3D Lighting

Thomas A. Funkhouser

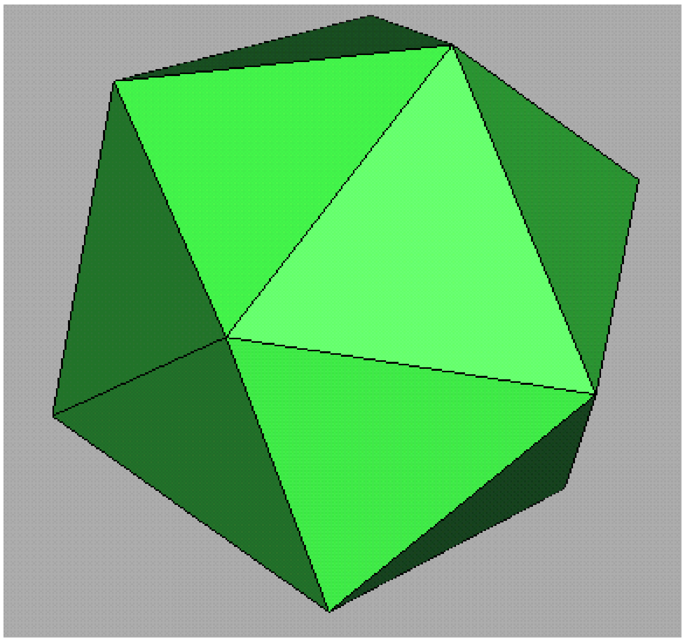
Overview

- **Motivation**
- **Lighting models**
  - Illumination models
  - Light sources
- **Polygon shading methods**
  - Flat
  - Gouraud
  - Phong

Motivation

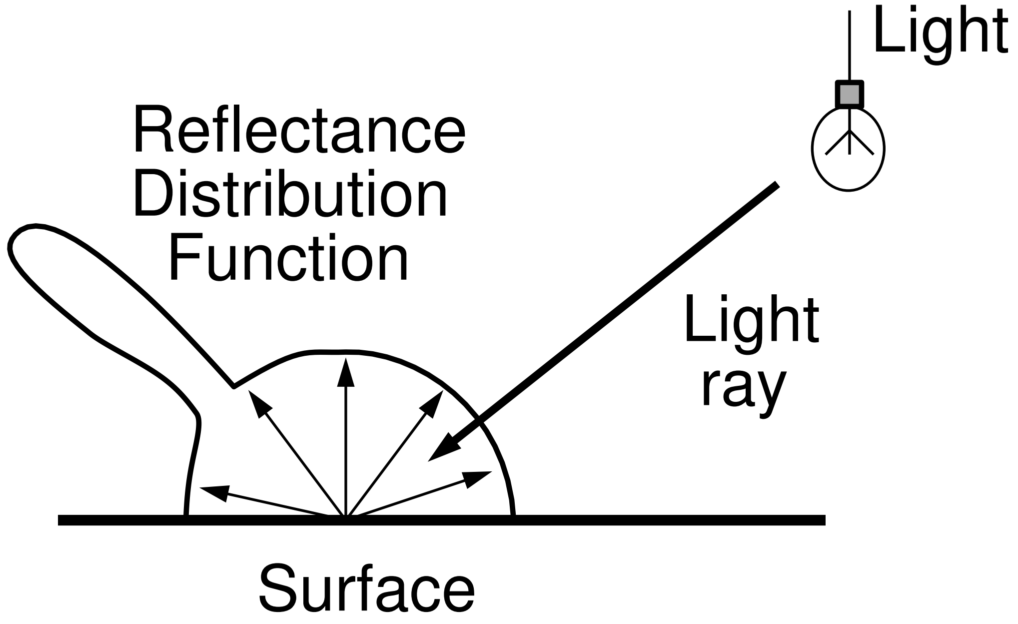
- Photorealism:

- Visual cues:



Surface Illumination

- **Emitted Light**
- **Diffuse reflection**
- **Specular reflection**
- **Ambient light**

