Light Reflection and Illumination

CS 4620 Lecture 21

Visual cues to 3D geometry

- size (perspective)
- occlusion
- shading

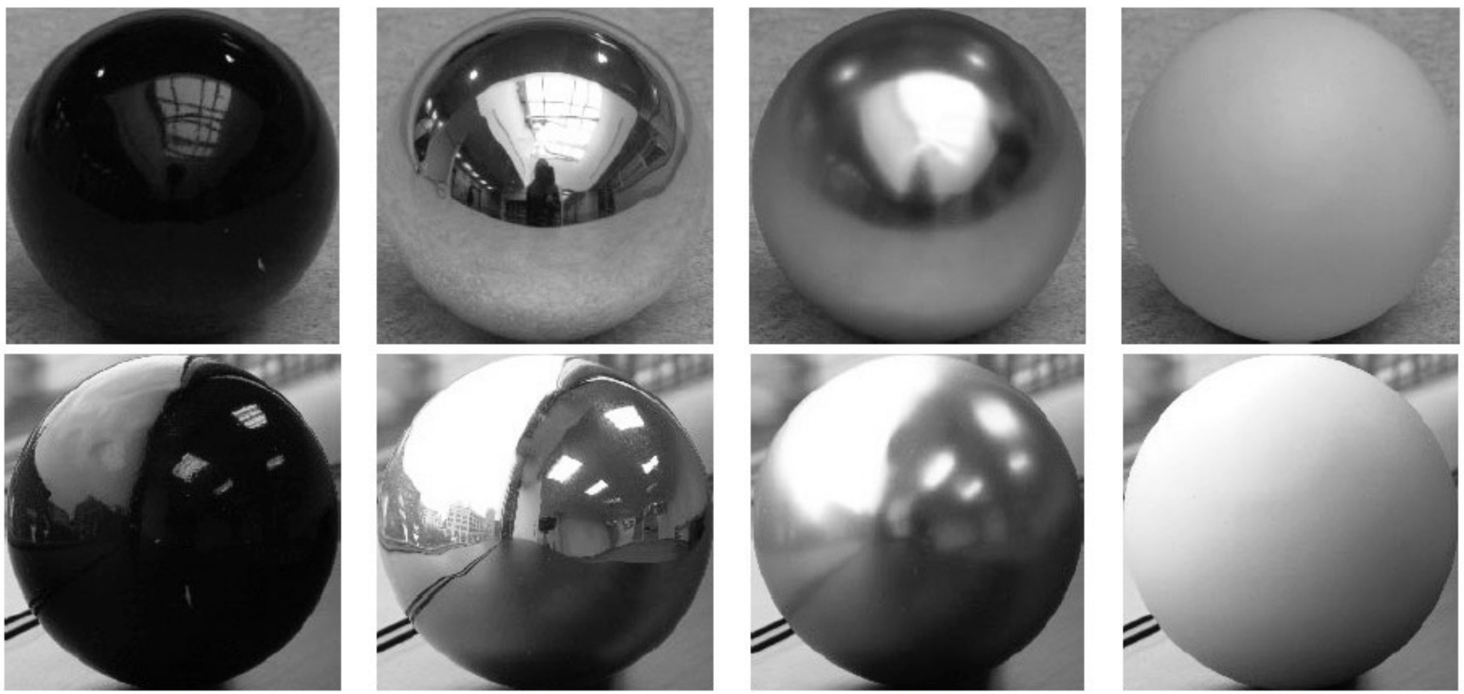
Shading

- Variation in observed color across an object
  - strongly affected by lighting
  - present even for homogeneous material
- caused by how a material reflects light
  - depends on
    - geometry
    - lighting
    - material
  - therefore gives cues to all 3

[Philip Greenspun]

Recognizing materials

- Human visual system is quite good at understanding shading



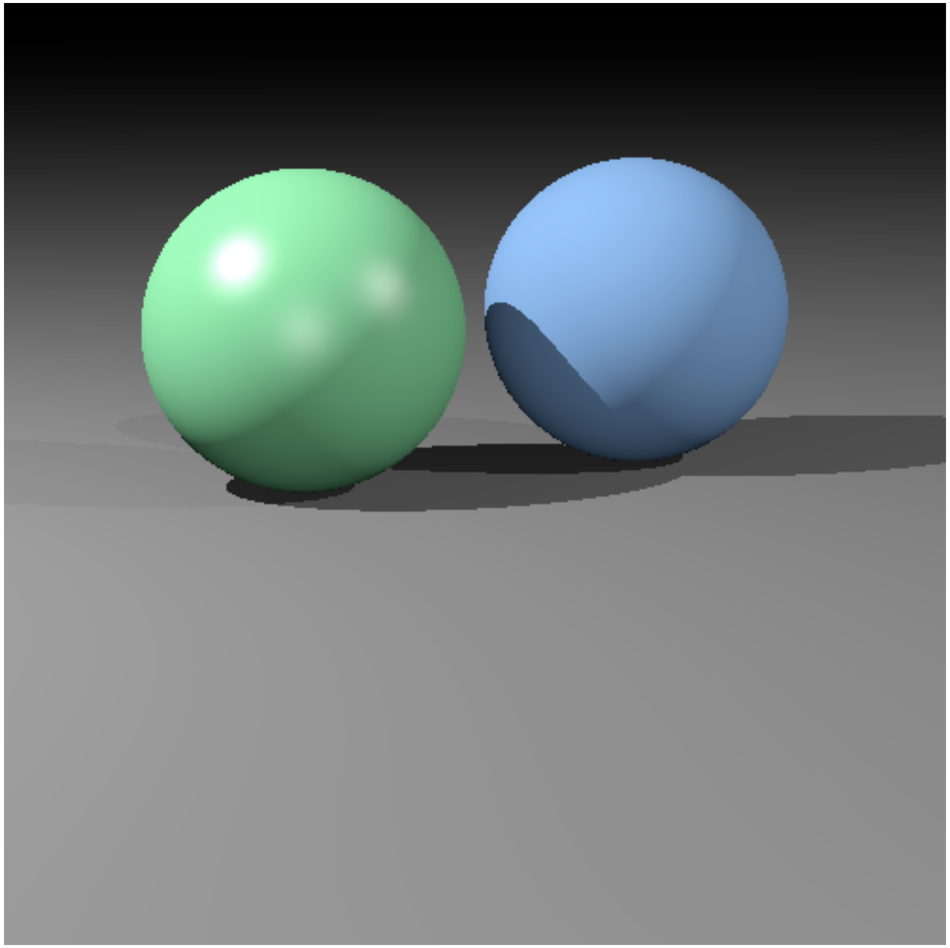
A B C D

[Dror, Adelson, & Willsky]

Shading for Computer Graphics

- Need to compute an image
  - of particular geometry
  - under particular illumination
  - from a particular viewpoint
- Basic question: how much light reflects from an object toward the viewer?

Diffuse + Phong shading



Mirror reflection

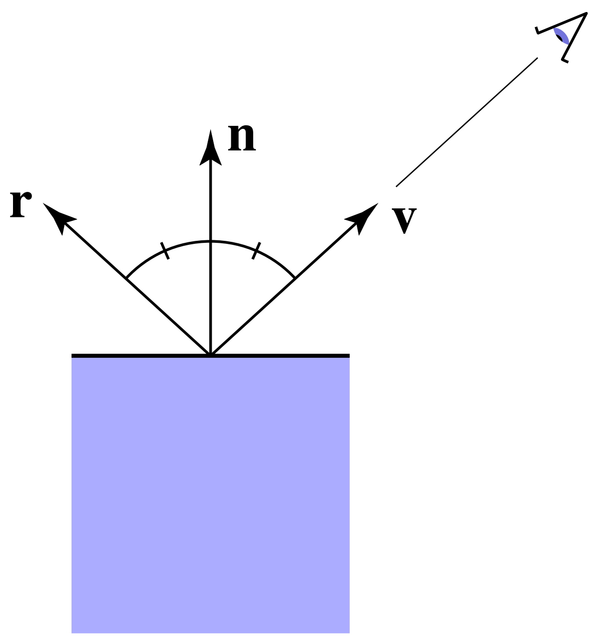
- Consider perfectly shiny surface
  - there isn't a highlight
  - instead there's a reflection of other objects
- Can render this using recursive ray tracing
  - to find out mirror reflection color, ask what color is seen from surface point in reflection direction
  - already computing reflection direction for Phong…
- "Glazed" material has mirror reflection and diffuse

