CS130 : Computer Graphics

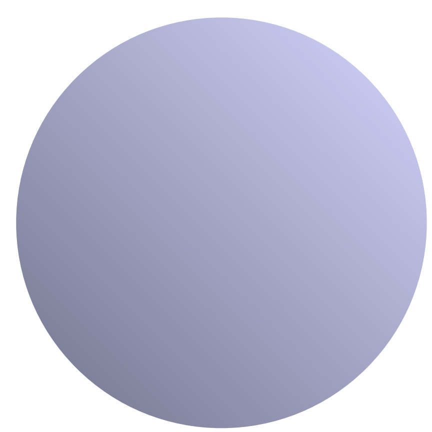
Lecture 7: Lighting and Shading

Tamar Shinar
Computer Science & Engineering
UC Riverside

Why we need shading

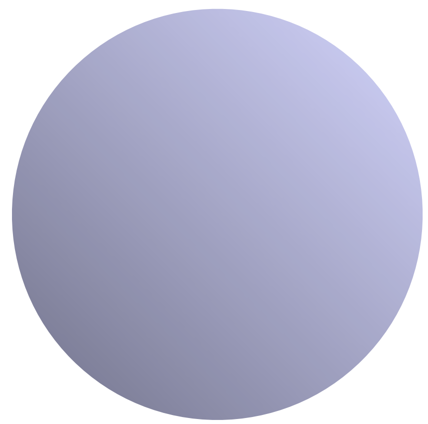
- Suppose we build a model of a sphere using many polygons and color each the same color. We get something like

- But we want



Shading

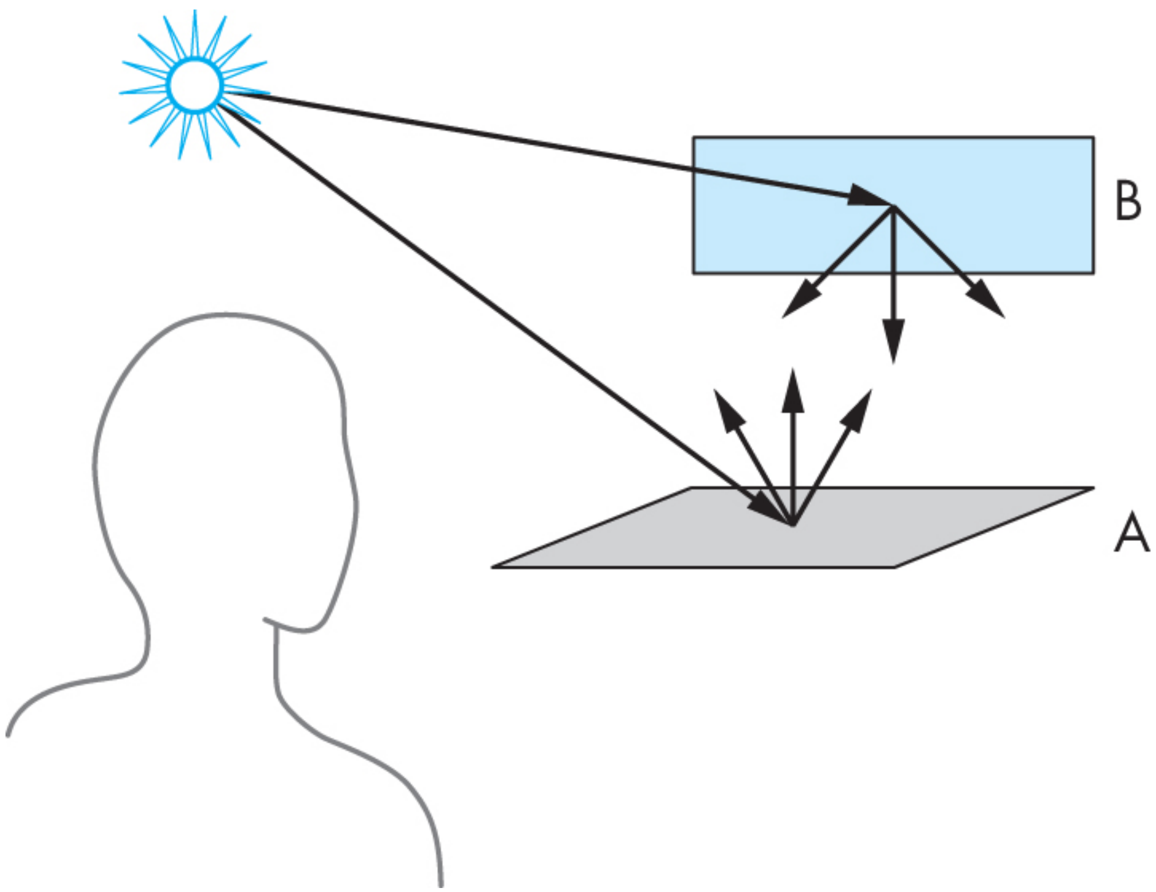
- Why does the image of a real sphere look like



- Light-material interactions cause each point to have a different color or shade
- Need to consider
  - Light sources
  - Material properties
  - Location of viewer
  - Surface orientation (normal)

General rendering

- The most general approach is based on physics - using principles such as conservation of energy
- a surface either **emits** light (e.g., light bulb) or **reflects** light for other illumination sources, or both
- light interaction with materials is **recursive**
- the **rendering equation** is an integral equation describing the limit of this recursive process

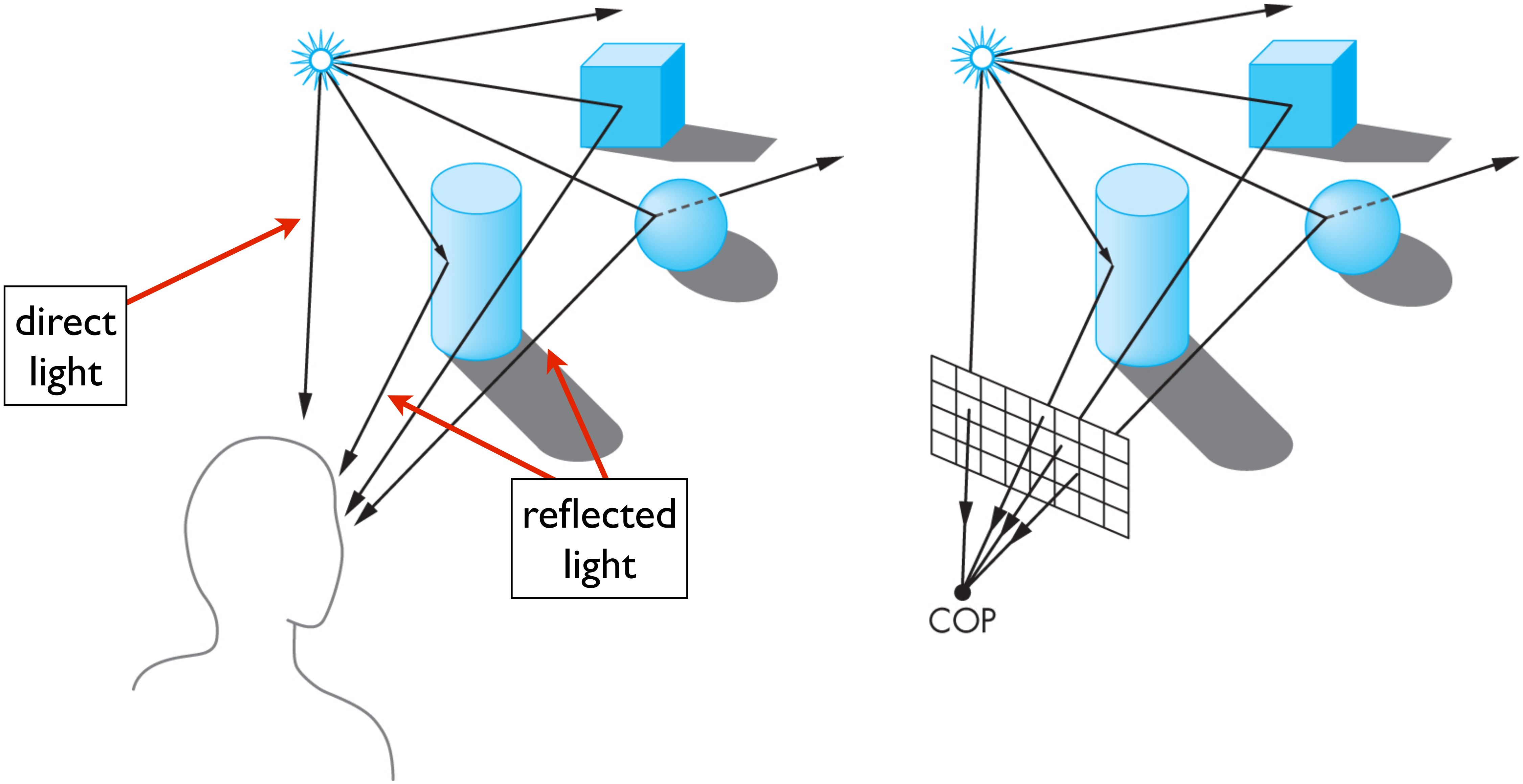


[Angel and Shreiner]

Fast local shading models

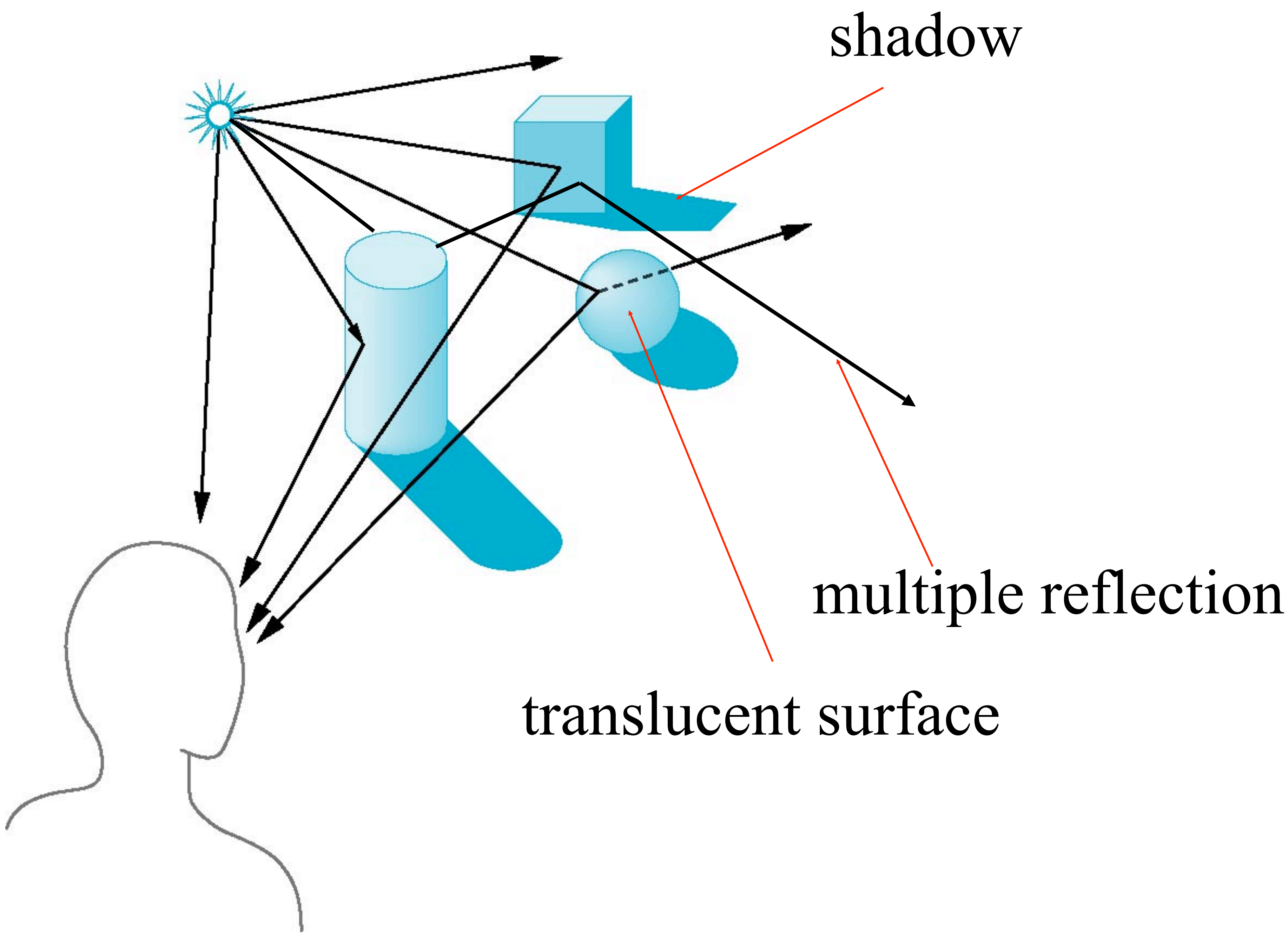
- the rendering equation can't be solved analytically
- numerical methods aren't fast enough for real-time
- for our fast graphics rendering pipeline, we'll use a **local** model where shade at a point is independent of other surfaces
- use **Phong reflection model**
  - shading based on local light-material interactions

Local shading model



[Angel and Shreiner]

Global Effects

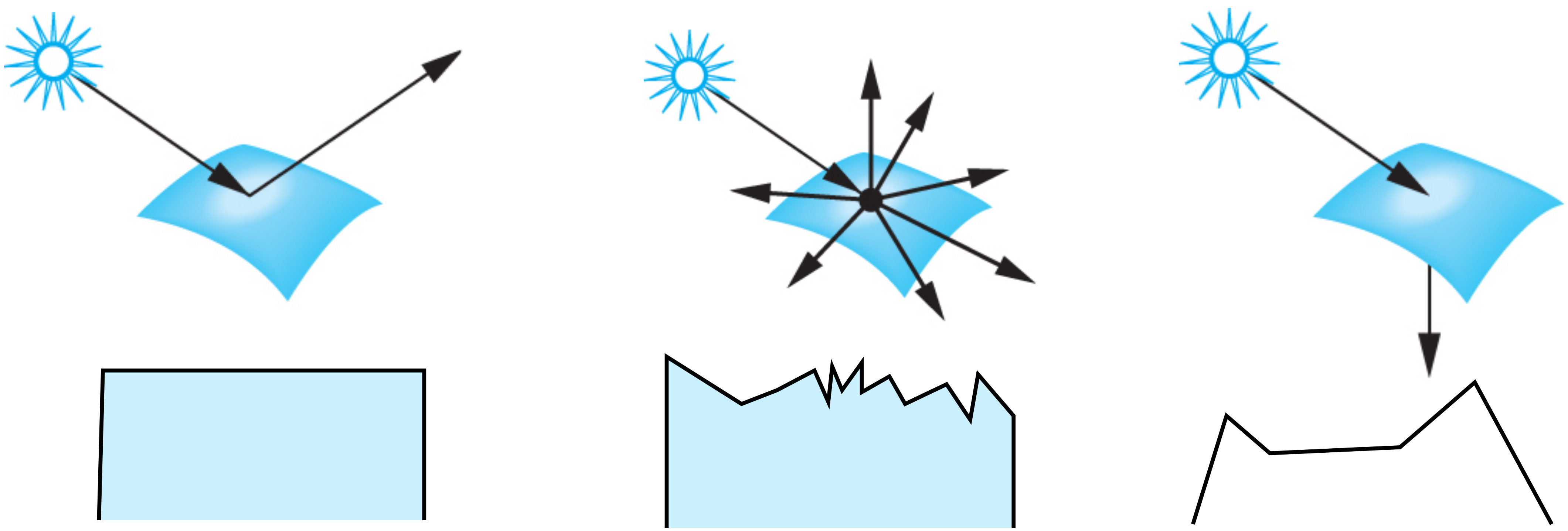


[Angel and Shreiner]

