



Illumination Models

Johns Hopkins Department of Computer Science
Course 600.456: Rendering Techniques, Professor: Jonathan Cohen



Things to Model

Light sources

- **What color, intensity, lines through space**

Reflection of light off surfaces

- **How much light reflected in each direction**
—**How are color and intensity changed**

Johns Hopkins Department of Computer Science
Course 600.456: Rendering Techniques, Professor: Jonathan Cohen



Real Lights

Real lights are complicated

- Sun light, iridescent bulbs, fluorescent bulbs
- Different spectra in different directions
 - probably time-varying as well, but we don't perceive much of that

Johns Hopkins Department of Computer Science
Course 600.456: Rendering Techniques, Professor: Jonathan Cohen



Simpler Light Models

- Point lights
- Directional lights
- Spot (warn) lights
- Area lights (not really so simple)

Johns Hopkins Department of Computer Science
Course 600.456: Rendering Techniques, Professor: Jonathan Cohen



Real Reflection

Again, pretty complicated

- **May be described by bidirection reflectance distribution function (BRDF)**
- **BRDF is 5D function**
 - **2D for incoming light direction**
 - **2D for outgoing light direction**
 - **1D for wavelength of light**

Johns Hopkins Department of Computer Science
Course 600.456: Rendering Techniques, Professor: Jonathan Cohen



Simpler Reflection Models

Phong illumination

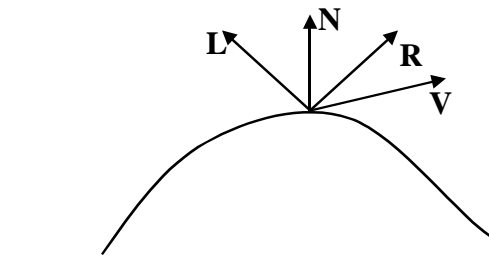
Cook and Torrance illumination

Johns Hopkins Department of Computer Science
Course 600.456: Rendering Techniques, Professor: Jonathan Cohen



Life on a Surface

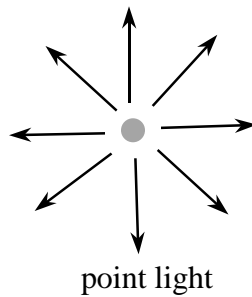
- L: direction to light**
- N: normal vector**
- R: reflection of light about normal**
- V: direction to viewer (i.e. reflection direction of interest)**



Johns Hopkins Department of Computer Science
Course 600.456: Rendering Techniques, Professor: Jonathan Cohen



Point Light



Specified by:

- position (x,y,z)
- intensity (r,g,b)

**Radiates equal intensity
in all directions**

$$\mathbf{L} = \mathbf{P}_{\text{light}} - \mathbf{P}_{\text{surface}}$$

Johns Hopkins Department of Computer Science
Course 600.456: Rendering Techniques, Professor: Jonathan Cohen



Directional Light

point light
at infinity

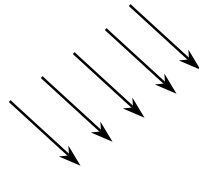
Point light at infinity

Specified by:

- direction (x,y,z)
- intensity (r,g,b)

All light rays are parallel

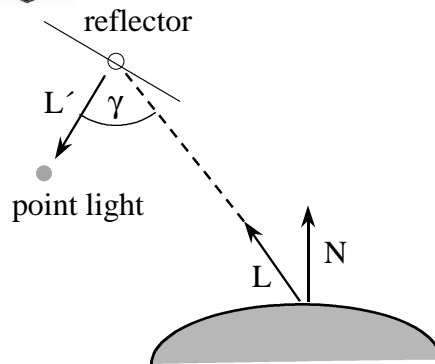
L = -direction



Johns Hopkins Department of Computer Science
Course 600.456: Rendering Techniques, Professor: Jonathan Cohen



Spot (Warn) Light



Specular reflection of point light source

Specified by:

- position of reflector
- position of point light (or direction to point light)
- intensity of point light
- falloff exponent

$$I_{\text{warn}} = I_{\text{point}} \cos^p \gamma = I_{\text{point}} (V \cdot R)^p = I_{\text{point}} (-L \cdot L')^p$$

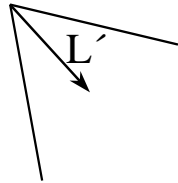
Johns Hopkins Department of Computer Science
Course 600.456: Rendering Techniques, Professor: Jonathan Cohen



Warn Light (cont.)

