# **03** Overview of Shading for Real Time

Steve Marschner

**CS5625** Spring 2022

1. Light reflection physics
2. The traditional basics
3. Modern shading basics

# Light reflection physics

# Sources of light

**Point sources**

- light from points **in** the local scene; can be directionally varying—spotlights

**Area sources**

- light from geometry **in** the local scene; can be spatially varying—stained glass windows

**Directional sources**

- light from points **far outside** the local scene; aka. from fixed directions—e.g. sun

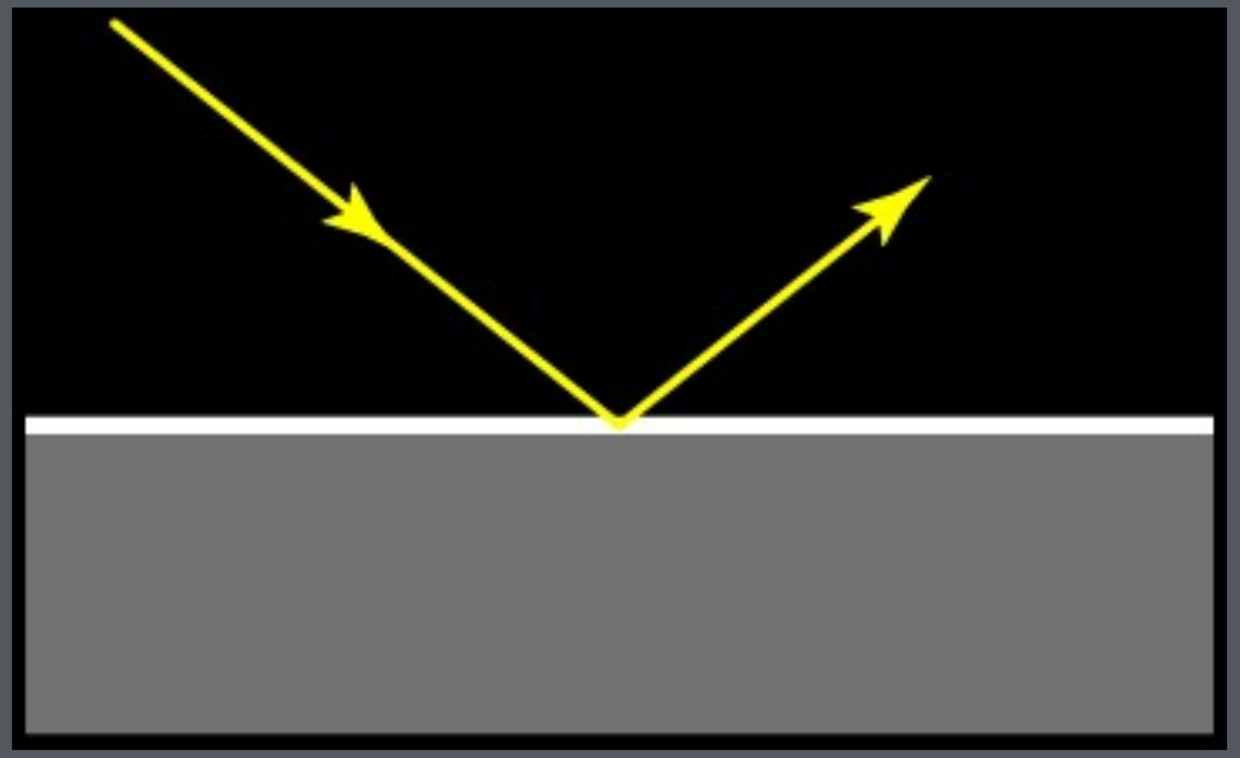
**Environment lighting**

- light from everything **far outside** the local scene—e.g. env. maps, sun-sky models

**Indirect lighting**

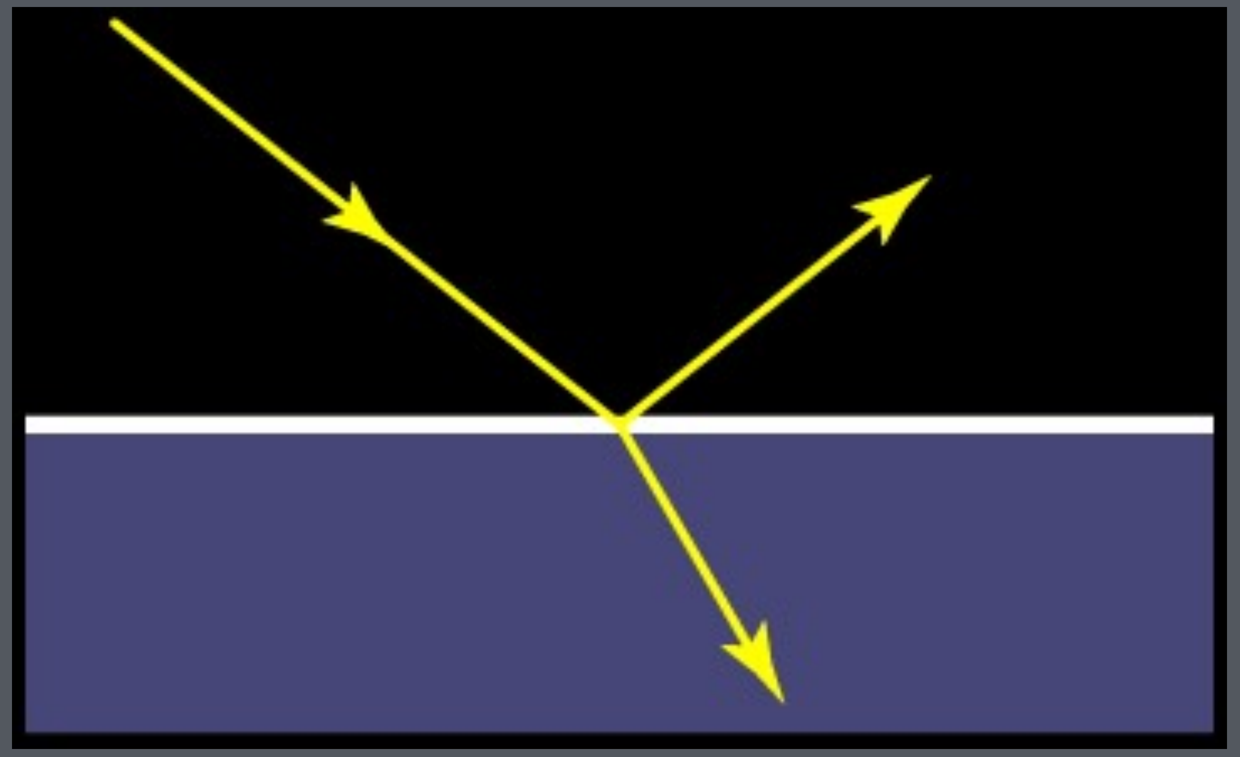
- light reflected from other surfaces—e.g. lighting from torchiere lamp

# Simple kinds of scattering



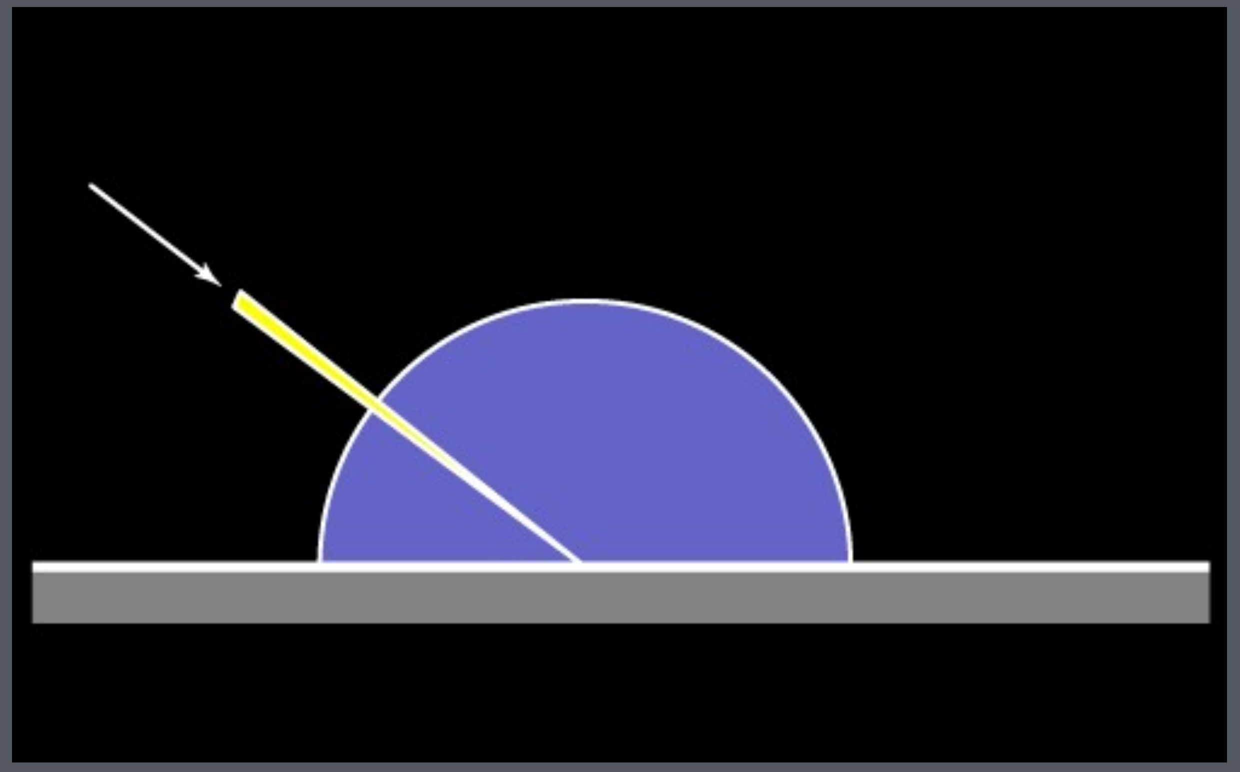
**Ideal specular reflection**

- incoming ray reflected to a single direction
- mirror-like behavior
- arises at smooth surfaces



**Ideal specular transmission**

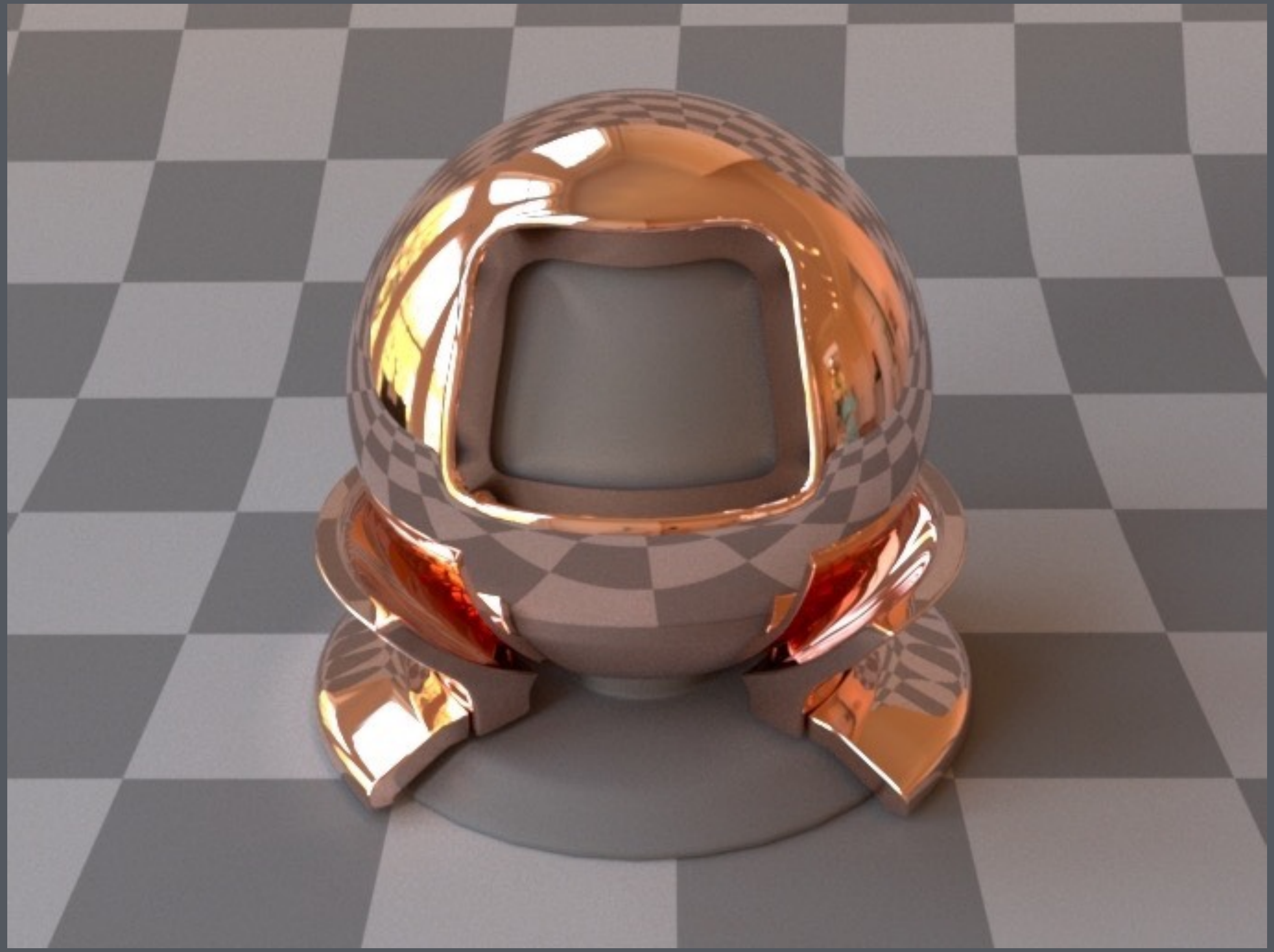
- incoming ray refracted to a single direction
- glass-like behavior
- arises at smooth dielectric (nonmetal) surfaces



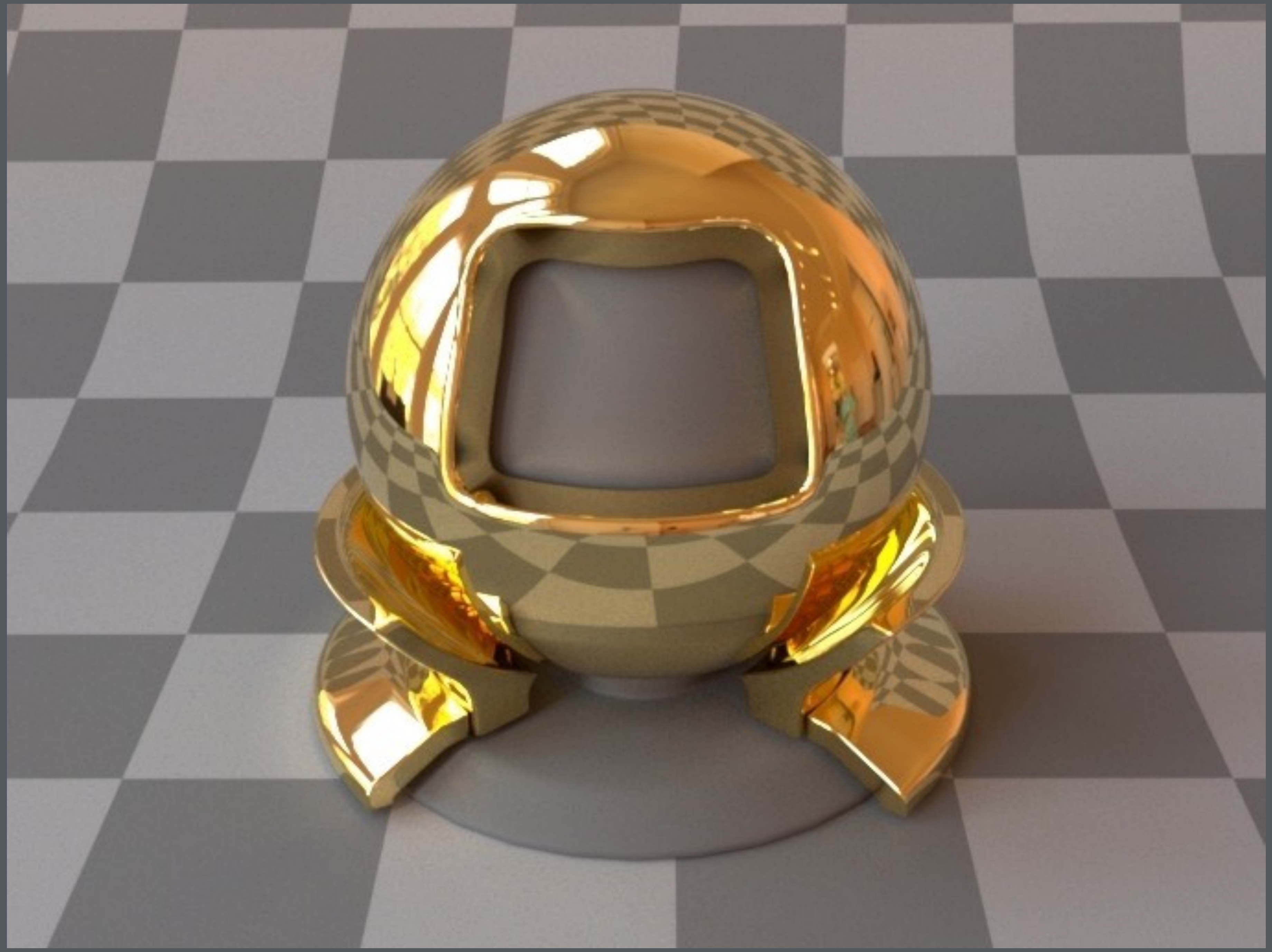
**Ideal diffuse reflection or transmission**

- outgoing radiance independent of direction
- arises from subsurface multiple scattering

