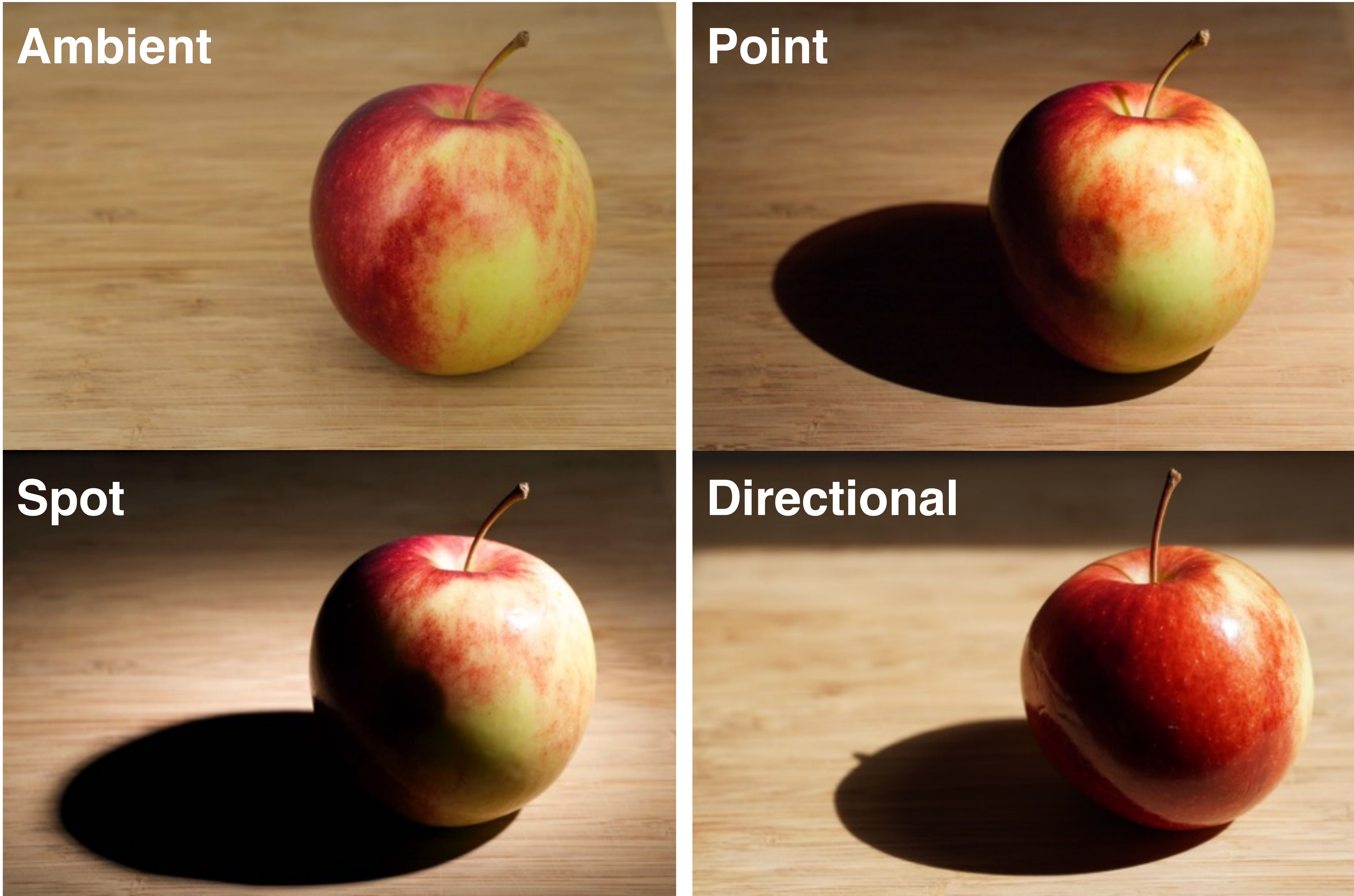
Luxo jr., Pixar Films

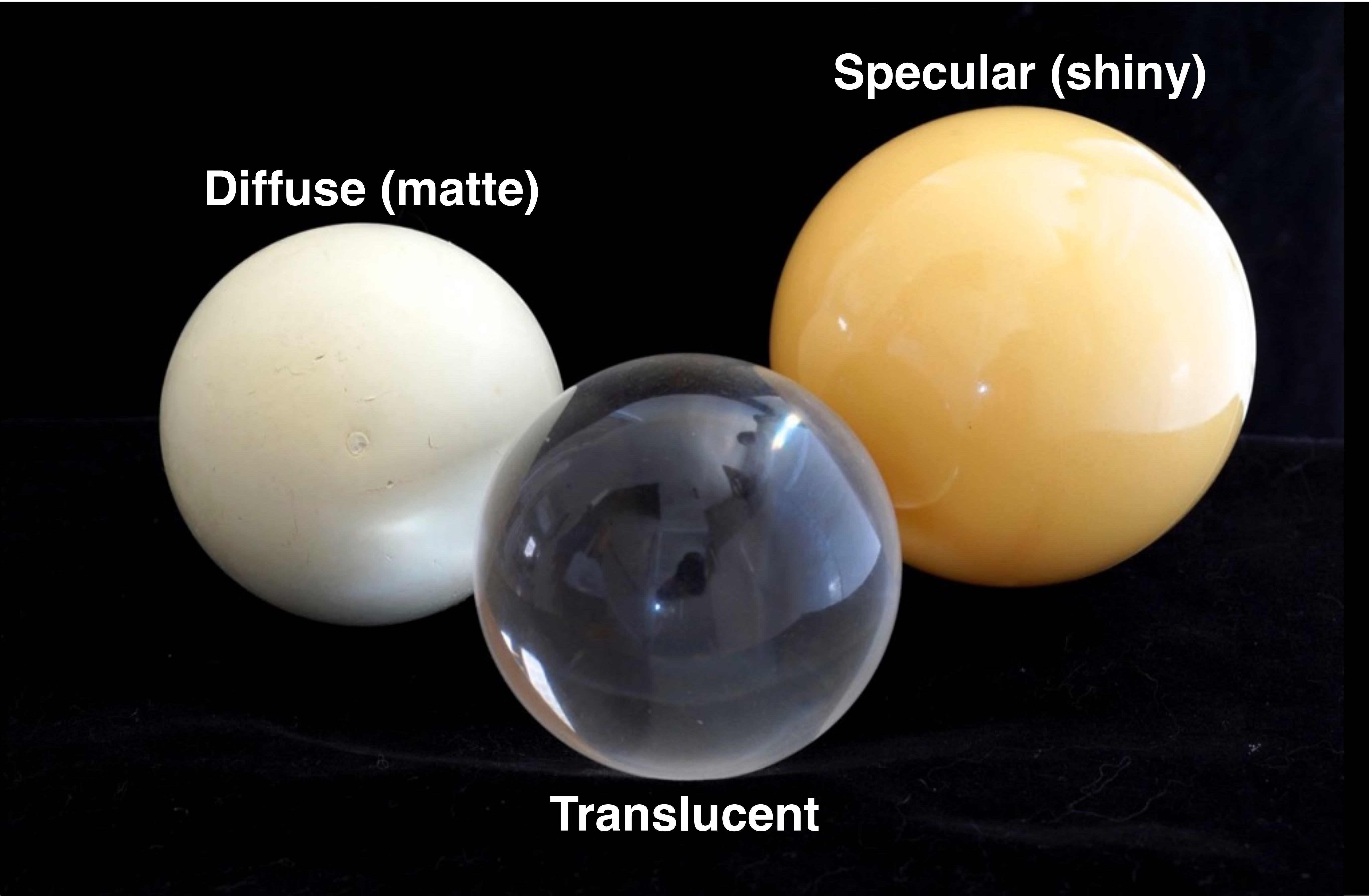
CS461: Lighting part II

C. Andrews

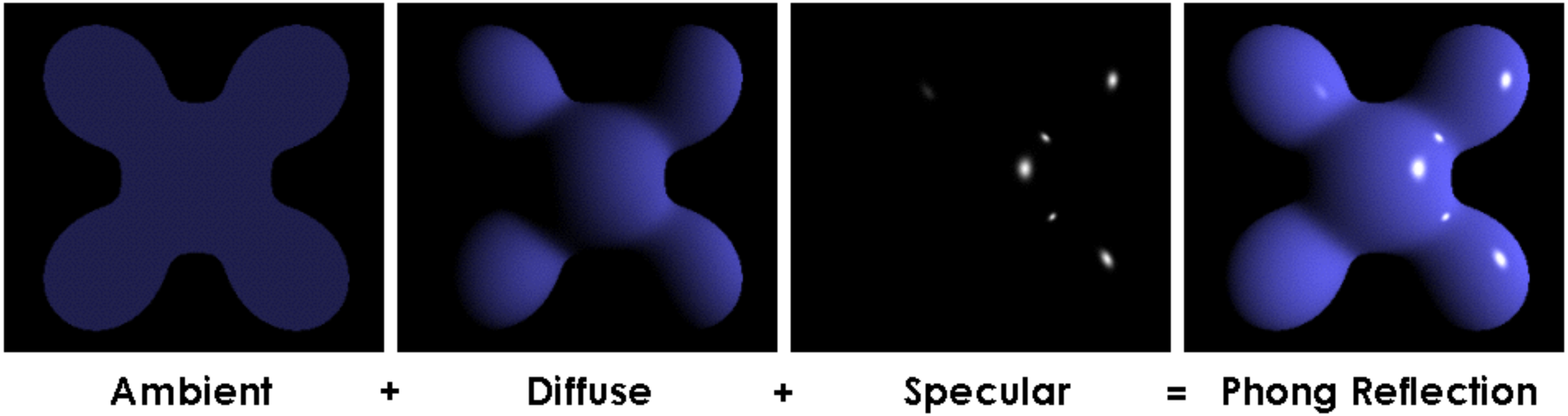
Lighting types



Material types



Phong lighting model



"Phong components version 4". Licensed under CC BY-SA 3.0 via Wikimedia Commons

Luminance

$$I = \begin{pmatrix} I_r \\ I_g \\ I_b \end{pmatrix}$$

$$I = I_a + I_d + I_s$$

Phong lighting model

Luminance

$$I = \begin{pmatrix} I_r \\ I_g \\ I_b \end{pmatrix}$$

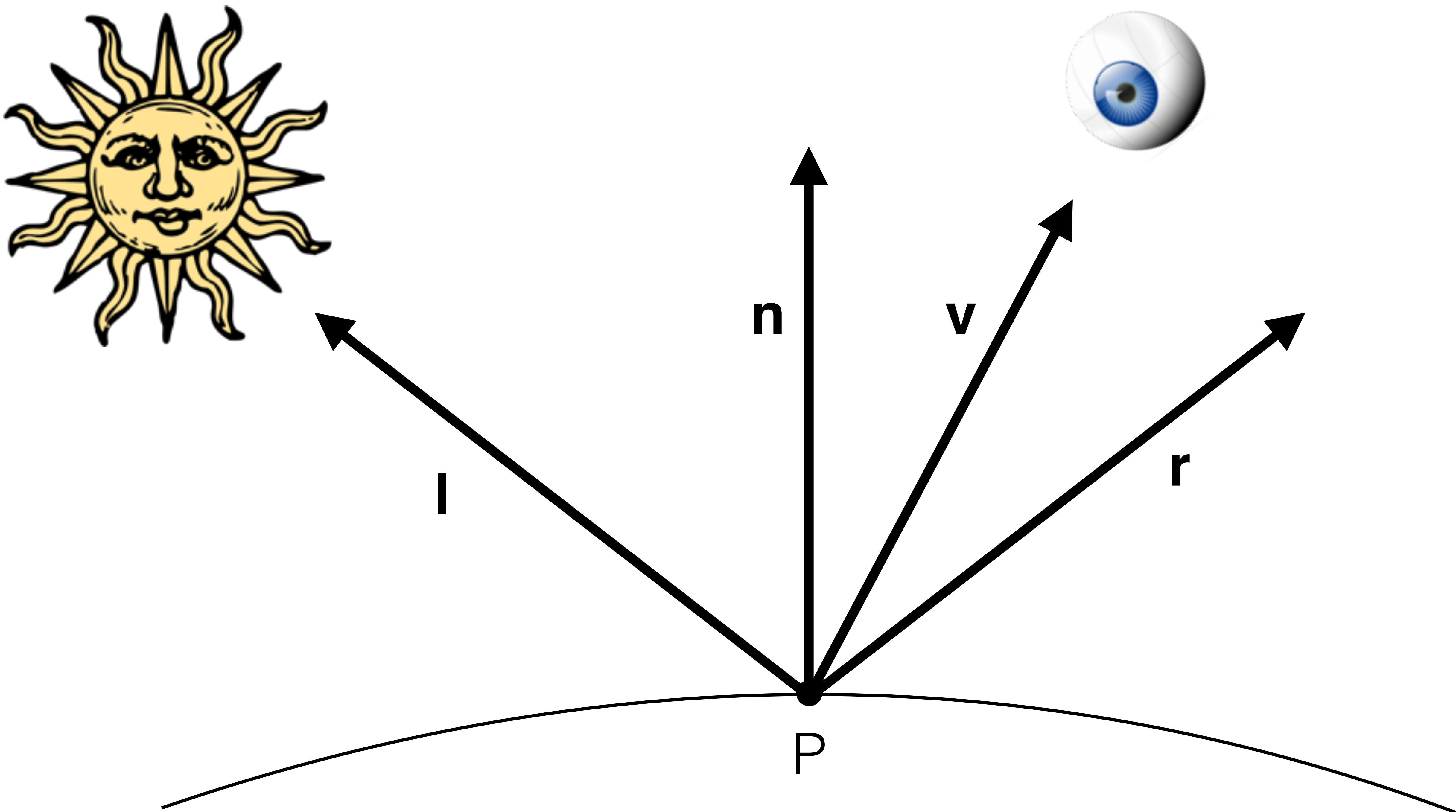