$$L = L_a + L_d + L_m$$

  - where L_m is evaluated by tracing a new ray

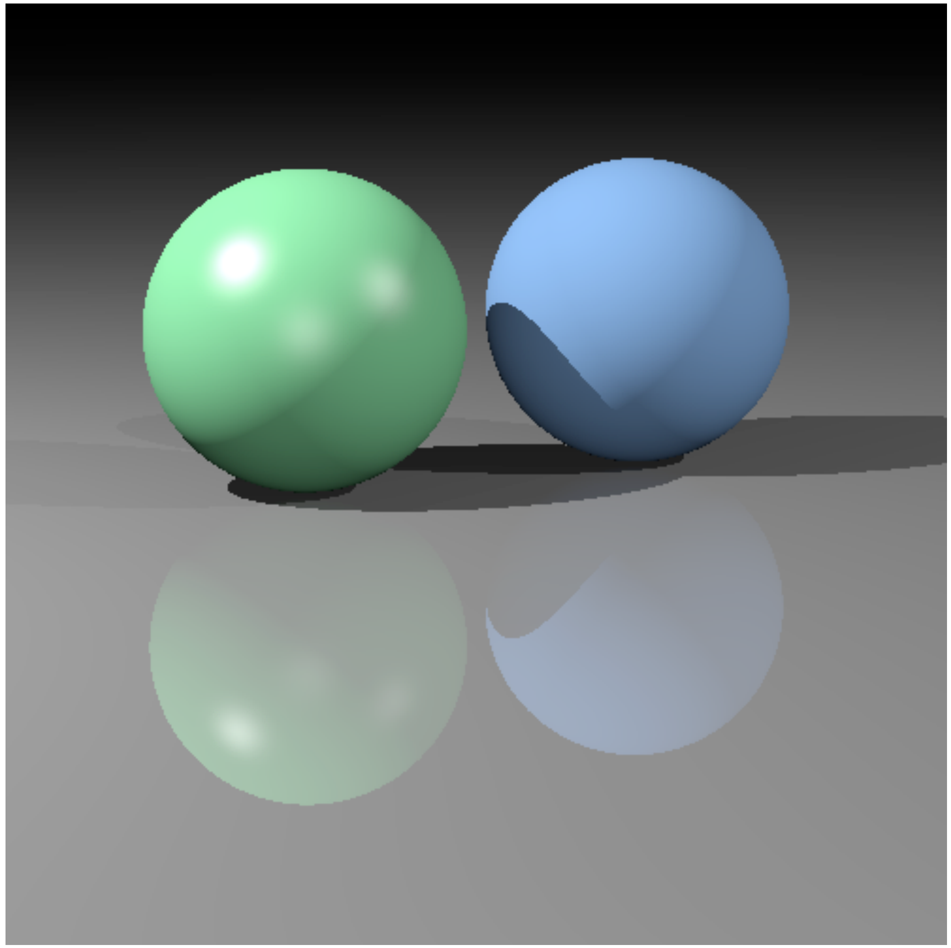
Mirror reflection

- Intensity depends on view direction
  - reflects incident light from mirror direction



$$\begin{aligned}\mathbf{r} &= \mathbf{v} + 2((\mathbf{n}\cdot\mathbf{v})\mathbf{n} - \mathbf{v}) \\ &= 2(\mathbf{n}\cdot\mathbf{v})\mathbf{n} - \mathbf{v}\end{aligned}$$

Diffuse + mirror reflection (glazed)



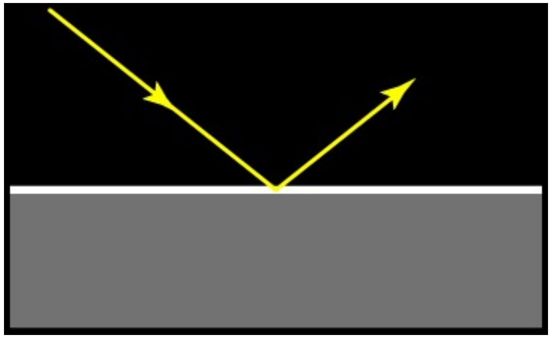
(glazed material on floor)

Fancier shading

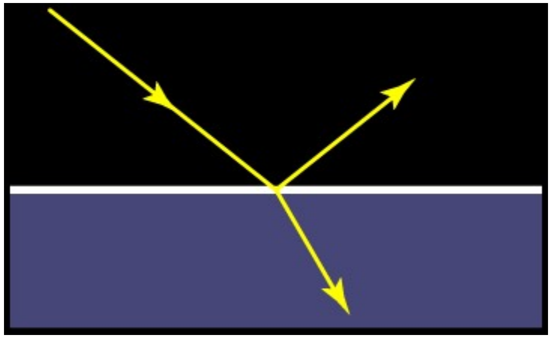
- Diffuse + Phong has long been the heuristic baseline for surface shading
- Newer/better methods are more based on physics
  - when writing a shader, think like a bug standing on the surface
  - bug sees an *incident distribution* of light that is arriving at the surface
  - physics question: what is the *outgoing distribution* of light?

Simple materials

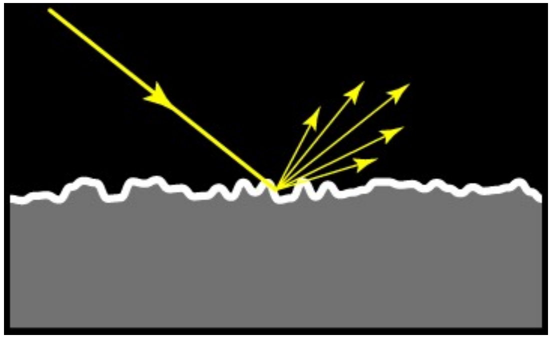
metal

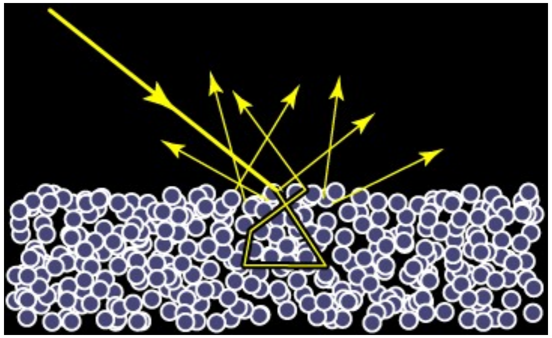


dielectric

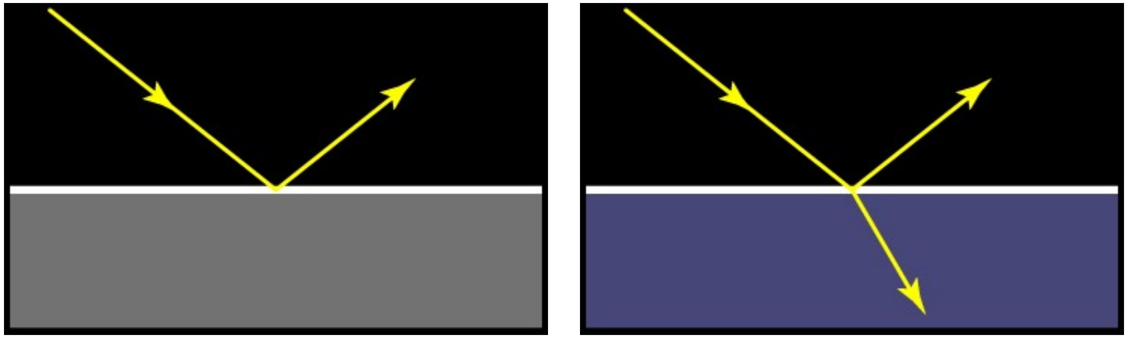


Adding microgeometry

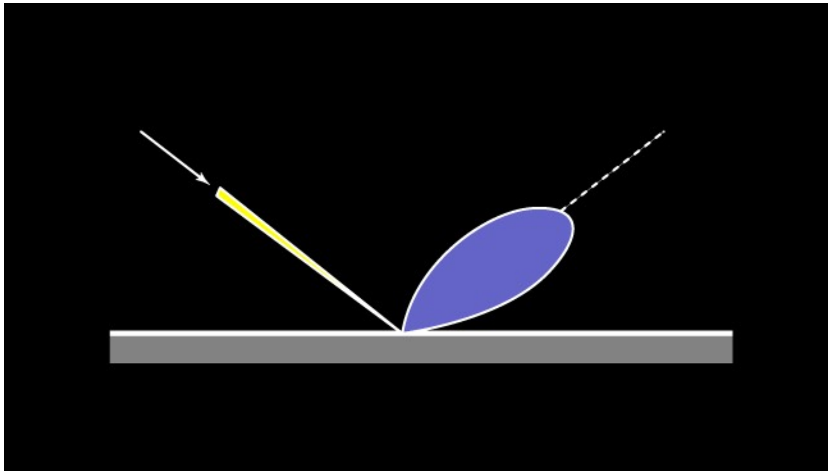




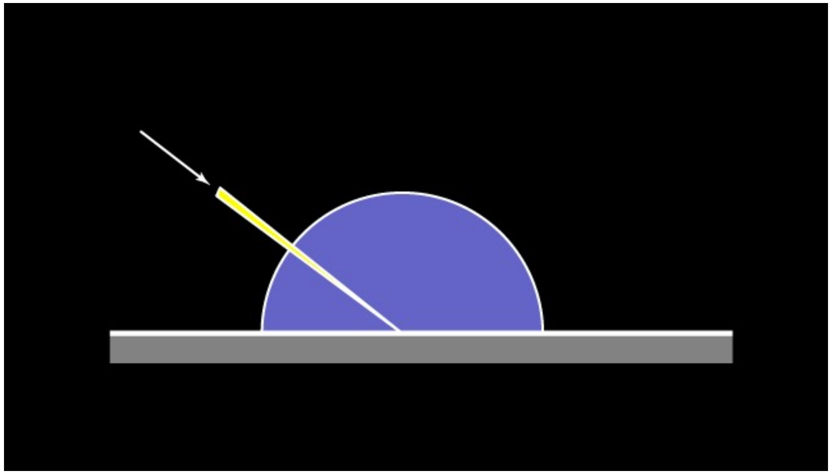
Classic reflection behavior



ideal specular (mirror)



