Computergrafik

Matthias Zwicker
Universität Bern
Herbst 2016

Today

Shading

- Introduction
- Radiometry & BRDFs
- Local shading models
- Light sources
- Shading strategies

Shading

- Compute interaction of light with surfaces
- Requires simulation of physics
  - Solve Maxwell's equations (wave model)?
    http://en.wikipedia.org/wiki/Maxwell's_equations
  - Use geometrical optics (ray model)?
    http://en.wikipedia.org/wiki/Geometrical_optics
    http://en.wikipedia.org/wiki/Ray_(optics)

"Global illumination" in computer graphics

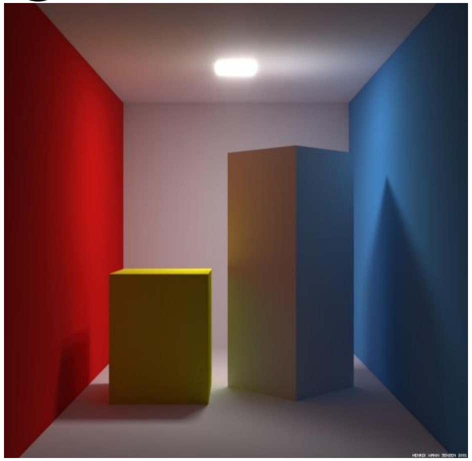
http://en.wikipedia.org/wiki/Global_illumination

- "Gold standard" for photorealistic image synthesis
- Based on geometrical optics (ray model)
- Multiple bounces of light
  - Reflection, refraction, volumetric scattering, subsurface scattering
- Computationally expensive, minutes per image
- Movies, architectural design, etc.

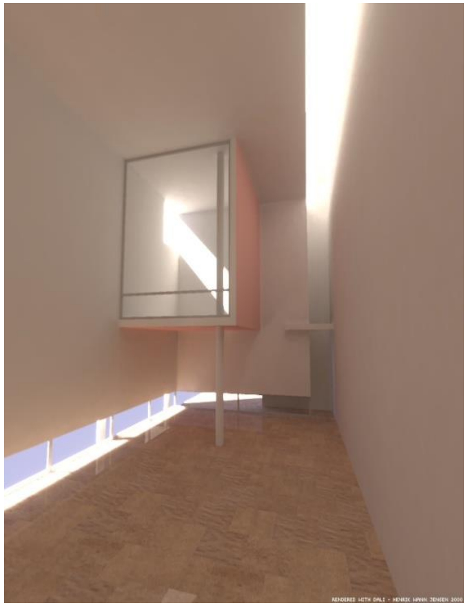
Global illumination

- Rendering algorithms, spring semester!

http://www.pbrt.org/gallery.php



Henrik Wann Jensen

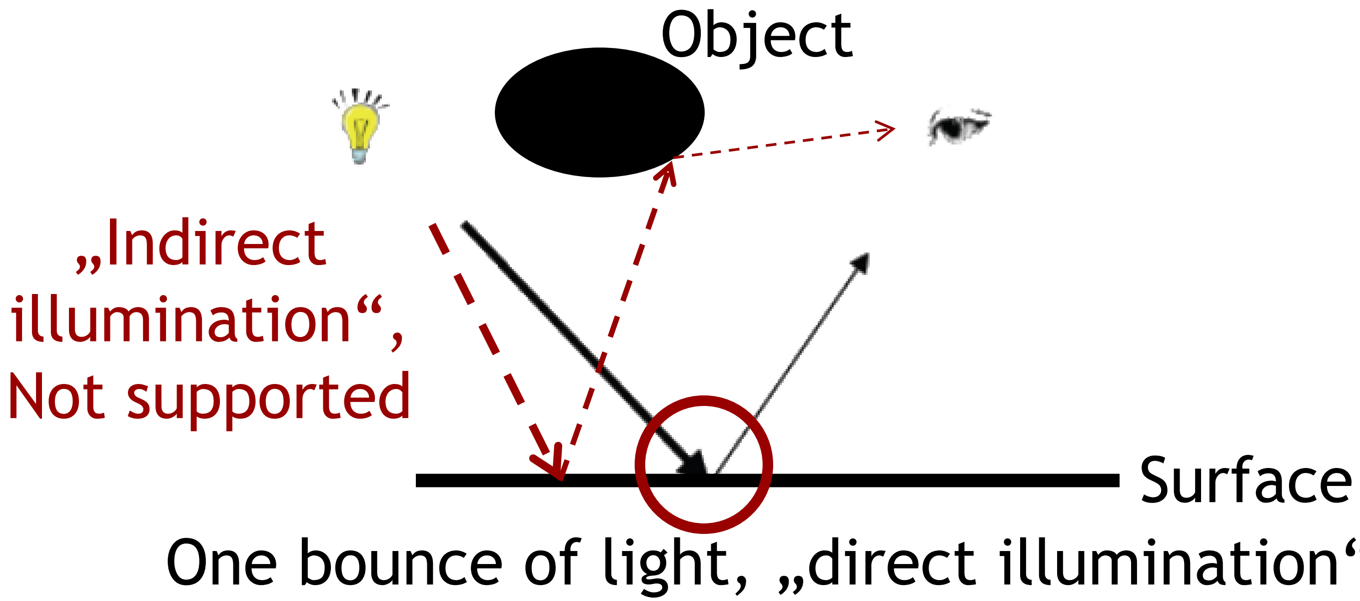


Henrik Wann Jensen

Henrik Wann Jensen

Interactive applications

- Approximations to global illumination possible, but not standard today
- Usually
  - Reproduce perceptually most important effects
  - One bounce of light between light source and viewer
  - “Local/direct illumination”



Today

Shading

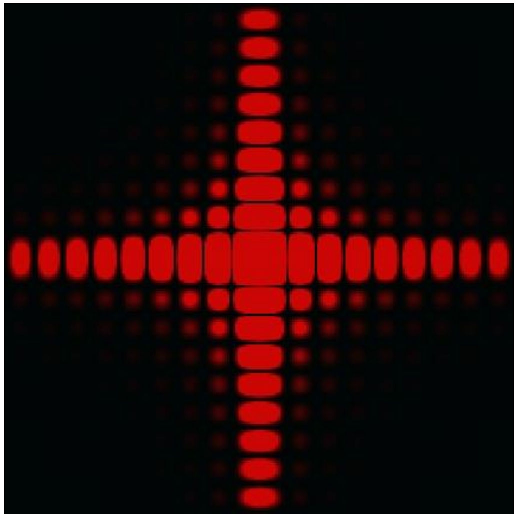
- Introduction
- Radiometry & BRDFs
- Local shading models
- Light sources
- Shading strategies

Material appearance

- What is giving a material its color and appearance?
- How is light reflected by a
  - Mirror
  - White sheet of paper
  - Blue sheet of paper
  - Glossy metal

Radiometry

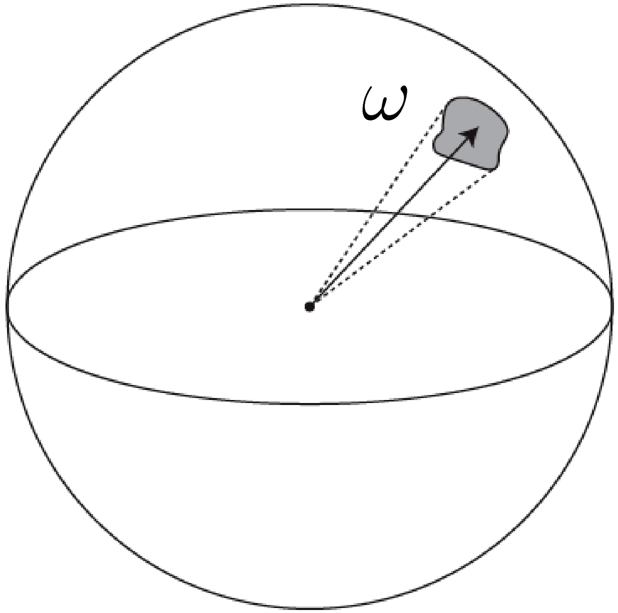
- Physical units to measure light energy
- Based on the geometrical optics model
- Light modeled as rays
  - Rays are idealized narrow beams of light
    http://en.wikipedia.org/wiki/Ray_%28optics%29
  - Rays carry a spectrum of electromagnetic energy
- No wave effects, like interference or diffraction



Diffraction pattern from square aperture

Solid angle

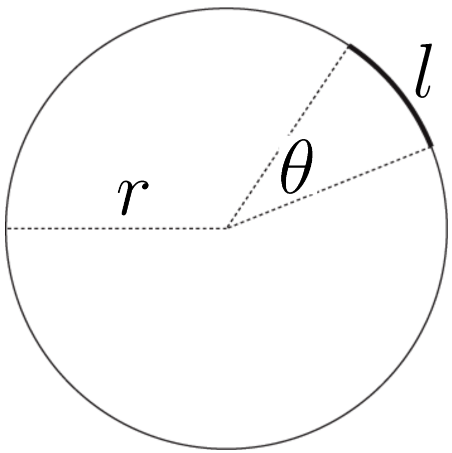
- Area of a surface patch on the unit sphere
  - In our context: area spanned by a set of directions
- Unit: steradian sr
- Directions usually denoted by ω



Unit sphere

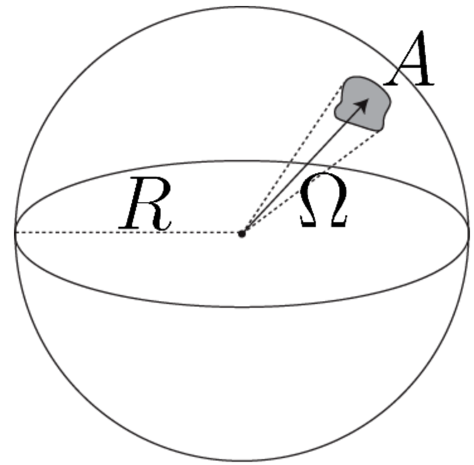
Angle and solid angle

Circle with radius r



- Angle $\theta = l/r$
- Unit circle has 2π radians

Sphere with radius R



- Solid angle $\Omega = A/R^2$
- Unit sphere has 4π steradians

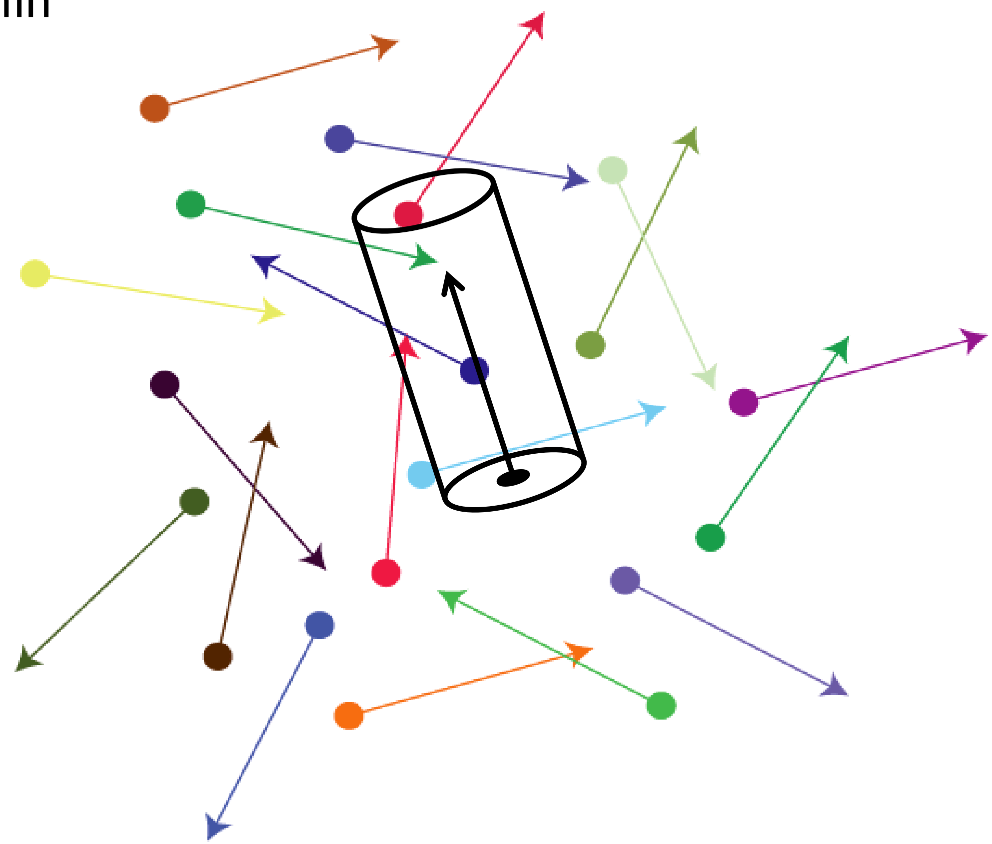
Radiance

http://en.wikipedia.org/wiki/Radiance

- „Energy carried along a narrow beam (ray) of light“
- Energy passing through a small area in a small bundle of directions, divided by area and by solid angle spanned by bundle of directions, in the limit as area and solid angle tend to zero
- Units: energy per area per solid angle $\left[W \cdot sr^{-1} \cdot m^{-2}\right]$