# Ideal specular reflection from metals



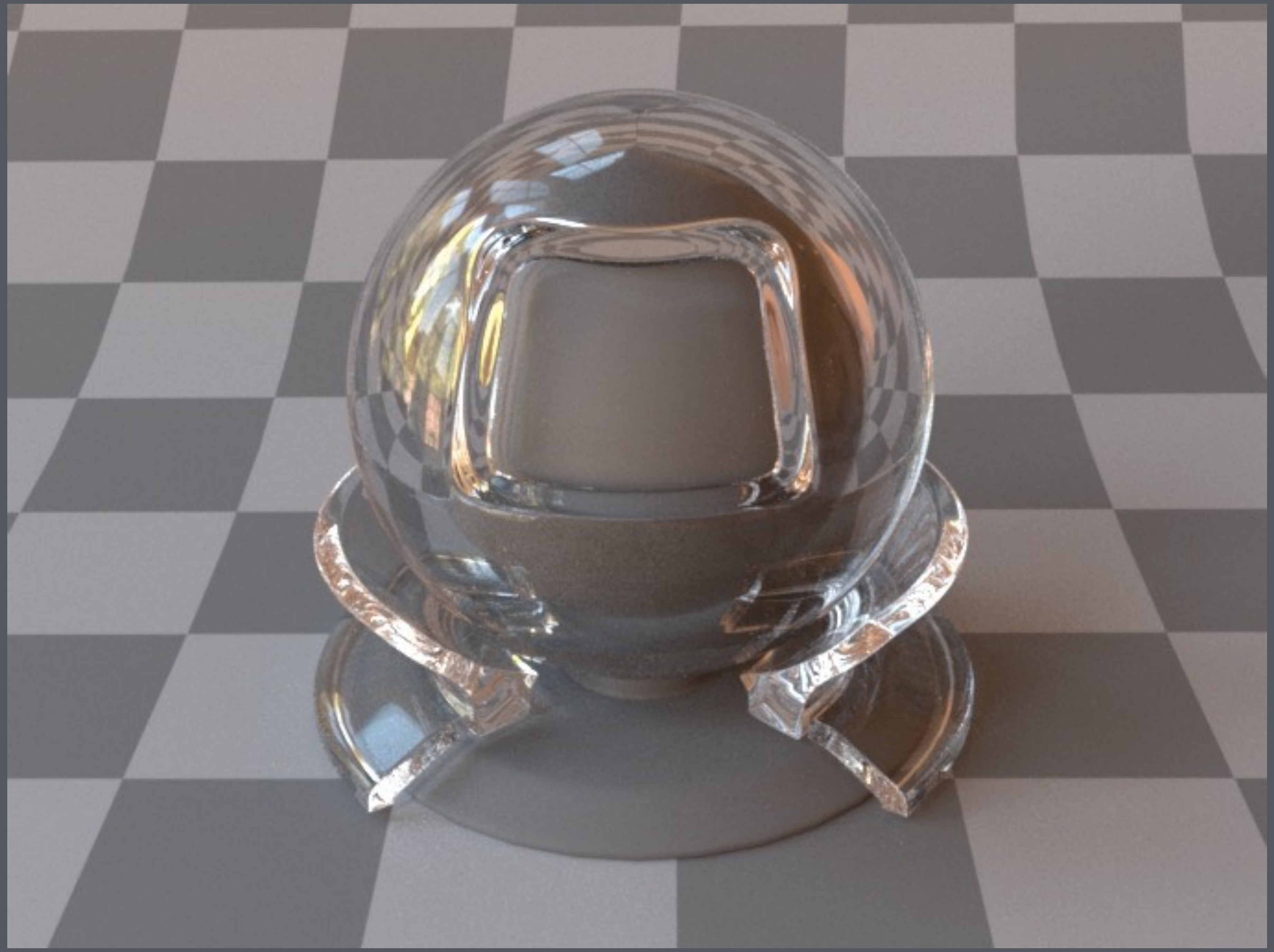
Cu



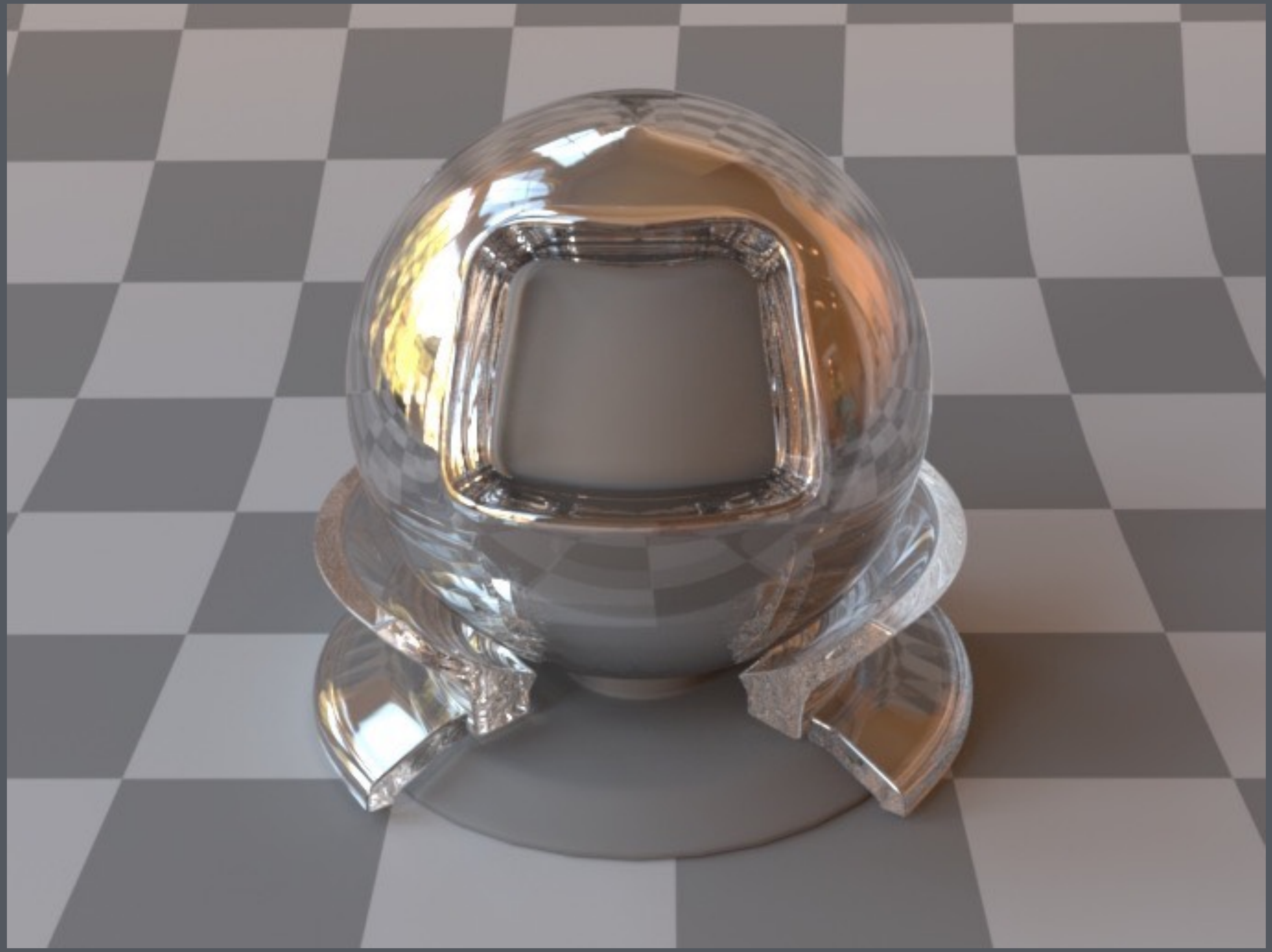
Au

Wenzel Jakob / Mistuba

# Ideal reflection and transmission from smooth dielectrics



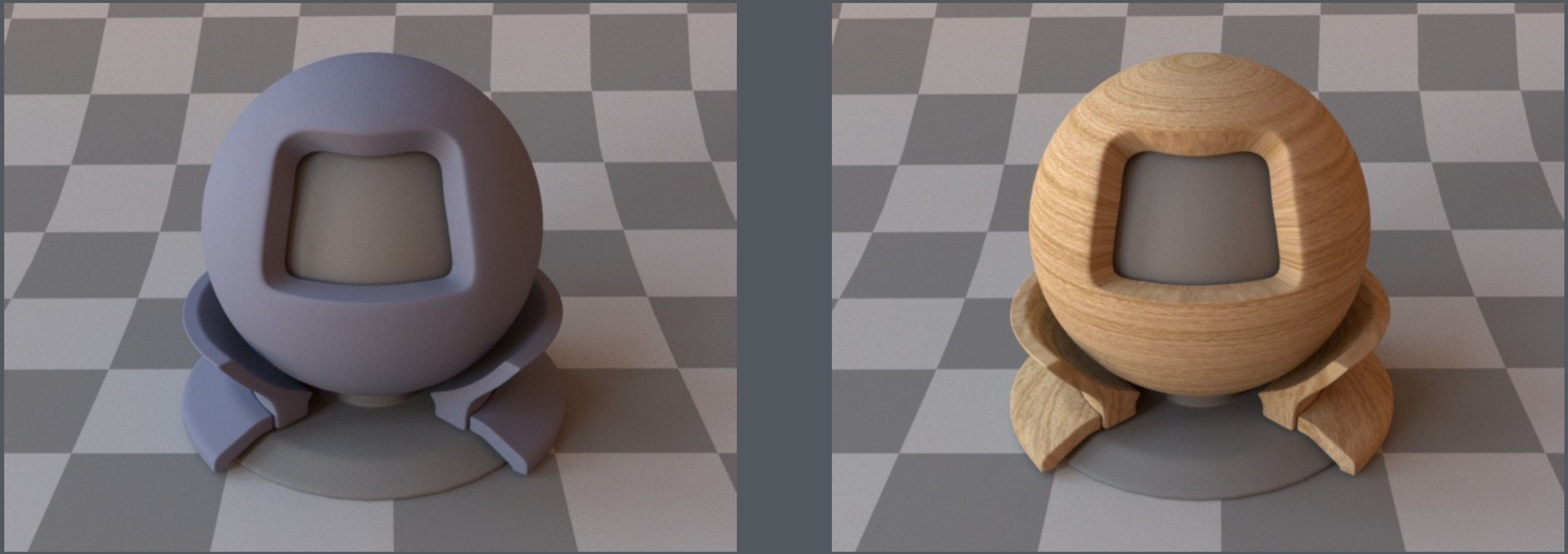
Water (ior = 1.33)



Diamond (ior = 2.4)

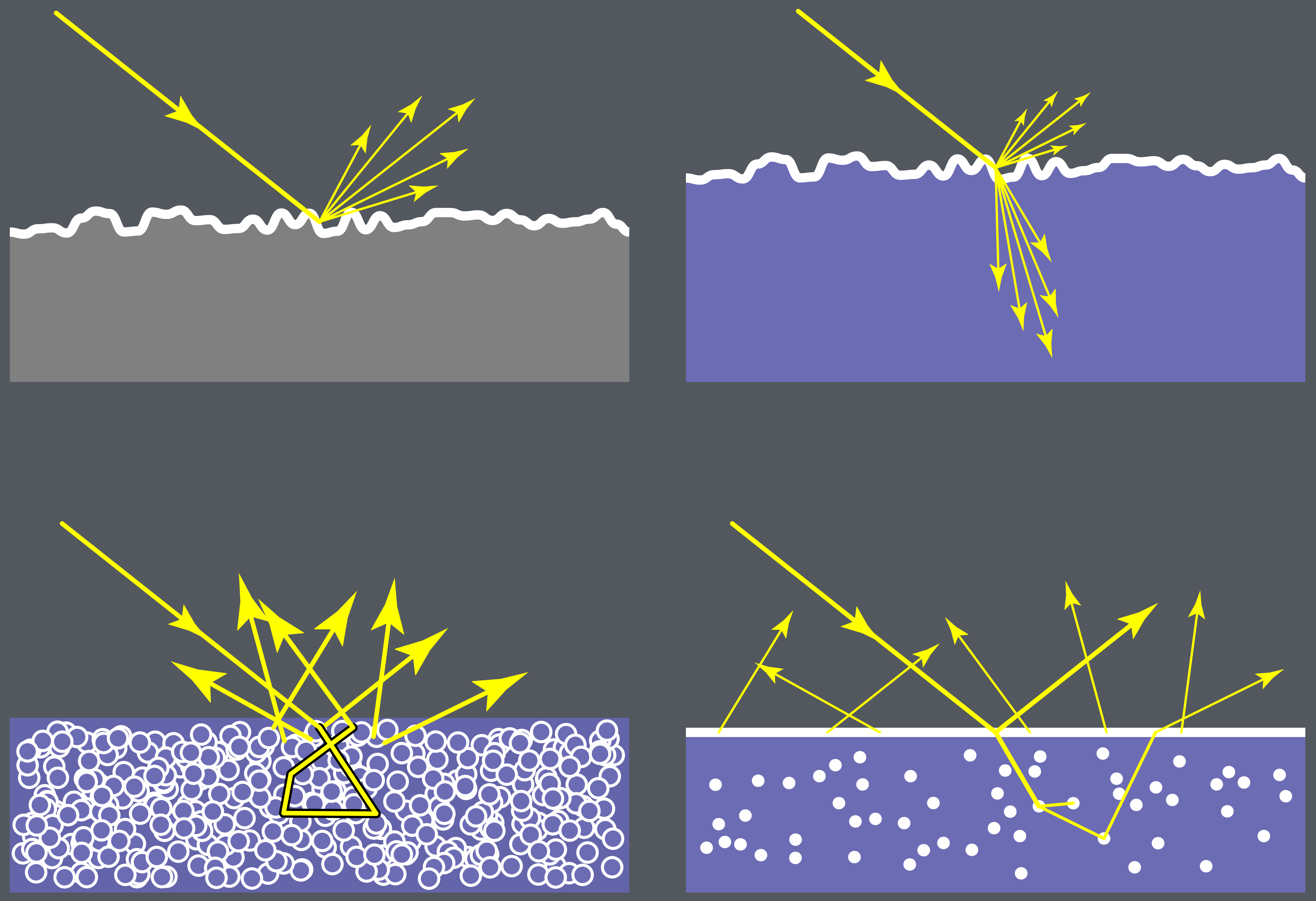
Wenzel Jakob / Mistuba

# Two diffuse surfaces



Wenzel Jakob / Mistuba

# More complex scattering



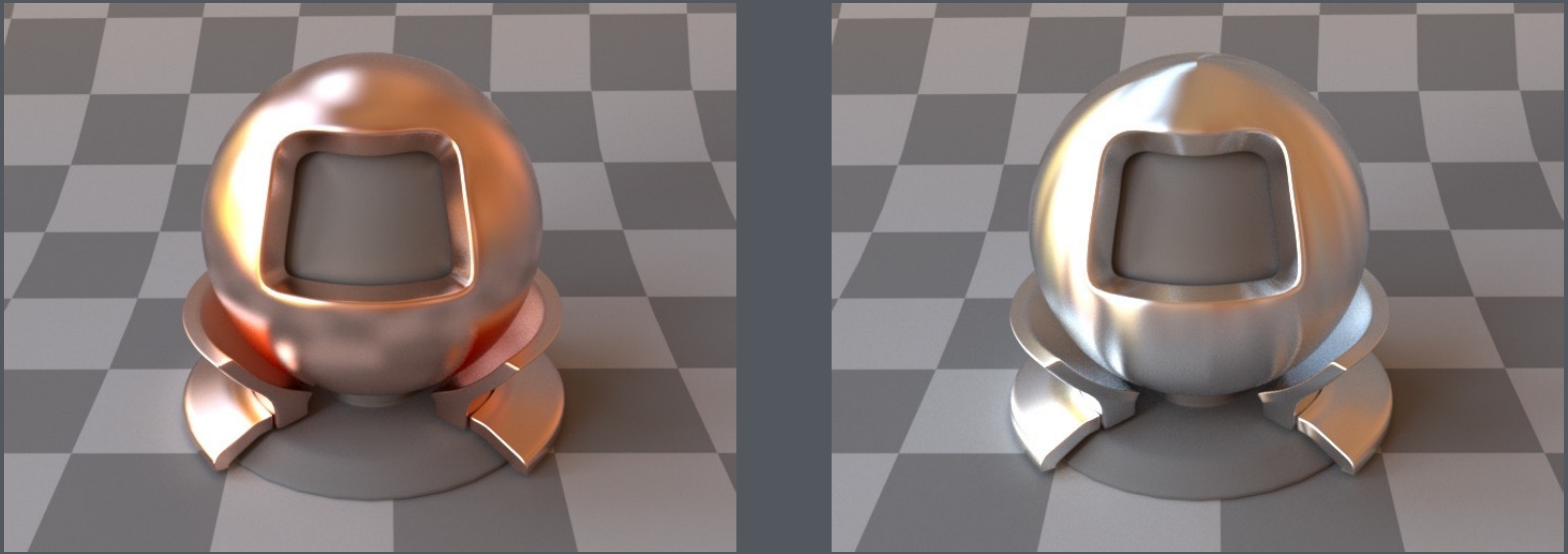
**Rough interfaces**

- metal interfaces: blurred reflection
- dielectric interfaces: blurred transmission

**Subsurface scattering**

- liquids—milk, juice, beer, …
- coatings—paint, glaze, varnish, …
- natural materials—wood, marble, …
- biological materials—skin, plants, …
- low optical density leads to *translucency*

# Reflection from rough metal interfaces



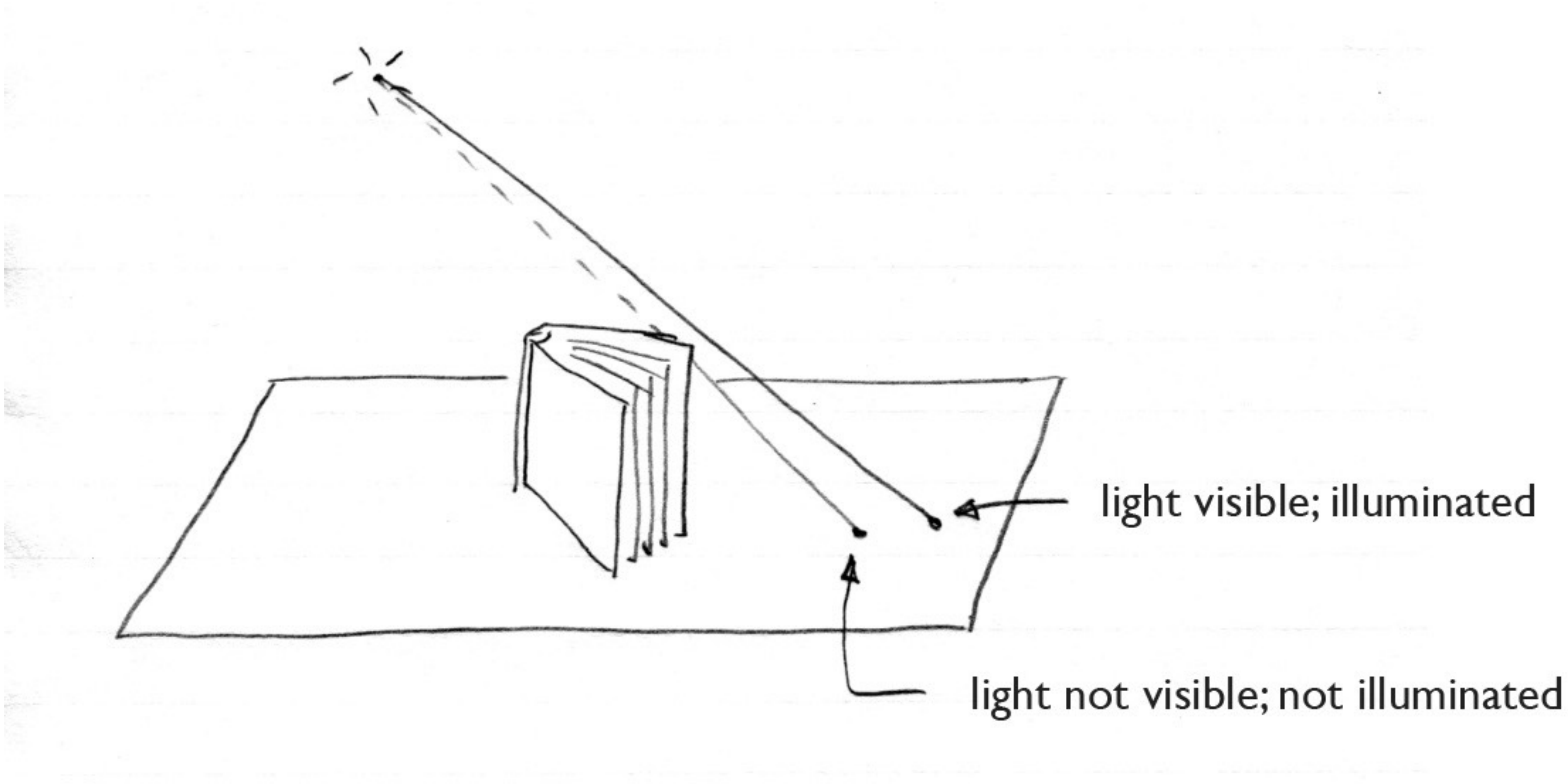
Cu ($\alpha = 0.1$)

Al (anisotropic)

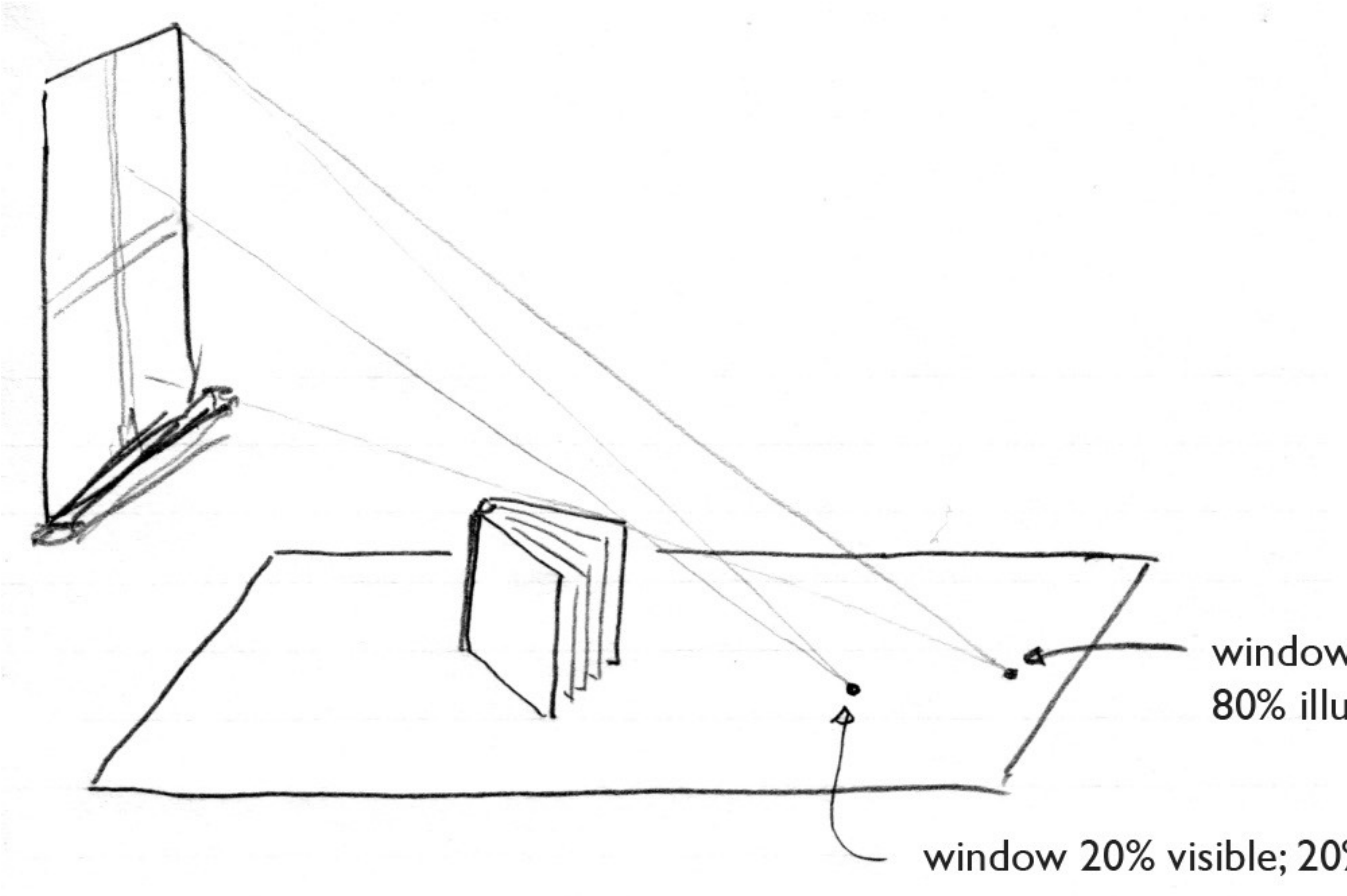
Wenzel Jakob / Mistuba

# Soft shadows

# Point lights cast hard shadows



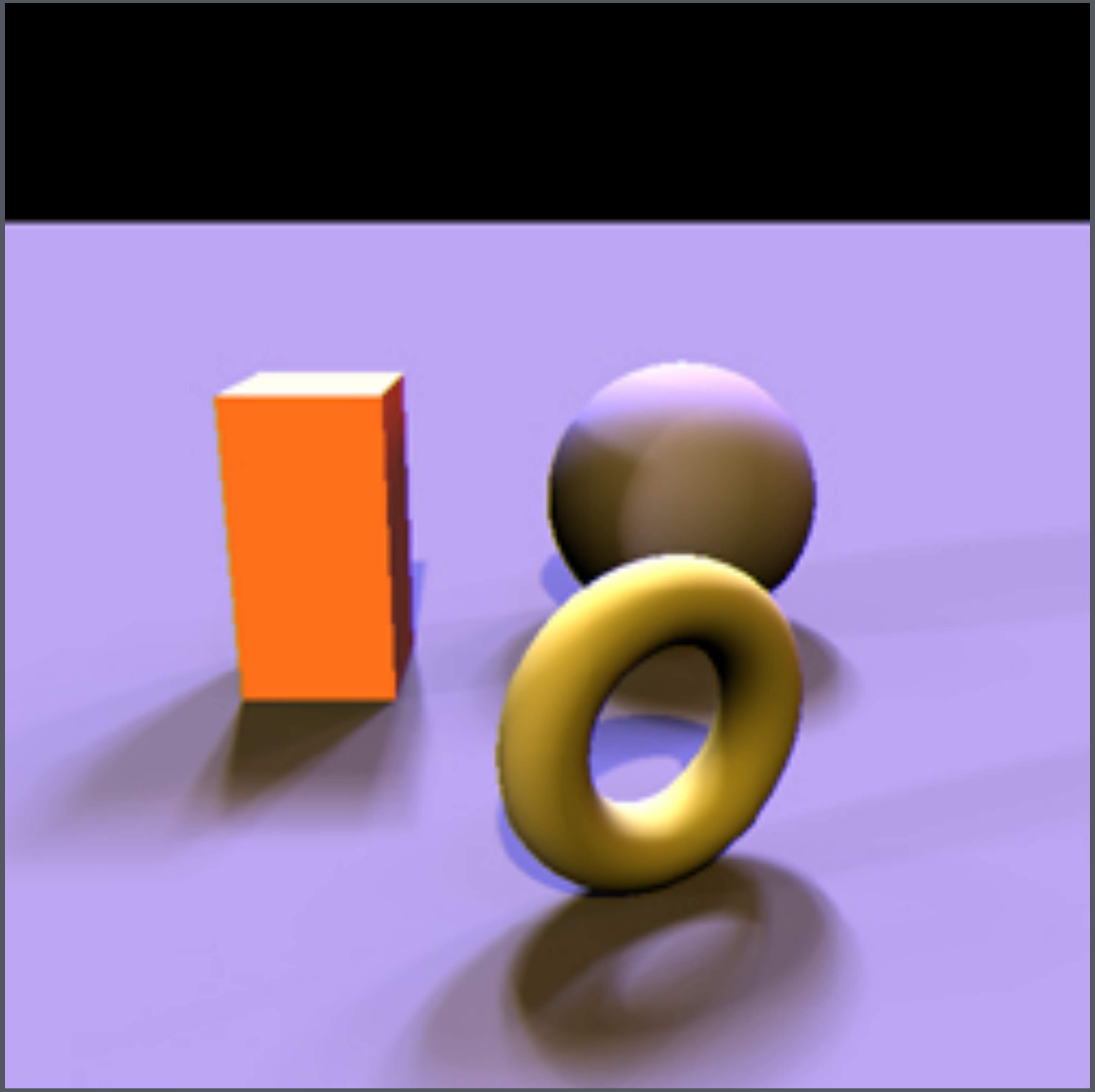
# Area lights cast soft shadows



window 80% visible;
80% illuminated

window 20% visible; 20% illuminated

# Shadows from environment lights

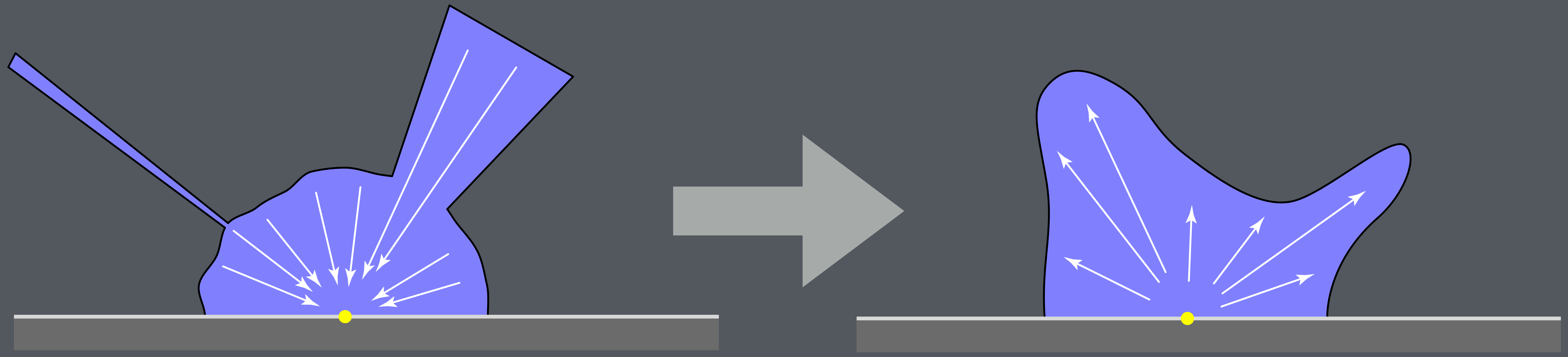


Ben-Artzi et al. / JGT 2004

# Light reflection: full picture

**all types of reflection reflect all types of illumination**

- diffuse, glossy, mirror reflection
- environment, area, point illumination



incident distribution
(function of direction)

