10 – surface shading

Illumination and Shading

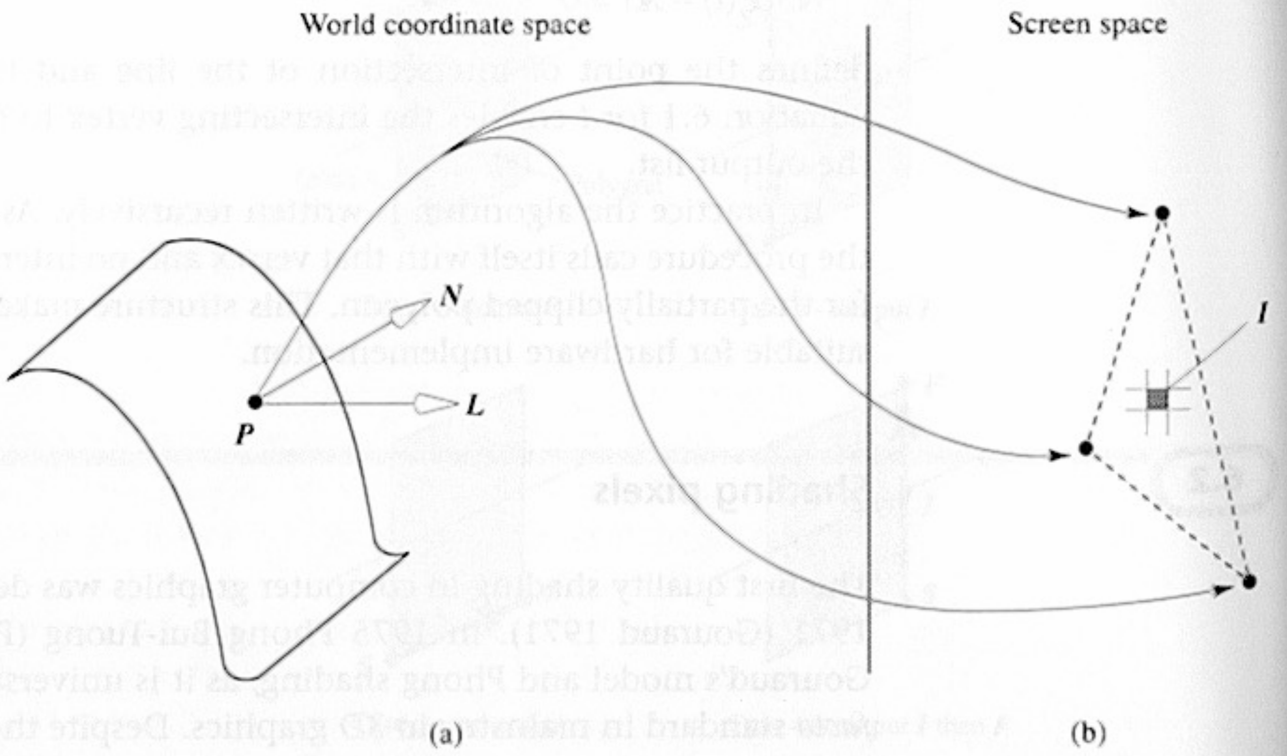


Figure 6.5 Illustrating the difference between local reflection models and shading algorithms. (a) Local reflection models calculate light intensity at any point ***P*** on the surface of an object. (b) Shading algorithms interpolate pixel values from calculated light intensities at the polygon vertices.

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

  $I = k_i$

- Use nondirectional lights

  $I = I_a k_a$

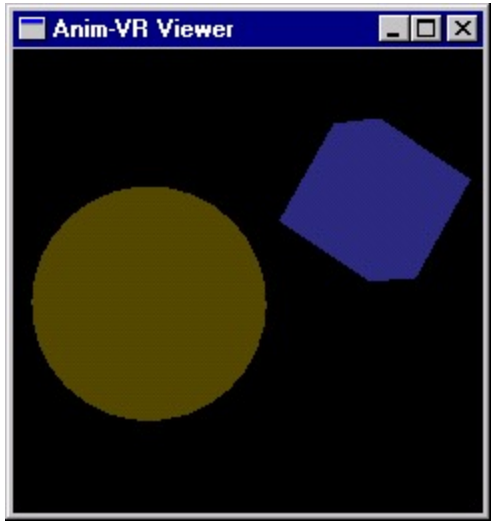
- I_a = ambient light intensity

  (I_{aR}, I_{aG}, I_{aB})

- k_a = ambient-reflection coefficient

  (k_{aR}, k_{aG}, k_{aB})

