3D Illumination and Shading

- Problem: Model light/surface point interactions to determine final color and brightness
- Actual light computation is too costly!
- Apply the lighting model at a set of points across the entire surface
Illumination Model

- The governing principles for computing the illumination
- An illumination model usually considers
  - Light attributes (intensity, color, position, direction, shape)
  - Object surface attributes (color, reflectivity, transparency, etc.)
  - Interaction among lights and objects

Basic Light Sources

- Light intensity can be independent or dependent of the distance between object and the light source

- Point light
- Spot light
- Directional light
- Area light
Local Illumination

- Only consider the light, the observer position, and the object material properties

Global Illumination

- Take into account the interaction of light from all the surfaces in the scene
- Example:
  - Ray Tracing
  - Model light rays bouncing around

object 1

object 2

object 3

object 4
Global Illumination (cont.)

- Example:
  - Radiosity
    - Model energy moving from emitters (e.g., lights) into the scene
    - View independent

Simple Local Illumination

- Reduce the complex workings of light to three components
  - Ambient
  - Diffuse
  - Specular

- Final illumination at a point (vertex) = ambient + diffuse + specular

- Materials reflect each component differently
  - Use different material reflection coefficients
    - $K_a$, $K_d$, $K_s$
### Ambient Light Contribution

- Ambient light = background light
- Light that is scattered by the environment
  - It’s just there
- **Frequently assumed to be constant**
- Very simple approximation of global illumination
- No direction: independent of light position, object orientation, observer’s position/orientation

\[
\text{Ambient} = I \times K_a
\]

### Diffuse Light Contribution

- Diffuse light: The illumination that a surface receives from a light source that reflects equally in all direction
  - Eye point does not matter
Diffuse Light Calculation

- Need to decide how much light the object point receives from the light source
  - Based on Lambert’s Law

\[
\text{Diffuse} = K_d \times I \times \cos(\theta)
\]

- \(K_d\): diffuse reflection coefficient
- \(I\): light intensity
- \(\theta\): angle between the light vector and the surface normal
- Light vector (vector from object to light)
- \(N\): surface normal
Diffuse Light Examples

$I = 1.0$  $K_d = 0.0$  $0.2$  $0.4$  $0.6$  $0.8$  $1.0$

Specular Light Contribution

- The bright spot on the object
- The result of total reflection of the incident light in a concentrate region

Sees no specular  Sees lots of specular
Specular Light Calculation

- How much reflection you can see depends on where you are
  - But for non-perfect surface you will still see specular highlight when you move a little bit away from the ideal reflection direction

Φ is deviation of view angle from mirror direction
- When Φ is small, you see more specular highlight

Specular Light Calculation (cont.)

- Phong lighting model
  - Not Phong shading model
  - Specular = $K_s \times I \times \cos(\phi)$

- The effect of 'f' in the Phong model

$$f = 10 \quad f = 30 \quad f = 90 \quad f = 270$$
Specular Light Examples

\[ K_s = 0.25 \]

\[ K_s = 0.5 \]

\[ K_s = 0.75 \]

\[ f = 3 \quad f = 6 \quad f = 9 \quad f = 25 \quad f = 200 \]

Putting It All Together

- Illumination from a light

\[ \text{Illum} = \text{ambient} + \text{diffuse} + \text{specular} \]

\[ = K_a \times I + K_d \times I \times \cos(\theta) + K_s \times I \times \cos^f(\phi) \]

- If there are \( N \) lights

Total illumination for a point \( P = \sum (\text{Illum}) \)

- Some more terms to be added
  - Self emission
  - Global ambient
  - Light distance attenuation and spot light effect
Putting It All Together (cont.)

- $\text{Illum} = \text{ambient} + \text{diffuse} + \text{specular}$

![Image of lighting components](image1)

ambiguous lighting example

![Image of ambient lighting example](image2)
Diffuse Lighting Example

Specular Lighting Example
Polygon Shading Models

- Flat shading
  - Compute lighting once and assign the color to the whole polygon (or mesh)

Gouraud Shading

- Lighting is calculated for each of the polygon vertices
- Colors are interpolated for interior pixels
Ambient Term Only

Flat Shading
Diffuse Shading + Interp. Normals

Gouraud Shading
Ambient + Diffuse + Specular

Ambient + Diffuse + Specular + Interpolated Normals
Radiosity

Radiosity + Texture Mapping
Texture Mapping + Ray Tracing