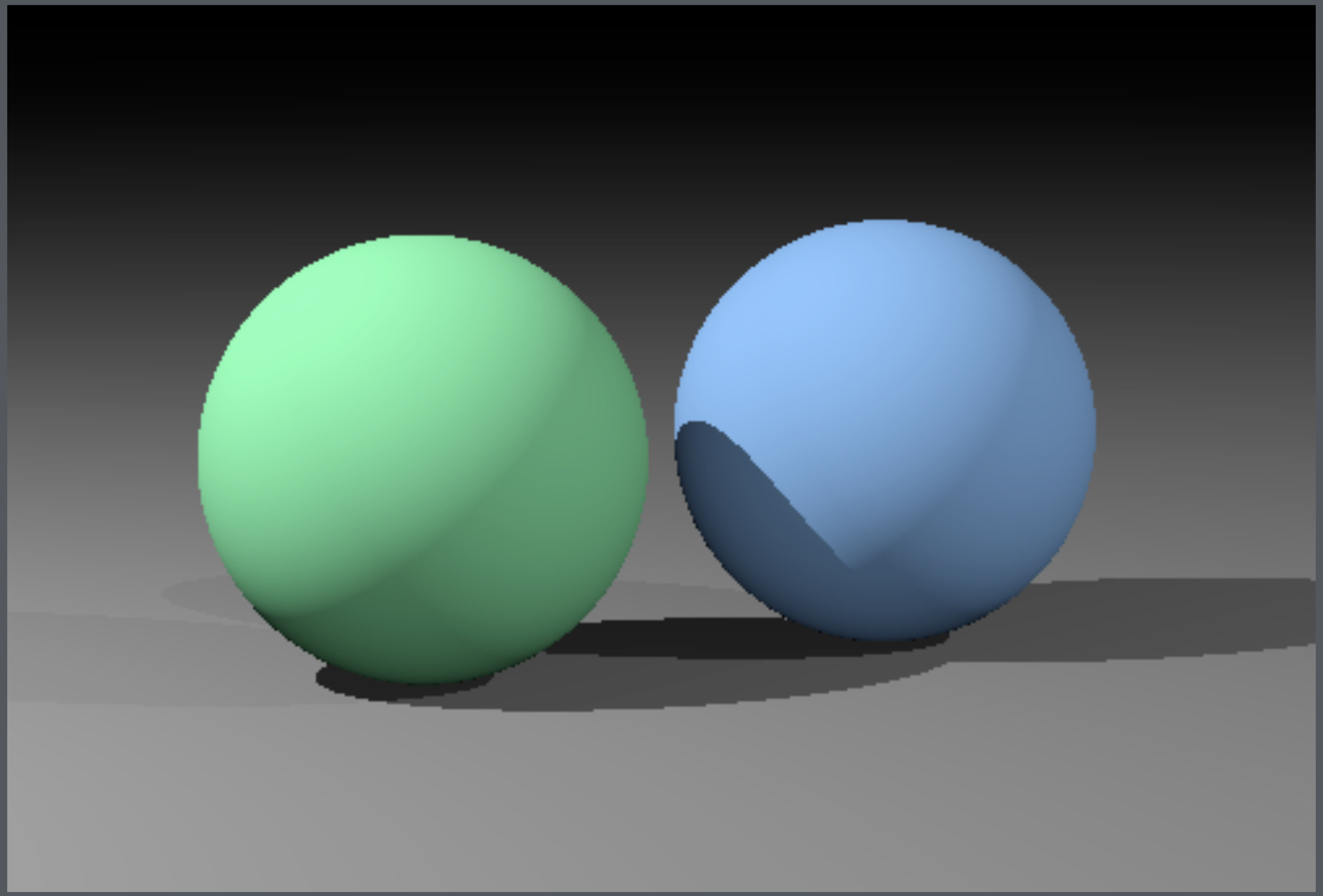
reflected distribution
(function of direction)

# Categories of illumination

| | diffuse | glossy | mirror |
|---|---|---|---|
| point/directional | hard shadows | simple specular highlight | point reflections |
| area | soft shadows | shaped specular highlight | reflected image of source |
| environment | soft shadows | blurry reflection of environment | reflected image of environment |
| indirect | soft indirect illumination | blurry reflections of other objects | reflected images of other objects |

# The traditional basics

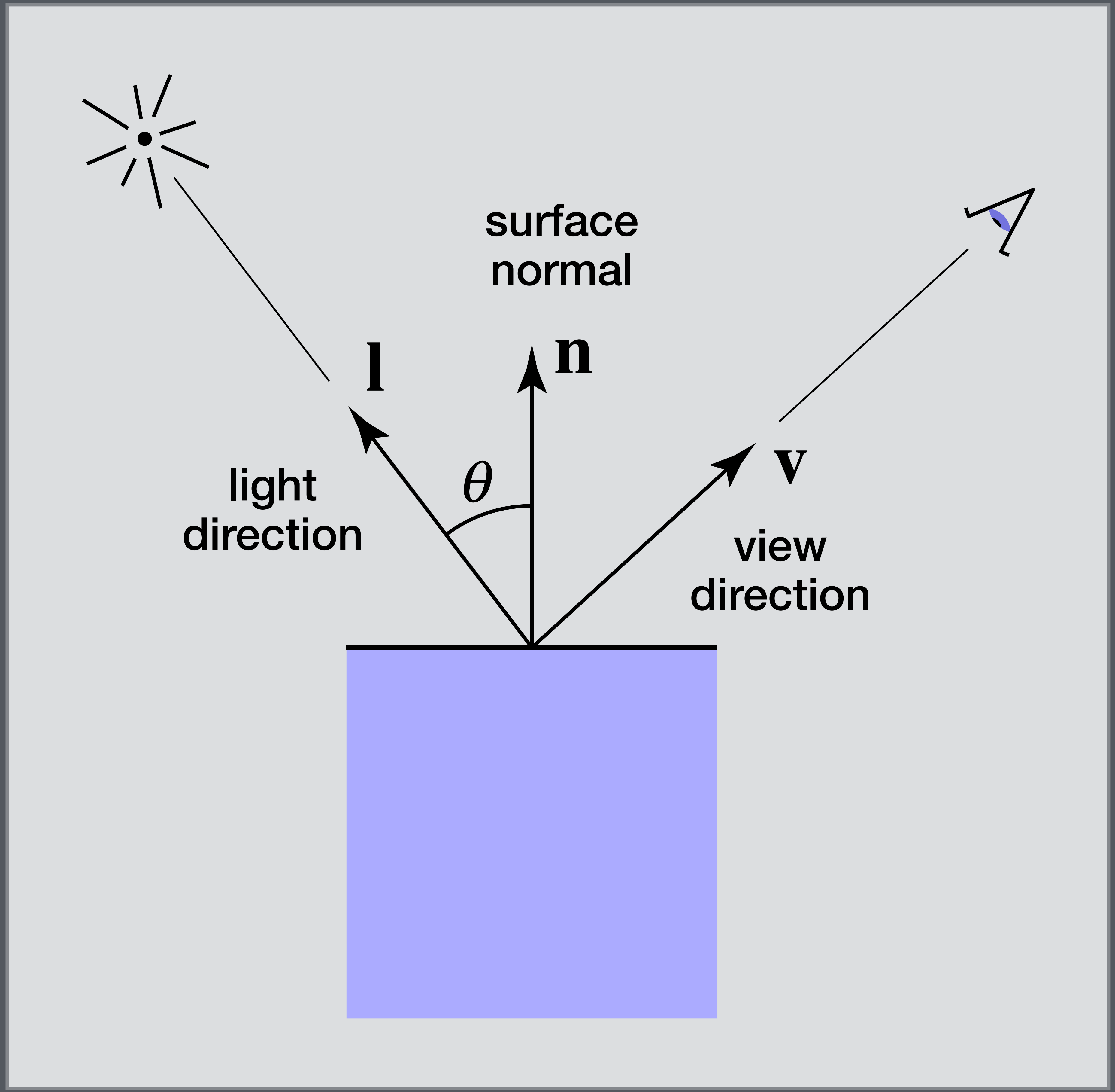
# Diffuse (Lambertian) shading, point light



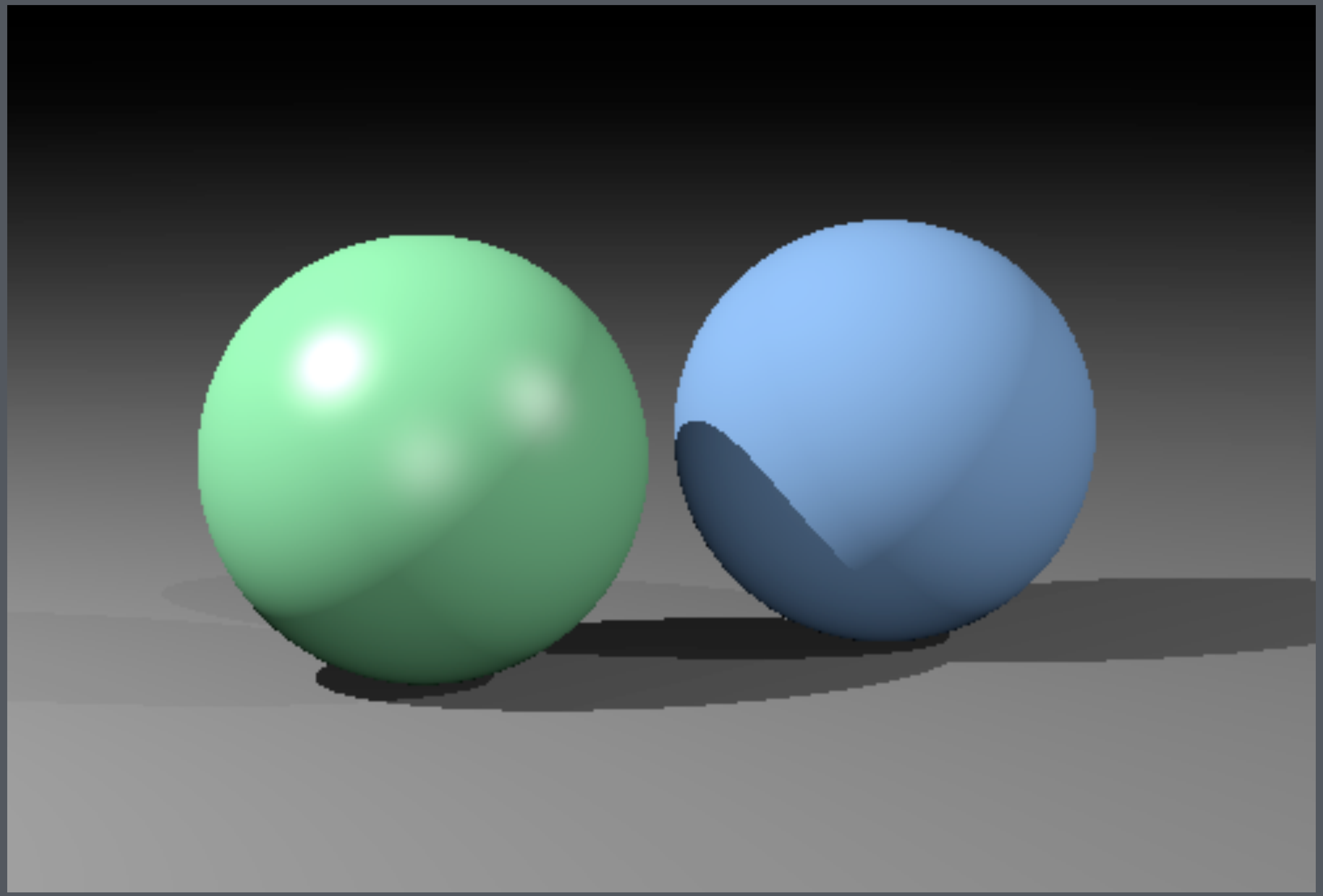
$$L = c_d \, c_l \, \cos\theta$$

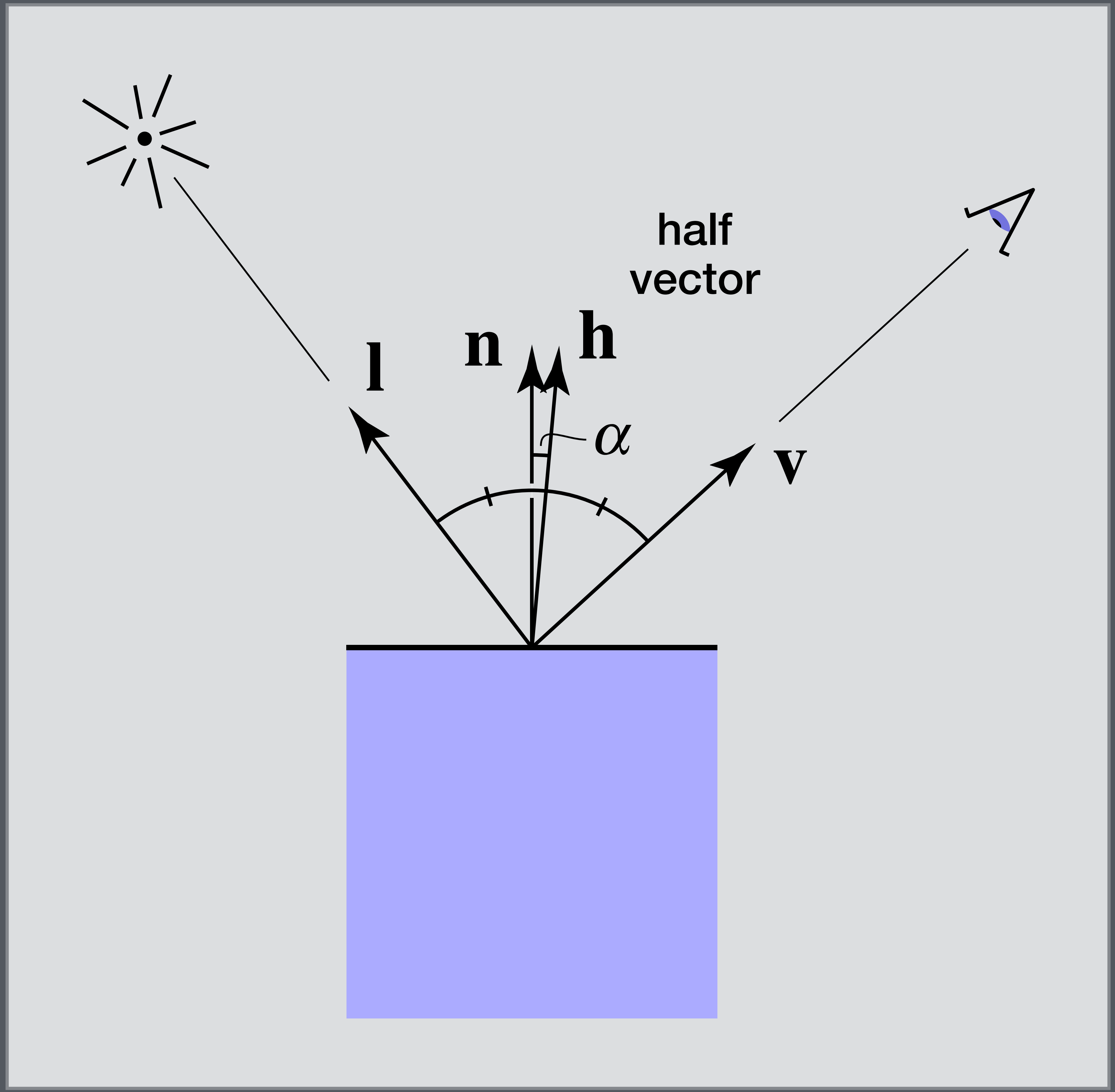
surface's diffuse color

light color



# Diffuse + specular (Phong) shading, point light



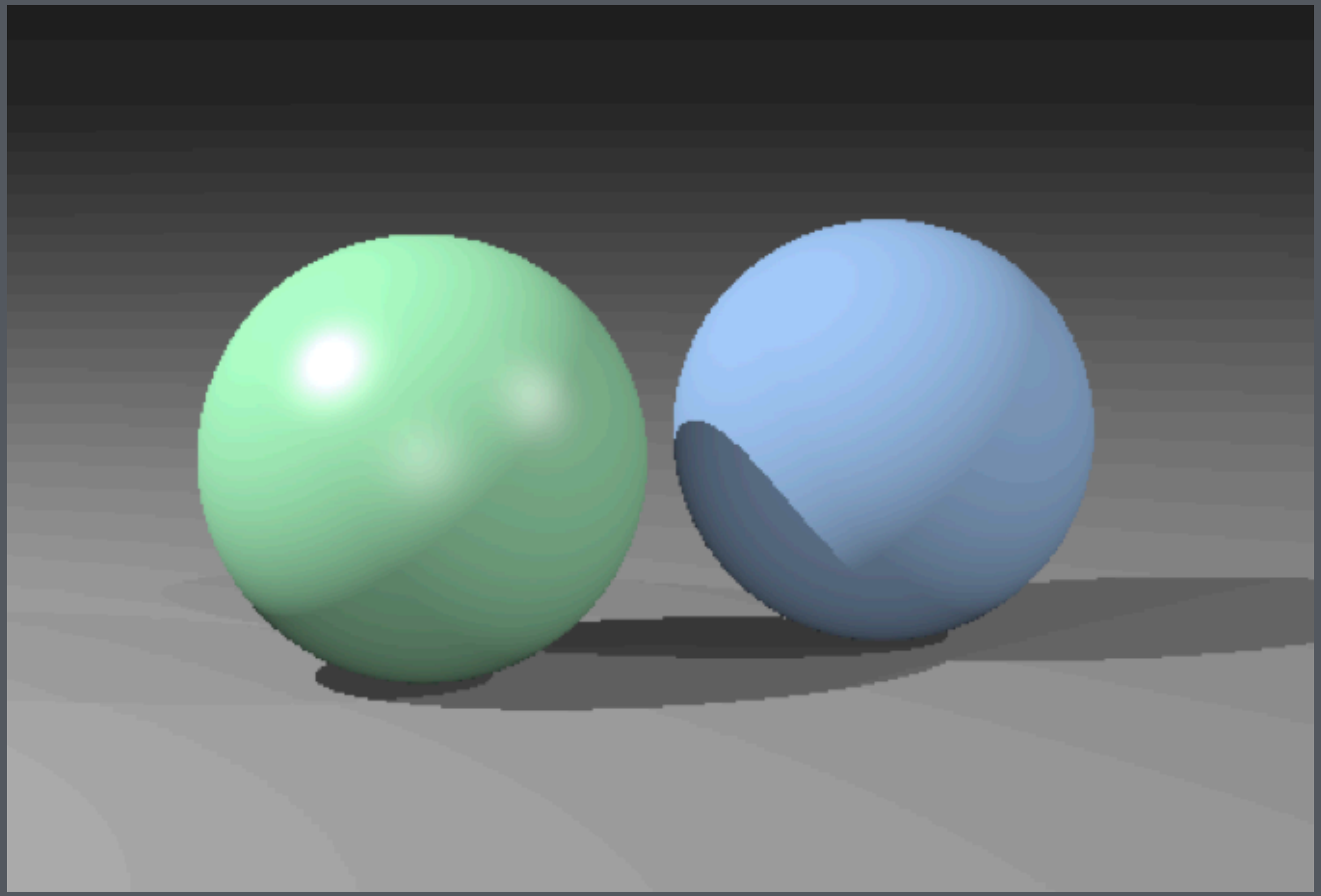


$$L = c_d\, c_l\ \cos\theta$$
$$+\, c_s\, c_l\, (\cos\alpha)^p$$

surface's specular color

specular exponent (shininess)

