Lab 1

In this Lab we are going to create a simple fog effect…

* Create two new shaders (fog.vert & fog.frag) and a new shader object

The code for the vertex shader is as follows, this code simply passes the vertex positions and normal to the fragment shader.

#version 400

layout (location = 0) in vec3 VertexPosition;

layout (location = 2) in vec3 VertexNormal;

uniform mat4 transform;

out vec3 v\_norm;

out vec4 v\_pos;

void main()

{

v\_norm = VertexNormal;

v\_pos = vec4(VertexPosition, 1.0);

gl\_Position = transform \* vec4(VertexPosition, 1.0);

}

* In the fragment shader declare the following variables...

#version 400

out vec4 FragColor;

in vec3 v\_norm;

in vec4 v\_pos;

uniform vec3 fogColor;

uniform mat4 u\_pm; // uniform\_ProjectionMatrix

uniform mat4 u\_vm; // uniform\_ViewMatrix

uniform float maxDist; //fog max distance

uniform float minDist; //fog min distance

* Set up the fog effect in the fragment shader, here is the code...

void main()

{

float dist = abs( v\_pos.z );

float fogFactor = **calculate from today’s lecture.**

fogFactor = clamp( fogFactor, 0.0, 1.0 );

vec3 lightColor = vec3(0.1,0.1,0.1);

vec3 color = mix( fogColor, lightColor, fogFactor);

FragColor = vec4(color, 1.0);

}

How it works...

The fog variables contain the parameters that define the extent and color of the fog. The field minDist is the distance from the eye to the fog's starting point, and maxDist is the distance to the point where the fog is maximal. The field fogColor is the color of the fog.

The variable dist is used to store the distance from the surface point to the eye position. The z coordinate of the position is used as an estimate of the actual distance. The variable fogFactor is computed using the preceding equation. Since dist may not be between minDist and maxDist, we clamp the value of fogFactor to be between zero and one.

* Now create a method in the application that sets the values of the uniform variables. I used the following values.

Fog.setVec3("lightDir", glm::vec3(0.5, 0.5, 0.5));

Fog.setMat4("u\_vm", myCamera.GetView());

Fog.setMat4("u\_pm", myCamera.GetProjection());

Fog.setVec3("fogColor", glm::vec3(0.2, 0.2, 0.2));

Fog.setFloat("minDist", -5.0f);

Fog.setFloat("maxDist", 5.0f);

* Run the code. If you have no errors you will notice that the model has changed colour but the colour is static. Why is the colour static? This is the same reason that the toonShading and rimShading is also static, while it could be argued that a static effect suitable for toon shading it is not really suitable for rim shading and certainly not suitable for fog. Update the code so the fog effect is no longer static, hint... think about the z values, i.e.
  + How have we passed the z values to the shader
  + How do we update them for model movement
  + How do we update them in the shader