Also possible to truncate region of effect

- flaps
- cone (used in OpenGL spotlight)



Johns Hopkins Department of Computer Science
Course 600.456: Rendering Techniques, Professor: Jonathan Cohen



Warn Light Profile and Examples

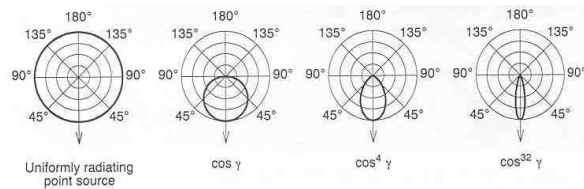


Fig. 16.14 Intensity distributions for uniformly radiating point source and Warn light source with different values of p .

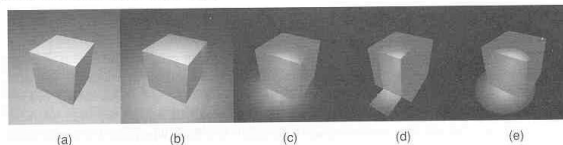


Fig. 16.15 Cube and plane illuminated using Warn lighting controls. (a) Uniformly radiating point source (or $p = 0$). (b) $p = 4$. (c) $p = 32$. (d) Flaps. (e) Cone with $\delta = 18^\circ$. (By David Kurlander, Columbia University.)

From Foley, vanDam, Feiner, and Hughes, Computer Graphics: Principles and Practice, 2nd edition, page 732, 733

Johns Hopkins Department of Computer Science
Course 600.456: Rendering Techniques, Professor: Jonathan Cohen



Phong Illumination

Empirically divides reflection into 3 components

- Ambient
- Diffuse (Lambertian)
- Specular

Johns Hopkins Department of Computer Science
Course 600.456: Rendering Techniques, Professor: Jonathan Cohen



Ambient Light

Independent of location of viewer, location of light, and curvature of surface

$$I = I_a k_a$$

- I_a is intensity of ambient light
- k_a is ambient coefficient of surface

Note: this is a total hack, of course

Johns Hopkins Department of Computer Science
Course 600.456: Rendering Techniques, Professor: Jonathan Cohen



Diffuse Reflection

Component of reflection due to even scattering of light by uniform, rough surfaces

Depends on direction of light and surface normal

$$I_d = I_p(L \cdot N)$$

- I_p is intensity of point light

Johns Hopkins Department of Computer Science
Course 600.456: Rendering Techniques, Professor: Jonathan Cohen



Important Note

When we write:

$$(N \cdot L)$$

we often really mean:

$$\max(N \cdot L, 0)$$

- **The latter computes 1-sided lighting**
- **For 2-sided lighting, use:**

$$\text{abs}(N \cdot L)$$

Johns Hopkins Department of Computer Science
Course 600.456: Rendering Techniques, Professor: Jonathan Cohen



Diffuse Reflection Examples



Fig. 16.3 Spheres shaded using a diffuse-reflection model (Eq. 16.4). For all spheres, $I_a = I_p = 1.0$. From left to right, $k_d = 0.4, 0.55, 0.7, 0.85, 1.0$. (By David Kurlander, Columbia University.)

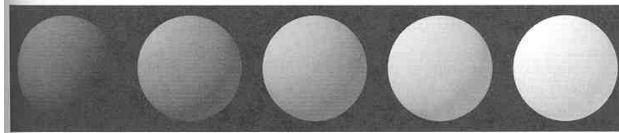


Fig. 16.4 Spheres shaded using ambient and diffuse reflection (Eq. 16.5). For all spheres, $I_a = I_p = 1.0$, $k_d = 0.4$. From left to right, $k_s = 0.0, 0.15, 0.30, 0.45, 0.60$. (By David Kurlander, Columbia University.)

