

COMP 3009: Assignment 2

Posted: March 11th, 2021 Due by 23:59 Sunday, March 21

Assignment Objectives:

- a. Familiarization with lighting models
- b. Familiarization with light sources
- c. Familiarization with the type of required matrices for the transformation
 - i Vertex position for lighting computation
 - ii Vertex position required by OpenGL (glposition)
 - iii Vertex Normal and its transformation
 - iv Output colour computation (glFragcolor)

Grades:

- 1. Assignment total marks: 100.
- 2. You can receive a 20 points bonus if you correctly complete the bonus section.

Submissions

- a. Code (VC++). Put the code related to each part in a folder.
- b. Report – a comparison report

1 Setting up the space (10 points)

To do:

- 1. Create a new project (you can use a project or code from previous assignments or tutorials).
 - 1.1 Create three spheres close to each other so that you will be able to compare the results of the light models. For example positions the spheres at position 100, 0, 100 and at position 140, 0, 100. The radius of the spheres should be 10 or so, in order to be able to see the effect of the light models.
 - 1.2 Colour the spheres red.

- 1.3 Set up a camera view position so that you can see the spheres. (e.g., 200,200,200) and set up the transformation matrix so that you can see the sphere.
- 1.4 Set up perspective projection (e.g., symmetric).
2. Camera controls:
 - 2.1 Pitch operation – the *up* and *down* arrow keys will rotate the camera around the x-axis by +1 degree and -1 degree respectively.
 - 2.2 Yaw operation – the *left* and *right* arrow keys will rotate the camera around the y-axis by +1 degree and -1 degree respectively.
 - 2.3 Roll operation – the “a” and “d” keys will rotate the camera around the z-axis by +1 degree and -1 degree respectively.
 - 2.4 Forward motion – the “w” key will move the camera forward by 1 unit along the “look at” vector.
 - 2.5 Backward motion - the “s” key will move the camera backward by 1 unit along the “look at” vector.
3. Display the spheres.

2 Create the Gouraud light model (10 points)

To do:

1. Create the Gouraud light model shader program
2. Use the Gouraud light model shader to render the sphere at position (100, 0, 100)
3. Use the Object material
 - 3.1 Material data is -
 - 3.1.1 Ambient (0.2, 0.2, 0.2)
 - 3.1.2 Diffuse (0.75, 0.75, 0.75)
 - 3.1.3 Specular (0.8, 0.8, 0.8)
 - 3.1.4 Level of shininess is 5 (allow user to change the level of shininess by)
 - (+) increases the level of shininess by 2
 - (-) decreases the level of shininess by 2 (note minimum should be 0)
4. Light source
 - 4.1 Light source is white light
 - 4.2 Light position is at (240, 210, 200)

3 Create the Phong light model (30 points)

To do:

1. Create the Phong light model shader program

2. Use the Phong light model on the sphere at position (140, 0, 100)
3. Use the Object material
 - 3.1 Ambient (0.2, 0.2, 0.2)
 - 3.2 Diffuse (0.75, 0.75, 0.75)
 - 3.3 Specular (0.8, 0.8, 0.8)
 - 3.4 Level of shininess is 5 (allow user to change the level of shininess by)
 - 3.4.1 + – increases the level of shininess by 2
 - 3.4.2 - – decreases the level of shininess by 2 (note minimum should be 0)
4. Light source
 - 4.1 Light source is white light
 - 4.2 Light position is at 240, 210, 200

4 Light Models controls (20 points)

Controls

1. Turning lighting model components on and off
 - 1.1 M – turn ambient light off
 - 1.2 m – turn ambient light on
 - 1.3 N – turn diffuse light off
 - 1.4 n – turn ambient light on
 - 1.5 B – turn specular light off
 - 1.6 b – turn specular light on
2. Scaling – enable scaling in the x and y directions using the i,j,k,l keyboard strokes –
 - 2.1 Pressing on i means increment the scale in the x-direction by 0.5 increments;
 - 2.2 Pressing on j means decrements the scale factor in the x-direction by 0.5. Min scale is 1;
 - 2.3 Pressing on k means increment the scale in the y-direction by 0.5 increments;
 - 2.4 Pressing on l means decrements the scale factor in the y-direction by 0.5. Min scale is 1;
 - 2.5 Pressing on r resets the scale in the x,y,z directions to 1.

5 Compare the lighting models(10 points)

Submit with the code project a short report about the two models:

1. Compare the two models (the Gouraut and the Phong models) with respect to, performance, programming efforts,
2. Compare the two models and state your opinion:
 - 2.1 Which model has better visualization? Why?

- 2.2 Add images (snapshots) of the spheres from your assignment to demonstrate the differences.
- 2.3 Which of the models would you choose when factoring in programming efforts, computation performance, quality of results, etc. Justify your answer.
3. Show the effect of correctly manipulating the normal when scaling the rendered objects.

6 Using half vectors (20 points)

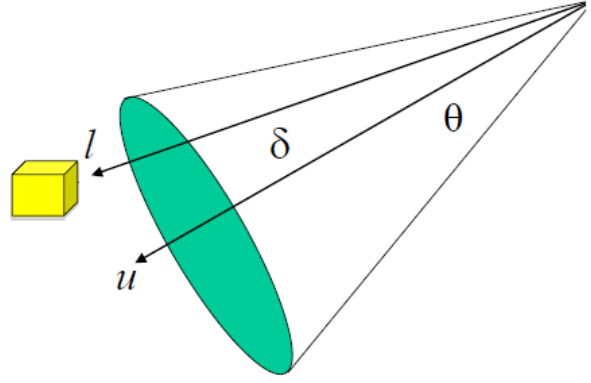
Modify your code in Tasks 2 and 3 and to show the models results when half vectors are used.

1. Controls. Switch to half vectors by using:
 - 1.1 H – switch to half vector
 - 1.2 h – switch back to normal view
2. Compare the results to those obtained in Tasks 2 and 3
 - 2.1 Are you satisfied with the results
 - 2.2 Add comparisons between the models.

7 Bonus – Create another light source (20)

1. Create a second light source, which is a spot light. Figure shows a spot light. In a spotlight, light intensity is proportional to $\cos(\delta)$, where δ is the angle between light ray, l , and cone centre vector, u . The specular intensity will be affected by angular attenuation. You can calculate the color of the object using formula in the figure.
2. The spot light should be positioned at (190, 200, 190)
3. The light cone angle should start at 35 degrees. Allow the user to control the light cone by pressing
 - 0 – to increase the cone opening angle in increments of 1 degree. Max angle is 120.
 - 9 – to decrease the cone opening angle in increments of 1 degree. Min angle is 1
4. Allow the user to control the light focus intensity (angular attenuation)
 - 8 – to increase the light angular attenuation in increments of 3.
 - 7 – to decrease the light angular attenuation in increments of 3. Min value is 1 or 0
5. The spot light source should be aimed towards the centre of the box (use the sphere)
6. Time permitted add radial attenuation ($1/(ar^2 + br + c)$ for $(a = 4, b = 0, c = 1)$)

Use an attenuation power a $\cos(\delta)^a$
 $f_{ang_atten} = (V_{obj} \cdot u)^a = \cos(\delta)^a, 0 \leq \delta \leq \theta$



$$Colour = I_a * L_a * k_a + I_d * L_d * k_d + I_s * L_s * k_s$$

$$Colour = I_a * L_a * k_a + \max(L \cdot N, 0) * L_d * k_d + (\max(V \cdot R, 0))^{n_s} * (\max(u \cdot V_{obj} - \cos(\theta), 0))^a * k_s$$

Figure 1: Spotlight Angular Attenuation