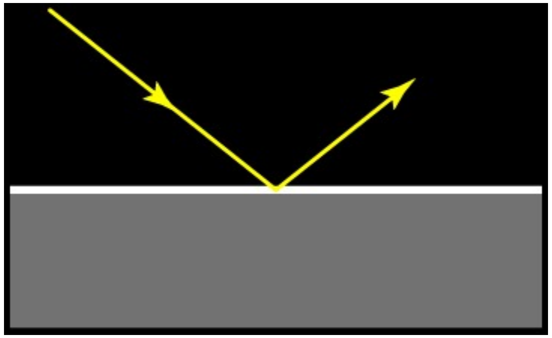
glossy specular



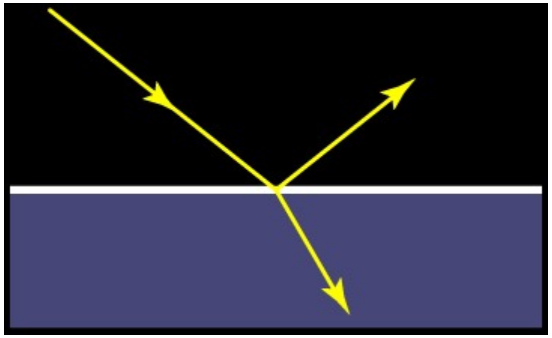
Lambertian

Specular reflection

- Smooth surfaces of pure materials have ideal specular reflection (said this before)
  - Metals (conductors) and dielectrics (insulators) behave differently
- Reflectance (fraction of light reflected) depends on angle

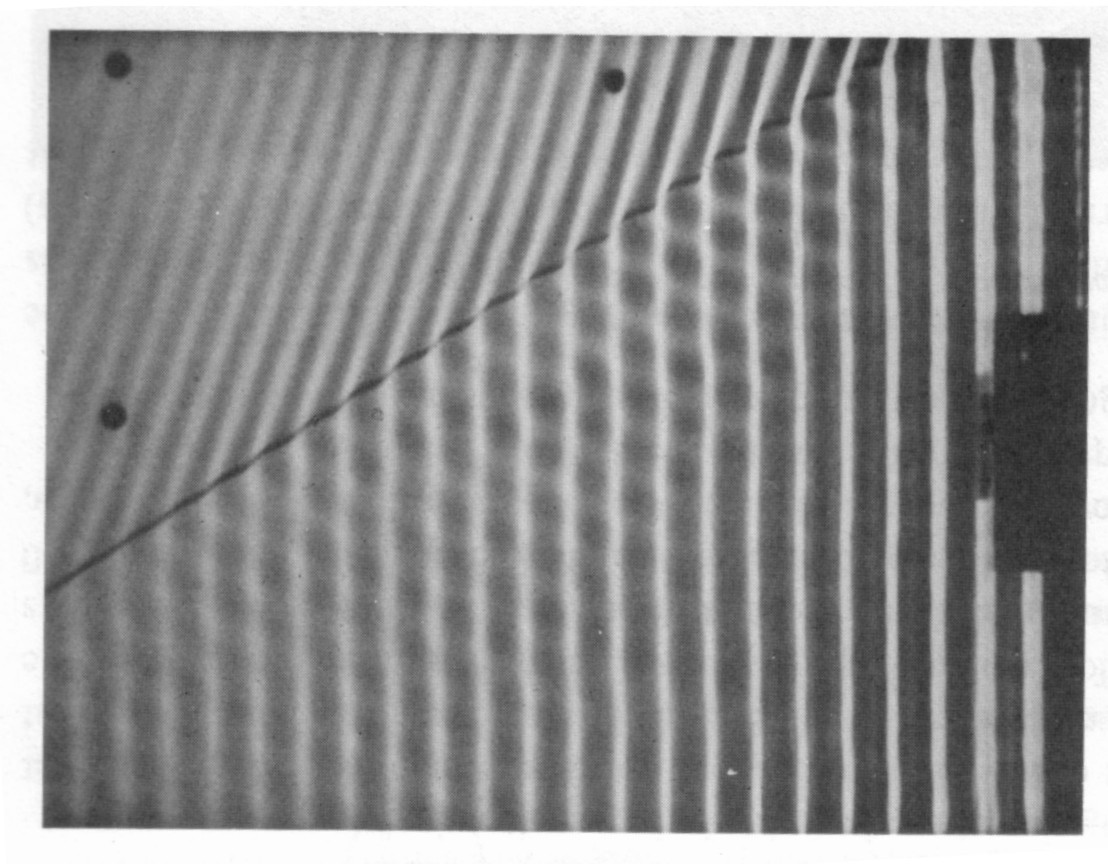


metal



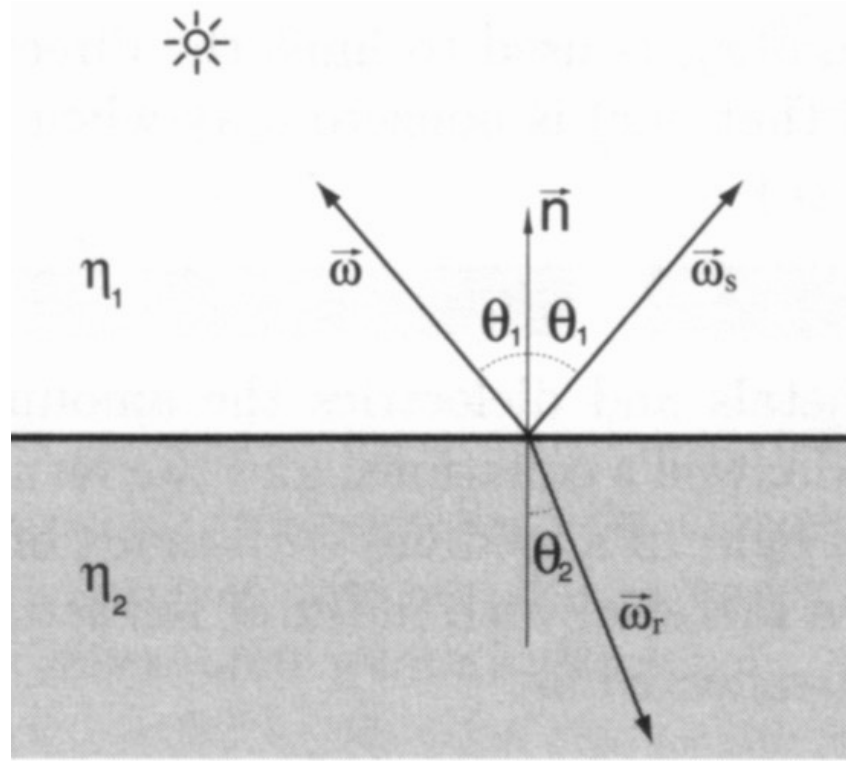
dielectric

Refraction at boundary of media



Snell's Law

- Tells us where the refracted ray goes
- Computation
  - ratio of sines is ratio of in-plane components
  - project to surface; scale by eta ratio; recompute normal-direction component
  - total internal reflection