Light-material interactions

at a surface, light is absorbed, reflected, or transmitted

specular | diffuse | translucent

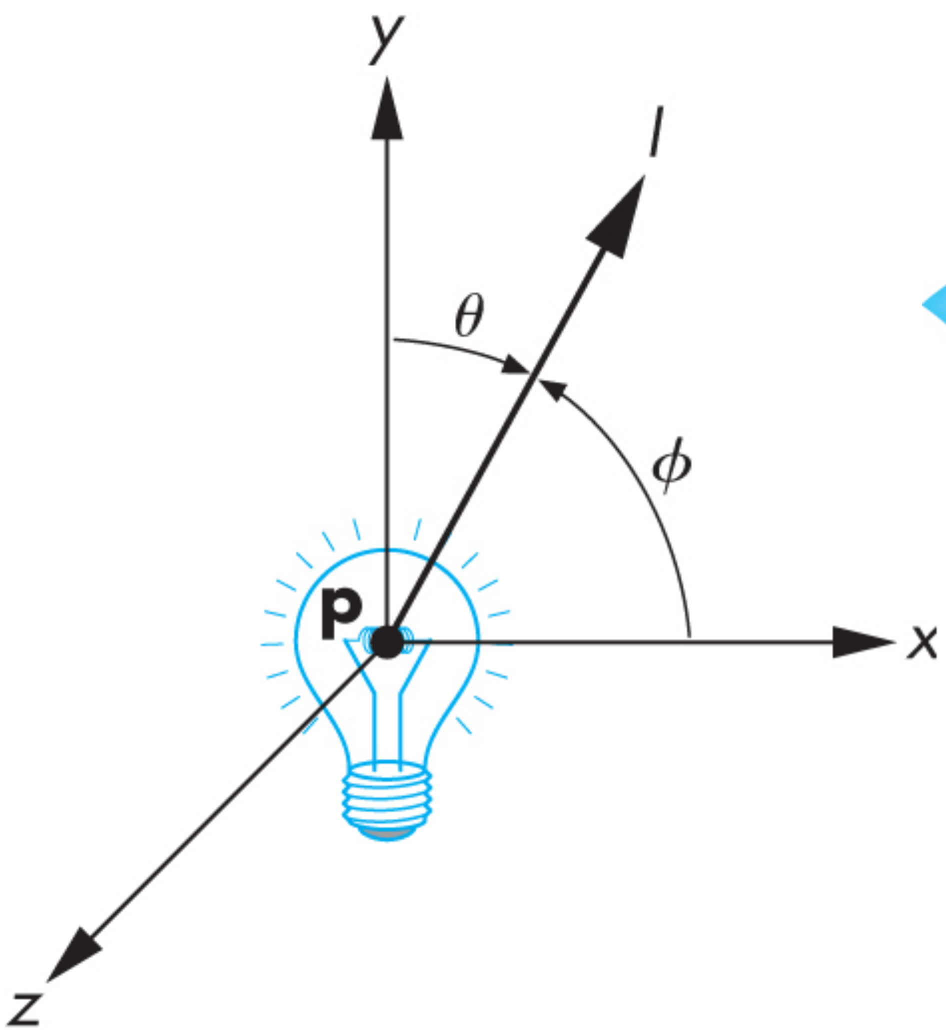


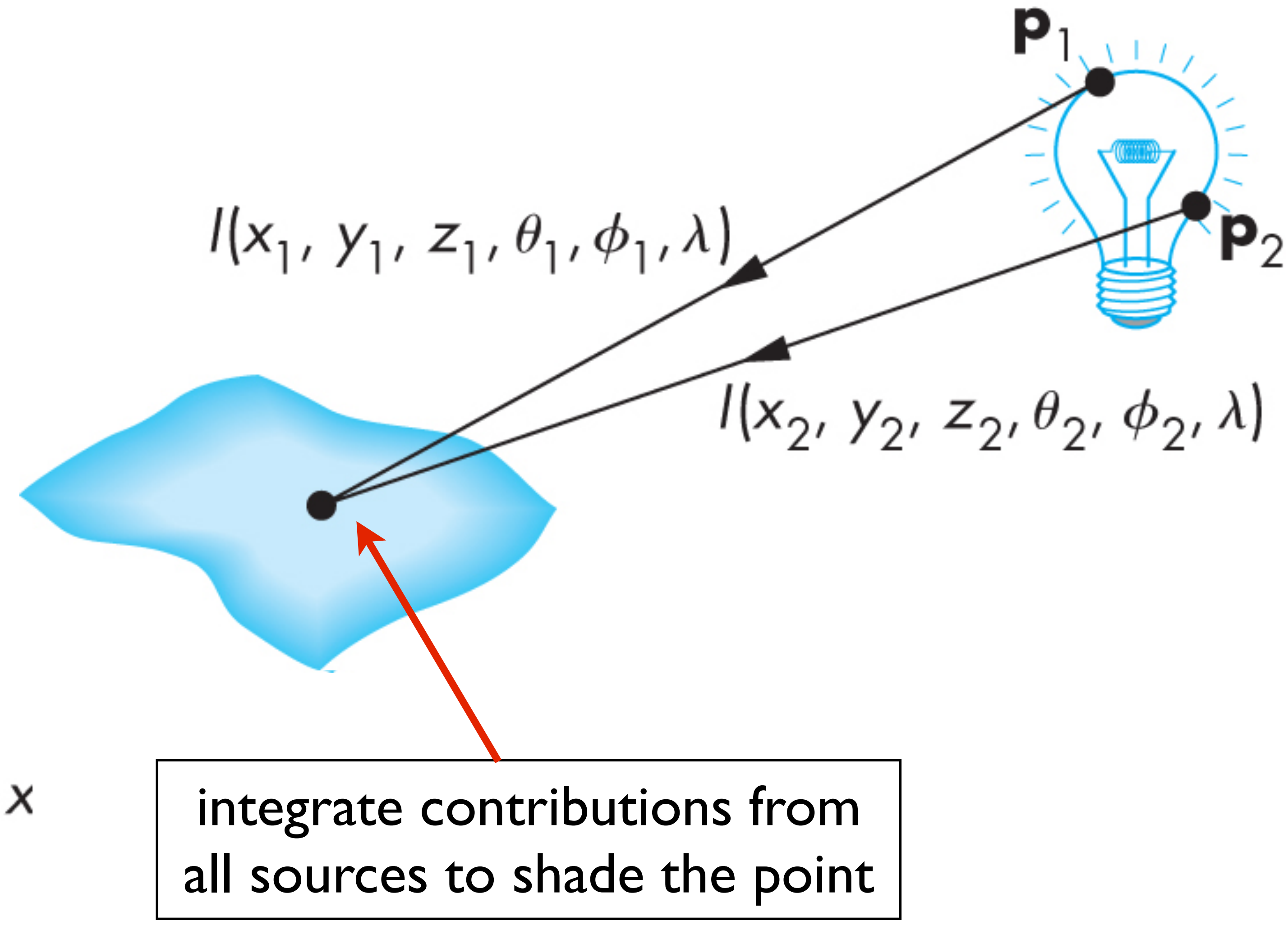
[Angel and Shreiner]

General light source

Illumination function:

$$l(x, y, z, \theta, \phi, \lambda)$$

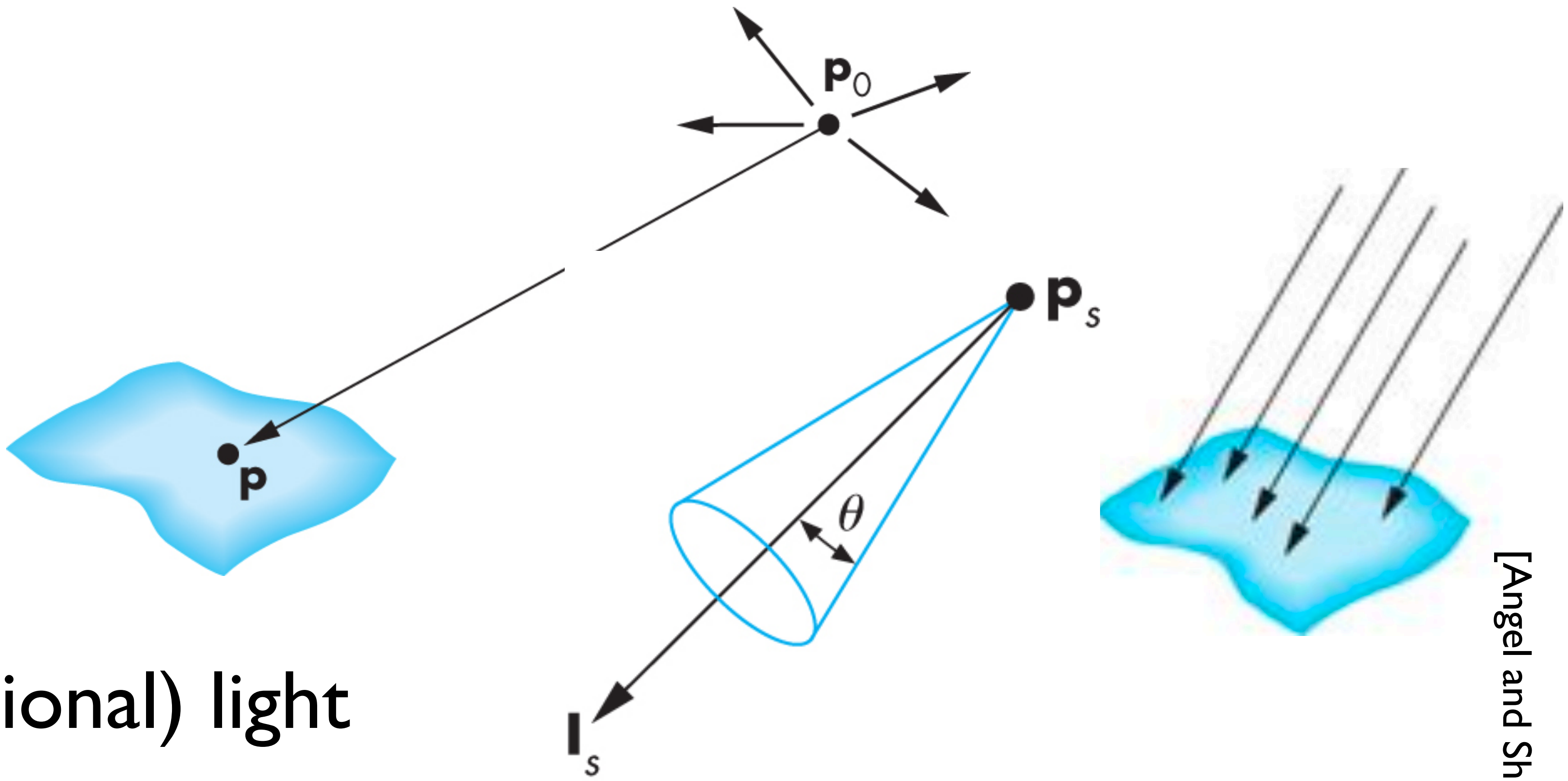




[Angel and Shreiner]

Idealized light sources

- Ambient light
- Point light
- Spotlight
- distant (directional) light



[Angel and Shreiner]

luminance: $\mathbf{L} = \begin{bmatrix} L_r \\ L_g \\ L_b \end{bmatrix}$

Ambient light source

- achieve a uniform light level
- no black shadows
- ambient light intensity at each point in the scene

$$\mathbf{L}_a = \begin{bmatrix} L_{ar} \\ L_{ag} \\ L_{ab} \end{bmatrix}$$

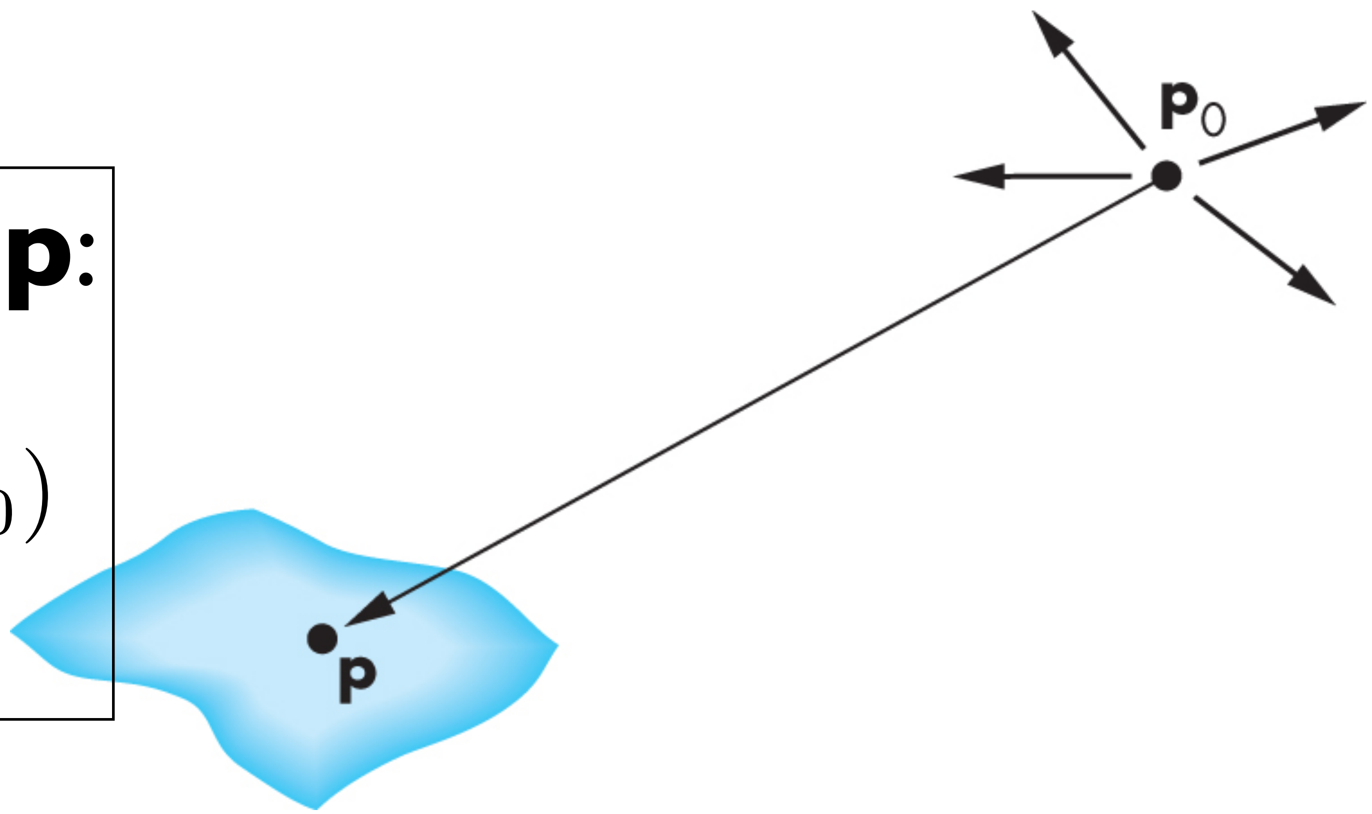
$$L_a$$

Point light source

$$\mathbf{L}(\mathbf{p}_0) = \begin{bmatrix} L_r(\mathbf{p}_0) \\ L_g(\mathbf{p}_0) \\ L_b(\mathbf{p}_0) \end{bmatrix} \qquad L(\mathbf{p}_0)$$

illumination intensity at **p**:

$$l(\mathbf{p}, \mathbf{p}_0) = \frac{1}{|\mathbf{p} - \mathbf{p}_0|^2} \mathbf{L}(\mathbf{p}_0)$$

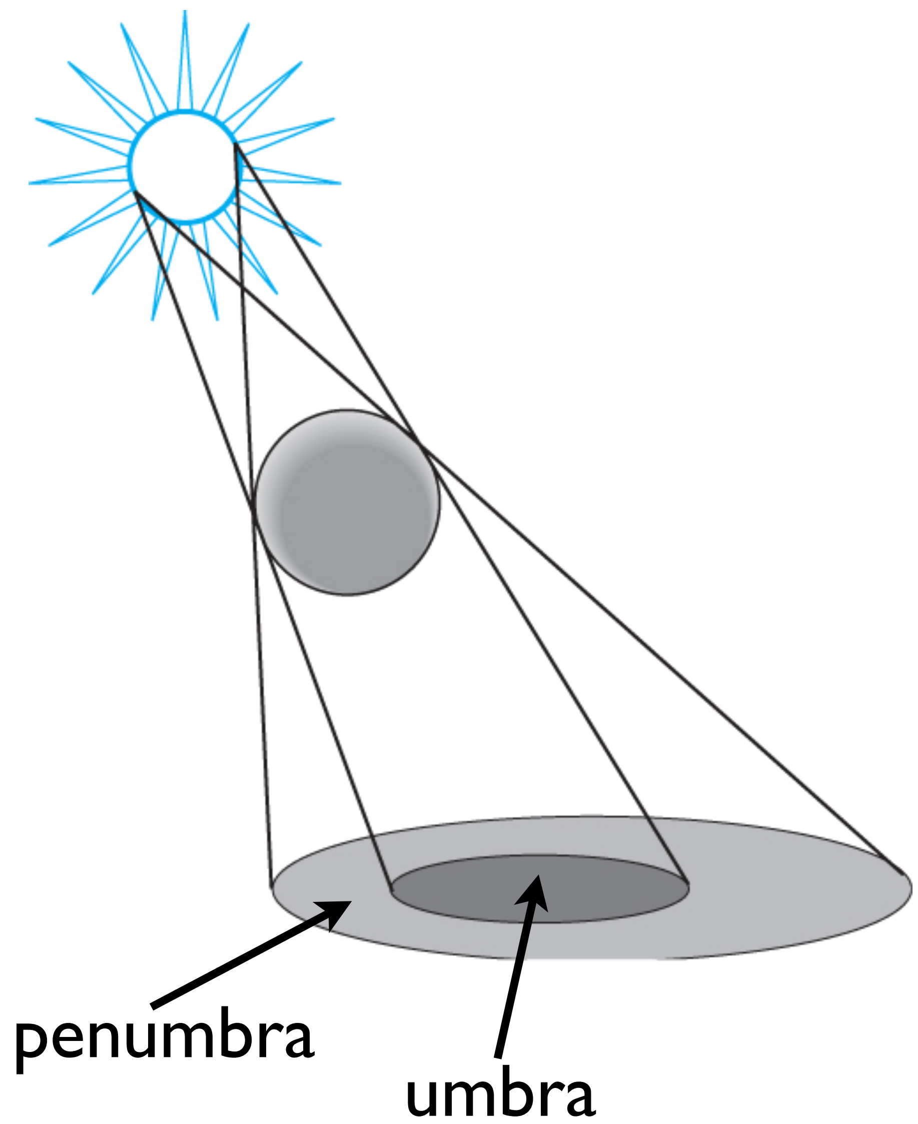


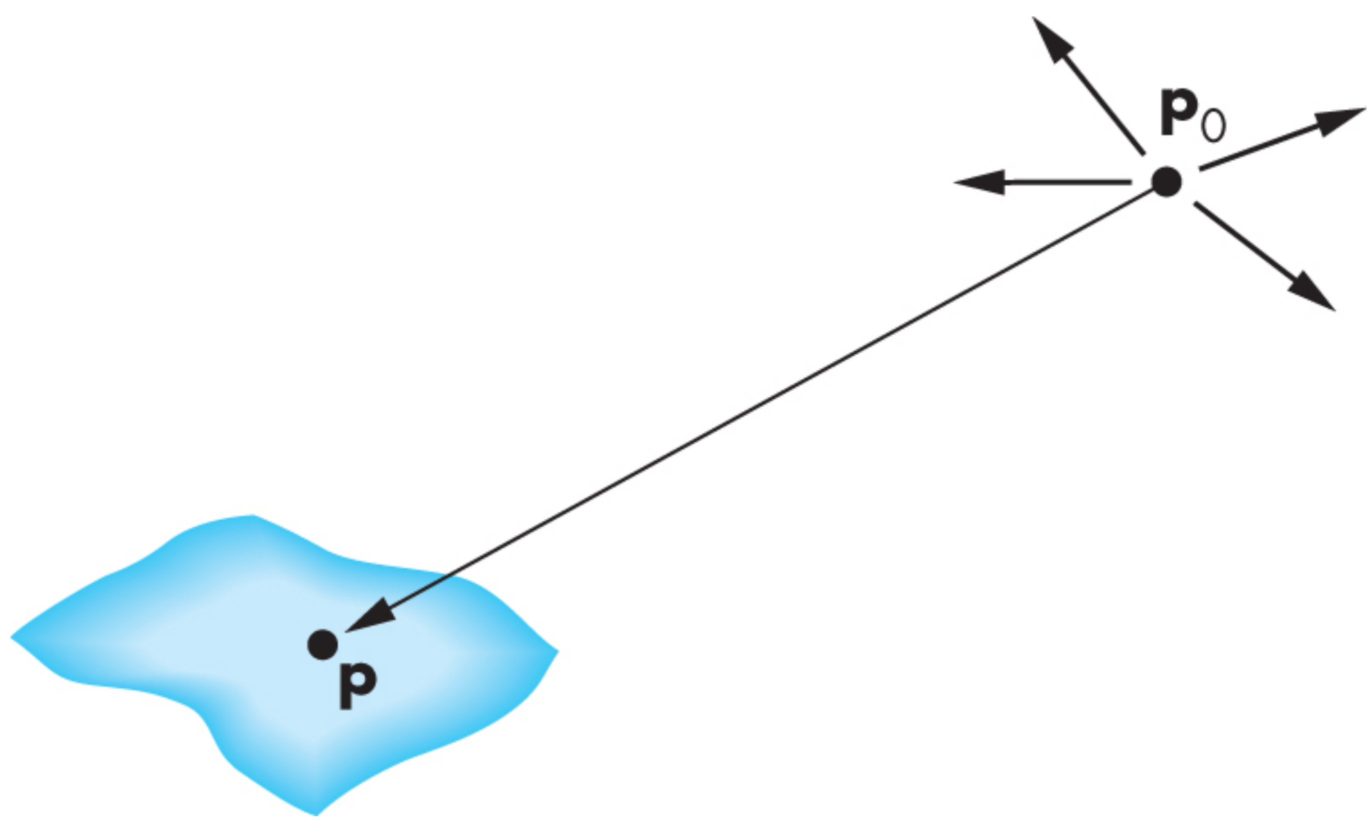
[Angel and Shreiner]

Point light source

Most real-world scenes have large light sources

Point light sources alone aren't too realistic
- **add ambient light to mitigate high contrast**