$$L_i = \begin{bmatrix} L_{ira} & L_{iga} & L_{iba} \\ L_{ird} & L_{igd} & L_{ibd} \\ L_{irs} & L_{igs} & L_{ibs} \end{bmatrix}$$

each light is broken into ambient, diffuse and specular components, with each of those broken into the three channels

$$\text{material} = \begin{bmatrix} k_{ra} & k_{ga} & k_{ba} \\ k_{rd} & k_{gd} & k_{bd} \\ k_{rs} & k_{gs} & k_{bs} \end{bmatrix}$$

each surface has material properties or **reflection coefficients** ($0 <= k <= 1$), which tells us how much of the light reflects off of the surface

Principle vectors of the Phong illumination model

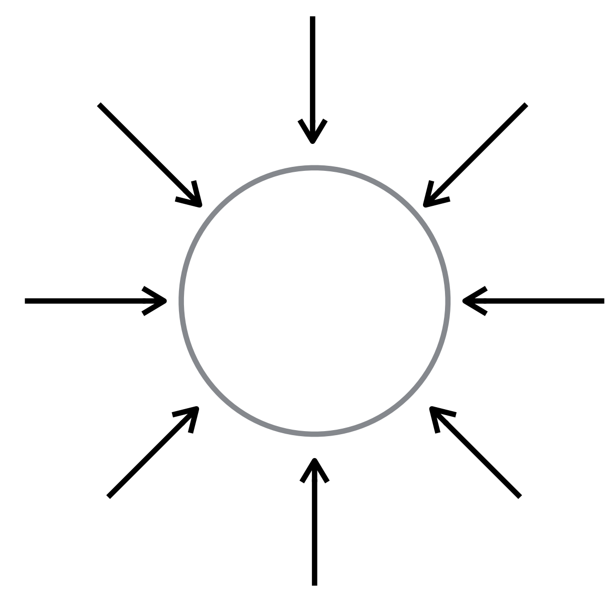


all vectors should be normalized

Ambient lighting

Use the material ambient reflection coefficient to determine the luminance of the light bouncing off of the surface

$$I_a = k_a L_a$$



L_a could be the contribution of a particular light, or a global value

Diffuse lighting

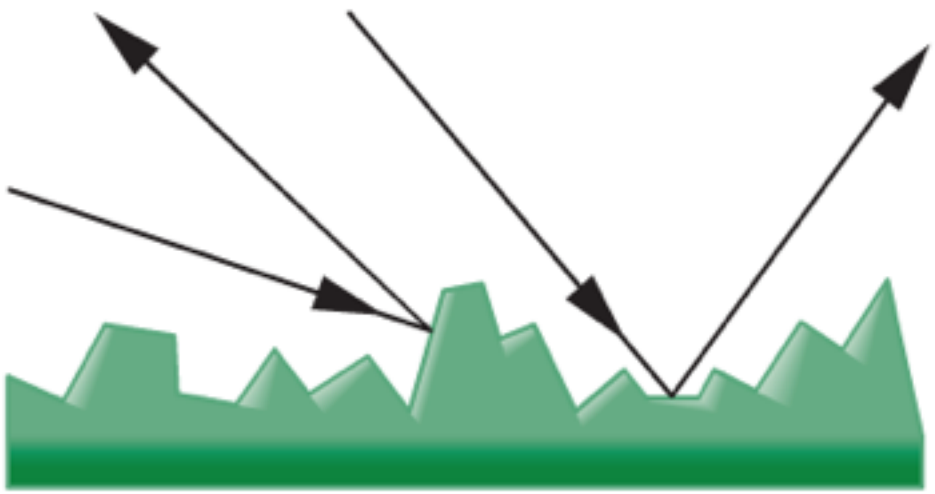


fig 6.14, Interactive Computer Graphics, 7e, Angel and Shreiner

Lambert's Law

the amount of reflected light is proportional to the cos(θ)

