Coursework Report

Svetlozar Georgiev
40203970@napier.ac.uk
Edinburgh Napier University - Computer Graphics (SET08116)

Abstract

The aim of this project is to create a 3D scene in OpenGL, using key graphical rendering principles. A renderer framework was used in order to create effects and reduce development time. A number of graphical techniques were implemented such as lighting, texturing, cameras. The resulting scene is a moonlit eerie house, surrounded by dead trees. The inspiration for the project came from the cartoon Courage the Cowardly Dog [1].

Figure 1: **Courage the Cowardly Dog** *Inspiration for the project*

Keywords – Phong shading, spot light, texturing, lighting, object transformation

1 Introduction

Main effects Multiple different effects were implemented in this project, the key ones being:

- Shading
- Lighting
- Texturing
- Post processing
- Transformations
- Optimisation
- Post processing

Shading Shading is a method which adds depth perception to 3D models to make them look more realistic. The direction of the light as well as the direction an object is facing are taken into account. Parts of an object facing directly the light source are more lit, hence why they appear brighter. On the other hand, parts facing away from the light source receive less light and are therefore darker.

Lighting Lighting refers to the technique of simulating light sources which affect certain aspects of the objects on the scene such as their colour and reflections.

Texturing Texturing means applying an image to the surface of an object which allows the addition of more detail. What is more, it creates the illusion of materials when used in conjunction with lighting and shading.

Transformations Three types of object transformations were used in the project:

- Scaling
- Rotation
- Translation

Scaling an object refers to increasing or decreasing its size.

Rotating an object changes which way it is facing.

Translating an object changes its position in the world.

Optimisation Optimisation aims to improve a scene's frame rate and build time. As the scene becomes more complicated, frame rate problems can be noticed. This is why optimisation is necessary.

Post processing Post processing allows the application of an effect on the whole scene more easily.

Motivation The motivation behind this project is the application of skills developed from undertaking the Computer Graphics (SET08116) module and gaining further knowledge in the field of graphics rendering by creating a 3D scene.

Limitations While creating the project, a number of problems were encountered, the main one being the lack of resources online. OpenGL is widely used, however many different versions exists and often information found on the internet was misleading. Ultimately, the time constraint turned out to be the biggest limitation.

2   Related work

While developing the project, the main source used was the SET08116 Workbook [2]. Most methods and practices can be found there or were inspired by it.

3   Implementation

The following key techniques were implemented in the project

- Phong shading
- Spot and point lighting
- Texturing
- Transformations
- Free camera
- Normal mapping
- Post processing
- Environmental mapping and tarnishing
- Particle effects and billboarding
- Code optimisation
- Skybox

Lighting   The light sources used in the project are a spotlight (Figure 2). It is place where the moon is and it lights the scene with blue light as if it were moonlight. The equations used to calculate the necessary values for the spot light are as follows: *Equation 1* is used to calculate spot light intensity using the direction of the light, the direction to the light and the power of the light. *Equation 2* is the calculation of the attenuation factor using the attenuation values. *Equation 3* provides the final colour.

$$I = max(-R.L, 0)^p \tag{1}$$

$$A = c + l * d + q * d^2 \tag{2}$$

$$C = (I/A) * L \tag{3}$$

where *C* is the final colour, *I* is the light intensity, *A* is the attenuation factor and *L* is the light colour. The resulting colour is then used in the Phong shading equation. Phong shading is implemented in conjunction with the spot light to create the effect of objects having depth. There are two additional point lights in front of the house which light the surrounding area.

Phong shading   Phong shading (Figure 3) refers to the principle of colouring different faces of objects depending on the texture and light levels. The colour of every pixel is calculated depending on the light direction, location and colour as well as material properties. *Equation 4* shows the calculation of the pixel colour.
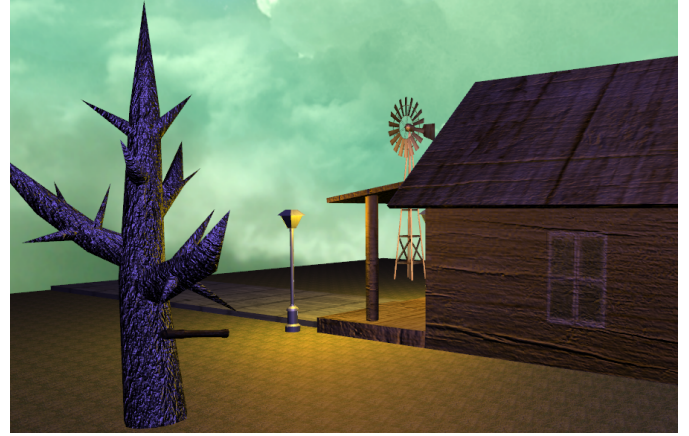
$$C = (E + D) * T + S \tag{4}$$