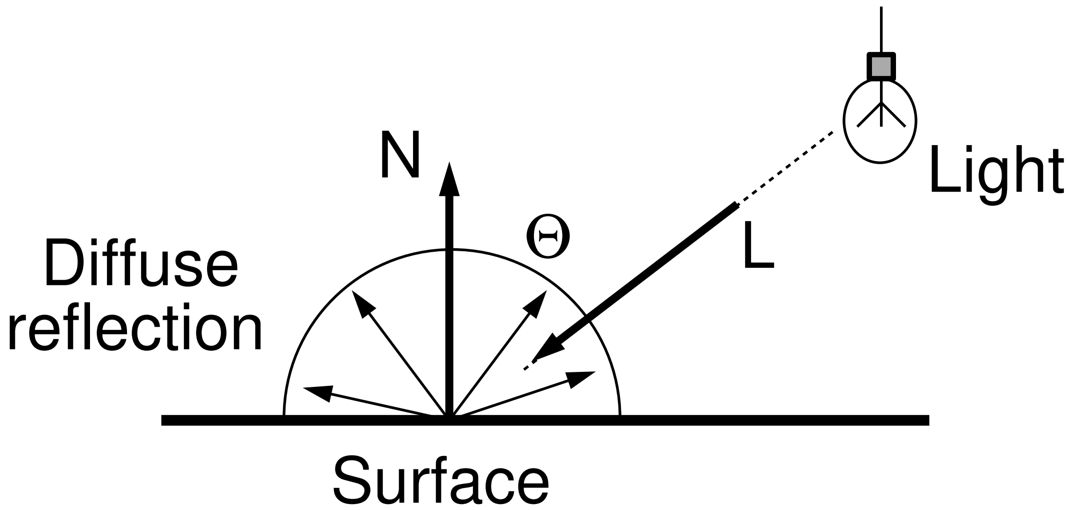


Emitted Light

- **Emitted light:**
  - Inherent emission
  - For rendering light sources

Diffuse Reflection

- **Diffuse reflection**
  – Reflection in all directions
  – Grainy surfaces (e.g., clay)



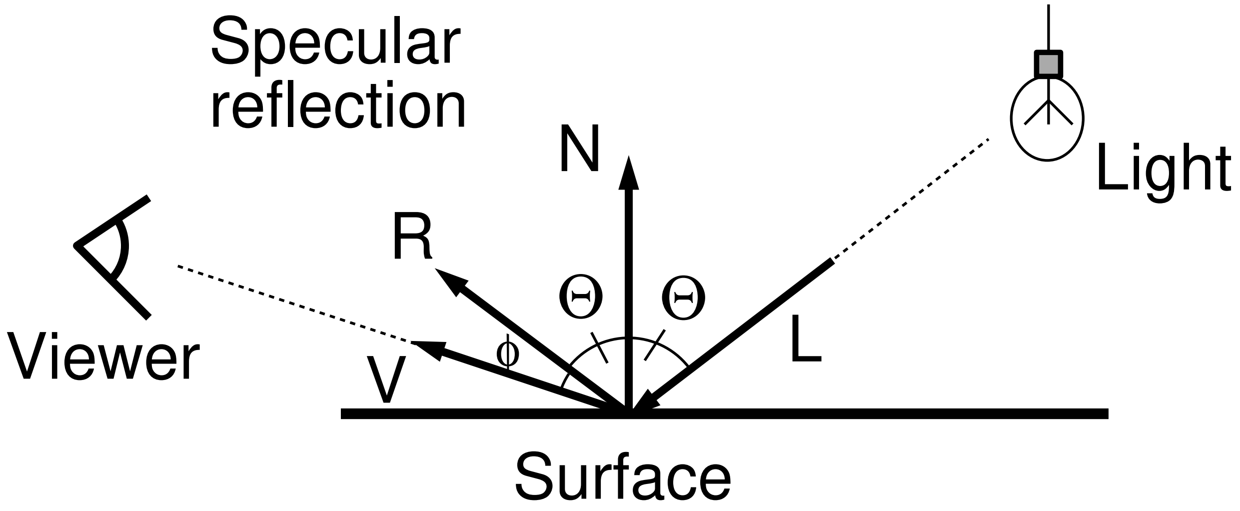
- **Lambertian model**

$$I_D = K_D\,(N \bullet L)\,I_L$$

K_D = material diffuse reflection
$N \bullet L = \cos(\Theta)$ ($|N|=1$ and $|L|=1$)

Specular Reflection

- **Specular reflection**
  - Reflection near incident angle
  - Shiny surfaces (e.g., metal)