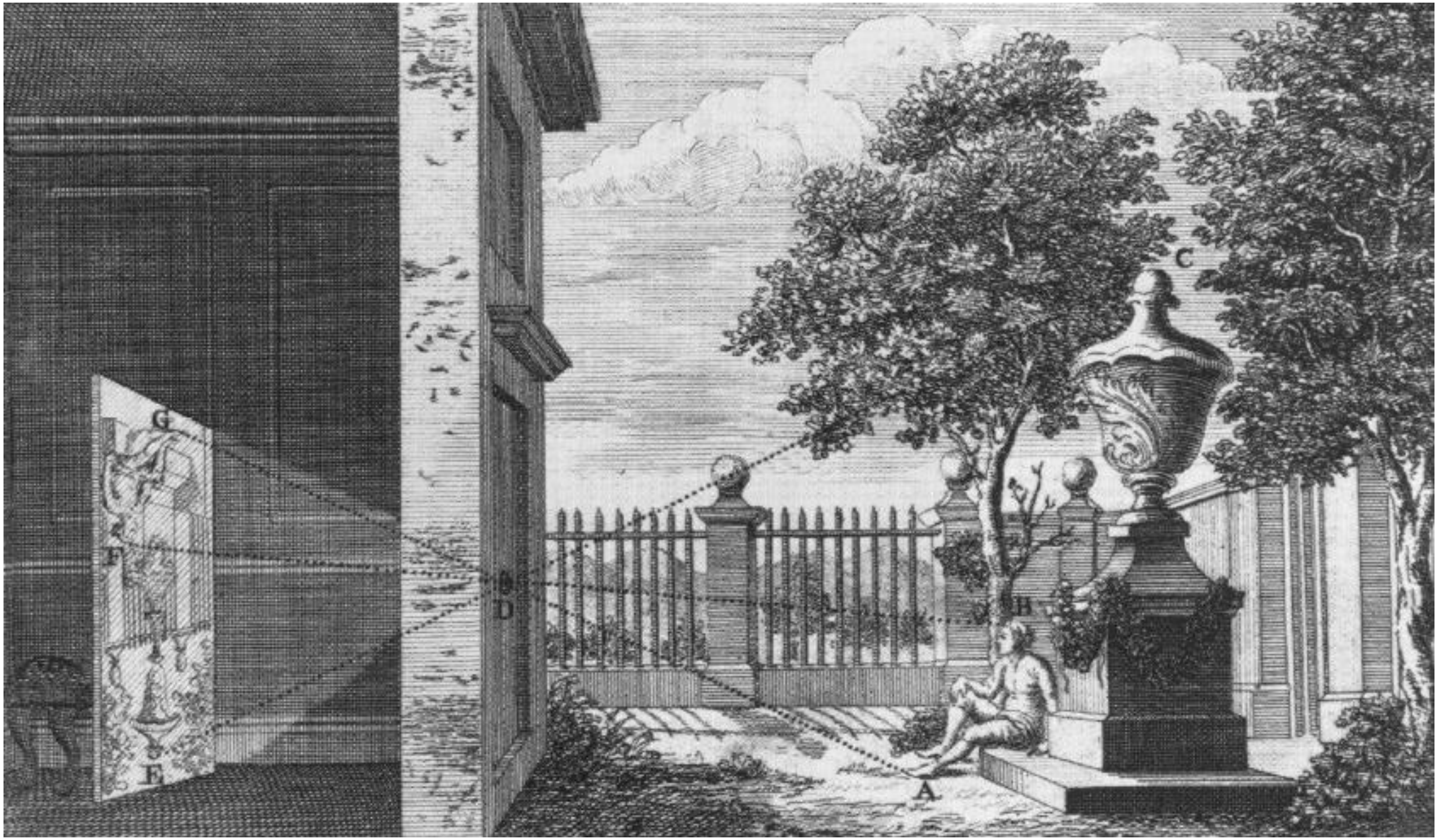
Radiance

- Think of light consisting of photon particles, each traveling along a ray
- Radiance is photon „ray density“
  - Number of photons per area per solid angle
  - „Number of photons passing through small cylinder, as cylinder becomes infinitely thin“



Pinhole camera

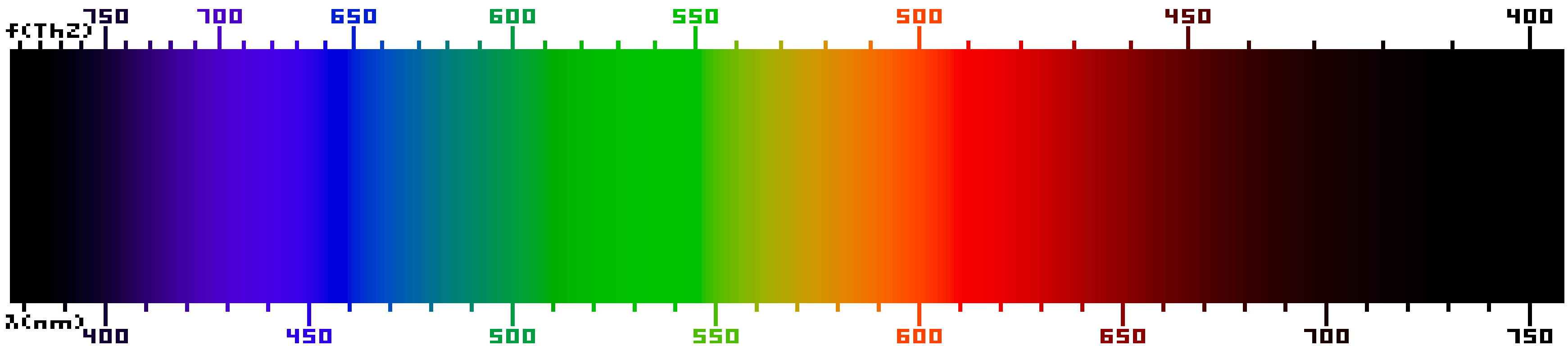
- Records radiance on projection screen

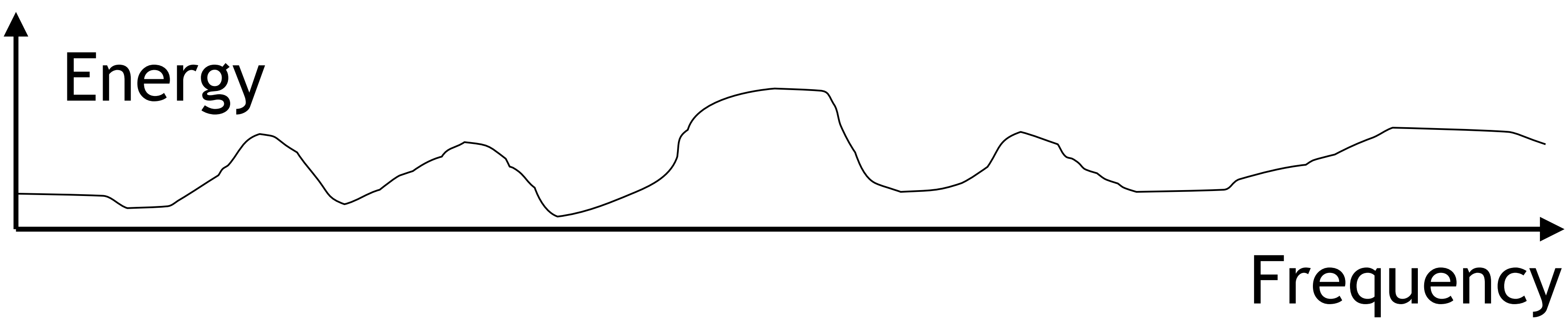


http://en.wikipedia.org/wiki/Pinhole_camera

Radiance

- Spectral radiance: energy at each wavelength/frequency (count only photons of given wavelength)
- Usually, work with radiance for three discrete wavelengths
  - Corresponding to R,G,B primaries

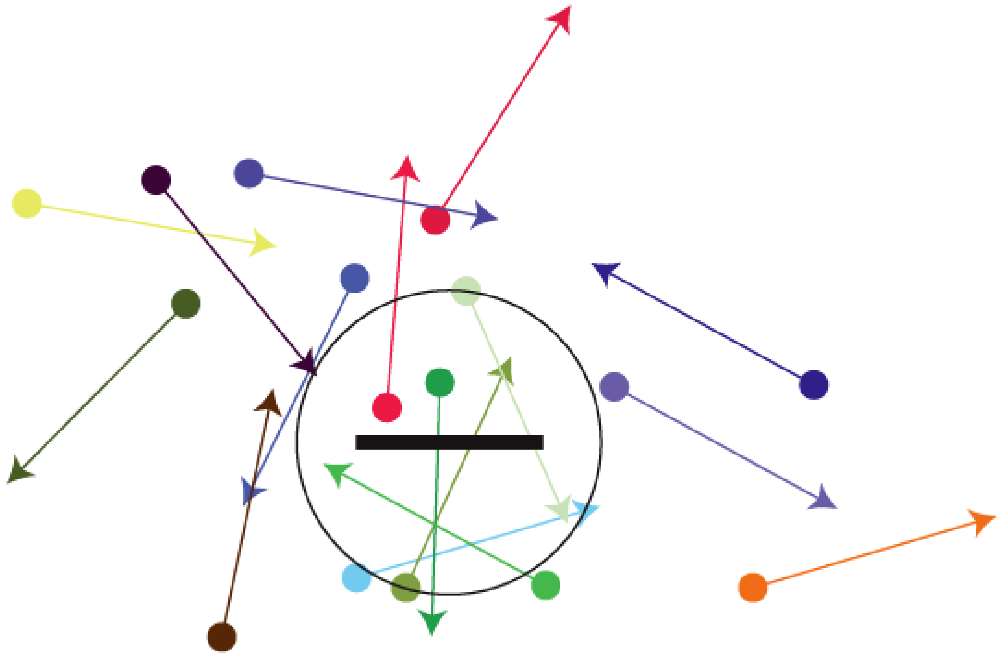




Irradiance

- Energy per area: „energy going through a small area, divided by size of area“
- „Radiance summed up over all directions“
- Units

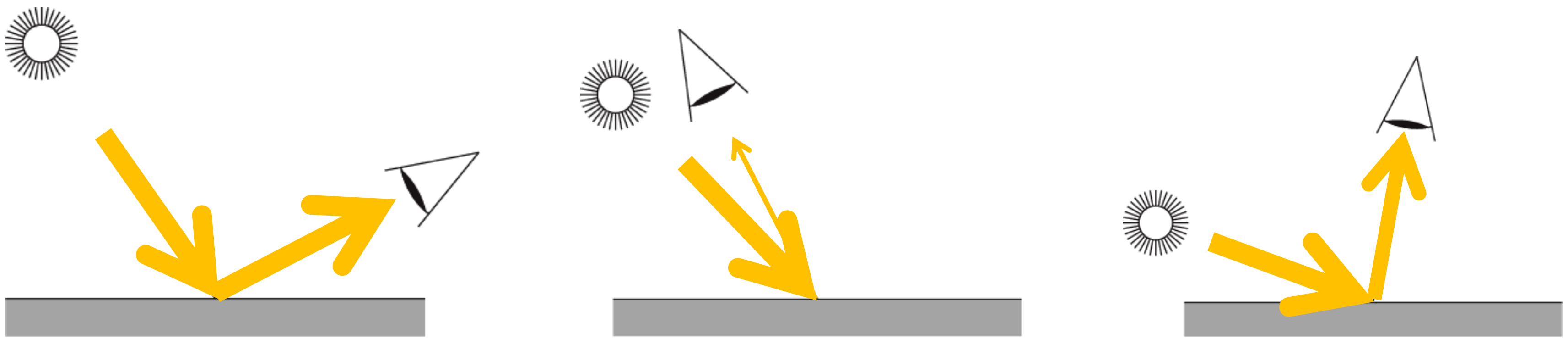
$$\left[W \cdot m^{-2}\right]$$



Irradiance: Count number of photons per area, in the limit as area becomes infinitely small

Local shading

- Goal: model reflection of light at surfaces
- Bidirectional reflectance distribution function (BRDF)
  http://en.wikipedia.org/wiki/Bidirectional_reflectance_distribution_function
  - "Given light direction, viewing direction, obtain fraction of light reflected towards the viewer"
  - For any pair of light/viewing directions!
  - For different wavelenghts (or R, G, B) separately



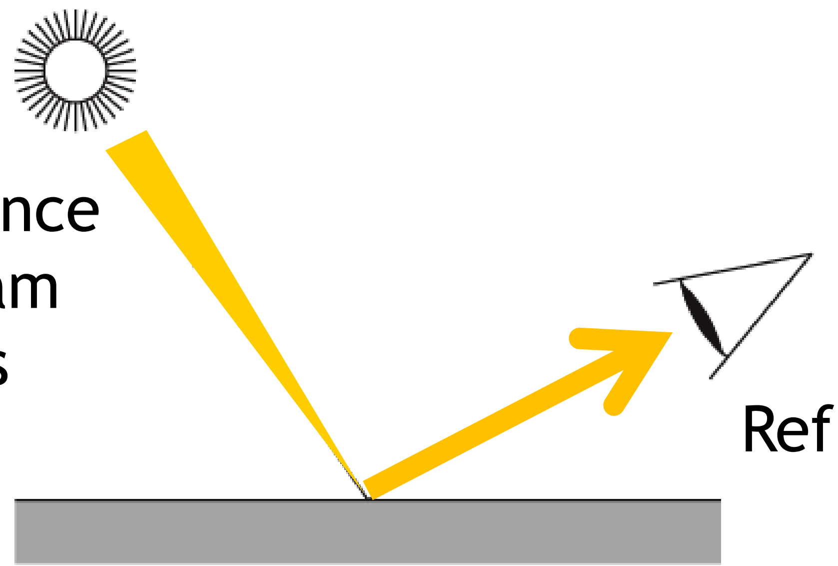
"For each pair of light/view direction, BRDF gives fraction of reflected light"