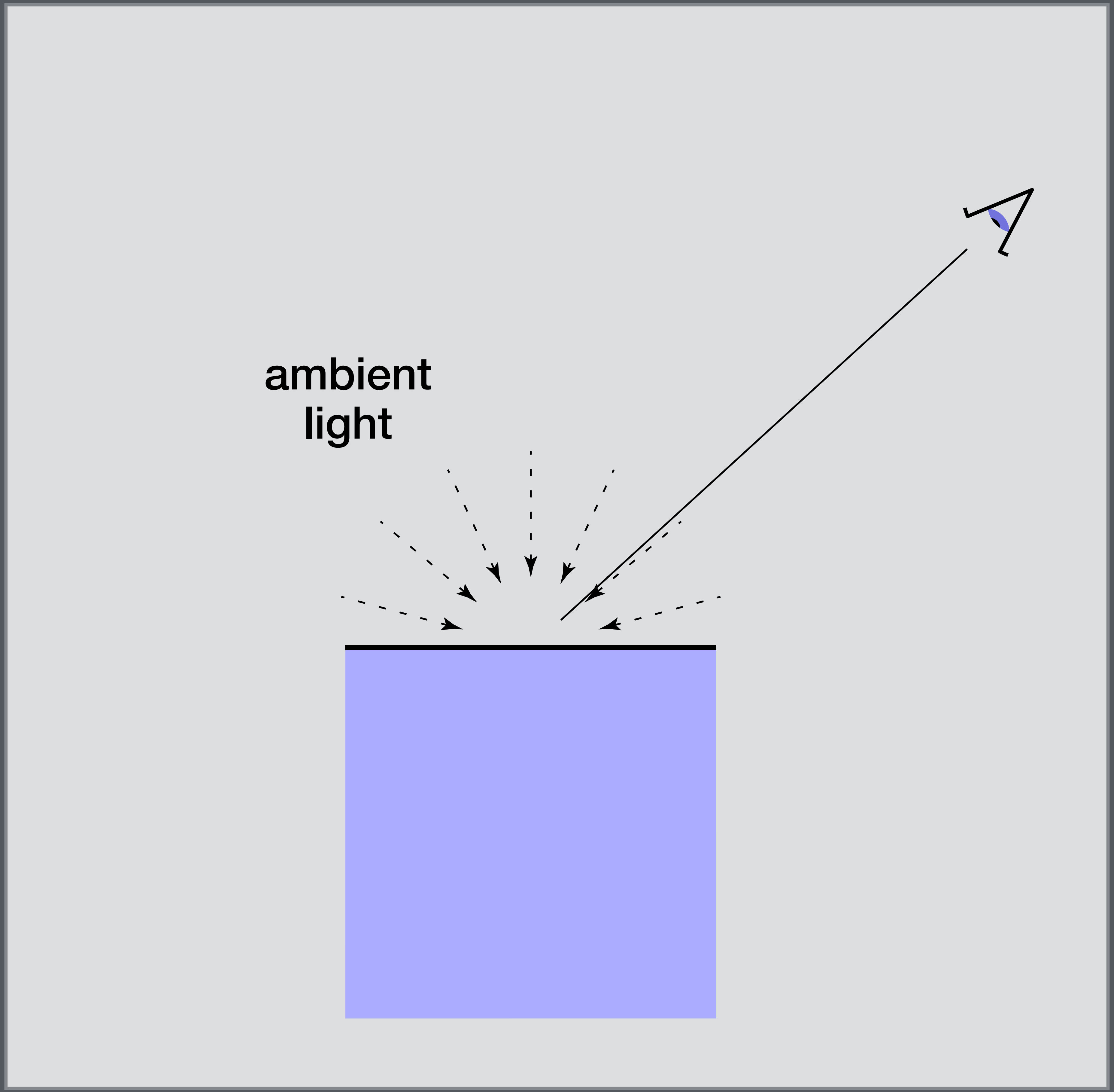
# Diffuse + specular + ambient shading



$$L = c_d\, c_l\, \cos\theta + c_s\, c_l\, (\cos\alpha)^p + c_d\, c_a$$

surface's diffuse color

ambient light color

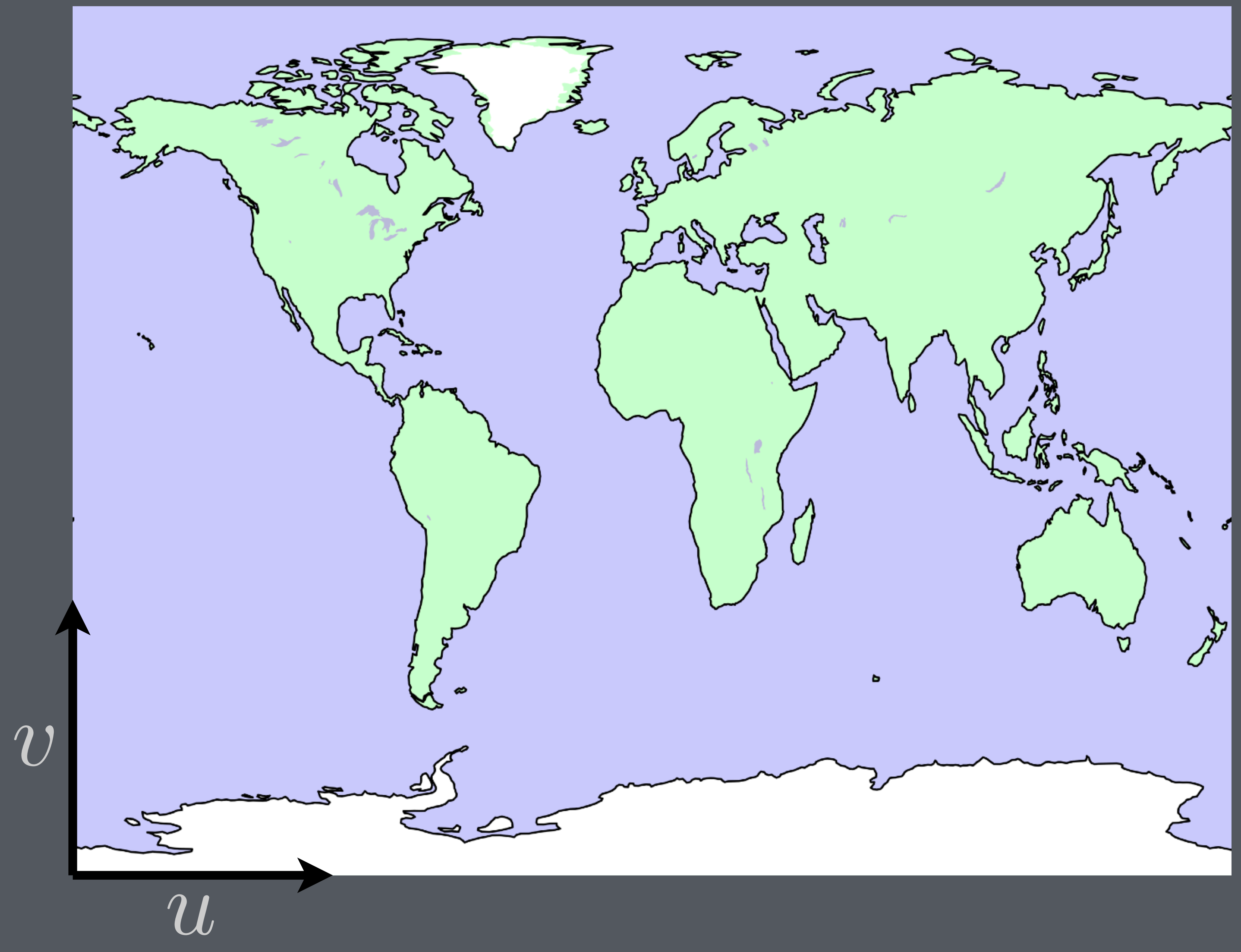


# Texture mapping the diffuse color

$$c_d = c_d(\underbrace{u, v})$$

surface's diffuse color

coordinates on surface

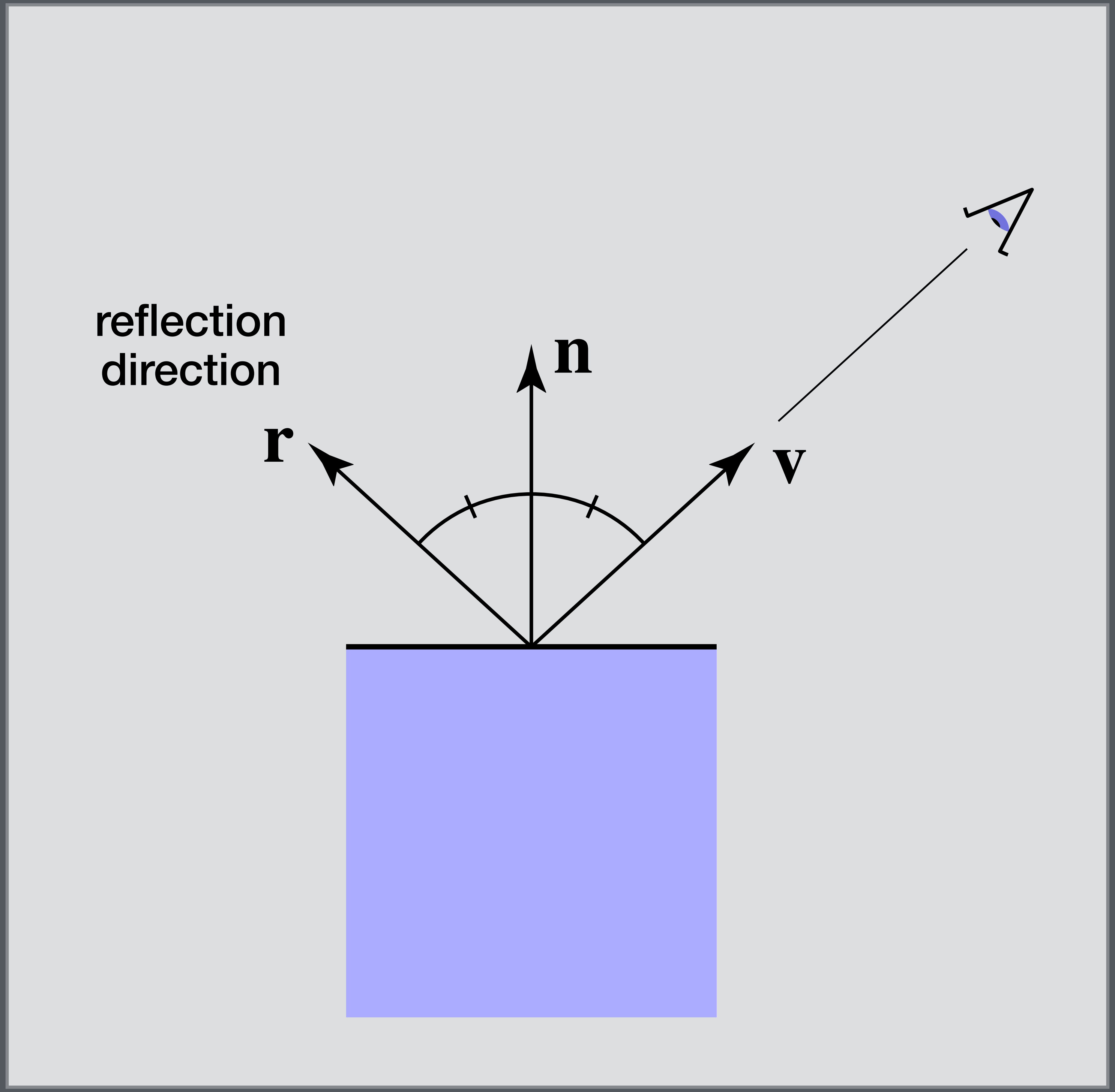


# Mirror reflection: texture map the reflected light

$$L = c_m \, L_i(\mathbf{r})$$

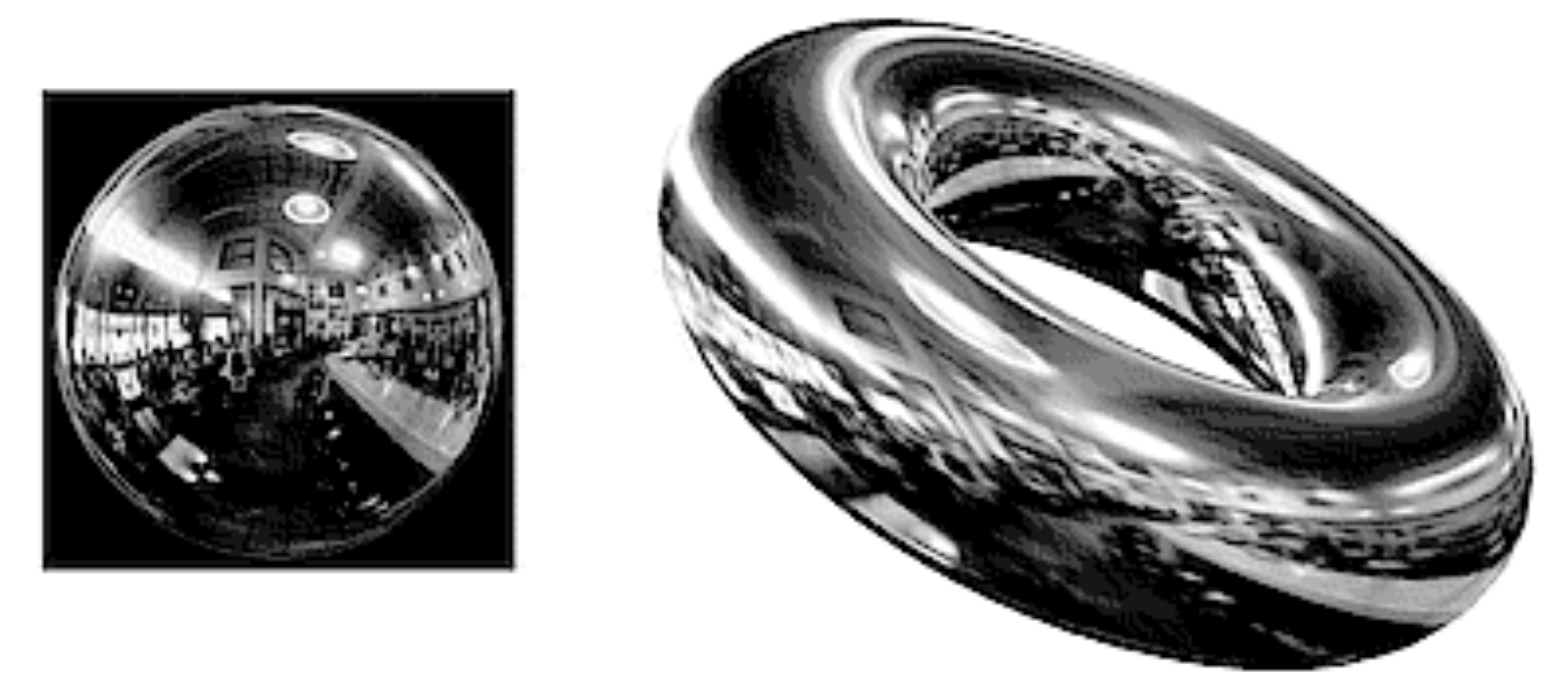
surface's
mirror
color

incoming
light in
mirror direction



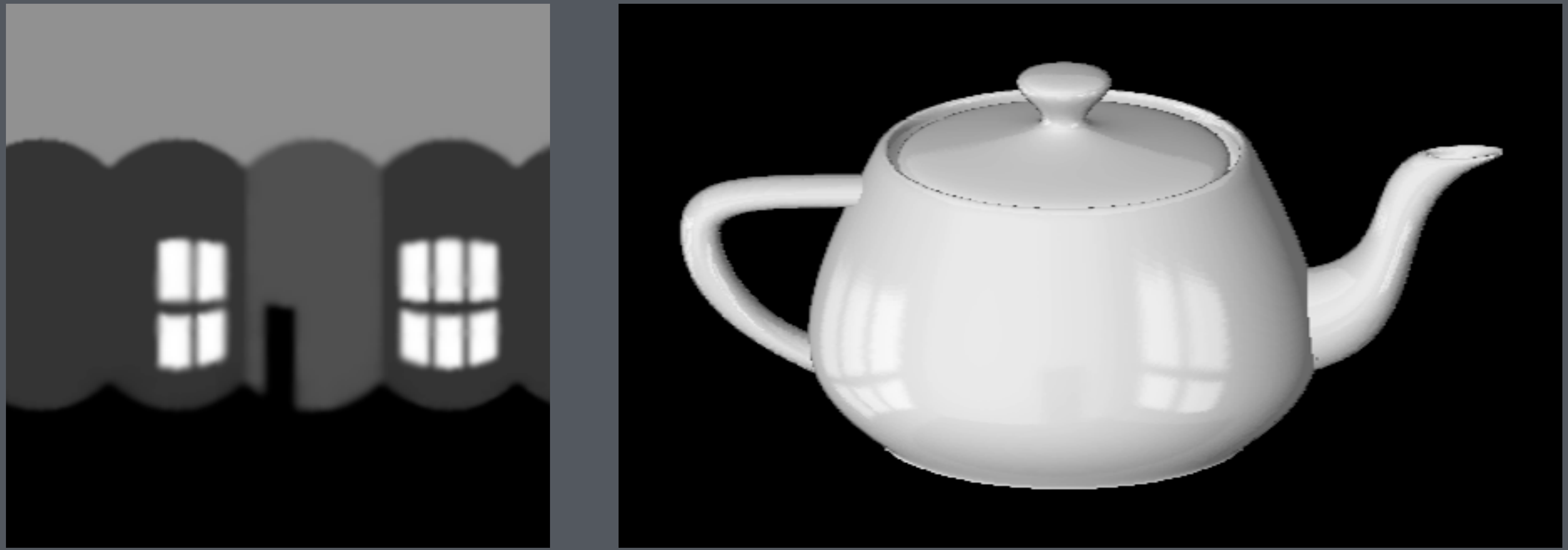
*Hand with Reflecting Sphere*. M. C. Escher, 1935. lithograph

# Reflection mapping with environment maps



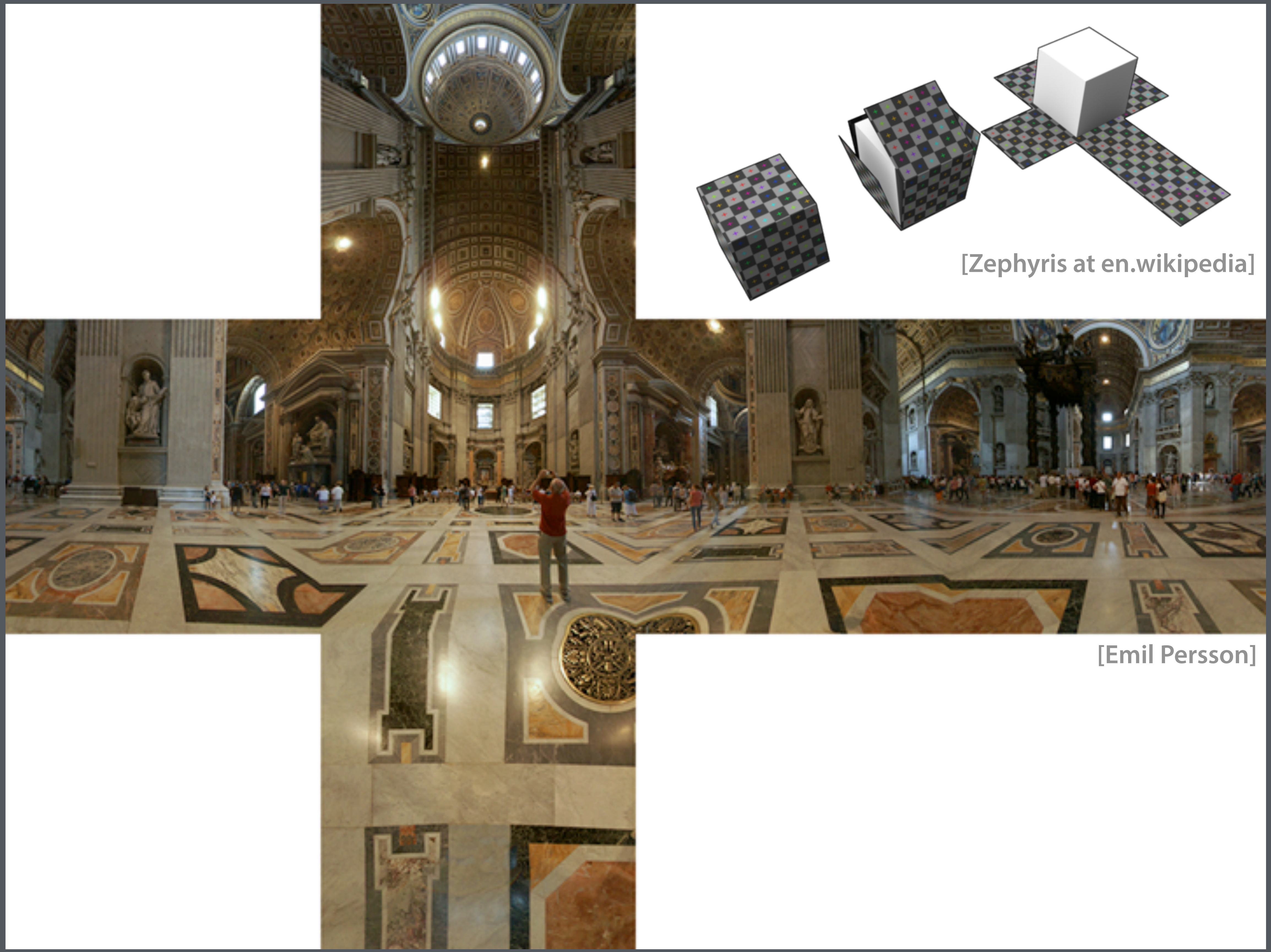
Haeberli and Segal, 1993

mirror-ball format



Blinn & Newell 1976

latitude-longitude (spherical) format

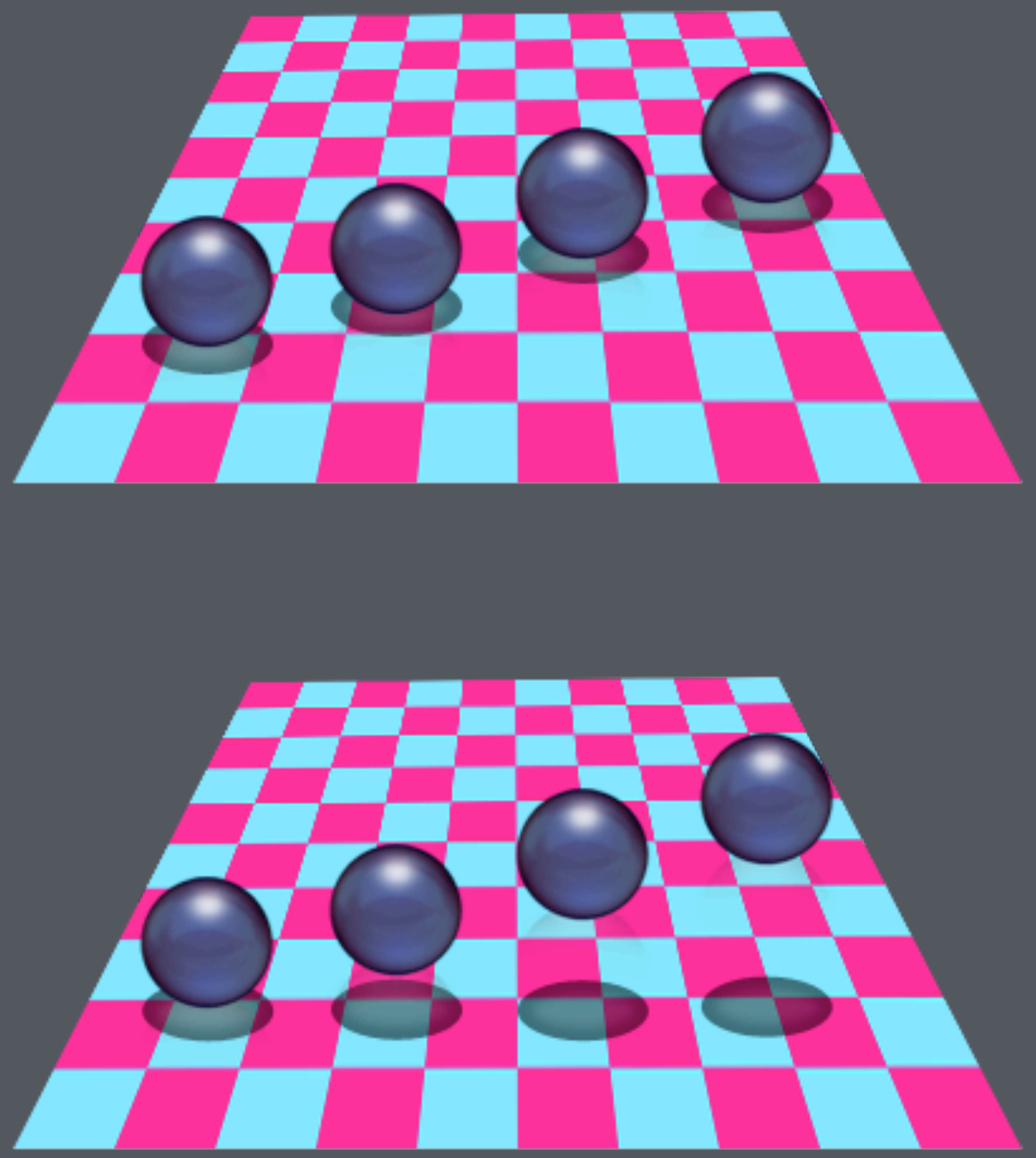


[Zephyris at en.wikipedia]

[Emil Persson]

