Illumination Models and Surface Rendering

Eric C. McCreath

School of Computer Science
The Australian National University
ACT 0200 Australia

ericm@cs.anu.edu.au

Overview

- Light Sources
- Ambient, Diffuse, and Specular Lighting
- Phong Reflection Model
- Surface Rendering - Phong , Gouraud

See chapter 10 of Hearn et al.

Introduction

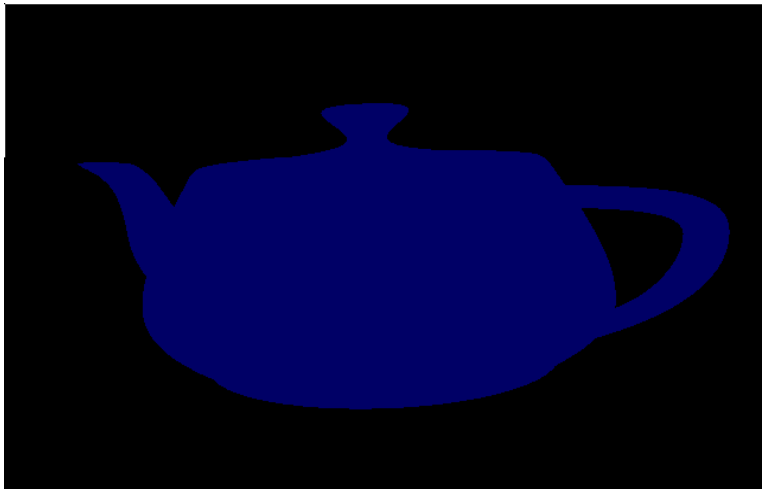
- The illumination model is used to determine the colour at points on an object's surface.
- Surface-rendering uses this information to generate the pixel colours that make up the entire scene.
- Lighting effects are complex, hence, simplification is required to reduce computational overhead.

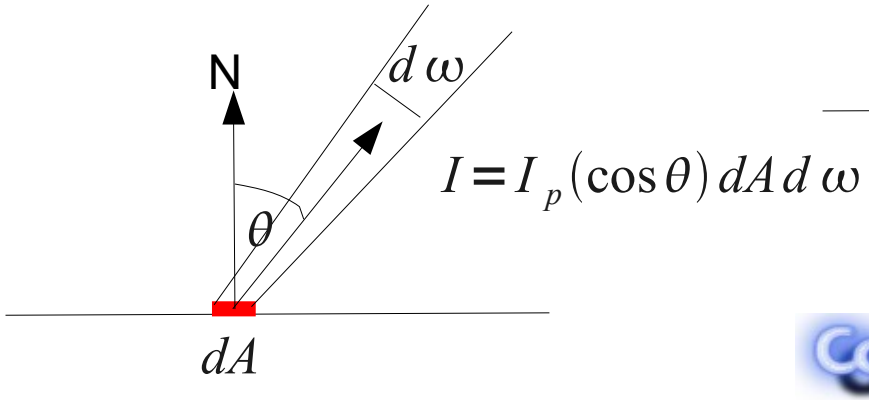
Light Sources

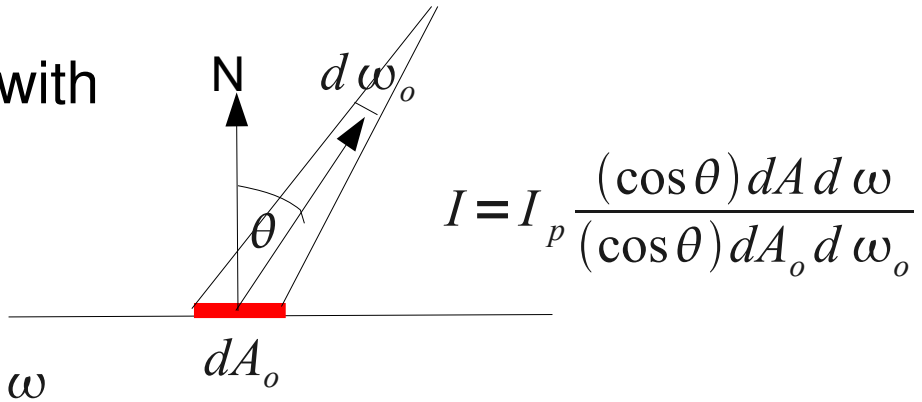
- Light sources contribute to the lighting effects and can be modeled in a number of ways including:
  - ambient
  - point light source
  - infinitely distant light source
  - radial intensity attenuation(inverse quadratic function)
  - directional light source(spot lights, angular intensity attenuation(powers of cos))
  - light surfaces