BRDFs

- BRDF describes appearance of material
  - Color
  - Diffuse
  - Glossy
  - Mirror
  - Etc.
- BRDF can be different at each point on surface
  - Spatially varying BRDF (SVBRDF)
  - Textures

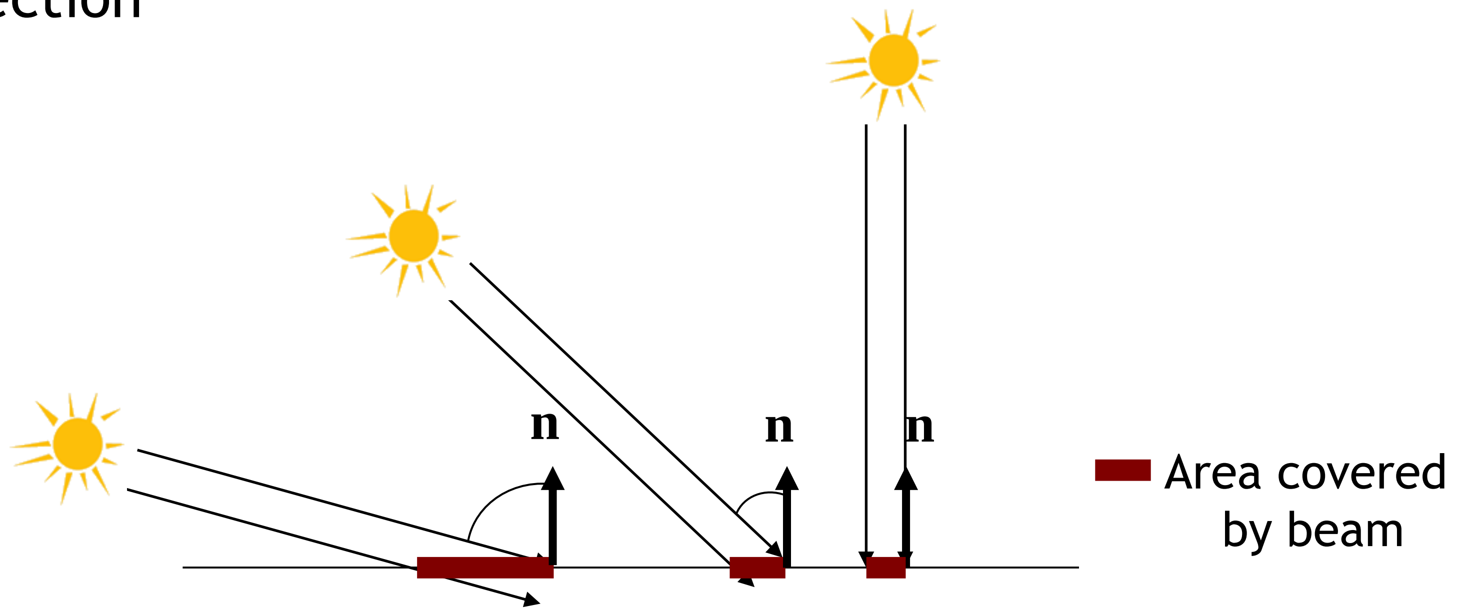
Technical definition

- Given incident and outgoing directions
- BRDF is fraction of ”radiance reflected in outgoing direction” over ”incident irradiance arriving from narrow beam of directions”
- Units $\left[W \cdot sr^{-1} \cdot m^{-2}\right] / \left[W \cdot m^{-2}\right] = sr^{-1}$



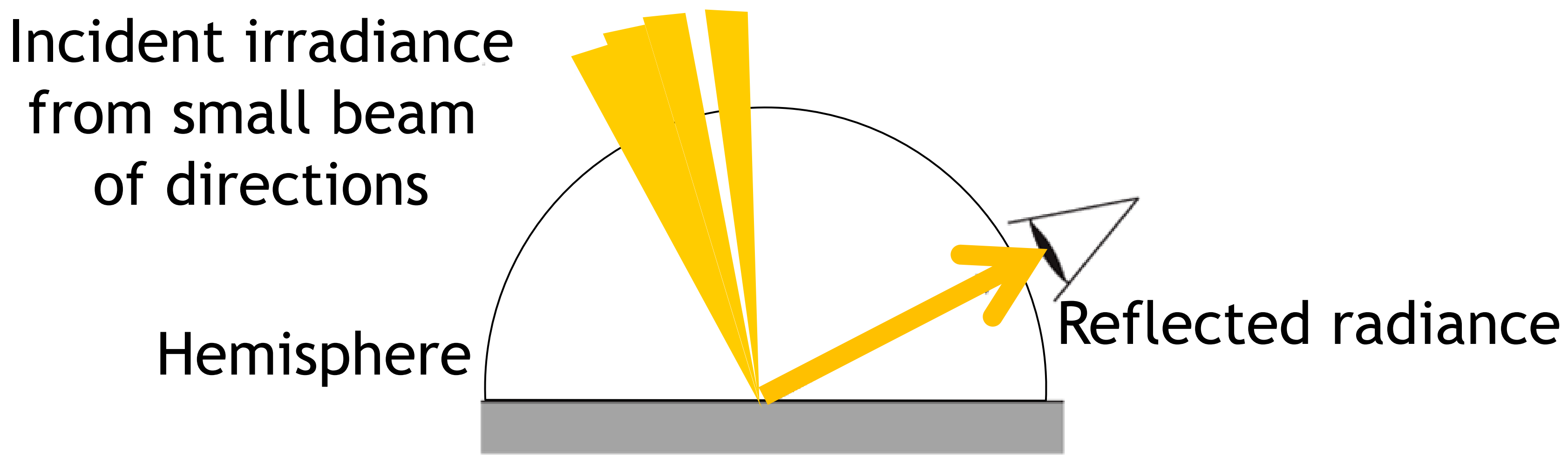
Irradiance from a narrow beam

- Narrow beam of parallel rays shining on a surface
  - Area covered by beam varies with the angle between the beam and the normal **n**
  - The larger the area, the less incident light per area
- Irradiance (incident light per unit area) is proportional to the cosine of the angle between the surface normal **n** and the light rays
- Equivalently, irradiance contributed by beam is radiance of beam times cosine of angle between normal **n** and beam direction



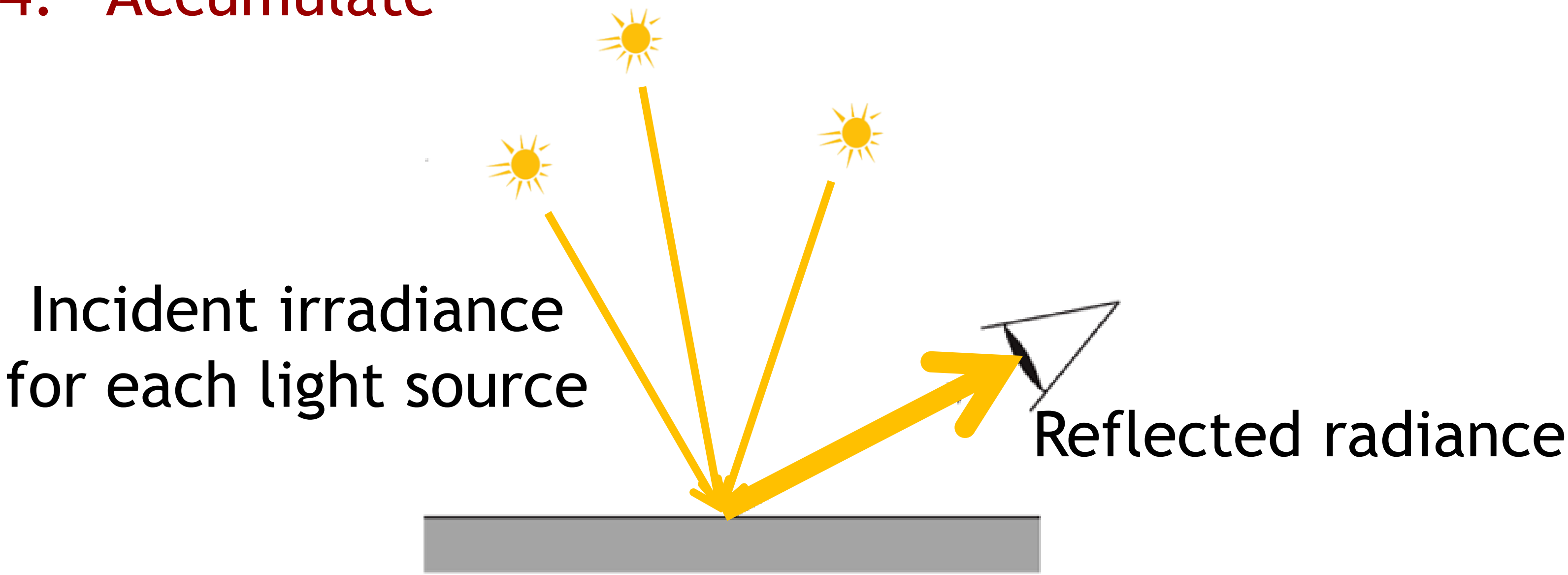
Shading with BRDFs

- Given radiance arriving from each direction, outgoing direction
- For all incoming directions over the hemisphere
  1. Compute irradiance from incoming beam
  2. Evaluate BRDF with incoming beam direction, outgoing direction
  3. Multiply irradiance and BRDF value
  4. Accumulate
- Mathematically, a hemispherical integral (”shading integral”)
  https://en.wikipedia.org/wiki/Rendering_equation



Shading with BRDFs

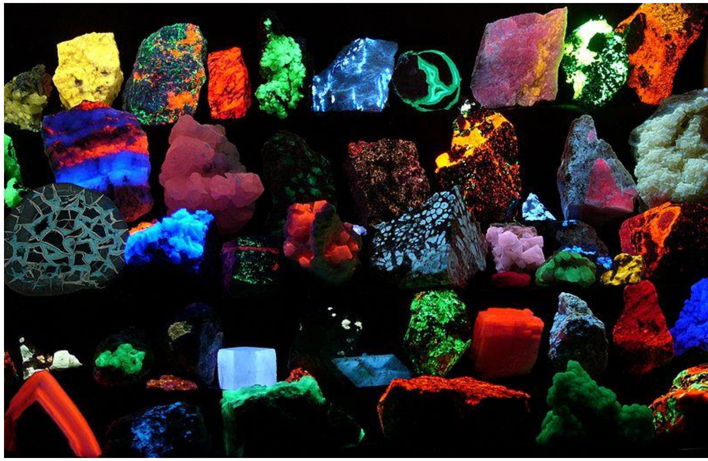
- If only discrete number of small light sources taken into account, need minor modification of algorithm
- For each light source
  1. Compute irradiance arriving from light source
  2. Evaluate BRDF with direction to light source, outgoing direction
  3. Multiply irradiance and BRDF value
  4. Accumulate

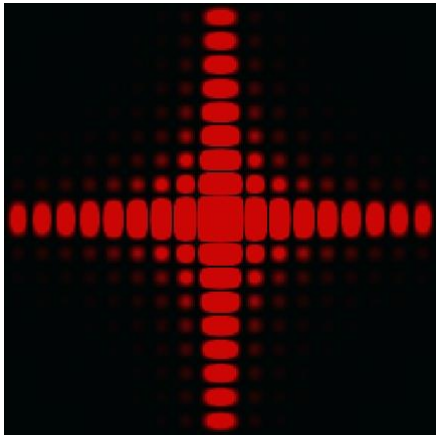


Limitations of BRDF model

Cannot model

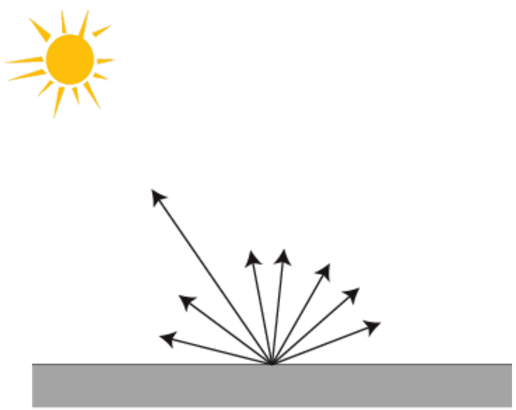
- Fluorescence
- Subsurface and volume scattering
- Polarization
- Interference/diffraction



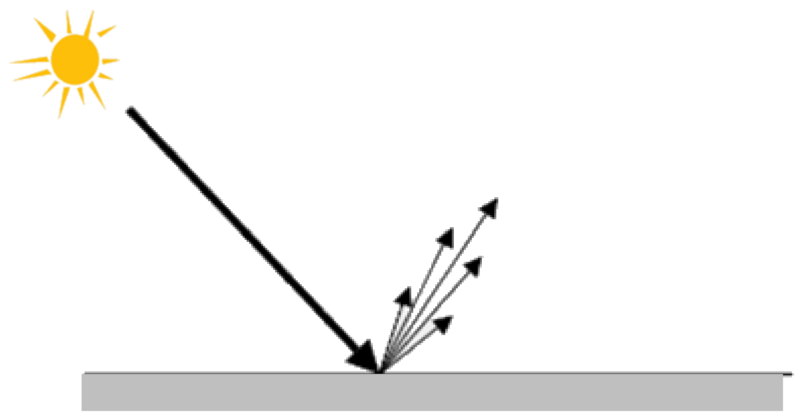


Visualizing BRDFs

- Given viewing or light direction, plot BRDF value over sphere of directions
- Illustration in „flatland“ (1D slices of 2D BRDFs)