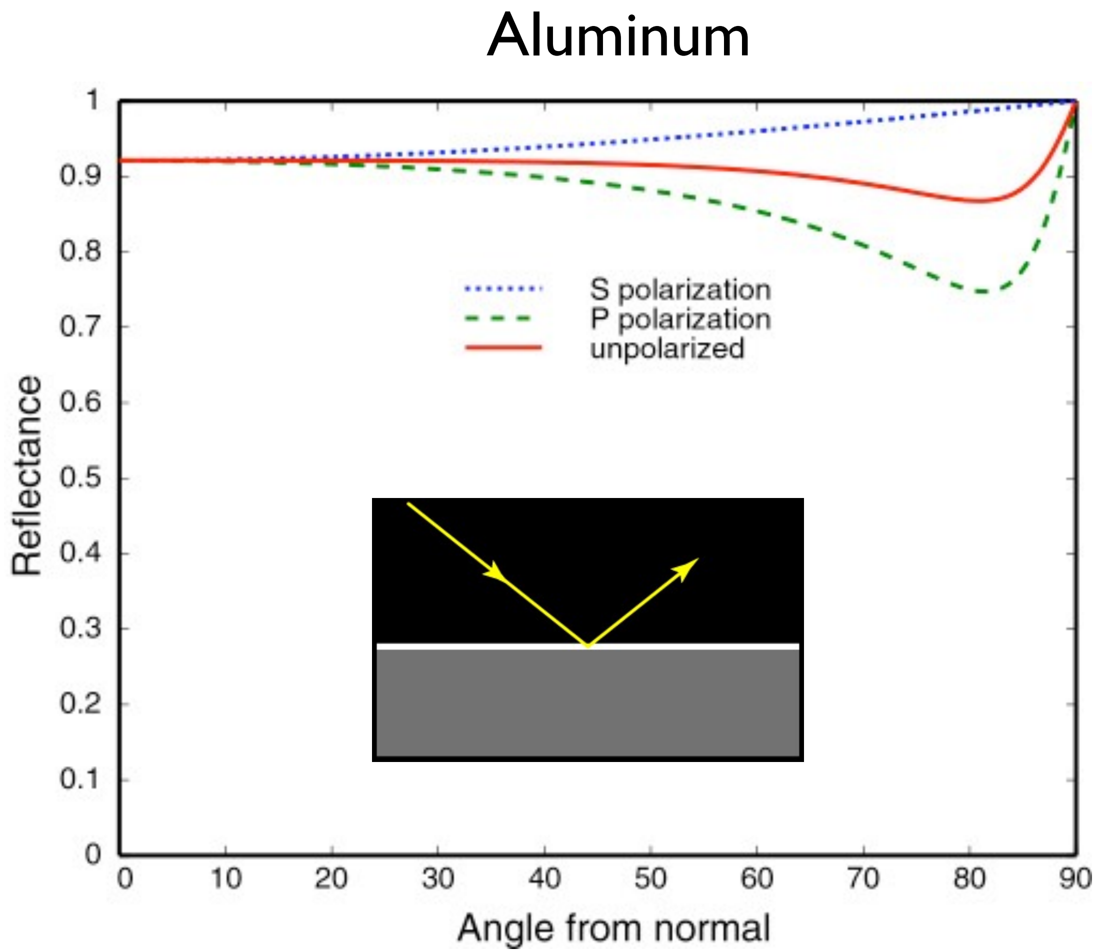
$$\eta_1 \sin \theta_1 = \eta_2 \sin \theta_2$$

Ray tracing dielectrics

- Like a simple mirror surface, use recursive ray tracing
- But we need two rays
  - One reflects off the surface (same as mirror ray)
  - The other crosses the surface (computed using Snell's law)
    - Doesn't always exist (total internal reflection)
- Splitting into two rays, recursively, creates a ray tree
  - Very many rays are traced per viewing ray
  - Ways to prune the tree
    - Limit on ray depth
    - Limit on ray attenuation

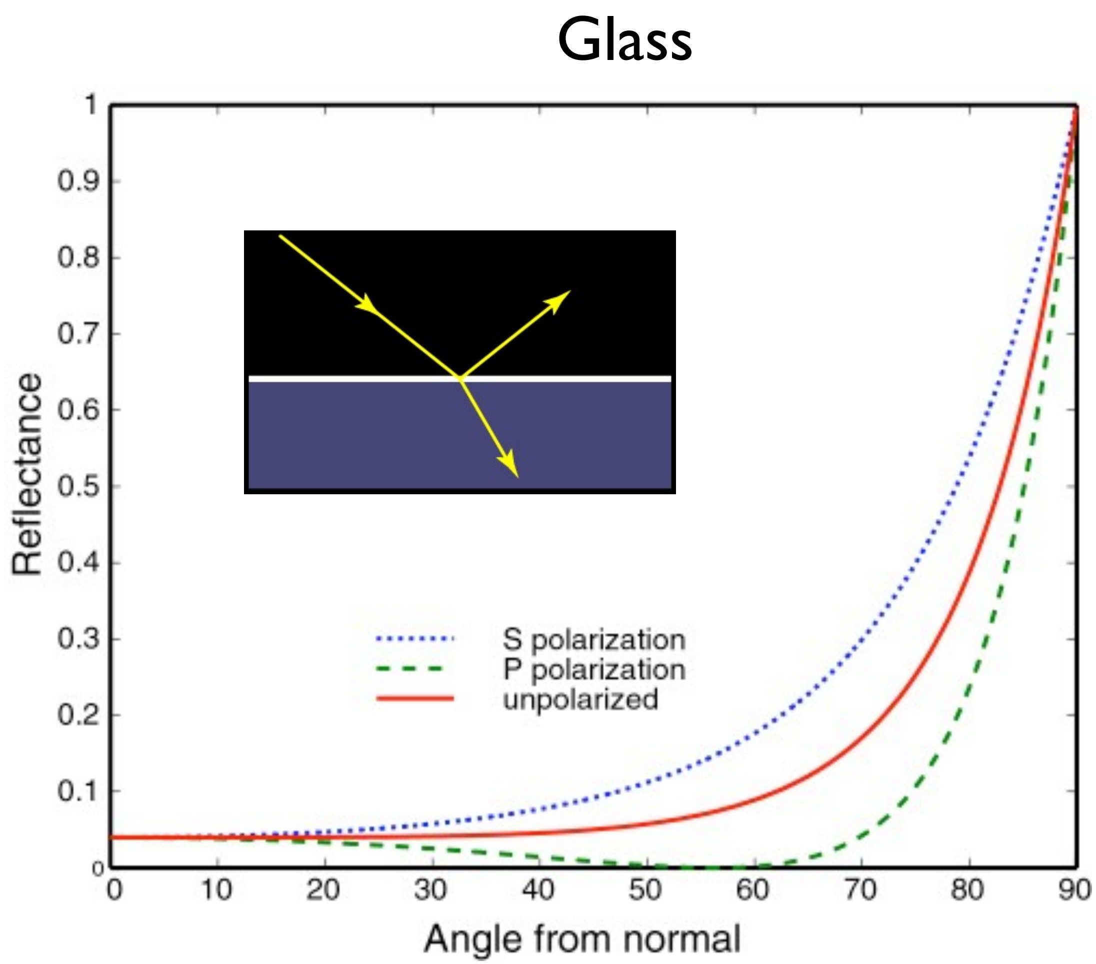
Specular reflection from metal

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



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



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

Schlick's approximation

- For graphics, a quick hack to get close with less computation:

$$\tilde{R} = R_0 + (1 - R_0)(1 - \cos\theta)^5$$

- R_0 is easy to compute:

$$F_p = \frac{\eta_2 - \eta_1}{\eta_2 + \eta_1}$$

$$F_s = \frac{\eta_1 - \eta_2}{\eta_1 + \eta_2}$$

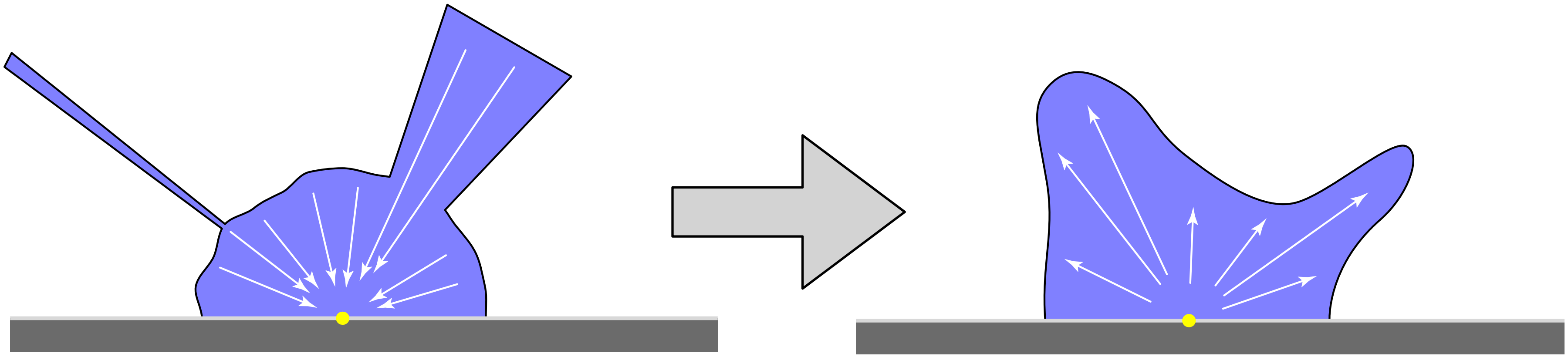
$$R_0 = \left(\frac{\eta_2 - \eta_1}{\eta_2 + \eta_1}\right)^2$$

[Josh Wills | 2003 UCSD Rendering Competition]

Fresnel reflection

[Mike Hill & Gaain Kwan | Stanford cs348 competition 2001]

