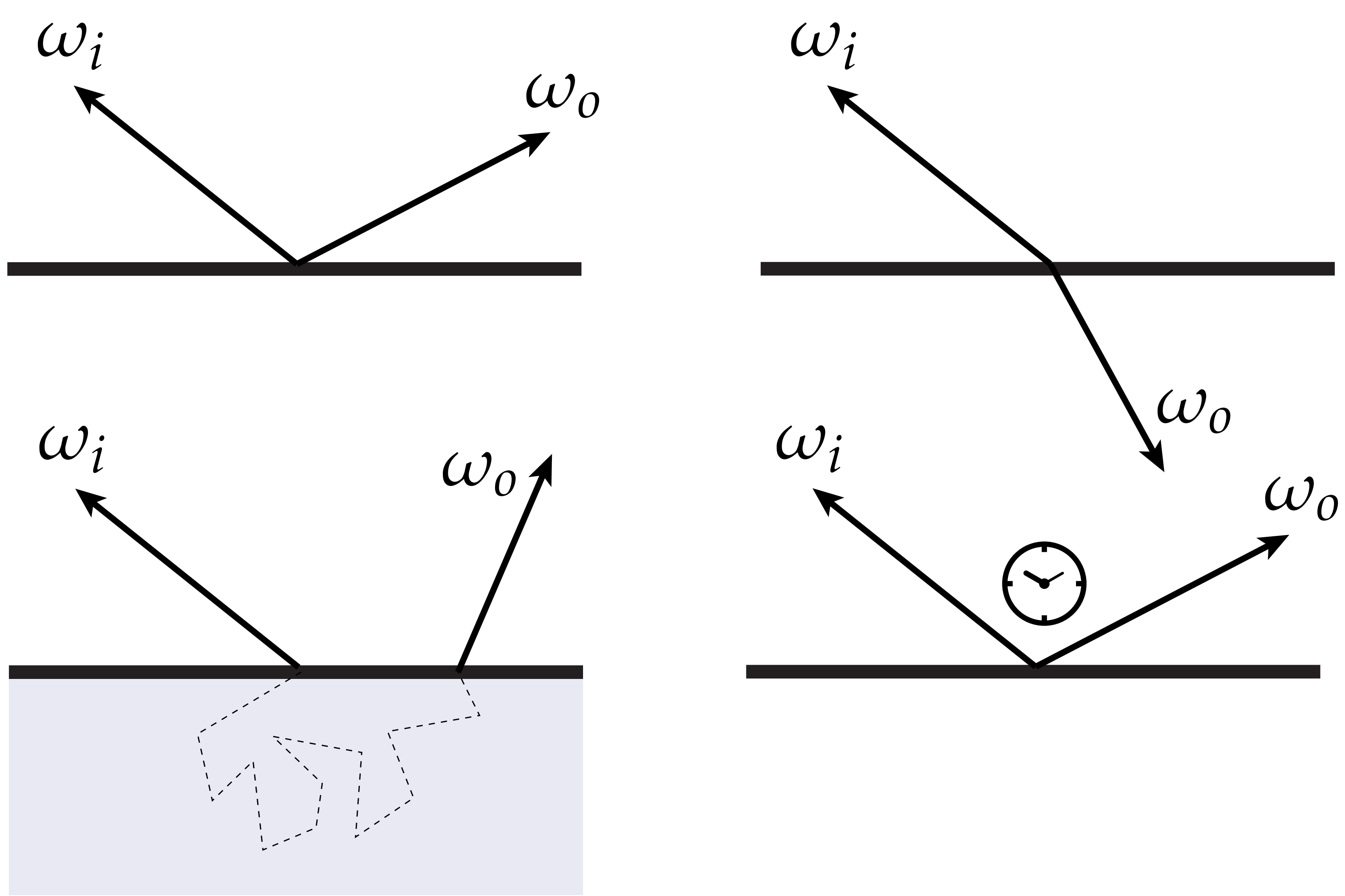
Materials: mirror

Materials: gold

Materials

Models of Scattering

- How can we model "scattering" of light?
- Many different things that could happen to a photon:
  - bounces off surface
  - transmitted through surface
  - bounces around inside surface
  - absorbed & re-emitted
  - ...

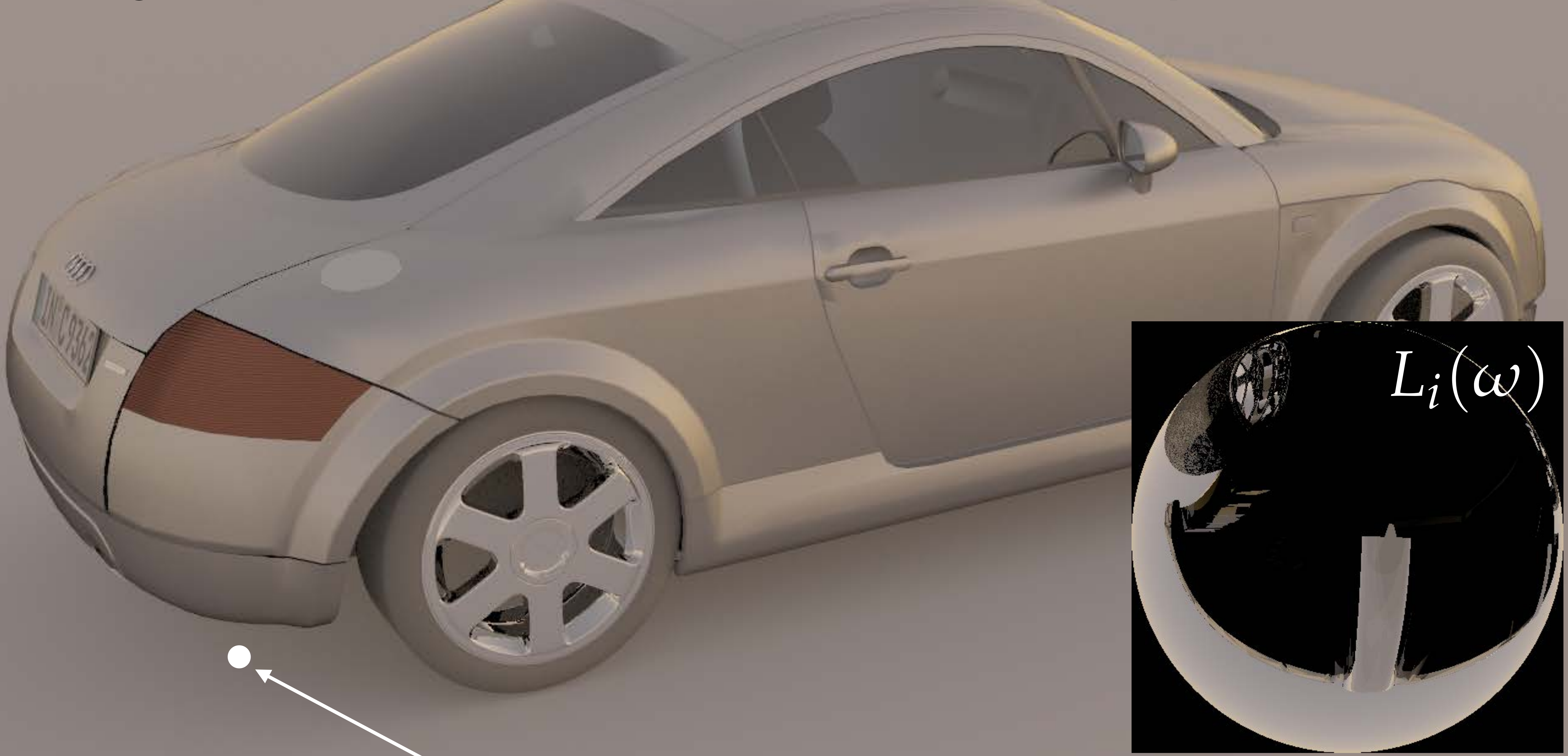


- What goes in must come out! (Total energy must be conserved)
- In general, can talk about *"probability*"* a particle arriving from a given direction is scattered in another direction

*Somewhat more complicated than this, because some light is absorbed!

Hemispherical incident radiance

At any point on any surface in the scene, there's an incident radiance field that gives the directional distribution of illumination at the point