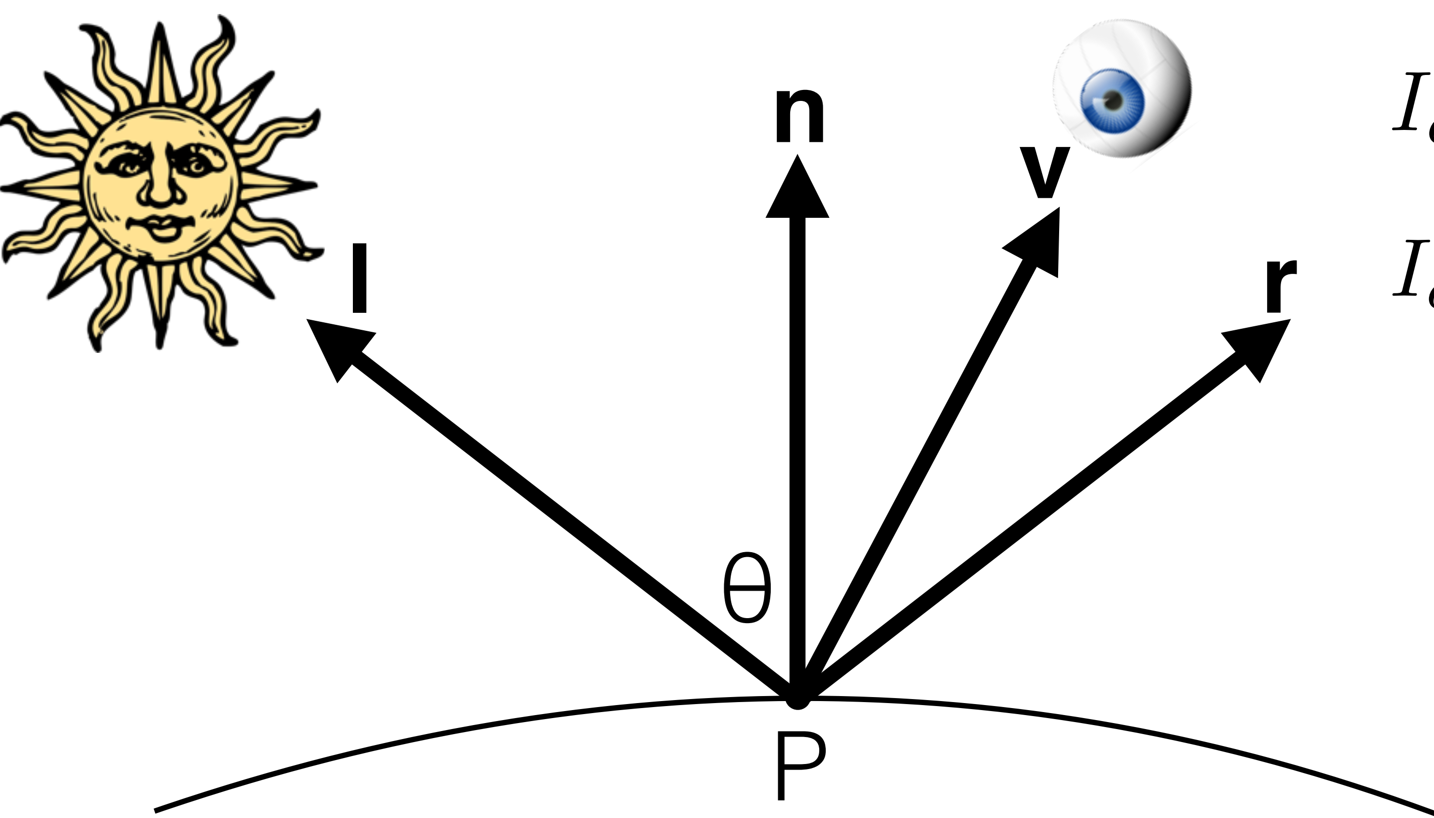
L_d — diffuse component of the light

k_d — diffuse reflection coefficient

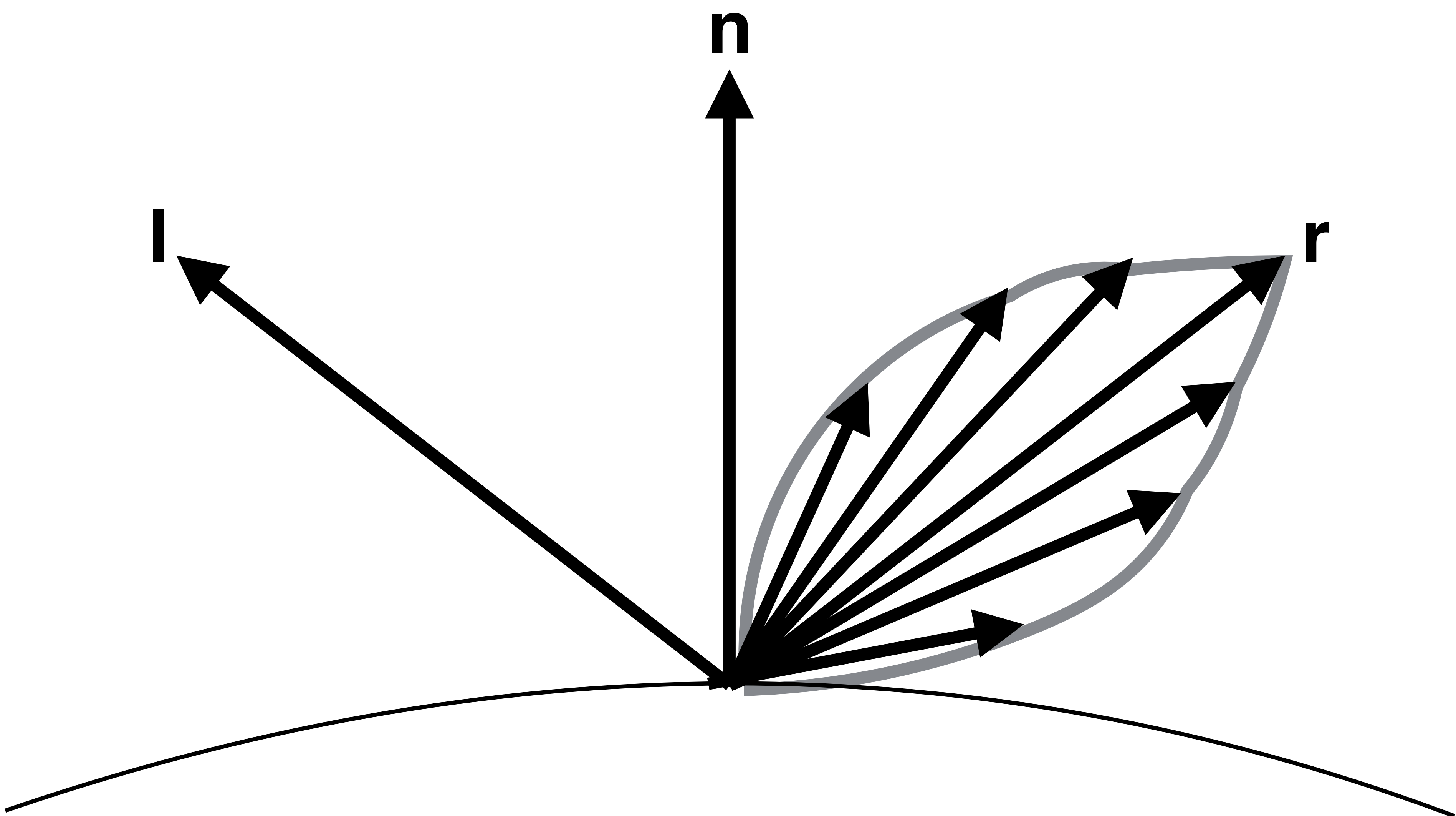
~~$I_d = k_d(\hat{l} \cdot \hat{n})L_d$~~