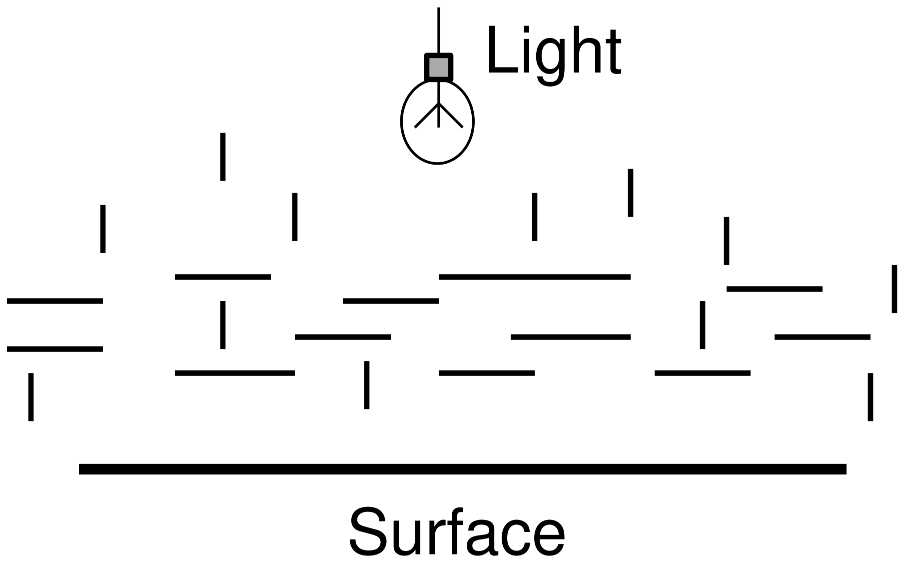
- **Phong model**

$$I_S = K_S \, (V \bullet R)^n \, I_L$$

K_s = material specular reflection
n = specular reflection exponent
$V \bullet R = \cos(\phi)$ ($|V|=1$ and $|R|=1$)

Ambient Light

- **Ambient light**
  - Background light
  - Equal everywhere
  - Approx for indirect lighting



- **Ambient model**

$$I_A = K_A I_{AL}$$

K_a = material ambient reflection

Surface Illumination

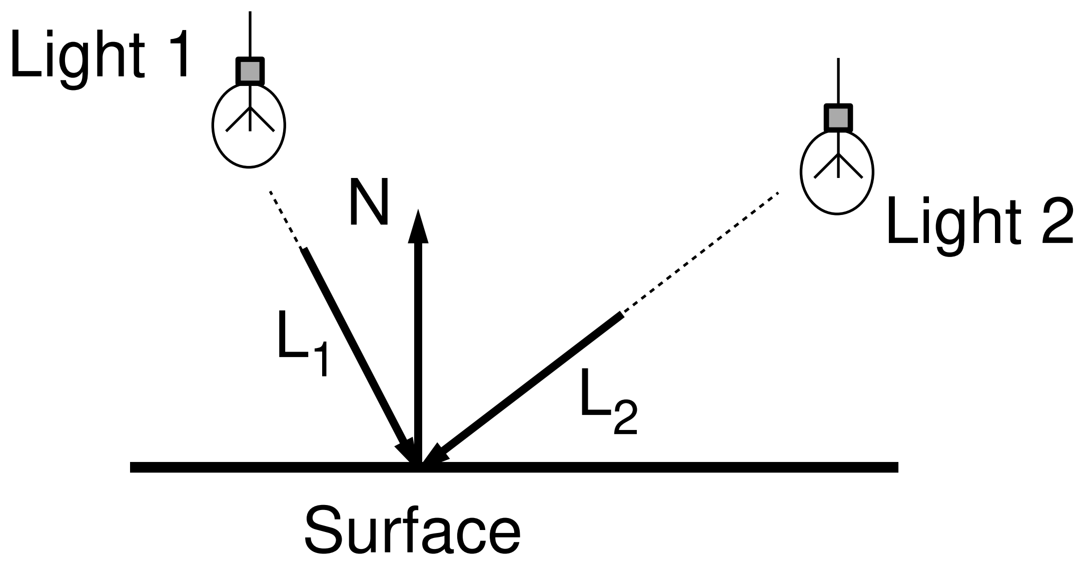
- **Single Light Source**

$$I = I_E + I_D + I_S + I_A$$

$$I = I_E + K_D\,(N \bullet L)\, I_L + K_S\,(V \bullet R)^n\, I_L$$

- **Multiple Light Sources**

$$I = I_E + K_A I_{AL} + \sum^{L_i} (\, K_D\,(N \bullet L_i) + K_S\,(V \bullet R_i)^n\,)\, I_{L_i}$$

