CSC 240 Computer Graphics
Day 16: Lighting and Shading

Nick Howe
Smith College
Lab 8: Robotic Arm

```javascript
// set up the cube for the lower arm
lowerCube = new THREE.Mesh(cubeGeom, cubeFaceMaterial);
lowerCube.position.x = 1; // move the lowerCube over
lowerCube.scale.set(2, 0.5, 0.5);

// add it to the lower arm object
lowerArm.add(lowerCube);
lowerArm.position.x = 2; // move the "origin" of the lowerCube over

// set up the cube for the upper arm (exactly the same transformations)
upperCube = new THREE.Mesh(cubeGeom, cubeFaceMaterial);
upperCube.position.x = 1;
upperCube.scale.set(2, 0.5, 0.5);

// add the lowerArm and upperCube to the full arm
arm.add(upperCube);
arm.add(lowerArm);
arm.rotation.set(0.2, -0.4, 0); // just to get a better view
```
Q. Can you please explain the matching question in more depth?

Q. I am still confused by question above - can we go over the answers again in class?

A. Yes! I’ll do that now.

Q. I don't understand what the graphs are supposed to be. What are the axis?

A. The horizontal axis is spatial – at what point on the surface are we computing the shading? The vertical axis shows the intensity of the shading (higher = brighter)
For diffuse reflectance, the graph shows maximum brightness where the surface faces the light source. It has its lowest values where the surface is most sloped, regardless of direction. (Position of the viewer doesn’t matter)

The decoy answer (B) suggests that surfaces closer to the light are brighter, which is not true.
Q. Also, can you explain in more detail why each answer was correct for the first check up?

A. For specular reflectance (C), there are sharp peaks where the angle of the surface reflects the incoming light ray towards the viewer.

The combined graph (D) is the sum of specular and diffuse reflectance.
Q. Can you explain how we got graph A for diffuse reflectance?

A. The graph shows maximum brightness where the surface faces the light source. It has its lowest values where the surface is most sloped, regardless of direction. (Note that the position of the viewer doesn’t matter)

Q. Can you explain more about matching the graphs to the type of reflection?

A. Identify the maximal points, and the minimal.
Q. Can you explain how we got graph A for diffuse reflectance?
A. The graph shows maximum brightness where the surface faces the light source. It has its lowest values where the surface is most sloped, regardless of direction. (Note that the position of the viewer doesn’t matter)

Q. Can you explain how to get the curves for question 1 and what do those curves represent?
A. Curves show reflectance of the surface
Your Questions

Q. Can you walk us through another example similar to this question (with the graphs of the different types of reflectance)?
Q. Do we have to find the dot product every time we want to use diffuse reflection?

A. The computer will do it for us, of course. But yes: for every pixel in the image, the graphics package is computing a dot product to determine the correct shading. For every animation frame!

Q. I’m having trouble wrapping my head around how the normal vector is found at the vertices of a triangle for Gouraud shading.

A. If the model is a low-poly version of an ideal shape (like a sphere or cylinder) that theoretical shape may supply the normal vector. Otherwise, average the normal vectors for all the planes that meet.

Q. The pdf slides include some slides that elaborate on Gouraud shading and triangle fill but it wasn’t covered in the video — will we be covering that in class this year?

A. Yes. I expanded the topic and moved it to Video #15 – the slides are wrong.
Q. How do we use the Phong Reflection Model? What do you mean by putting it on top of specular reflection?

A. *The illumination from diffuse shading is added to the illumination from specular reflection.*

Q. Can you reexplain what alpha represents when calculating the specular highlight using the Phong reflection model?

A. *Alpha controls how tightly concentrated the specular reflection spot is. Smaller alpha correspond to rougher surfaces with a wide spot. Larger alpha simulate smooth surfaces, with a narrow spot.*

\[ k_S (\hat{R} \cdot \hat{V})^\alpha \]
Do we have to find the dot product every time we want to use diffuse reflection? How do we use the Phong Reflection Model? What do you mean by putting it on top of specular reflection?

I’m having trouble wrapping my head around how the normal vector is found at the vertices of a triangle for Gouraud shading.