$$I_d = k_d \; \max(0, (\hat{l} \cdot \hat{n})L_d)$$

$$I_d = \frac{k_d}{(a + bd + cd^2)} \; \max(0, (\hat{l} \cdot \hat{n})L_d)$$

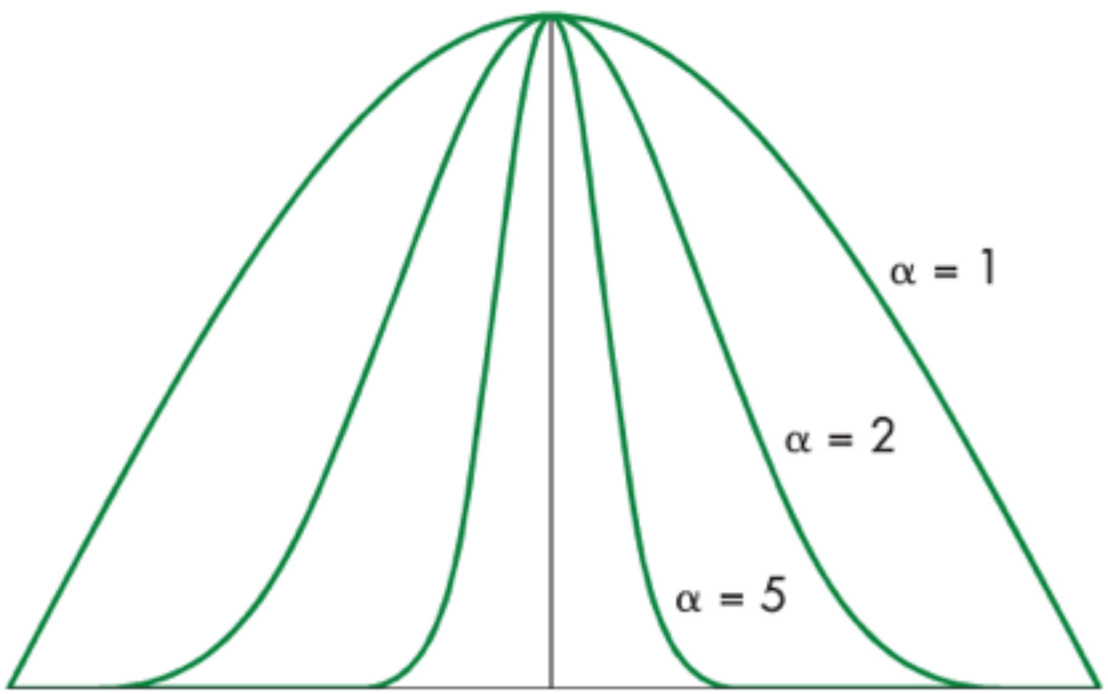


Specular lighting



as the angle between **v** and **r** grows, the amount of specular highlight falls off rapidly