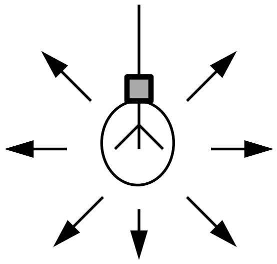


Light Source Models

- **Point light sources**
- **Directional light sources**
- **Spot light sources**
- **Polygonal light sources**

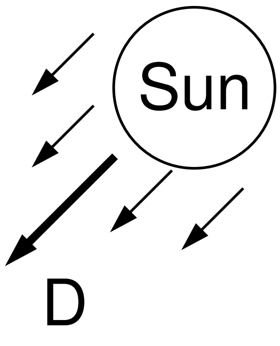
Light Source Models

- **Point light sources**
  - Position, illumination (I)
  - Works best if size of light source is small relative to scene
  - Example: light bulb



$$I_L = I / (d_0 + d)^2$$

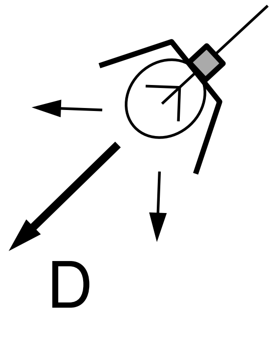
- **Directional light sources**
  - Direction (D), illumination (I)
  - Works best if distance to light source is large relative to scene
  - Example: the sun



$$I_L = I\,(D \bullet L)$$

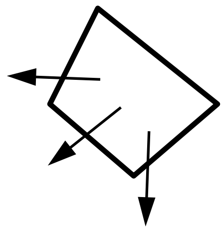
Light Source Models

- **Spot light sources**
  - Position, direction (D), illumination (I)



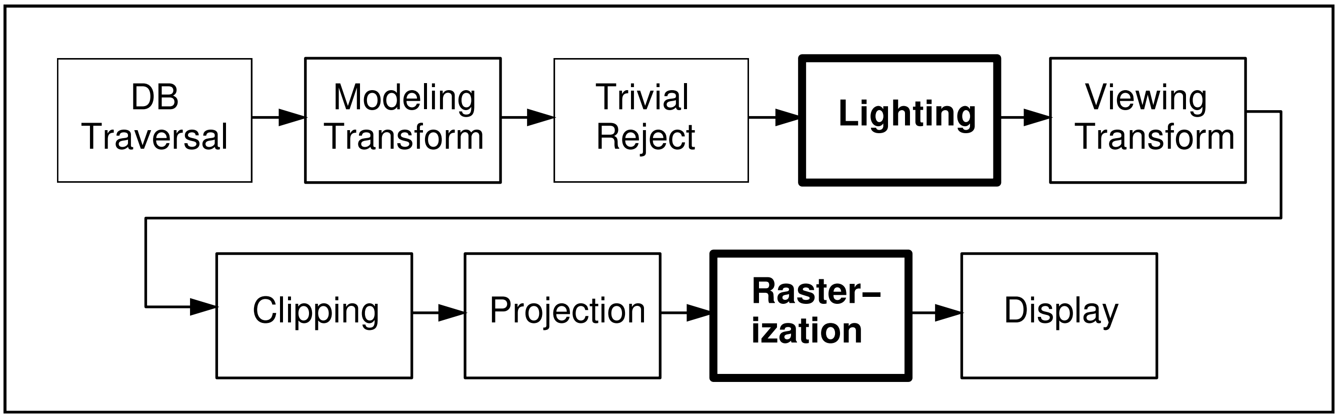
$$I_L = I\,(D \bullet L) / (d_0 + d)^2$$

- **Polygonal light sources**
  - Polygon, illumination
  - Computing shading requires integration over area
  - Used in radiosity