The pdf slides include some slides that elaborate on Gouraud shading and triangle fill but it wasn’t covered in the video — will we be covering that in class this year?

Can you reexplain what alpha represents when calculating the specular highlight using the Phong reflection model?

Can you walk us through another example similar to this question (with the graphs of the different types of reflectance)?
Q. So for Phong shading, are the normals computed or interpolated? (Or is that the same thing?)
A. They are computed via interpolation.

Q. Is it possible for the color black to result from the colors washing out? I feel like it should be possible for something to be so dark that it has to be clipped in that direction, but I don't see how it would be possible mathematically.
A. Pure black is the absence of any emitted or reflected light. You cannot get any darker than that. All our shading models are based on adding light from a baseline of zero.
Your Questions

From Eleni:
http://learnwebgl.brown37.net/09_lights/lights_combined.html. used this link to visualize the ideas better

Thanks!

Q. Why do you need a material color for ambient light?
A. Shading is always a product of the light color and the material color. A red light will never illuminate a red object, whether it is ambient or not. White light will cause surfaces to be shaded in their material color.

Q. What happens when you have specular reflection but a matte surface/diffuse with a shiny object? Does one rule out the other?
A. The surface material determines the reflectance model. In Three.js, MeshLambertMaterial doesn’t even calculate the specular reflection.
Q. Is it possible to go over the vector math a little more. Maybe with a specific example.

A. Yes. *Suppose the sun is 60° above the horizon, lighting a cube. How much reflectance do we get off each face?*

\[
\text{Pixel color} = k_d \times (\text{material color}) \times (\text{light color}) \times [(\text{light direction}) \cdot (\text{surface normal})]
\]

Suppose these are all 1

Use cosine to compute this

\[
\cos 30° = 0.866
\]

\[
\cos 60° = 0.500
\]

*The side face is only 0.500/0.866 = 58% as bright as the top.*
Your Questions

Q. Why are the length of the normal and incidence vector 1? I'm a bit confused by the mathematical equations.

A. We use **unit vectors** when we want to represent a 3D direction. All we need to know is the cosine between the rays, which is conveniently given by the dot product of the unit vectors.

Q. For Gouraud shading, can you change how much the light changes when interpolating between vertices (such as changing how "steep" a rounded surface, like a sphere vs a narrower egg-shape)? Also, since the graphics simulate real objects, why/how would an empirical method work if there's no actual basis?

A. A rounded surface will be composed of multiple polygons. Each will perform its own Gouraud shading computation based on its angle. We may look a little later at **physics based rendering**.
Dot Product

- Normal unit vector $\hat{N} = (n_x, n_y, n_z)$
- Incidence vector (points to light) $\hat{I} = (i_x, i_y, i_z)$
- Angle between is $\theta$
- Dot product:

$$\hat{N} \cdot \hat{I} = n_x i_x + n_y i_y + n_z i_z$$
$$= \|\hat{N}\| \|\hat{I}\| \cos \theta$$
$$= \cos \theta$$
Q. Please explain the answer to the previous question "Which color component(s) are washed out?"

A. Color values greater than 255 cannot be stored or displayed. So if multiple lights added together give more than 255, it is “washed out”

Q. Could you explain more about the difference between interpolating colors and normals?

A. They are actually quite similar. Both have three components, and you interpolate each component separately.
Your Questions

Q. What do you mean by right hand rule for normal surface vector is perpendicular?

A. The normal point perpendicularly out from the surface, on the “up” side as specified by the right hand rule.

Q. Can you give an example of what behavior requires additional flags?

A. We’ve already seen flatShading. Others include visible, transparent, and flags we’ll see later like fog, clipShadows, etc.

```javascript
// Creates a material for the pyramid that is "matte" not "shiny".
var pyramidFaceMaterial = [
    new THREE.MeshLambertMaterial( { color: 0xffffff, flatShading: true } ),
    new THREE.MeshLambertMaterial( { color: 0x999999, flatShading: true } ),
    new THREE.MeshLambertMaterial( { color: 0xff9999, flatShading: true } ),
    new THREE.MeshLambertMaterial( { color: 0xff9999, flatShading: true } )
];
```
Lab 9: Lighting a Sphere