- Uniform across surface



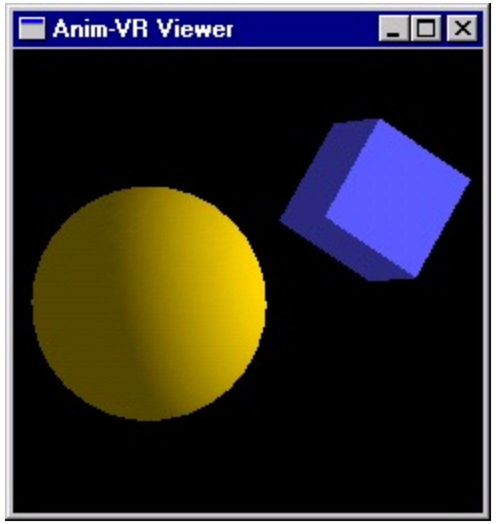
Diffuse Light

- 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)

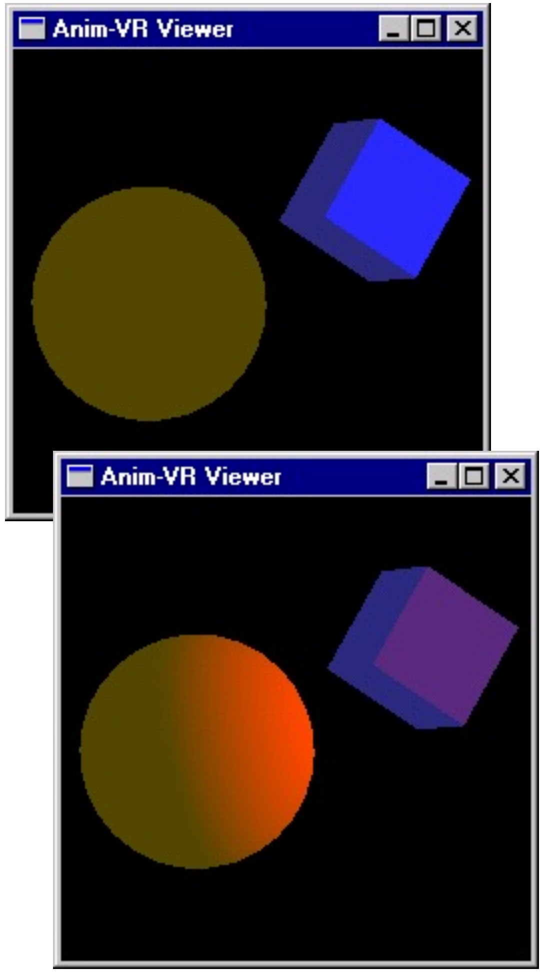
- O_d: diffuse color

  $O_{d} = (O_{dR}, O_{dG}, O_{dB})$

- Compute for each component
- i.e. red component is

  $I_R = I_{aR} k_{aR} O_{dR} + f_{att} I_{pR} k_d O_{dR} (N \cdot L)$

- 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

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

- Realistic

  $f_{\text{att}} = 1/(\mathrm{d}_L{}^2)$

- Hard to control, so often use

  $f_{\text{att}} = 1/(c_1 + c_2 \mathrm{d}_L + c_3 \mathrm{d}_L{}^2)$

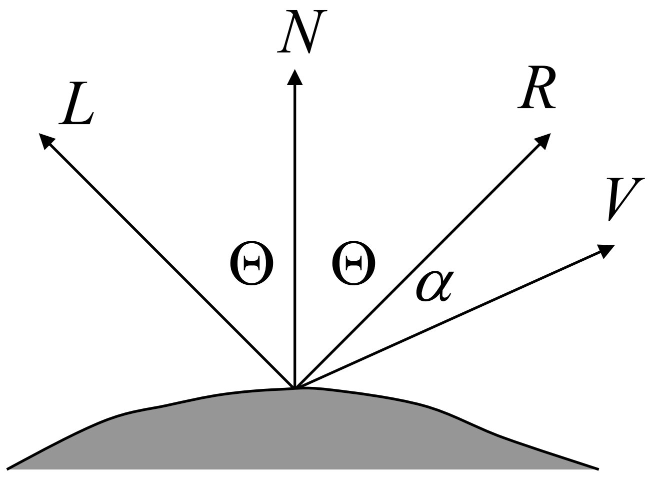
Recall Reflectance Equation

Attenuation: Atmospheric (fog, haze)

- z_f and z_b: near/far depth-cue plane
- s_f and s_b: scale factors
- I_{dc}: depth cue color
- Given $z_f > z_0 > z_b$ interpolate $s_f > s_0 > s_b$
- Adjust intensity
  $I' = s_0 I + (1 - s_0) I_{dc}$