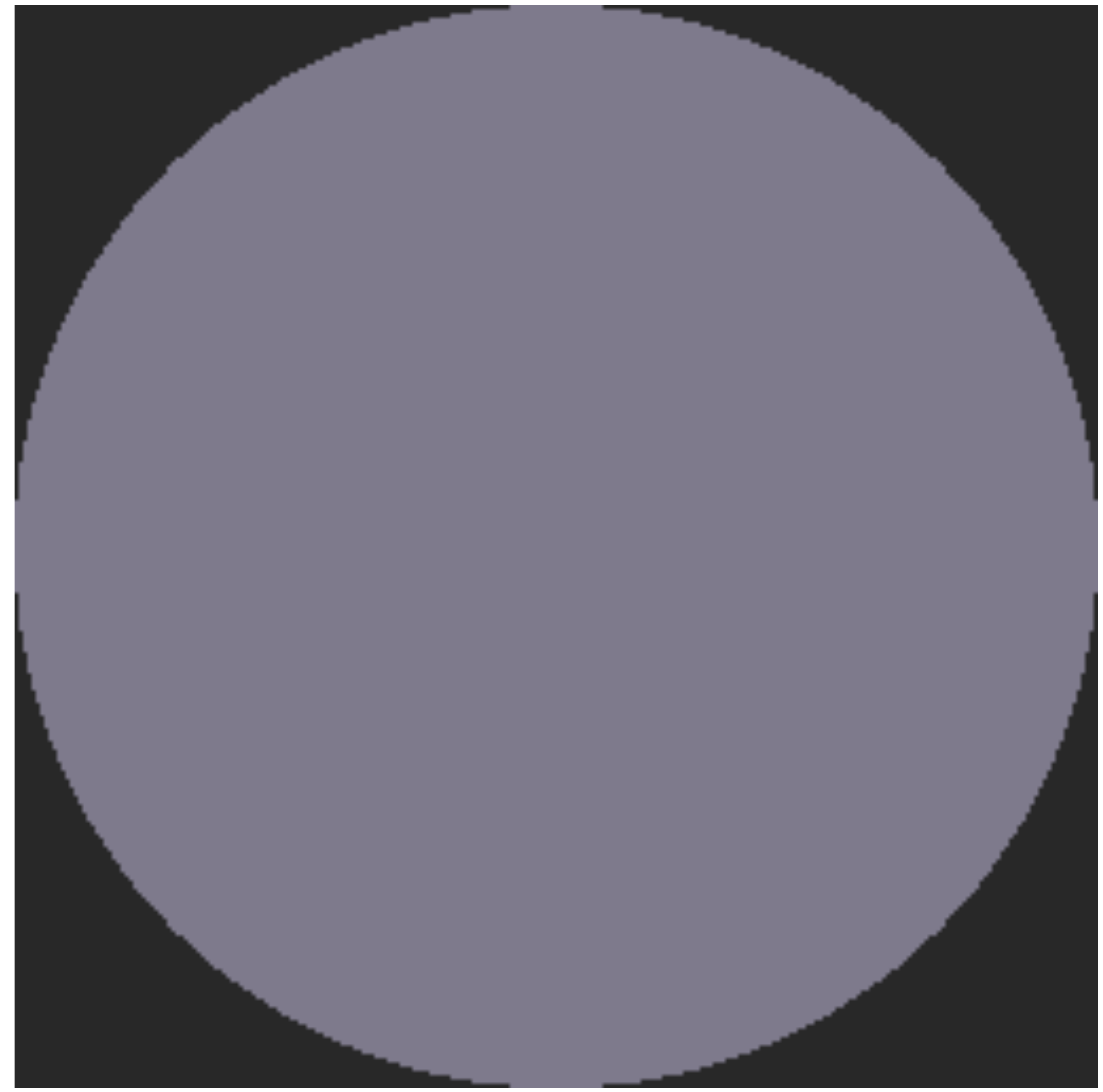


Consider view of hemisphere from this point

Diffuse reflection

Exitant radiance is the same in all directions

Incident radiance

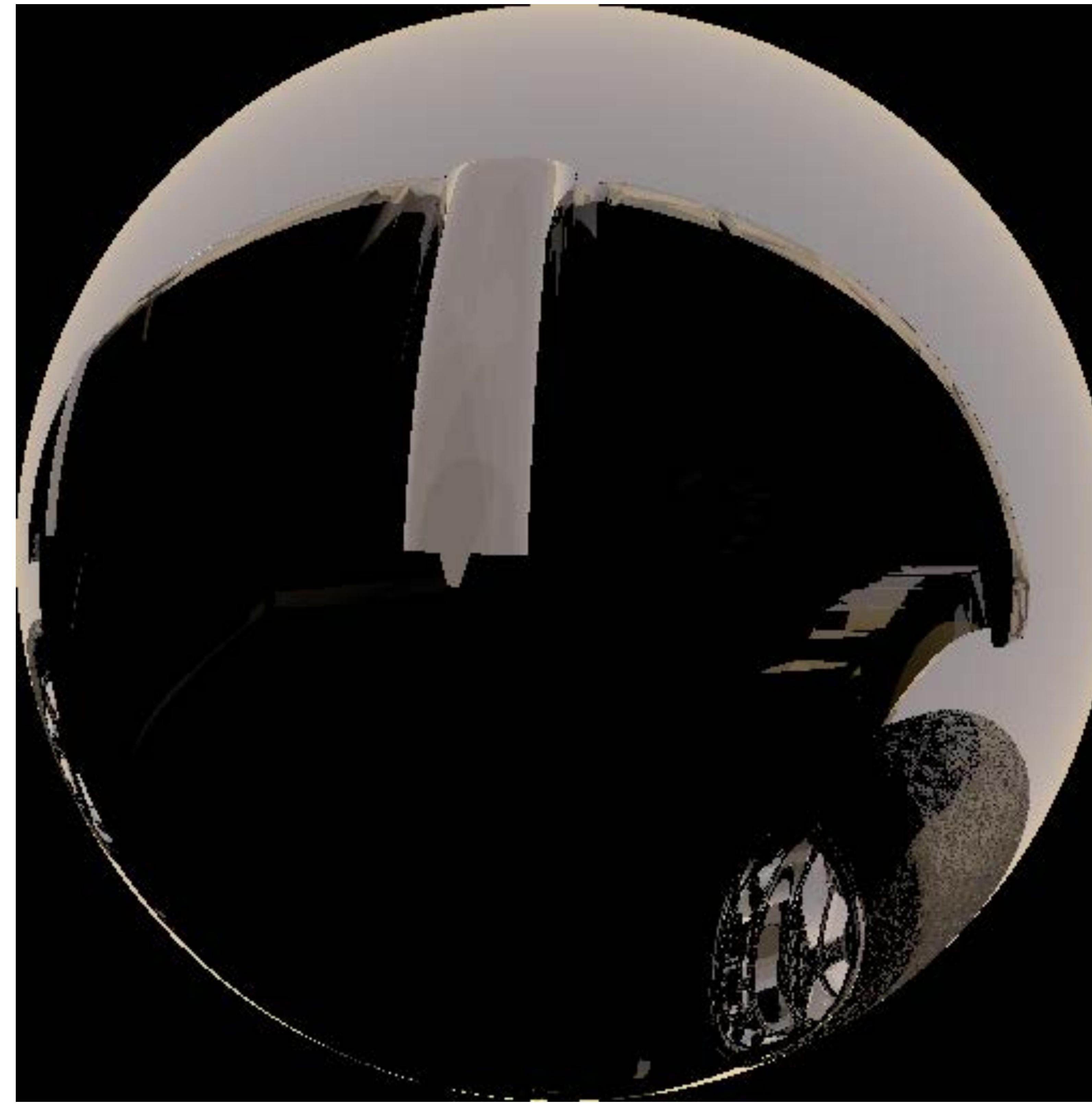


Exitant radiance

Ideal specular reflection

Incident radiance is “flipped around normal” to get exitant radiance

Incident radiance



Exitant radiance

Plastic

Incident radiance gets “flipped and blurred”

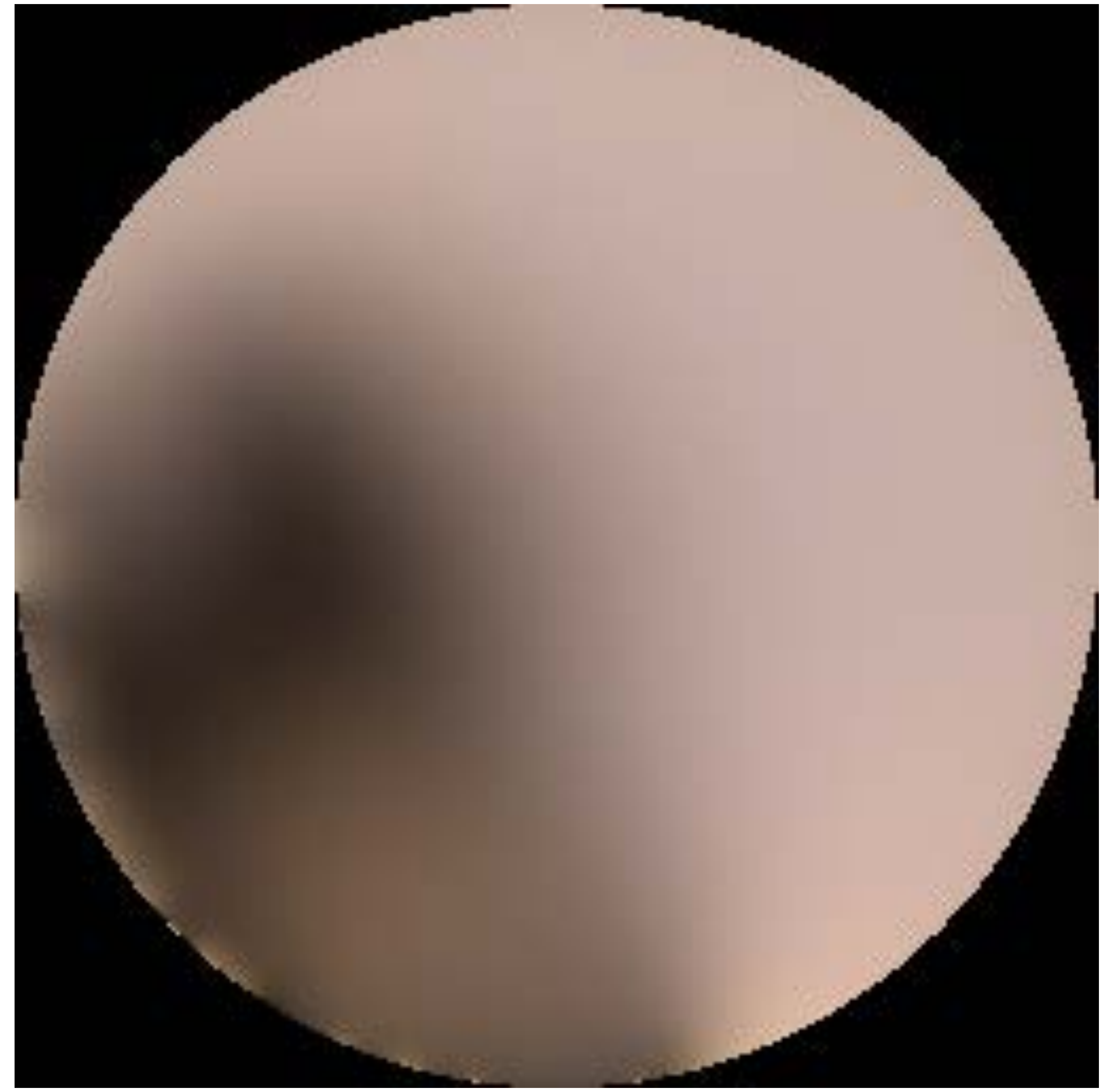
Incident radiance

Exitant radiance

Copper

More blurring, plus coloration (nonuniform absorption across frequencies)

Incident radiance



Exitant radiance

Scattering off a surface: the BRDF

- "Bidirectional reflectance distribution function"
- Encodes behavior of light that "bounces off" surface
- Given incoming direction ω_i, how much light gets scattered in any given outgoing direction ω_o?
- Describe as distribution $f_r(\omega_i \to \omega_o)$

$$f_r(\omega_i \to \omega_o) \geq 0$$

why *less* than or equal?

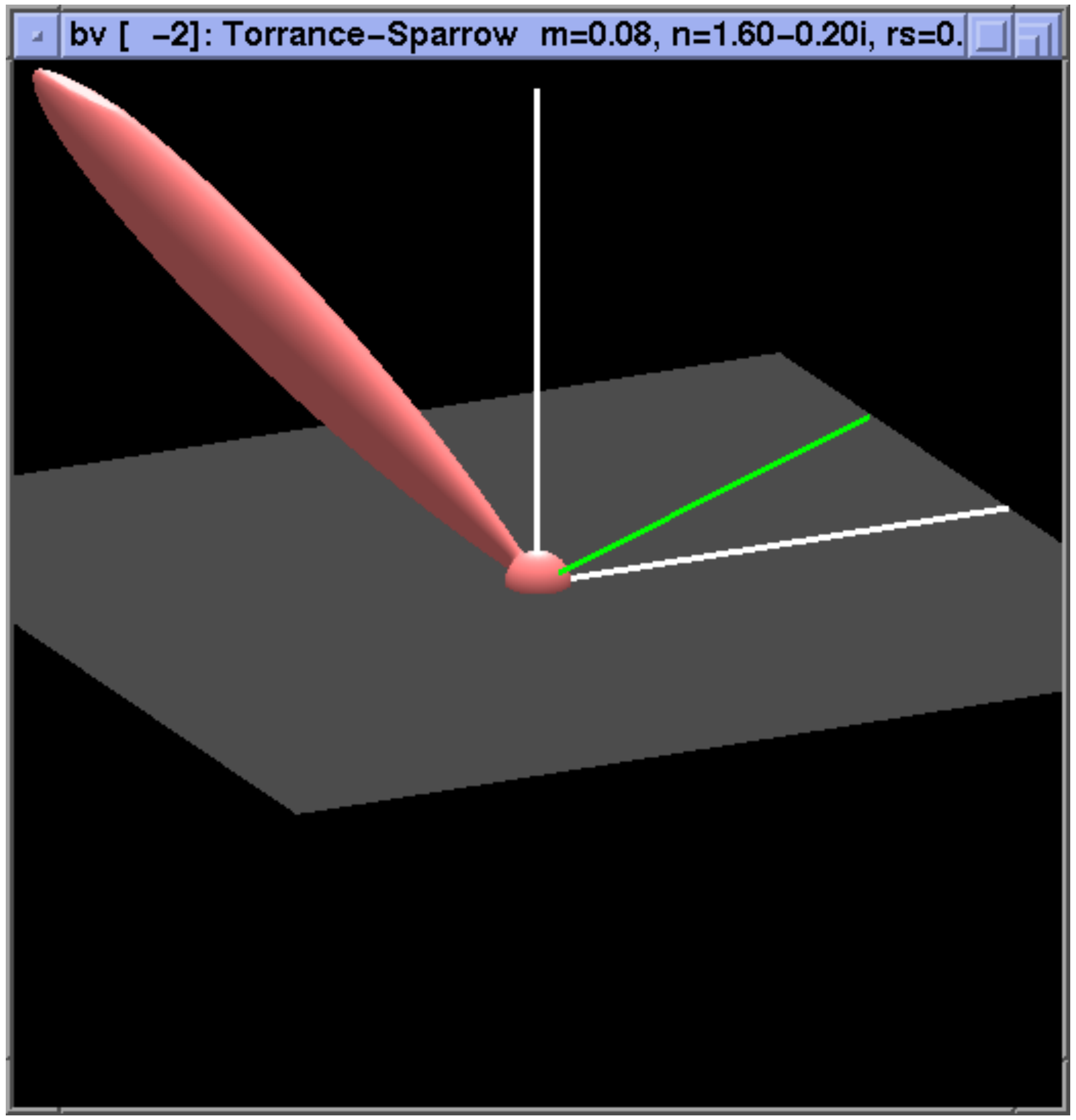
$$\int_{\mathcal{H}^2} f_r(\omega_i \to \omega_o) \cos\theta \, d\omega_i \leq 1$$

where did the rest of the energy go?!

$$f_r(\omega_i \to \omega_o) = f_r(\omega_o \to \omega_i)$$

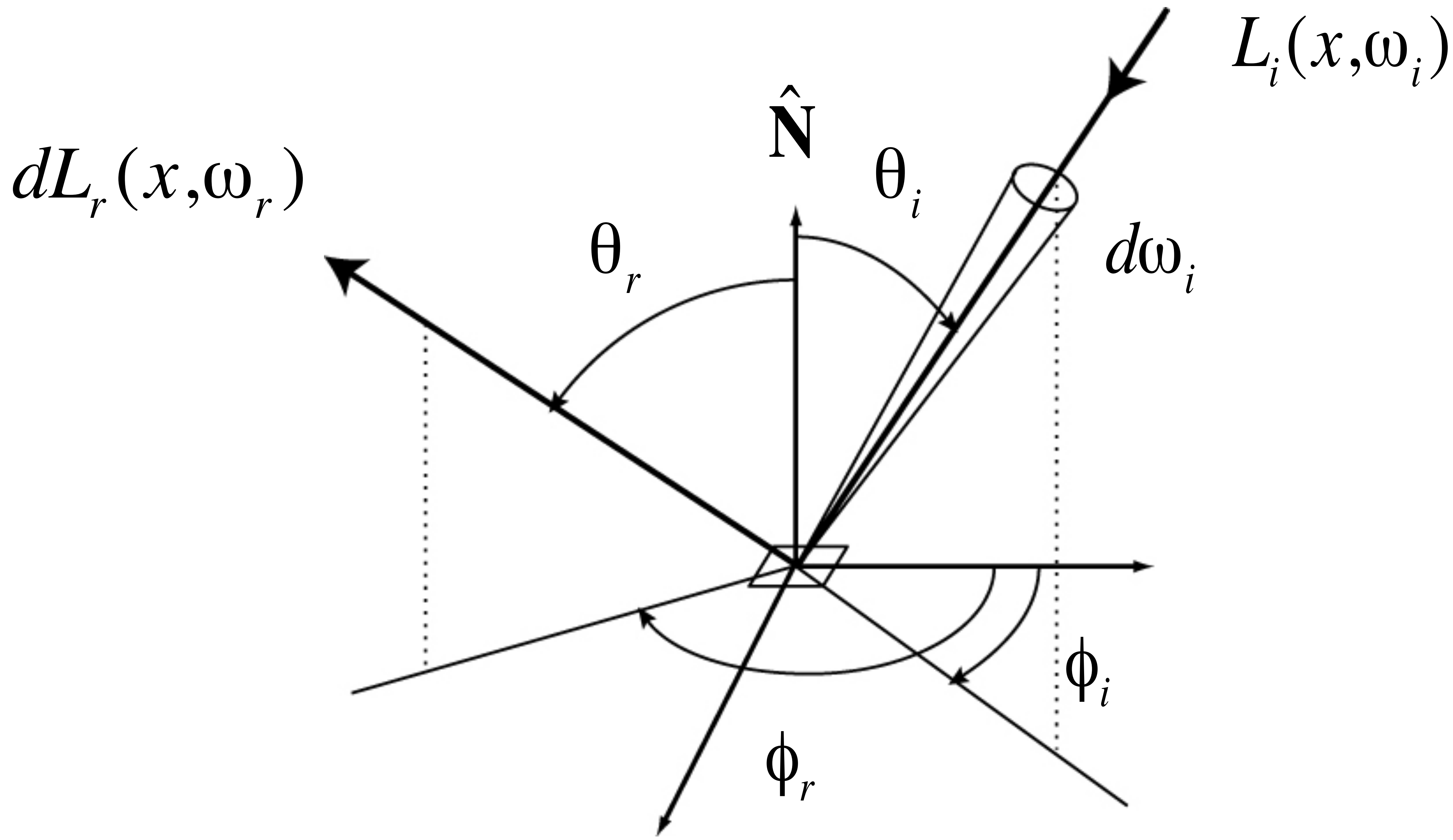
"Helmholtz reciprocity"