Ambient Light

- Light bounces(reflects) around a scene. So from any point in the scene there is background light moving in all directions. A simple way of modeling this is called ambient lighting.
- Ambient lighting can applied to the diffuse material properties of a surface's material.
- Ambient lighting produces flat looking images.
- Colour of a surface does not change with different viewing angles.







- To get lighting working in OpenGL you need to first turn lighting on:

```
gl.glEnable(GL2.GL_LIGHTING);
```

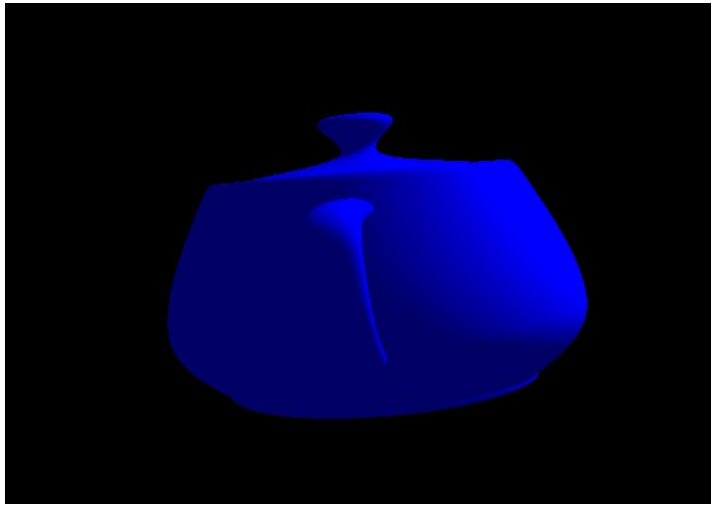
- Then you can set up and enable a light:

```
float ac[] = { 0.2f, 0.2f, 0.2f, 1.0f };
gl.glLightfv(GL2.GL_LIGHT1, GL2.GL_AMBIENT, ac, 0);
gl.glEnable(GL2.GL_LIGHT1);
```

- The material needs to be given a colour:

```
float df[] = { 1.0f, 0.2f, 0.0f, 0.0f };
gl.glMaterialfv(GL2.GL_FRONT_AND_BACK, GL2.GL_AMBIENT_AND_DIFFUSE, df, 0);
```

