Edinburgh Napier UNIVERSITY

Coursework Report

Stefan Hristov
40284739@live.napier.ac.uk
Edinburgh Napier University - Computer Graphics (SET08116)

Abstract

The goal of the second part of the coursework was to develop our project further and bring it to shine. Unfortunately, I believe my space scene was not suitable for much further development. In my new scene there is a sniper rifle which is better suited to display the different material specular reflection, ambient occlusion, visible shadows and Fresnel effect. Up to this point I managed to only implement to some extend the ambient occlusion.

1 Introduction

For the scene I used a Phong shading model, which is a fragment based shading model, but also incorporated a texture, specular map, normal map and an ambient occlusion map. That way the model looks more alive and has details such as 3d feeling due to the normal map and more realistic ambient texture. The use of ambient occlusion map is more of a solution for the problem that actual occlusion would likely be calculated with the help of ray tracing. The difficulty with ambient light is that in the real world it is very complex calculation of light bouncing around from different objects which is almost impossible to render in real time.

2 Implementation

2.1 Ambient

The implementation is fairly straightforward. For the Phong shading model we need the ambient, diffuse and specular component. The ambient colour is computed by multiplying the material's diffuse reflection (its colour) with the colour of the light for which we are doing the calculation. In my work, I multiplied the result with the texture sample in order to get the colour with respect to the texture. Then I multiplied this result with the ambient occlusion sample in order to give this slight difference in colour in the dark portion of the object where the light doesn't reach and otherwise is coloured by our artificial ambient light.

2.2 Diffuse

For the diffuse component I calculated the diffuse component by taking the maximum between the dot product of the normal and the light direction of the fragment, and 0, since we do not want to have a negative component. Then multiplied the diffuse component by our ambient component (the diffuse reflection of the material with the light colour). Finally multiply that with the texture sample once again in order to apply the actual texture to the model.

2.3 Specular

For the specular component I first calculated the half vector of the fragment. Then took the maximum between the dot product of the normal and the half vector, and 0, and then raised it to the power of the shininess index of the material. Followed by multiplication with the product of the specular reflection of the material and the light colour. At the end I applied the specular map by multiplying the specular component with the specular map sample. For the final colour of each fragment the three components were added together.

2.4 Code

Listing 1: The implementation in my fragment shader

```
1  // Calculate ambient
2  ambient = mat.diffuse_reflection * light.light_colour;
3  ambient = ambient * tex_sample;
4  ambient = ambient * occlu_sample;
5
6  // Calculate diffuse
7  float diffuseK = max(dot(normal, light.light_dir), 0.0f);
8  vec4 primary = diffuseK * (mat.diffuse_reflection * light.↩
     light_colour);
9  diffuse = mat.emissive + primary;
10 diffuse = diffuse * tex_sample;
11
12 // Calculate specular
13 vec3 half_vector = normalize(light.light_dir + view_dir);
14 float specularK = pow(max(dot(normal, half_vector), 0.0f), ↩
     mat.shininess);
15 specular = specularK * (mat.specular_reflection * light.↩
     light_colour);
16 specular = specular * spec_sample;
17
18 // Phong = ambient + diffuse + specular
19 result = ambient + diffuse + specular;
```

3 Future Work

I still need to implement the shadows and get rid of the shadow artifacts. Implement Fresnel into the computation and also try to calculate real-time ambient occlusion by casting rays.

4 Conclusion

In the end I ended up with much less than I planned to produce. I am happy with how the occlusion turned out even though I could not implement my initially planed ray trace

occlusion. Due to time constrain I did not get to write the shadows and the Fresnel for this project. An excellent way to improve my scene is to add more objects and more lights, cameras to it but than again, time was limited.