Specular lighting

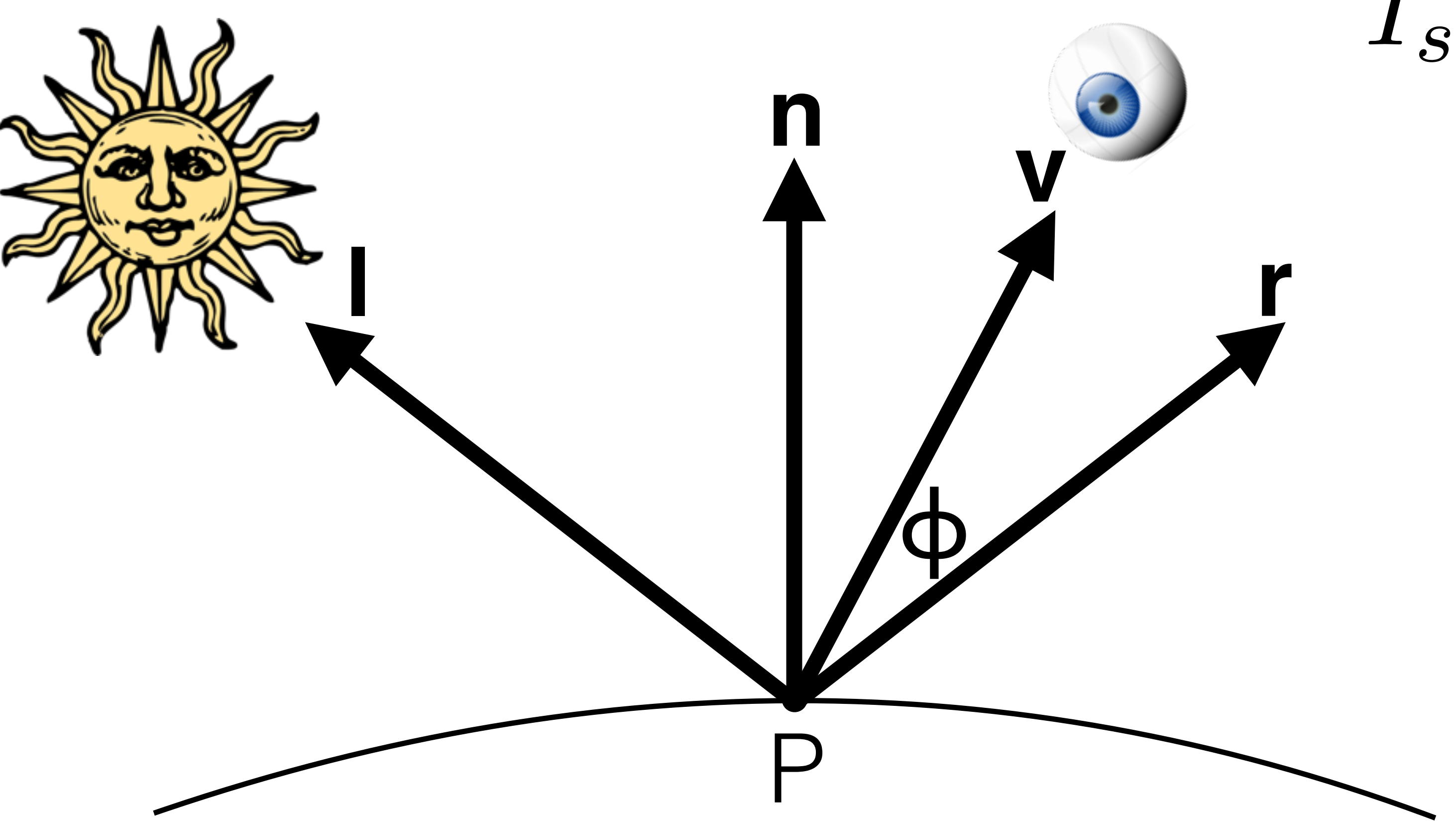


Phong's model

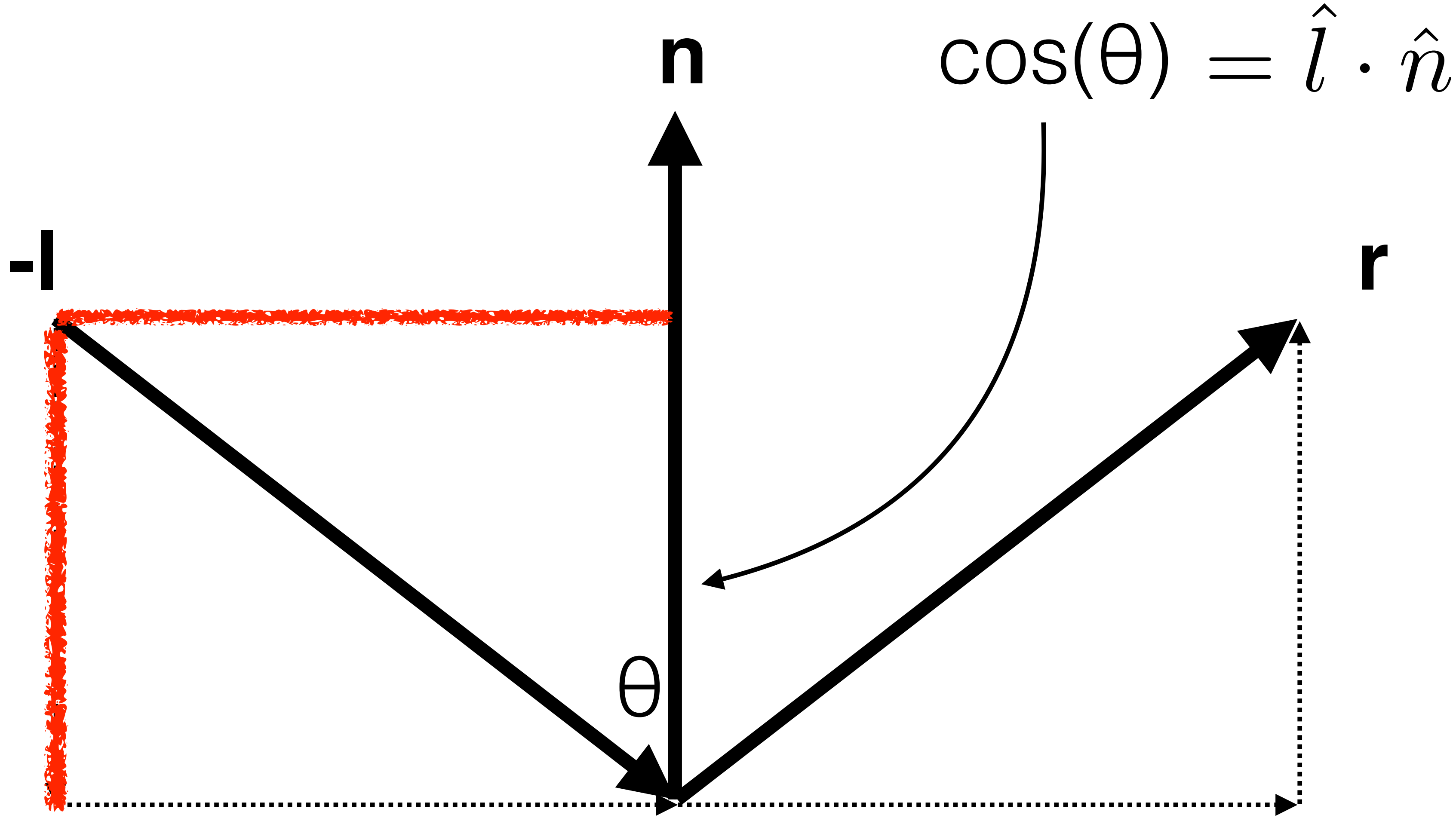
the amount of reflected light is proportional to $\cos^{\alpha}(\phi)$