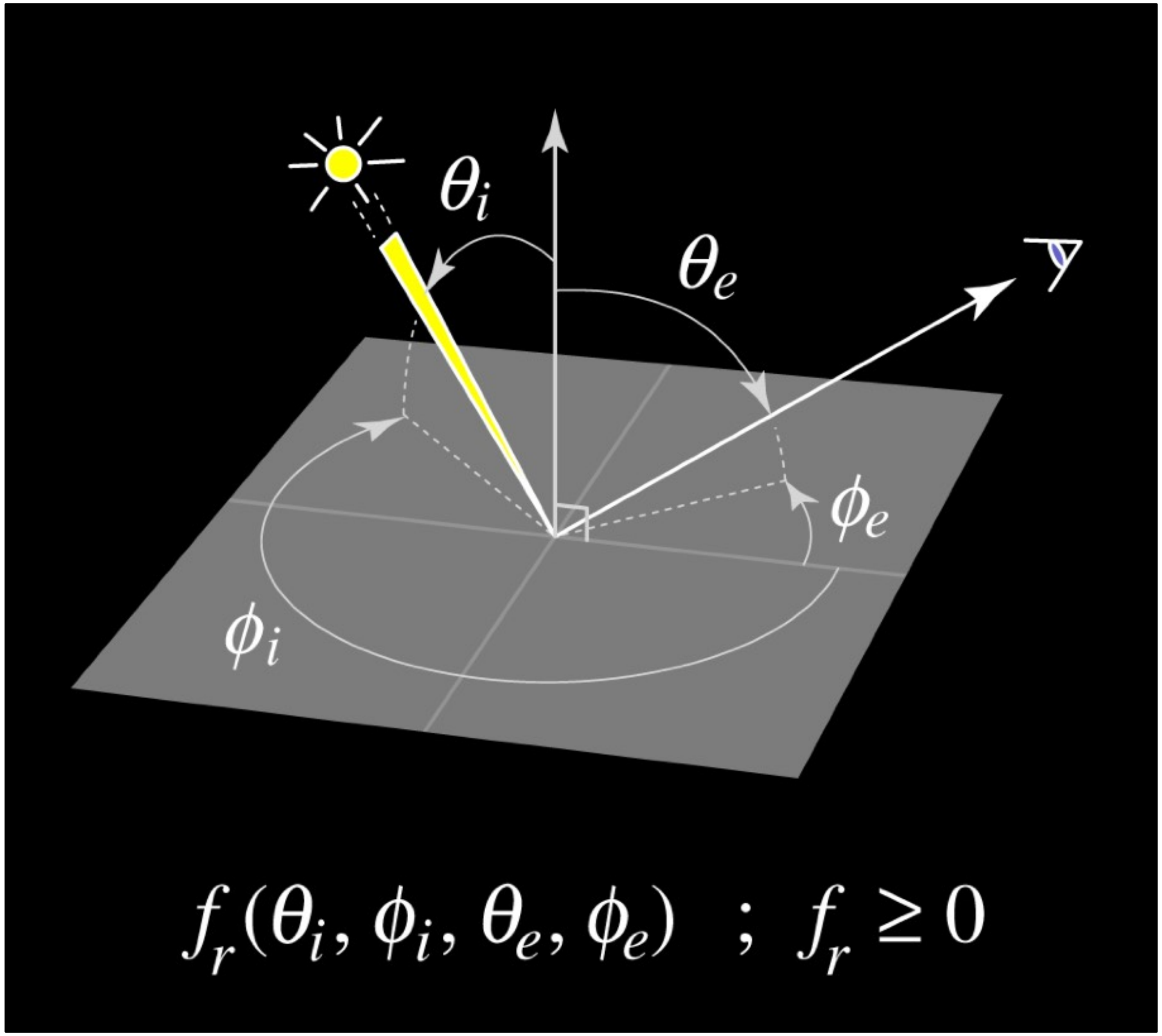
Light reflection: full picture



incident distribution
(function of direction)

reflected distribution
(function of direction)

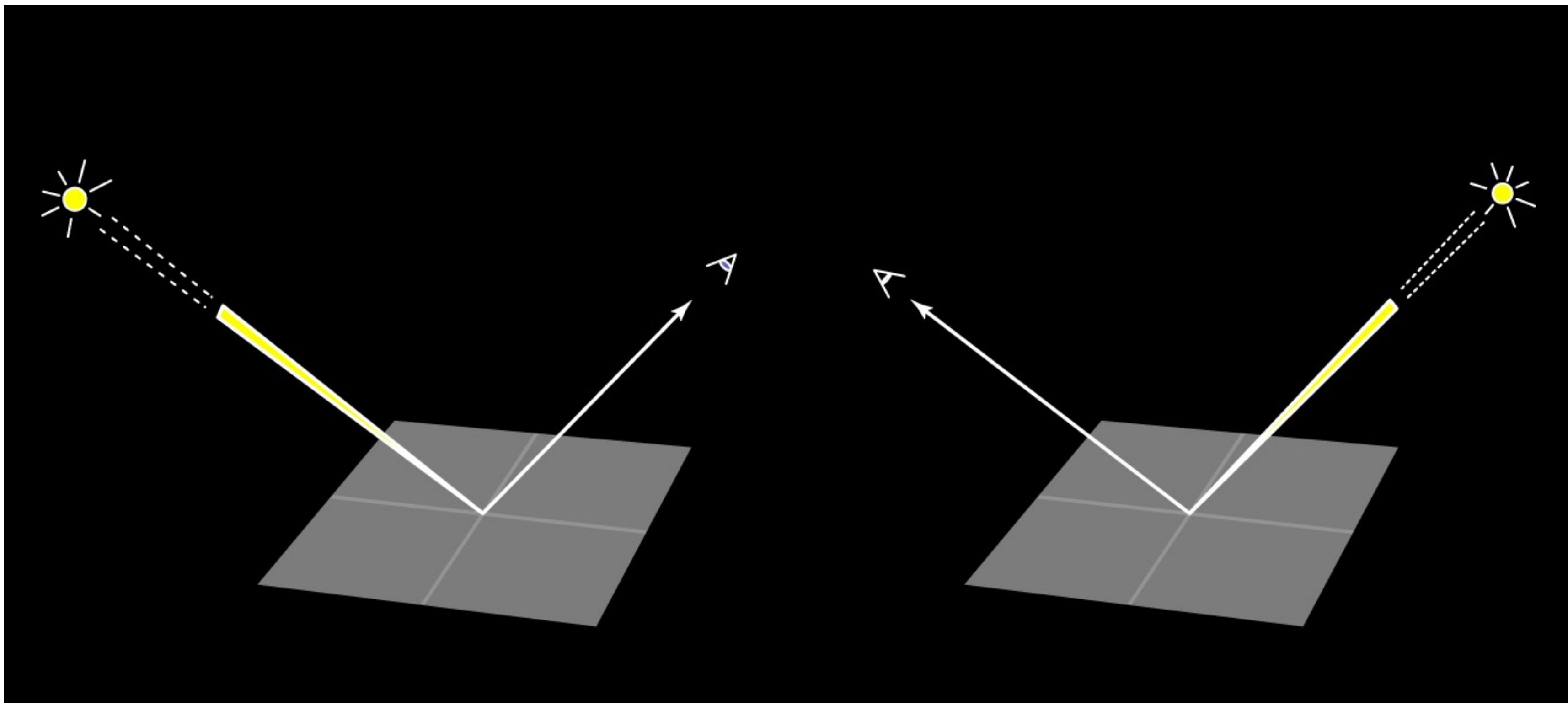
BRDF



Bidirectional Reflectance Distribution Function

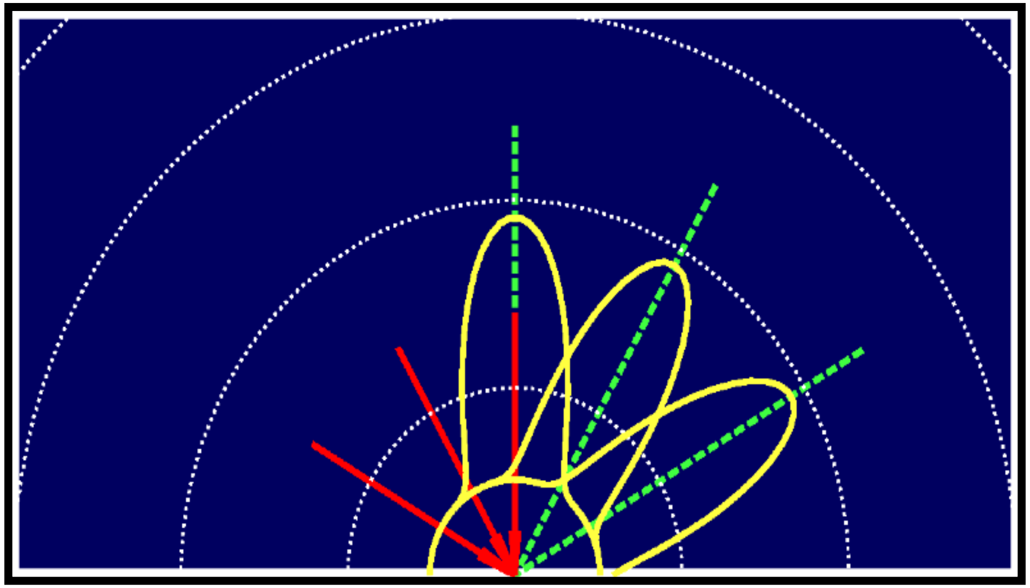
Reciprocity

- Interchanging arguments
- Physical requirement



Phong behavior

- For all incident angles, the maximum is 1.0
- Peak is always in the specular direction