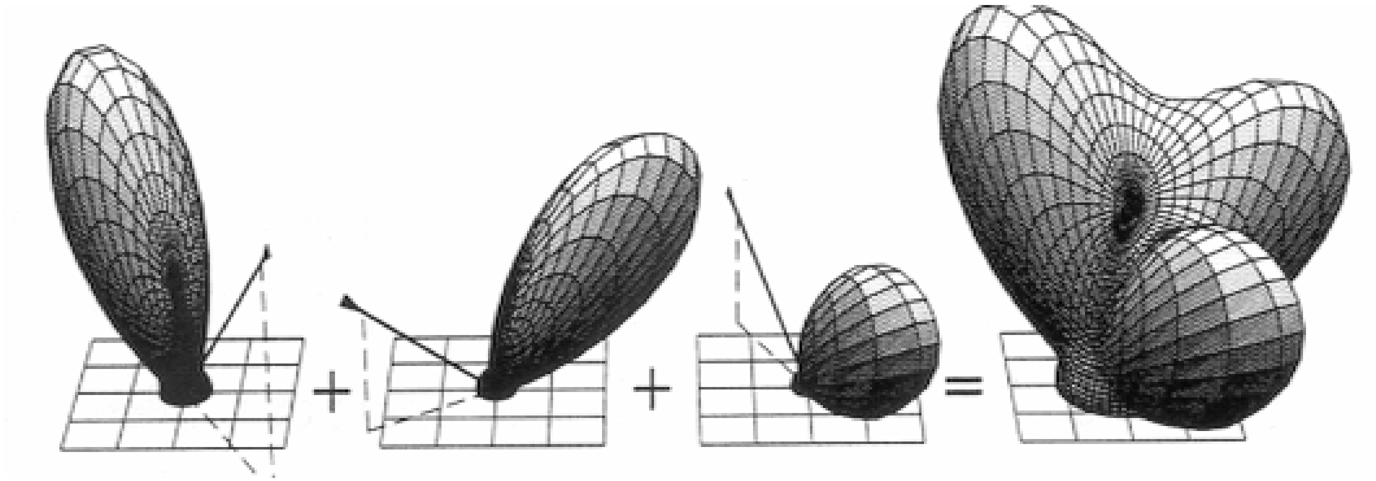
Diffuse reflection



Glossy reflection

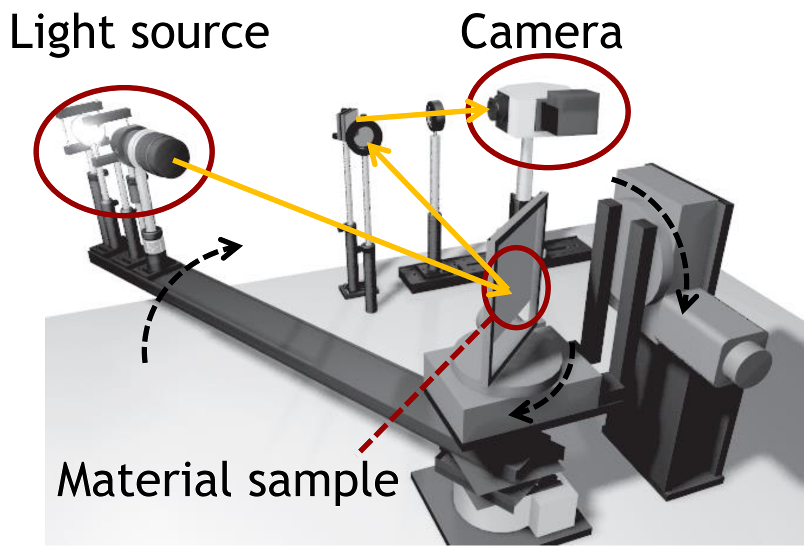
Visualizing BRDFs

- Can add up several BRDFs to obtain more complicated ones



BRDF representation

- How to define and store BRDFs that represent physical materials?
- Physical measurements
  - Gonioreflectometer: robot with light source and camera
  - Measures reflection for each light/camera direction
  - Store measurements in table
- Too much data for interactive application
  - 4 degrees of freedom!



Cornell University Gonioreflectometer

BRDF representation

- Analytical models
  - Try to describe phyiscal properties of materials using mathematical expressions
- Many models proposed in graphics
  http://en.wikipedia.org/wiki/Bidirectional_reflectance_distribution_function
  http://en.wikipedia.org/wiki/Cook-Torrance
  http://en.wikipedia.org/wiki/Oren-Nayar_diffuse_model
- Will restrict ourselves to simple models here

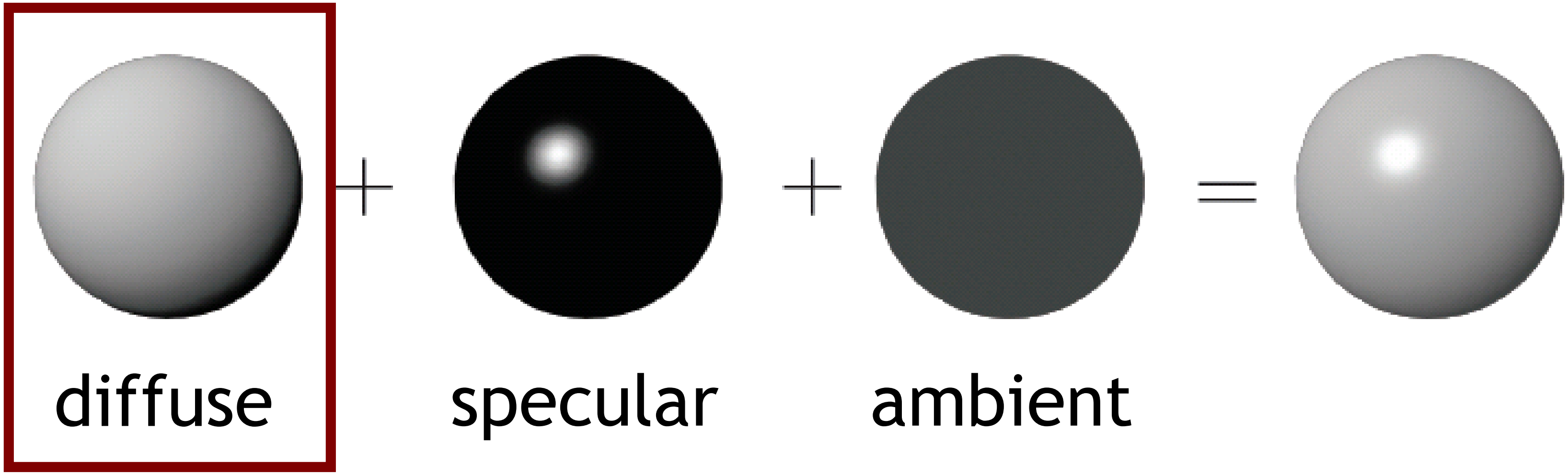
Today

Shading

- Introduction
- Radiometry & BRDFs
- Local shading models
- Light sources
- Shading strategies

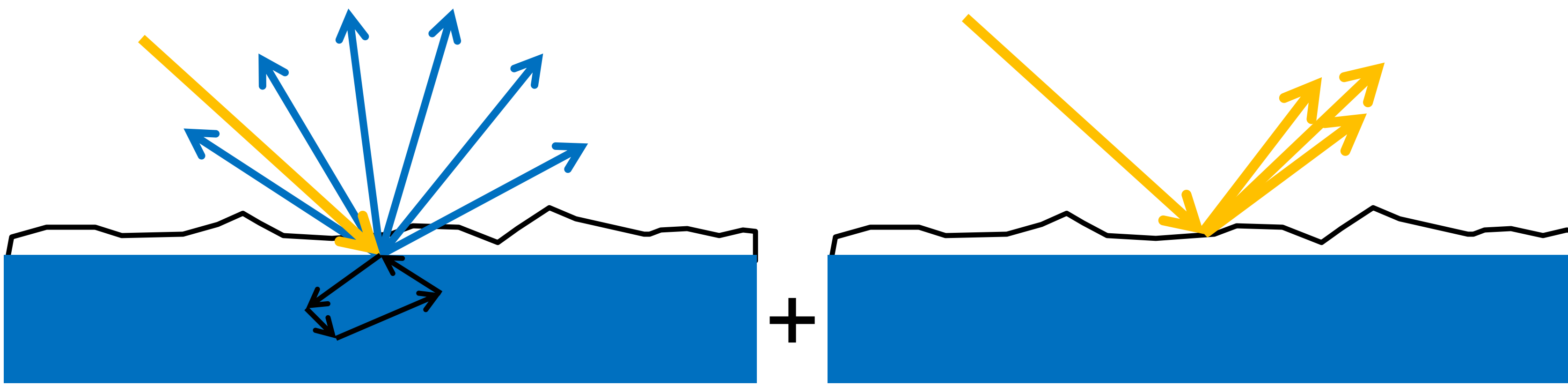
Simplified model

- BRDF is sum of diffuse, specular, and ambient components
  - Covers a large class of real surfaces
  - Each is simple analytical function
- Incident light from discrete set of light sources (discrete set of directions)
- Model is not completely physically justified!

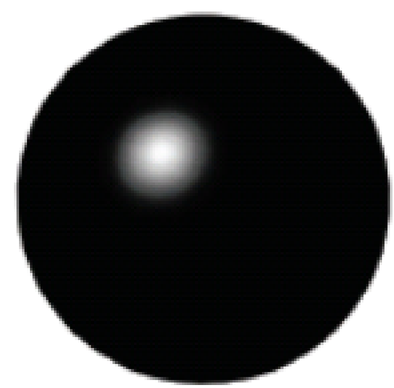


Simplified physical model

- Approximate model for two-layer materials
- Subsurface scattering leading to diffuse reflection on bottom layer
- Mirror reflection on (rough) top layer



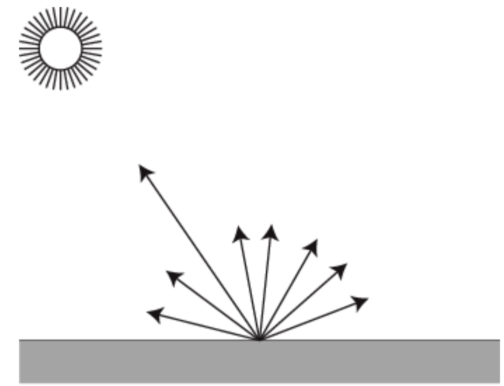
diffuse



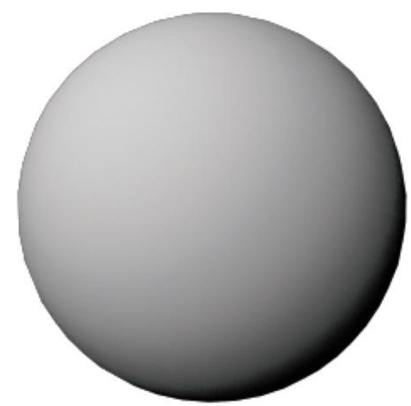
specular

Diffuse reflection

- Ideal diffuse material reflects light equally in all directions
  - Also called Lambertian surfaces
    http://en.wikipedia.org/wiki/Lambert's_cosine_law
- View-independent
  - Surface looks the same independent of viewing direction
- Matte, not shiny materials
  - Paper
  - Unfinished wood
  - Unpolished stone



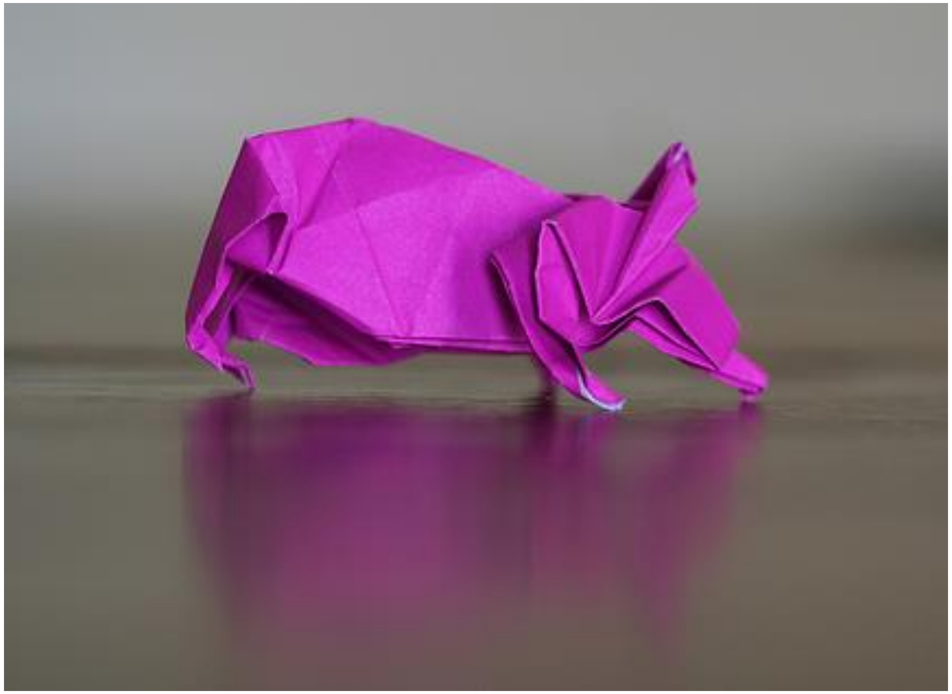
Diffuse reflection



Diffuse sphere

Diffuse reflection

- “Radiance reflected by a diffuse (“Lambertian”) surface is constant over all directions“
- Hm, why do we see brightness variations over diffuse surfaces ?