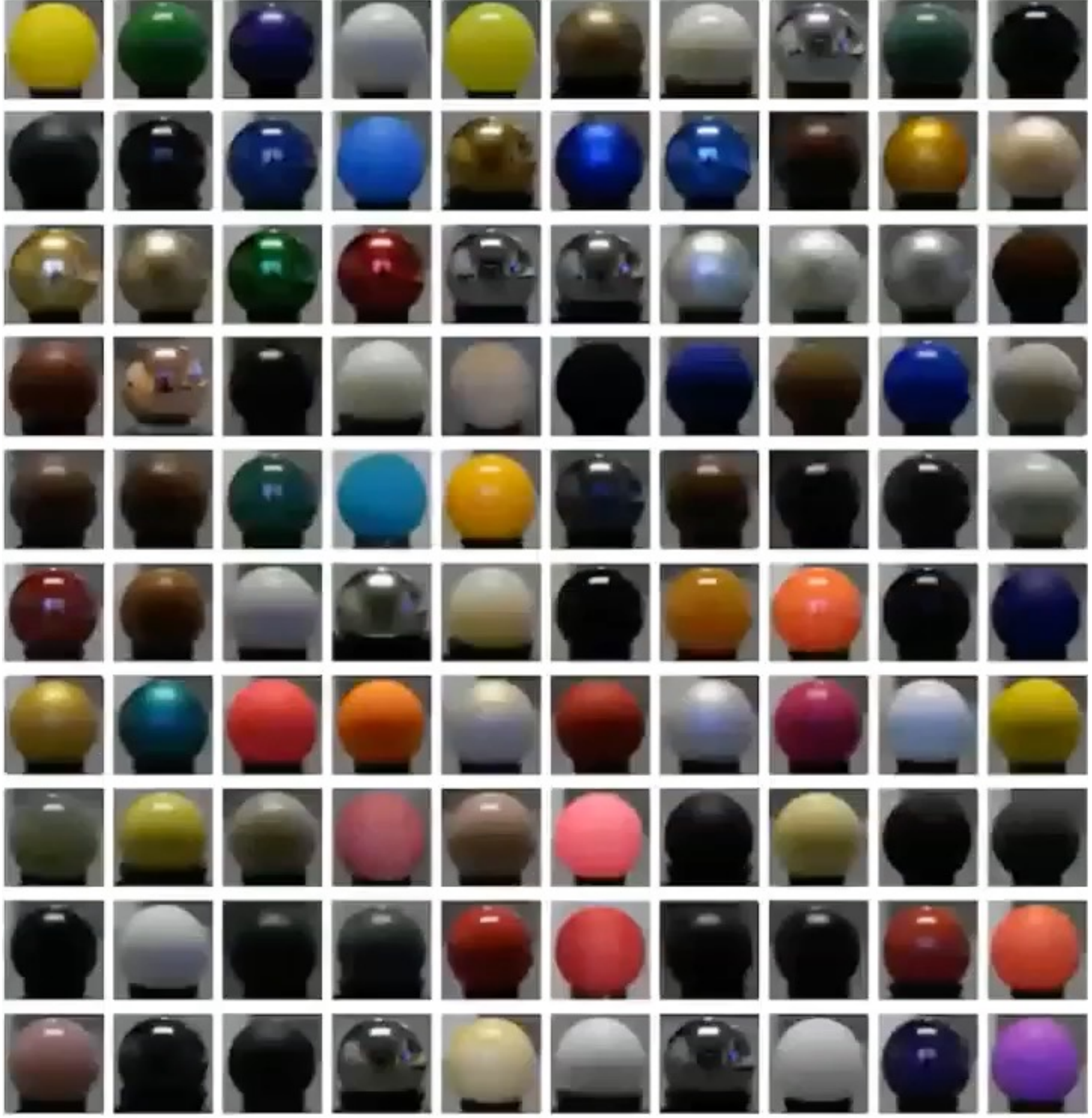
Specular Reflection: Phong Model

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



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

Specular Power



Materials, Highlight Color

Multiple Light Sources

Obvious summation over *m* lights:

$$I = I_a k_a O_d + \sum_{1 \le 0 \le m} f_{att i} I_{p i} \,[\, k_d O_d \,(N \cdot L_i) + k_s \,(R_i \cdot V)^n]$$