α — the shininess coefficient

$$I_s = k_s \ \max(0, (\hat{r} \cdot \hat{v})^{\alpha} L_s)$$

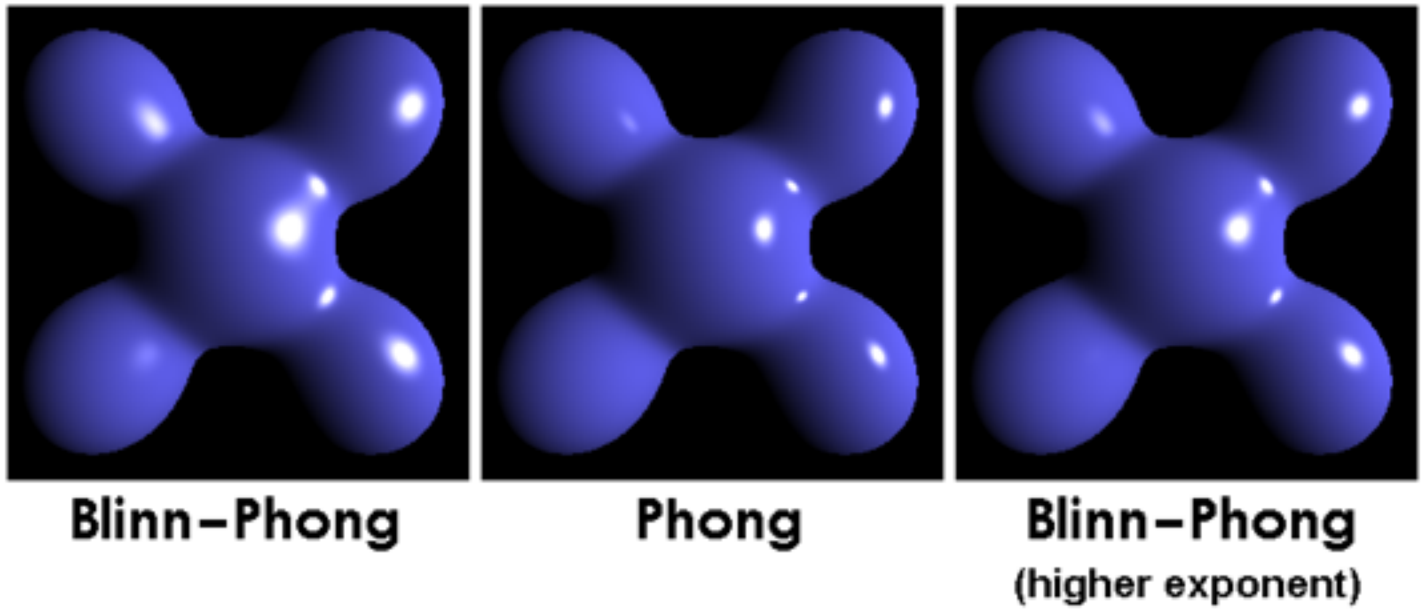


Calculating r



$$\vec{r} = -\hat{l} + 2(\hat{l} \cdot \hat{n})\hat{n}$$

Specular lighting



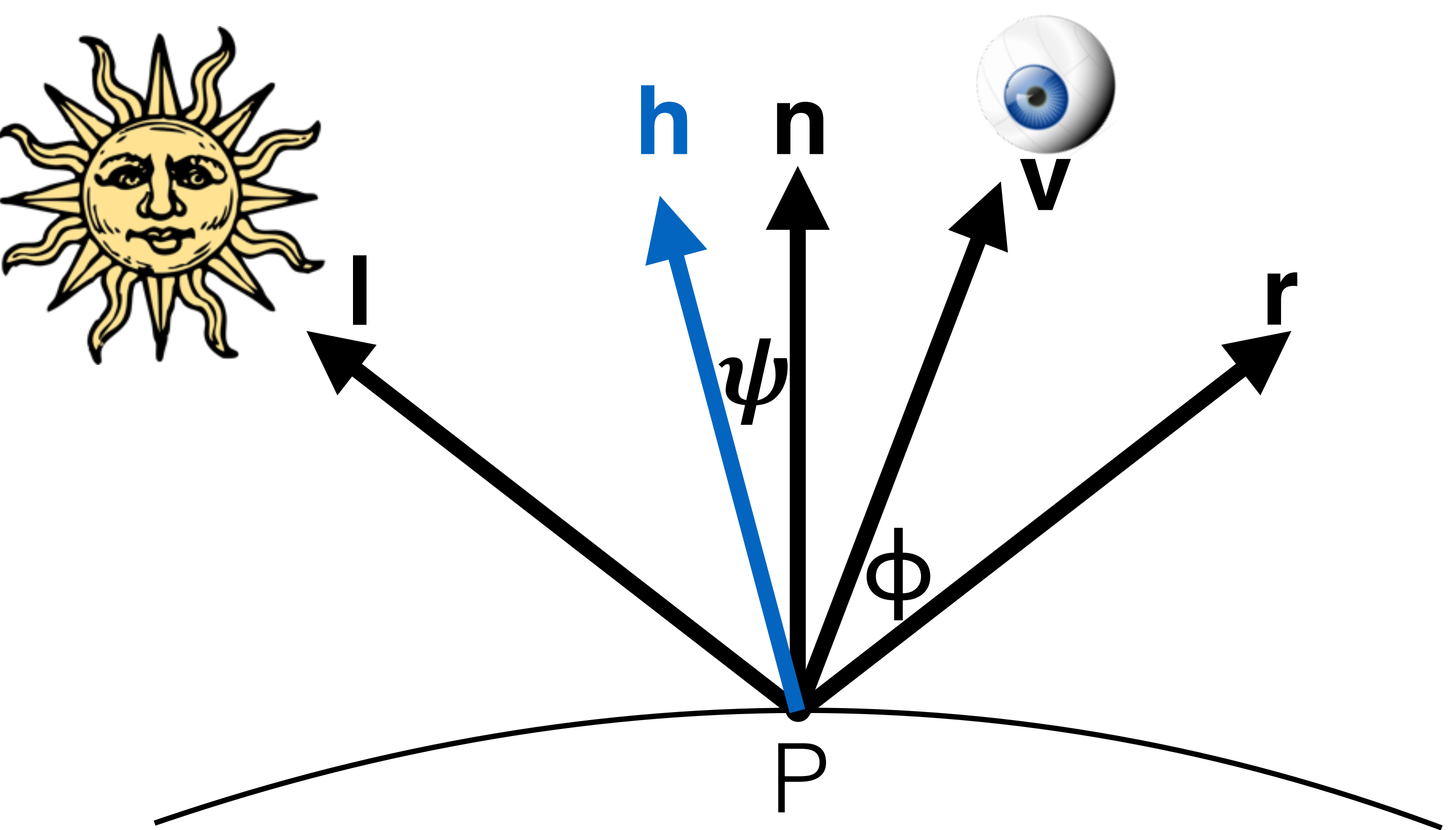
"Blinn phong comparison". Licensed under CC BY-SA 3.0 via Wikimedia Commons