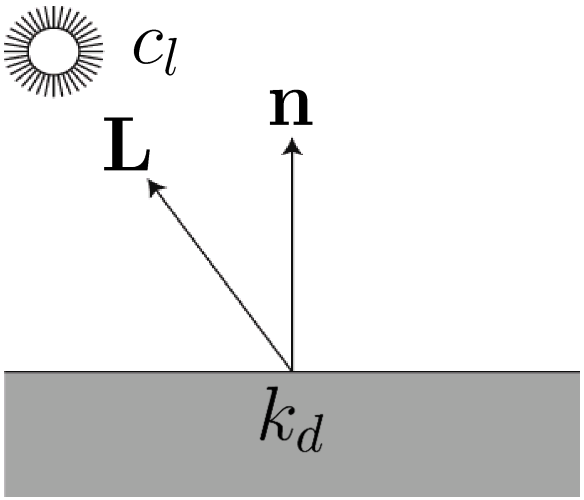


Diffuse reflection

- Given
  - Light color (radiance) c_l
  - Unit surface normal $\mathbf{n}$
  - One light source, unit light direction $\mathbf{L}$
  - Material diffuse reflectance (material color) k_d
- Diffuse reflection c_d

$$c_d = c_l \underbrace{(\mathbf{n} \cdot \mathbf{L})} k_d$$

Cosine between
normal and light,
converts radiance to incident irradiance



Diffuse reflection

Notes on $c_d = c_l(\mathbf{n} \cdot \mathbf{L})k_d$

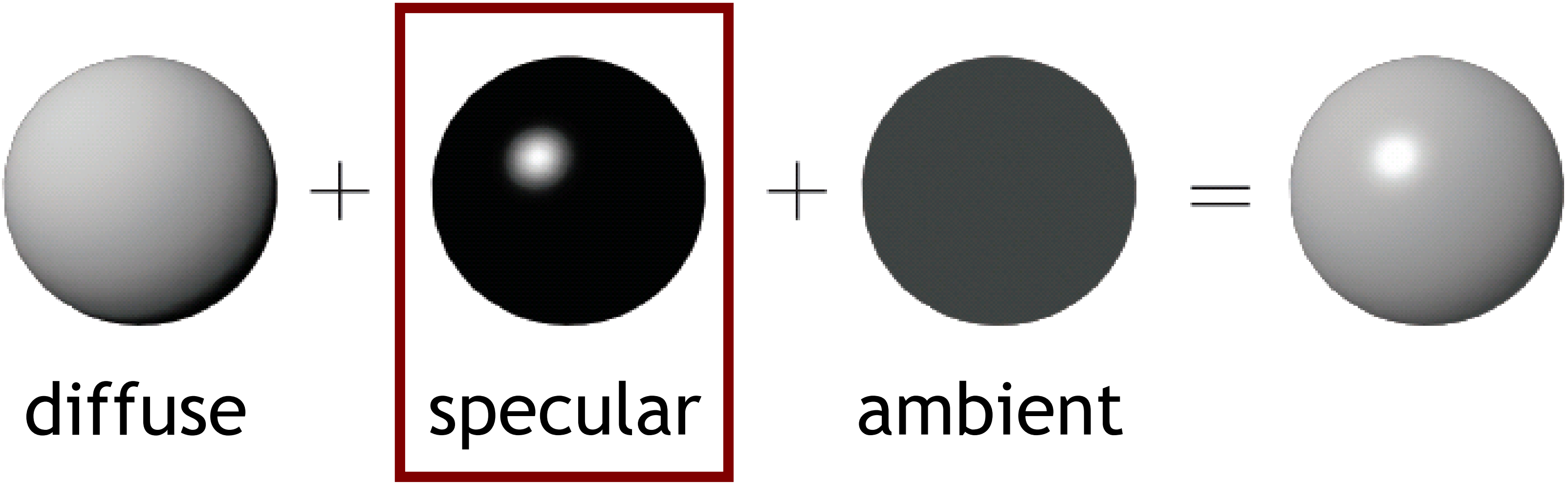
- Parameters k_d, c_l are r,g,b vectors
- c_l: radiance of light source
- $c_l(\mathbf{n} \cdot \mathbf{L})$: irradiance on surface
- k_d is diffuse BRDF, a constant!
- Compute r,g,b values of reflected color c_d separately

Diffuse reflection

- Provides visual cues
  - Surface curvature
  - Depth variation

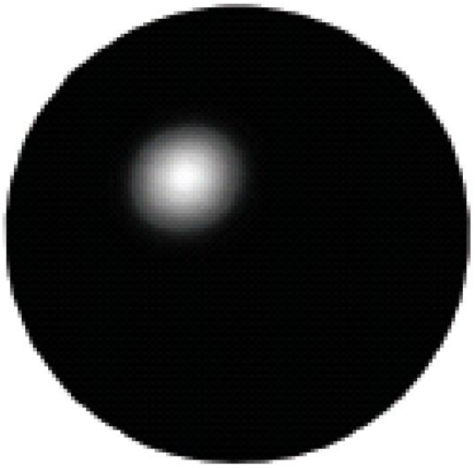
Lambertian (diffuse) sphere under different lighting directions

Simplified model



Specular reflection

- Shiny or glossy surfaces
  - Polished metal
  - Glossy car finish
  - Plastics
- Specular highlight
  - Blurred reflection of the light source
  - Position of highlight depends on viewing direction

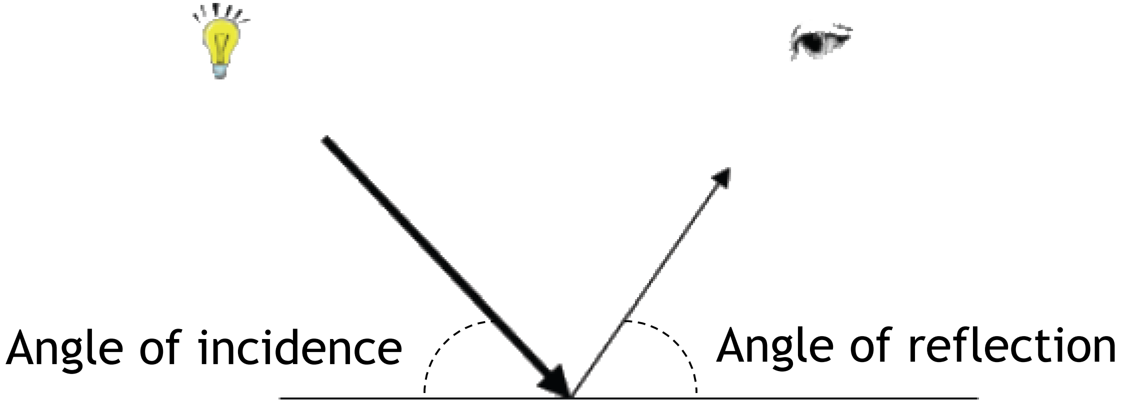


Sphere with specular highlight

Shiny or glossy materials

Specular reflection

- Ideal specular reflection is mirror reflection
  - Perfectly smooth surface
  - Incoming light ray is bounced in single direction
  - Angle of incidence equals angle of reflection

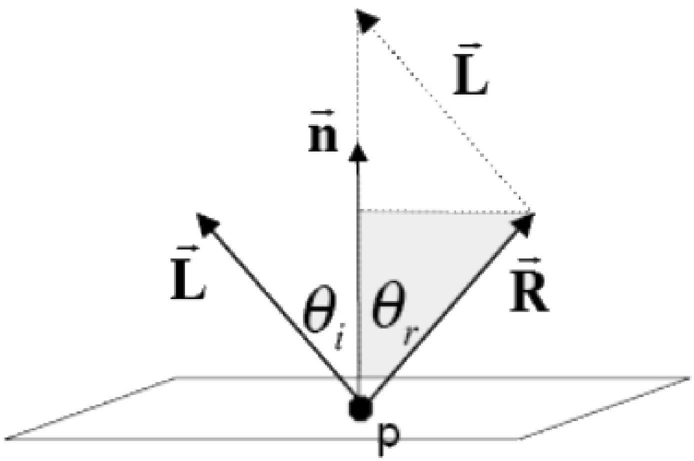


Law of reflection

- "Angle of incidence equals angle of reflection" applied to 3D vectors **L** and **R**
- Equation expresses constraints:
  1. Normal, incident, and reflected direction all in same plane (**L**+**R** is a point along the normal)
  2. Angle of incidence θ_i = angle of reflection θ_r

$$\vec{\mathbf{R}} + \vec{\mathbf{L}} = 2\cos\theta\ \vec{\mathbf{n}} = 2\left(\vec{\mathbf{L}} \cdot \vec{\mathbf{n}}\right)\vec{\mathbf{n}}$$

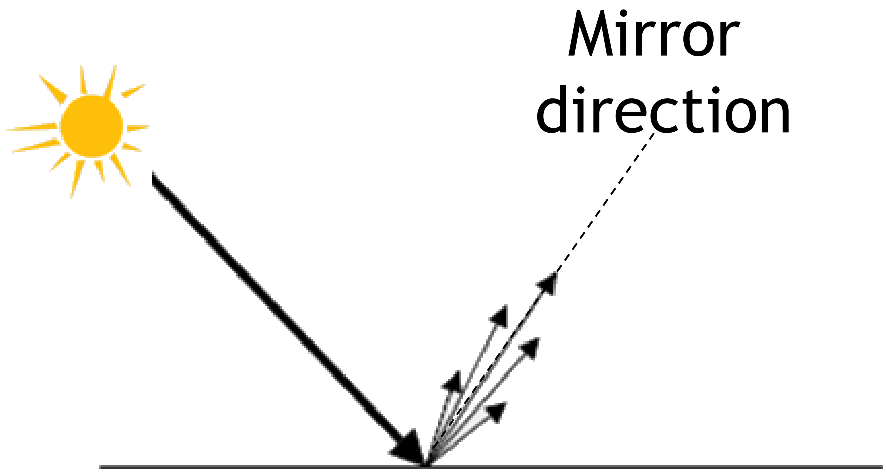
$$\vec{\mathbf{R}} = 2\left(\vec{\mathbf{L}} \cdot \vec{\mathbf{n}}\right)\vec{\mathbf{n}} - \vec{\mathbf{L}}$$



$$\theta_r = \theta_i$$

Glossy materials

- Many materials not quite perfect mirrors
- Glossy materials have blurry reflection of light source



Glossy teapot with highlights from many light sources

Physical model

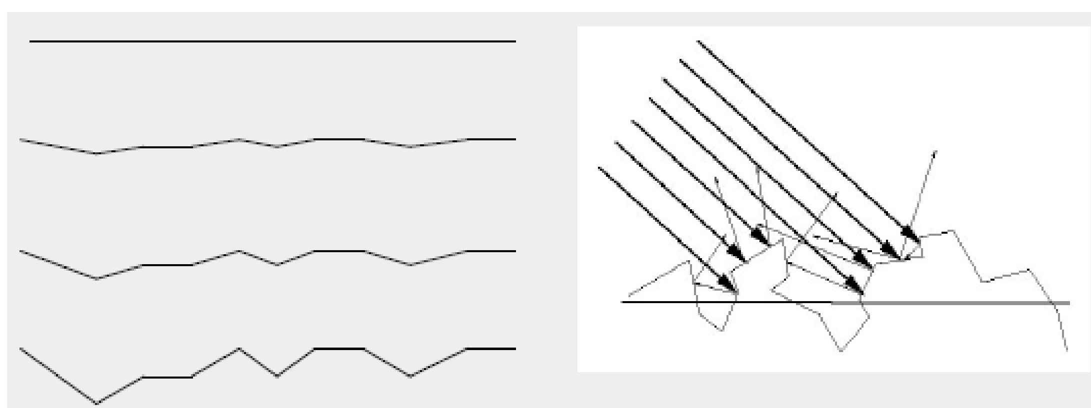
- Assume surface composed of small mirrors with random orientation (microfacets)
- Smooth surfaces
  - Microfacet normals close to surface normal
  - Sharp highlights
- Rough surfaces
  - Microfacet normals vary strongly
  - Leads to blurry highlight

Polished

Smooth

Rough

Very rough



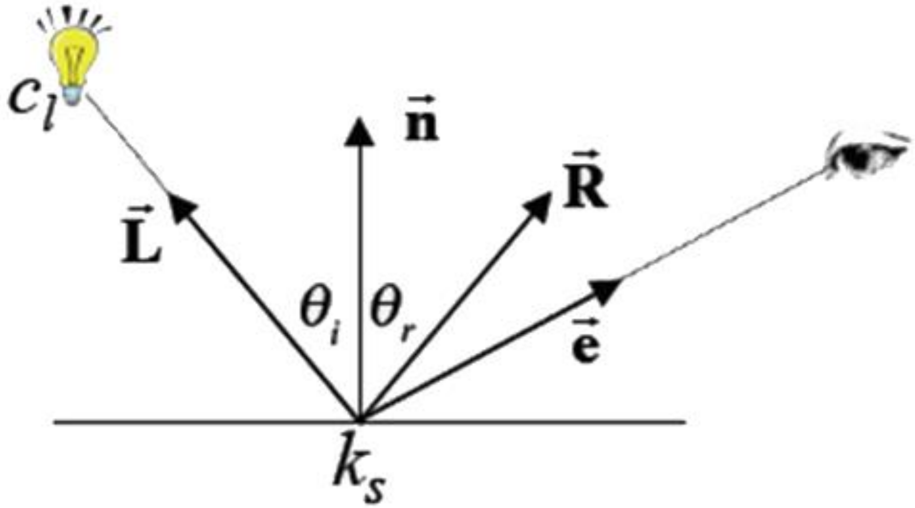
Physical model

- Expect most light to be reflected in mirror direction
- Because of microfacets, some light is reflected slightly off ideal reflection direction
- Reflection
  - Brightest when view vector is aligned with reflection
  - Decreases as angle between view vector and reflection direction increases