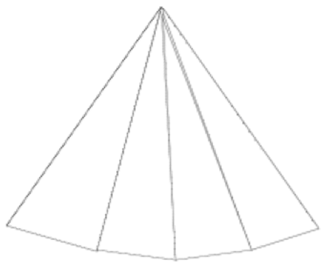
Shading Models

Surface color in this model = ambient + diffuse + specular

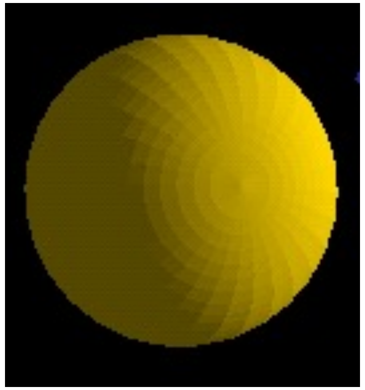
To shade triangles:

1) Per Triangle
2) Per Vertex
3) Per Pixel

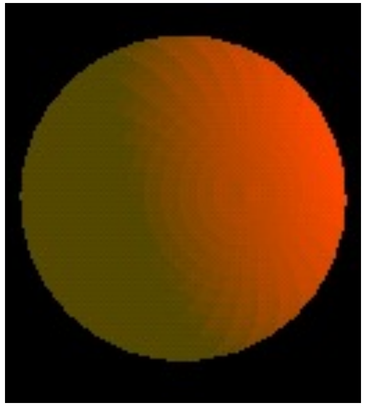
Shading Models: Per Triangle (Flat Shading)



- Compute one color for polygon
  - Use polygon normal in lighting eqs.
- Every pixel is assigned same color

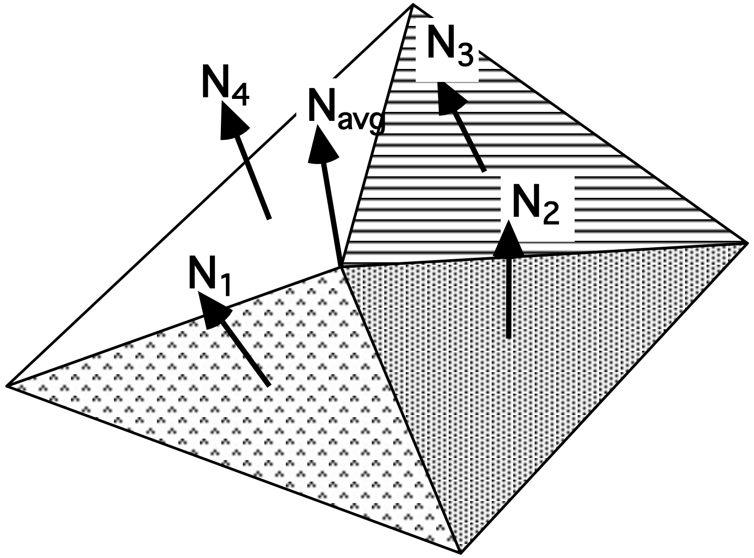


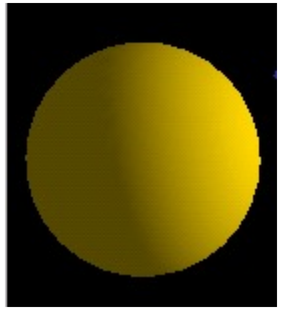
- Fast and simple
- Shade of polygons independent

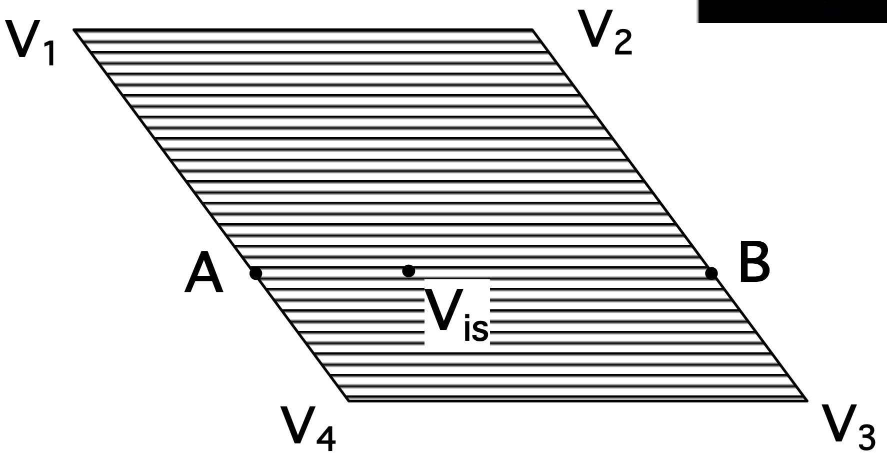


Shading Models: Per Vertex (Gouraud Shading)