Blinn-Phong's model

the amount of reflected light is proportional to $\cos^{\alpha}(\boldsymbol{\psi})$

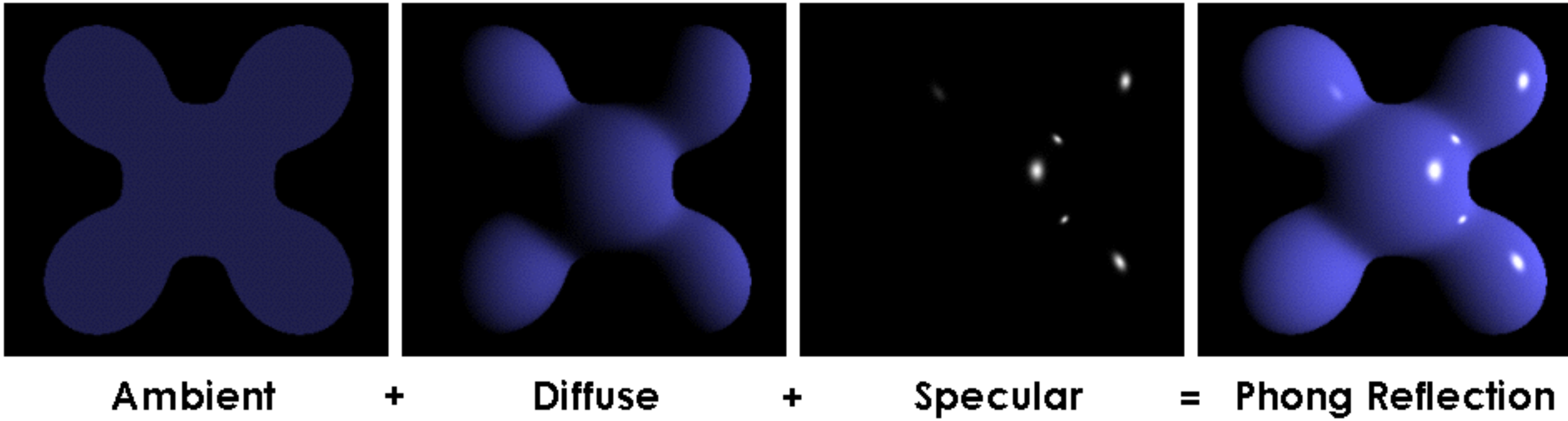
h — the halfway vector, halfway between **l** and **v**

$$\hat{h} = \frac{\hat{l} + \hat{v}}{|\hat{l} + \hat{v}|}$$



$$I_s = k_s \ \max(0, (\hat{n} \cdot \hat{h})^{\alpha} L_s)$$

Phong lighting model

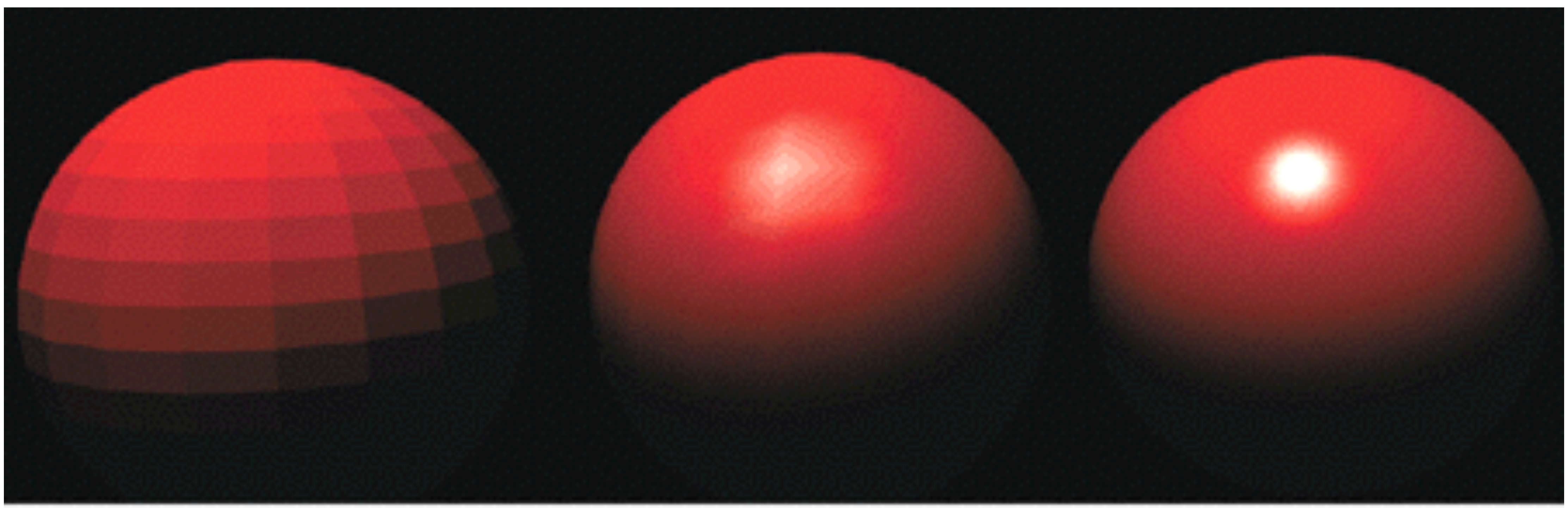


"Phong components version 4". Licensed under CC BY-SA 3.0 via Wikimedia Commons

$$I = I_a + I_d + I_s$$

$$I = k_a L_a + k_d \max(0, (\hat{l} \cdot \hat{n}) L_d) + k_s \max(0, (\hat{n} \cdot \hat{h})^{\alpha} L_s)$$

Implementing shading



Flat Gouraud Phong