

Cosc 4370 Homework 3 Report

1 Introduction

The assignment requires us to write the vertex and fragment shader programs according to Phong's lighting for rendering a cube. The cube is lit by a light source. We have to also complete the Camera class and define the view and projection matrix for the vertex shader program.

2 Method

I first completed the GetViewMatrix for the Camera class, then defined the projection matrix. Finally I wrote the shader programs. I used the LearnOpenGL website to learn about Phong's lighting principles.

3 Implementation

31 GetViewMatrix

I used the glm::lookAt function to define the view matrix of the camera. Since we already know the front vector and up vector of the camera object, we can just plug those variables into the function to get the resulting view matrix.

32 ProjectionMatrix

I used the glm::perspective function to get the projection matrix for the camera because we want to get a perspective projection for the camera.

33 Vertex Shader

After consulting with the LearnOpenGL website, I set the gl_position to the result of multiplying the position vector with the model matrix, view matrix, and projection matrix in that order. The fragPos is the position of the fragment in model space so I calculated that by multiplying the position vector by the model matrix. The normal vector can be passed as is to fragment shader.

34 Fragment Shader

First I only used the diffuse and specular color to shade the cube, but the scene is really dark where the light source doesn't reach, so I added an ambient light modifier to lighten up the scene more. The final result looks closer to the result that I needed to have.

4 Result

The cube is lit from a light source and has specular highlights.