Phong model

http://en.wikipedia.org/wiki/Phong_shading

- Simple "implementation" of the physical model
- Radiance of light source c_l
- Specular reflectance coefficient k_s
- Phong exponent p
  - Higher p, smaller (sharper) highlight



Reflected radiance

$$c = c_l k_s \left(\mathbf{R} \cdot \mathbf{e}\right)^p$$

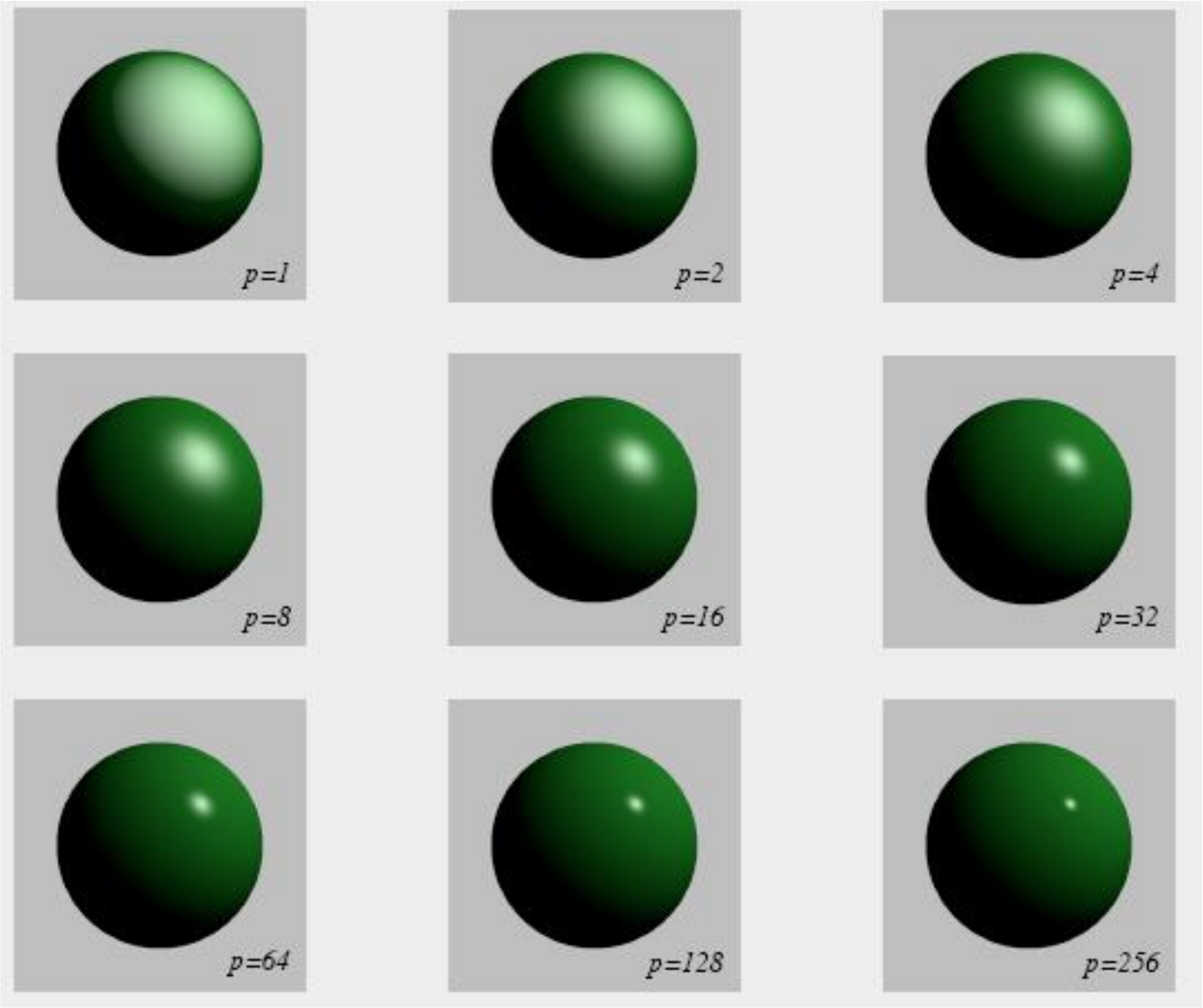
Note

- Technically, Phong „BRDF“ is

$$c = \underbrace{c_l \left(\cancel{\mathbf{n} \cdot \mathbf{L}}\right)}_{\text{Irradiance}} \underbrace{\frac{k_s \left(\mathbf{R} \cdot \mathbf{e}\right)^p}{\cancel{\mathbf{n} \cdot \mathbf{L}}}}_{\text{„BRDF“}}$$

- Phong model is not usually considered a BRDF, because it violates energy conservation
http://en.wikipedia.org/wiki/Bidirectional_reflectance_distribution_function#Physically_based_BRDFs

Phong model

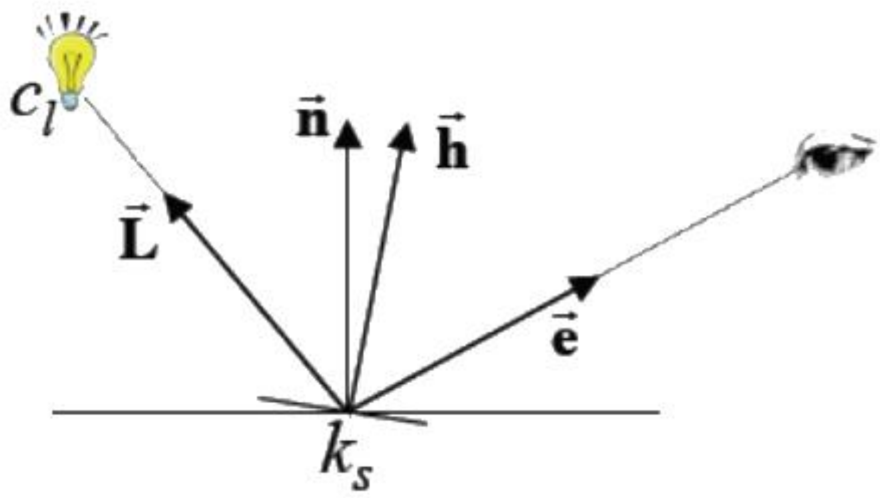


Blinn model (Jim Blinn, 1977)

- Alternative to Phong model
- Define unit halfway vector

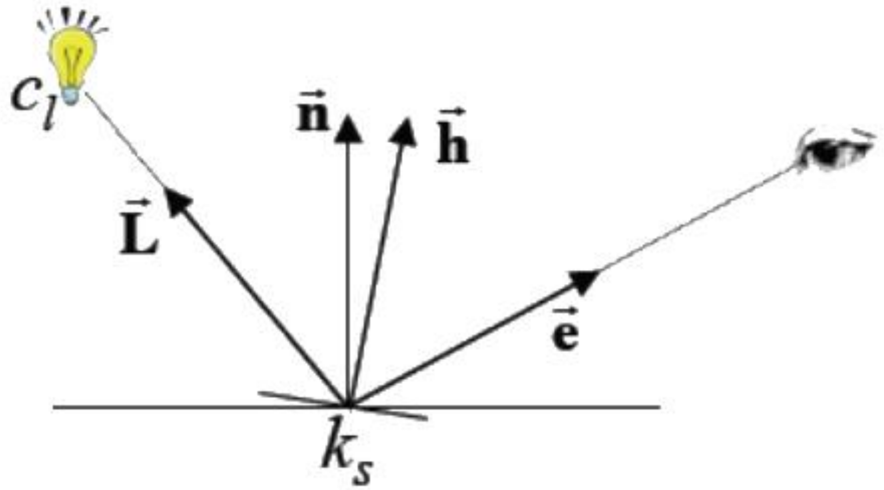
$$\mathbf{h} = \frac{\mathbf{L} + \mathbf{e}}{\|\mathbf{L} + \mathbf{e}\|}$$

- Halfway vector represents normal of microfacet that would lead to mirror reflection to the eye



Blinn model

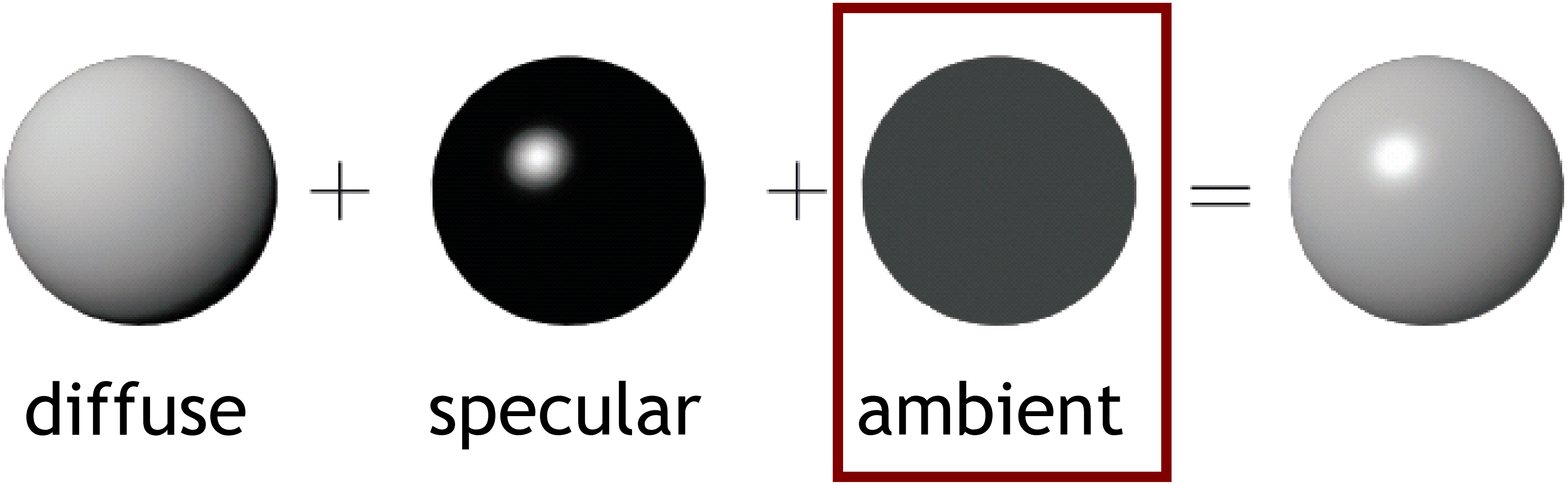
- The larger the angle between microfacet orientation and normal, the less likely
- Use cosine of angle between them
- Shininess parameter s
- Very similar to Phong



Reflected radiance

$$c = c_l k_s \left(\mathbf{h} \cdot \mathbf{n}\right)^s$$

Simplified model



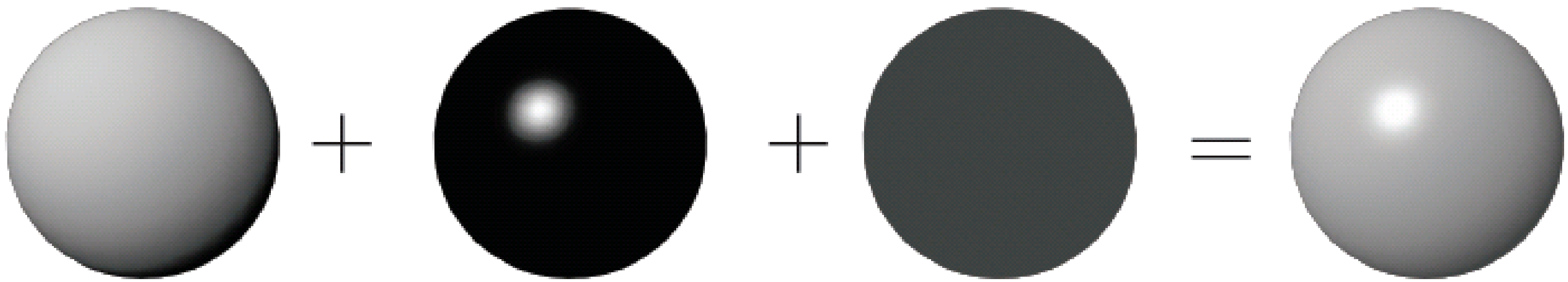
Ambient light

- In real world, light is bounced all around scene
- Could use global illumination techniques to simulate
- Simple approximation
  - Add constant ambient light at each point $k_a c_a$
  - Ambient light c_a
  - Ambient reflection coefficient k_a
- Areas with no direct illumination are not completely dark