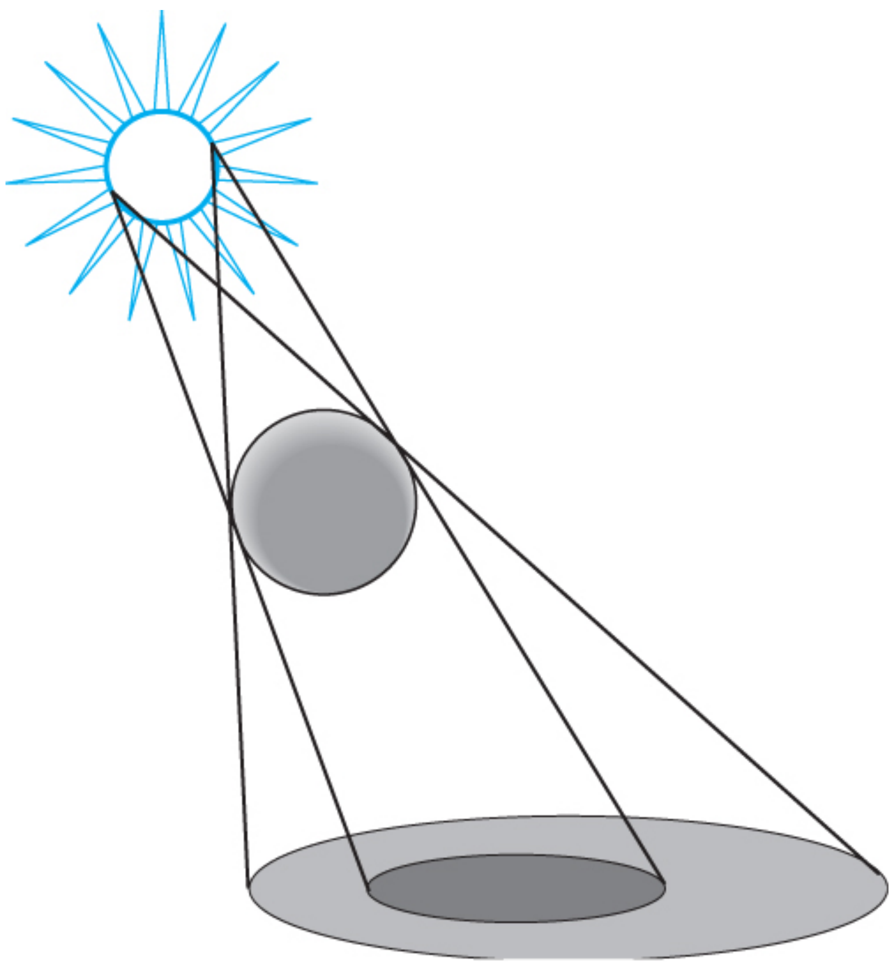




[Angel and Shreiner]

Point light source

Most real-world scenes have large light sources



Point light sources alone aren't too realistic
- drop off intensity more slowly

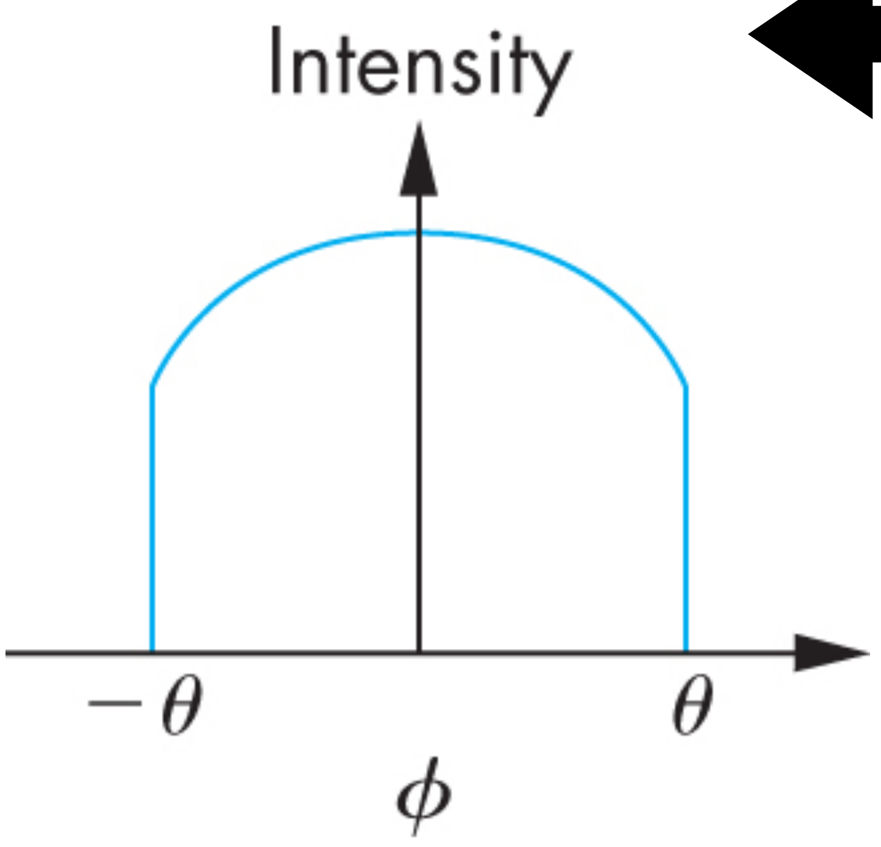
$$l(\mathbf{p}, \mathbf{p}_0) = \frac{1}{d^2}\mathbf{L}(\mathbf{p}_0)$$

$$\Downarrow$$

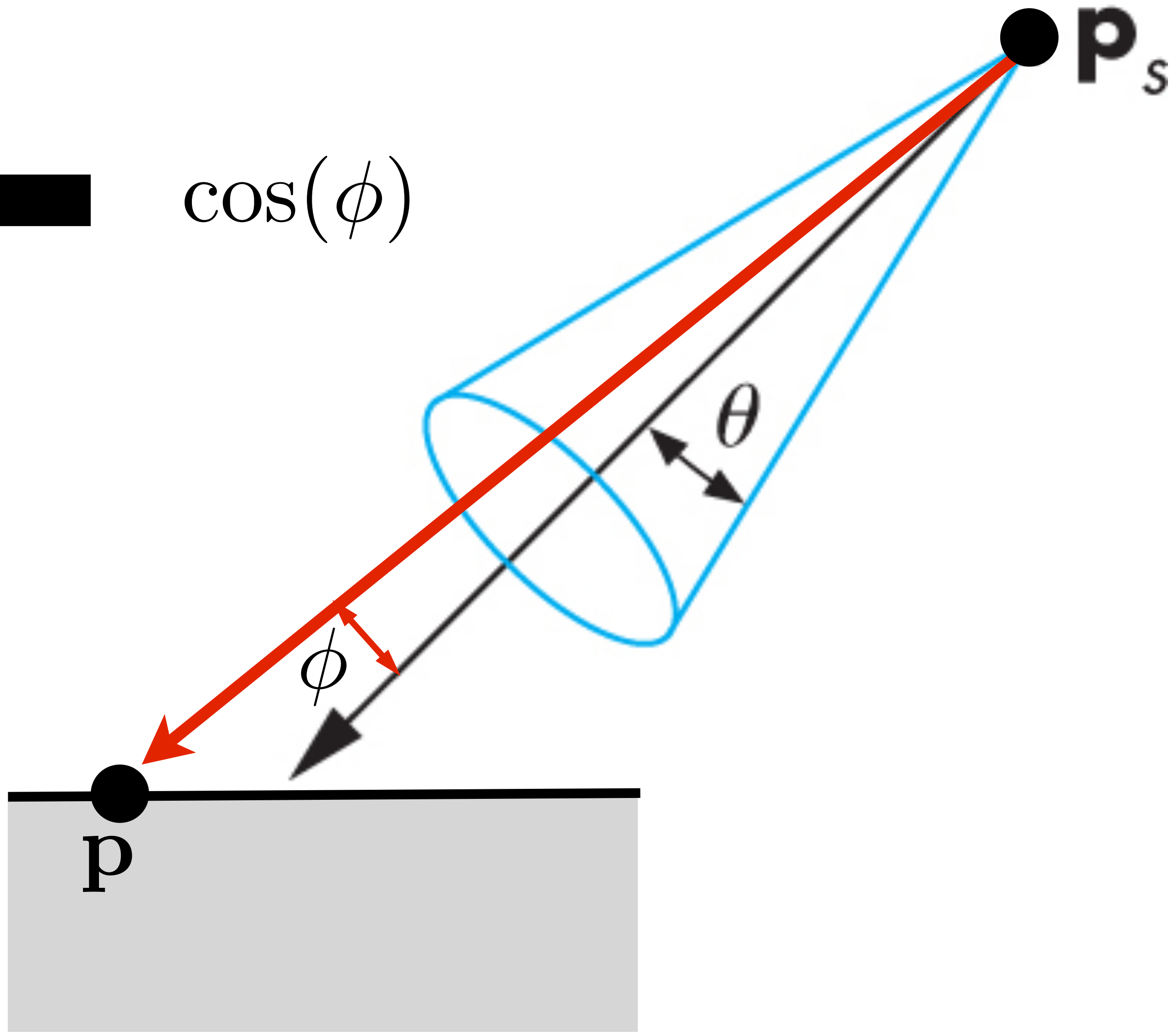
$$l(\mathbf{p}, \mathbf{p}_0) = \frac{1}{a + bd + cd^2}\mathbf{L}(\mathbf{p}_0)$$

[Angel and Shreiner]

Spotlights

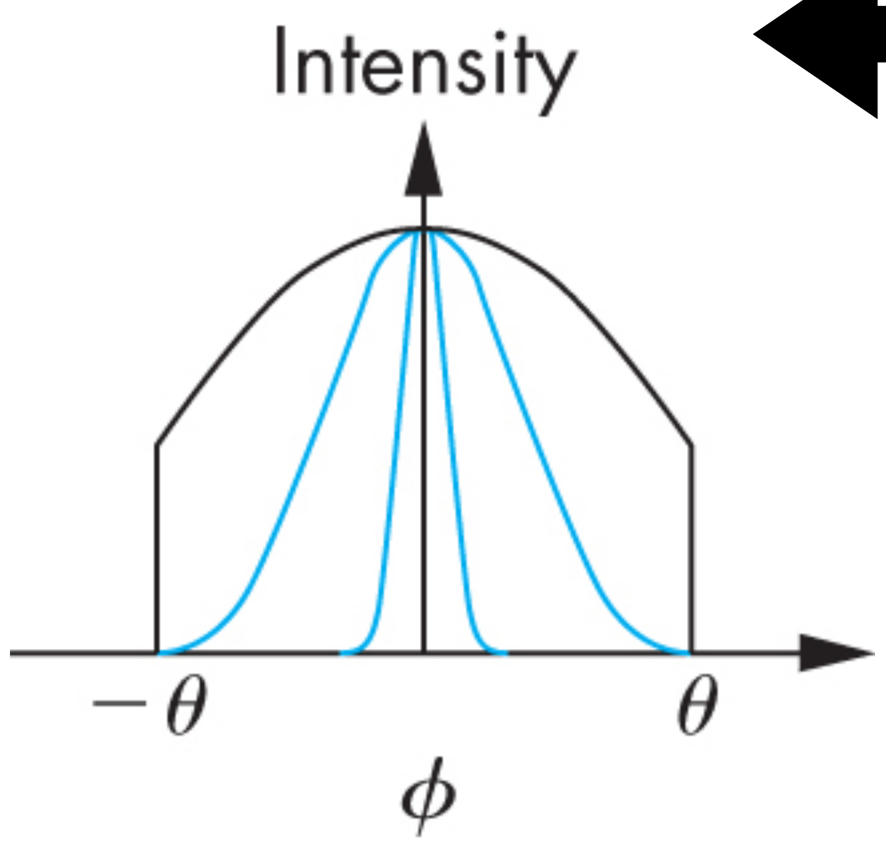


$\cos(\phi)$

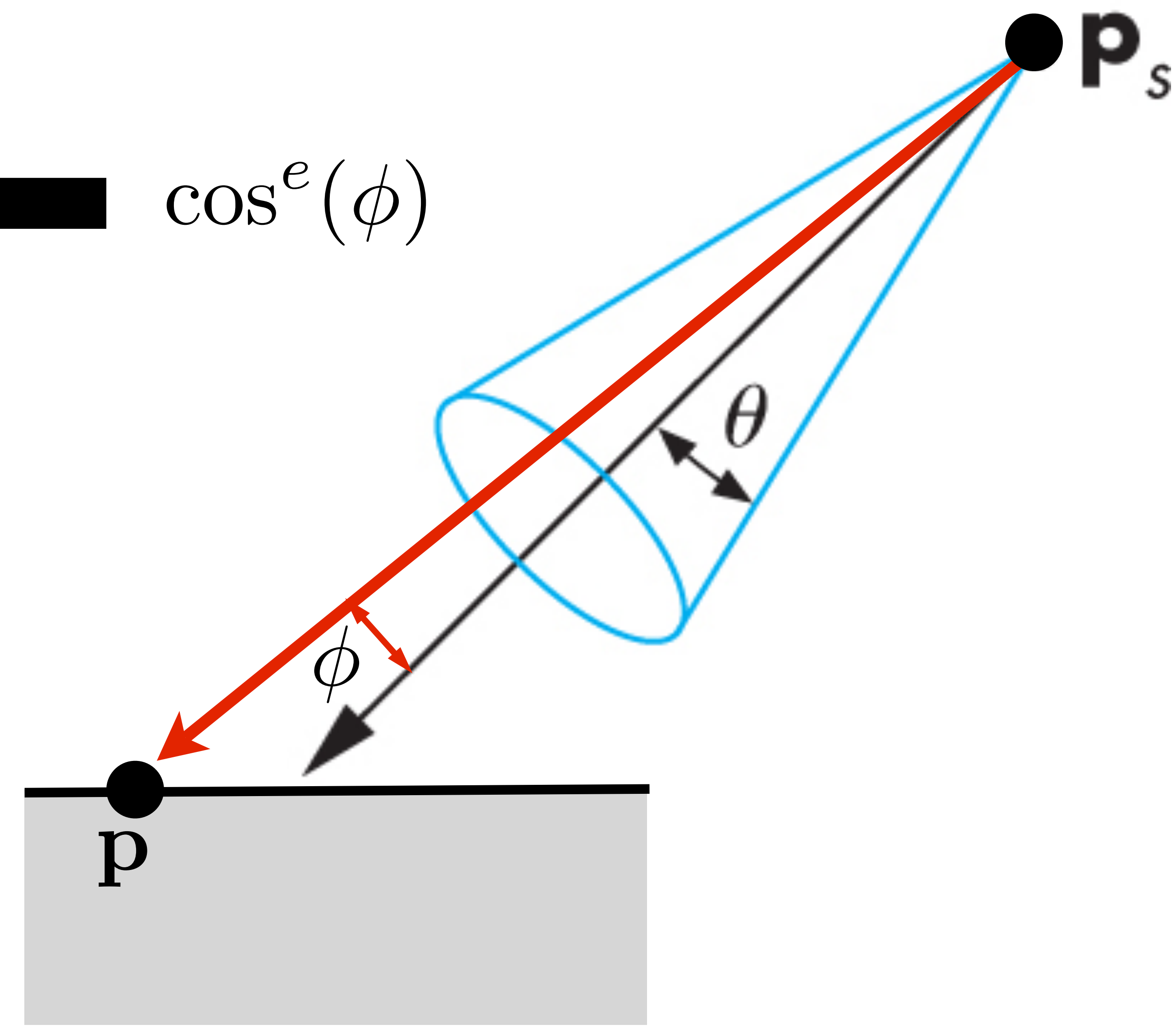


[Angel and Shreiner]

Spotlights



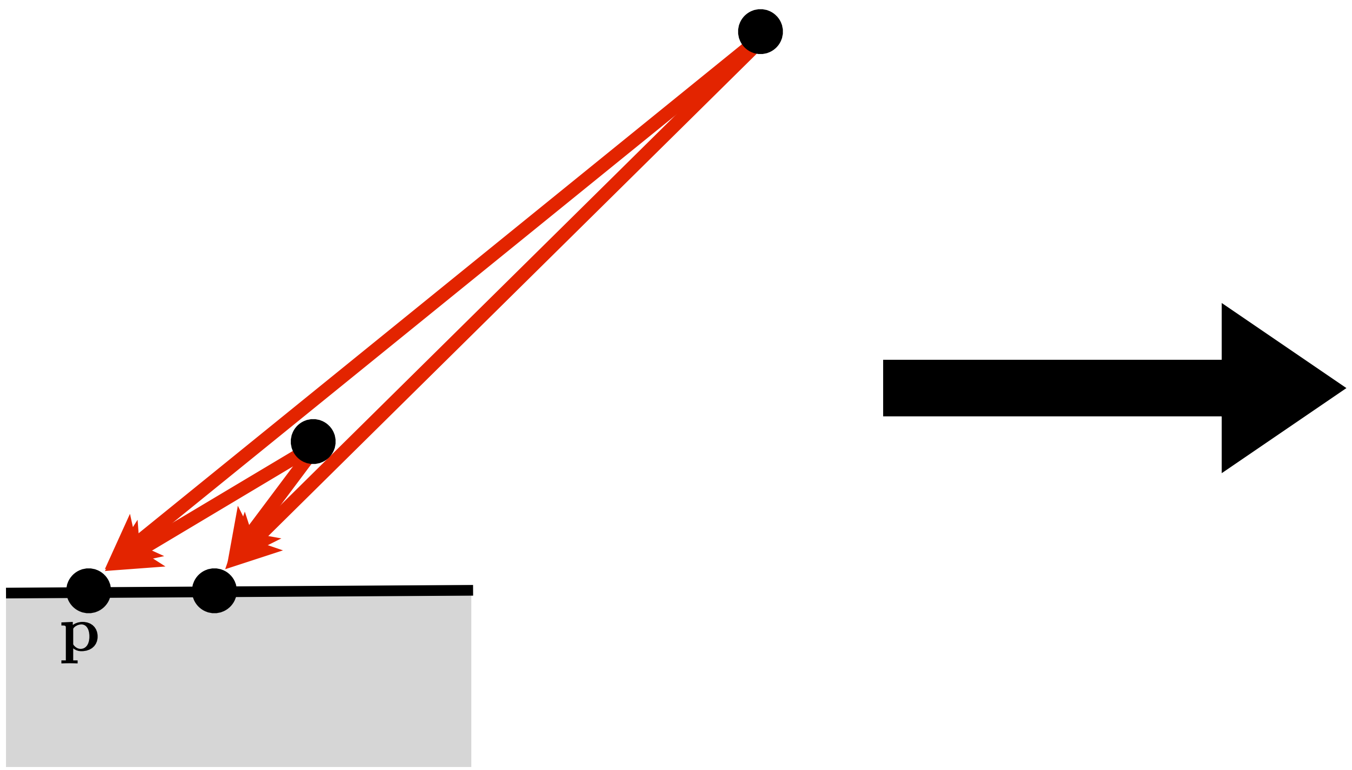
$\cos^e(\phi)$



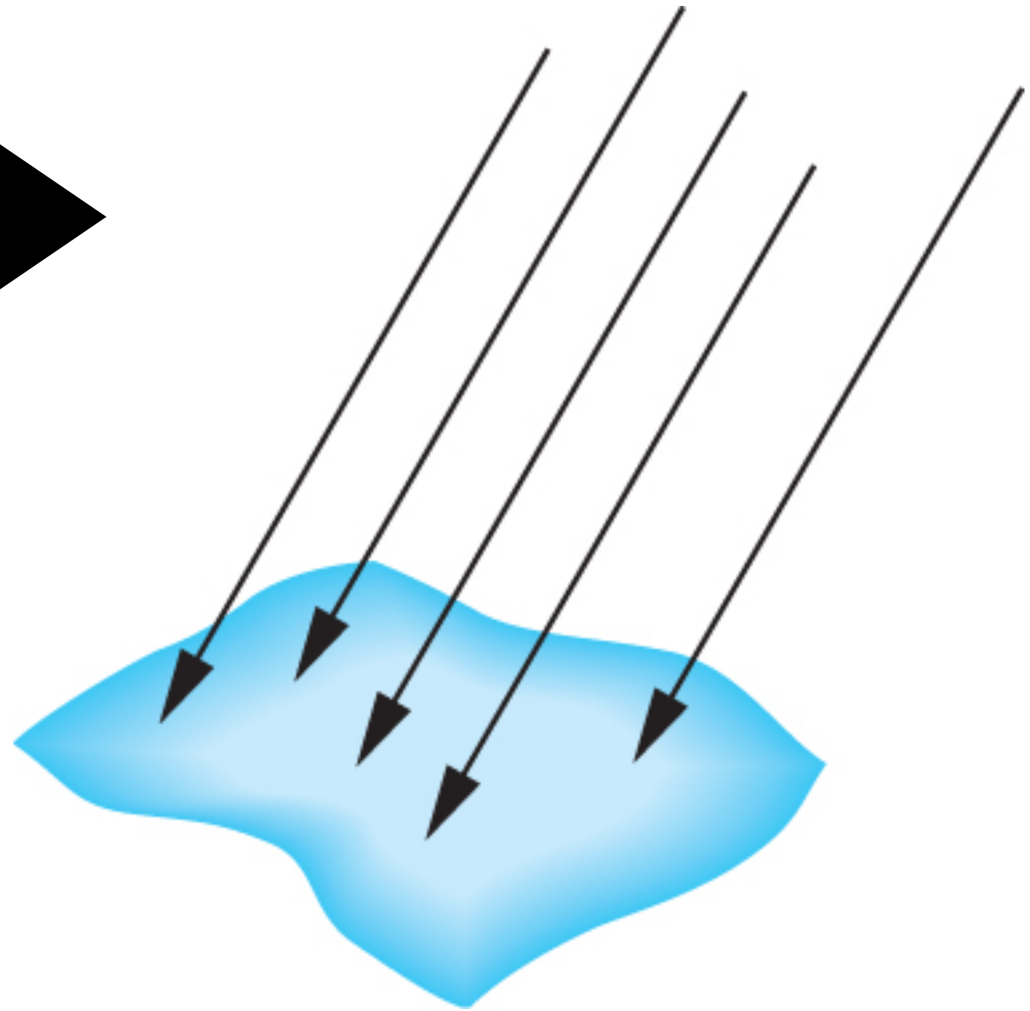
$$\cos^e(\phi)\; l(\mathbf{p}, \mathbf{p}_s)$$