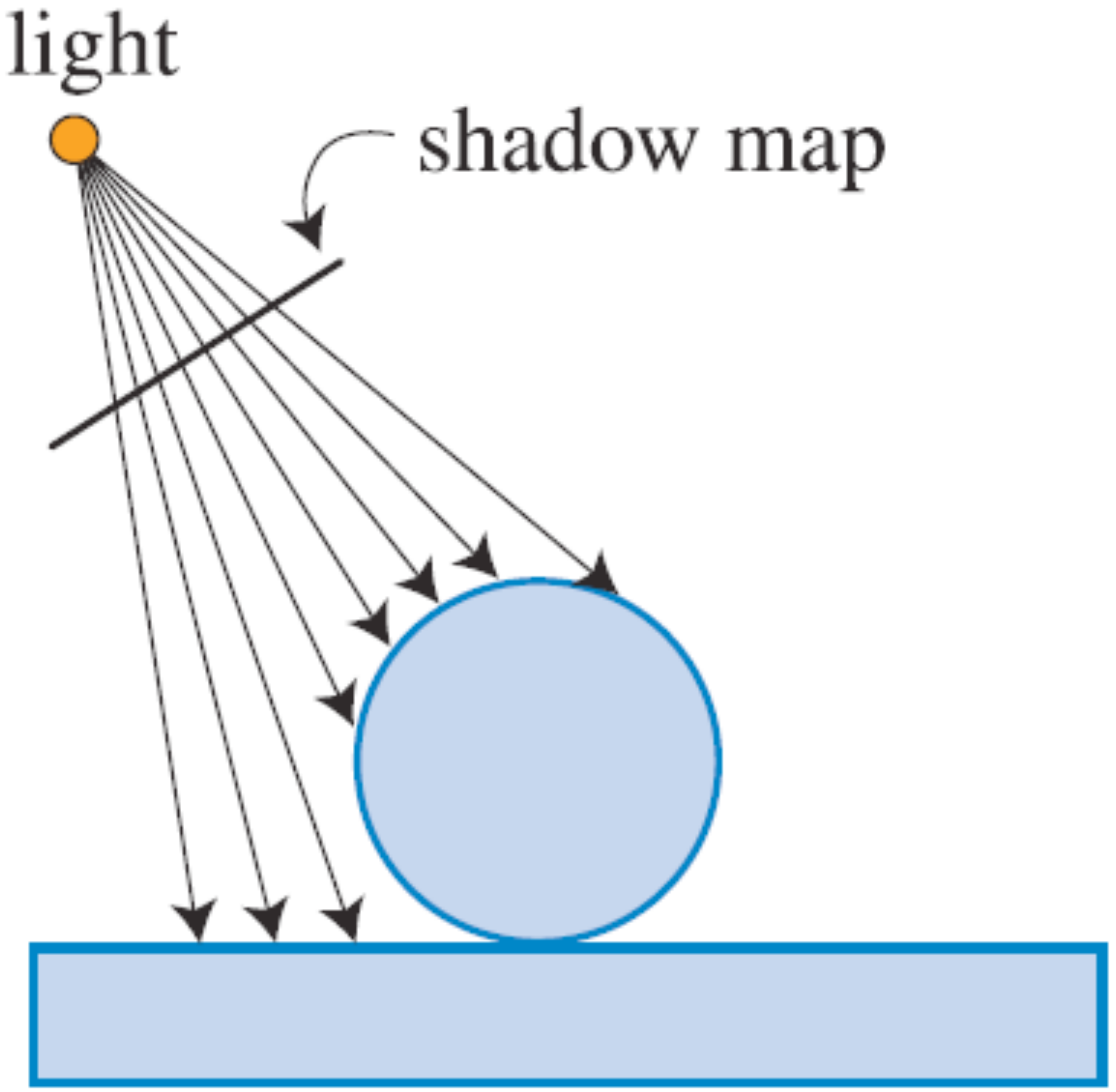
cubemap format

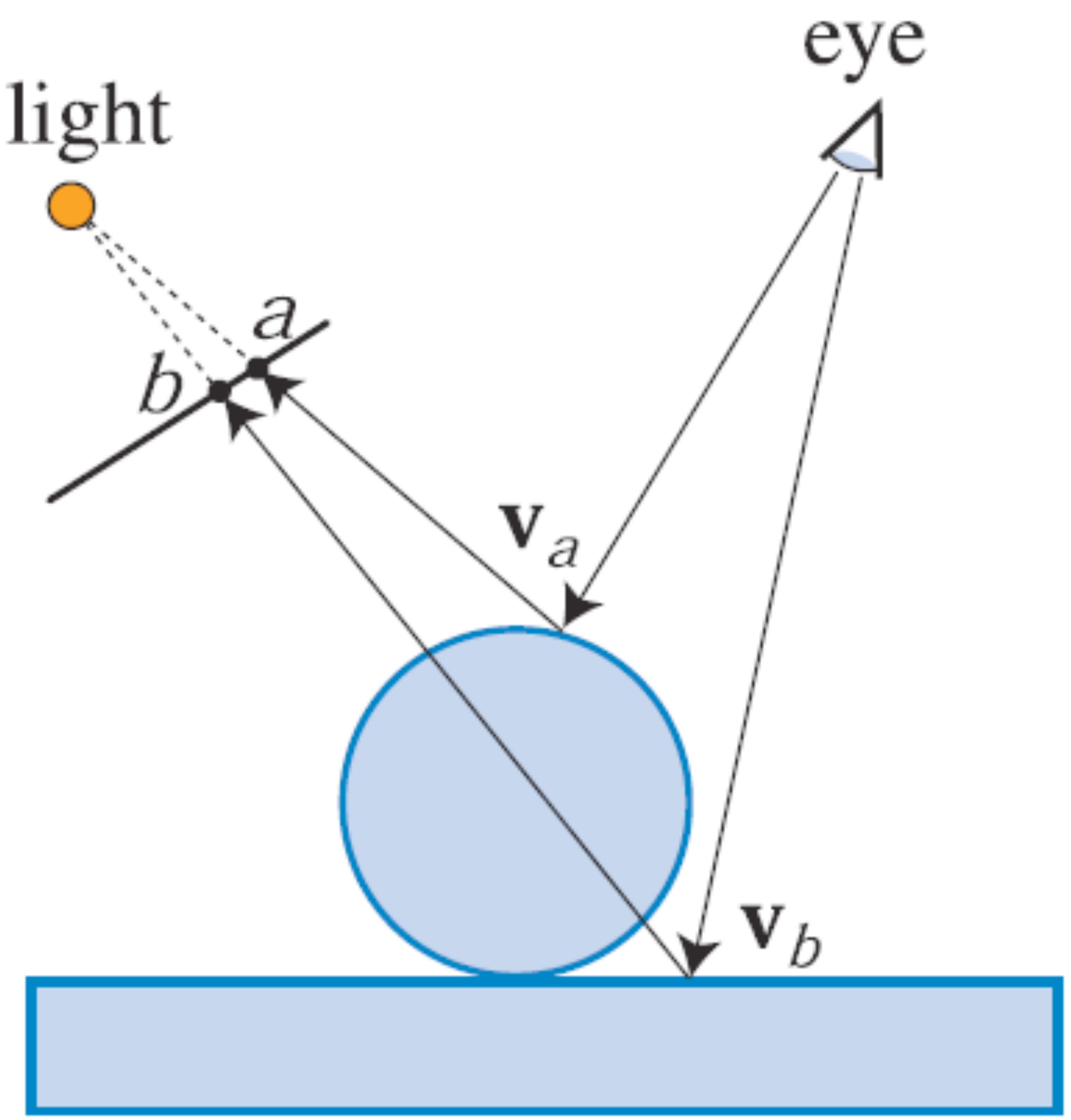
# Shadows as depth cue

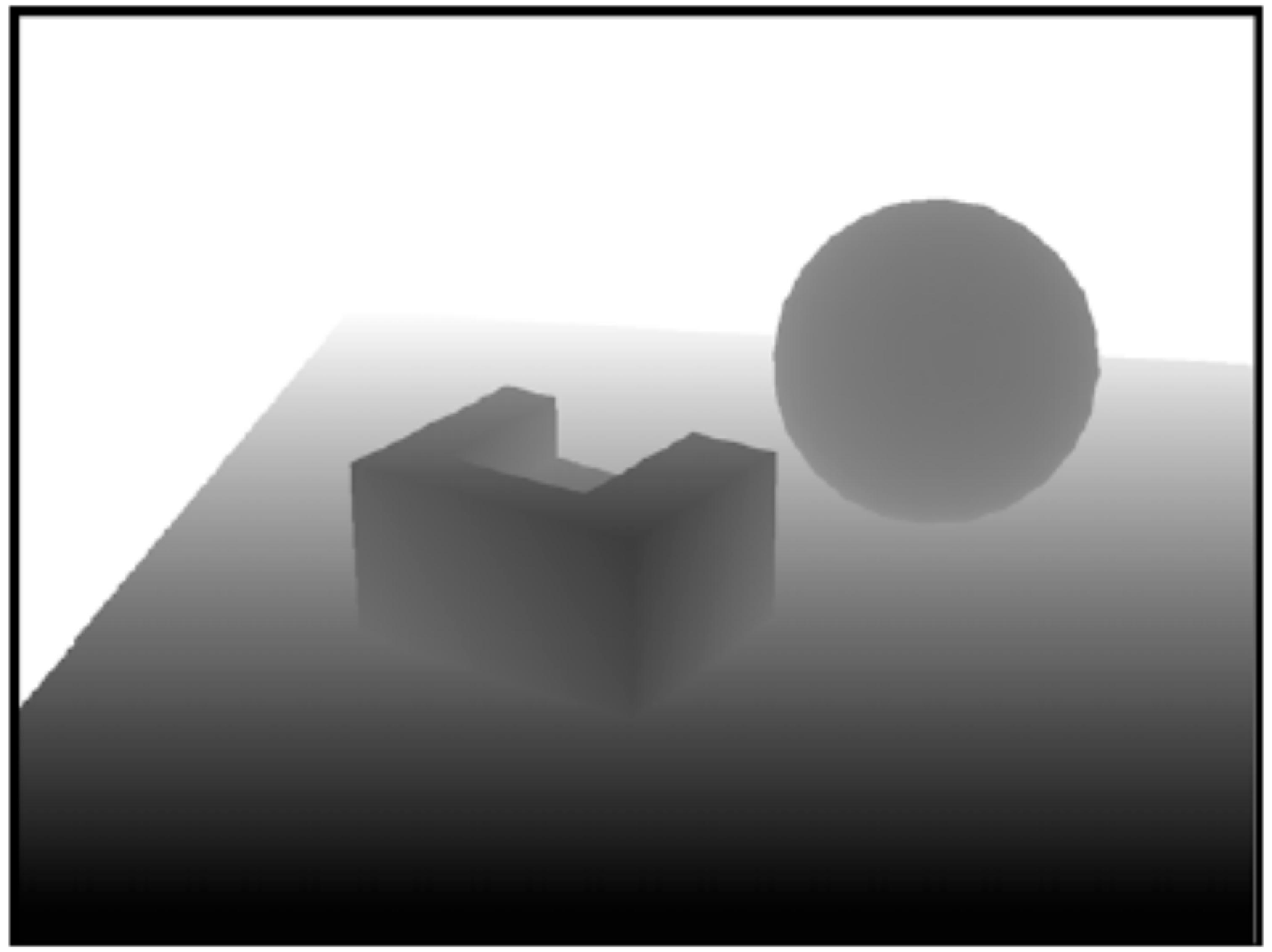


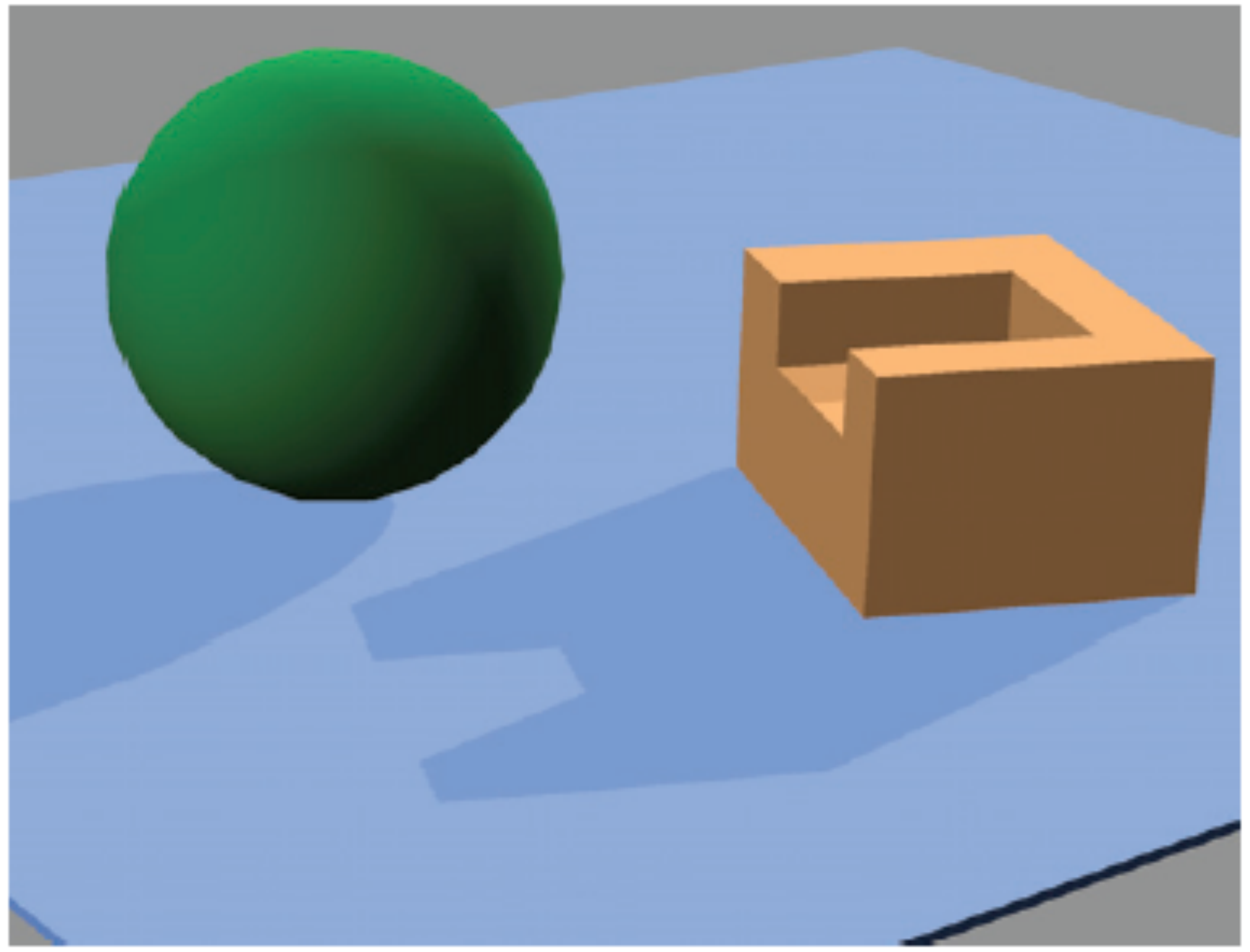
[tricks-and-illusions.com]

# Shadows as anchors

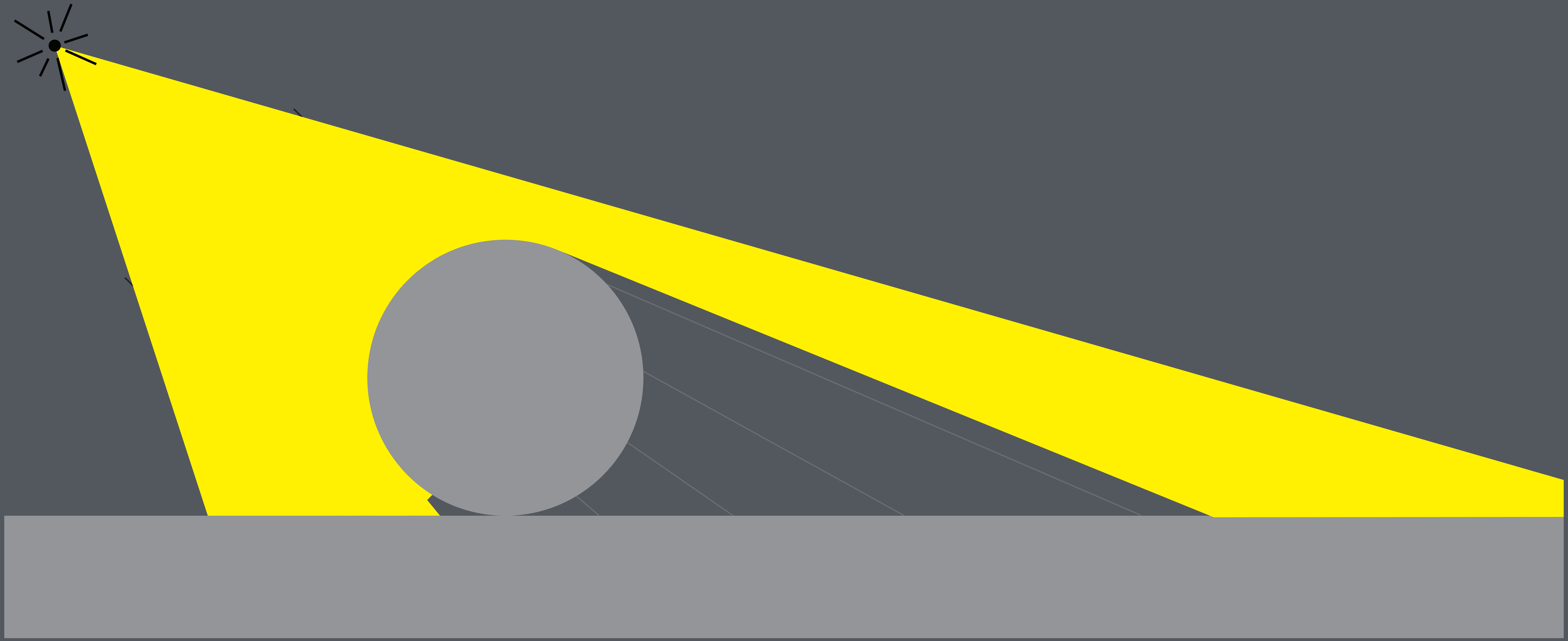








[Möller et al. *RTR*]



Pixar—*Toy Story* (1995)

# Categories of illumination

| | diffuse | glossy | mirror |
|---|---|---|---|
| point/directional | hard shadows | simple specular highlight | point reflections |
| area | soft shadows | shaped specular highlight | reflected image of source |
| environment | soft shadows | blurry reflection of environment | reflected image of environment |
| indirect | soft indirect illumination | blurry reflections of other objects | reflected images of other objects |

🟩 = easy to compute using traditional basics

# Modern shading basics

# “Physics based rendering” in real time

**One hears a lot about “PBR” — what does this boil down to?**

**1. Using reflection models designed to resemble reality**

- normally Microfacet models
- models should be energy conserving (reflected energy < incident energy)

**2. Making some effort to use physical units consistently**

- point lights have inverse-square falloff; area lights and environment lights are consistent
- use floating point pixels when appropriate (allows high dynamic range)
- pay careful attention to gamma correction / sRGB quantization

**3. Generally, thinking of shading as approximating a real illumination setup**

# Reflection models

**Lambertian diffuse**

- fine as is, physically plausible

**Mirror specular**

- fine with Fresnel factor

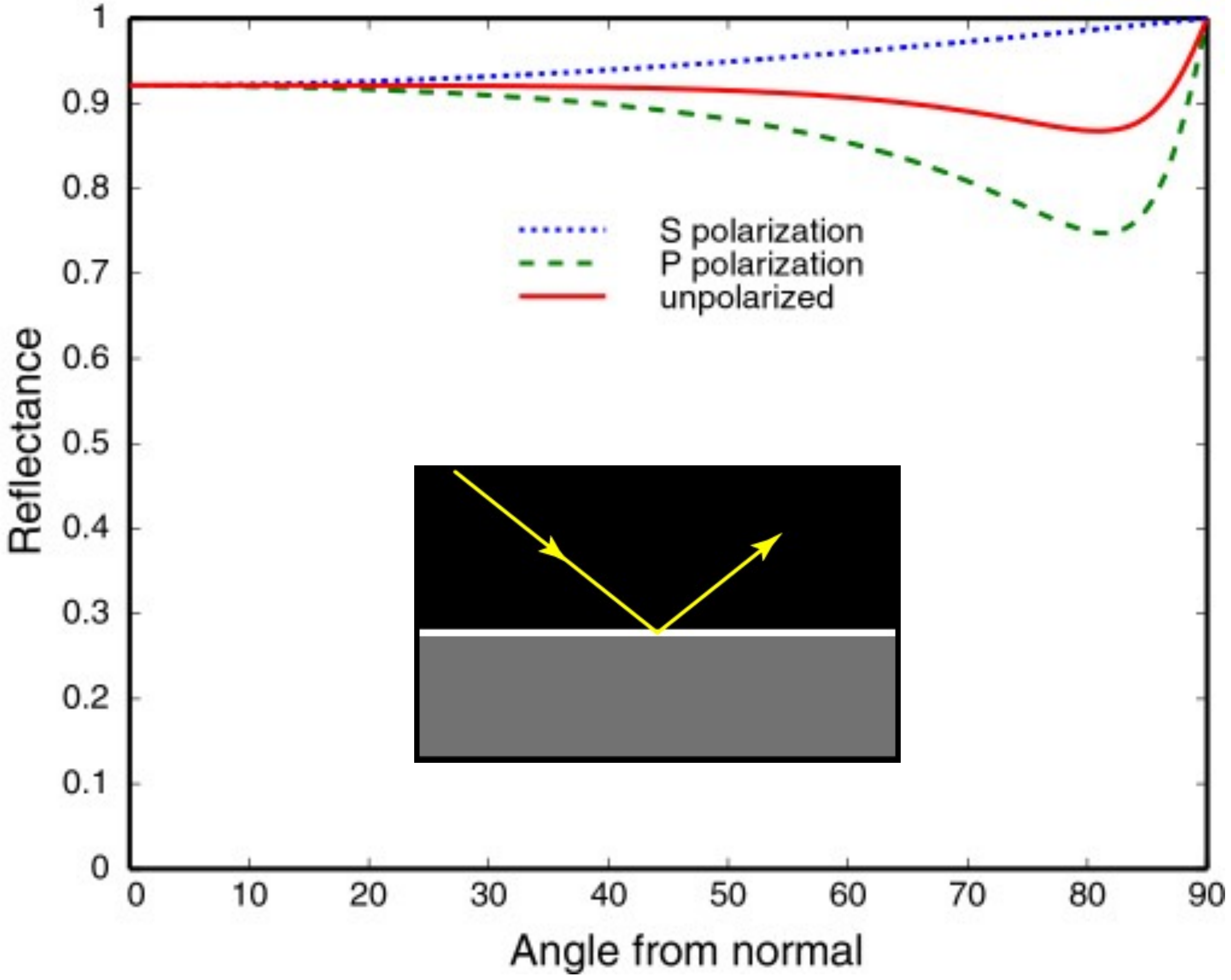
**Glossy specular**

- needs upgrade

# Specular reflection from metal

- **Reflectance does depend on angle**
  - but not much
  - safely ignored in basic rendering

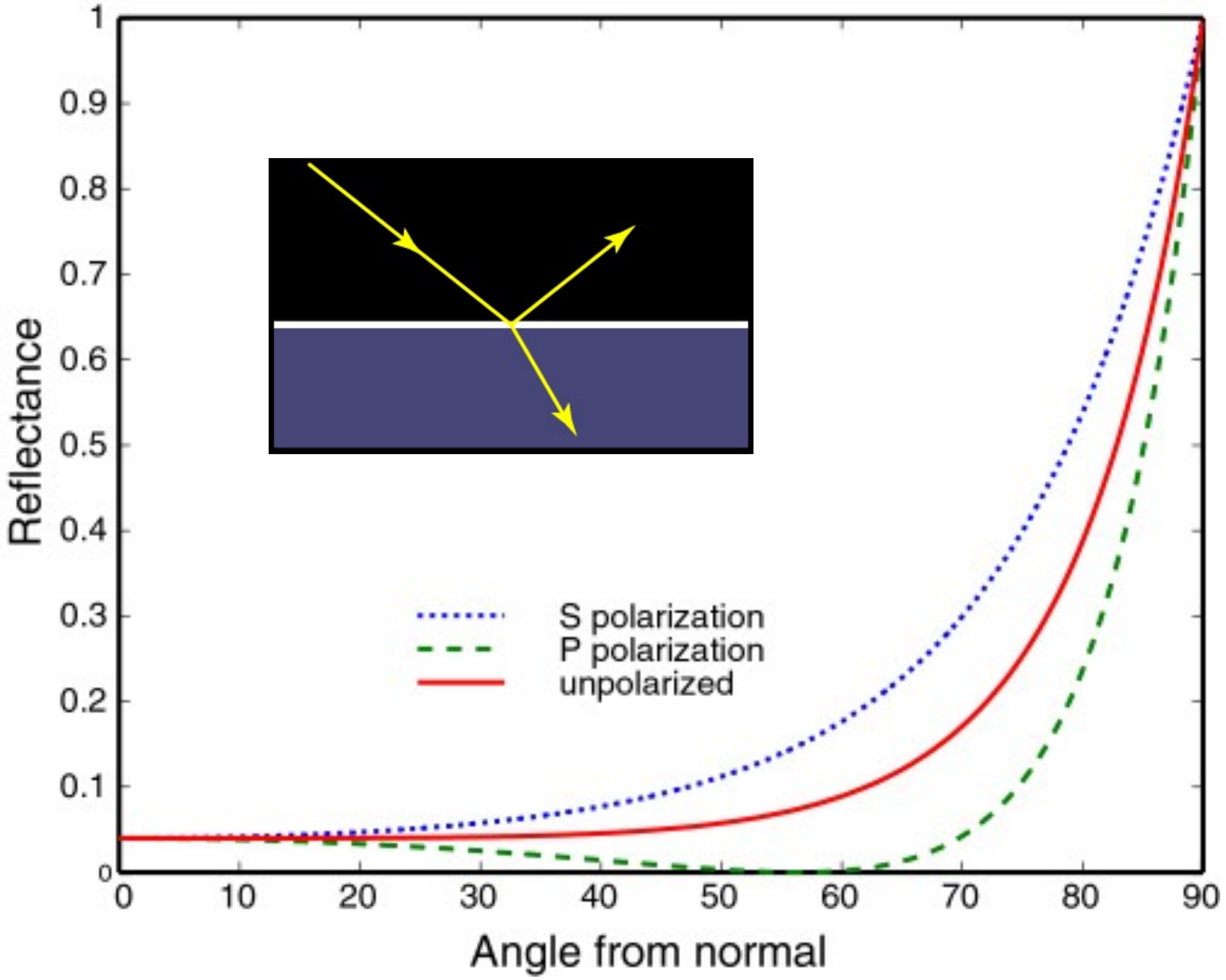
Aluminum



# Specular reflection from glass/water

- **Dependence on angle is dramatic!**
  - about 4% at normal incidence
  - always 100% at grazing
  - remaining light is transmitted
- **This is important for proper appearance**