- Compute vertex normals
  - Average normals of abutting polygons
- Use vertex normal in lighting eqs.
- Linearly interpolate vertex intensities
  - Along edges
  - Along scan lines







Gouraud Shading

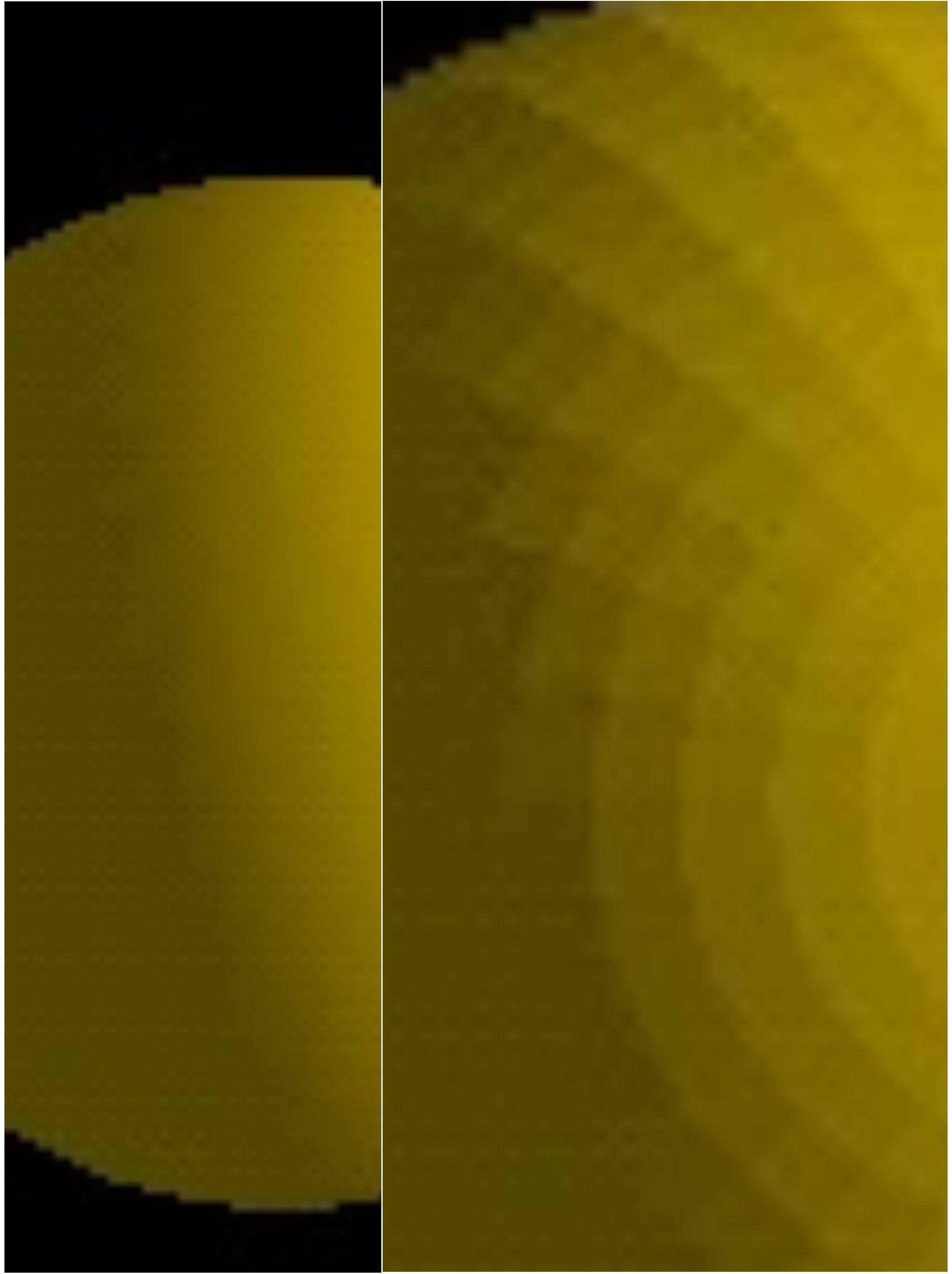
Often appears dull, chalky

- Lacks accurate specular component
  - If included, will be averaged over entire polygon