Q: Why should Helmholtz reciprocity hold? Think about little mirrors…



bv (Szymon Rusinkiewicz)

Radiometric description of BRDF

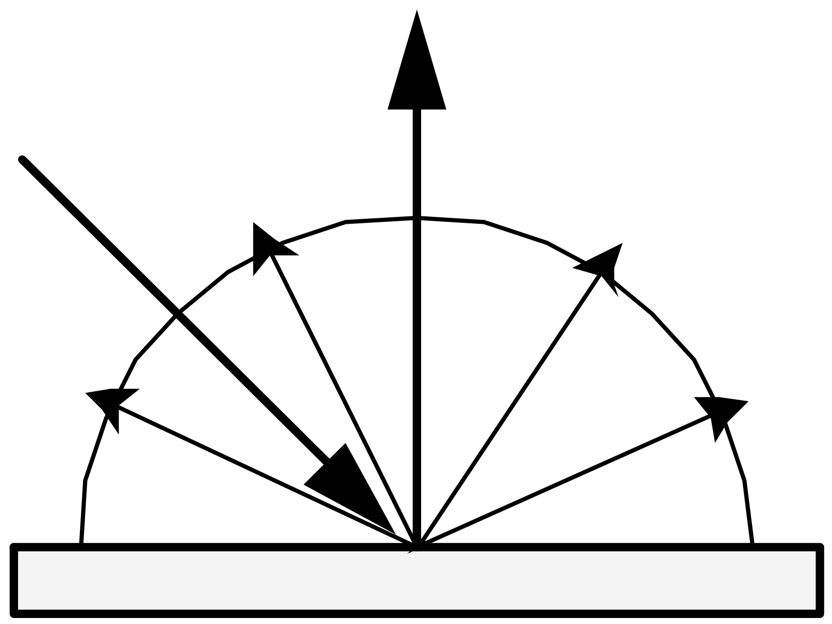


$$f_r(\omega_i \to \omega_o) = \frac{dL_o(\omega_o)}{dE_i(\omega_i)} = \frac{dL_o(\omega_o)}{dL_i(\omega_i)\cos\theta_i} \left[\frac{1}{sr}\right]$$

"For a given change in the incident irradiance, how much does the exitant radiance change?"

Example: Lambertian reflection

Assume light is equally likely to be reflected in each output direction



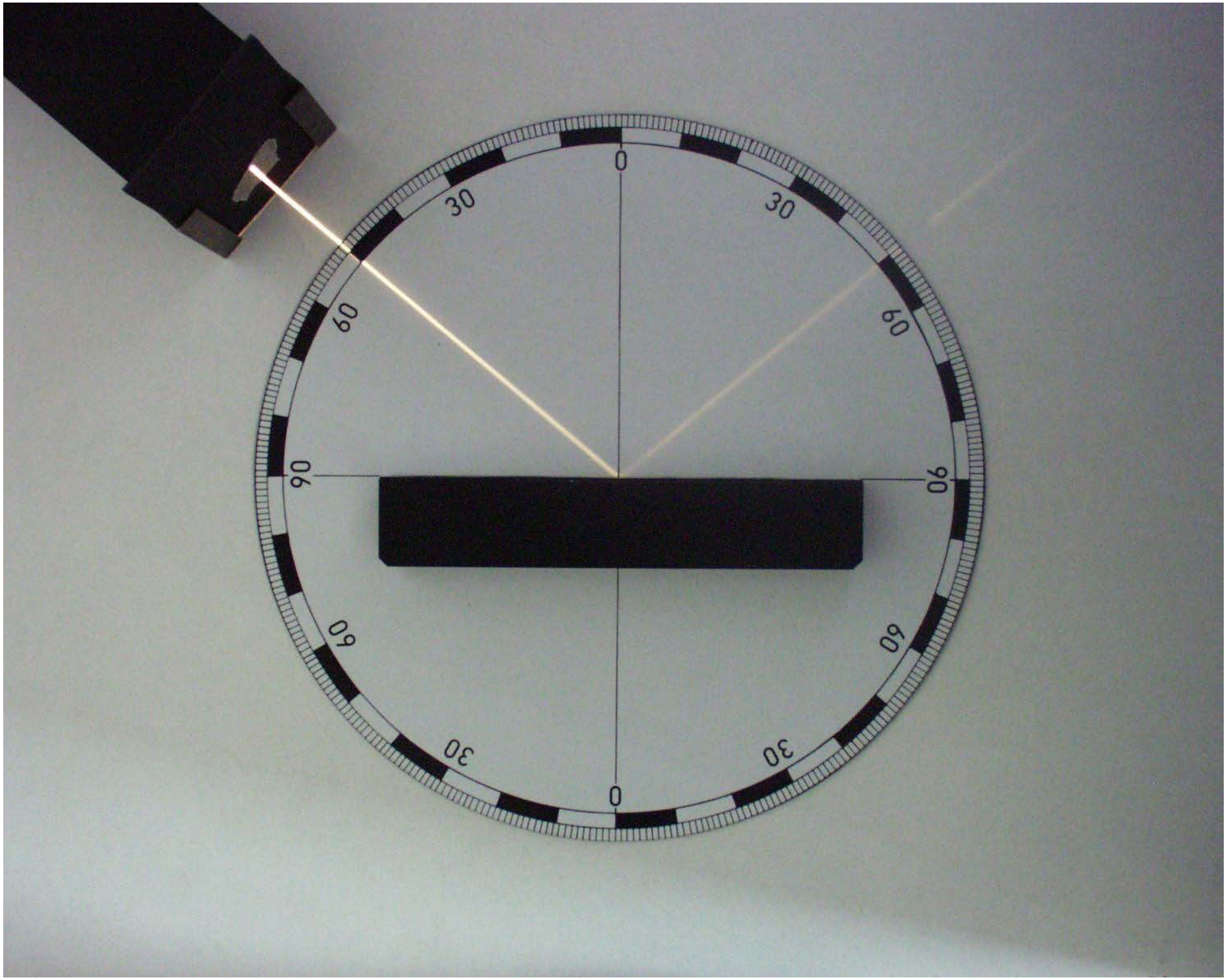
$$f_r = c$$

$$L_o(\omega_o) = \int_{H^2} f_r\, L_i(\omega_i)\, \cos\theta_i\, \mathrm{d}\omega_i$$
$$= f_r \int_{H^2} L_i(\omega_i)\, \cos\theta_i\, \mathrm{d}\omega_i$$
$$= f_r E$$

"albedo" (between 0 and 1)

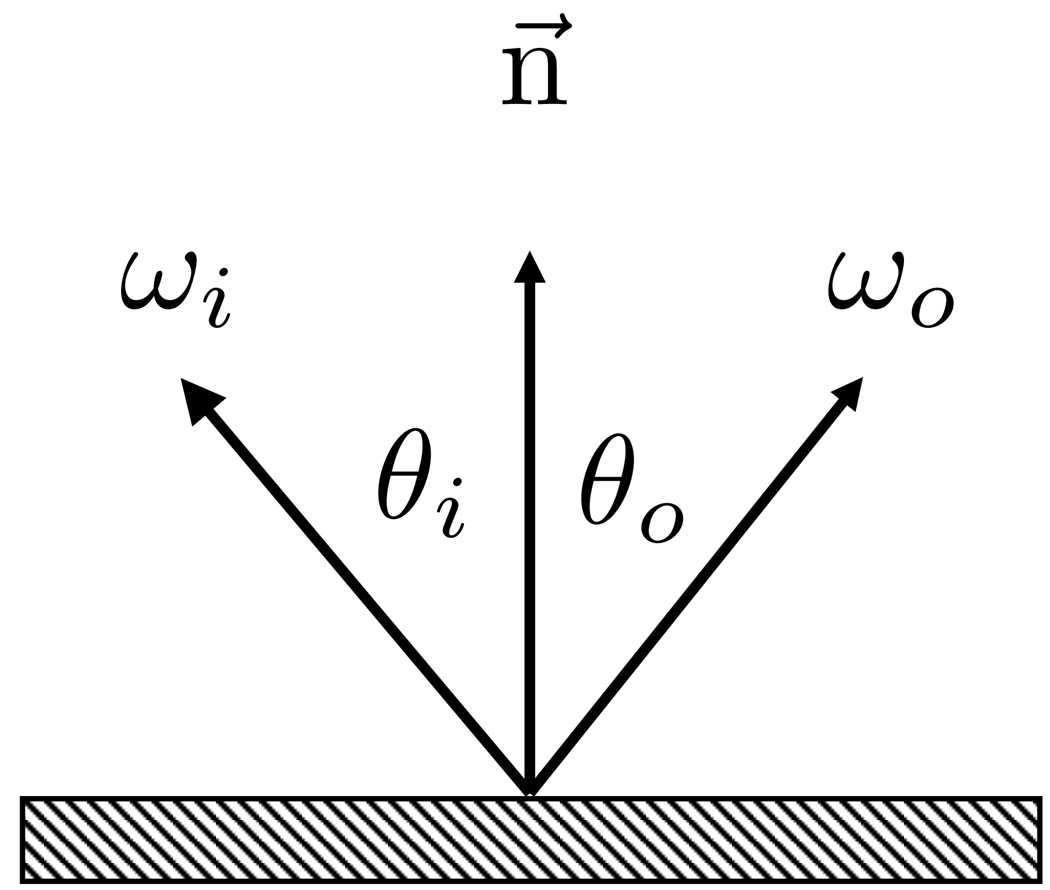
$$f_r = \frac{\rho}{\pi}$$

Example: perfect specular reflection



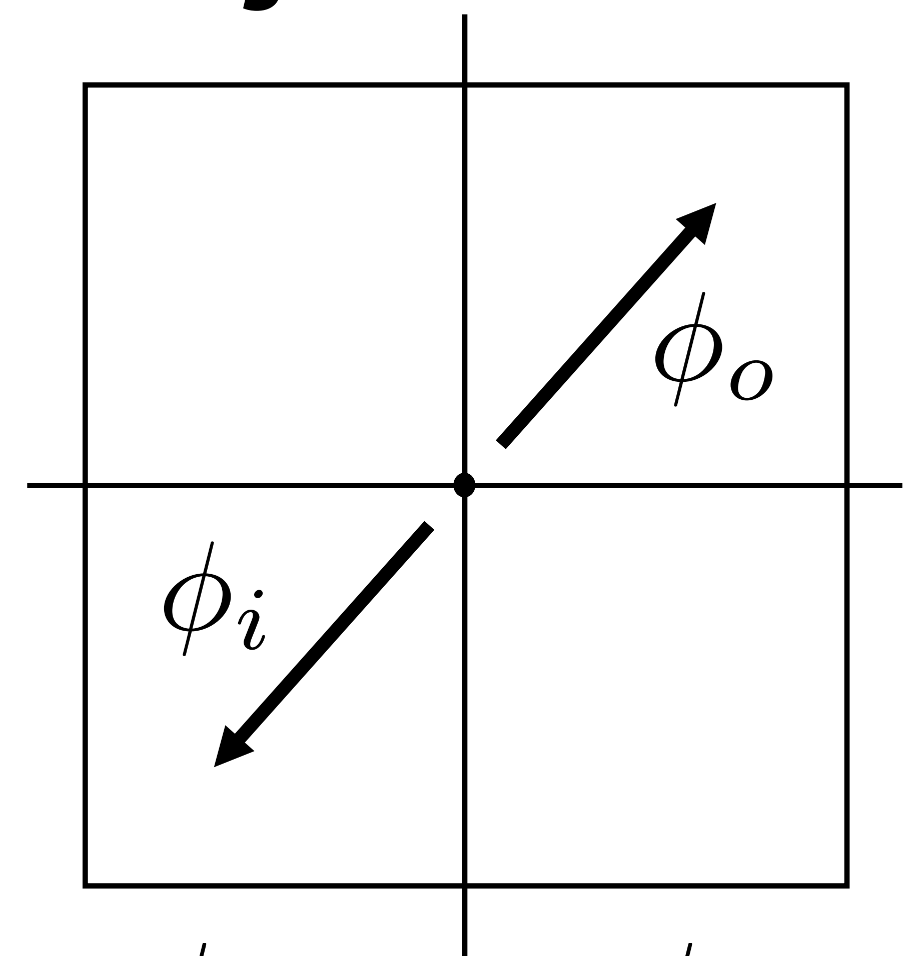
[Zátonyi Sándor]