Glass



# Fresnel's formulas

- **They predict how much light reflects from a smooth interface between two materials**
  - usually one material is empty space

$$F_p = \frac{\eta_2 \cos\theta_1 - \eta_1 \cos\theta_2}{\eta_2 \cos\theta_1 + \eta_1 \cos\theta_2}$$

$$F_s = \frac{\eta_1 \cos\theta_1 - \eta_2 \cos\theta_2}{\eta_1 \cos\theta_1 + \eta_2 \cos\theta_2}$$

$$R = \frac{1}{2}\left(F_p^2 + F_s^2\right)$$

  - $R$ is the fraction that is reflected
  - $(1 - R)$ is the fraction that is transmitted

where

$$\eta_1 \sin\theta_1 = \eta_2 \sin\theta_2$$

**note:** the formula in the notes and assignment is different but equivalent.

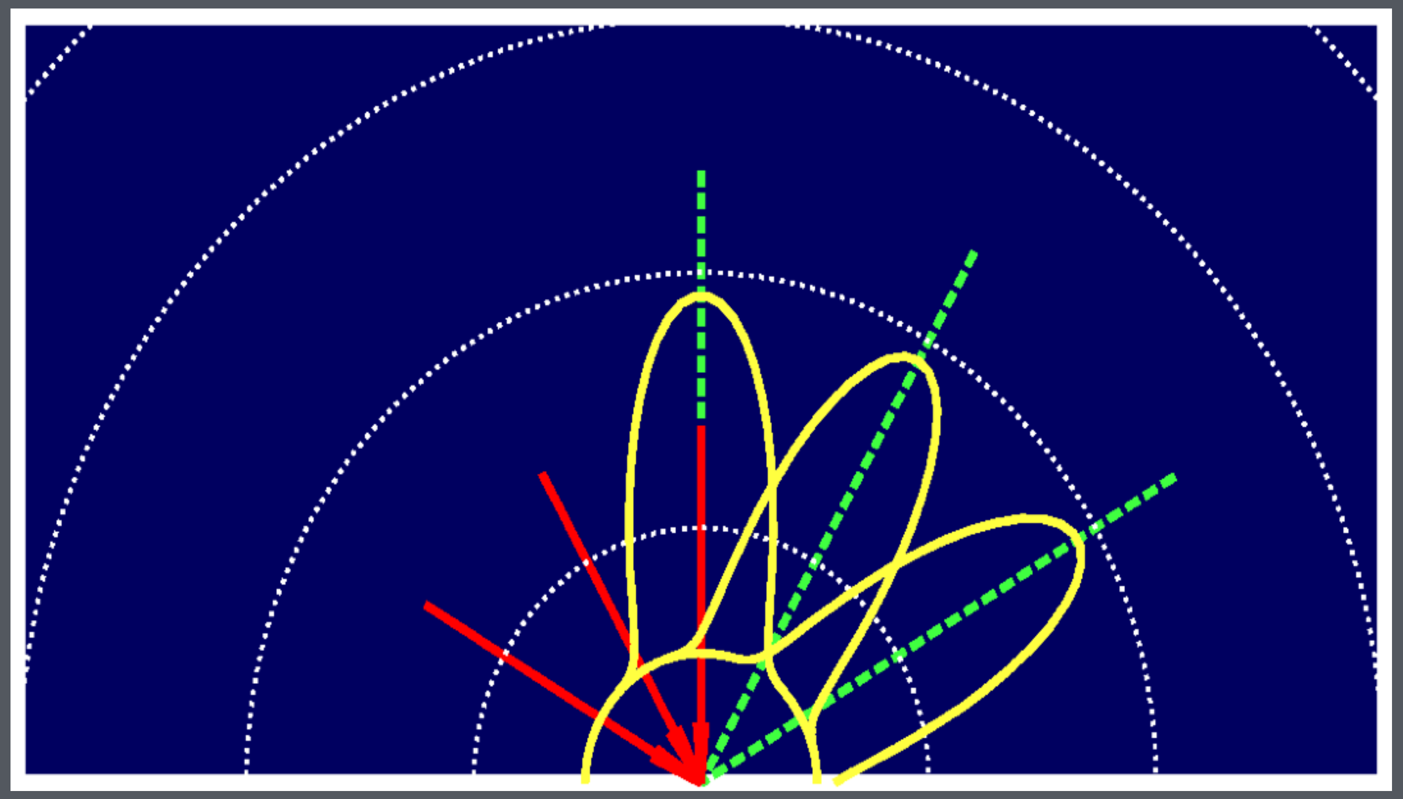
# Types of glossy reflection models

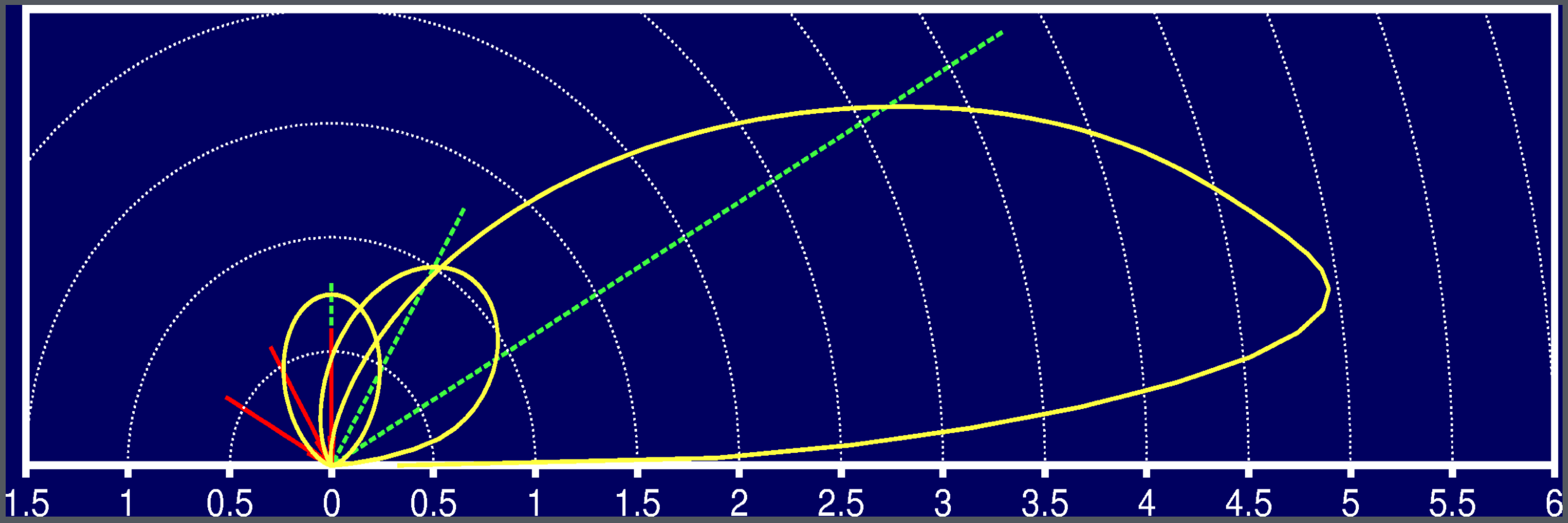
**Ad hoc formulas**

- e.g.Phong

**Physics-based analytical models**

- Microfacet-based models
- Kirchhoff-based models





# Microfacet BRDF Model

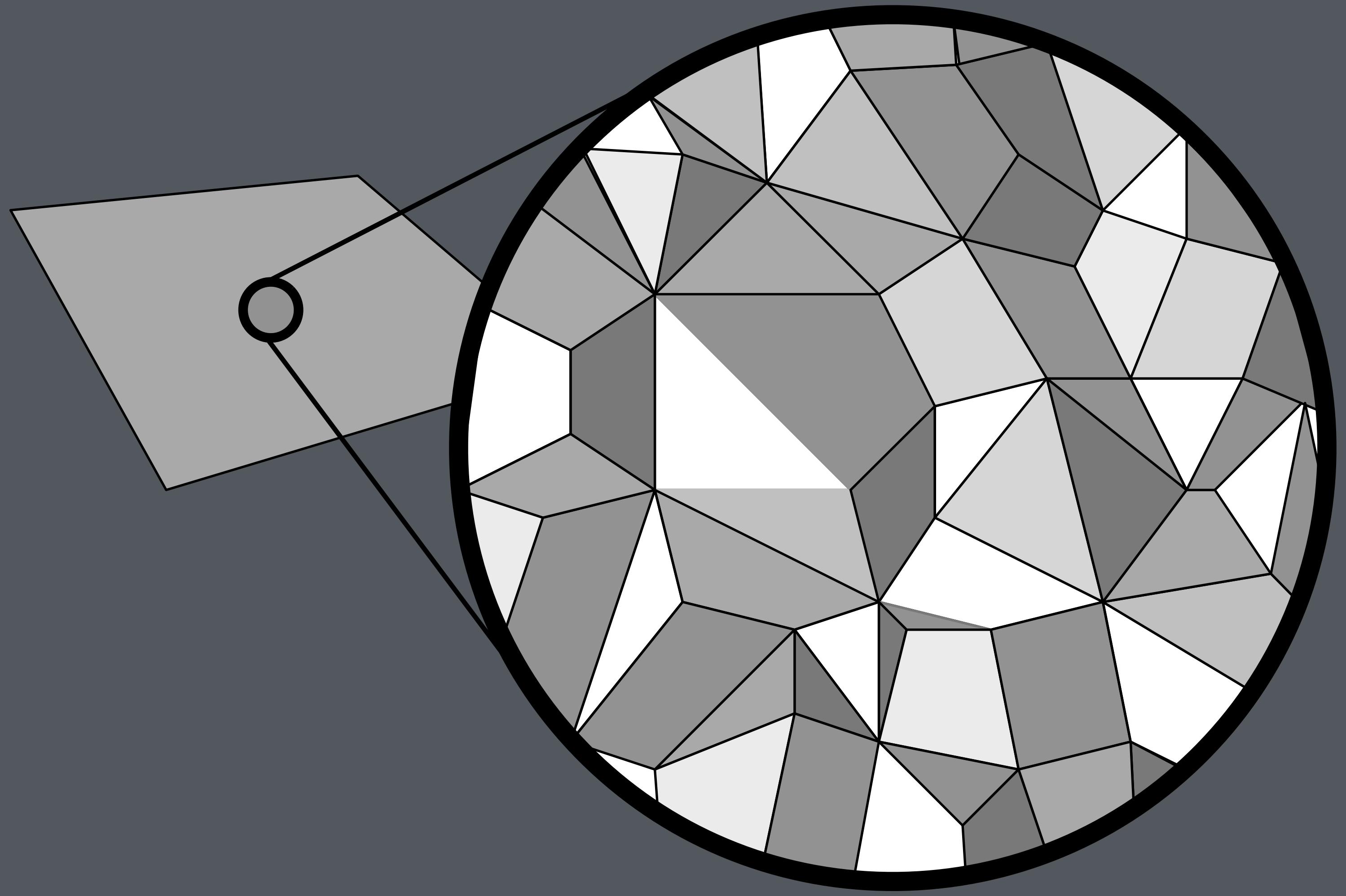
**The *microfacet* idea**

- surface modeled as random collection of planar facets
- an incoming ray hits exactly one facet, at random

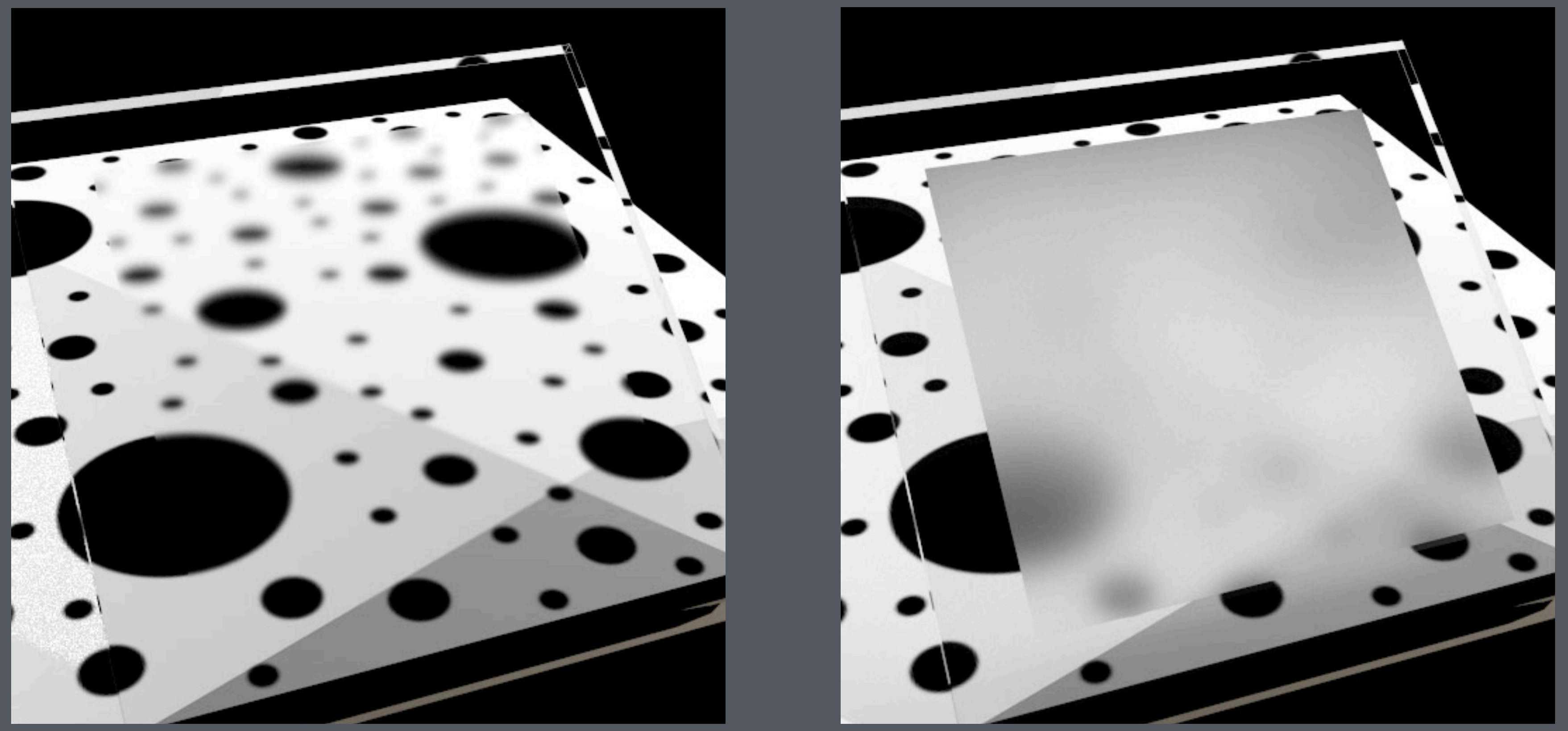
**Key input: probability distribution of facet angle**

**Common choices:**

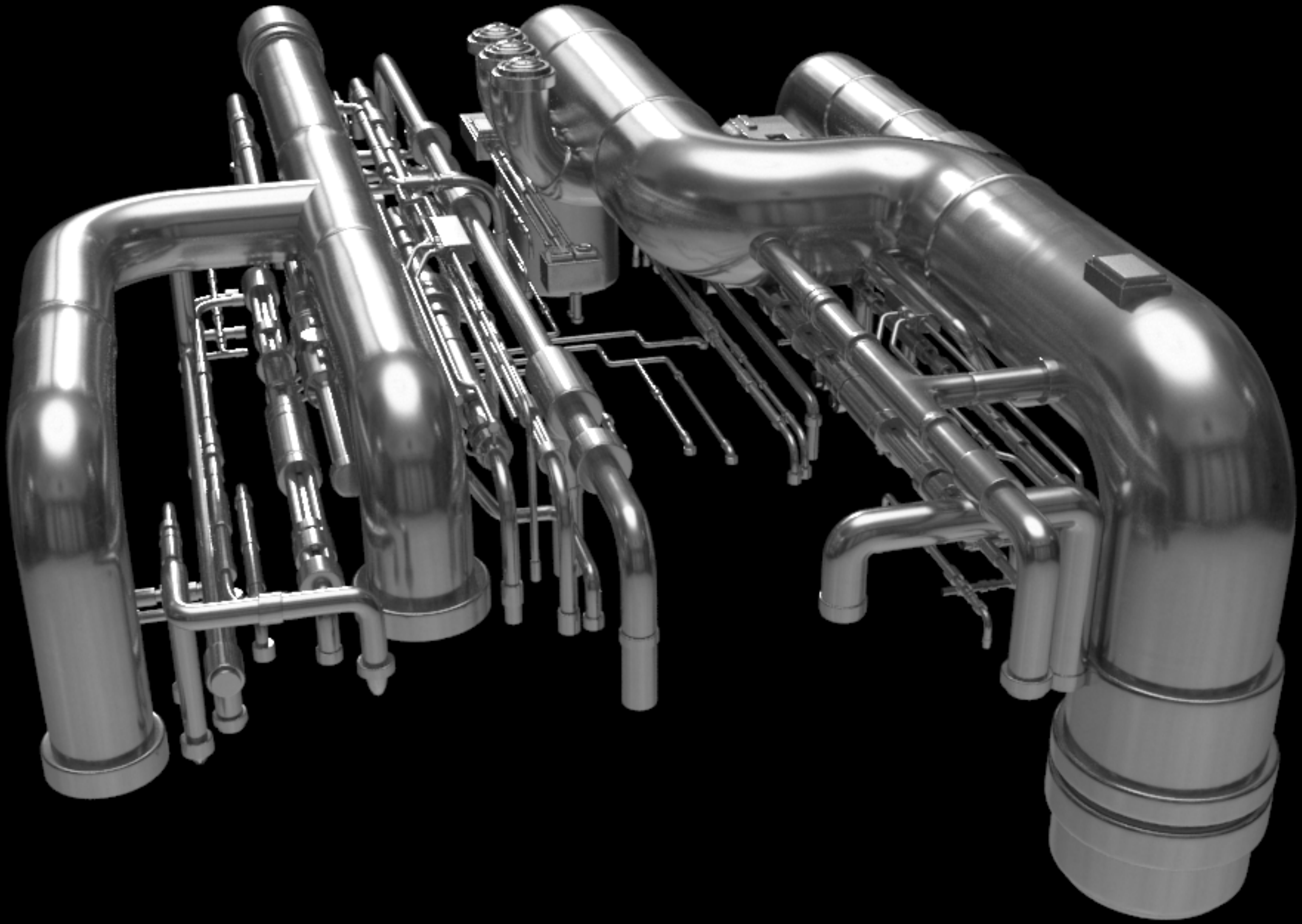
- Gaussian (Beckmann)
- GGX (orig. for glass, but has become broadly used)



[Stephen Westin]



[Walter et al. EGSR 2007]



Gaussian
(Beckmann)

GGX
(Trowbridge-Reitz)

Neil Blevins | neilblevins.com

# Reflection models

**Lambertian diffuse**

- fine as is, physically plausible

**Mirror specular**

- fine with Fresnel factor

**Glossy specular**

- needs upgrade—Microfacet

# Sources of light

**Point and directional lights**

- Direct shading computations, simple shadow maps

**Area lights**

- shading: special solutions for particular shapes
- shadows: blurred point-light shadows or pre-baked

**Environment lights**

- shading: various methods
- shadows: approx. with directional lights, pre-bake, or use ambient occlusion

**Indirect light**

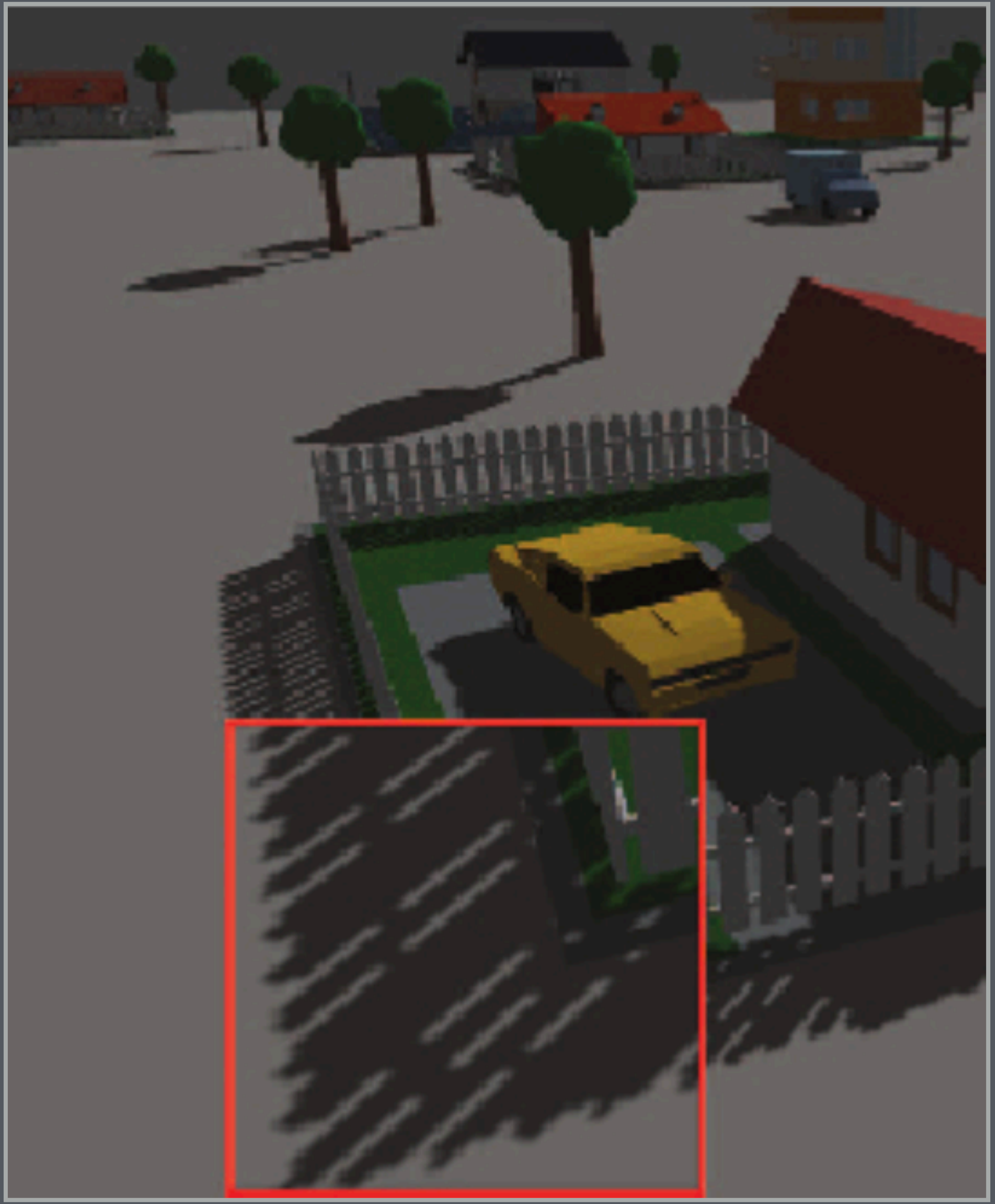
- mirror: dynamic reflection maps
- diffuse: approximate with ambient occlusion or pre-bake