Diffuse Lighting



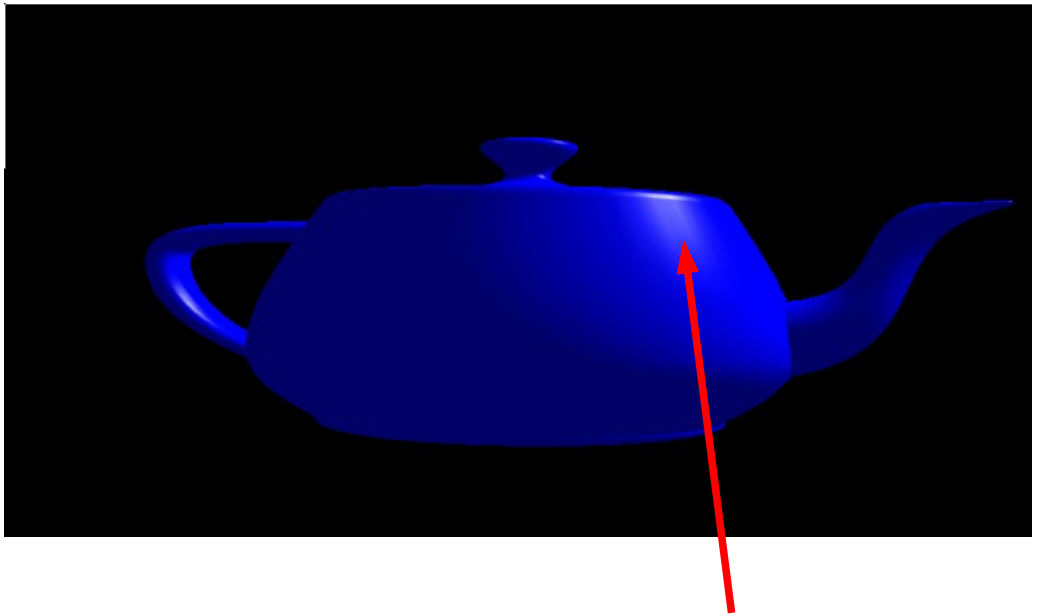
- Diffuse lighting is good for approximating rough surfaces where light is scattered in all directions. These are known as Lambertian surfaces. The viewing direction makes no difference on the intensity of the observed surface, however, the direction the angle light sources hits the surface does make a difference.
- The intensity is proportional to $\cos\theta$ of the angle the light hits the surface.

```
float f[] = { 0.0f, 0.0f, 5.0f, 1.0f };
float dc[] = { 1.0f, 1.0f, 1.0f, 1.0f };
gl.glLightfv(GL2.GL_LIGHT1, GL2.GL_DIFFUSE, dc, 0);
gl.glLightfv(GL2.GL_LIGHT1, GL2.GL_POSITION, f, 0);
```

Setting up the diffuse lighting in OpenGL. (diffuse material set in the previous slide)

Specular Lighting

- Specular lighting helps model shiny surfaces.



Setting up the light:

```
float dc[] = { 1.0f, 1.0f, 1.0f, 1.0f };
gl.glLightfv(GL2.GL_LIGHT1, GL2.GL_SPECULAR, dc, 0);
```

Setting up the material:

```
float sf[] = { 1.0f, 1.0f, 1.0f, 0.0f };
gl.glMaterialfv(GL2.GL_FRONT_AND_BACK, GL2.GL_SPECULAR, sf, 0);
gl.glMaterialf(GL2.GL_FRONT_AND_BACK, GL2.GL_SHININESS, 10.0f); // 1 - 128
```

Phong Reflection Model

- The Phong reflection model combines: ambient, diffuse, and spectral lighting effects.

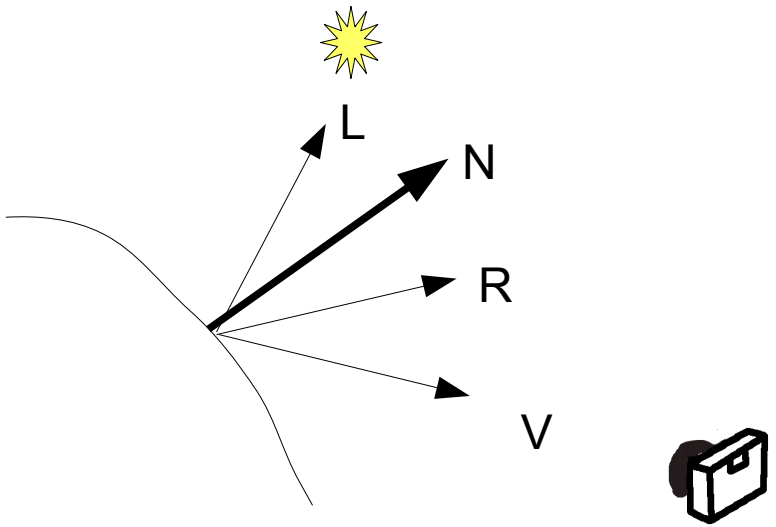
Note OpenGL uses cos of the angle between L+V and N rather than R and V.