Geometry of specular reflection



$$\theta = \theta_o = \theta_i$$

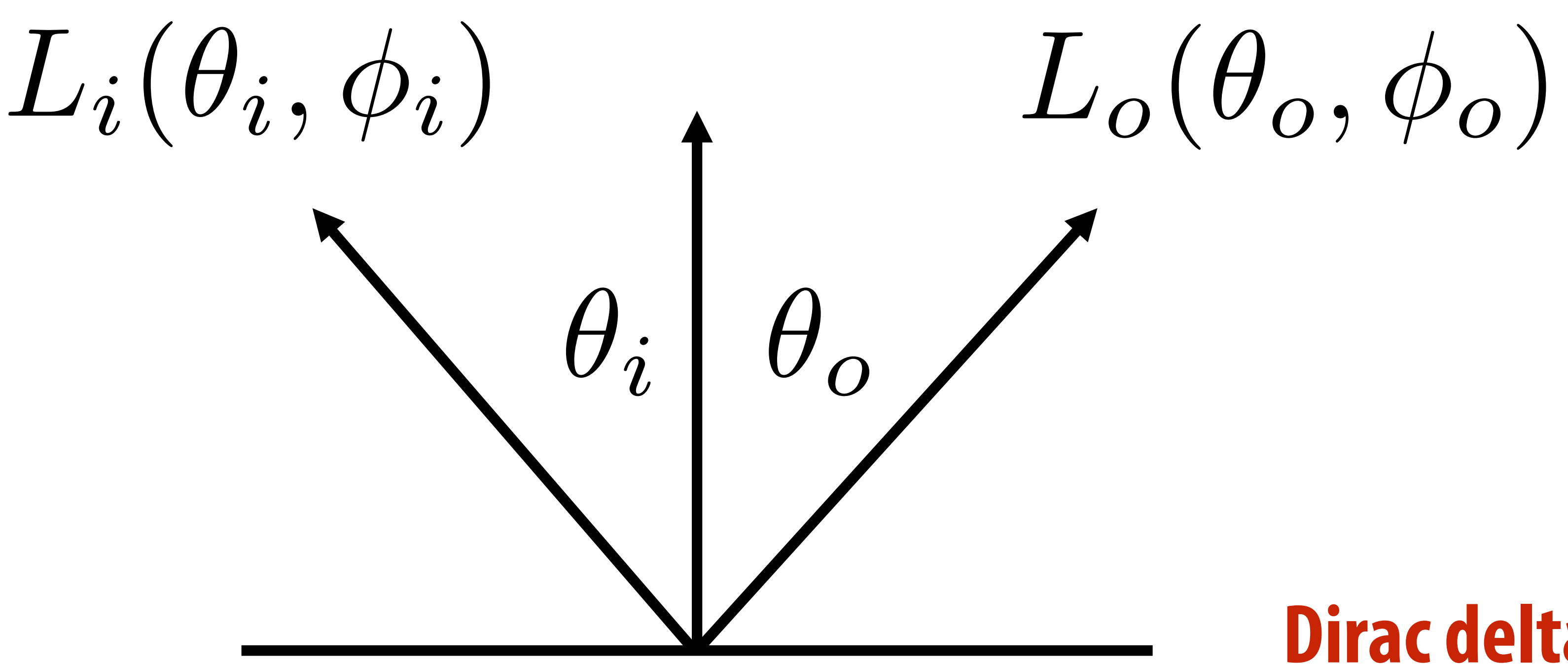
Top-down view
(looking down on surface)



$$\phi_o = -\phi_i$$

$$\omega_o = -\omega_i + 2(\omega_i \cdot \vec{n})\vec{n}$$

Specular reflection BRDF



$$L_o(\theta_o, \phi_o) = L_i(\theta_i, \phi_i)$$

Dirac delta

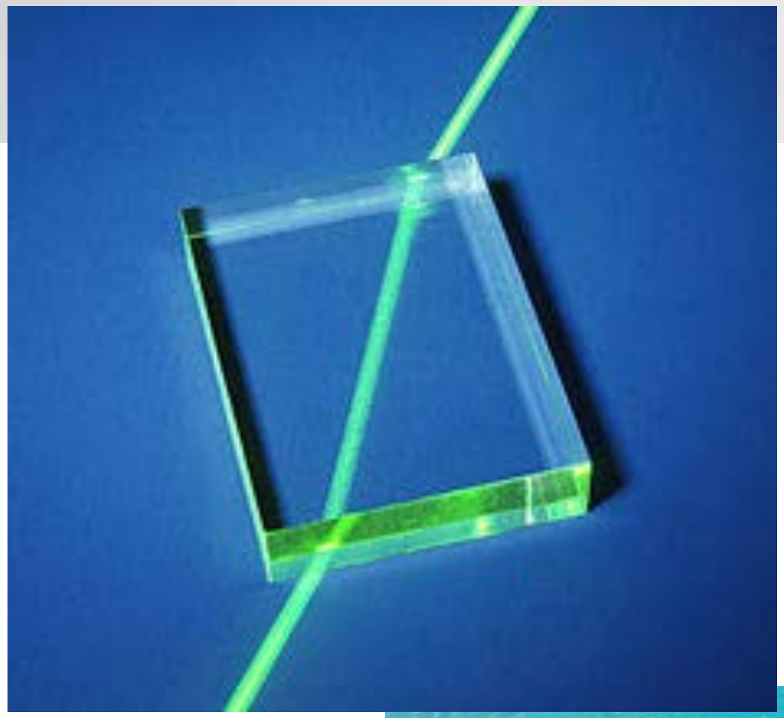
$$f_r(\theta_i, \phi_i; \theta_o, \phi_o) = \frac{\delta(\cos\theta_i - \cos\theta_o)}{\cos\theta_i}\delta(\phi_i - \phi_o \pm \pi)$$

- **Strictly speaking, f_r is a *distribution*, not a function**
- **In practice, no hope of finding reflected direction via random sampling; simply pick the reflected direction!**

Transmission

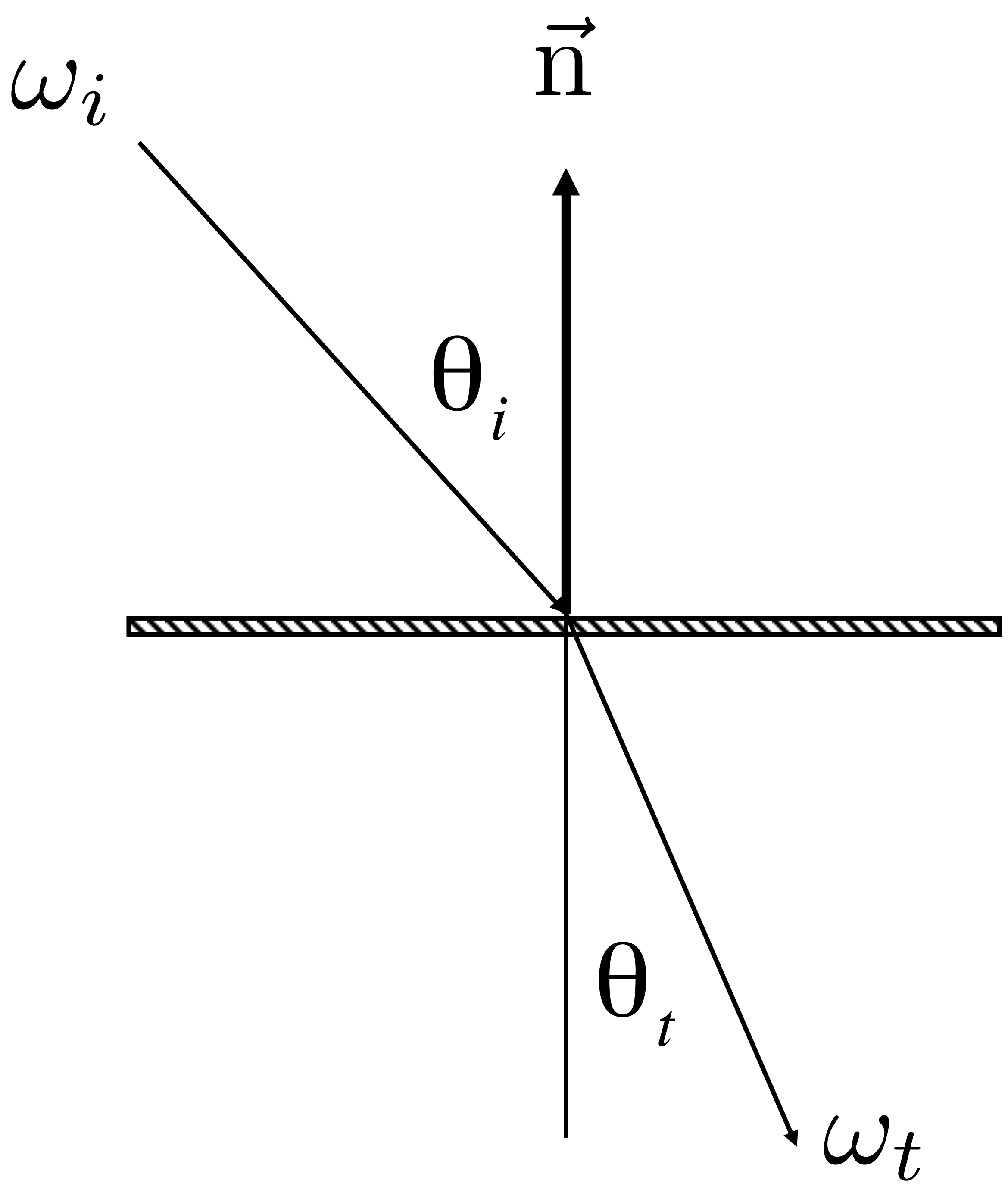
In addition to reflecting off surface, light may be transmitted through surface.

Light refracts when it enters a new medium.

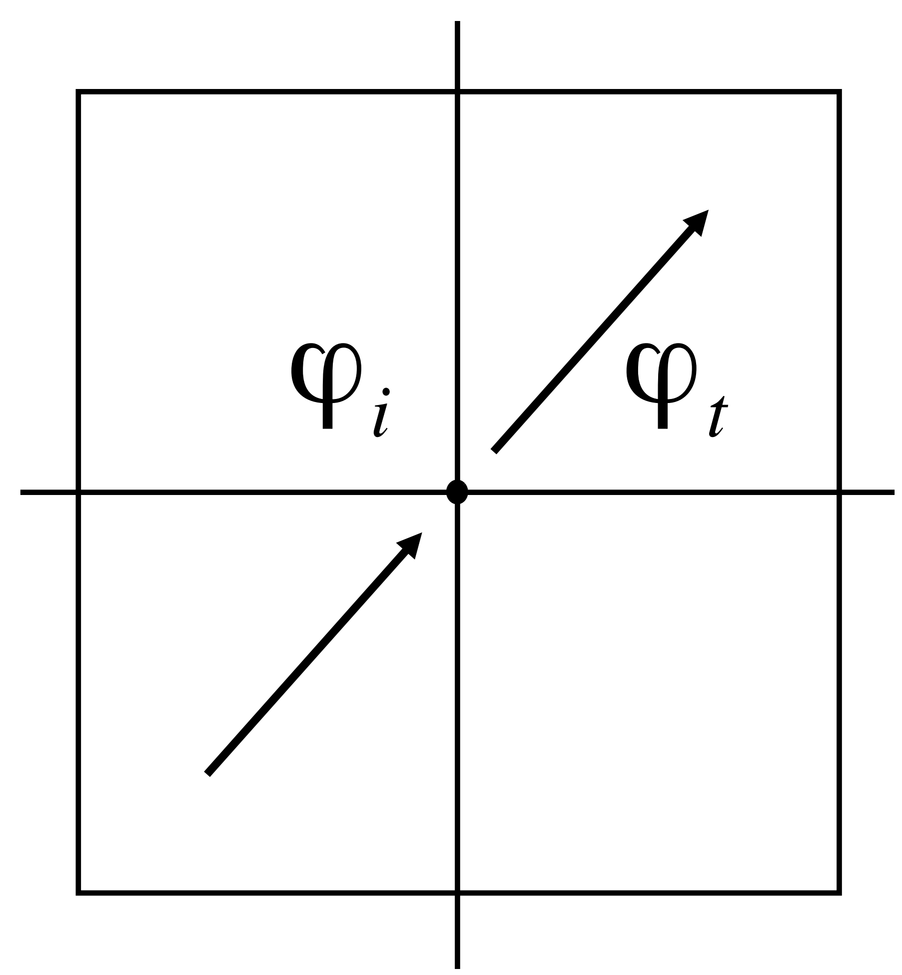


Snell's Law

Transmitted angle depends on relative index of refraction of material ray is leaving/entering.