# Shading from point and directional lights

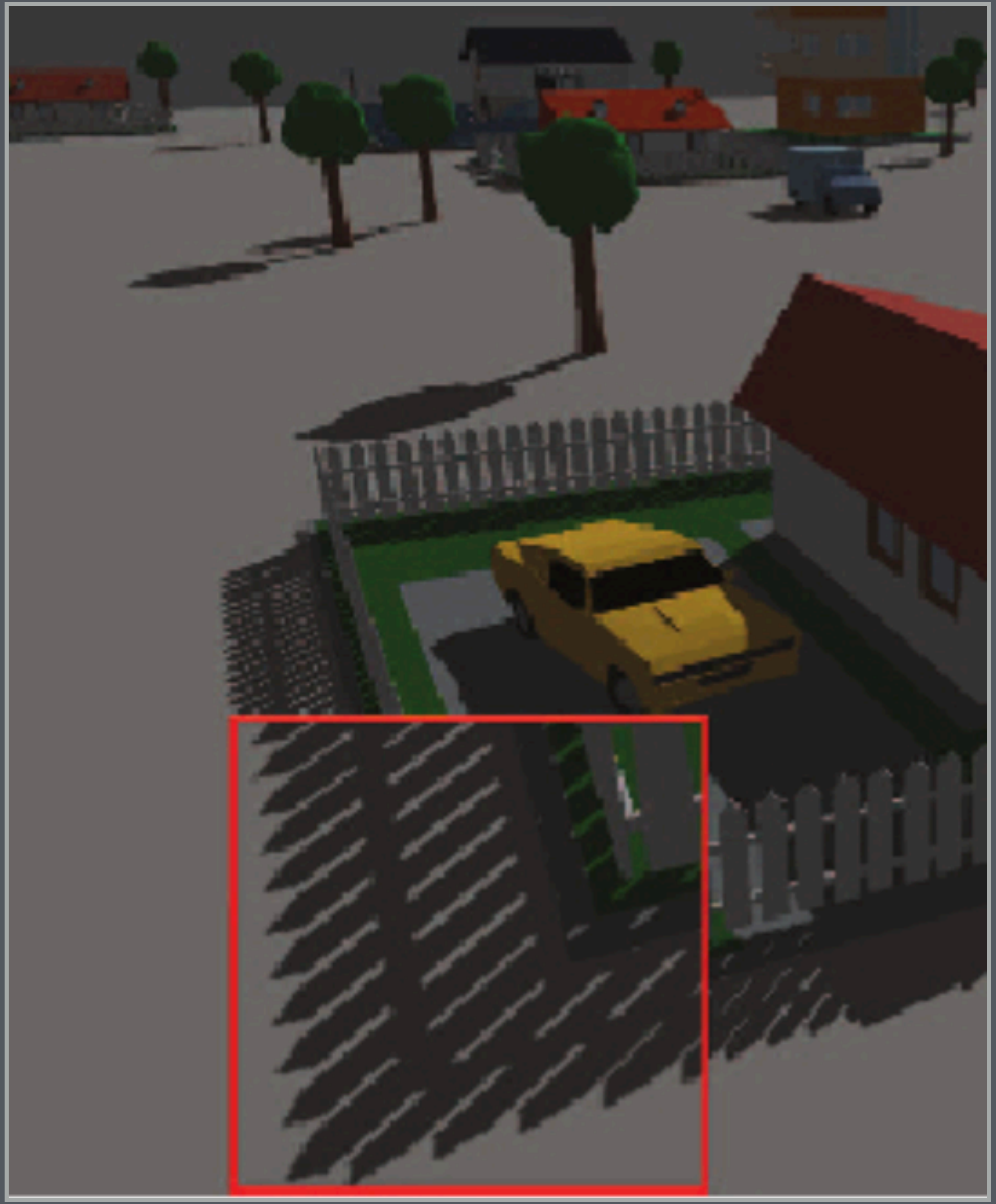
**Shading is straightforward**

**Shadow maps work for shadows**

- shadow map resolution is always an issue
- if shadows are jagged, need more resolution or different projection



Single shadow map, 2048x2048



Four 1024x1024 shadow maps (equal memory)

Fan Zhang, Chinese U. Hong Kong

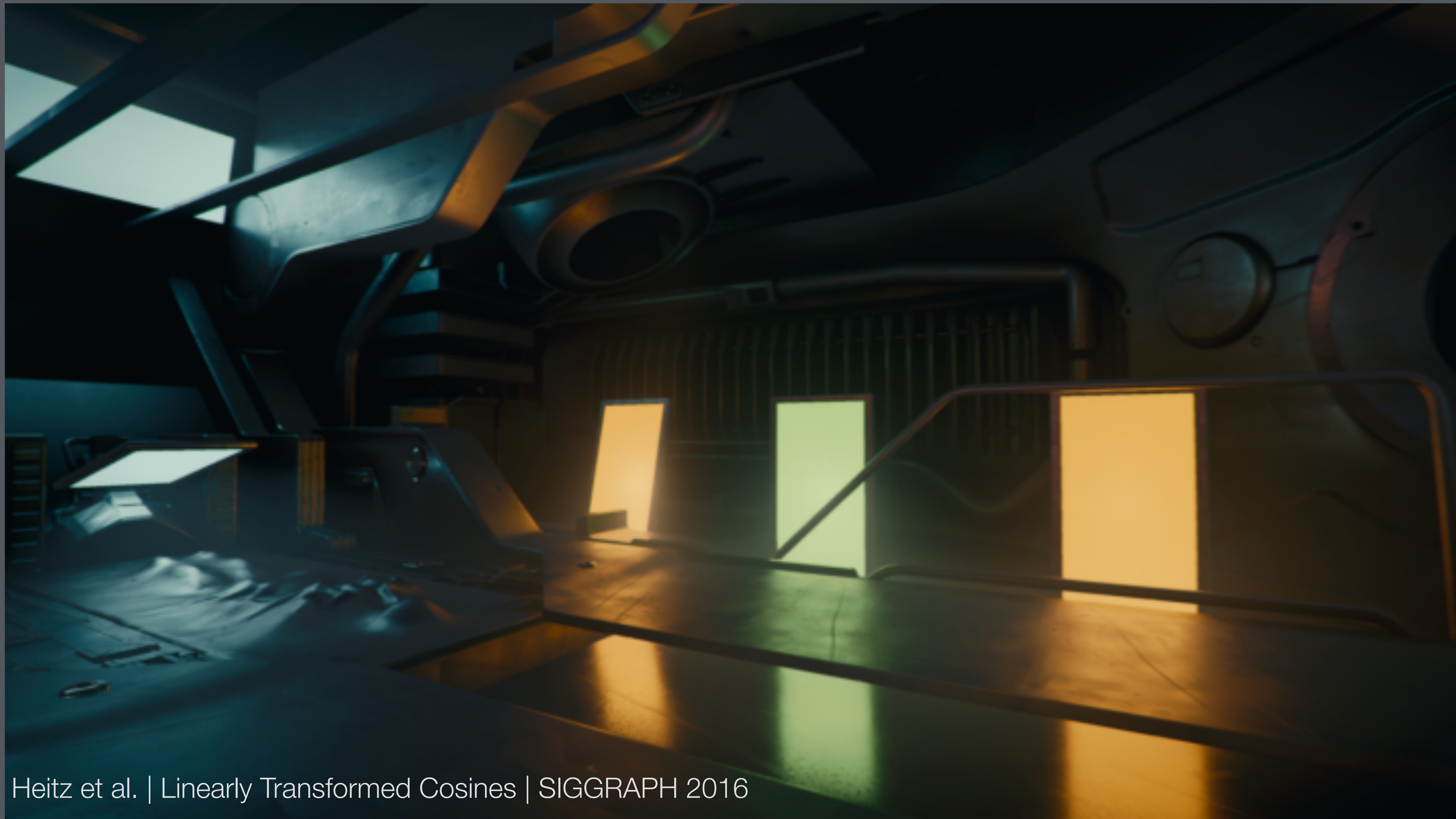
# Shading from area lights

## Traditionally not really doable in real time

- can approximate small area lights with multiple point lights

## Some methods work for particular shapes

- polygons and spheres
- exact formulas for diffuse
- good approximations available for microfacet



Heitz et al. | Linearly Transformed Cosines | SIGGRAPH 2016

# Shadows from area lights

**For small lights, blur the results of shadow mapping**

- not very accurate, but OK in many settings

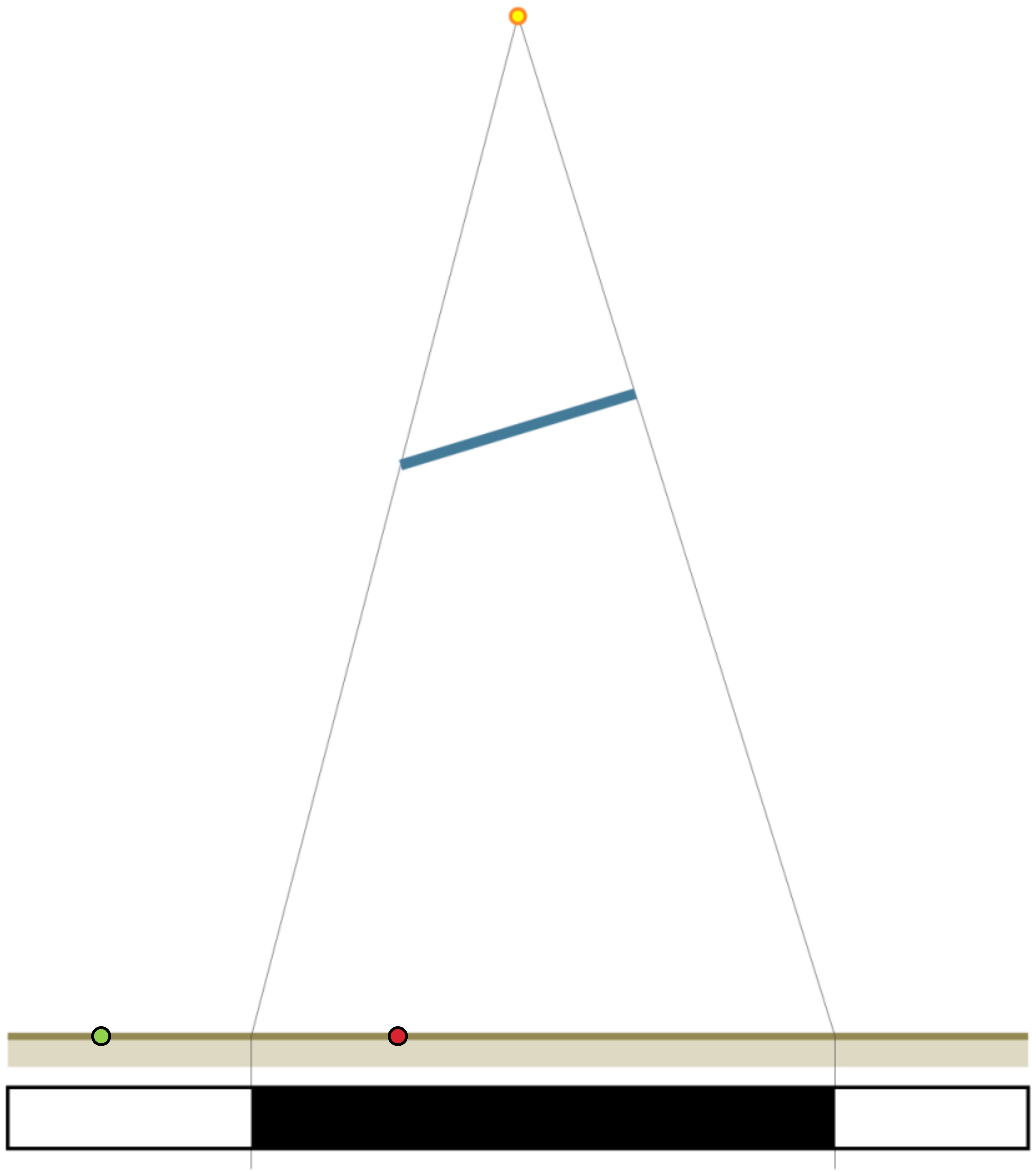
**For larger lights, one approach is pre-baked shadows**

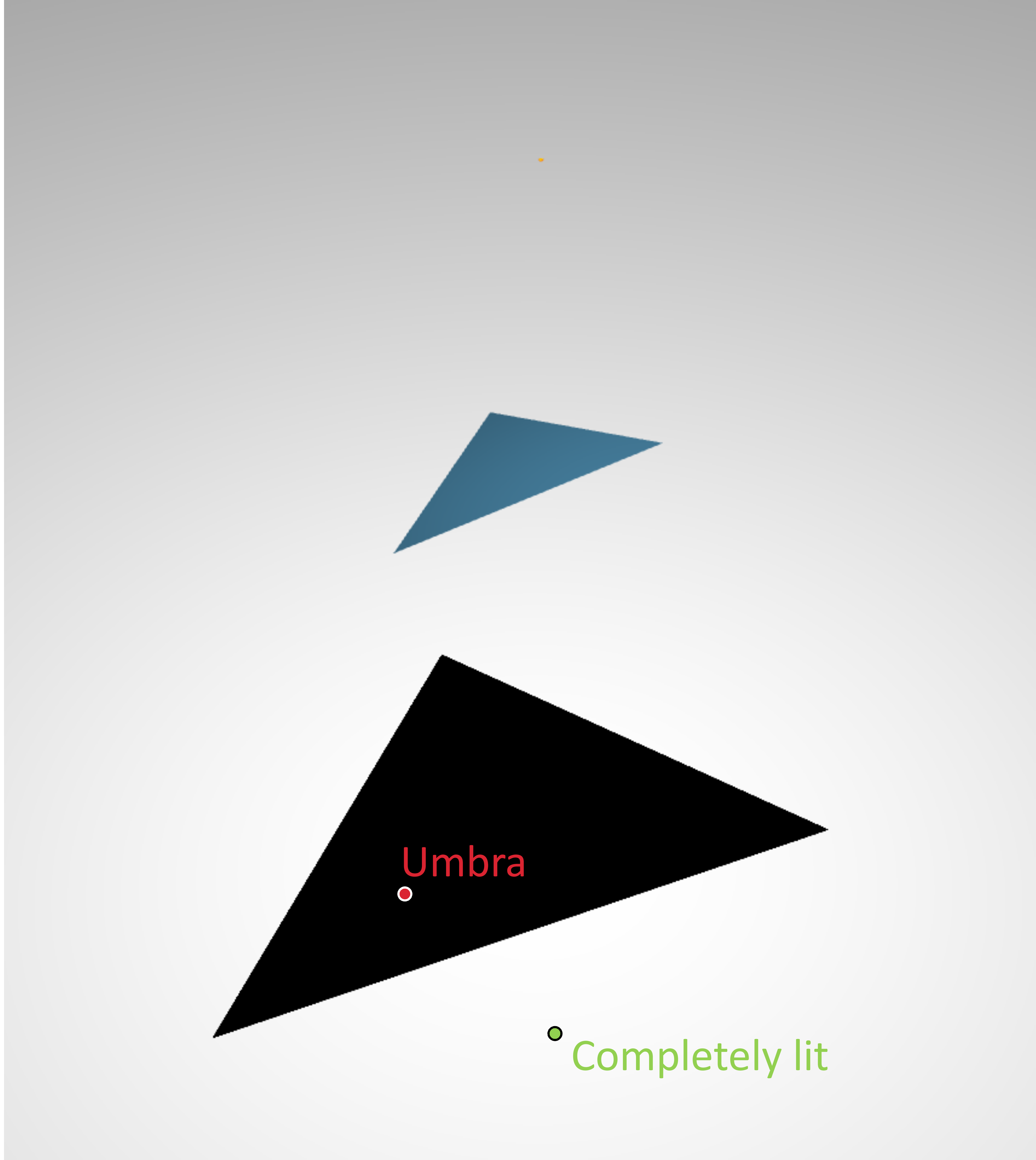
- for geometry that will not move, shadows are fixed
- compute ahead of time, stash in texture map

**In near future, ray tracing can be used for accurate shadows**

# Hard Shadows

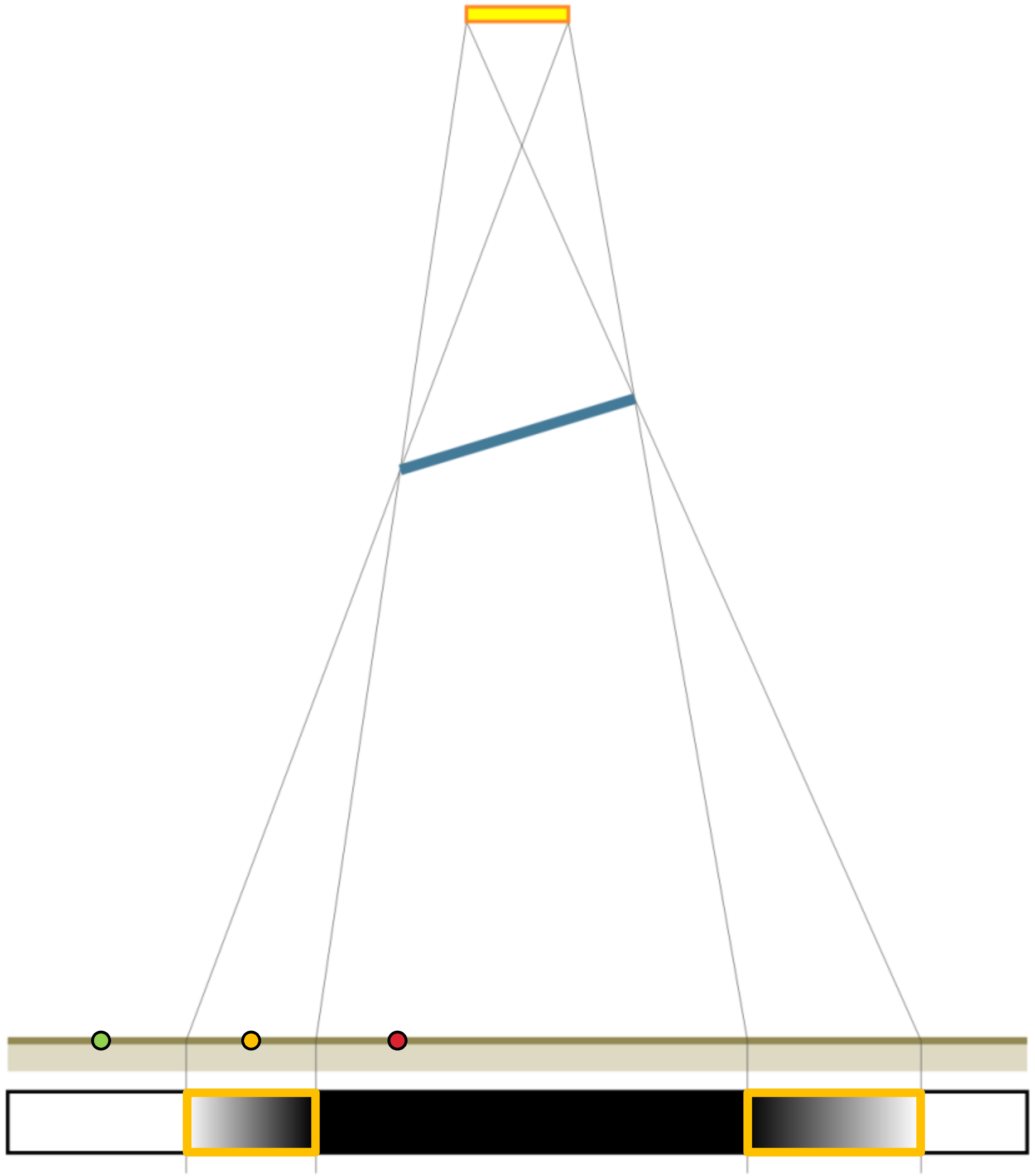
Michael Schwarz, SIGGRAPH 2013 Real Time Shadows course





# Soft Shadows

Michael Schwarz, SIGGRAPH 2013 Real Time Shadows course



# Shadow Hardening on Contact

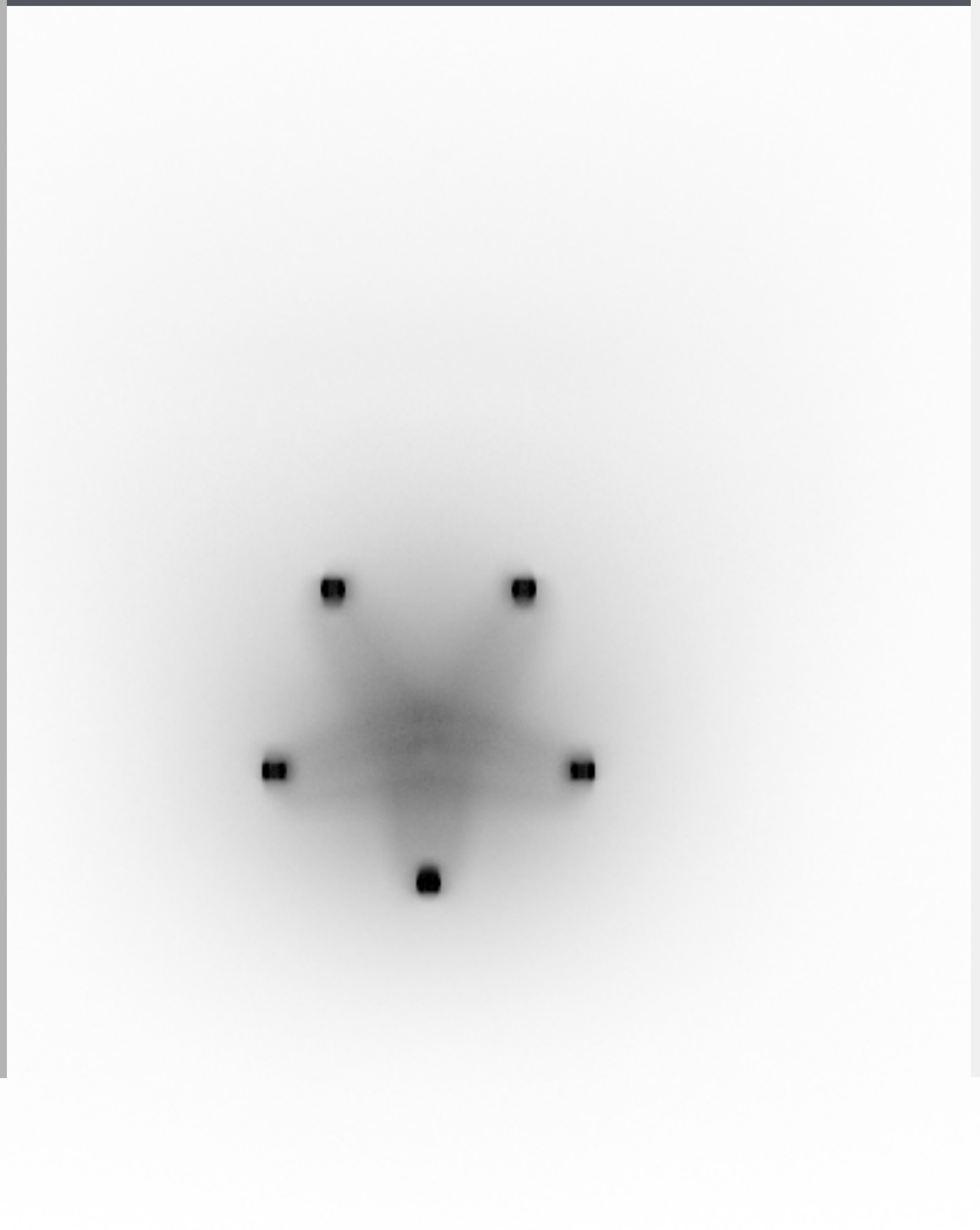
Michael Schwarz, SIGGRAPH 2013 Real Time Shadows course

# Percentage-closer soft shadows

Fernando, NVidia whitepaper ~2005

# Shadow baking

Rendering with no shadows,
darker diffuse floor



Irradiance texture computed
using rectangular light

Floor shaded with irradiance
from shadow texture

# Environment lights

**A key tool for modern lighting**

- especially in outdoor scenes

**Remember they are infinitely far away**

- you can never get closer to the sky by moving around the scene

**In physically based context, should be high dynamic range**

- Consistent lighting requires environments to illuminate everything (not just be reflected in mirror surfaces)

**Provided as texture maps or as analytical sun-sky models**

**A spherical panorama, aka. enironment map**

openfootage.net

# Shading with environment lighting

**Mirror surfaces**

- standard reflection mapping

**Diffuse surfaces**

- shading is a very smooth function of the surface normal
- can be handled cheaply using *irradiance maps*

**Glossy surfaces**

- surface color is a local average of the environment
- not simple, but approaches exist based on pre-blurred environments (next slide)
- in near future, ray tracing can be used for accurate glossy reflections

# Shadows from environment lights

**Shadows from very large / environment sources are very soft**

- can be beautiful and subtle tools for anchoring objects in 3D
- however, there are not many good and fast methods to compute them