[Angel and Shreiner]

Distant light source

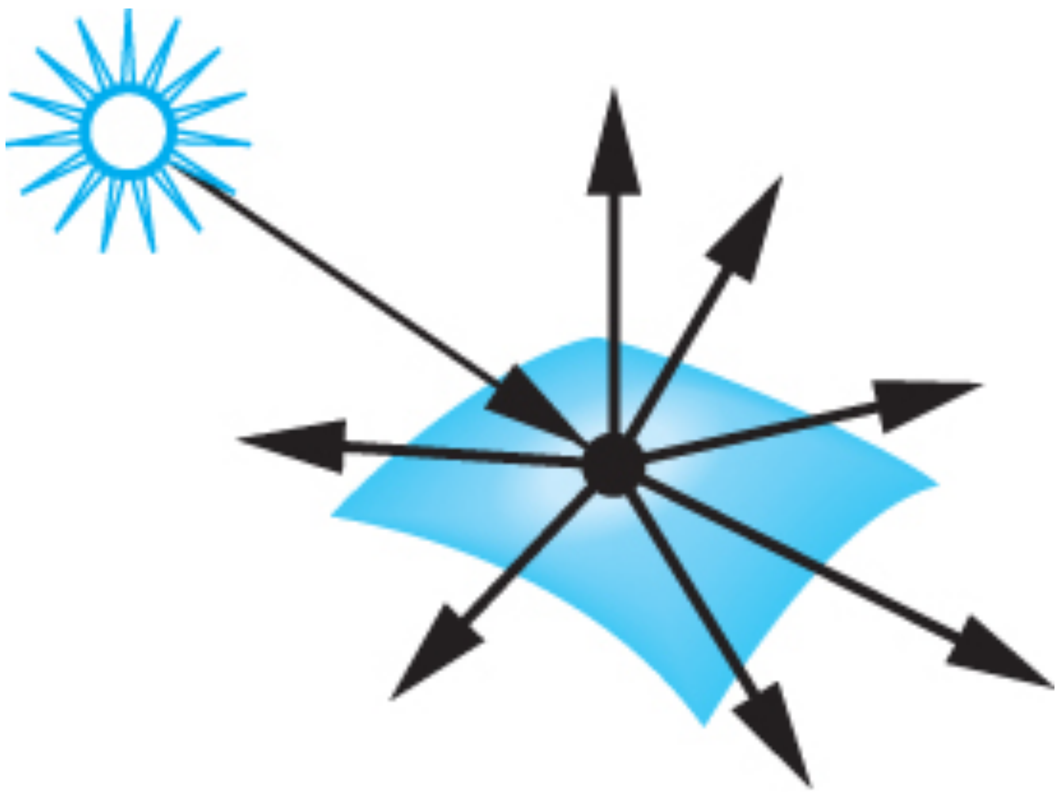


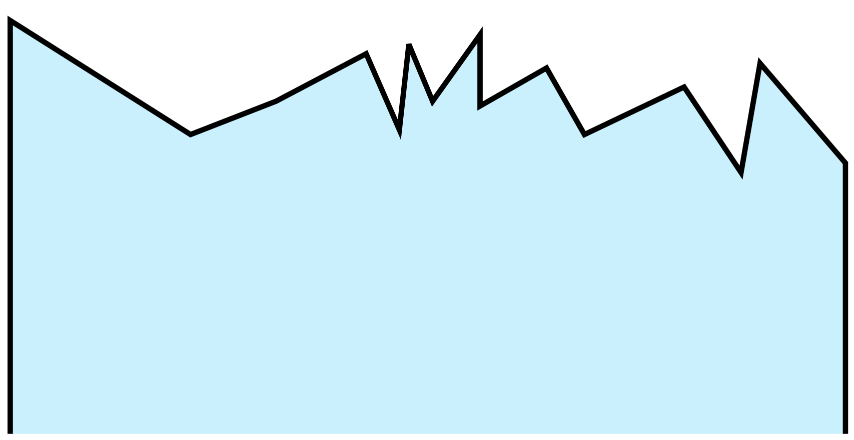
characterized by direction



[Angel and Shreiner]

