10 – surface shading

Illumination and Shading

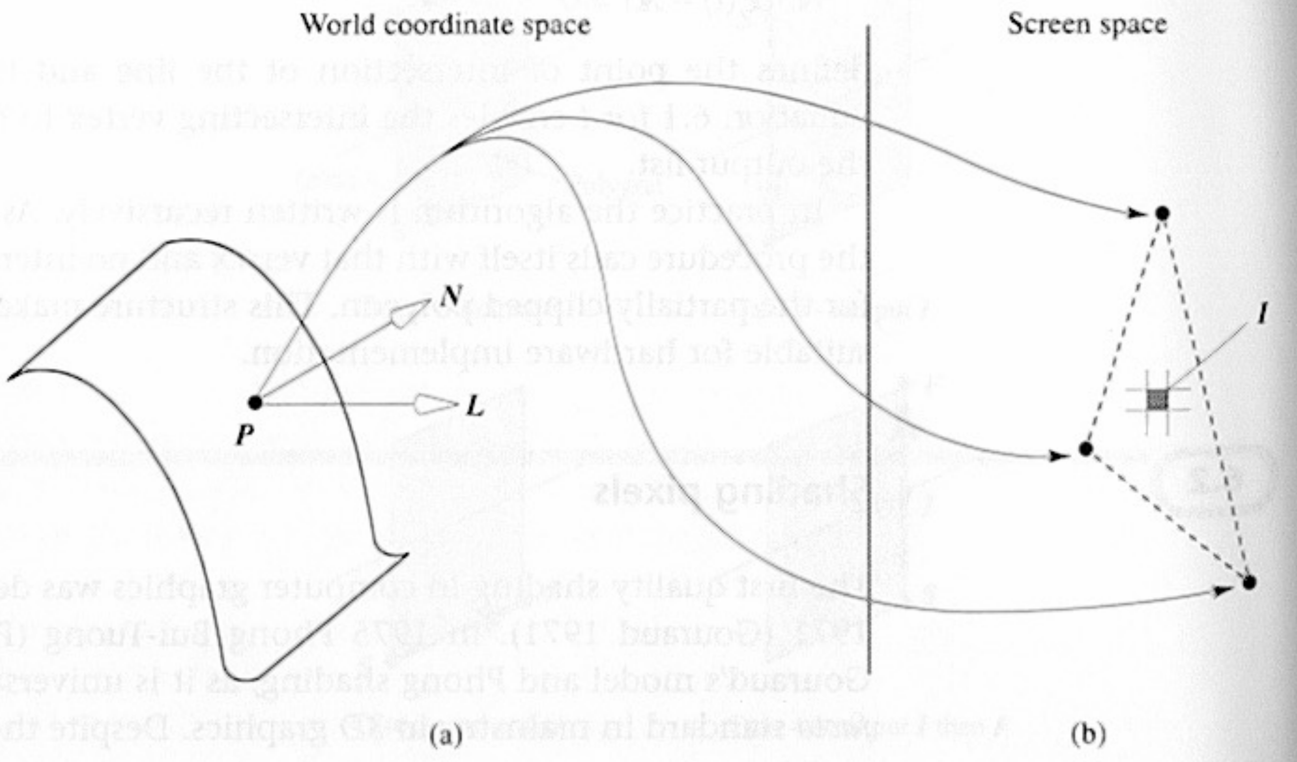


Figure 6.5
Illustrating the difference between local reflection models and shading algorithms. (a) Local reflection models calculate light intensity at any point ***P*** on the surface of an object. (b) Shading algorithms interpolate pixel values from calculated light intensities at the polygon vertices.

Surface Normals

Illumination and Shading

- Illumination Models
  - Ambient
  - Diffuse
  - Attenuation
  - Specular Reflection
- Interpolated Shading Models
  - Flat, Gouraud, Phong
  - Problems

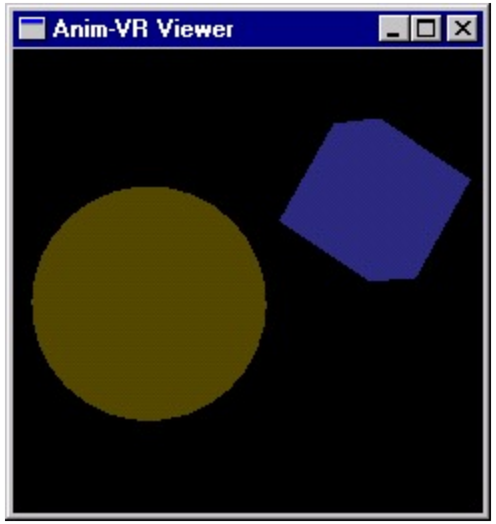
Surface Shading

Illumination Models: Ambient Light

- Simple illumination model

$$I = I_a k_a$$

- I_a = ambient light intensity
  (I_{aR}, I_{aG}, I_{aB})
- k_a = ambient-reflection coefficient
  (k_{aR}, k_{aG}, k_{aB})
- Uniform across surface