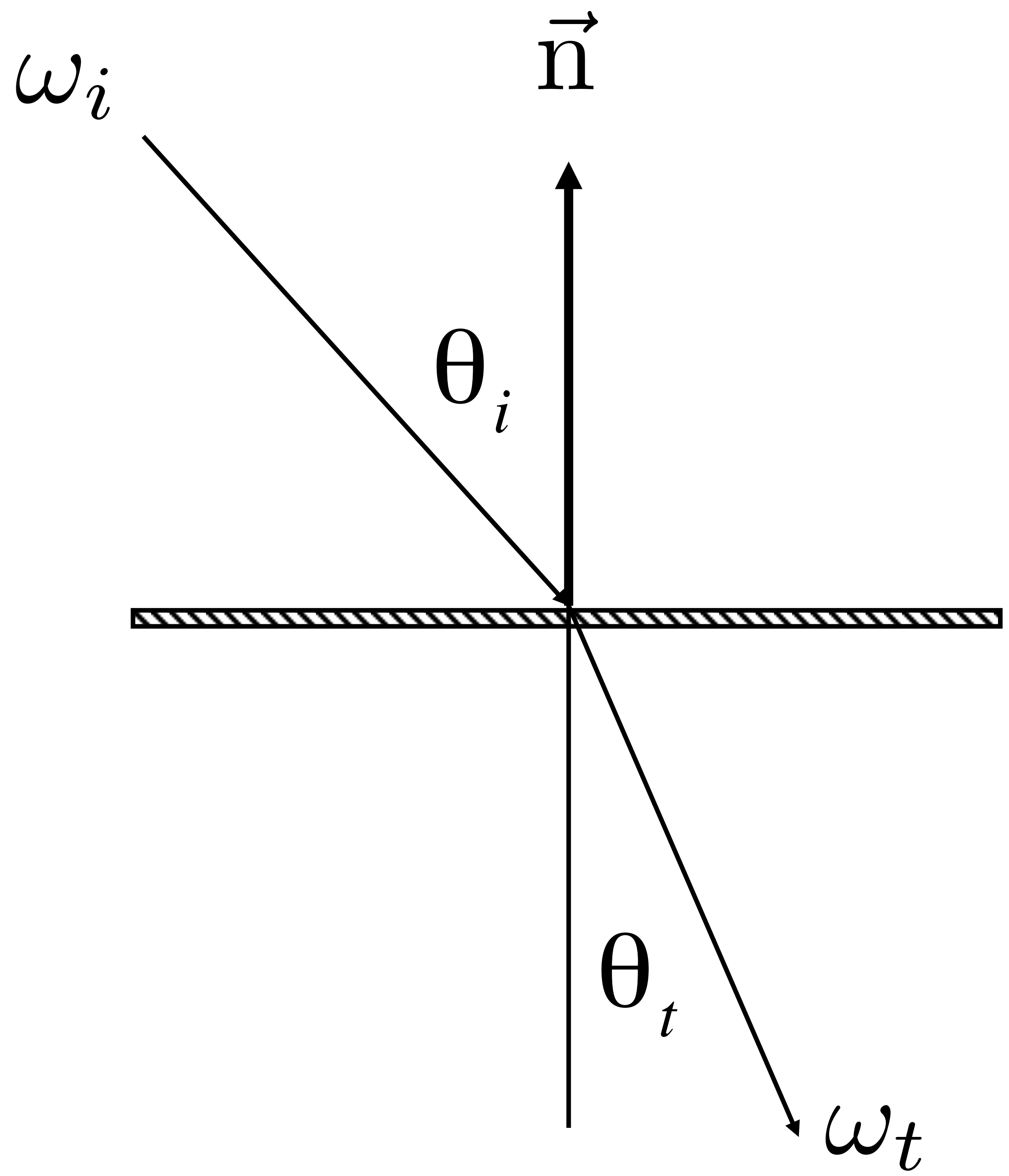
$$\eta_i \sin\theta_i = \eta_t \sin\theta_t$$



Medium	η*
Vacuum	1.0
Air (sea level)	1.00029
Water (20°C)	1.333
Glass	1.5-1.6
Diamond	2.42

*** index of refraction is wavelength dependent (these are averages)**

Law of refraction



$$\eta_i \sin\theta_i = \eta_t \sin\theta_t$$

$$\cos\theta_t = \sqrt{1 - \sin^2\theta_t}$$
$$= \sqrt{1 - \left(\frac{\eta_i}{\eta_t}\right)^2 \sin^2\theta_i}$$
$$= \sqrt{1 - \left(\frac{\eta_i}{\eta_t}\right)^2 (1 - \cos^2\theta_i)}$$

Total internal reflection:

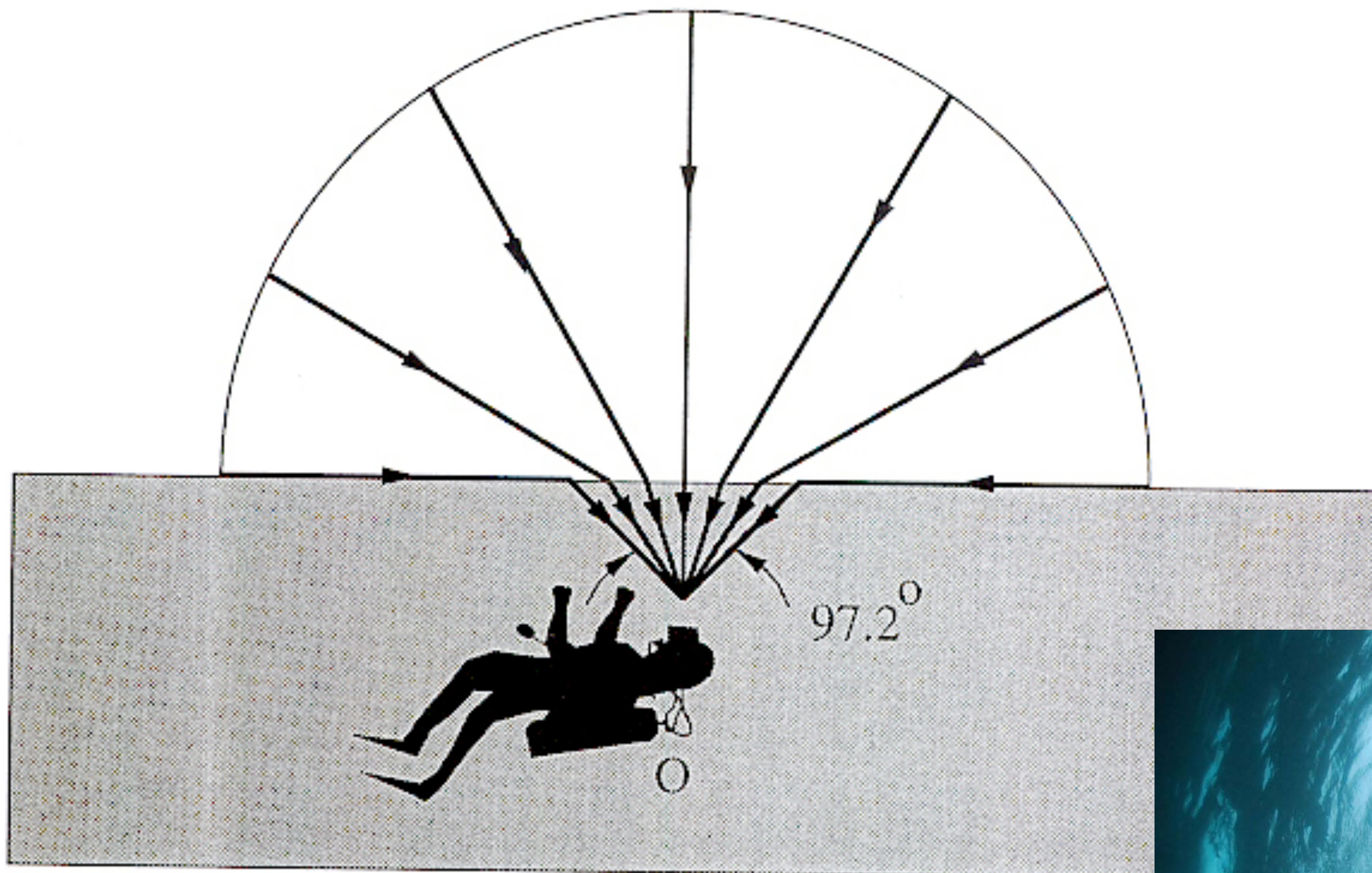
$$1 - \left(\frac{\eta_i}{\eta_t}\right)^2 (1 - \cos^2\theta_i) < 0$$

When light is moving from a more optically dense medium to a less optically dense medium: $\frac{\eta_i}{\eta_t} > 1$

Light incident on boundary from large enough angle will not exit medium.

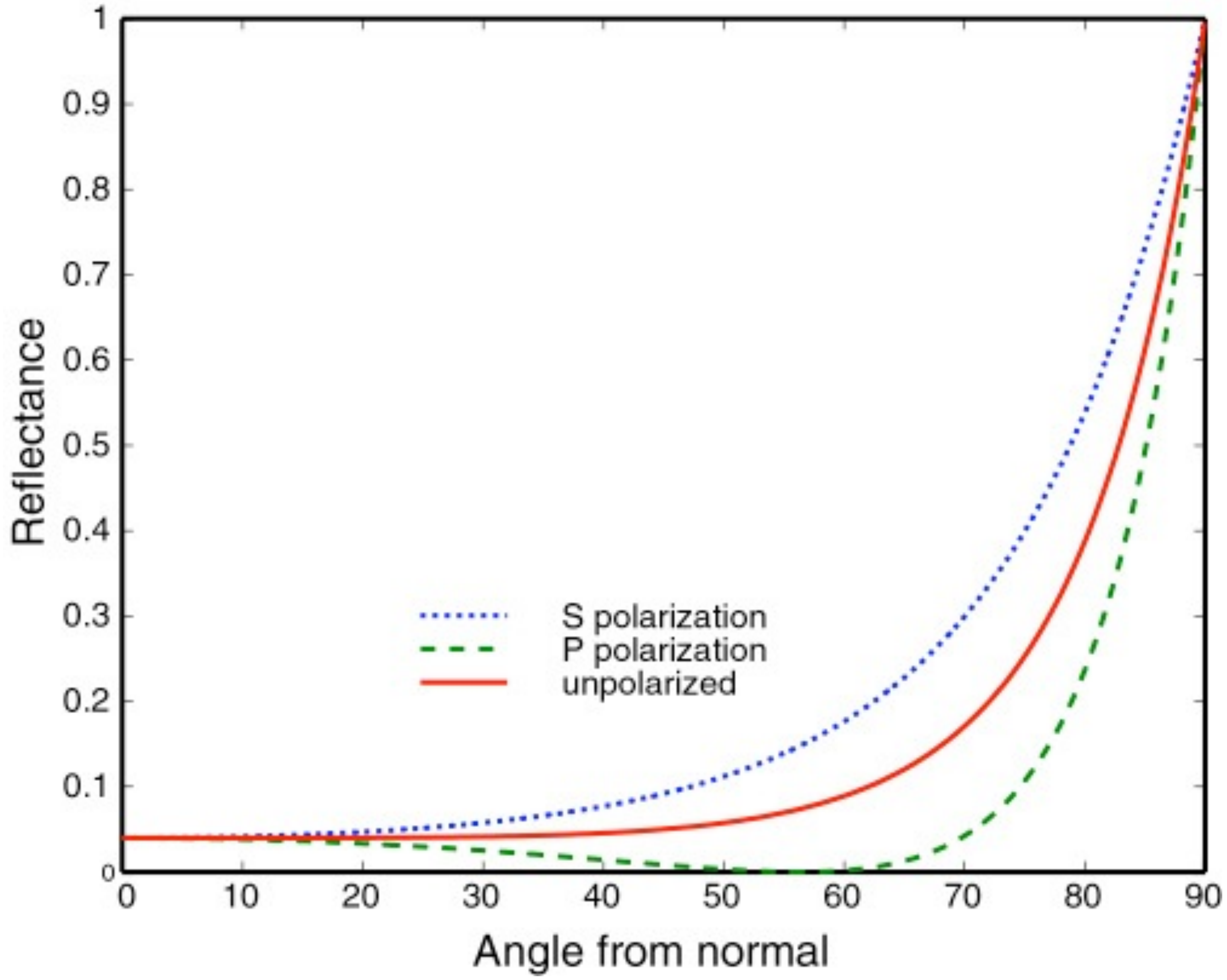
Optical manhole

Only small "cone" visible, due to total internal reflection (TIR)



