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

Emmanuel Agu

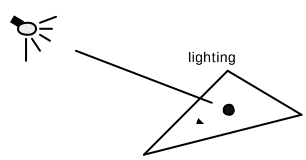
Announcements

- 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

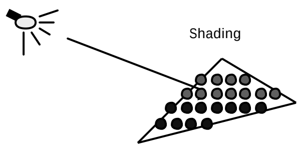
Illumination and Shading

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



Shading

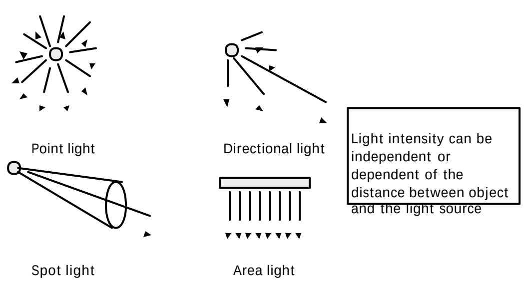
- Apply the lighting model at a set of points across the entire surface



Illumination Model

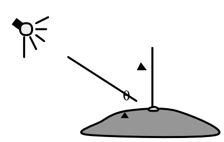
- The governing principles for computing the illumination
- A illumination model usually considers:
  - Light attributes (intensity, color, position, direction, shape)
  - Object surface attributes (color, reflectivity, transparency, etc)
  - Interaction among lights and objects

Basic Light Sources



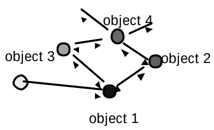
Local Illumination

- 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

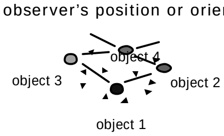


Simple Local Illumination

- 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

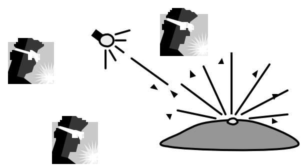
Ambient Light Calculation

- Each light source has ambient light contribution (Ia)
- Different objects can reflect different amounts of ambient
- Different ambient reflection coefficients Ka, 0 <= Ka <= 1
- So, ambient light from an object is:

$$\boxed{\text{Ambient} = \text{Ia} \times \text{Ka}}$$

Diffuse Light Contribution

- Diffuse light: The illumination that a surface receives from a light source and reflects equally in all direction

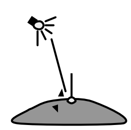


It does not matter where the eye is

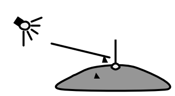
Diffuse Lighting Example

Diffuse Light Calculation

- Need to decide how much light the object point receive from the light source – based on Lambert's Law



Receive more light



Receive less light

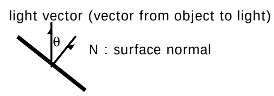
Diffuse Light Calculation

- 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

θ: angle between the light vector and the surface normal



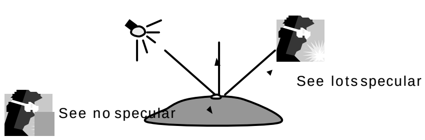
Diffuse Light Calculation

- Like ambient case, different objects reflect different amount of diffuse light
- different diffuse reflection coefficient Kd, ($0 <= Kd <= 1$)
- So, the amount of diffuse light that can be seen is:

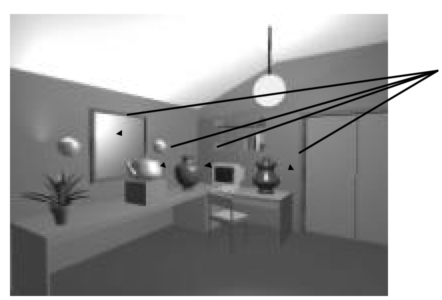
$$\boxed{\text{Diffuse} = Kd \times I \times \cos(\theta)}$$

Specular light contribution

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



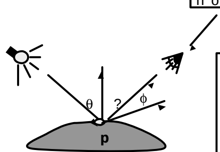
Specular light example



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 idea reflection direction

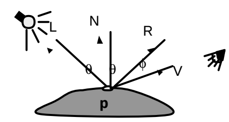
When φ is small, you see more specular highlight

Specular light calculation

- Phong lighting model

$$\text{specular} = Ks \times I \times \cos^{n}(\phi)$$

Ka: specular reflection coefficient
N: surface normal at P
I: light intensity
V: vector from P to viewer's eye
R: mirror-reflection direction
φ: angle between V and R



$\cos^{n}(\phi)$: the larger is n, the smaller is the cos value
$\cos(\theta) = R.V$

Specular light calculation

- The effect of 'n' in the phong model

n = 10


n = 90


n = 30


n = 270


Put it all together

- Illumination from a light:
  Illum = ambient + diffuse + specular
  $= Ka \times I + Kd \times I \times (N.L) + Ks \times I \times (R.V)^{n}$
- If there are N lights
  Total illumination for a point P = S (Illum)
- Some more terms to be added (in OpenGL):
  - Self emission
  - Global ambient
  - Light distance attenuation and spot light effect

References

- Hill, chapter 8