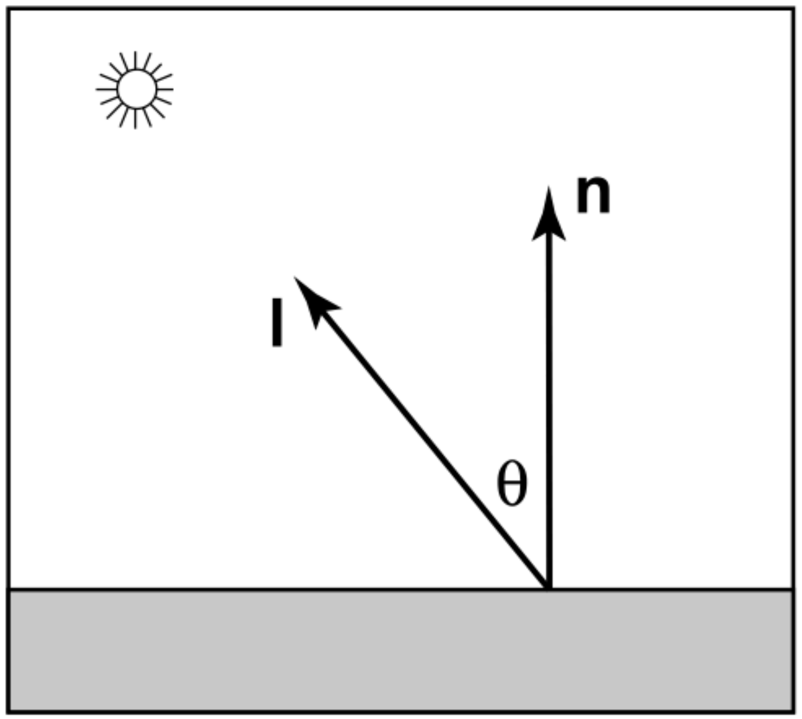
Lambertian Reflection Model

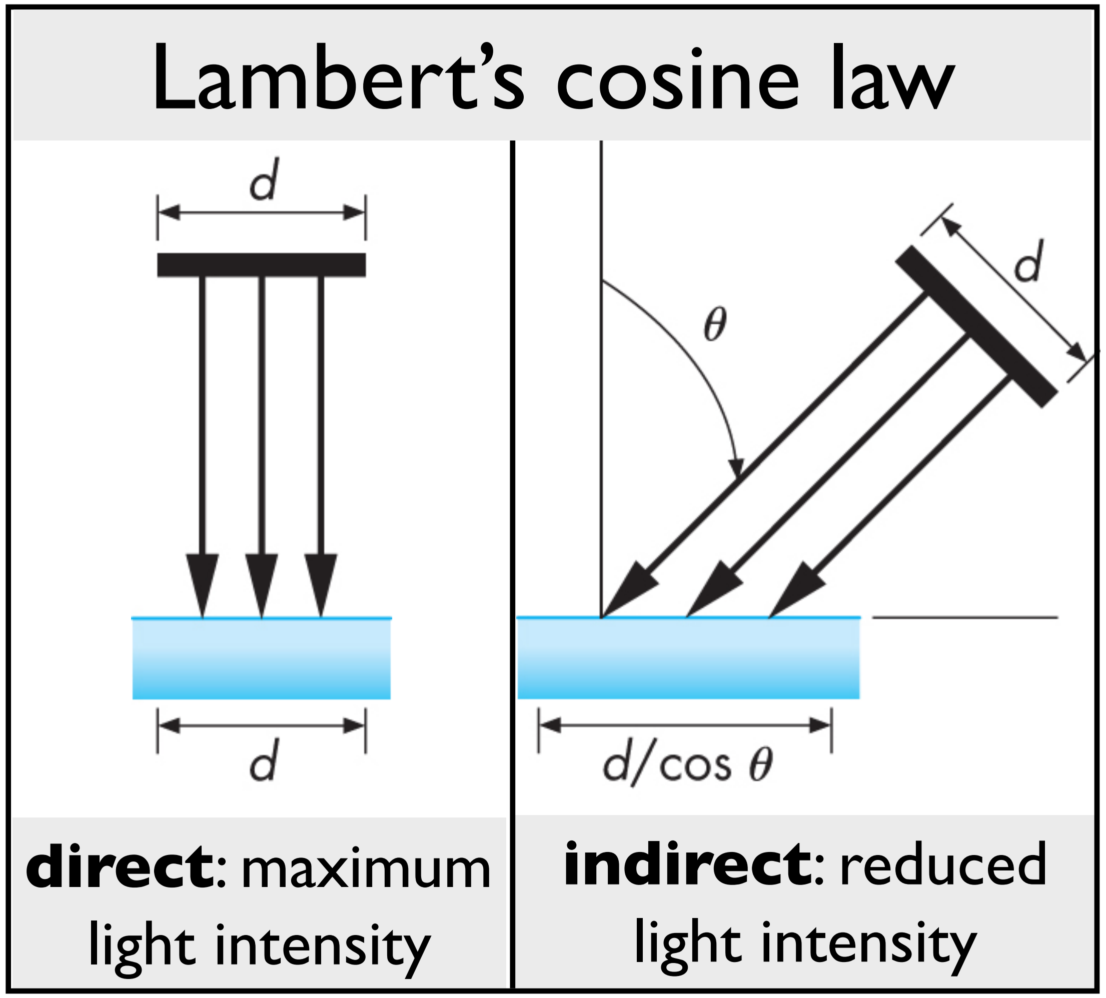




http://www.andreafisherpottery.com/cgi-bin/artistlnk.cgi?Jody_Folwell

Lambertian Reflection Model

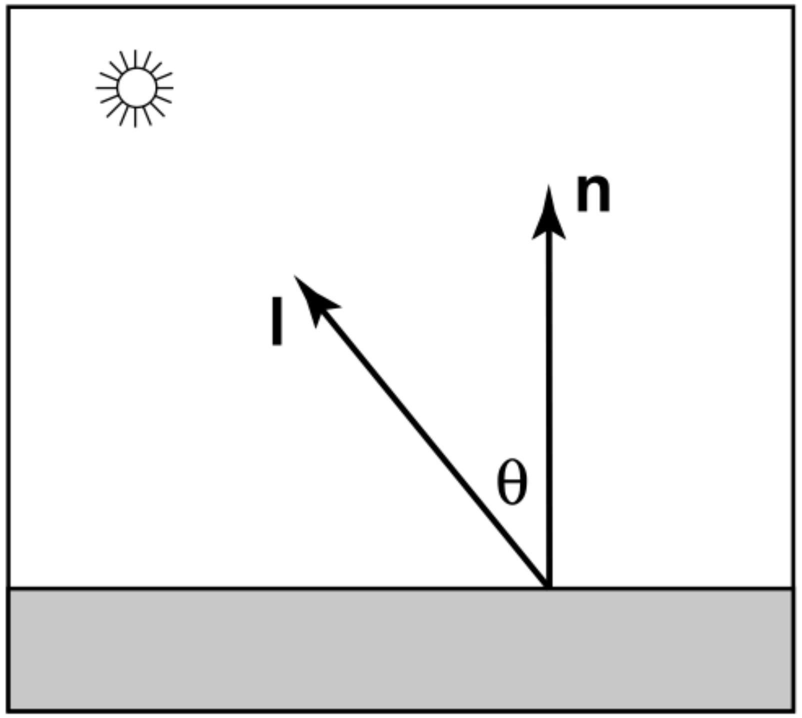


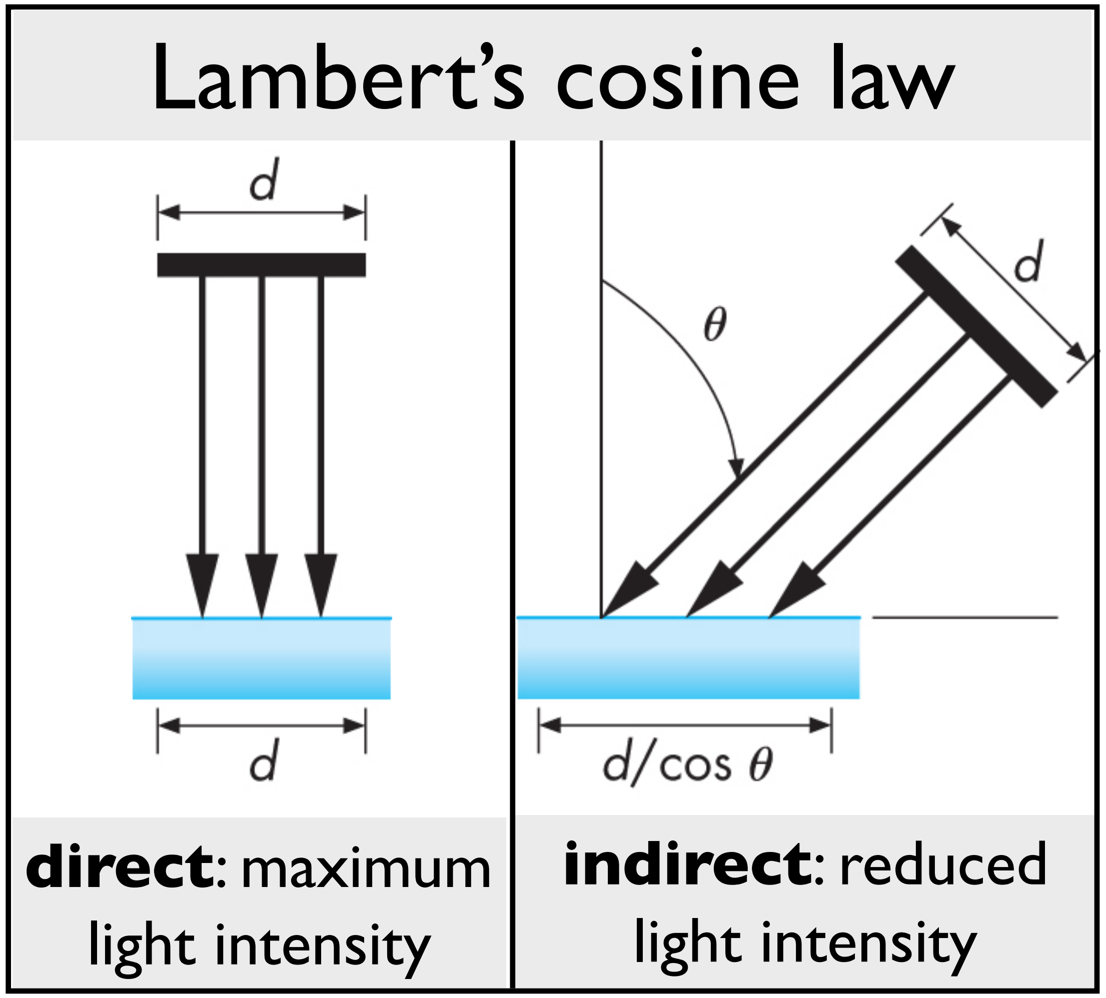


$$I \propto \cos\theta$$

color intensity

Lambertian Reflection Model

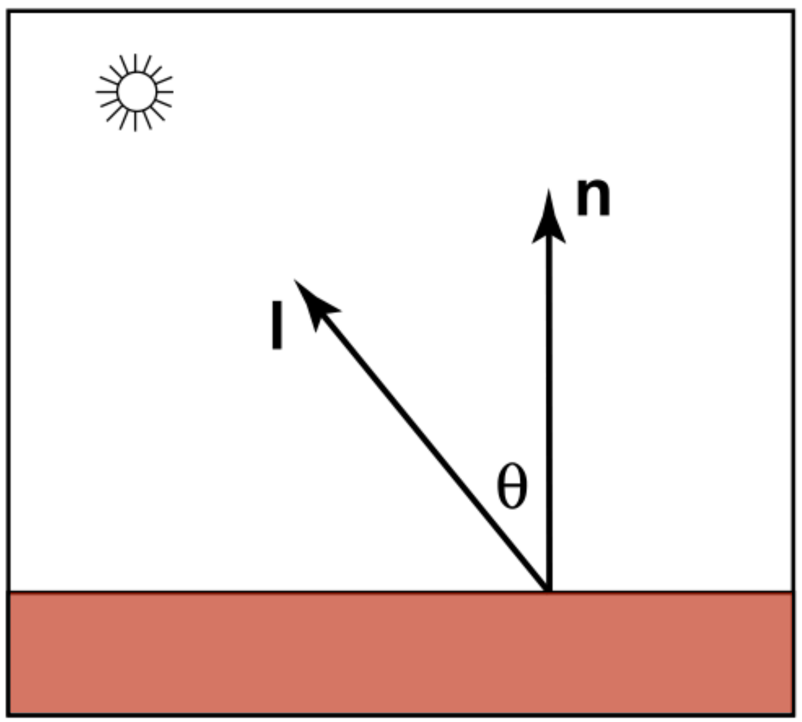


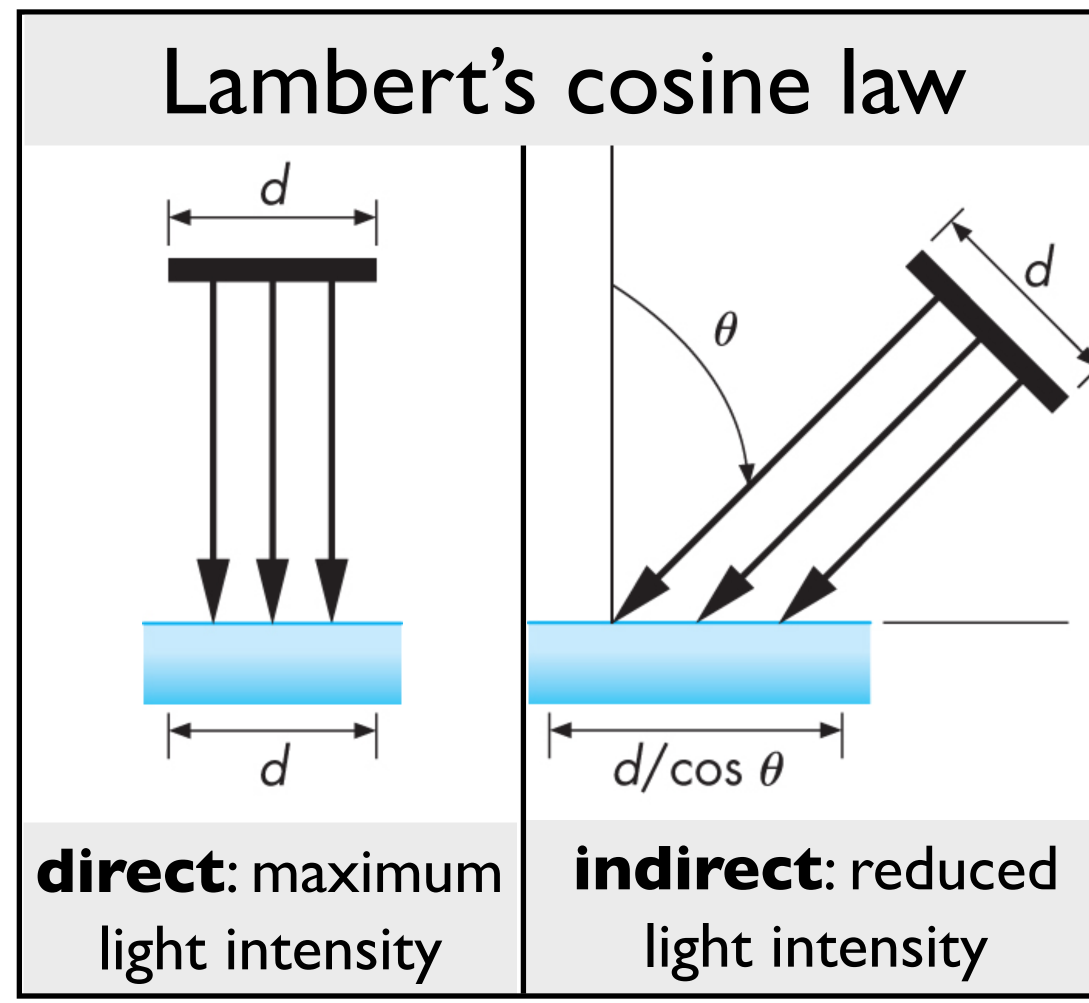


$$I \propto \mathbf{n} \cdot \mathbf{l}$$

color intensity

Lambertian Reflection Model



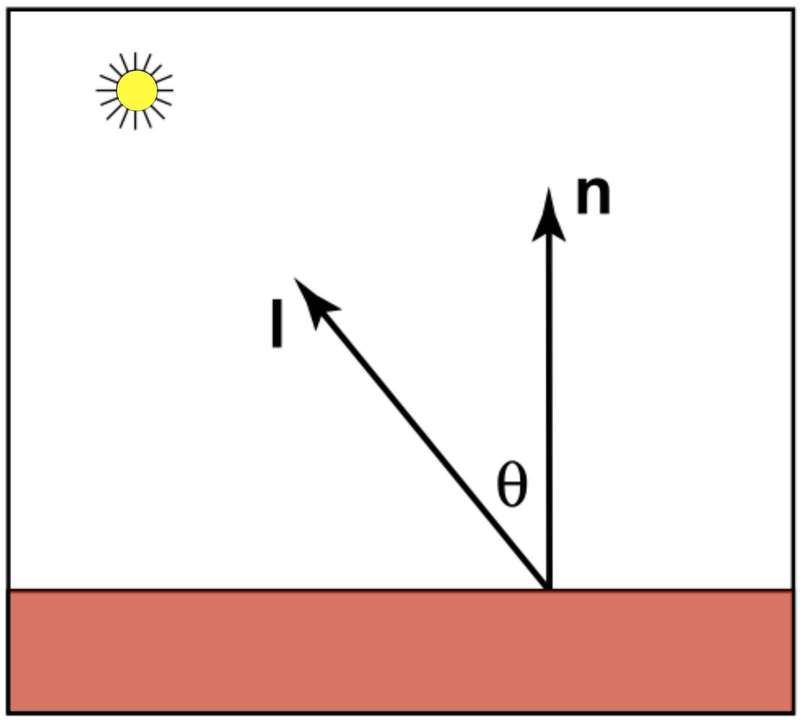


$$I \propto R\mathbf{n} \cdot \mathbf{l}$$

color intensity

reflectance

Lambertian Reflection Model

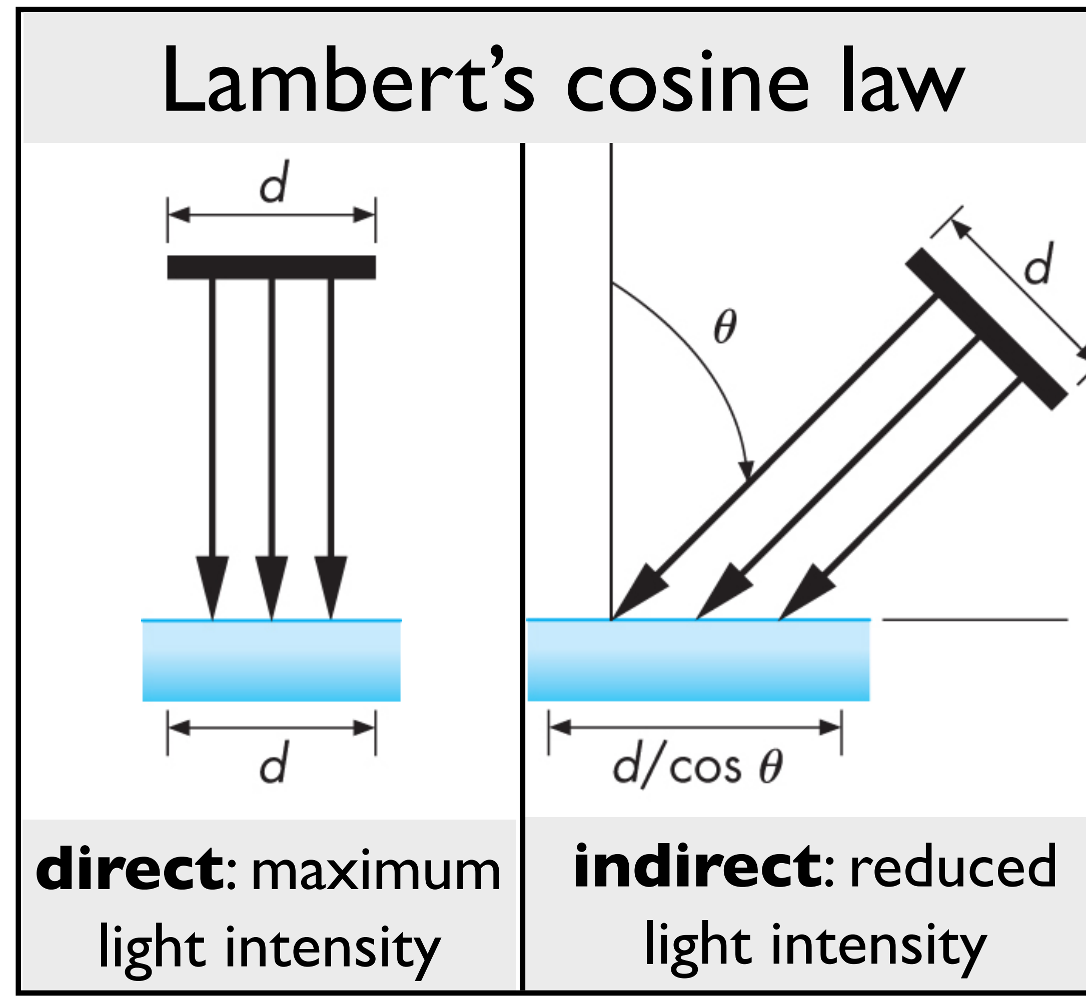


illumination

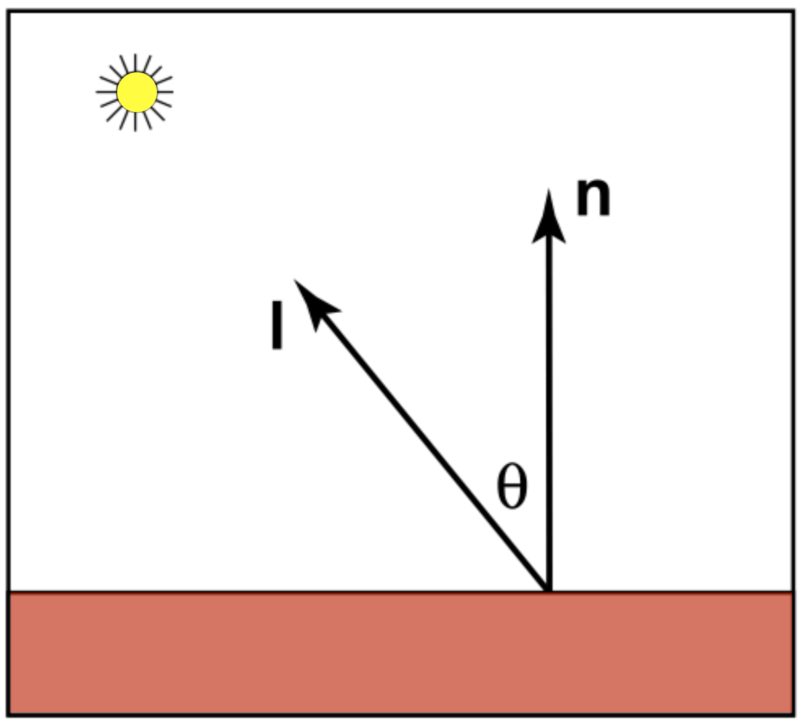
$$I = LR\mathbf{n}\cdot\mathbf{l}$$

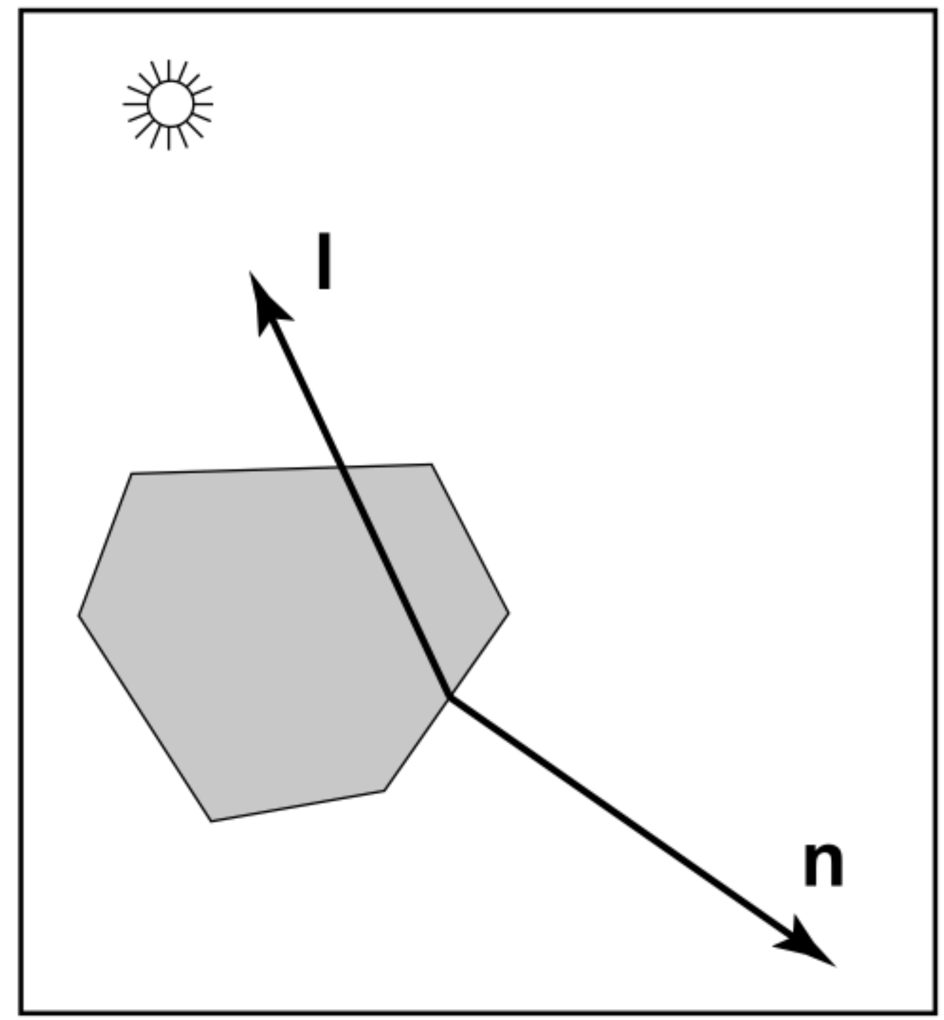
color intensity

reflectance



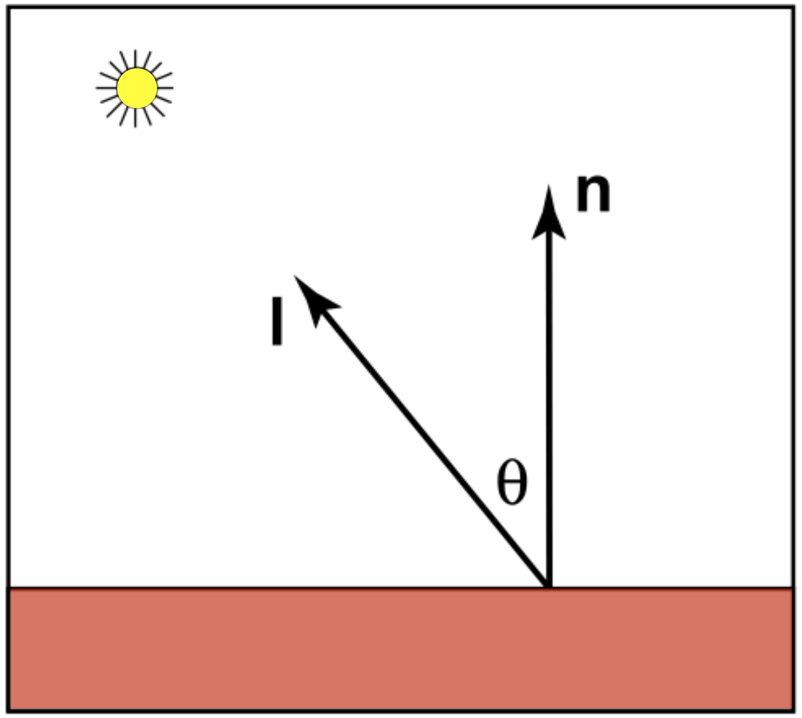
Lambertian Reflection Model



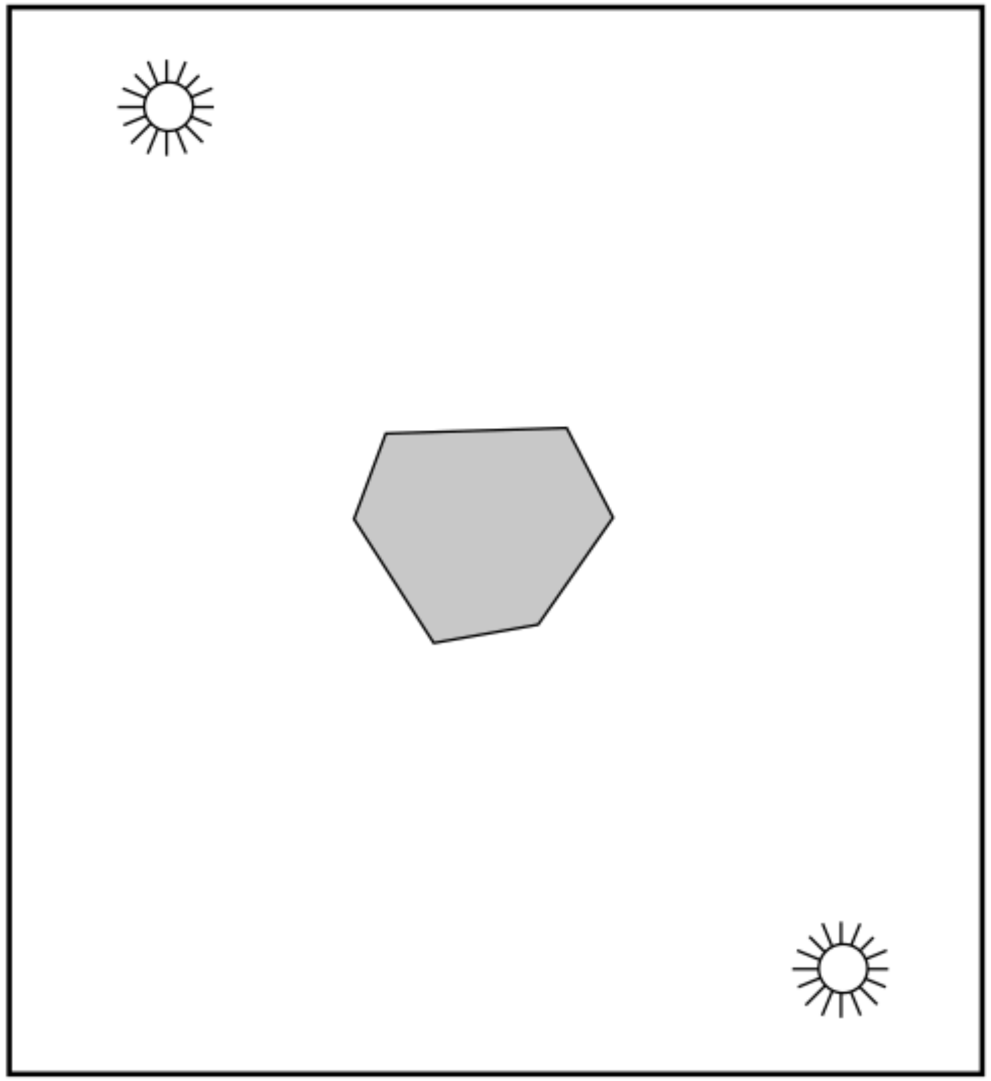


$$I = LR\max(0, \mathbf{n} \cdot \mathbf{l})$$

Lambertian Reflection Model



$$I = LR|\mathbf{n} \cdot \mathbf{l}|$$

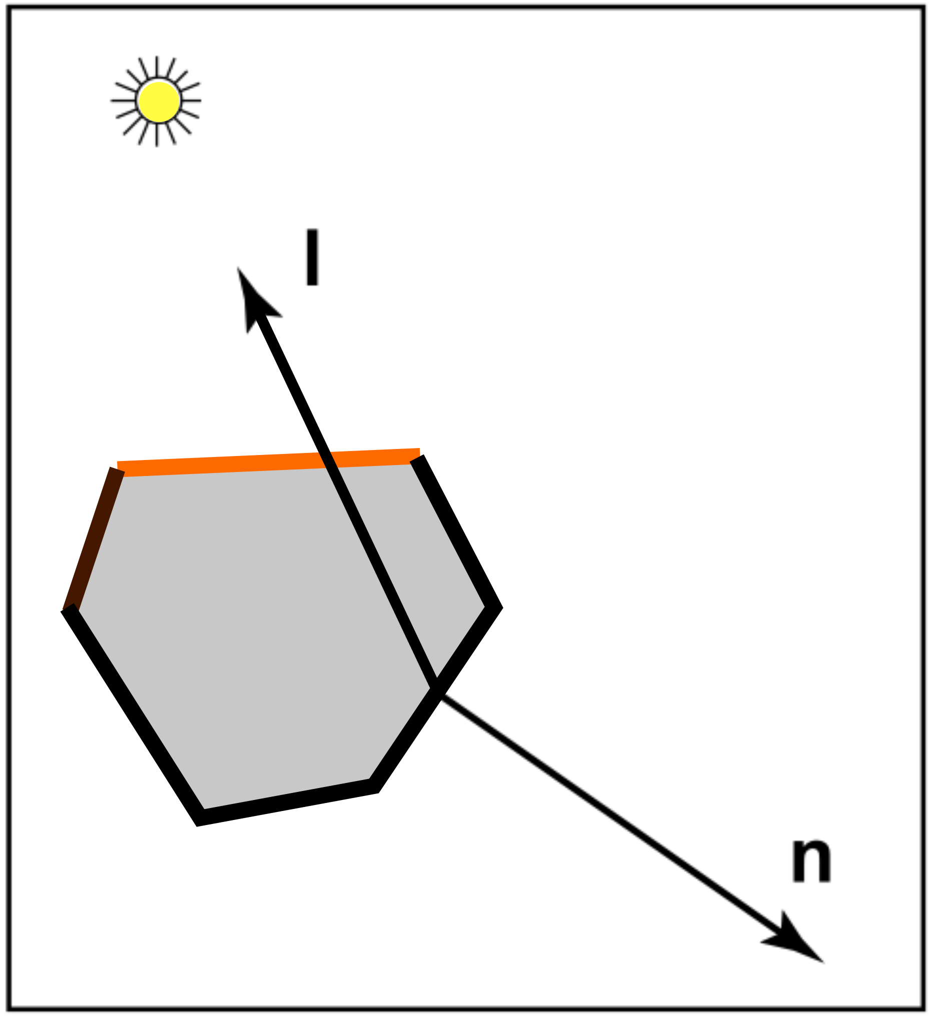


