

CS 770/870

Assignment 5: Lighting, Material Properties, and Texture

October 29, 2008

Due: Monday, 11/10 by 23:59:59

Last allowed submission: 11/14 23:59:59

The purpose of this assignment is to give you practice with OpenGL lighting, material properties, and texture mapping facilities. For this assignment, you will implement a graphical user interface (GUI) that allows a user to modify material properties and textures of objects in a scene. Furthermore, your GUI will allow the user to change lighting properties of the scene and see the effect they have on how the scene is rendered.

1. **Lighting.** Your application will have two lights of different types.

- Directional lighting. You will have one light source that is a directional light, similar to the sun. You should provide two scrollbars that allow the user to change the direction the light is coming from. One scrollbar changes the elevation angle of the light (example settings: from ground level like at sunset, from straight above like at noon in the summer, etc.). The other scrollbar changes the heading direction of the light (example settings: from the northeast, from the west, from 231° , etc.). If you cannot get this working, three scrollbars that go from 0 to 1.0 in x, y, and z will get you partial marks.
- Point source lighting. You will have one light source that is a point source, similar to a light bulb. You should provide three scrollbars that allow the user to move the light around the scene in x, y, and z. You should also provide a way to change the color and intensity of the light.

2. **Material Properties.** Your application should define at least two different material properties that can be assigned to different objects in the scene. You should implement this by creating a class to represent material properties, and you should have a way in your GUI to assign different instances of this material class to shapes. (See the Texture class in Demo 6 for a model of how you might go about structuring this class.) Constraints on the two required materials are as follows:

- Shiny material. You must have a material that is highly reflective of a color other than a shade of white. In other words, it should have a slight hue such as red or copper or blue, etc.
- Dull material. You must have a material that is not specularly reflective, but does have enough diffuse reflection that you can easily see the effect of lighting on shapes that take on this material property.

Any other materials you add can be whatever you like, but be sure to indicate which ones are the required shiny and dull materials.

3. **Textures.** Your application should use at least two different textures that can be assigned to different objects in the scene. You should have a way in your GUI to assign different textures to shapes. Constraints on the two required textures are as follows:

- Clamped non-modulated image. You must have a texture that does not repeat when texture coordinates go outside the bounds $[0, 1]$ of the texture. It should not be affected by the underlying lighting. This image should probably be a photograph or a rendering so it is easy to distinguish it from other types of textures. (Examples where this might be used in a real application: television/projection screens, picture-in-picture video, mirrors/reflections)

- Repeating modulated texture. You must have a texture that repeats when texture coordinates go outside the bounds $[0, 1]$ of the texture. This image should blend with the underlying material properties, allowing them to be seen along with the texture. This image should probably be a pattern such as brick, wood, or leopard skin, so it is easy to distinguish from other types of textures.

Any other textures you add can be whatever you like, and act any way you like, but be sure to indicate which ones are the required ones, and which one is which.

4. **Viewing.** You must use a reasonable perspective view, and you must provide GLUI controls for scaling the scene and rotating it about the vertical axis. You can (and should) limit the amount the user can scale the scene, as long as it is still possible to zoom out to see a fairly large scene.

5. **Predefined scene.** You must display a predefined scene that contains several different solids (and potentially some flat faces). The scene must include objects that approximate curved surfaces so that they can illustrate lighting and material properties well. These objects must also have correct normals to effect good lighting. You must have at least one object that specifies texture coordinates, some of which are outside the range $[0, 1]$ (the place this happens must be readily visible, and the object must be easily identified in your GUI, perhaps by naming it “textureMe” or something). You must also have at least one object that shows the clamped image well—if you do it right, these objects can be one and the same.

Point allocation

- 15 Predefined scene visible in a perspective view with the ability to rotate and scale the scene from the gui.
- 10 Directional lighting is implemented, and shape surfaces respond properly to it.
 - 5 A user can control the lighting direction from the gui.
- 10 A point-source light is implemented, and shape surfaces respond properly to it.
 - 5 A user can control the position of the point-source light from the gui.
 - 5 A user can control the color and intensity of the point-source light from the gui.
- 10 A shiny material can be assigned to any shape, and it responds properly to light.
- 10 A dull material can be assigned to any shape, and it responds properly to light.
- 10 A clamped, non-modulated image can be assigned to at least two different shapes, and lighting does not affect it. On at least one shape, the image does not cover the entire object.
- 10 A repeating, modulated texture can be assigned to at least two different shapes, and lighting and material properties affect how it appears. On at least one shape, the image must obviously repeat itself to cover the object.

Features that might qualify for the last 10 points include but are not limited to doing different effects with lighting, material properties, and textures other than those required. Be sure to document anything new that you do so we know to test it.

Note:

Points are earned by correctly implementing features robustly. Points are deducted for bugs, incorrect implementation, poor style, poor design decisions, etc.