[Cornell PCG]

Phong: Reality Check

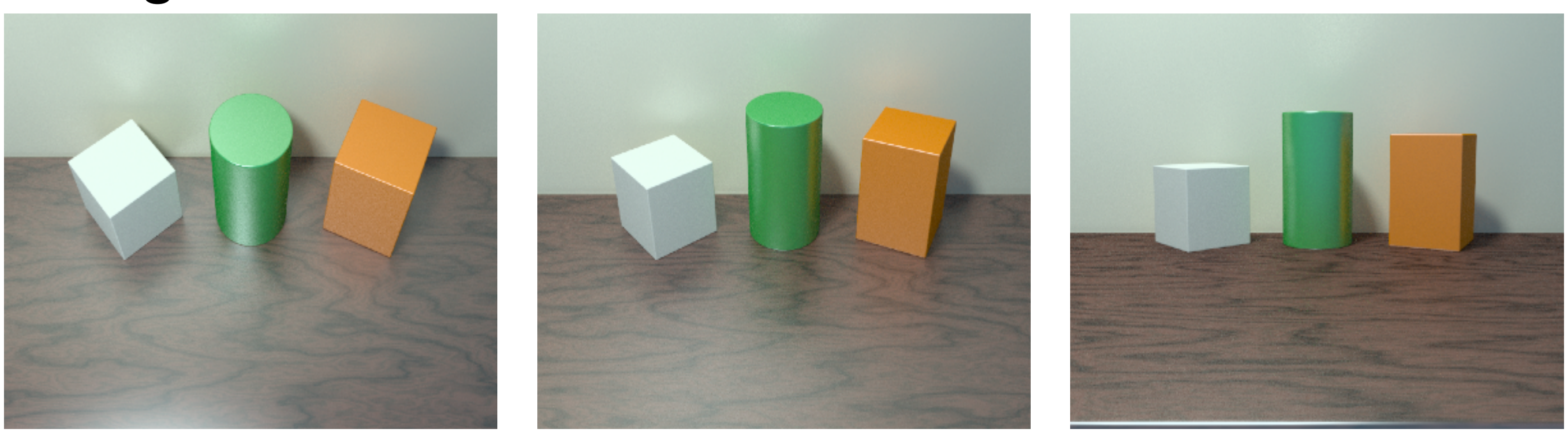
Real photographs

[Lafortune et al.]

Phong: Reality Check

Real photographs

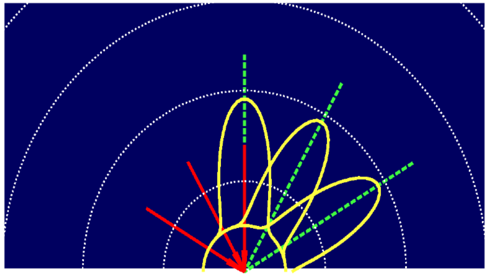
Phong model



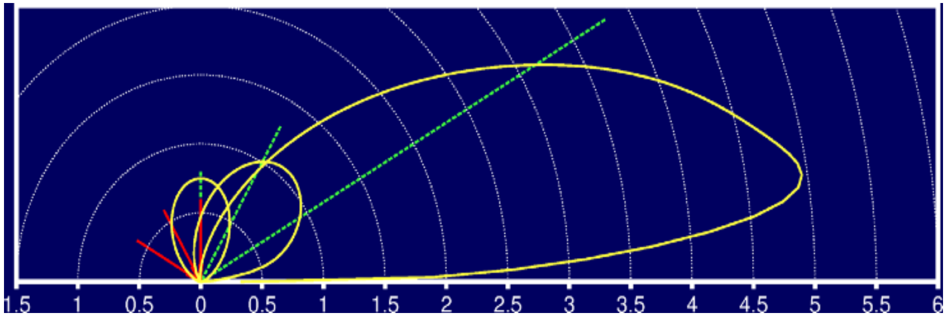
[Lafortune et al.]

Phong: Reality Check

Phong model



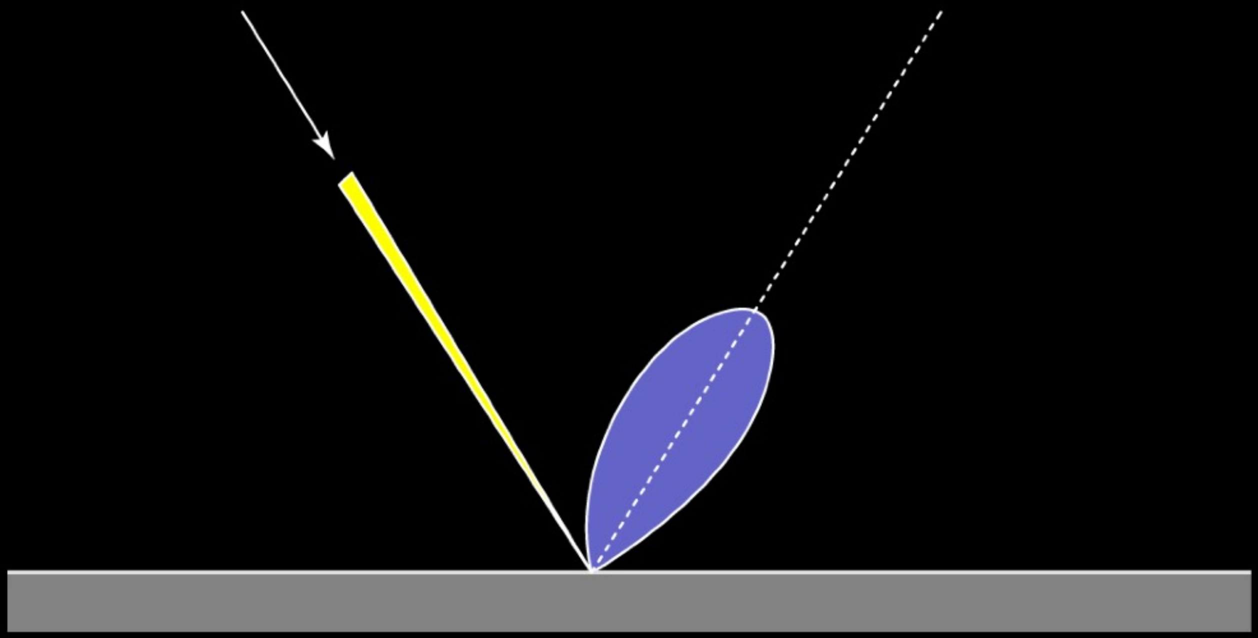
Physics-based model



[Cornell PCG]

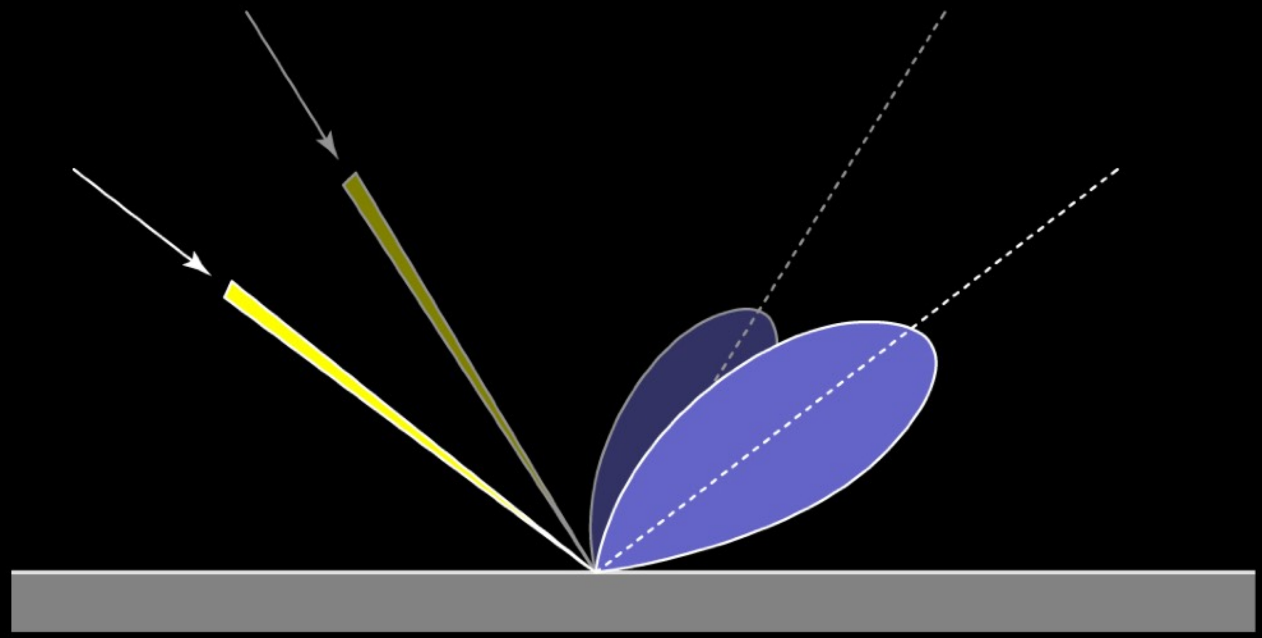
- Doesn't represent physical reality
  - energy not conserved
  - not reciprocal
  - maximum always in specular direction