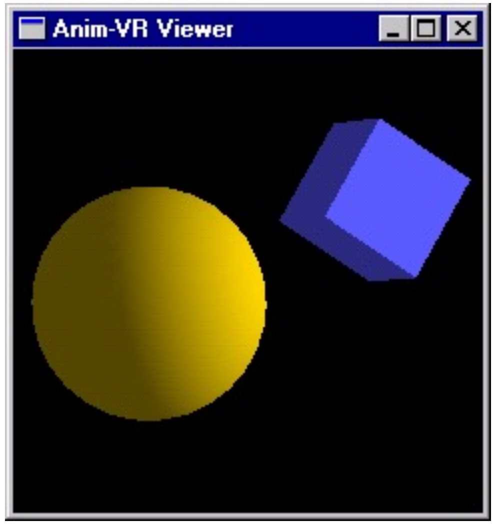
Diffuse Light

- Account for light position
  - Ignore viewer position
- Proportional to $\cos\Theta$ between *N* and *L*

$$I = I_p k_d \cos\Theta$$
$$= I_p k_d (N \cdot L)$$

- Model:

$$I = I_a k_a + I_p k_d (N \cdot L)$$

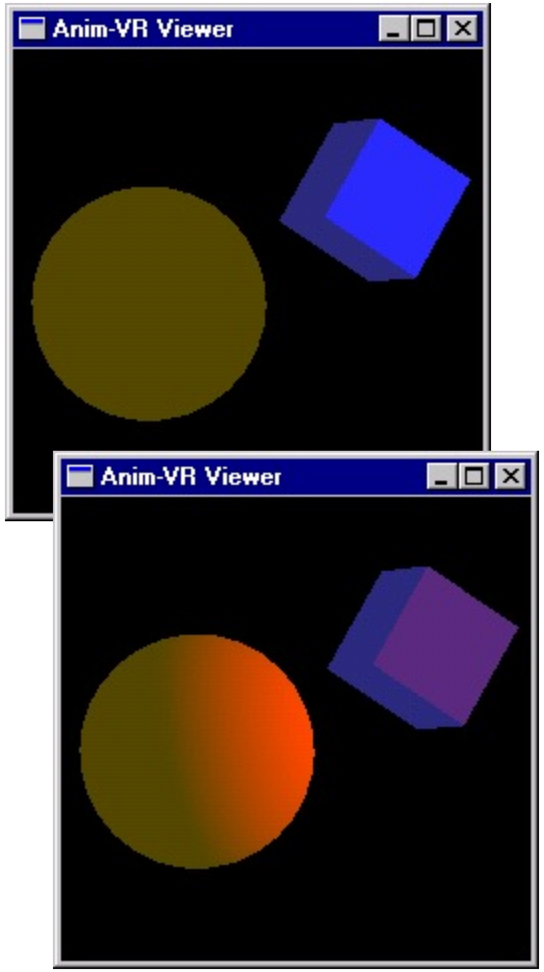


Again, Colored Lights
(slightly different, but equivalent, to book)

- O_d: diffuse color

  $O_d = (O_{dR}, O_{dG}, O_{dB})$
- Compute for each component
- i.e. red component is

  $I_R = I_{aR}k_{aR}O_{dR} + f_{att}I_{pR}k_{dR}O_{dR}\,(N \cdot L)$
- Note: use O_d for ambient and diffuse



Light Intensity Values

- I_a, I_d
  - Represent intensity
  - Have R,G,B components
  - Do not need to fall in the 0..1 range!
    - Often need I_d>1
    - Final computed $I \leq 1$

Attenuation: Distance

- f_{att} models distance from light

  $I = I_a k_a + f_{att} I_p k_d \, (N \cdot L)$

- Realistic

  $f_{att} = 1/(d_L^2)$

- Hard to control, so often use

  $f_{att} = 1/(c_1 + c_2 d_L + c_3 d_L^2)$

Recall Reflectance Equation

Attenuation: Atmospheric (fog, haze)

- z_n and z_f: near and far depth-cue plane
- s_n and s_f: scale factors
- I_{dc}: depth cue color
- Given $z_n < z_0 < z_f$
  interpolate $s_n < s_0 < s_f$