**One approach: approximate with directional lights**

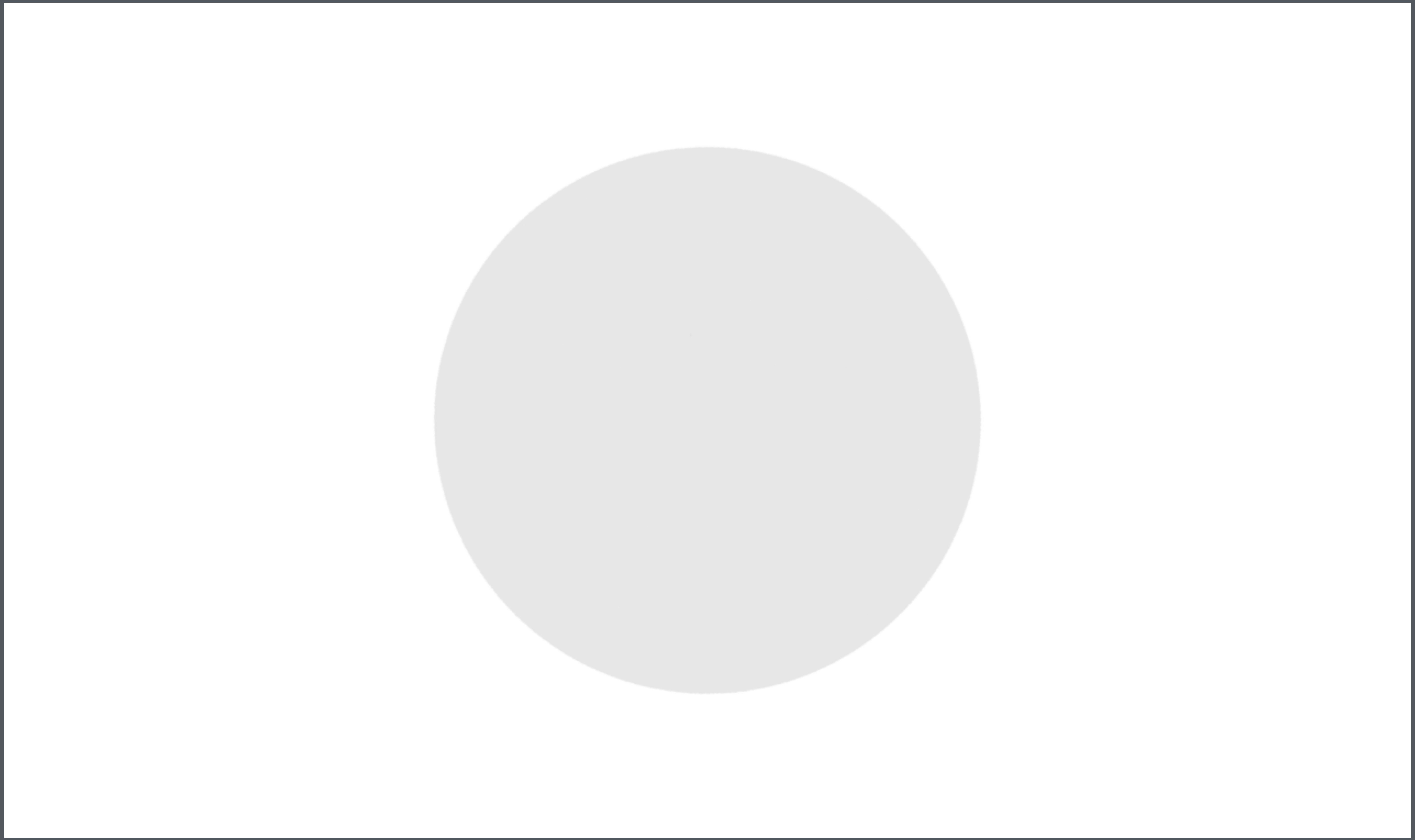
# Ambient occlusion

**The extreme case of large light sources is a constant environment**

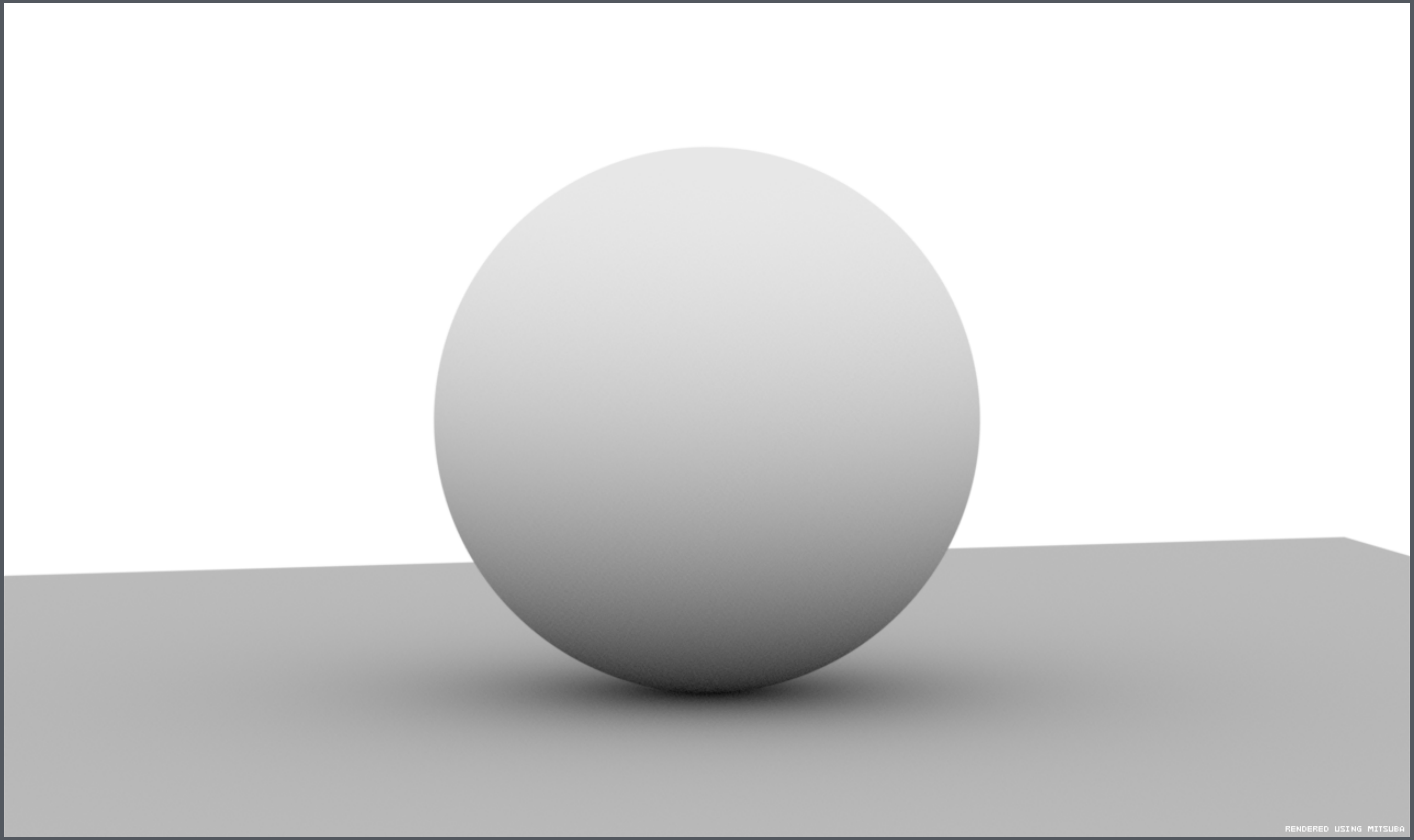
- ordinarily might call this “shadowless” lighting
- but the subtle shadows are still very important

**This extreme case allows for some handy approximations**

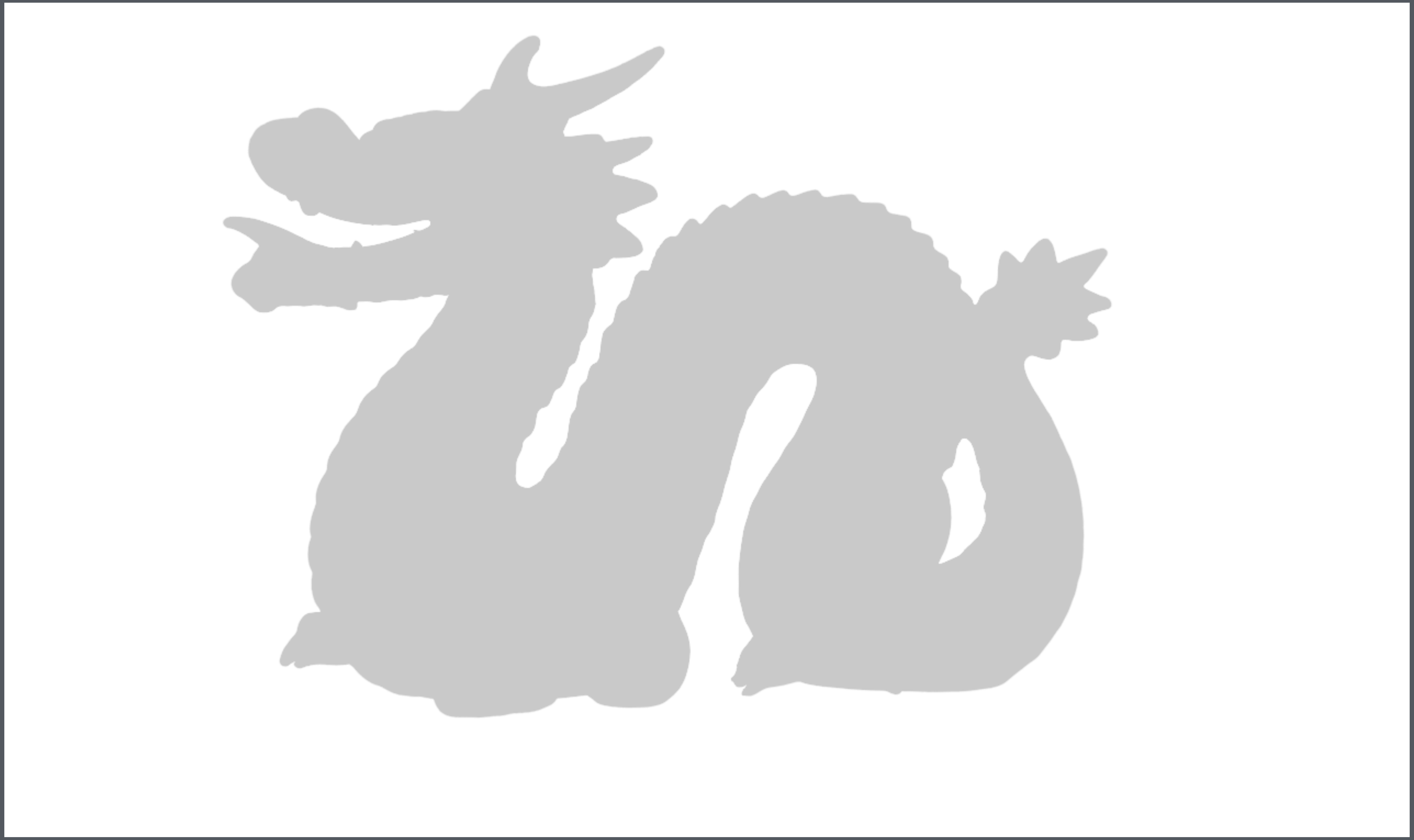
- for diffuse surfaces, shadowing depends only on how much “sky” a point can see
- this “fraction of sky” is called ambient occlusion



a convex diffuse object in a constant-radiance environment

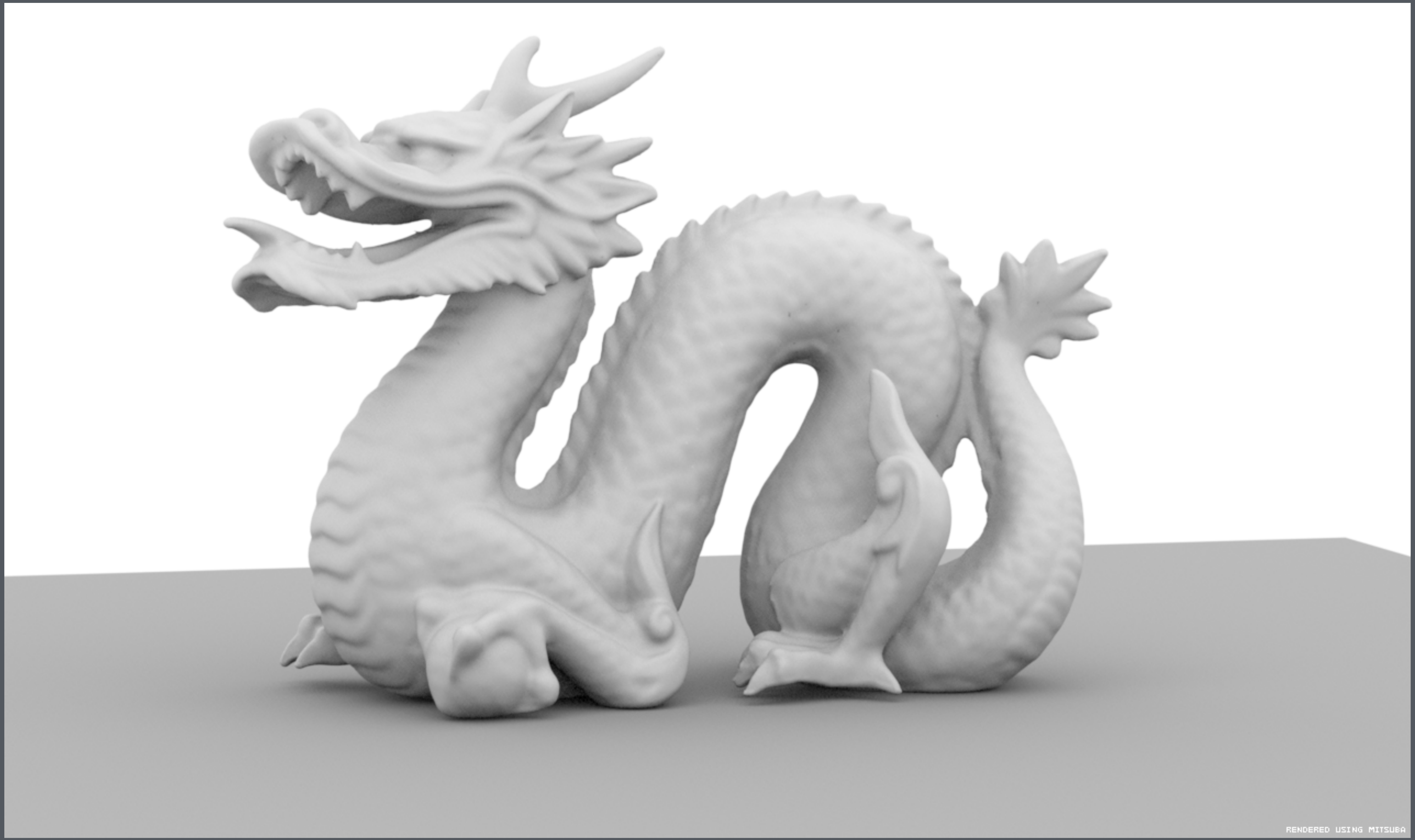


a non-convex diffuse scene under constant-radiance illumination



a non-convex object in a constant-radiance environment with no shadowing

a non-convex diffuse object in a constant-radiance environment



a non-convex diffuse scene under constant-radiance illumination

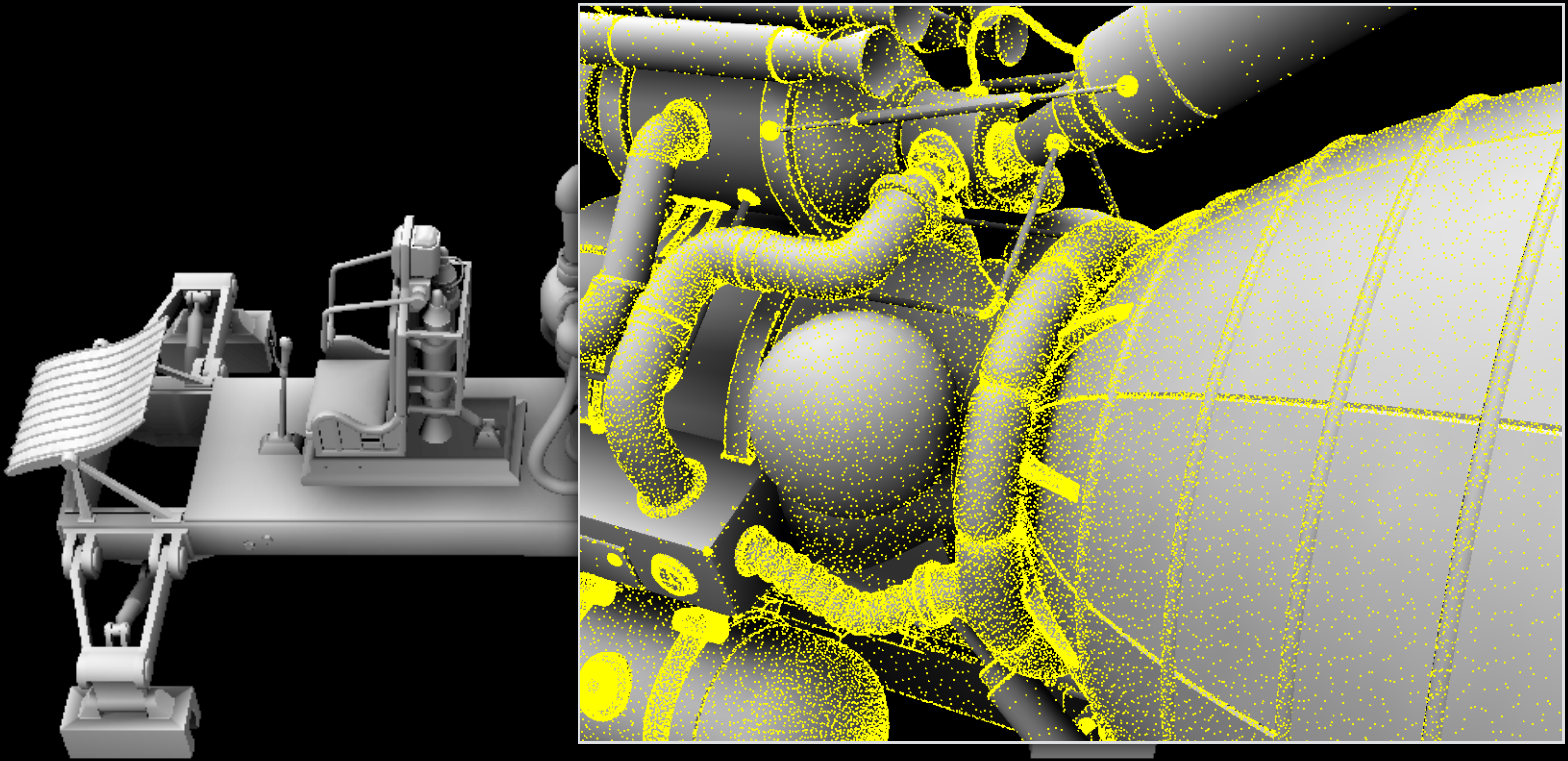
# Computing ambient occlusion

**Can pre-bake AO**

- basically the same idea as a baked shadow for an area light
- can store AO as a vertex attribute (quality depends on triangle sizes)
- …or in a texture map (can be higher quality but requires texture coordinates, lots of textures)

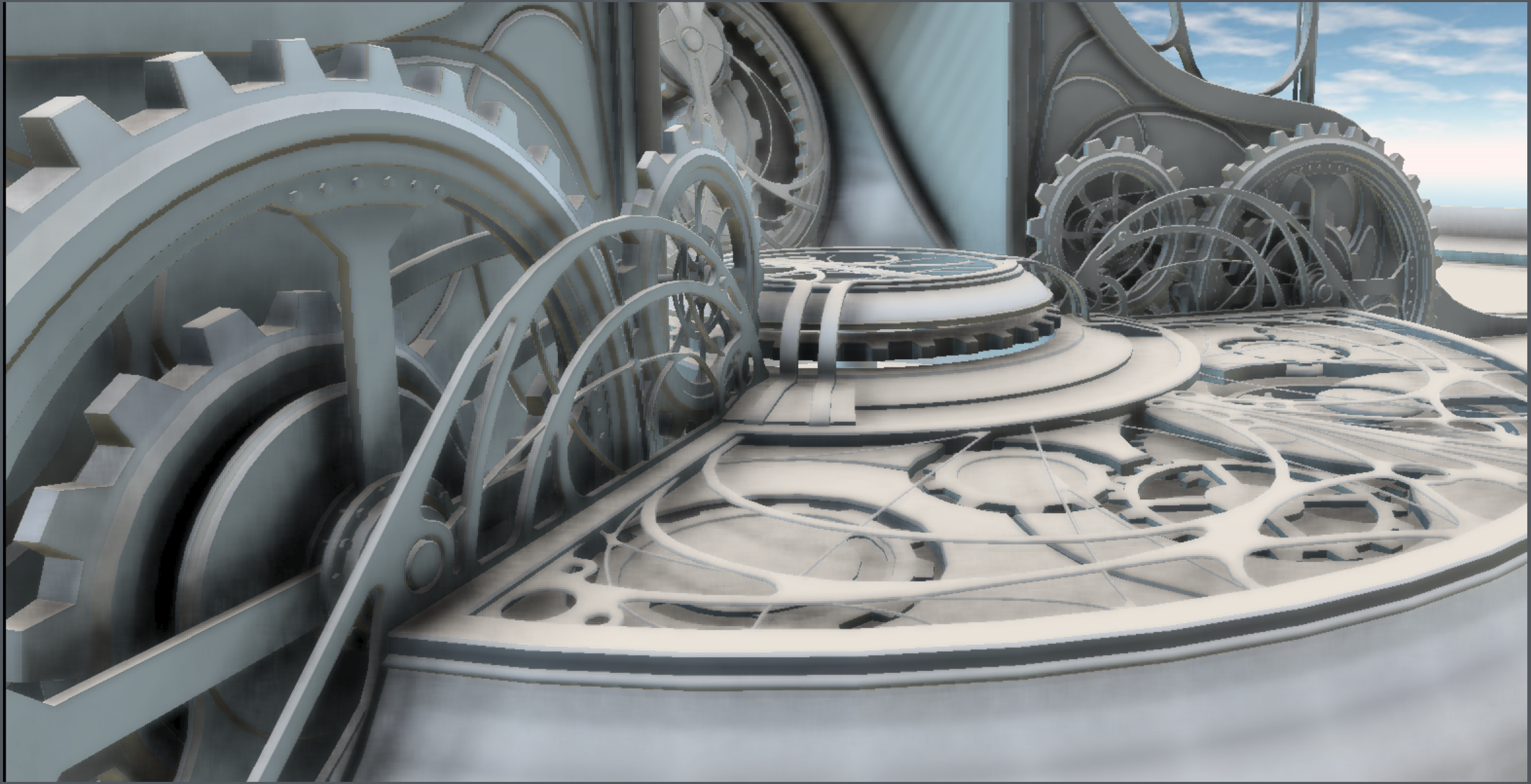
**Can compute approximate AO in real time**

- using clever tricks with the depth buffer
- quite approximate
- has limitations (only can be shadowed by visible surfaces, etc.)
- but rather effective in practice, fairly widely used



https://developer.nvidia.com/optix-prime-baking-sample

# Irradiance map + SSAO



McGuire et al. HPG ’11 10.1145/2018323.2018327

# Indirect light

## Mirror reflections

- for small/bounded objects: dynamic cubemaps
- for better results in near future: use ray tracing

webglsamples.org

## Diffuse reflections

- ambient obscurance
- variant of ambient occlusion
- difference: fraction of rays that escape *to a particular distance*

Joshua Senouf

# Sources of light

**Point and directional lights**

- Direct shading computations, simple shadow maps

**Area lights**

- shading: special solutions for particular shapes
- shadows: blurred point-light shadows or pre-baked

**Environment lights**

- shading: various methods
- shadows: approx. with directional lights, or use ambient occlusion (baked or SSAO)

**Indirect light**

- mirror: dynamic reflection maps
- diffuse: approximate with ambient occlusion or pre-bake light maps

# Categories of illumination redux

| | diffuse | glossy | mirror |
|---|---|---|---|
| point/directional | hard shadows | simple specular highlight | point reflections |
| area | soft shadows | shaped specular highlight | reflected image of source |
| environment | soft shadows | blurry reflection of environment | reflected image of environment |
| indirect | soft indirect illumination | blurry reflections of other objects | reflected images of other objects |

🟩 = easy to compute using standard shaders

🟦 = can be approximated pretty well