Fresnel reflection

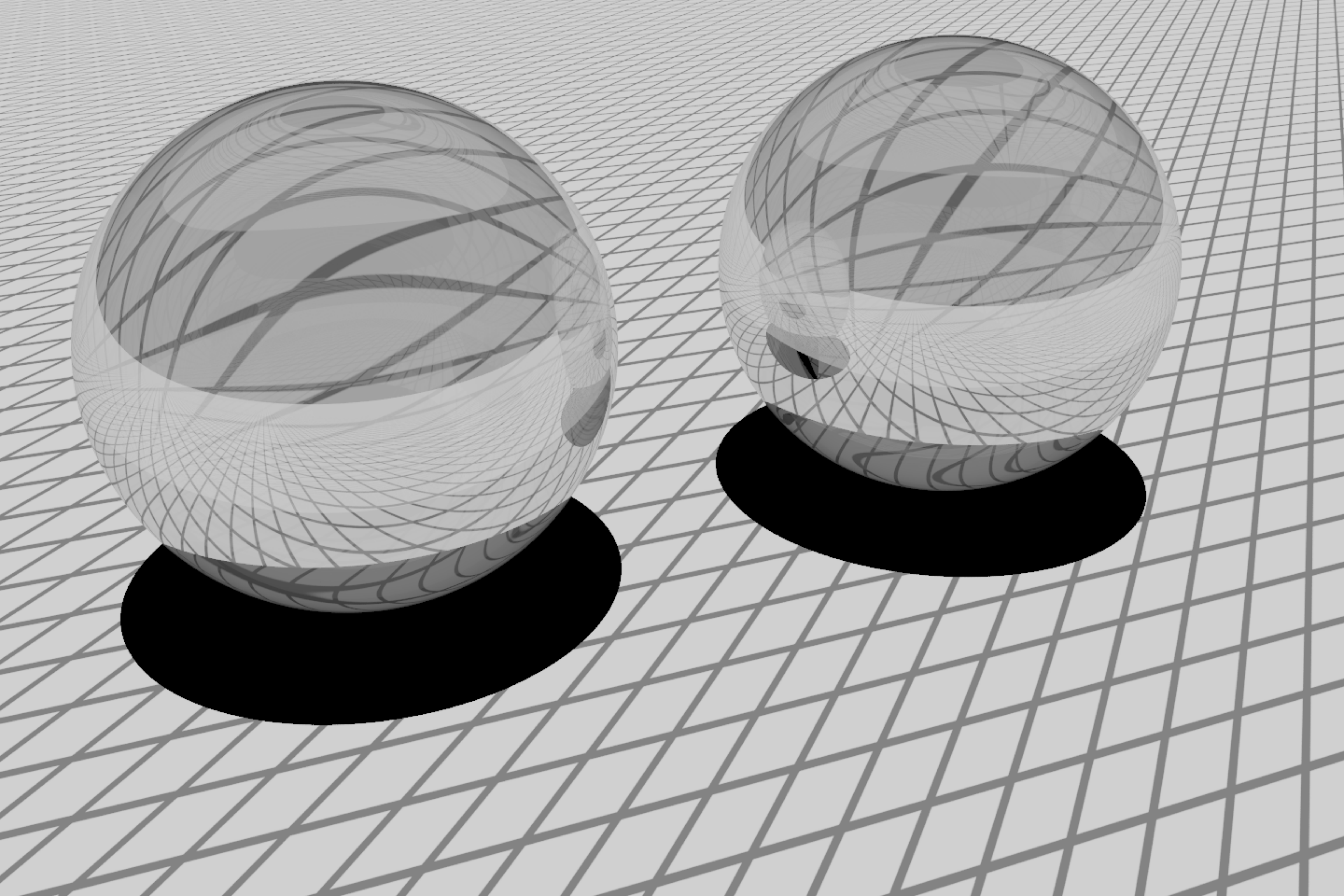
Many real materials: reflectance increases w/ viewing angle



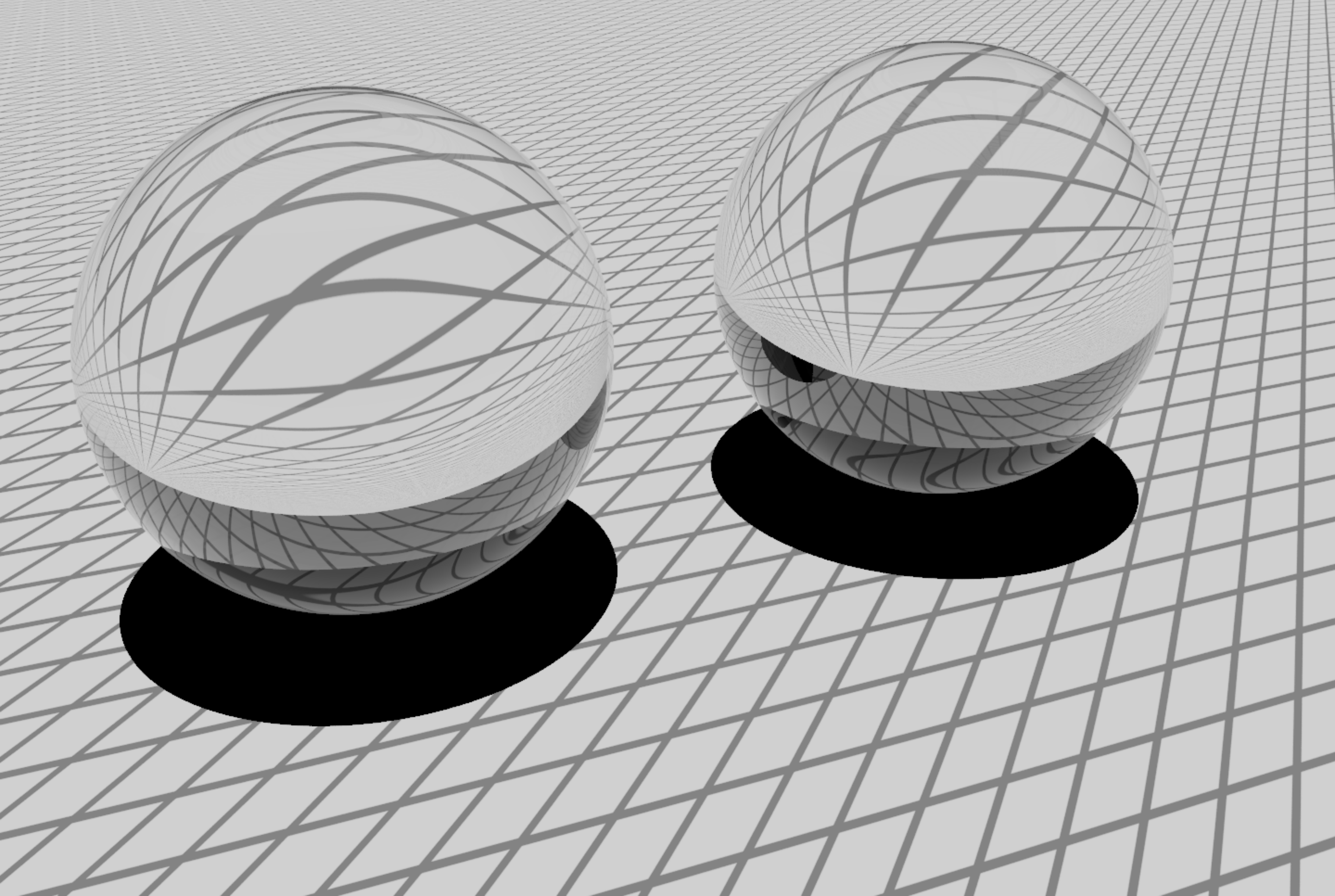
[Lafortune et al. 1997]

Snell + Fresnel: Example

Without Fresnel (fixed reflectance/transmission)



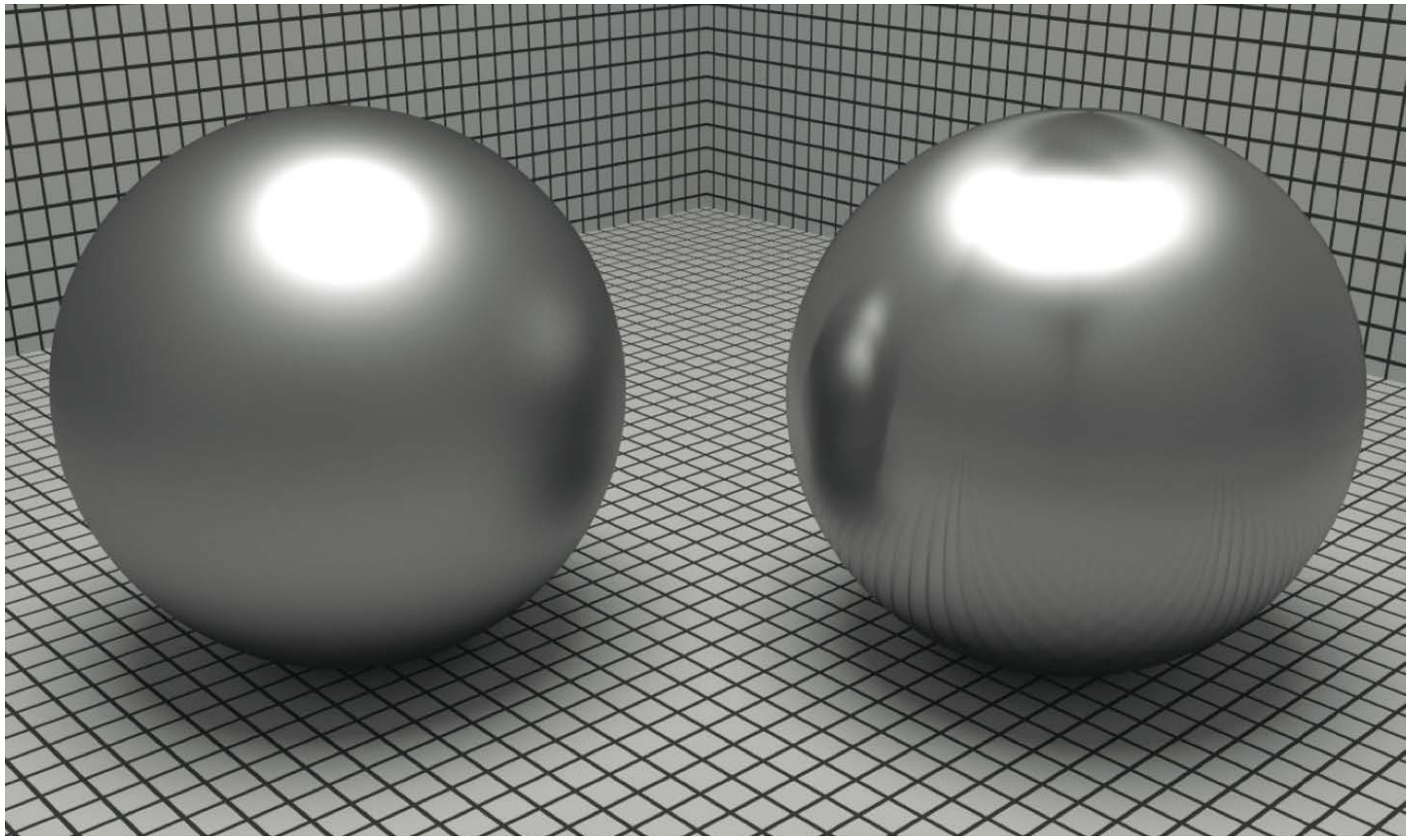
Glass with Fresnel reflection/transmission



Anisotropic reflection

Reflection depends on azimuthal angle ϕ

Results from oriented microstructure of surface e.g., brushed metal



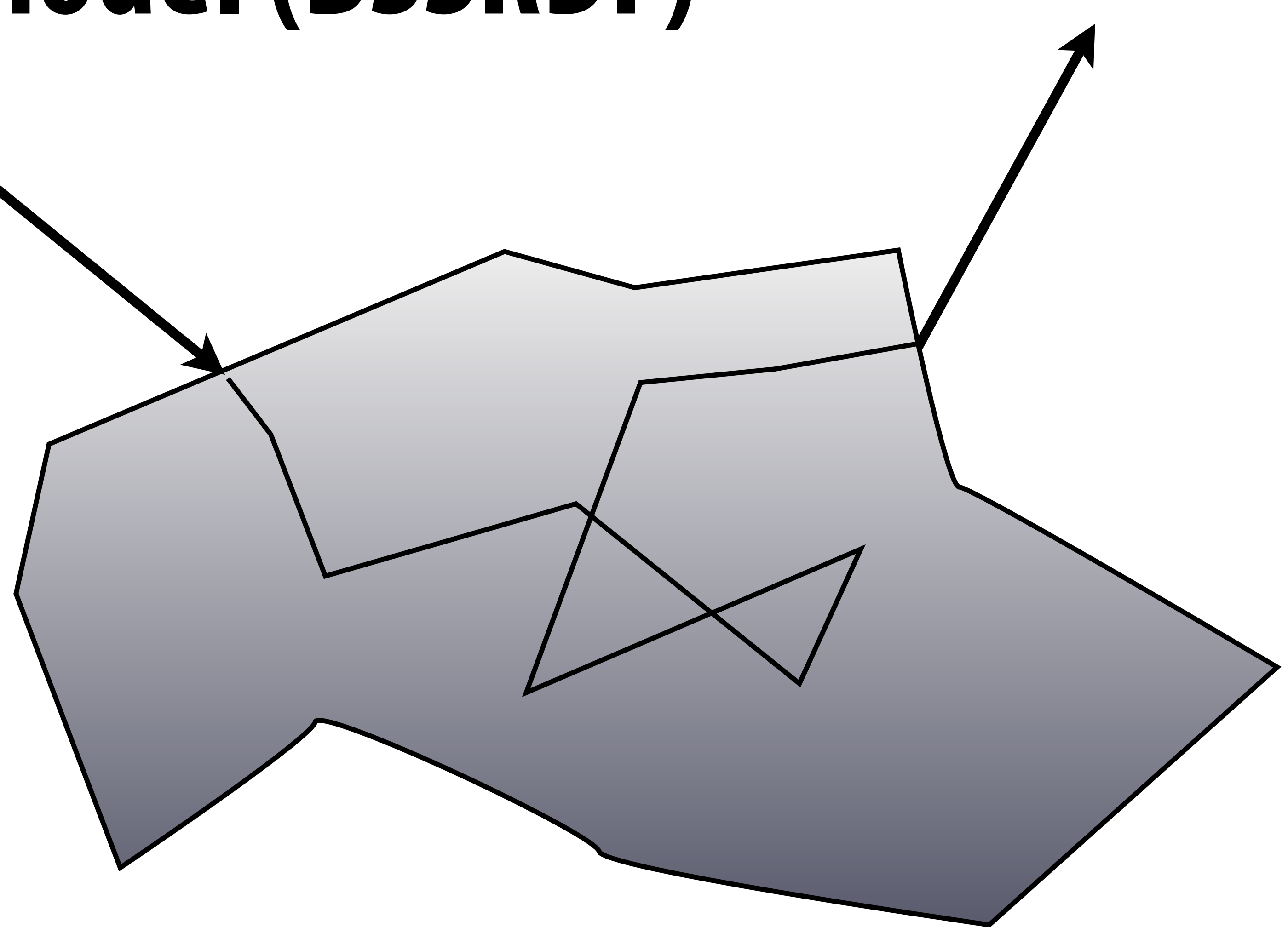
Translucent materials: Jade

Translucent materials: skin

Translucent materials: leaves

Subsurface scattering

- Visual characteristics of many surfaces caused by light entering at different points than it exits
  - Violates a fundamental assumption of the BRDF
  - Need to generalize scattering model (BSSRDF)