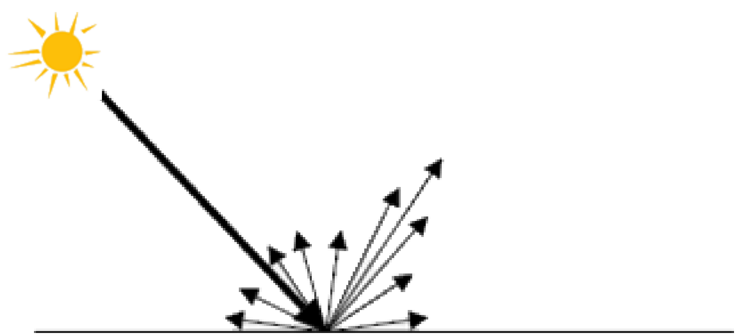
Complete model

- Blinn model with several light sources i

$$c = \sum_i c_{l_i} \left(k_d \left(\mathbf{L}_i \cdot \mathbf{n} \right) + k_s \left(\mathbf{h}_i \cdot \mathbf{n} \right)^s \right) + k_a c_a$$



diffuse | specular | ambient



Notes

$$c = \sum_i c_{l_i} \left(k_d \left(\mathbf{L}_i \cdot \mathbf{n} \right) + k_s \left(\mathbf{h}_i \cdot \mathbf{n} \right)^s \right) + k_a c_a$$

- All colors, reflection coefficients have separate values for R,G,B
- Usually, ambient = diffuse coefficient
- For metals, specular = diffuse coefficient
  - Highlight is color of material
- For plastics, specular coefficient = (x,x,x)
  - Highlight is color of light

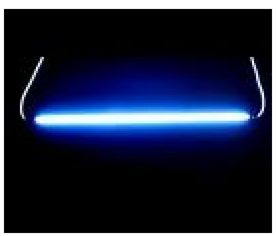
Today

Shading

- Introduction
- Radiometry & BRDFs
- Local shading models
- Light sources
- Shading strategies

Light sources

- Light sources can have complex properties
  - Geometric area over which light is produced
  - Anisotropy in direction
  - Variation in color
  - Reflective surfaces act as light sources

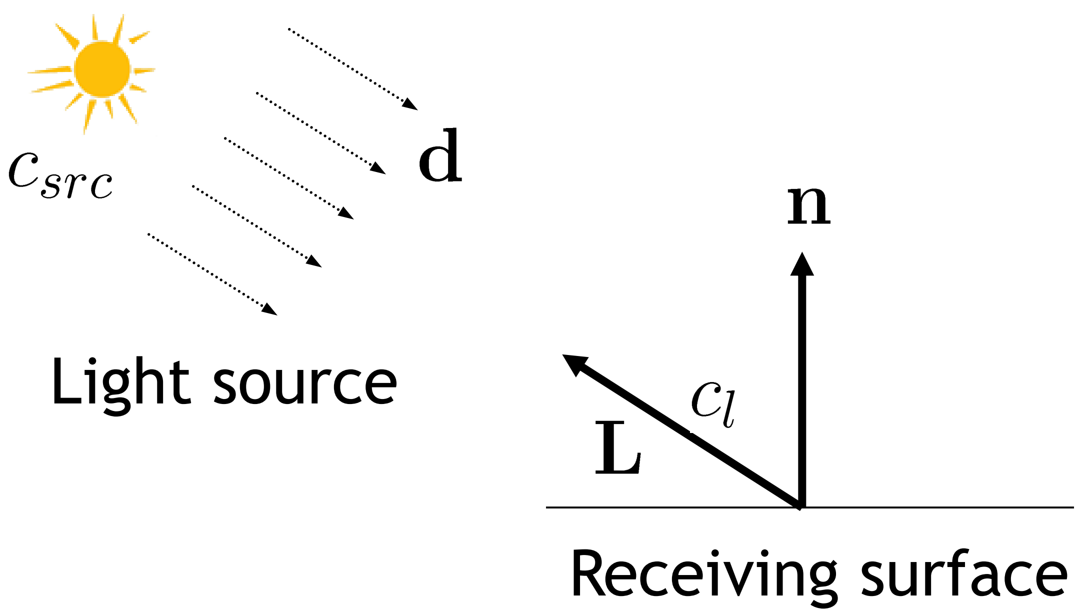
- Interactive rendering is based on simple, standard light sources

Light sources

- At each point on surfaces need to know
  - Direction of incoming light (the $\mathbf{L}$ vector)
  - Radiance of incoming light (the c_l values)
- Standard, simplified light sources
  - Directional: from a specific direction
  - Point light source: from a specific point
  - Spotlight: from a specific point with intensity that depends on the direction
- No model for light sources with an area!

Directional light

- Light from a distant source
  - Light rays are parallel
  - Direction and radiance same everywhere in 3D scene
  - As if the source were infinitely far away
  - Good approximation to sunlight
- Specified by a unit length direction vector, and a color



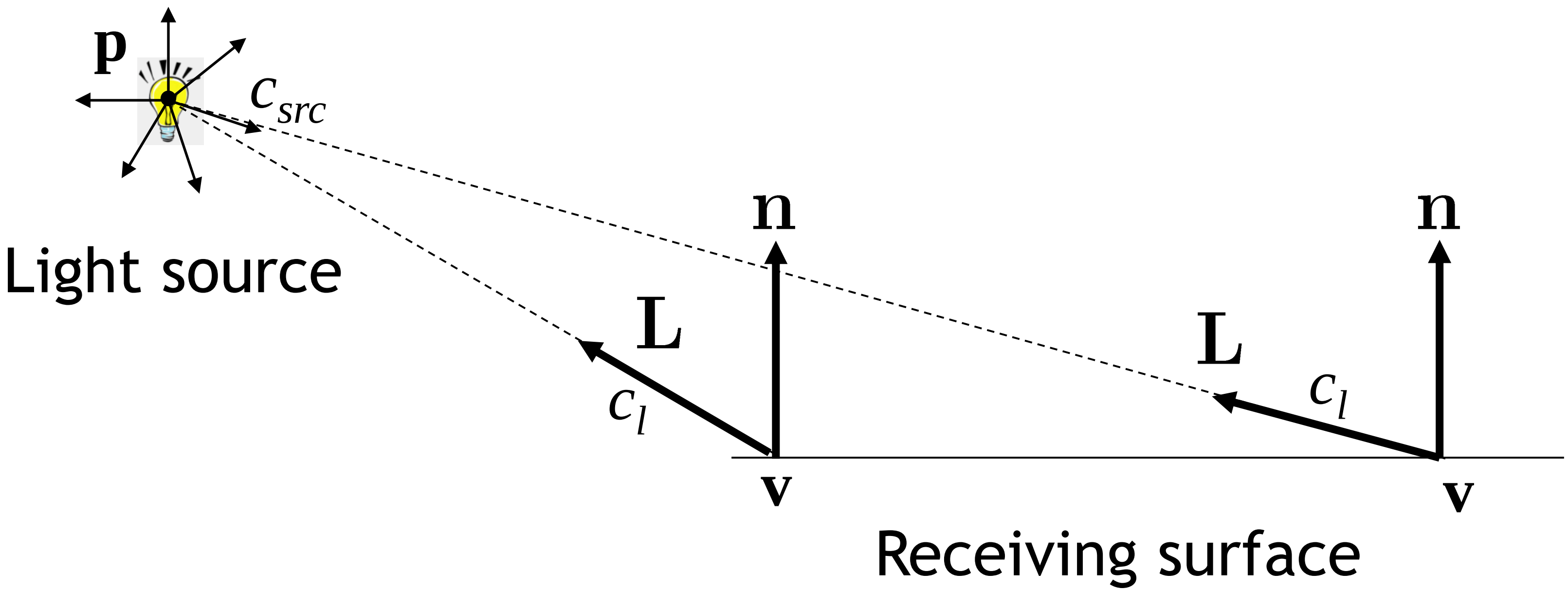
$$\mathbf{L} = -\mathbf{d}$$

$$c_l = c_{src}$$

Point lights

- Simple model for light bulbs
- Infinitesimal point that radiates light in all directions equally
  - Light vector varies across the surface
  - Radiance drops off proportionally to the inverse square of the distance from the light
  - Intuition for inverse square falloff?
- Not physically plausible!

Point lights



Incident light direction $$\mathbf{L} = \frac{\mathbf{p} - \mathbf{v}}{\|\mathbf{p} - \mathbf{v}\|}$$

Radiance $$c_l = \frac{c_{src}}{\|\mathbf{p} - \mathbf{v}\|^2}$$

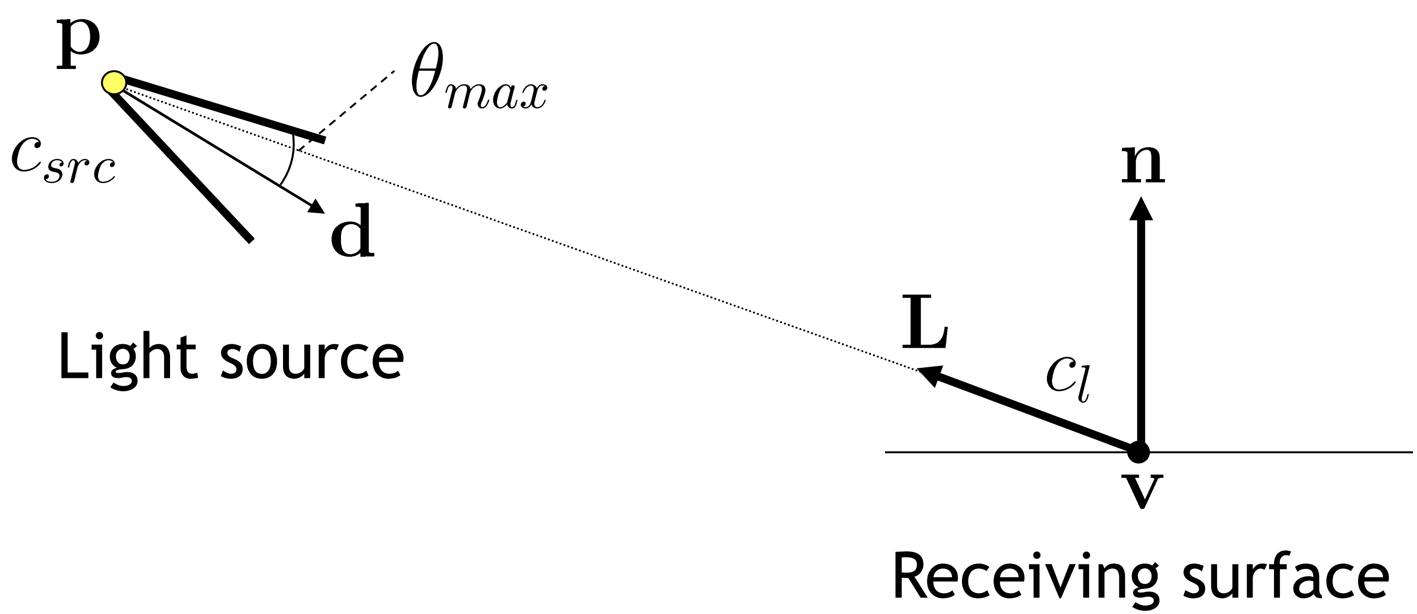
Spotlights

- Like point source, but radiance depends on direction

Parameters

- Position, the location of the source
- Spot direction, the center axis of the light
- Falloff parameters
  - how broad the beam is (cone angle θ_{max})
  - how light tapers off at edges of he beam (cosine exponent f)

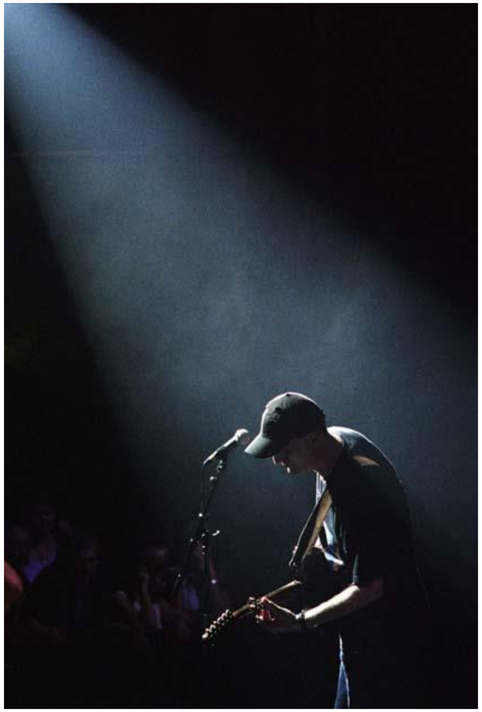
Spotlights



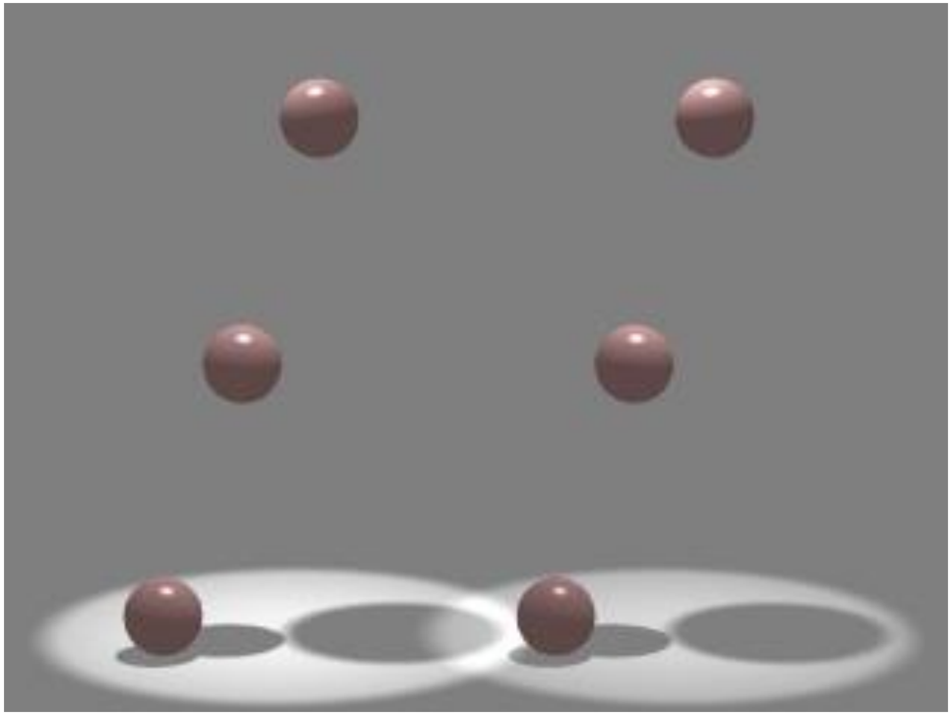
$$\mathbf{L} = \frac{\mathbf{p} - \mathbf{v}}{\|\mathbf{p} - \mathbf{v}\|}$$

$$c_l = \begin{cases} 0 & \text{if} \quad -\mathbf{L} \cdot \mathbf{d} \leq \cos(\theta_{max}) \\ c_{src} \, (-\mathbf{L} \cdot \mathbf{d})^f & \text{otherwise} \end{cases}$$