Increasing Specularity



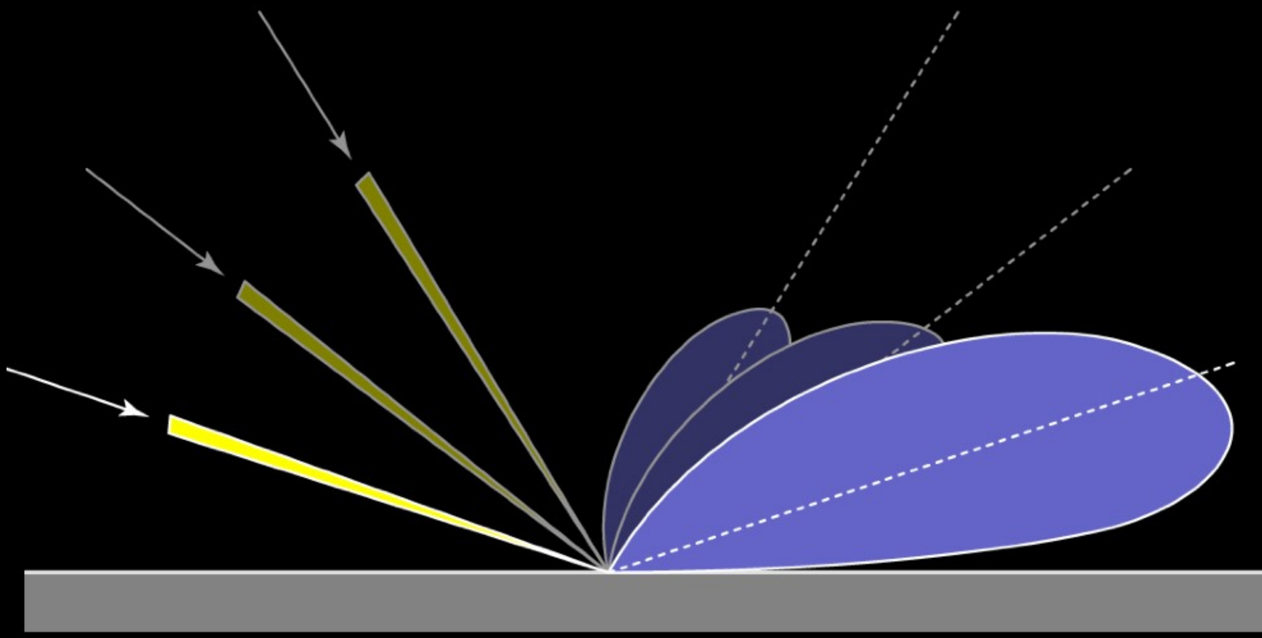
“Specular”

Increasing Specularity



"Specular"

Increasing Specularity



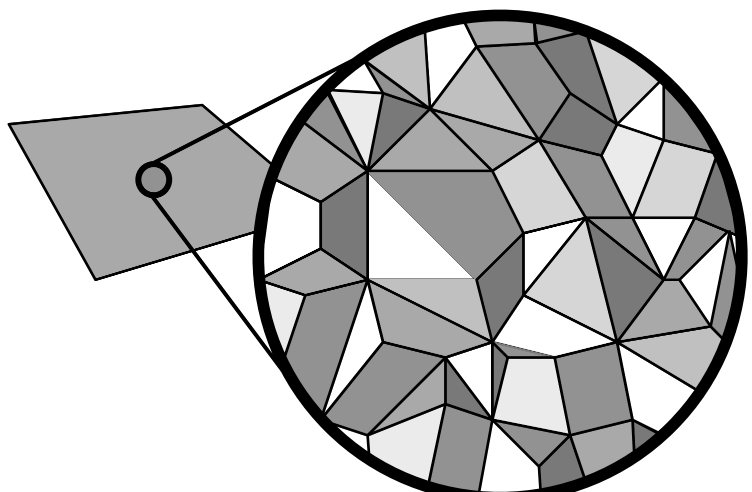
“Specular”

Broad modeling approaches

- Empirical expressions
  - a long and glorious history…
  - you know these: Phong, Ward, Kajiya, etc.
- Microfacet models
  - a geometric optics approach for surface reflection
  - based on statistical averaging over microgeometry
- Other geometric-optics surface models
  - including Oren-Nayar and other diffuse models
  - also several grooved-surface models
- Subsurface scattering models
  - Hanrahan-Kreuger; diffusion models

Microfacet BRDF Model

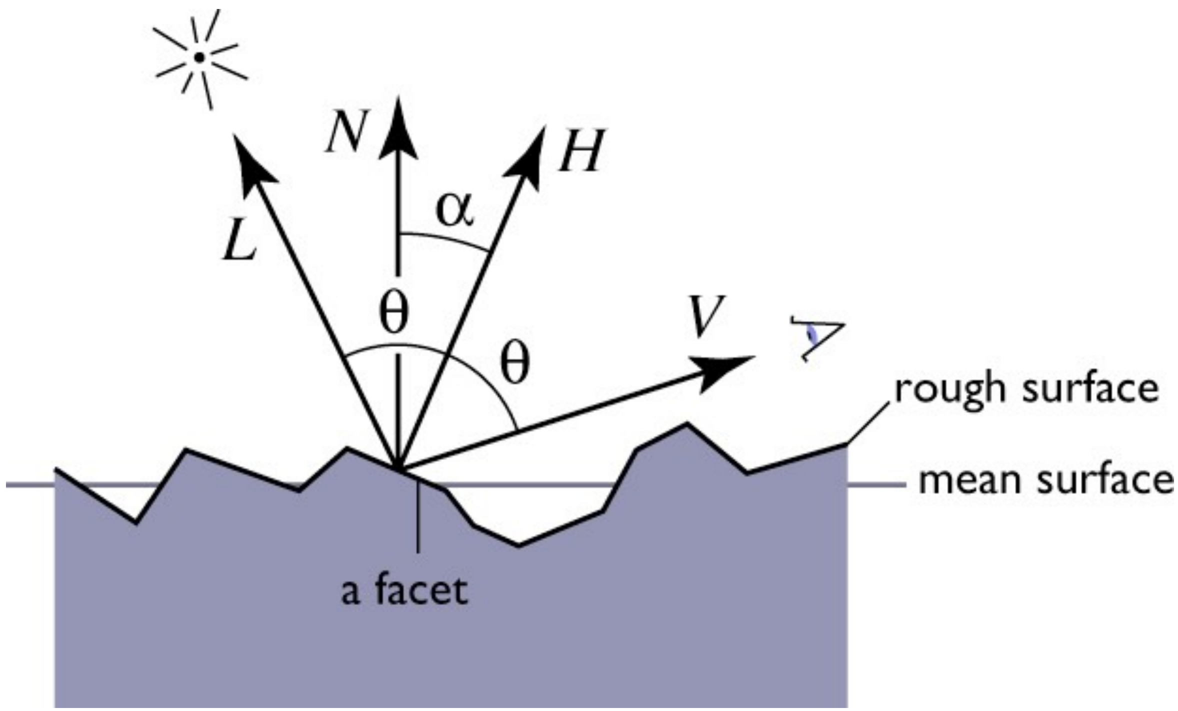
- The *microfacet* idea
  - surface modeled as random collection of planar facets
  - an incoming ray hits exactly one facet, at random
- Key input: probability distribution of facet angle



[Stephen Westin]

Facet Reflection

- *H* vector used to define facets that contribute
  - L and V determine H; only facets with that normal matter
  - reflected light is proportional to number of facets



Microfacet BRDF Model

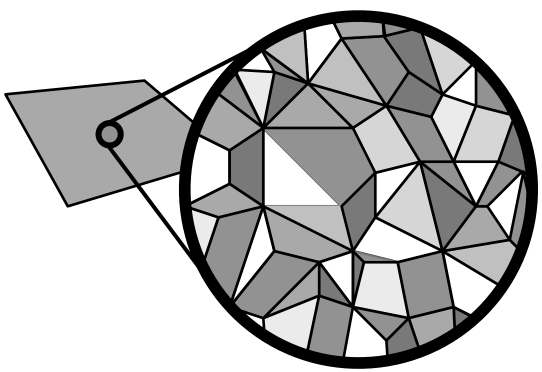
- "Specular" term (really glossy, or directional diffuse)

$$f_r(\mathbf{n}, \mathbf{l}, \mathbf{v}) = \frac{F(\mathbf{l}, \mathbf{h}) D(\mathbf{h}) G(\mathbf{l}, \mathbf{v}, \mathbf{h})}{4|\mathbf{n} \cdot \mathbf{l}||\mathbf{n} \cdot \mathbf{v}|}$$