two-sided lighting

Ambient Reflection

$$I = LR\max(0, \mathbf{n} \cdot \mathbf{l})$$

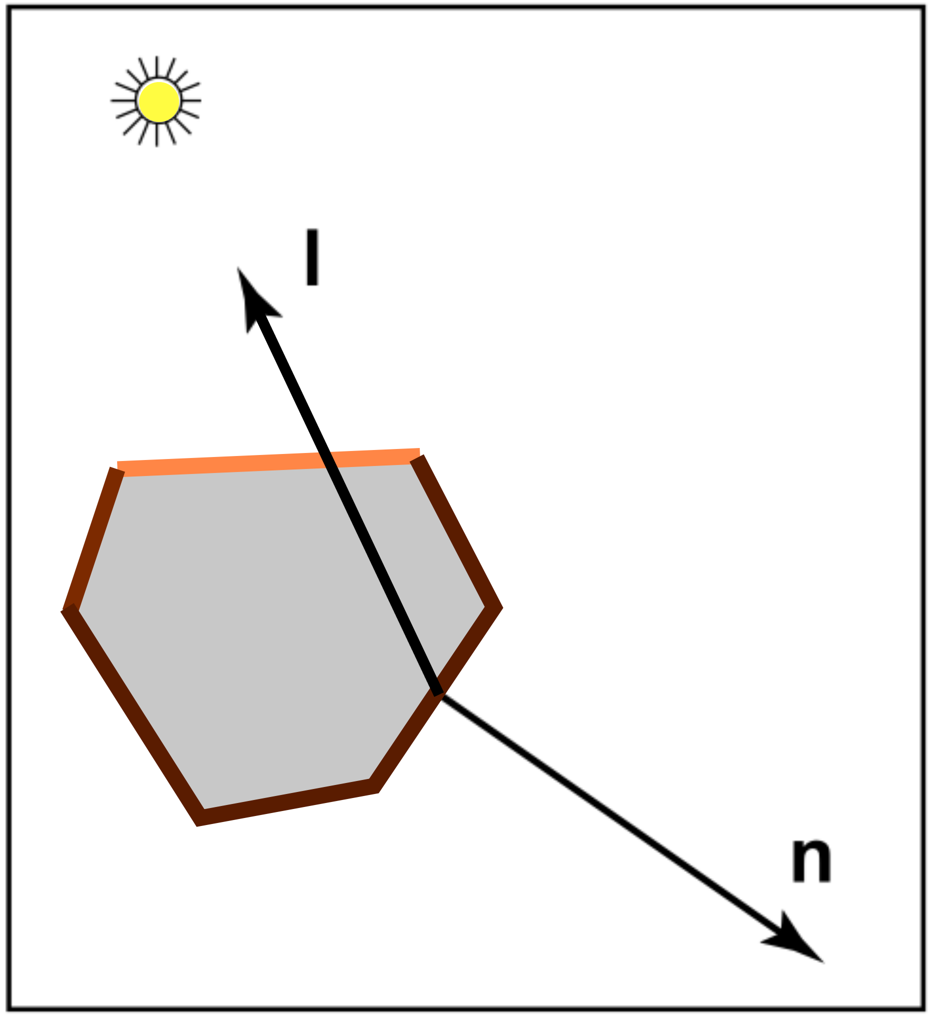
Surfaces facing away from the light will be totally **black**



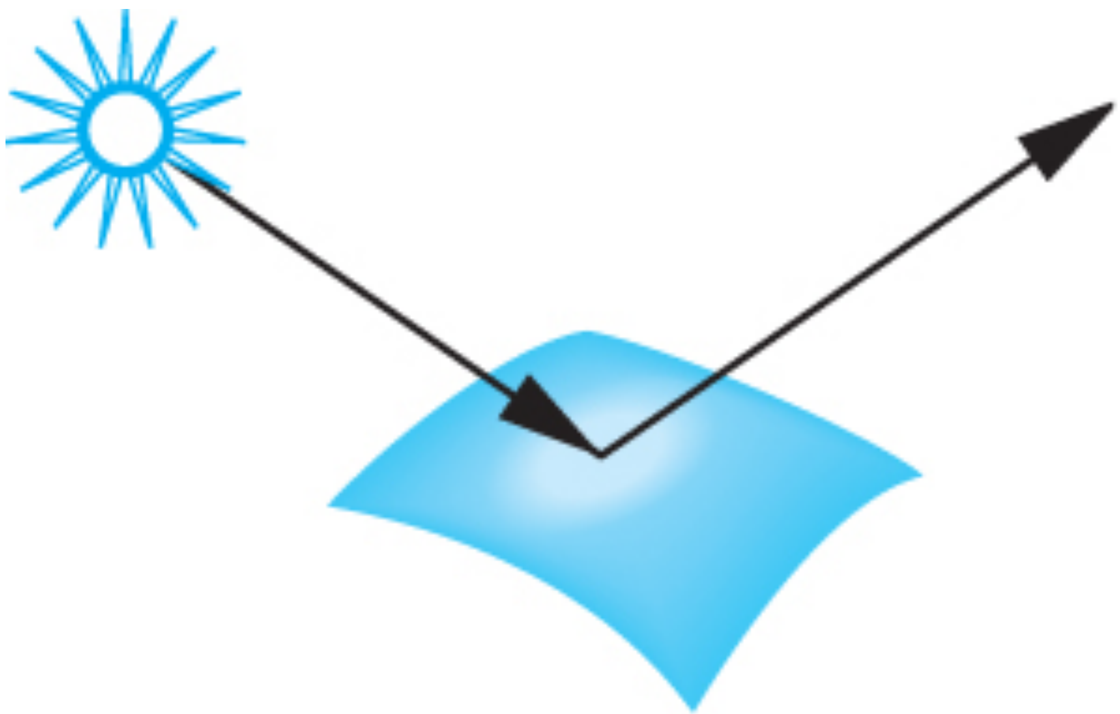
Ambient Reflection

$$I = L_a R_a + L_d R_d \max(0, \mathbf{n} \cdot \mathbf{l})$$

All surfaces get same amount of ambient light

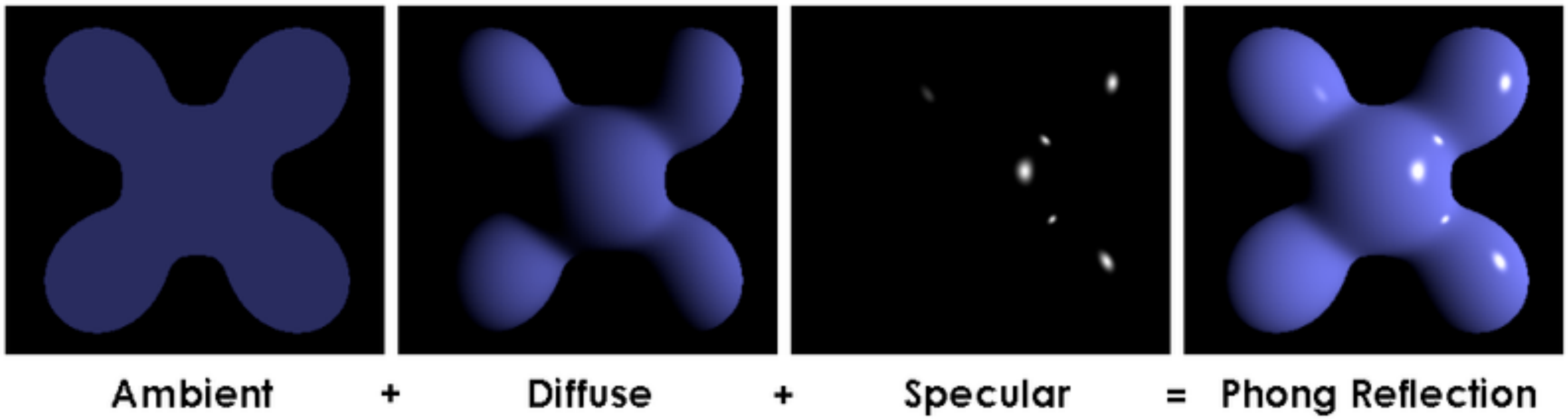


Phong Reflection Model



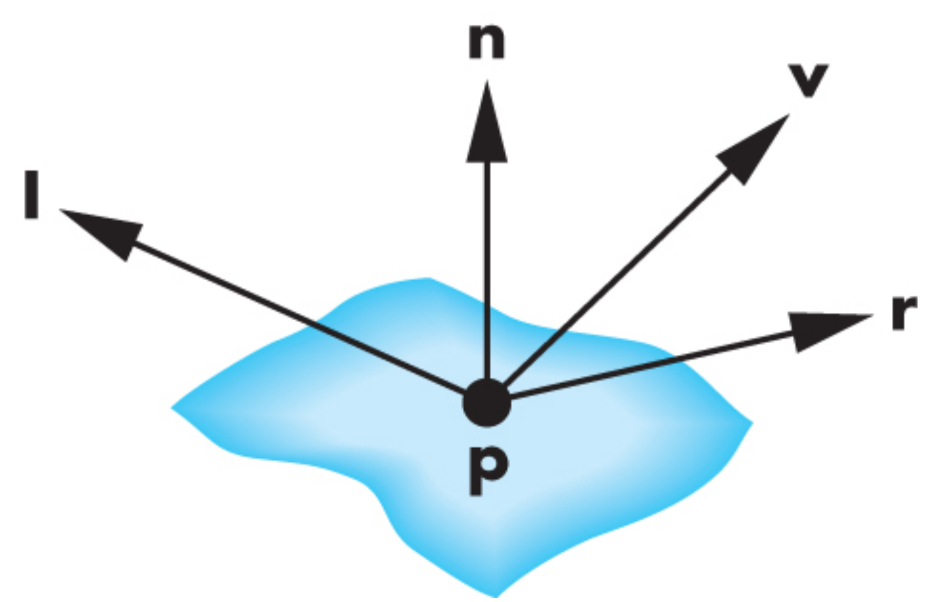
http://www.andreafisherpottery.com/cgi-bin/artistlnk.cgi?Jody_Folwell

Phong Reflection Model

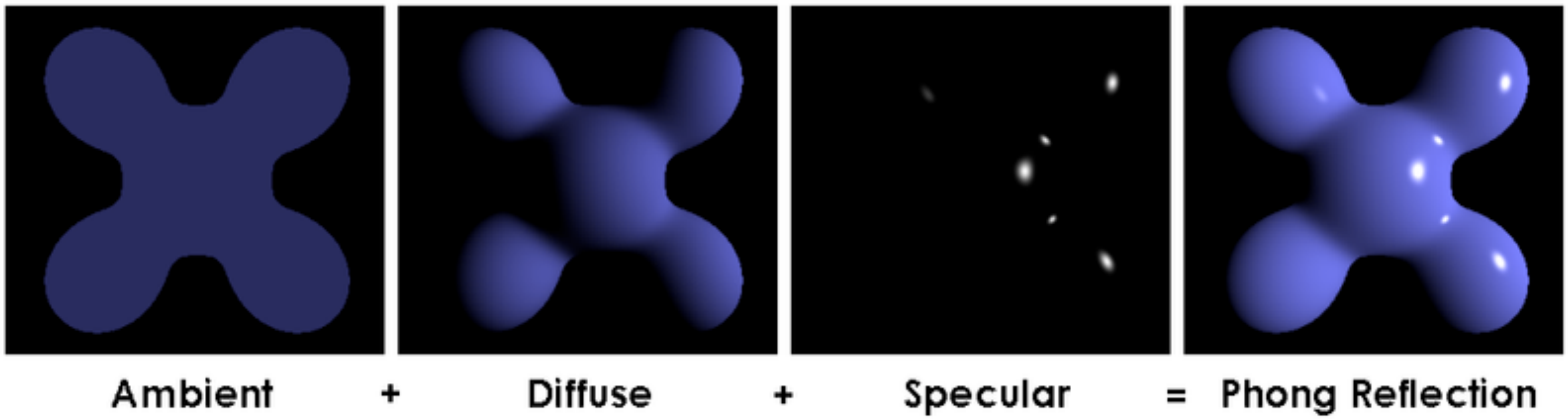


[Brad Smith, Wikimedia Commons]

- efficient, reasonably realistic
- 3 components
- 4 vectors



Phong Reflection Model



[Brad Smith, Wikimedia Commons]

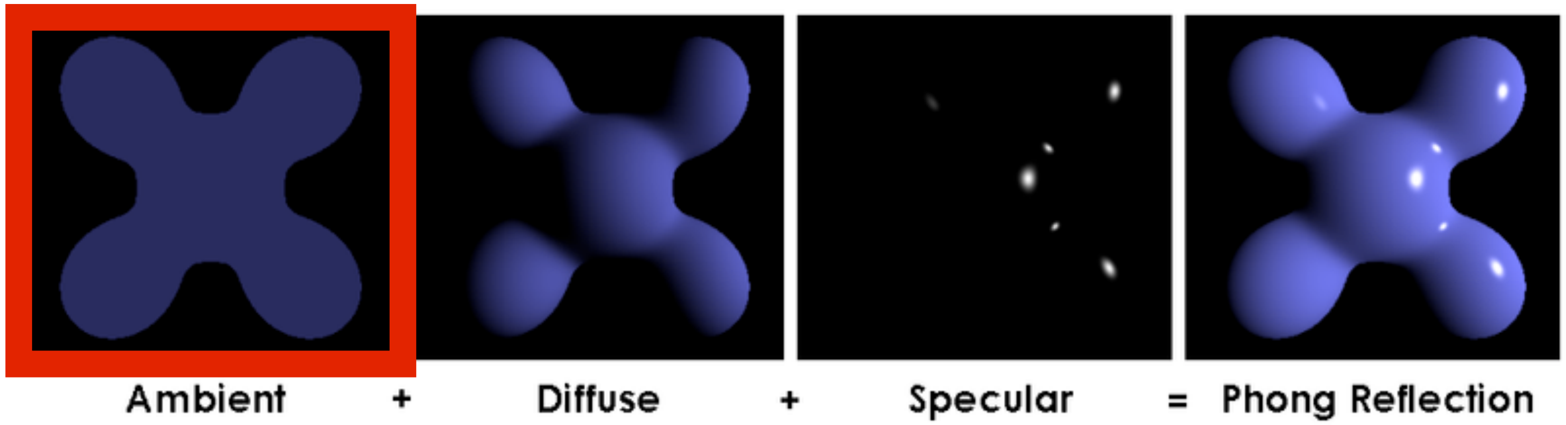
$$
\begin{aligned}
I &= I_a + I_d + I_s \\
&= R_a L_a + R_d L_d \max(0, \mathbf{l} \cdot \mathbf{n}) + R_s L_s \max(0, \cos\phi)^{\alpha}
\end{aligned}
$$

color intensity

reflectance

illumination

Ambient reflection

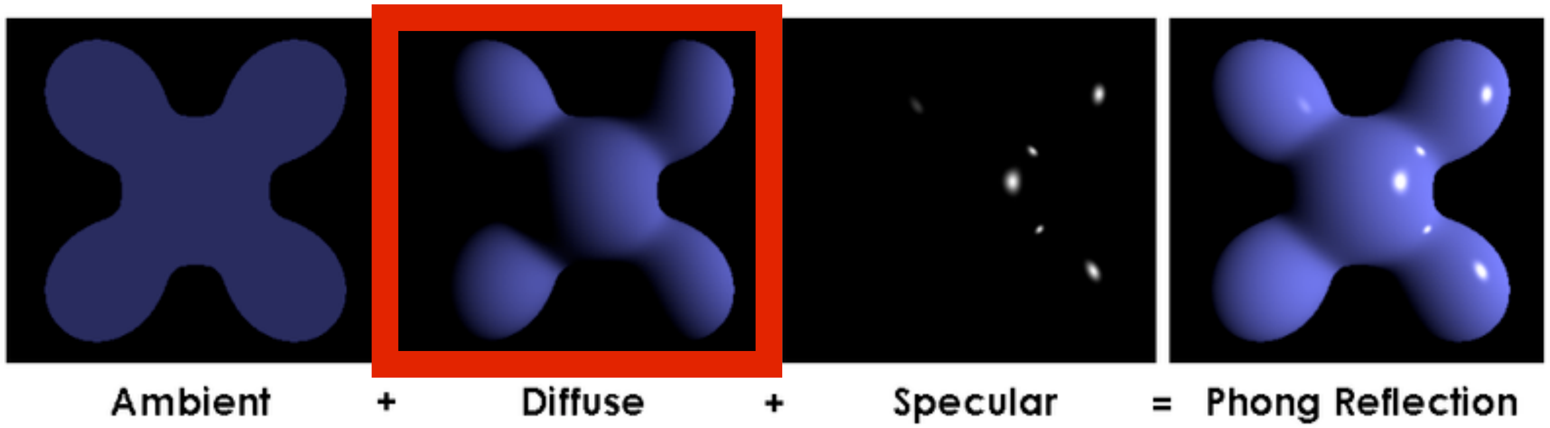


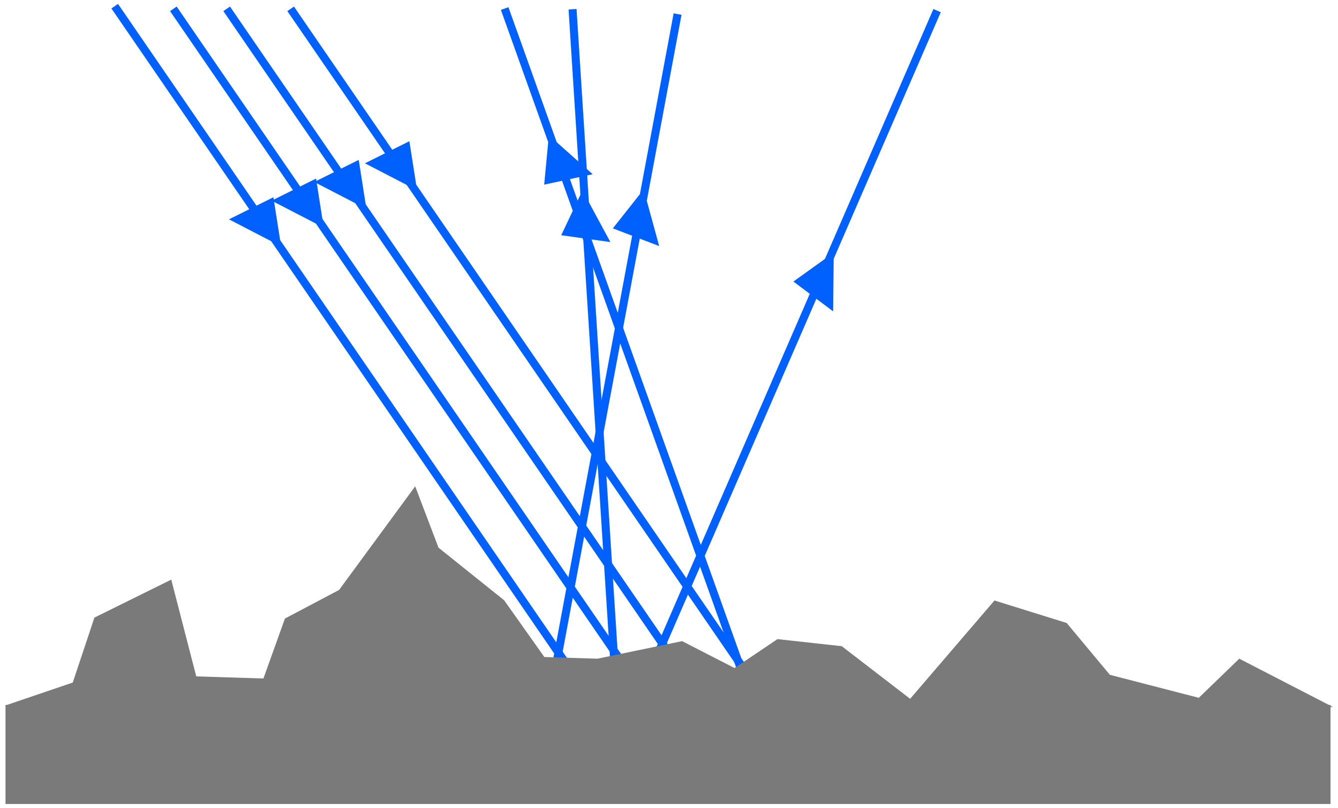
different ambient coefficients for different colors