Spotlights



Photograph of spotlight



Spotlights in OpenGL

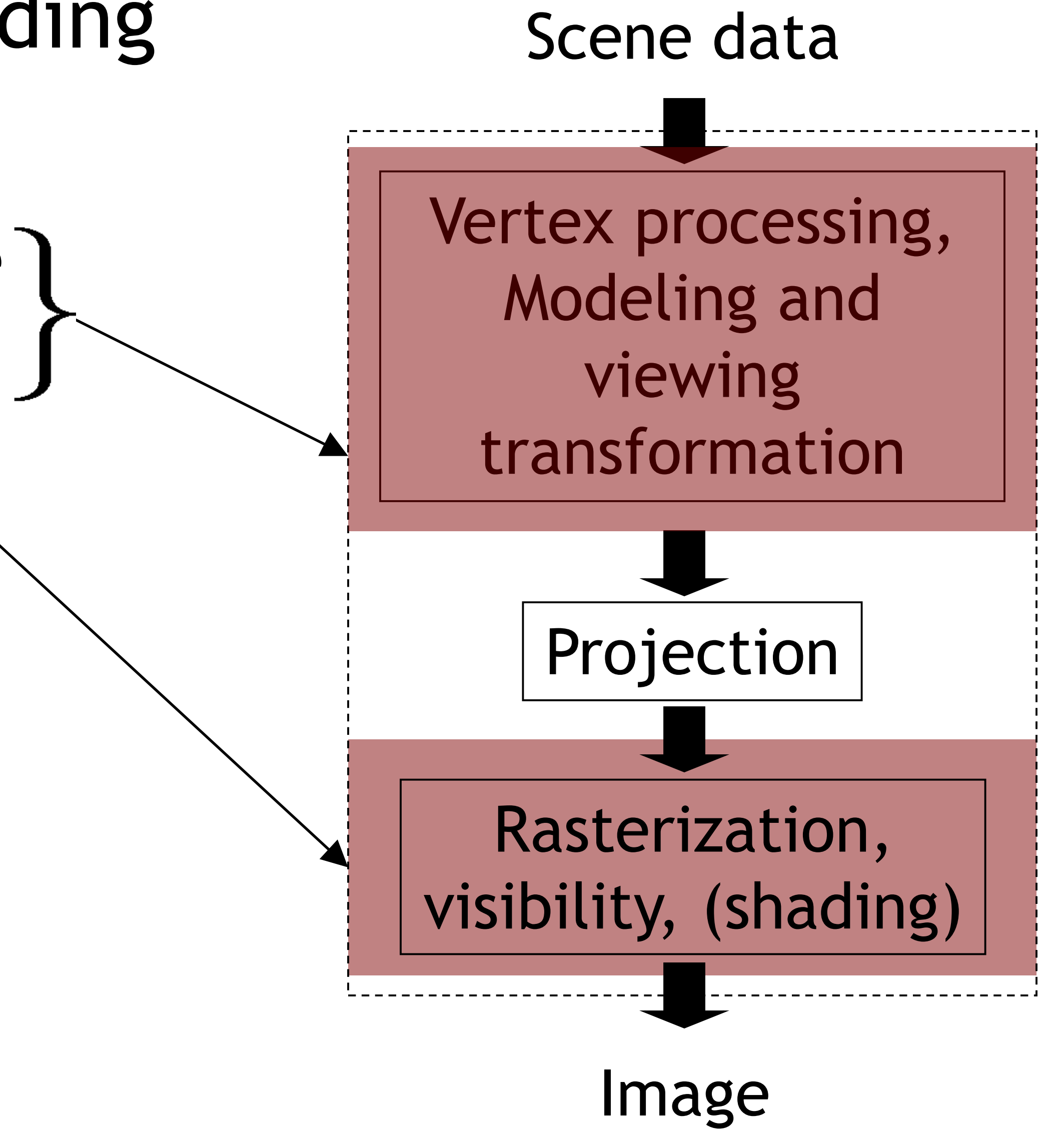
Today

Shading

- Introduction
- Radiometry & BRDFs
- Local shading models
- Light sources
- Shading strategies

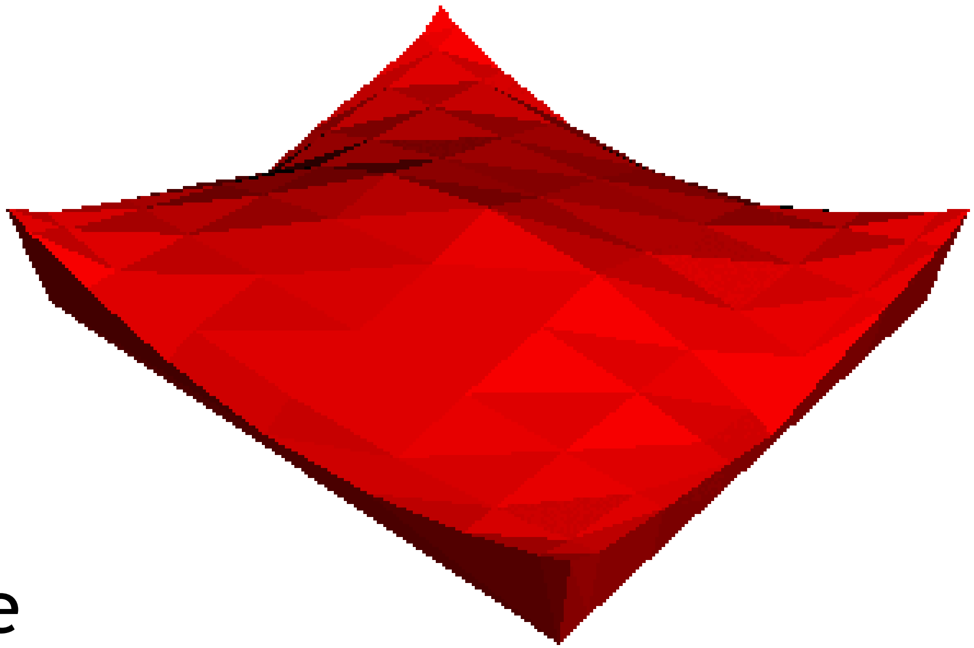
Per-triangle, -vertex, -pixel shading

- May compute shading operations
  - Once per triangle
  - Once per vertex
  - Once per pixel



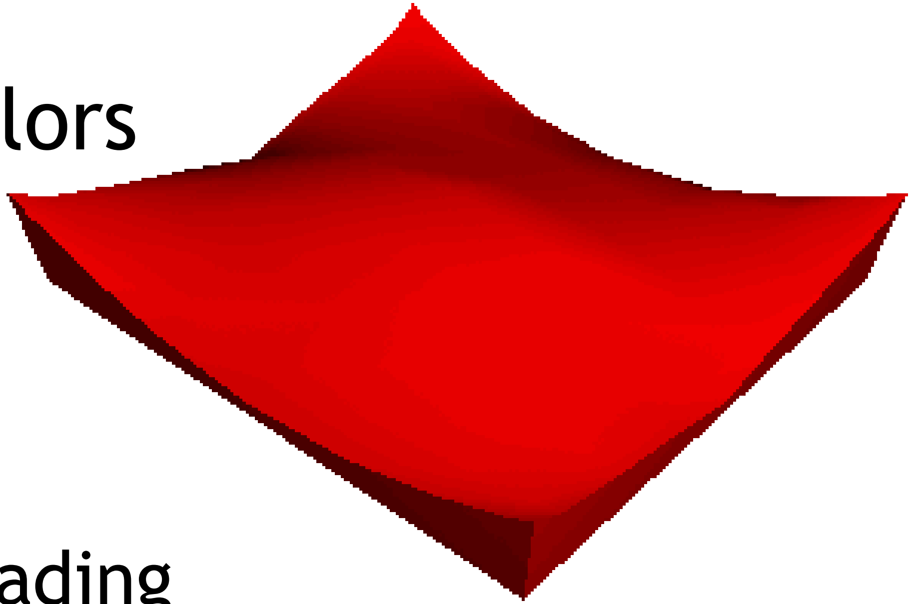
Per-triangle shading

- Known as flat shading
- Evaluate shading once per triangle using per-triangle normal
- Advantages
  - Fast
- Disadvantages
  - Faceted appearance



Per-vertex shading

- Known as Gouraud shading (Henri Gouraud 1971)
- Per-vertex normals
- Interpolate vertex colors across triangles
- Advantages
  - Fast
  - Smoother than flat shading
- Disadvantages
  - Problems with small highlights

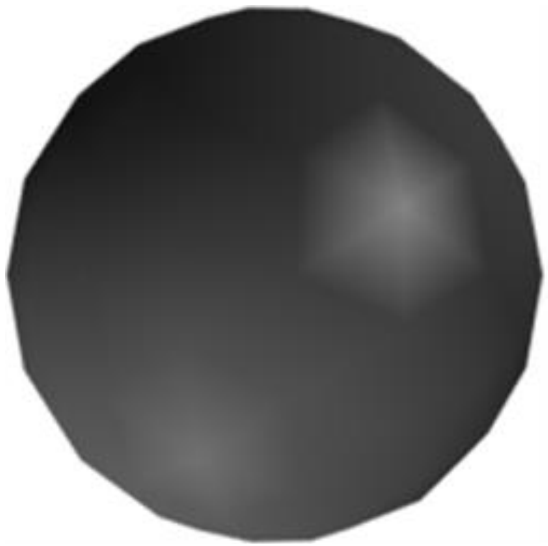


Per-pixel shading

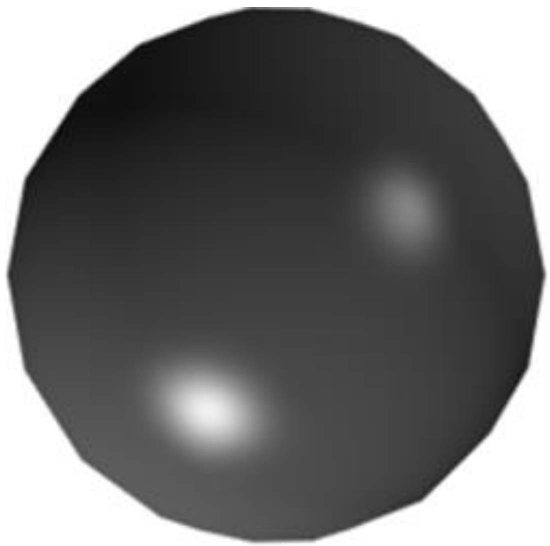
- Also known as Phong interpolation (not to be confused with Phong illumination model)
  - Rasterizer interpolates normals across triangles
  - Illumination model evaluated at each pixel
  - Implemented using programmable shaders (next week)
- Advantages
  - Higher quality than Gouraud shading
- Disadvantages
  - Much slower, but no problem for today's GPUs

Gouraud vs. per-pixel shading

- Gouraud has problems with highlights
- Could use more triangles…



Gouraud



Per-pixel,
same triangles

What about shadows?

- Standard shading assumes light sources are visible everywhere
  - Does not determine if light is blocked
  - No shadows!
- Shadows require additional work
- Later in the course

What about textures?

- How to combine „colors“ stored in textures and lighting computations?
- Interpret textures as shading coefficients
- Usually, texture used as ambient and diffuse reflectance coefficient k_d, k_a
- Textures provide spatially varying BRDFs
  - Each point on surface has different BRDF parameters, different appearance

Summary

- Local illumination (single bounce) is computed using BRDF
- BRDF captures appearance of a material
  - Amount of reflected light for each pair of light/viewing directions
- Simplified model for BRDF consists of diffuse + Phong/Blinn + ambient
  - Lambert‘s law for diffuse surfaces
  - Microfacet model for specular part
  - Ambient to approximate multiple bounces
- Light source models
  - Directional
  - Point, spot, inverse square fall-off
- Different shading strategies
  - Per triangle, Gouraud, per pixel

Next time

- Programmable shaders