[Jensen et al 2001]

[Donner et al 2008]

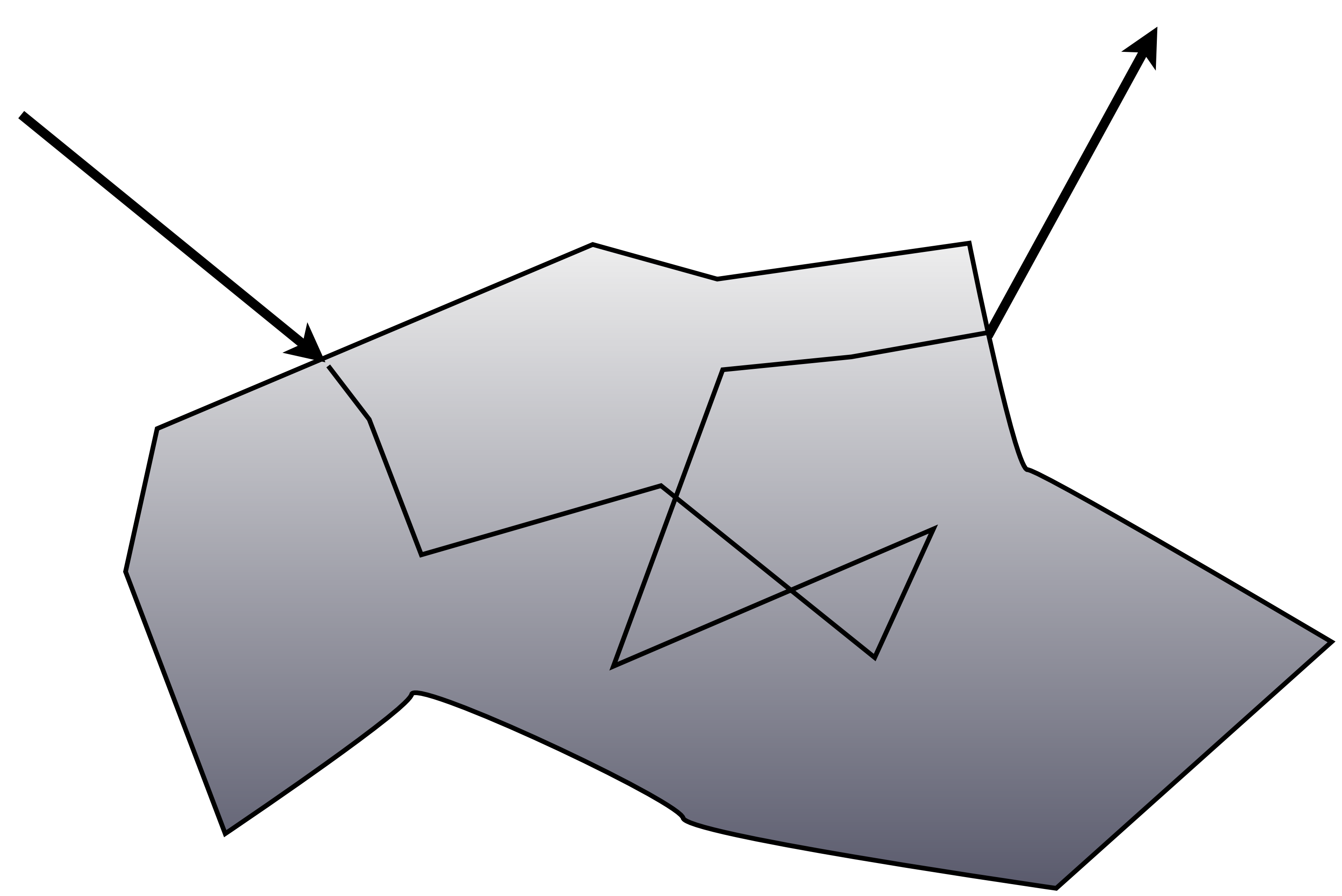
Scattering functions

- **Generalization of BRDF; describes exitant radiance at one point due to incident differential irradiance at another point:**

$$S(x_i, \omega_i, x_o, \omega_o)$$

- **Generalization of reflection equation integrates over all points on the surface and all directions(!)**

$$L(x_o, \omega_o) = \int_A \int_{H^2} S(x_i, \omega_i, x_o, \omega_o)\, L_i(x_i, \omega_i) \cos\theta_i \,\mathrm{d}\omega_i \,\mathrm{d}A$$



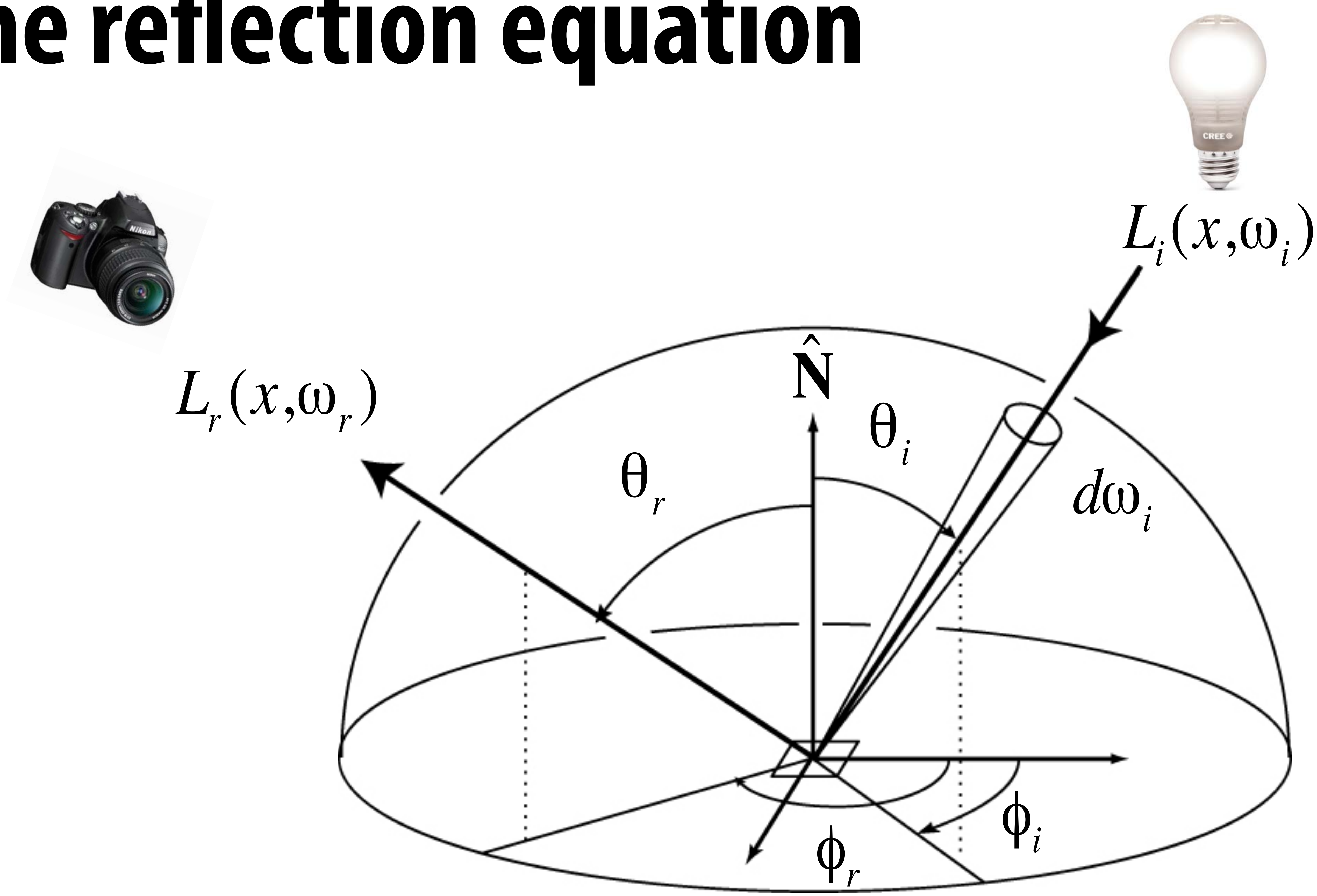
BRDF

BSSRDF

Ok, so scattering is *complicated!*

What's a (relatively simple) algorithm that can capture all this behavior?

The reflection equation



$$\mathrm{d}L_r(\omega_r) = f_r(\omega_i \to \omega_r)\,\mathrm{d}L_i(\omega_i)\,\cos\theta_i$$

$$L_r(\mathrm{p},\omega_r) = \int_{H^2} f_r(\mathrm{p},\omega_i \to \omega_r)\,L_i(\mathrm{p},\omega_i)\,\cos\theta_i\,\mathrm{d}\omega_i$$

The reflection equation

- **Key piece of overall rendering equation:**

$$L_r(\mathrm{p}, \omega_r) = \int_{H^2} f_r(\mathrm{p}, \omega_i \to \omega_r)\, L_i(\mathrm{p}, \omega_i)\, \cos\theta_i \,\mathrm{d}\omega_i$$

  - **Approximate integral via *Monte Carlo integration***
  - **Generate directions ω_j sampled from some distribution $p(\omega)$**
  - **Compute the estimator**

$$\frac{1}{N}\sum_{j=1}^{N} \frac{f_r(\mathrm{p}, \omega_j \to \omega_r)\, L_i(\mathrm{p}, \omega_j)\, \cos\theta_j}{p(\omega_j)}$$

  - **To reduce variance $p(\omega)$ should match BRDF or incident radiance function**

Estimating reflected light

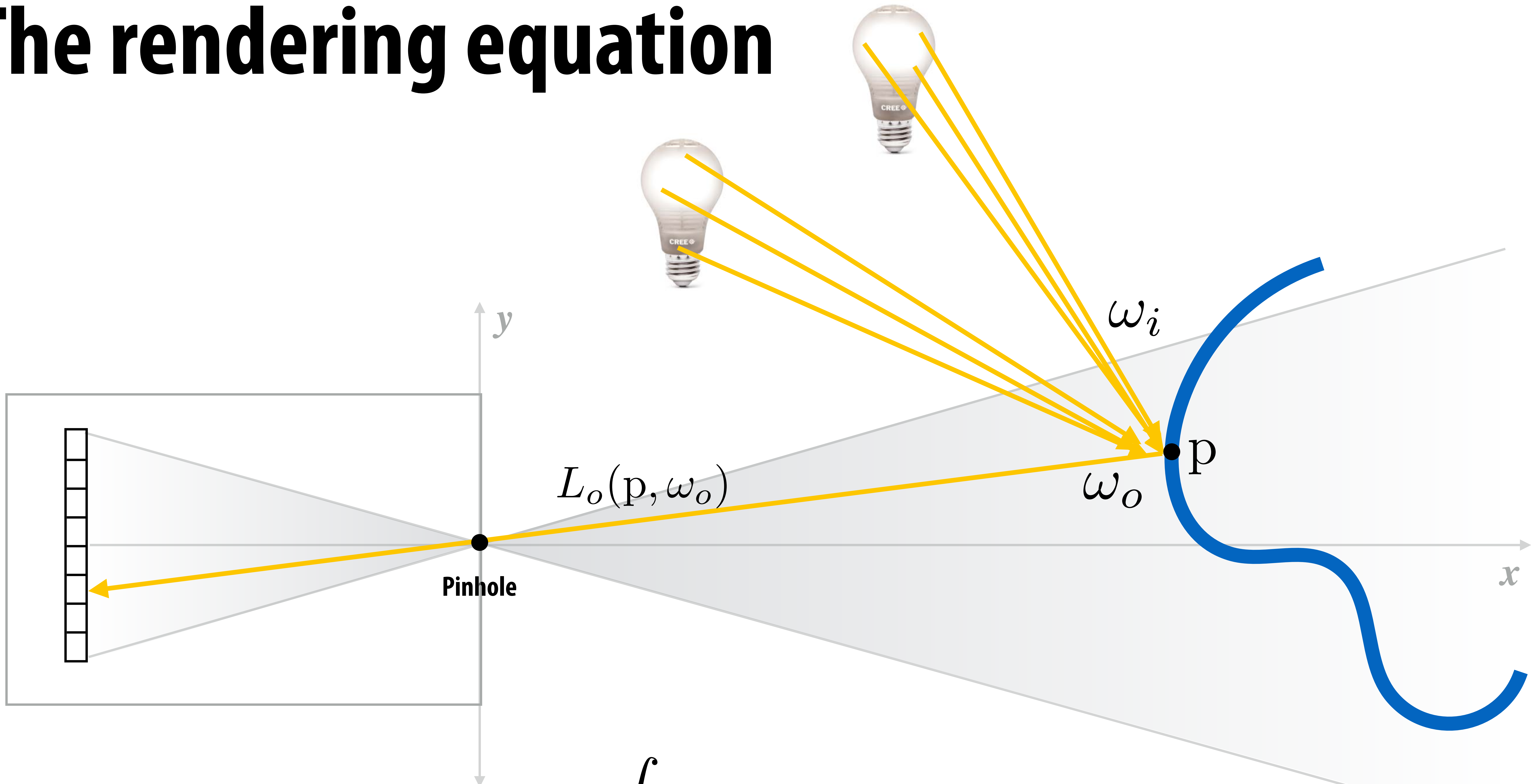
```
// Assume:
//   Ray ray hits surface at point hit_p
//   Normal of surface at hit point is hit_n

Vector3D wr = -ray.d;   // outgoing direction
Spectrum Lr = 0.;
for (int i = 0; i < N; ++i) {
    Vector3D wi;        // sample incident light from this direction
    float pdf;          // p(wi)

    generate_sample(brdf, &wi, &pdf);     // generate sample according to brdf

    Spectrum f = brdf->f(wr, wi);
    Spectrum Li = trace_ray(Ray(hit_p, wi));  // compute incoming Li
    Lr += f * Li * fabs(dot(wi, hit_n)) / pdf;
}
return Lr / N;
```