$$I_a = R_a L_a, \qquad 0 \leq R_a \leq 1$$

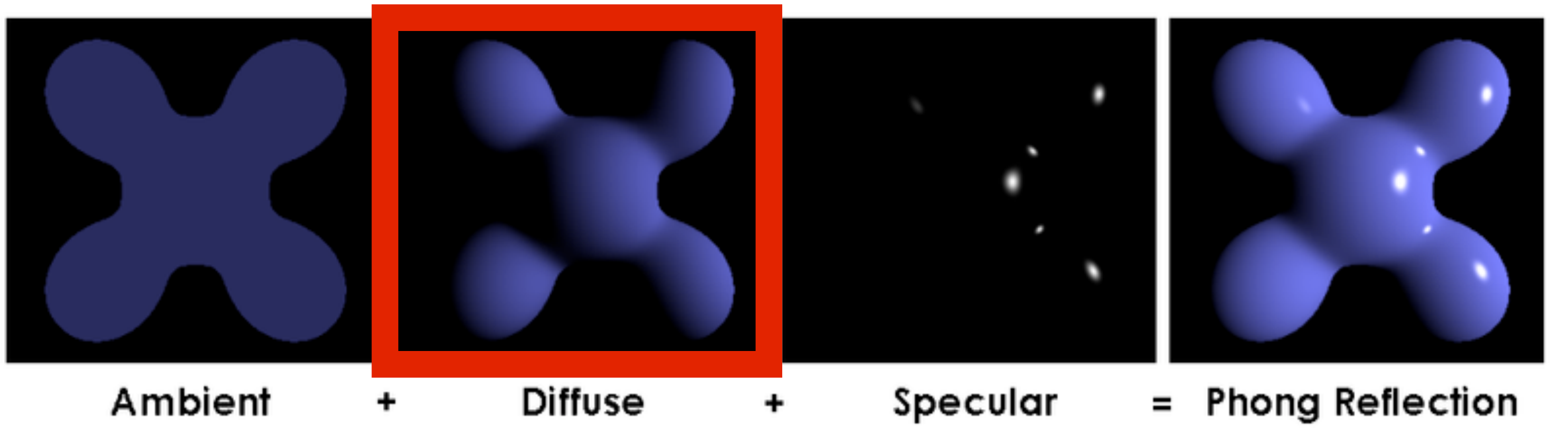
ambient reflection coefficient

Diffuse reflection



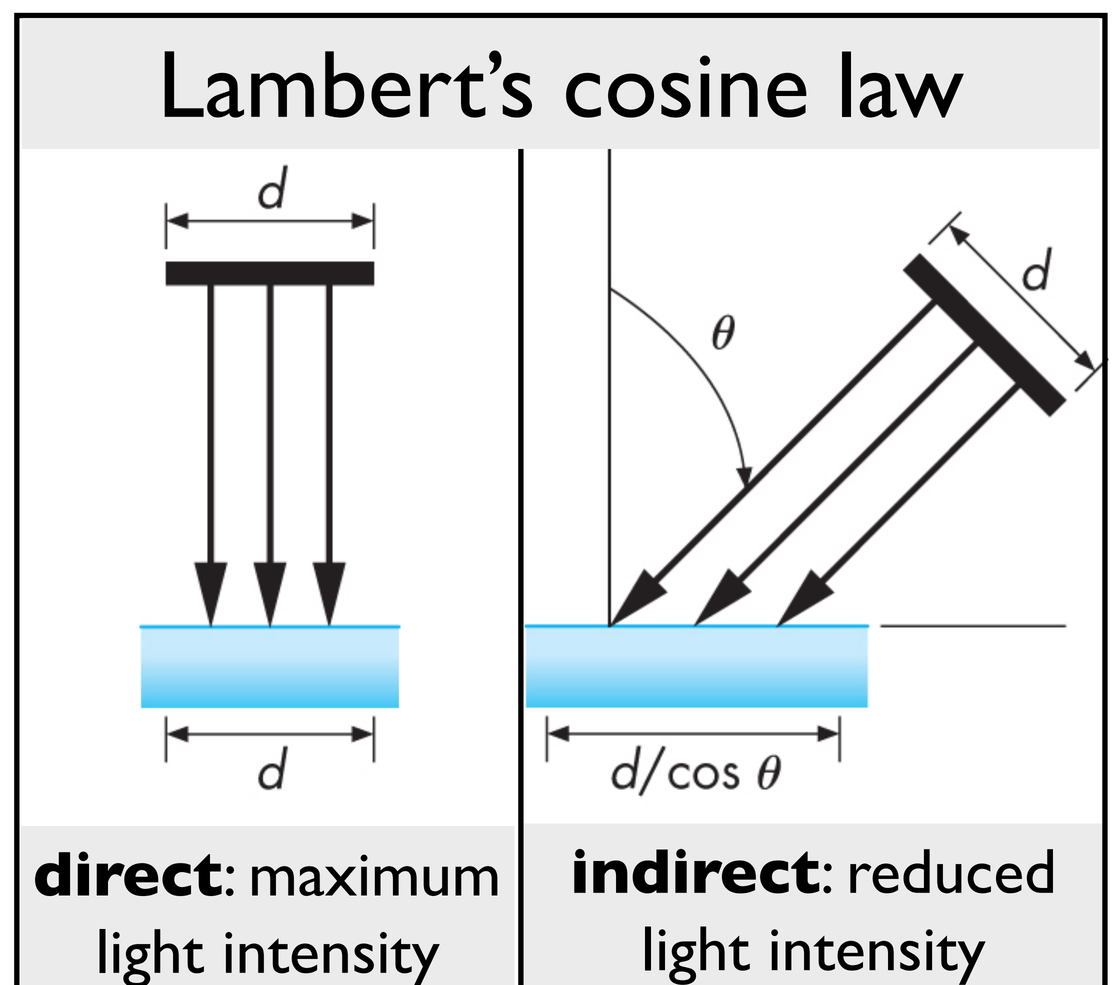


Diffuse reflection

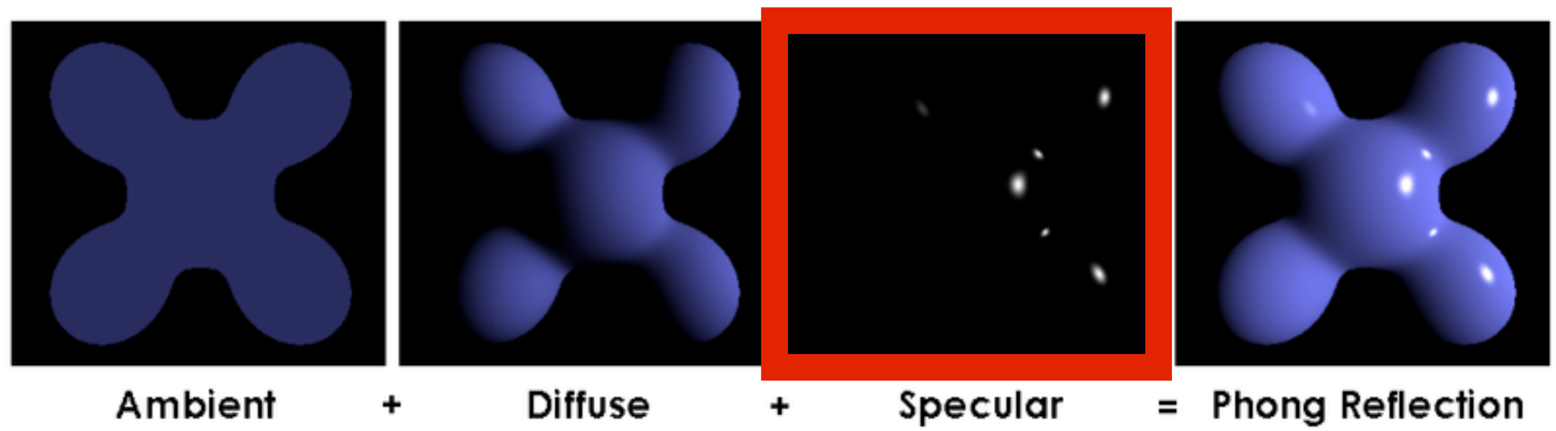


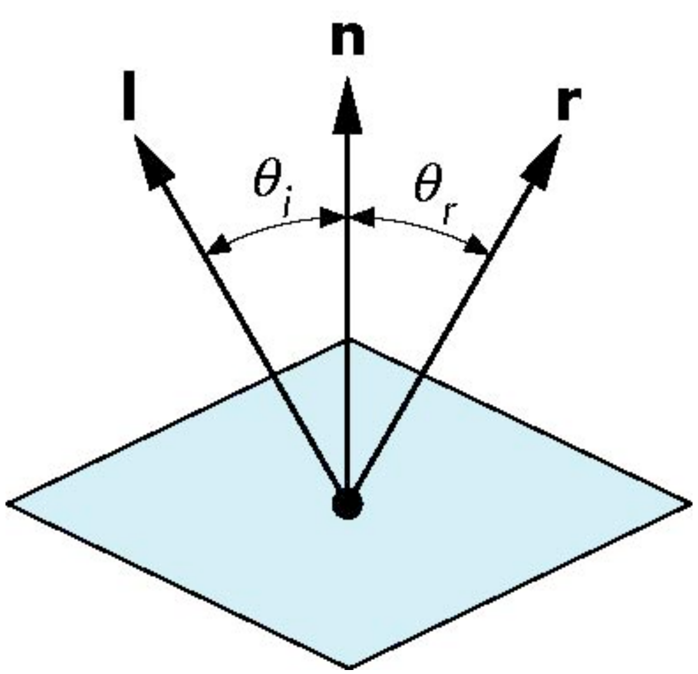
$$I_d = R_d L_d \max(0, \mathbf{l} \cdot \mathbf{n})$$