Polygon Shading

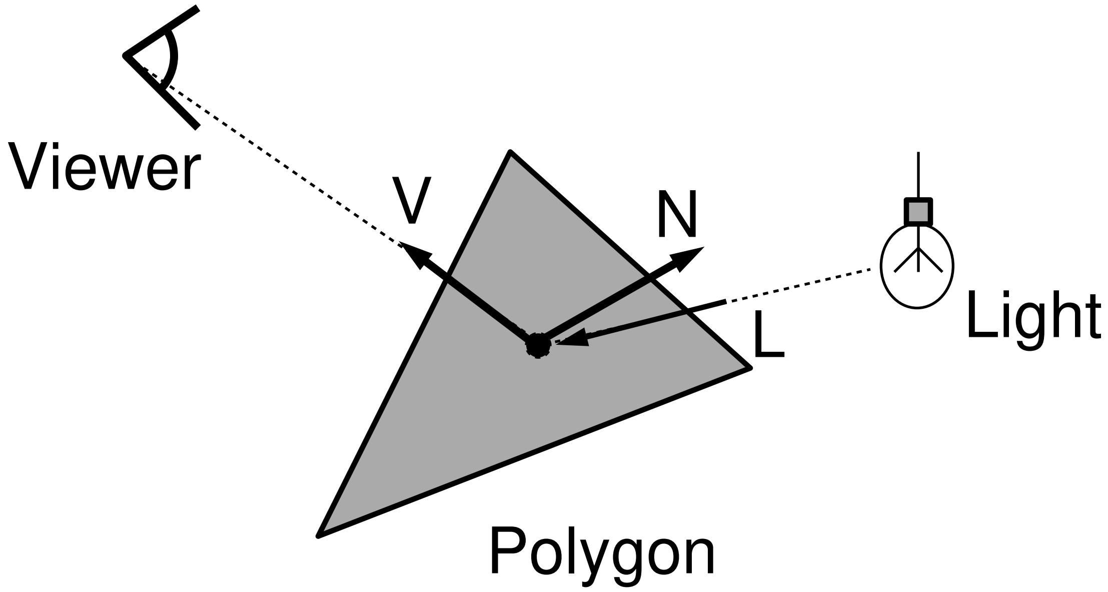
- **Goal**
  - Fill pixels as quickly as possible with most realistic shading.
- **Polygon shading methods**
  - Flat
  - Gouraud
  - Phong



Sample Rendering Pipeline

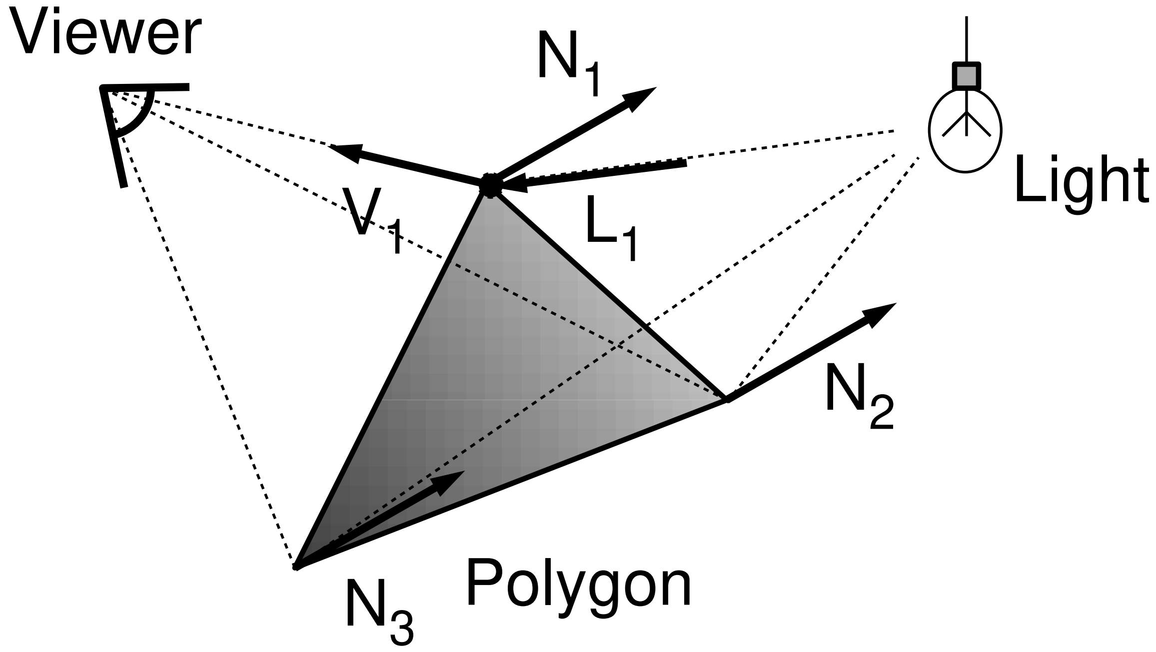
Flat Shading

- **Method**
  - Apply constant color for entire polygon based on illumination computed at centroid