Flat Shading

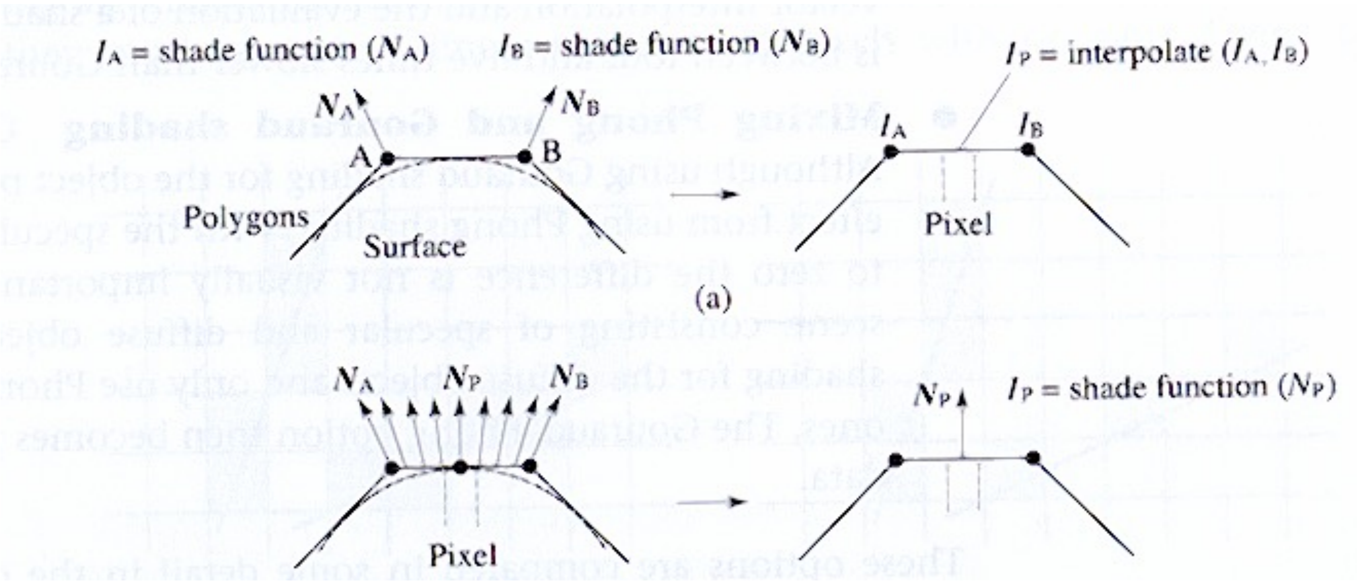
Mach banding

- Artifact at discontinuities in intensity or intensity slope



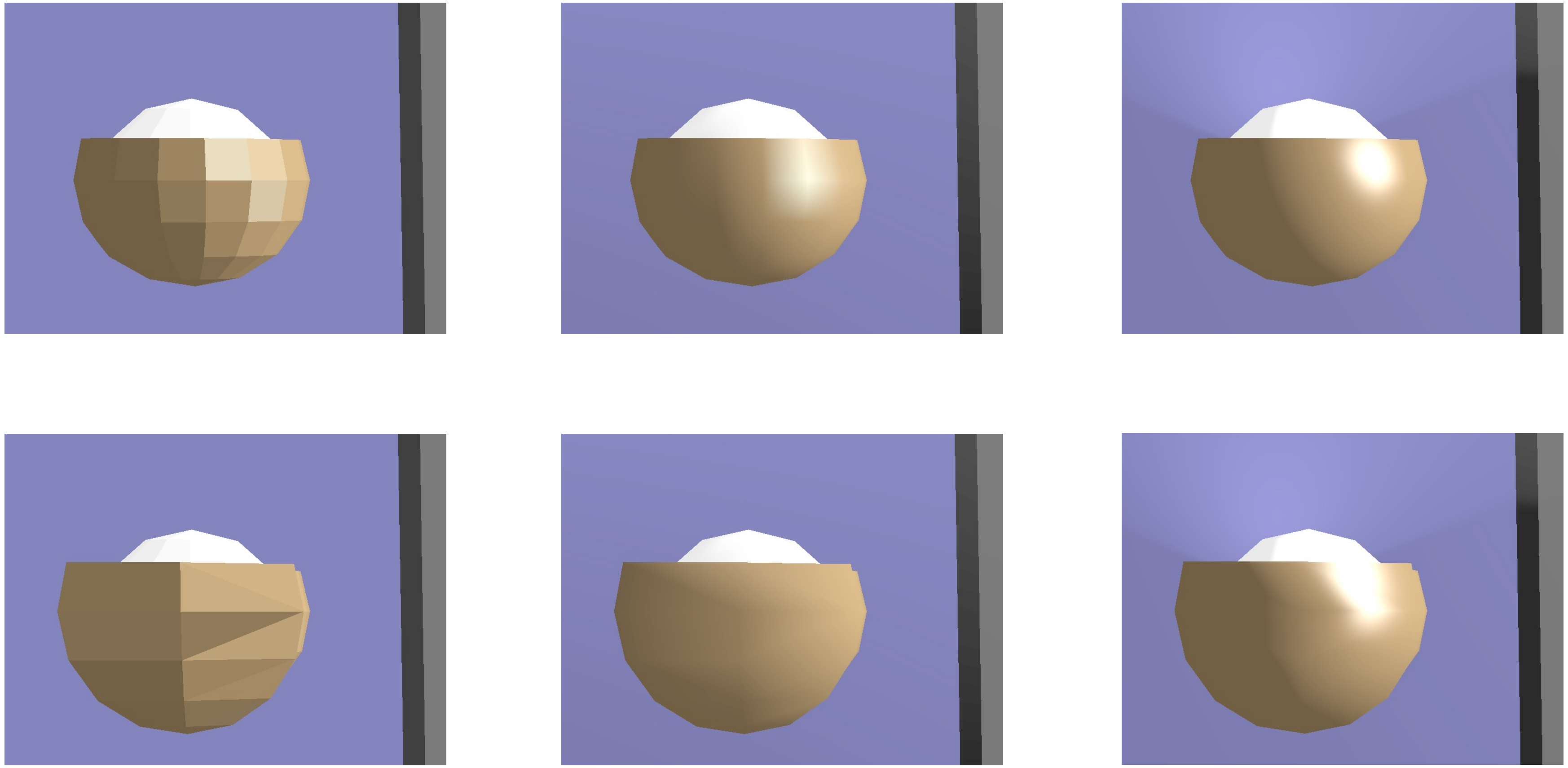
Shading Models: Per Pixel (Phong Shading)