diffuse reflection coefficient



Specular reflection





Ideal reflector

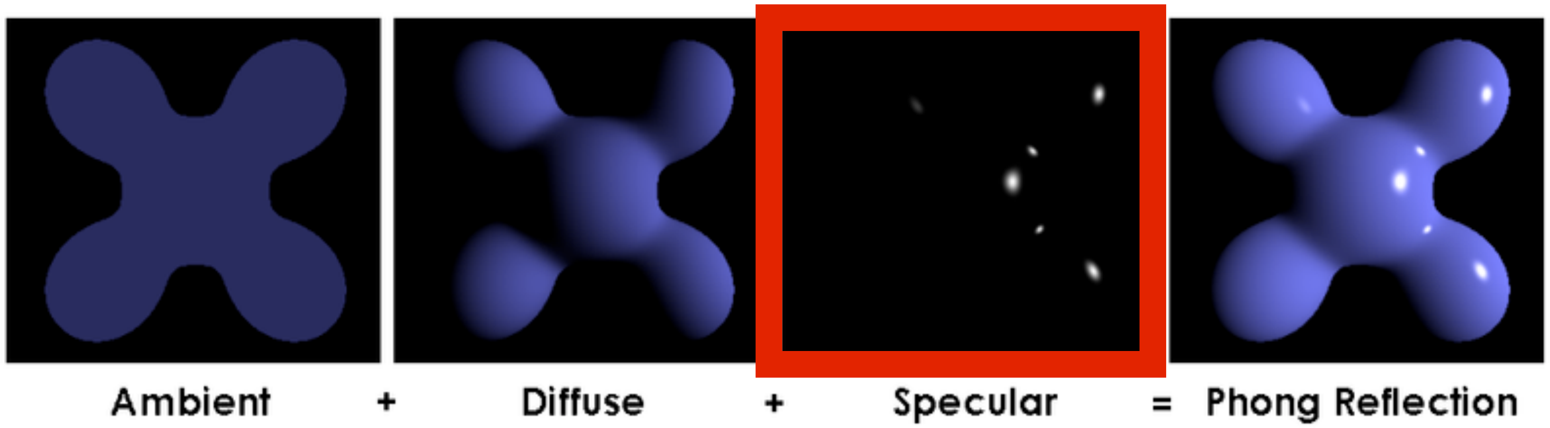
$$\theta_i = \theta_r$$

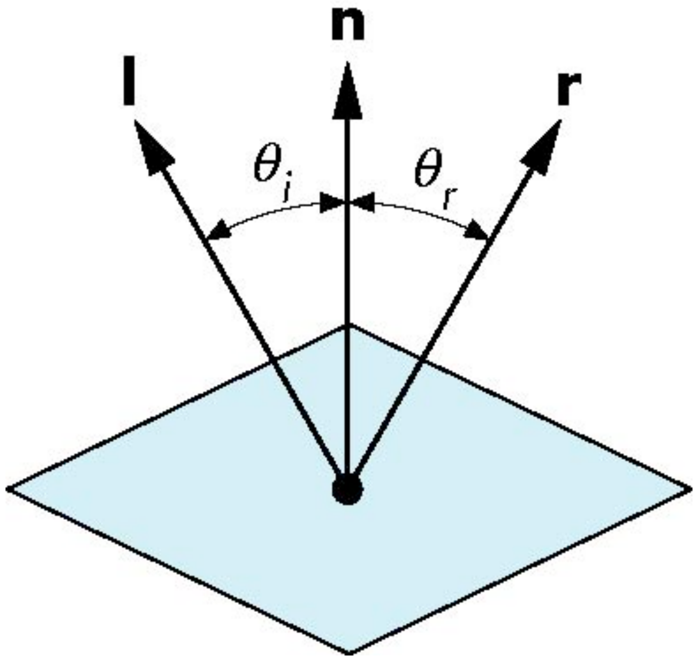
angle of incidence

angle of reflection

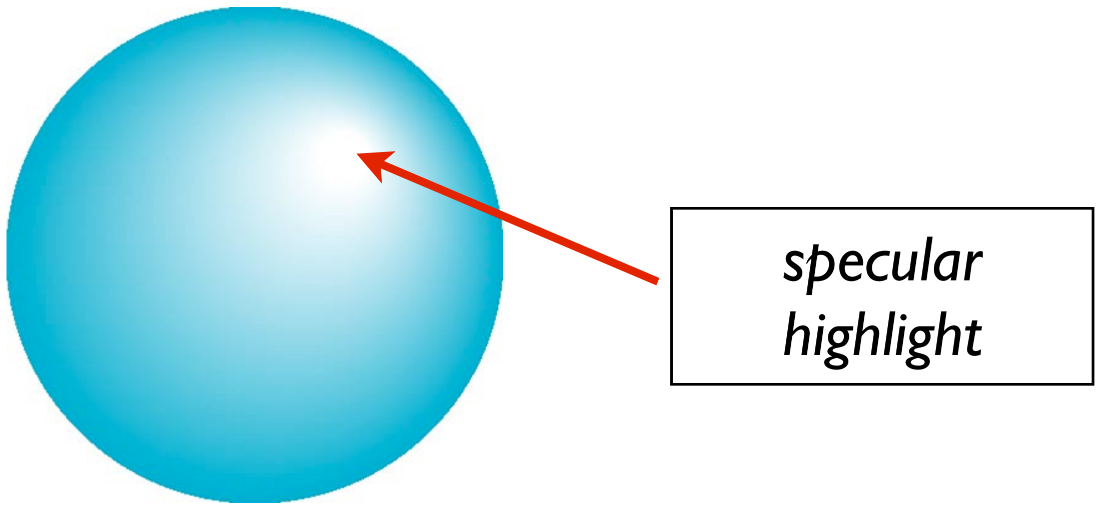
r is the mirror reflection direction

Specular reflection



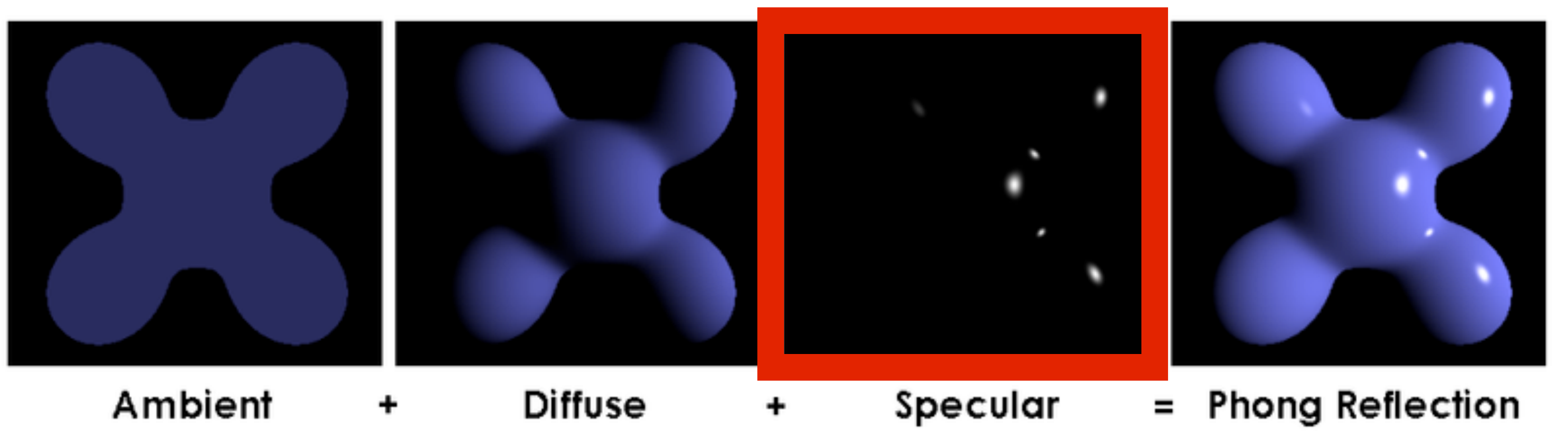


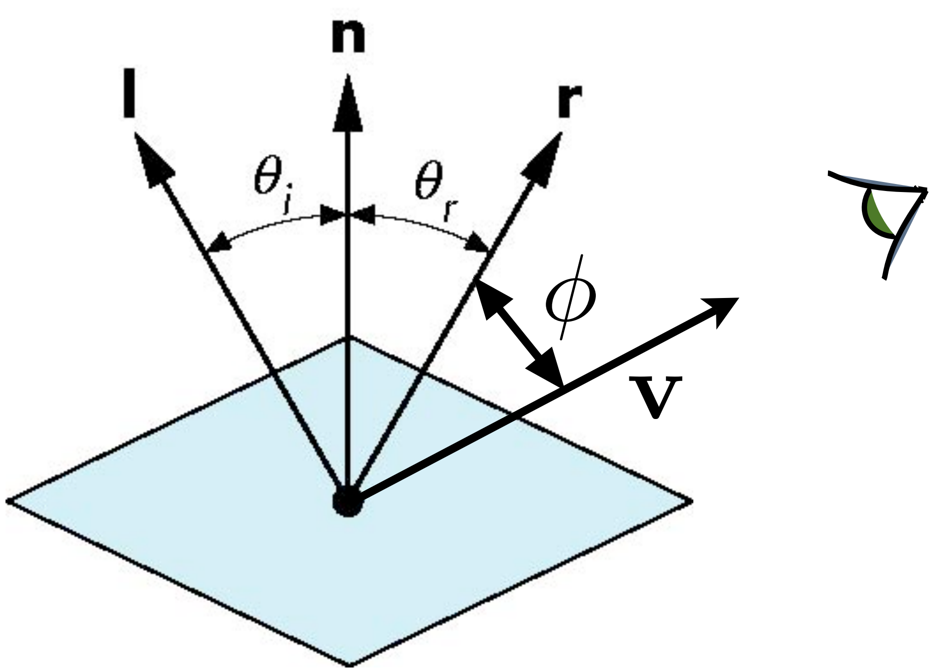
Specular surface



specular reflection is strongest in mirror reflection direction

Specular reflection





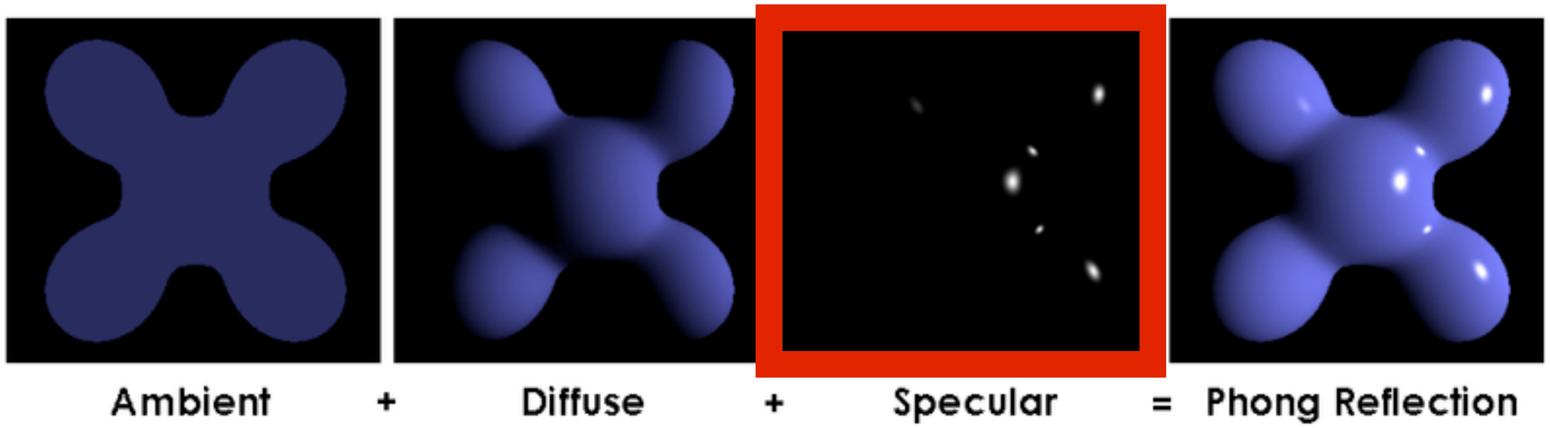
$$I_s = R_s L_s \cos^{\alpha} \phi$$

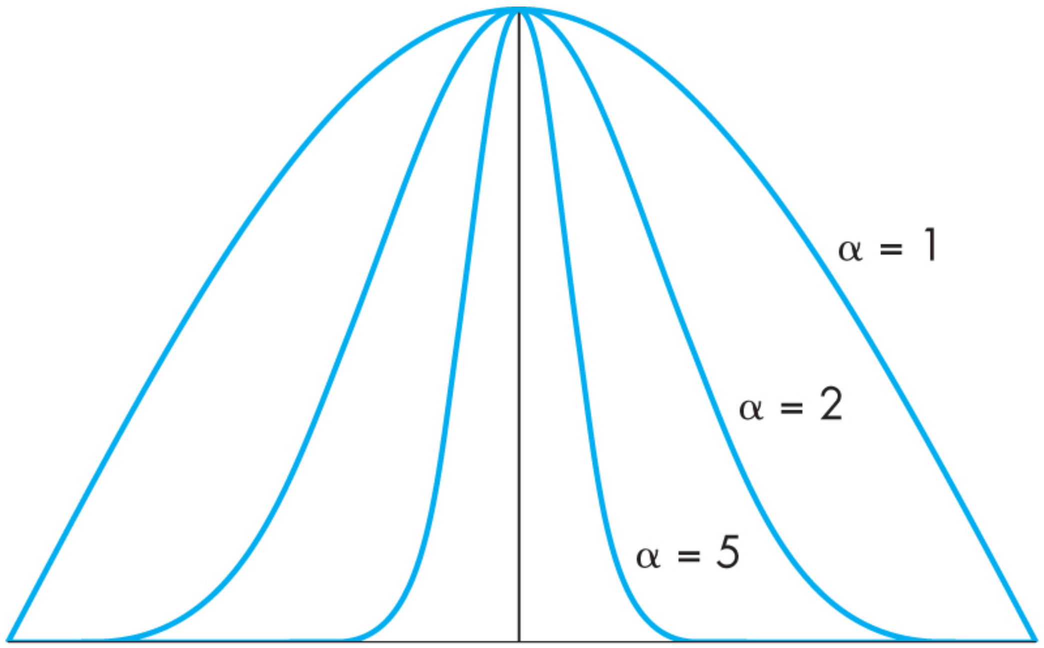
specular reflection coefficient (R_s)

Phong exponent (α)

specular reflection drops off with increasing angle ϕ

Specular reflection





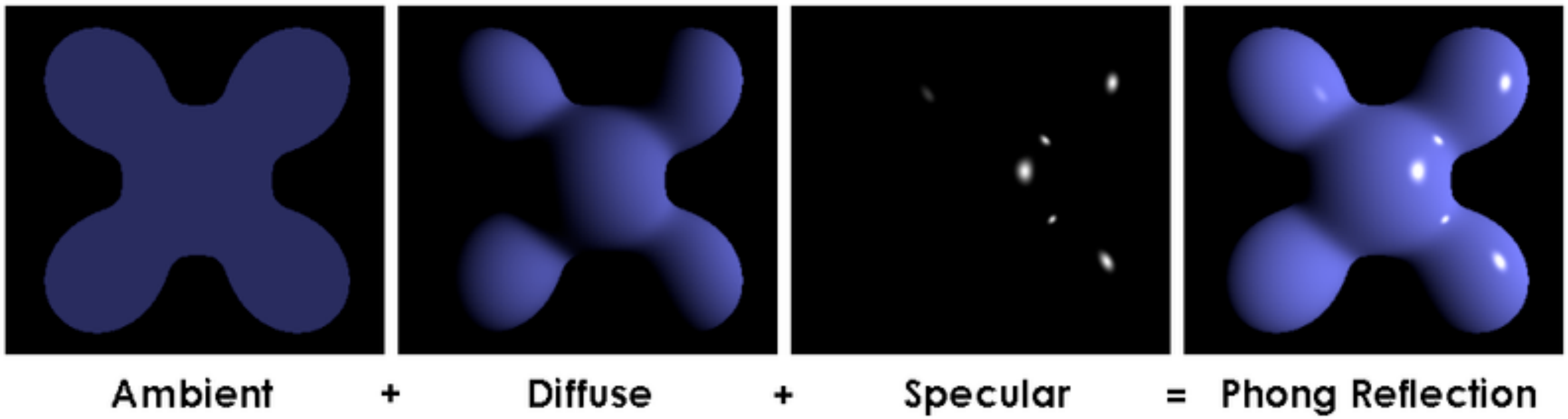
$$I_s = R_s L_s \max(0, \cos\phi)^{\alpha}$$

Phong exponent

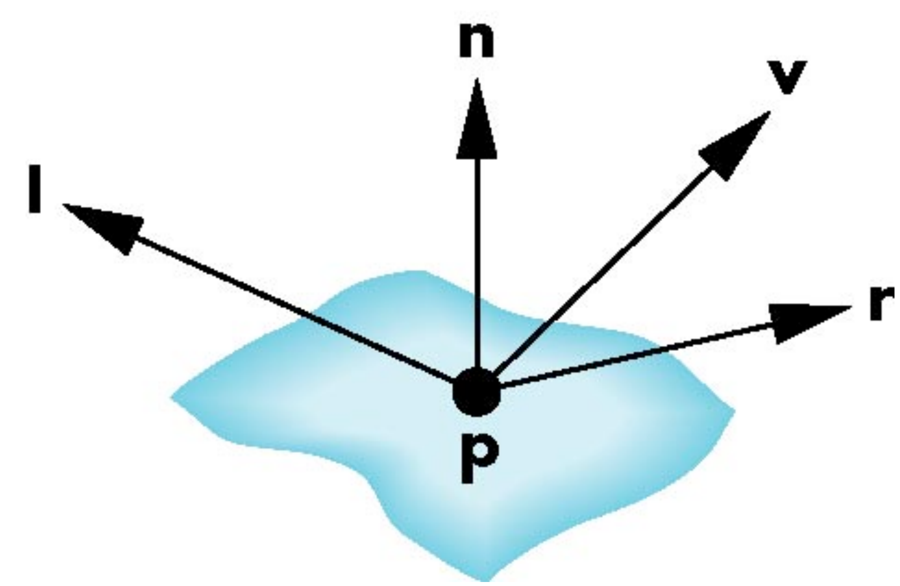
$\alpha = 5..10$ **plastic**

$\alpha = 100..200$ **metal**

Phong Reflection Model



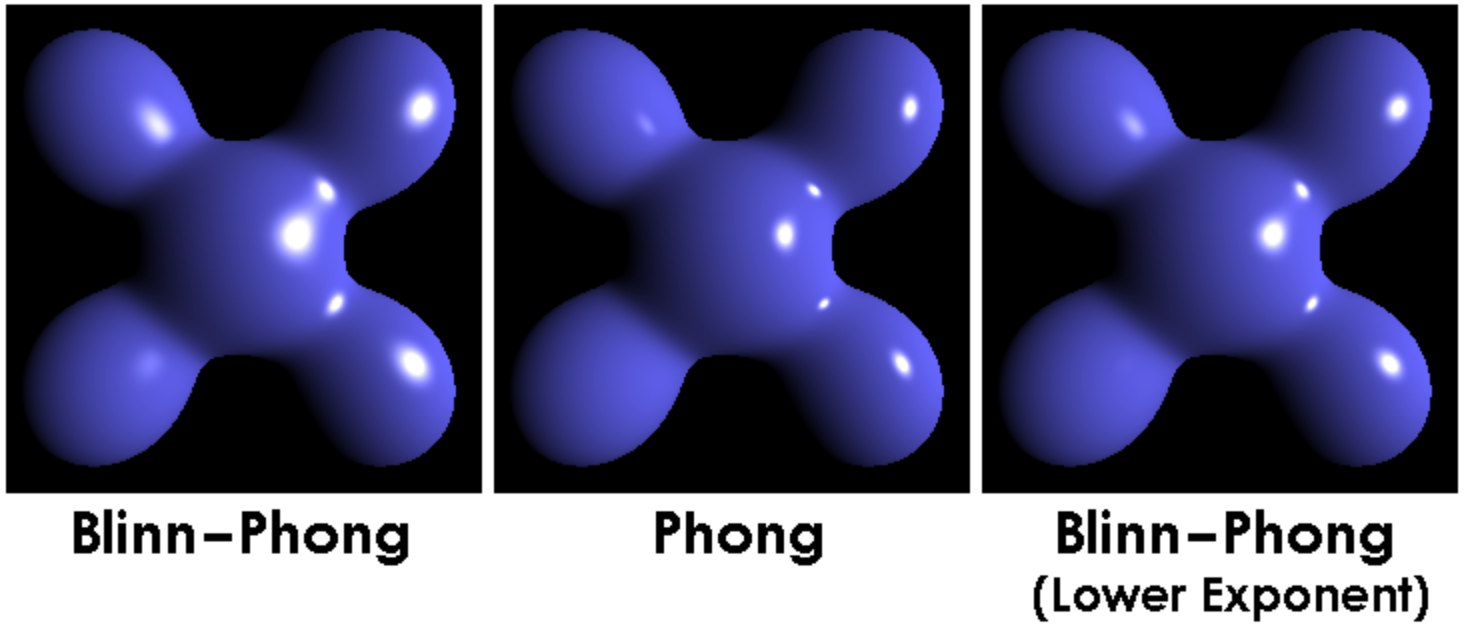
[Brad Smith, Wikimedia Commons]



$$
\begin{aligned}
I &= I_a + I_d + I_s \\
&= R_a L_a + R_d L_d \max(0, \mathbf{l} \cdot \mathbf{n}) + R_s L_s \max(0, \mathbf{v} \cdot \mathbf{r})^{\alpha}
\end{aligned}
$$

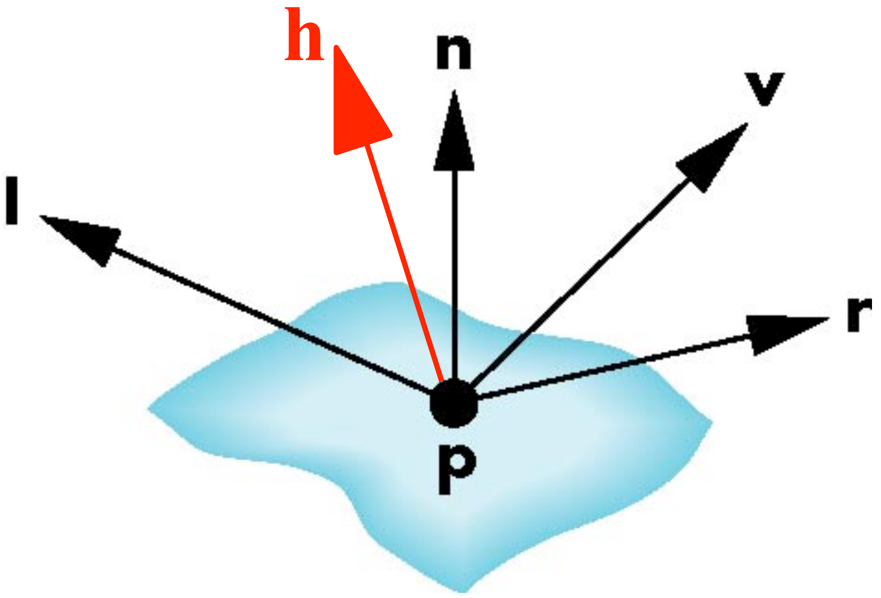
Ambient　　Diffuse　　Specular

Alternative: Blinn-Phong Model



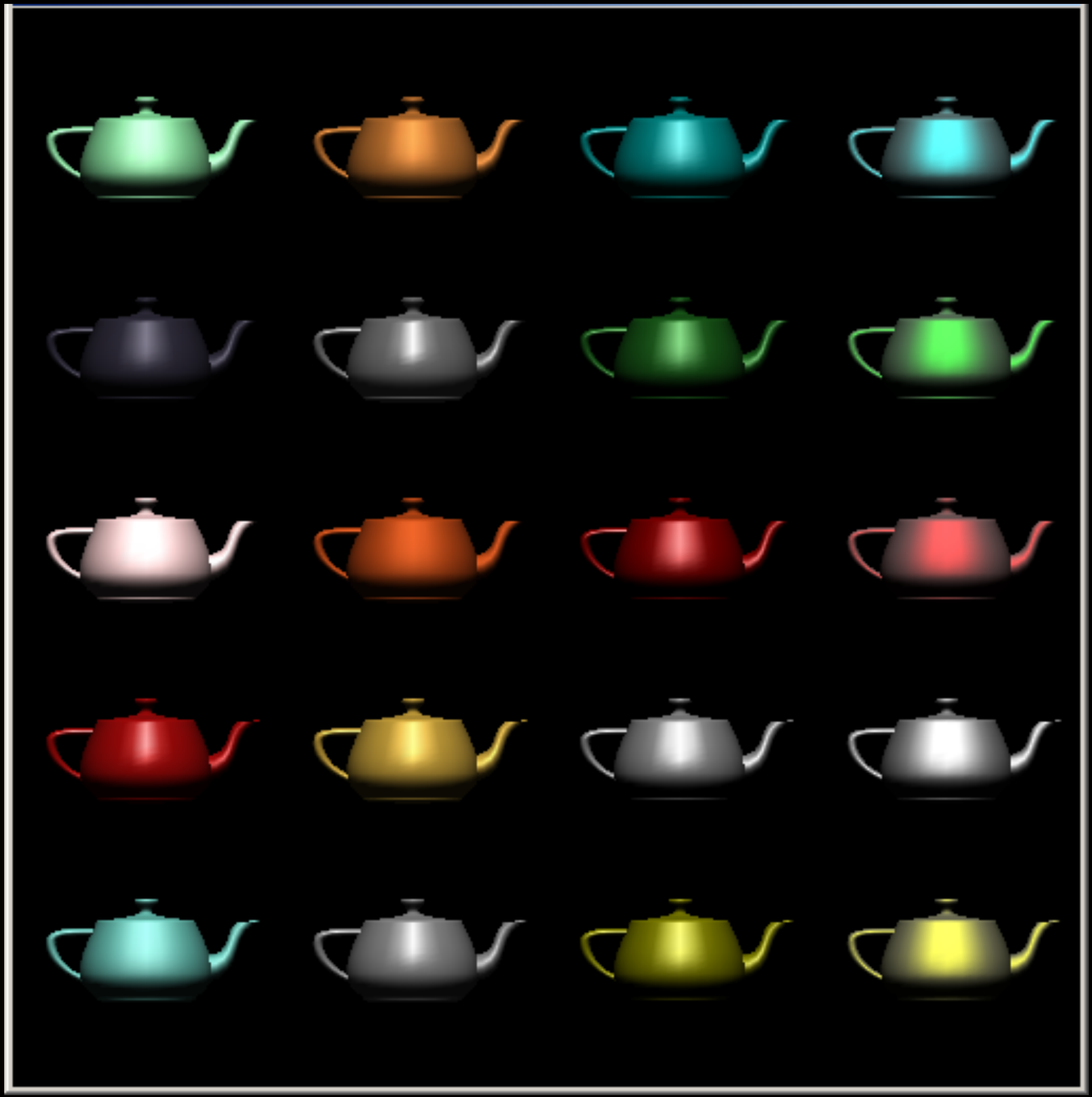
[Brad Smith, Wikimedia Commons]

halfway vector $$\mathbf{h} = \frac{\mathbf{l} + \mathbf{v}}{|\mathbf{l} + \mathbf{v}|}$$



$$\begin{aligned} I &= I_a + I_d + I_s \\ &= R_a L_a + R_d L_d \max(0, \mathbf{l} \cdot \mathbf{n}) + R_s L_s \max(0, \mathbf{h} \cdot \mathbf{n})^{\alpha} \end{aligned}$$

Ambient Diffuse Specular



α

10: eggshell
100: shiny
1000: glossy
10000: mirror-like