From Foley, vanDam, Feiner, and Hughes, Computer Graphics: Principles and Practice, 2nd edition, page 725

Johns Hopkins Department of Computer Science
Course 600.456: Rendering Techniques, Professor: Jonathan Cohen



Specular Reflection

Component of reflection due to mirror-like reflection off shiny surface

Depends on perfect reflection direction, viewer direction, and surface normal

$$I_s = I_p (\mathbf{R} \cdot \mathbf{V})^n$$

- **n is specular exponent, determining falloff rate**

Johns Hopkins Department of Computer Science
Course 600.456: Rendering Techniques, Professor: Jonathan Cohen



Phong Illumination Example

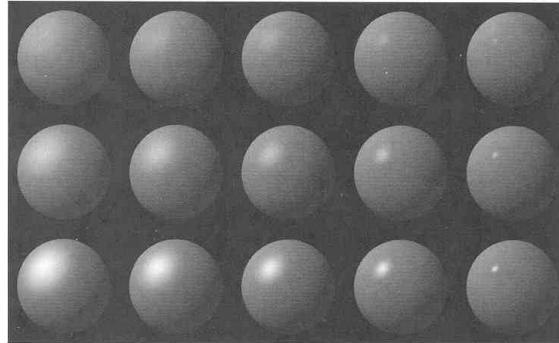


Fig. 16.10 Spheres shaded using Phong's illumination model (Eq. 16.14) and different values of k_s and n . For all spheres, $i_s = i_p = 1.0$, $k_d = 0.1$, $k_r = 0.45$. From left to right, $n = 3.0, 5.0, 10.0, 27.0, 200.0$. From top to bottom, $k_s = 0.1, 0.25, 0.5$. (By David Kurlander, Columbia University.)

From Foley, vanDam, Feiner, and Hughes, Computer Graphics: Principles and Practice, 2nd edition, page 730

Johns Hopkins Department of Computer Science
Course 600.456: Rendering Techniques, Professor: Jonathan Cohen



Illumination with Color

Surface reflection coefficients and light intensity may vary by wavelength

For RGB color

- Light intensity specified for R, G, and B
- Surface reflection coefficients also for R, G, B
- Compute reflected color for R, G, and B

Johns Hopkins Department of Computer Science
Course 600.456: Rendering Techniques, Professor: Jonathan Cohen



Cook and Torrance Illumination

Replace specular component with more physically accurate model

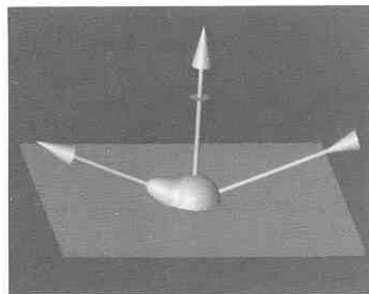
$$\rho_s = F_\lambda DG/\pi[(N.V)(N.L)]$$

- F_λ is Fresnel term, which accounts for change of highlight color with respect to angle of incidence
- D is microfacet distribution term, for more accurate measurement specular reflection off tiny microfacets
- G is geometry term, which models self-shadowing effects

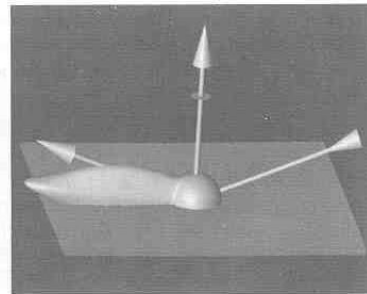
Johns Hopkins Department of Computer Science
Course 600.456: Rendering Techniques, Professor: Jonathan Cohen



Phong vs. Cook/Torrance Example



(a) Phong model



(b) Torrance-Sparrow model

Fig. 16.44 Comparison of Phong and Torrance–Sparrow illumination models for light at a 70° angle of incidence. (By J. Blinn [BLIN77a], courtesy of the University of Utah.)

From Foley, vanDam, van Dam, and Hughes, Computer Graphics: Principles and Practice, 2nd edition, page 768

Johns Hopkins Department of Computer Science
Course 600.456: Rendering Techniques, Professor: Jonathan Cohen