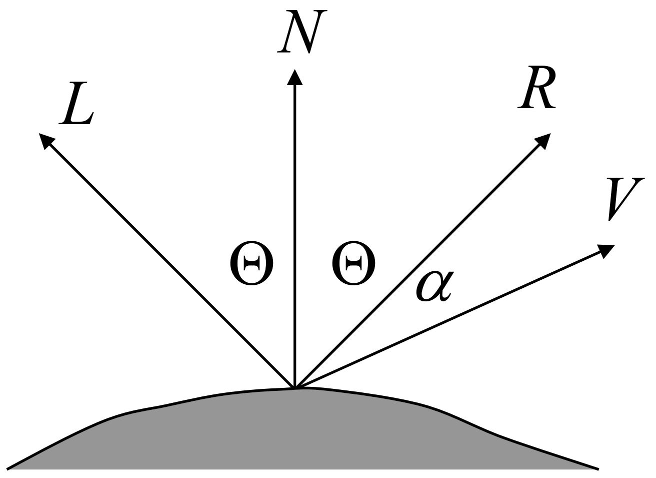
- Adjust intensity

  $I' = s_0 I + (1 - s_0) I_{dc}$

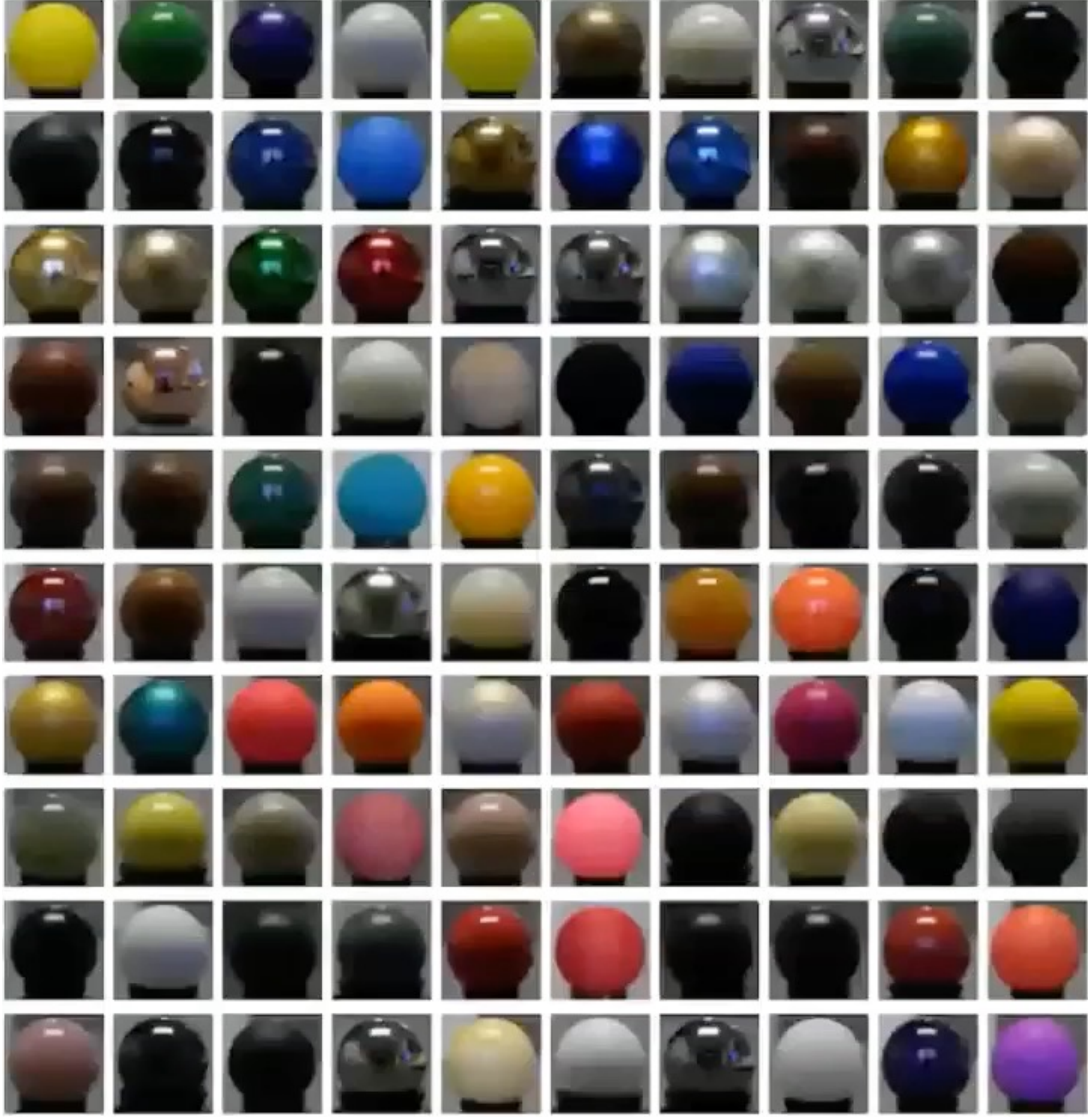
Specular Reflection: Phong Model

- Account for viewer position
  - Create highlights
- Based on $\cos^n \alpha = (R \cdot V)^n$
  - Larger *n*, smaller highlight
- k_s: specular reflection coef.

$$I = I_a k_a O_d + f_{att} I_p [\, k_d O_d (N \cdot L) + k_s (R \cdot V)^n]$$



Specular Power



Materials, Highlight Color

Multiple Light Sources

Obvious summation over *m* lights:

$$I = I_a k_a O_d + \sum_{1 \le i \le m} f_{att i} I_{pi} [\, k_d O_d (N \cdot L_i) + k_s (R_i \cdot V)^n]$$