

Coursework Report

Bradley Jones
40167459@live.napier.ac.uk
Edinburgh Napier University - Module Title (SET00000)

Abstract

The purpose of was to develop a 3D scene that demonstrate understanding of these features. At a minimum, textures and lighting will be implemented to add an element of realism.

Keywords – Computer, Graphics, Napier, Pyramid, OpenGL, Edinburgh, Desert

1 Introduction

My scene known as "Pyramids" was chosen as a coursework task because of the opportunities it allowed in creating a visually distinct scene, as well as personal interest in ancient Egypt. Currently, the coursework was focused on adding the appropriate objects and textures to create a scene reminiscent of a ruin near the Great Pyramids of Giza.



Figure 1: **Pyramids of Giza** - Photograph of the real world pyramids, used as reference material.

{Wikipedia}.

2 1 Overview

The project was created in Visual Studio 2015 and has been focused on the creation of objects, texturing them and the addition of light sources. It is hoped that lights and normal mapping will be implemented in the near future.

During the development of the current scene many other ideas were taken into consideration, for example a woodland scene, however in the end a desert scene inspired by the pyramids seemed like it would be the most unique setting for a OpenGL project.

3 2 Methods/Techniques

The project uses a great variety of different effects in an effort to create a aesthetically pleasing scene.

One technique that was used to great effect was texturing. This is a critical technique as poor textures can severely hamper the realism of a 3D project, while good textures can greatly enhance a scene. An example of this is the use of a appropriate pyramid brick texture on the objects. As it makes it look like the bricks are getting smaller and smaller as they reach the apex of the structure, as well as giving off the impression of depth in the object without the use of normal mapping.

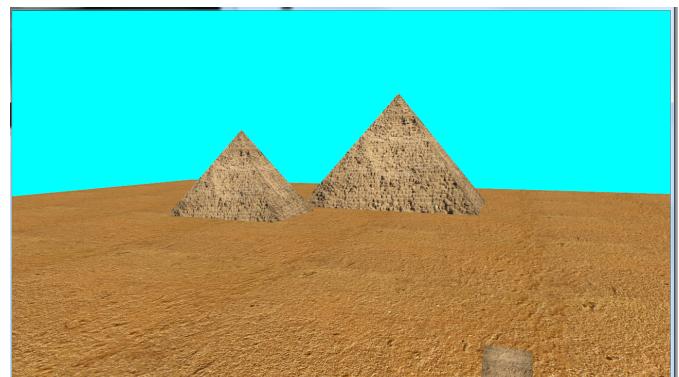


Figure 2: **Example of Texture Use** - A work in progress screen shot of the scene.

Another technique that was used to great effect was the use of Point Lighting. Lighting is a critical part of the creation of a OpenGL scene, as it allows users to show off their objects and post processing techniques.

Illustrated below is a short section of code from the .cpp file, showing this

Listing 1: The section of code used for creating the point light

```

1
2 // Set lighting values, Position (-25, 10, -10)
3 light.set_position(vec3(-10.0f, 20.0f, -30.0f));
4 // Light colour white

```

```
5 light.set_light_colour(vec4(1.0f, 1.0f, 1.0f, 1.0f));
6 // Set range to 20
7 light.set_range(2000);
8 // Load in shaders
9 eff.add_shader("shaders/point.vert", GL_VERTEX_SHADER);
10 eff.add_shader("shaders/point.frag", ↵
11     GL_FRAGMENT_SHADER);
12 tex_eff.add_shader("27_Texturing_Shader/simple_texture.vert", ↵
13     GL_VERTEX_SHADER);
14 tex_eff.add_shader("71_Greyscale/greyscale.frag", ↵
15     GL_FRAGMENT_SHADER);
16 eff.build();
```

4 Conclusion

In conclusion, at this moment in time, the scene is working mostly as expected, however it is missing some key features that hopefully will be added soon.