Figure 2: **Spot light** *Moonlight effect with spot light*

where *C* is the colour of the pixel that is to be calculated, *E* is the emissive reflection of the material, *D* is the diffuse component, *T* is the colour of the texture at that pixel and *S* is the specular component.
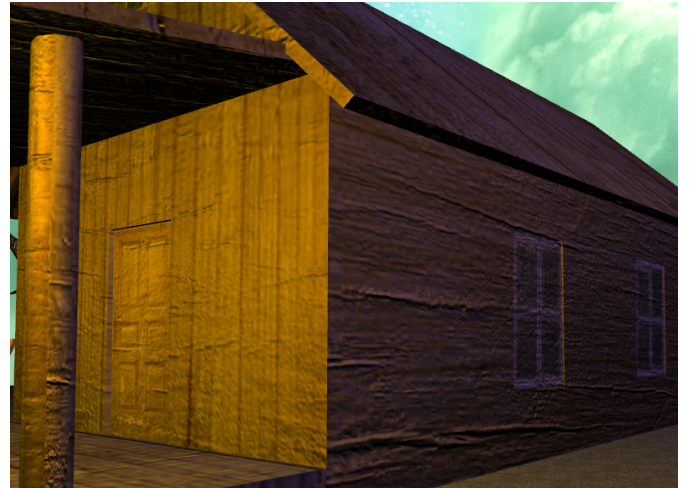

Figure 3: **Phong shading**

Free camera   A free camera was implemented allowing free movement around the scene. The controls for the camera are the mouse, used to change the view direction, and the keyboard to move the camera.

Normal mapping   Normal mapping is a technique which creates more realistic surfaces of the objects by creating bumps or dents. Normal maps are commonly RGB images where the RGB components correspond to the X, Y, and Z coordinates, respectively, of the surface normal. An example is shown in Figure 4.

Post processing   Post processing is used for effects which affect the whole scene. Normally, the effect would have to be applied to each object, however when using post processing this is not necessary. Instead, the whole scene is rendered in a frame buffer which can then be saved as a texture. Effects can then be applied on the

2

Figure 4: **Normal mapping** *Effect of tree bark having bumps and dents*

texture. Motion blur and a sepia filter were used in the project.

Motion blur occurs when the camera moves. The information of what the scene looked like in the previous frame is merged with the current frame. This creates a distorted effect as if the viewer were passing by the objects quickly. An example can be seen on Figure 5.

The sepia filter colours the whole scene in a reddish-brown colour. The effect is achieved by rendering the whole scene in the frame buffer and storing it as a texture. The texture is then sampled using texture coordinates. The final colour is calculated by scaling the original colour with the RGB values of the filter. *Equation 5* shows the calculation of the final colour.

$$C_f = C * (0.314_r, 0.169_g, -0.090_b) \qquad (5)$$

Environmental mapping and tarnishing Environmental mapping and tarnishing were combined to create the effect of an object looking like a moon.

Environmental mapping is a technique used for creating reflective surfaces. A sphere's normal and texture coordinates are used to apply a texture which creates the effect of reflections.

Tarnishing is the effect of blending two textures. An image of the moon's surface is applied to the reflective sphere created with environmental mapping. The result can be seen on Figure 6.

Particle effects and billboarding A particle system was implemented in the project. Particles are small objects used to create effects which require a large number of small sprites. Particles are created above the scene with random X and Y coordinates. They move down until they reach the ground. In conjuction with billboarding, which is when an object is always facing the camera, the effect of rain is created.

Figure 5: **Motion blur**

Code optimisation Some basic code optimisation was done. Matrices, such as the view and projection matrix, are precalcuated as they are the same for every object that is rendered. What is more, the number of "gl" calls was reduced by storing uniform variable locations in variables before rendering.

Skybox A skybox is a cube in which the scene is placed. Textures are then applied to each of its' faces to create the effect of an environment surrounding the scene.

4 Future work

Although a number of key graphics principles were implemented, the project still has room for improvement. The scene could have benefitted from more animated objects. Due to the lack of lighting, shadowing was decided not to be implemented. The addition of more lights or a day and night cycle could make shadowing essential.

5 Conclusion

To conclude, this project employs a number of basic graphical techniques in order to create a realistic and visually pleasing scene using OpenGL. Although the result
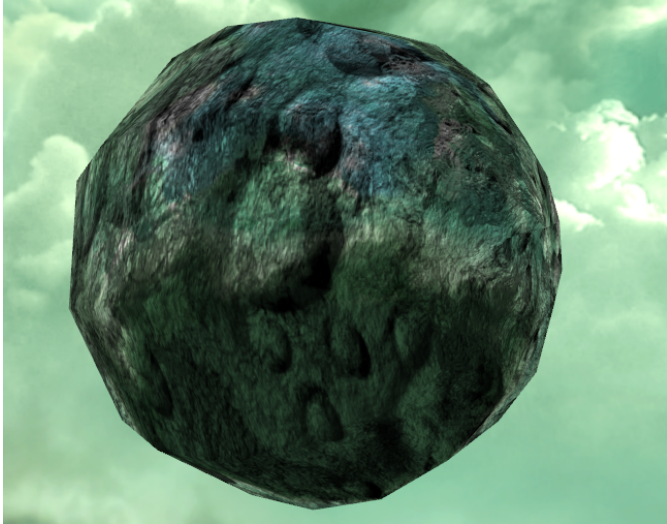
Figure 6: **Moon** *Moon created with environmental mapping and tarnishing*

is satisfactory, implementing more features would certainly improve the project's aesthetics and effects.

References

[1] John R. Dilworth, *Courage the cowardly dog*, 1999.

[2] Kevin Chalmers and Sam Serrels, *SET08116 Computer Graphics Workbook 2016/17*, 2016.