- Linearly interpolate vertex normals
  - Compute lighting eqs. at each pixel
    - Normals must be backmapped to WC



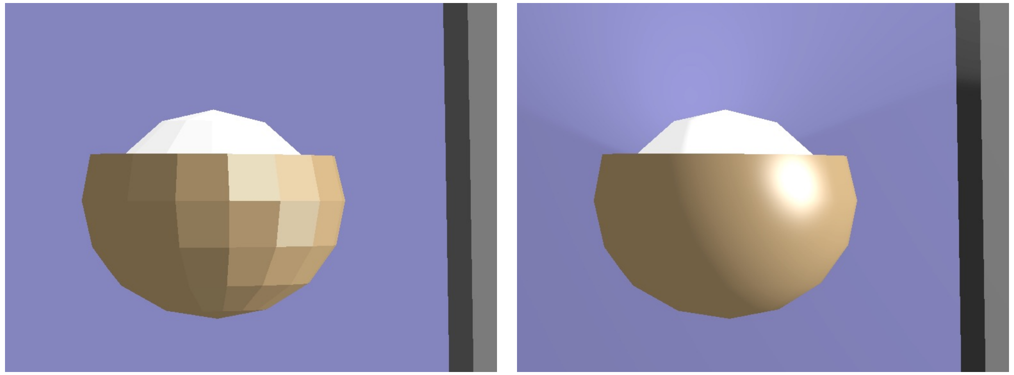
- Can use specular component

Closeup: Flat, Gouraud, Phong

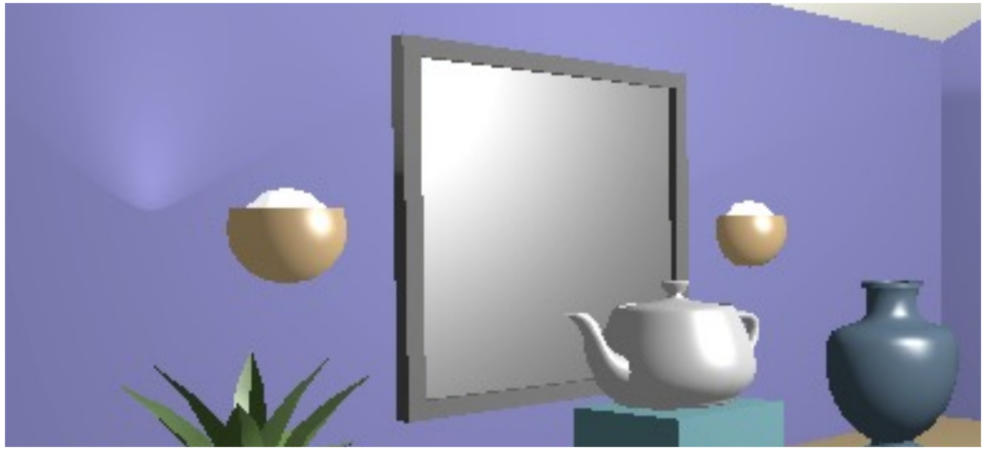


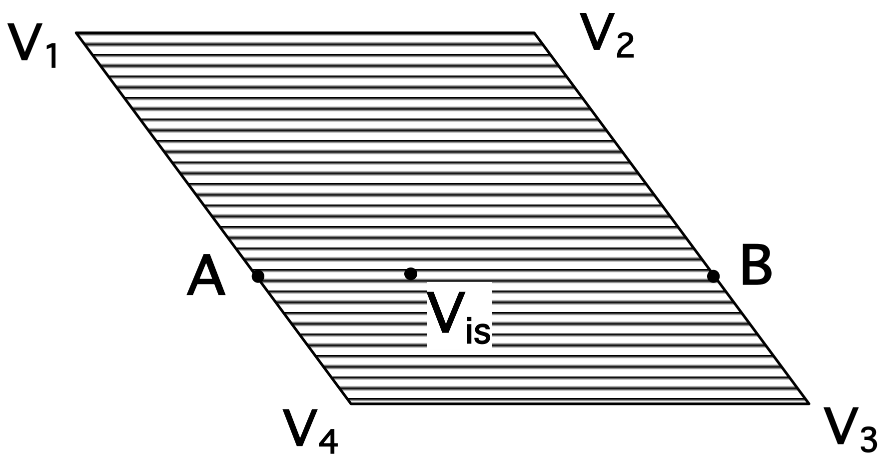
Problems with Interpolated Shading

- Polygonal silhouette