The rendering equation



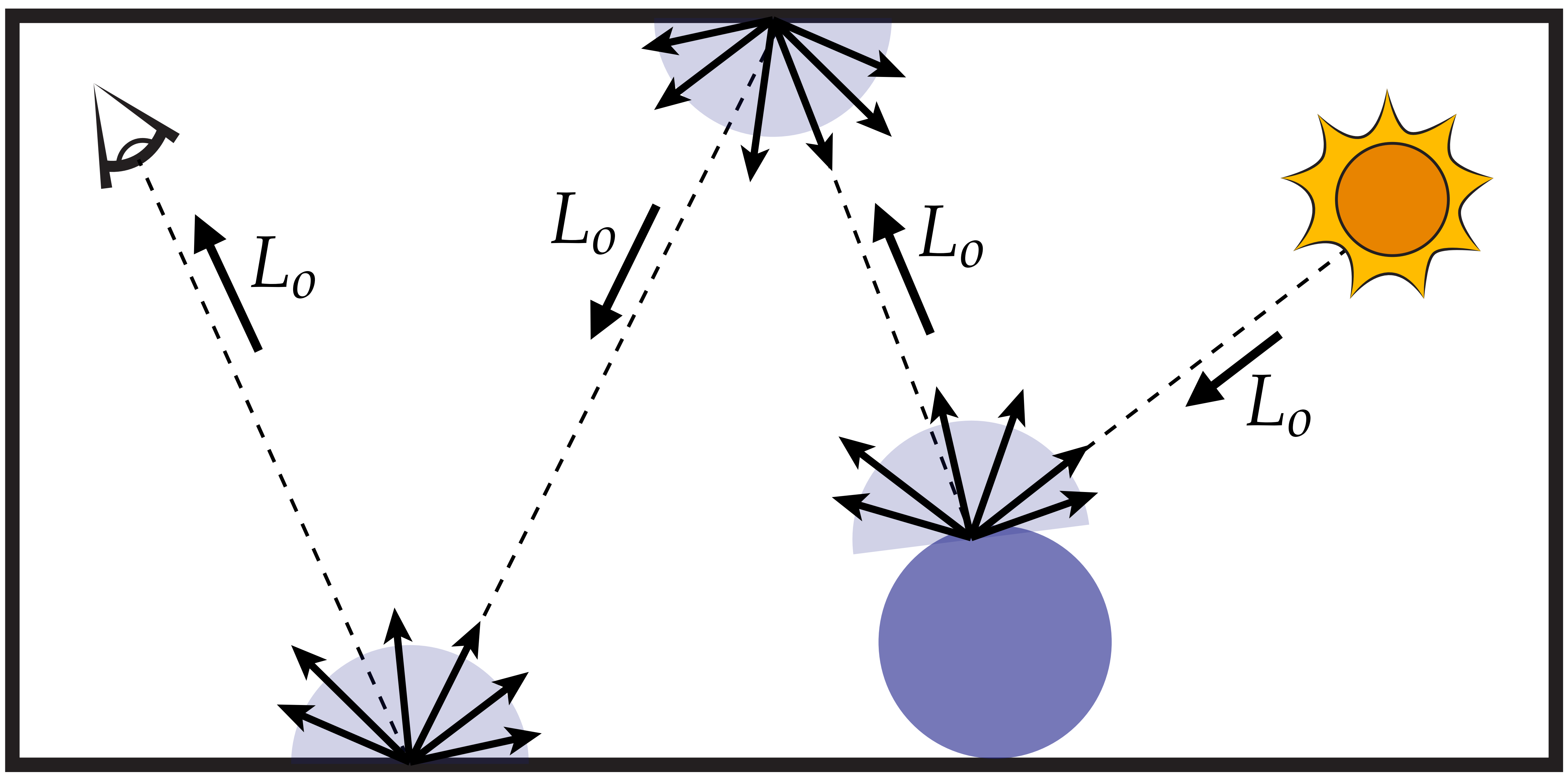
$$L_o(\mathrm{p}, \omega_o) = L_e(\mathrm{p}, \omega_o) + \int_{H^2} f_r(\mathrm{p}, \omega_i \to \omega_o)\, L_i(\mathrm{p}, \omega_i)\, \cos\theta_i \, \mathrm{d}\omega_i$$

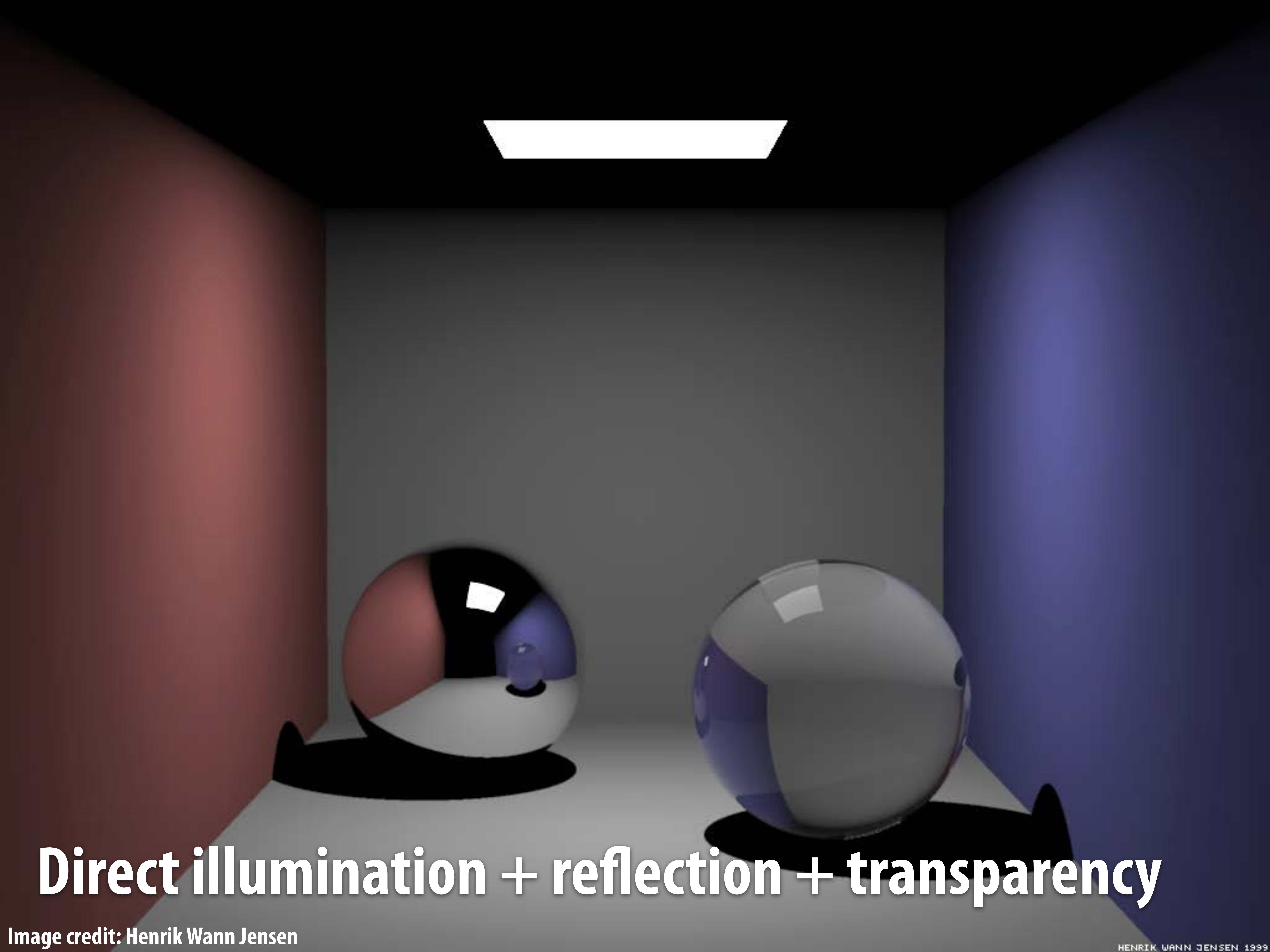
Now that we know how to handle reflection, how do we solve the full rendering equation? Have to determine incident radiance…

Key idea in (efficient) rendering: take advantage of special knowledge to break up integration into "easier" components.

Path tracing: overview

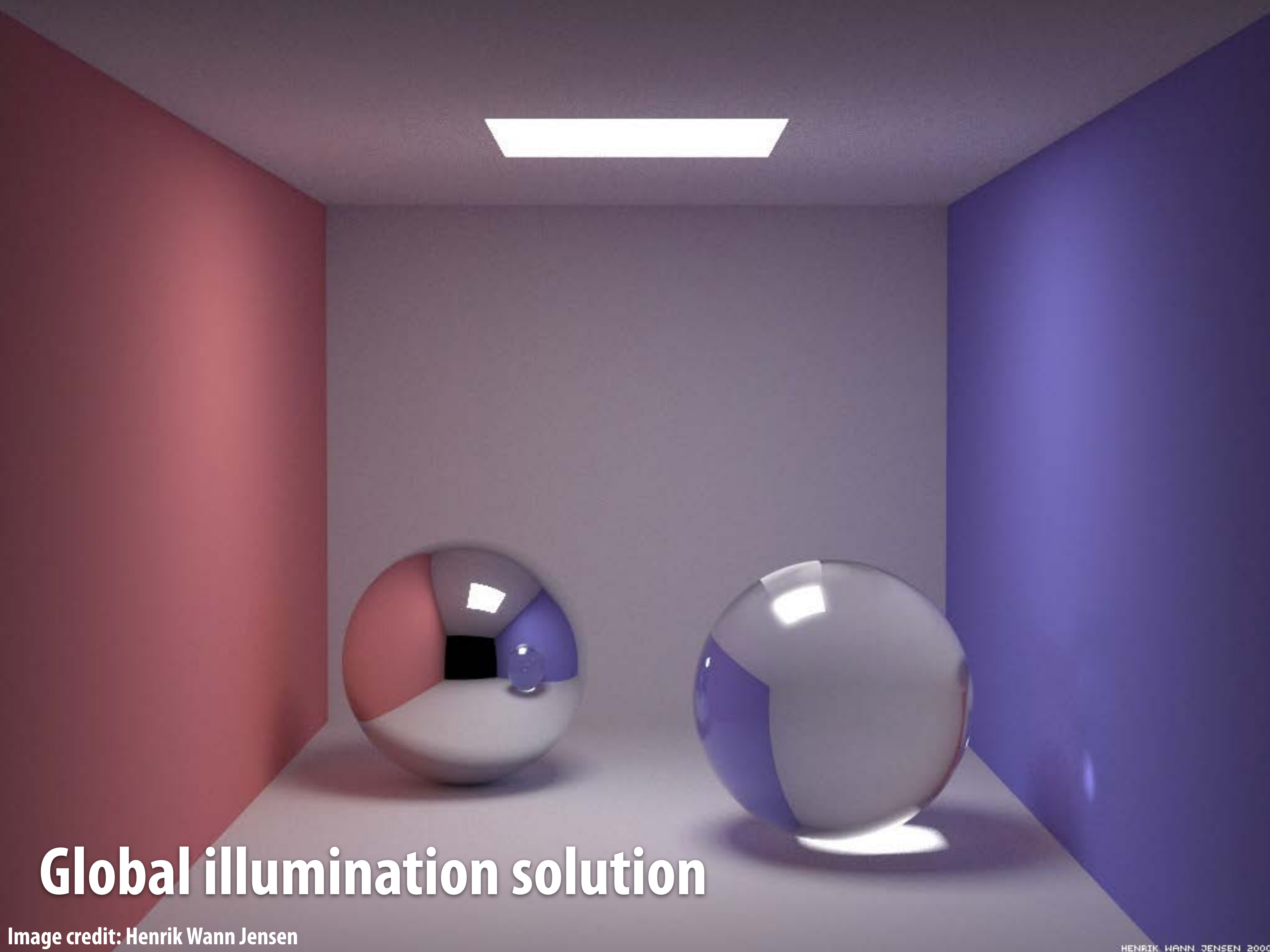
- Partition the rendering equation into direct and indirect illumination
- Use *Monte Carlo* to estimate each partition separately
  - One sample for each
  - Assumption: 100s of samples per pixel
- Terminate paths with *Russian roulette*





Direct illumination + reflection + transparency

Image credit: Henrik Wann Jensen



Global illumination solution

Image credit: Henrik Wann Jensen

Next Time: Monte Carlo integration

$$\int_\Omega f(p)\ dp \quad \approx \quad \mathrm{vol}(\Omega)\frac{1}{N}\sum_{i=1}^{N} f(X_i)$$