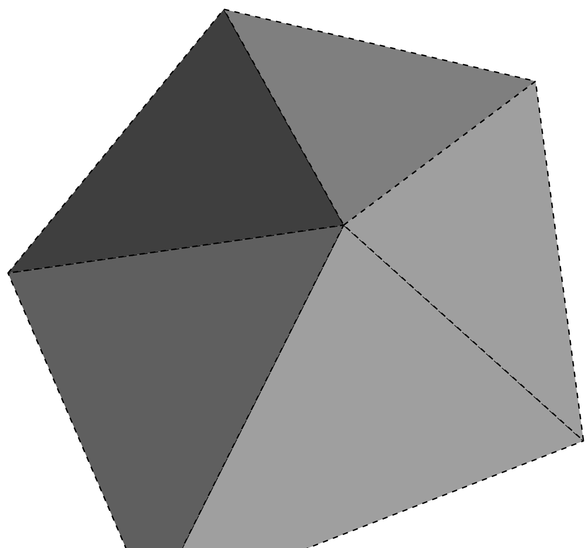
Gouraud Shading

- **Method**
  - Apply bilinear interpolated color across polygon based on illumination computed at vertices

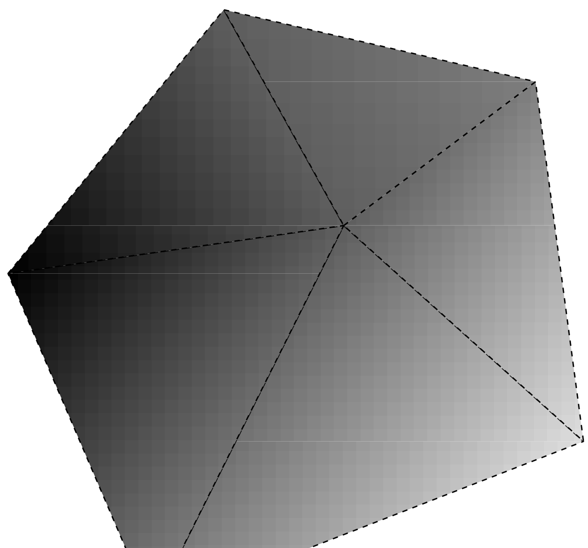


Gouraud Shading

- **Primary use:**
  - Polygon approximations of curved surfaces
  - Shade based on averaged normal for vertices shared by more than one adjacent polygon