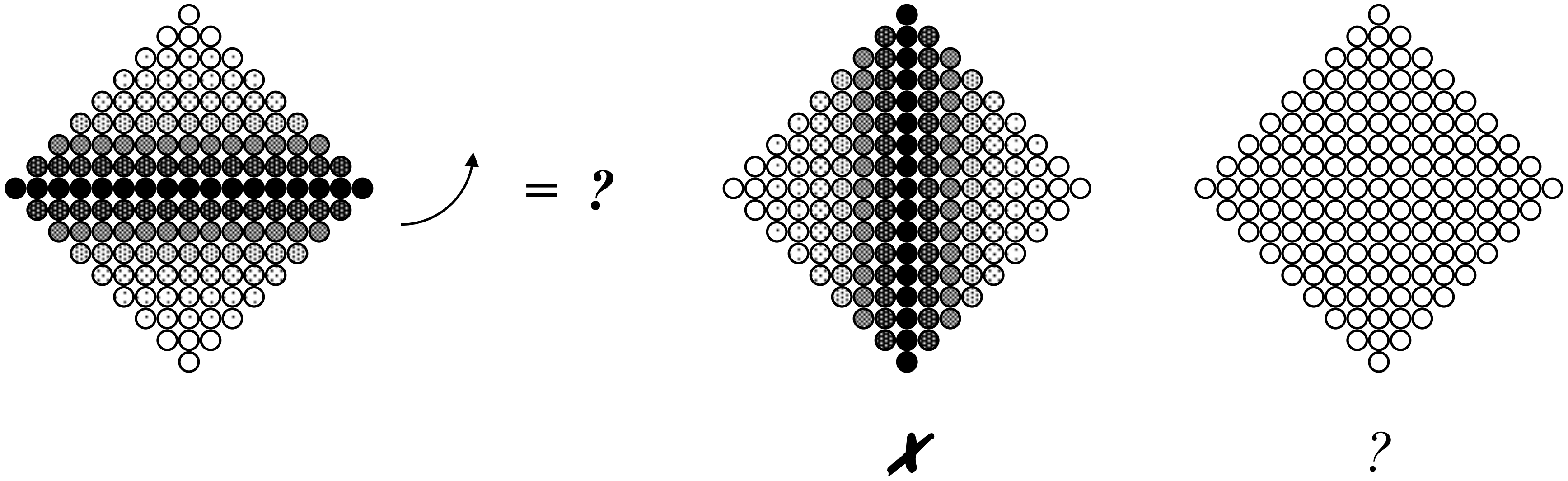
- Perspective distortion





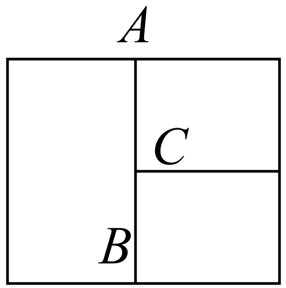
Problems with Interpolated Shading

- Scanline/orientation dependent
  - Creates temporal aliasing when used to render animation frames:



Problems with Interpolated Shading

- Shared vertices



- Unrepresentative vertex normals
  - Missed specular highlights
  - Missed geometry