Microfacet BRDF Model

Facet distribution

$$f_r(\mathbf{n}, \mathbf{l}, \mathbf{v}) = \frac{F(\mathbf{l}, \mathbf{h})\,D(\mathbf{h})\,G(\mathbf{l}, \mathbf{v}, \mathbf{h})}{4|\mathbf{n} \cdot \mathbf{l}||\mathbf{n} \cdot \mathbf{v}|}$$



[Stephen Westin]

Facet Distribution

- D function describes distribution of H
- Many choices, depending on surface characteristics
- A classic choice is due to Beckmann
  - derivation based on Gaussian random processes

$$D(\mathbf{h}) = \frac{e^{-\frac{\tan^2(\mathbf{h},\mathbf{n})}{m^2}}}{\pi m^2 \cos^4(\mathbf{h},\mathbf{n})}$$

Cook-Torrance BRDF Model

Fresnel Reflectance

$$f_r(\mathbf{n}, \mathbf{l}, \mathbf{v}) = \frac{F(\mathbf{l}, \mathbf{h}) D(\mathbf{h}) G(\mathbf{l}, \mathbf{v}, \mathbf{h})}{4|\mathbf{n} \cdot \mathbf{l}||\mathbf{n} \cdot \mathbf{v}|}$$

- Fresnel reflectance for smooth facet
  - more light reflected at grazing angles

Cook-Torrance BRDF Model

Masking/shadowing

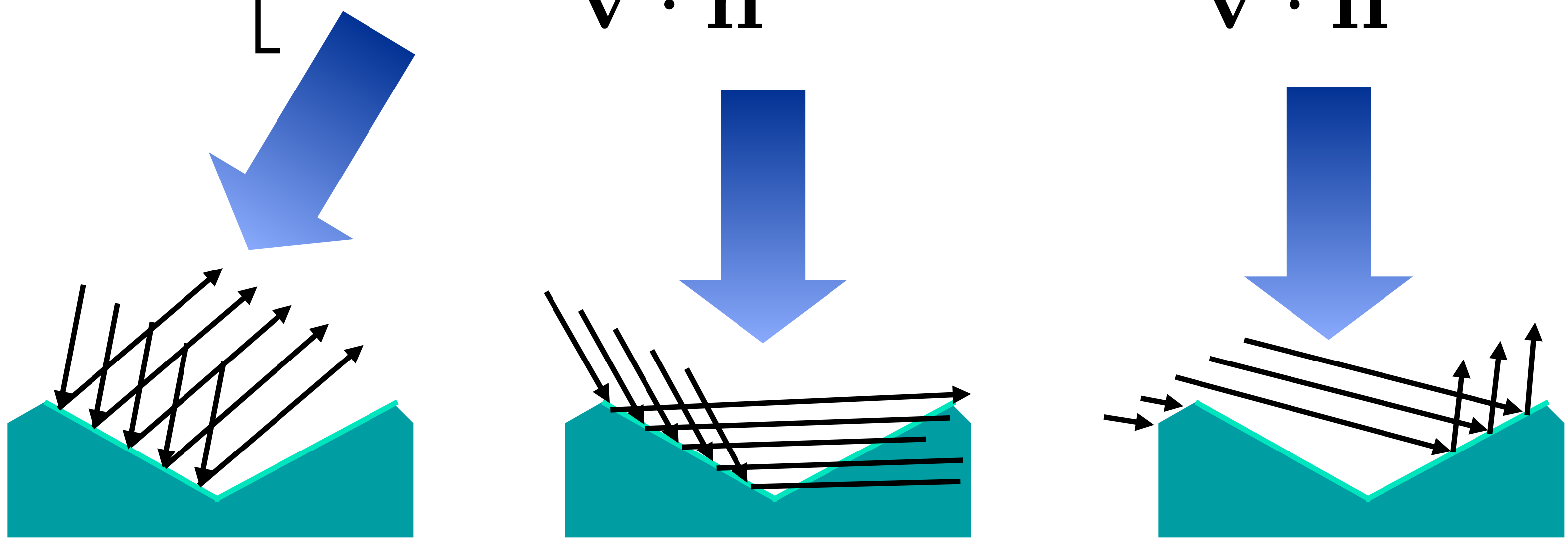
$$f_r(\mathbf{n}, \mathbf{l}, \mathbf{v}) = \frac{F(\mathbf{l}, \mathbf{h}) D(\mathbf{h}) G(\mathbf{l}, \mathbf{v}, \mathbf{h})}{4 |\mathbf{n} \cdot \mathbf{l}| |\mathbf{n} \cdot \mathbf{v}|}$$

Masking and Shadowing

- Many options; C-T chooses simple 2D analysis:

$$G(\mathbf{l}, \mathbf{v}, \mathbf{h}) =$$

$$\min\left[1, \frac{2(\mathbf{n}\cdot\mathbf{h})(\mathbf{n}\cdot\mathbf{v})}{\mathbf{v}\cdot\mathbf{h}}, \frac{2(\mathbf{n}\cdot\mathbf{h})(\mathbf{n}\cdot\mathbf{l})}{\mathbf{v}\cdot\mathbf{h}}\right]$$



[Stephen Westin]

Microfacet BRDF Model

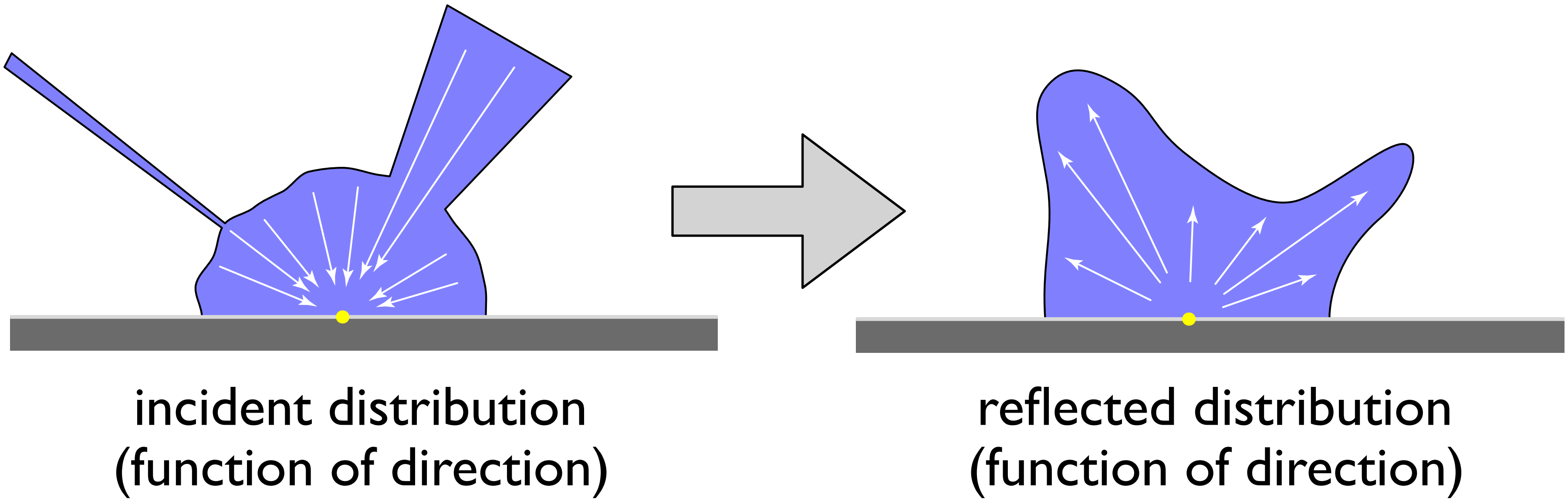
$$f_r(\mathbf{n}, \mathbf{l}, \mathbf{v}) = \frac{F(\mathbf{l}, \mathbf{h}) D(\mathbf{h}) G(\mathbf{l}, \mathbf{v}, \mathbf{h})}{4 |\mathbf{n} \cdot \mathbf{l}| |\mathbf{n} \cdot \mathbf{v}|}$$

- reasons for cosine terms in denominator
- if one is there they clearly both have to be there (by reciprocity)

Sources of illumination

- Point sources
  - energy emanating from a single point
- Directional sources
  - aka. point sources at infinity
- Area sources
  - energy emanating from an area of surface
- Environment illumination
  - energy coming from far away
- Light reflected from other objects
  - leads to *global illumination*

Light reflection: full picture



- all types of reflection reflect all types of illumination
  - diffuse, glossy, mirror reflection
  - environment, area, point illumination

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

= easy to include in “classic” ray tracer

Illumination using the easy cases

- Render mirror reflections of everything but point/ directional sources using recursive rays
- Render all other BRDF components (diffuse, glossy) using single shadow ray + BRDF evaluation