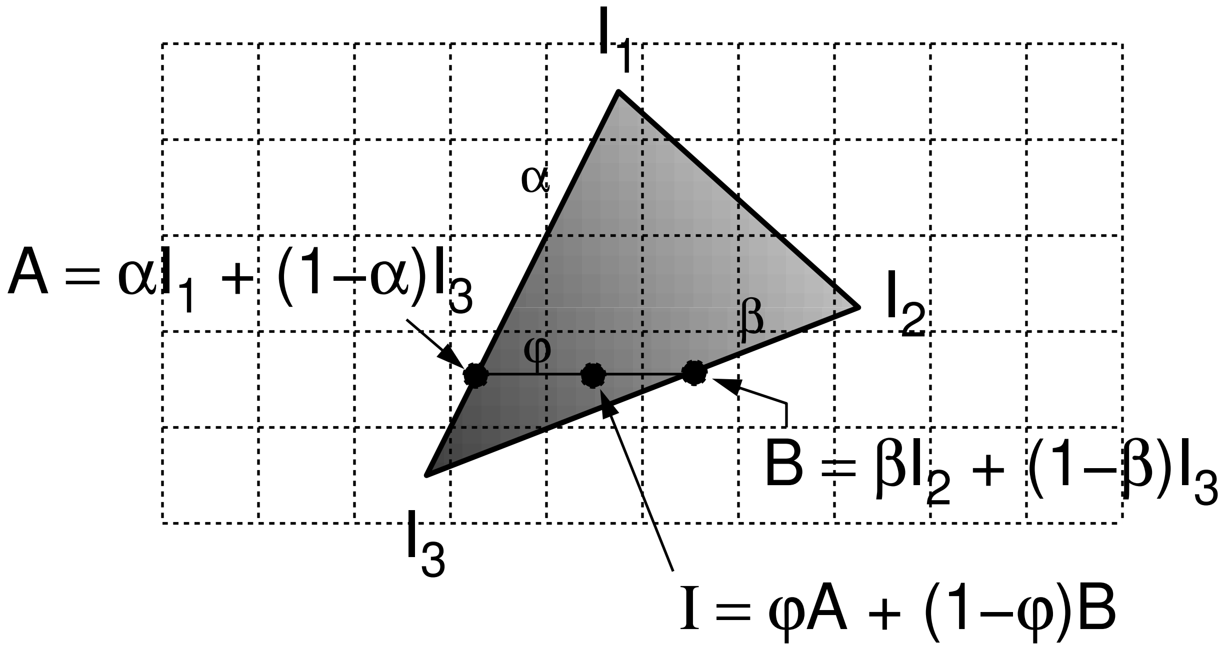
Flat Shading



Gouraud Shading

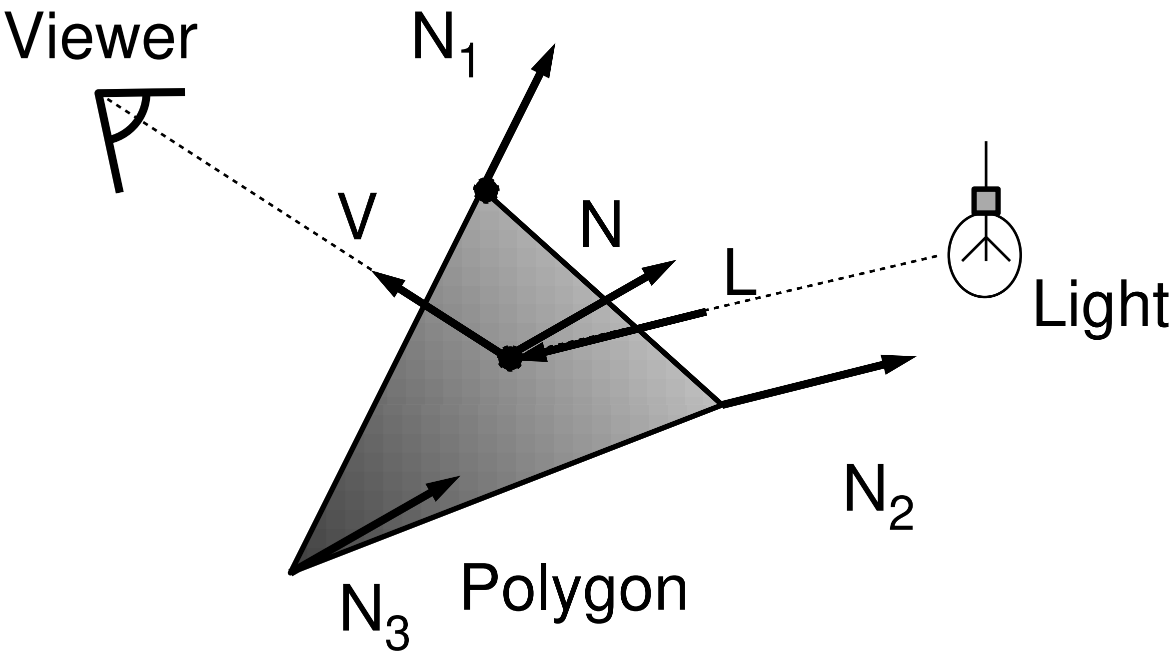
Gouraud Shading

- **Scan conversion**
  - Interpolate vertex colors down and across scan lines



Phong Shading

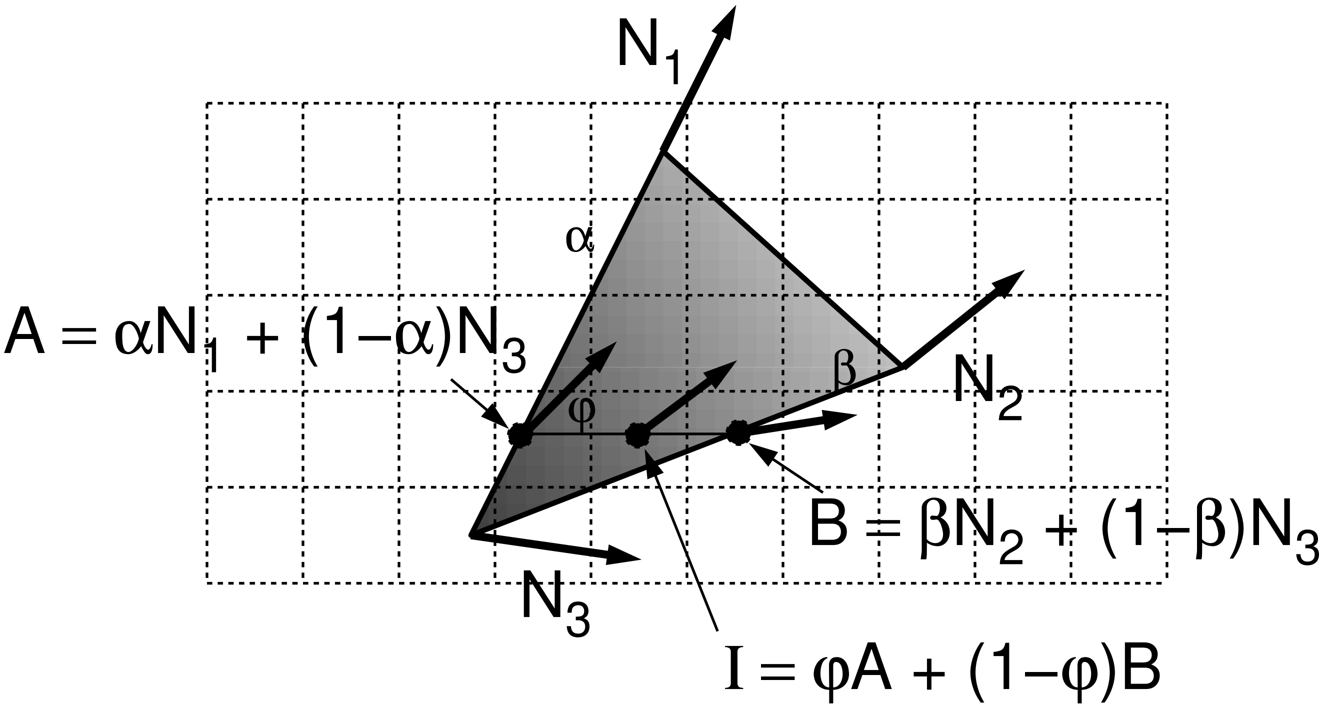
- **Method**
  - Compute illumination at every pixel based on bilinear interpolation of vertex normals



  - Useful for specular highlights

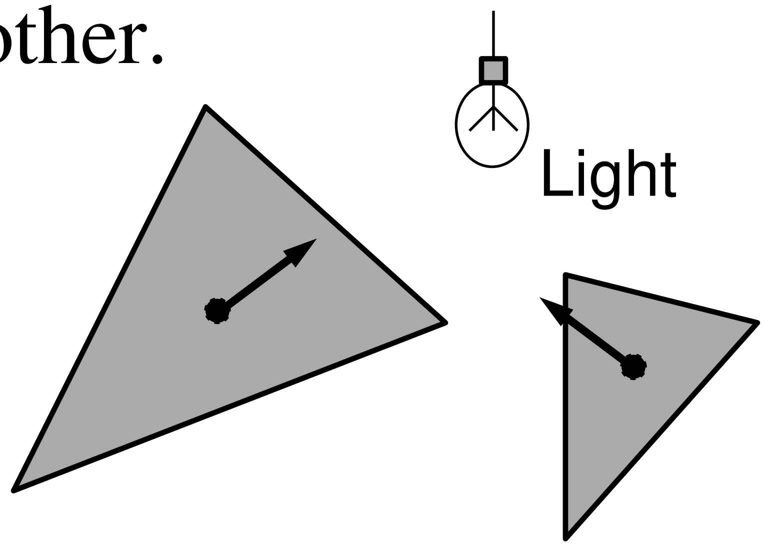
Phong Shading

- **Scan conversion**
  - Interpolate vertex normals down and across scan lines



Comments

- **Goal:**
  - Simulate physics
- **The Rendering Equation:**
  - Expresses the light transfered from one point in scene to another.



- **Final note:**
  - Everything in this lecture represents cheap approximations of physical effects that are very difficult to compute perfectly

Conclusion

- **Direct illumination models**
  - Surface reflection
  - Light sources
- **Direct polygon shading**
  - Flat
  - Gouraud
  - Phong
  - Textures (next time)
- **Indirect illumination**
  - Ray tracing
  - Radiosity