CS 4731: Computer Graphics
Lecture 15: Illumination Models Part 1

Emmanuel Agu

Announcements

- Midterm
  - Return on Monday
  - Scores will be on myWPI over the weekend
- Project 4:
  - On class website later today
  - Due next Friday

Illumination and Shading

Problem: Model light/surface points interaction to determine final color and brightness

Illumination and Shading
Shading
- Apply the lighting model at a set of points across the entire surface

Illumination Model
- The governing principles for computing the illumination
- A 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
- Point light
- Directional light
- Spot light
- Area light

Local Illumination
- Local illumination: only consider the light, the observer position, and the object material properties
- OpenGL does this
Global Illumination

- Global illumination: take into account the interaction of light from all the surfaces in the scene
- Example: Ray tracing

Simple Local Illumination

- The model used by OpenGL
- Consider three types of light contribution to compute the final illumination of an object
  - Ambient
  - Diffuse
  - Specular
- Final illumination of a point (vertex) = ambient + diffuse + specular

Ambient Light Contribution

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

Ambient Light Example
**Ambient Light Calculation**
- Each light source has ambient light contribution \( I_a \)
- Different objects can reflect different amounts of ambient
- Different ambient reflection coefficients \( K_a \), \( 0 \leq K_a \leq 1 \)
- So, ambient light from an object is:

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

**Diffuse Light Contribution**
- Diffuse light: The illumination that a surface receives from a light source and reflects equally in all directions.

**Diffuse Lighting Example**
- Need to decide how much light the object point receive from the light source – based on Lambert’s Law

**Diffuse Light Calculation**
- Receive more light
- Receive less light
**Diffuse Light Calculation**

- Lambert’s law: the radiant energy \( D \) that a small surface patch receives from a light source is:
  \[
  D = I \times \cos(\theta)
  \]
- \( I \): light intensity
- \( \theta \): angle between the light vector and the surface normal

\[
\text{light vector (vector from object to light)} \quad N : \text{surface normal}
\]

- Different objects reflect different amount of diffuse light
- Different diffuse reflection coefficient \( K_d \) (\( 0 \leq K_d \leq 1 \))
- So, the amount of diffuse light that can be seen is:
  \[
  \text{Diffuse} = K_d \times I \times \cos(\theta)
  \]

\[
\text{Specular light contribution}
\]

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

\[
\text{Specular light example}
\]
Specular light calculation

- How much reflection you can see depends on where you are

Only position the eye can see specular from P if object has an ideal reflection surface

But for non-perfect surface you will still see specular highlight when you move a little bit away from the ideal reflection direction

When $\phi$ is small, you see more specular highlight

Specular light calculation

- Phong lighting model
  
  \[ \text{specular} = K_s \times I \times \cos(\phi) \]

  Ka: specular reflection coefficient
  N: surface normal at P
  I: light intensity
  V: vector from P to viewer's eye
  R: mirror-reflection direction
  $\phi$: angle between $V$ and $R$
  \( \cos(\phi) \): the larger is $n$, the smaller is the cos value
  \( \cos(\theta) \) = $R \cdot V$

Specular light calculation

- The effect of 'n' in the phong model

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

\[ = K_a \times I + K_d \times I \times (N \cdot L) + K_s \times I \times (R \cdot V) \]

- If there are N lights
  
  Total illumination for a point $P = \sum (\text{Illum})$

- Some more terms to be added (in OpenGL):
  
  - Self emission
  - Global ambient
  - Light distance attenuation and spot light effect

Put it all together
References

- Hill, chapter 8