reflection constants

shininess

$$I_p = k_a i_a + \sum_{lights} \left(k_d (L.N) i_d + k_s (R.V)^{\alpha} i_s \right)$$

lighting intensities



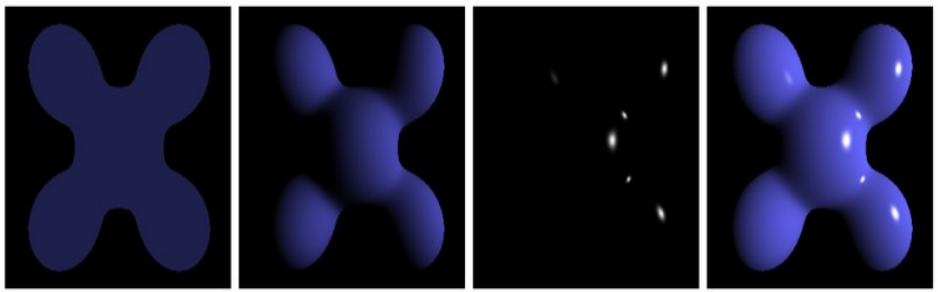
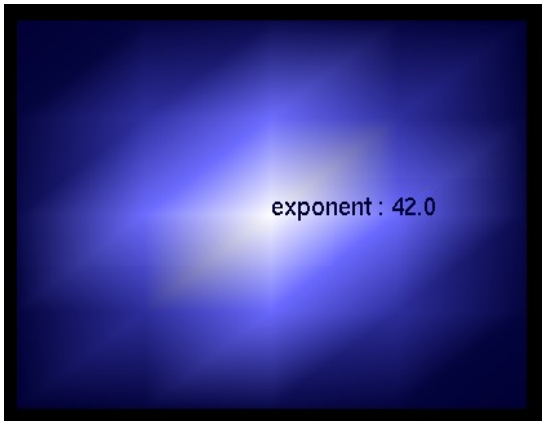


Image Brad Smith 2006 ShareALike 3.0 obtained from wikipedia

Gouraud Shading

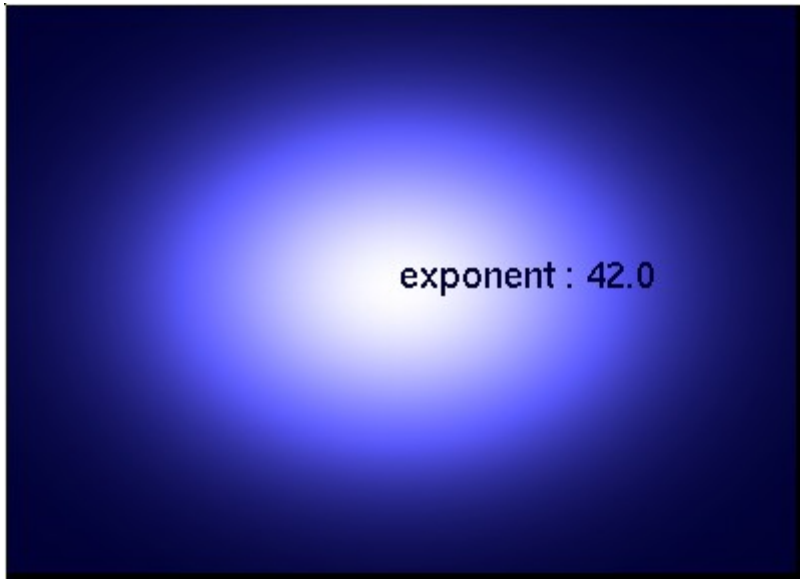
- Colours are calculated at the vertices and then interpolated across the pixels of the polygon.
- Note, the surfaces norms can be different at different vertices on the polygon.
- Advantages:
  - fast,
  - better than flat shading(entire polygon has the same colour in flat shading).
- Limitations:
  - specular highlights are not properly rendered across the polygon,
  - often require lots of small polygons.



Spot light on a tiled surface.

Phong Shading

- The surface norms are interpolated across the pixels of the polygon and than the Phong reflection model is used to calculate the colour at each pixel.
- Phong shading requires a lot more computation at each pixel(in comparison to Gouraud shading).



Spot light on a tiled surfac, with a lot more tiles then the previous image.
(should give an effect like phong shading)