**IGB381 Game Engine Technology -** Semester 2 2017

**Report on Multiple Light Shaders – Assignment 1**

**For:** Executive Producer ofSpigot **-** Steve Binkman **By:** Michael Cartwright

1. **Description of the Shader Techniques**

Texture Mapping and Normal Mapping has been applied to the custom shaders. This allows for Objects in the scene to have a color and normal textures. While discussed and code briefed in Workshops of IGB381, the source for this information was found in a unitycookie video on YouTube [3] which provided new understanding with scaling these textures.

Texture Anti-Aliasing was attempted based on Ross Brown’s video [2]. This lead to Trilinear filtering and a high Anisotropic value for all textures. However, the Unity engine appears too already apply these effects.

Bump Mapping has been applied to the custom shaders. This allows for bump/rough effects to be applied to the Object. The source for this information was found in a unitycookie video on YouTube [5].

Specular Lighting has been applied to the custom shaders. This allows for bright light segments to appear on Objects.

Diffuse Lighting has been applied to the custom shaders. This allows for the diffuse effect to occur after the Specular lighting effect providing a realistic lighting effect.

Ambient Lighting has been applied to the custom shaders. This allows for a darker effect to take place on an Object after the diffuse lighting effect.

Rim Lighting has been applied to the custom shaders. This allows for light to appear around the edge of an Object as if it was a back light. This effect was discussed in the unitycookie video on YouTube in detail and then implemented [7].

Light Attenuation has been added to the custom shaders. This allows for the attenuation of a light source to change based on distance.

Multiple lights were achieved using two passes. The first pass for the Sun directional light. The second pass for spotlights. The use of Blend One One and ForwardAdd was discussed from another unitycookie video [4].

Spotlight Cookie Attenuation has been applied to the custom shaders. Originally had Light blocks transposing and directing themselves towards the ThirdPersonCamera. This solution was found here [1]. This allows for crisp light projections to occur.

1. **Description of the Parameters**

**Spotlight\_1.cs and Spotlight\_2.cs**

For Spotlight\_1.cs and Spotlight\_2.cs files. The parameters available to be tweaked are speed and angle.

A higher speed value increases how quickly the spotlight will move while rotating. A negative speed value will result in reversing the direction of the rotation. If speed is 0, no pivoting.

Increasing the angle value will result in the spotlight rotating on a larger angle projection.

textSpeed and textAngle are used for the User Interface.

**SunOrbit.cs**

For the SunOrbit.cs file. The parameters available to be tweaked are speed, time and intensity. (Note this script runs on the ThirdPersonCamera in Ethan). Some code and the idea came from Hibberd’s Day/Night Cycle YouTube video [6].

A higher speed value increases how quickly the Sun will orbit around. A negative speed value will result in reversing the direction of the orbit. If speed is 0, the Sun will stand still.

Time reflects where the Sun will be located. The Sun starts at 6:00am to perform a sunrise.

Intensity is designed to increase or decrease the light effect of the directional light. This changes over the day/night cycle.

Sun Trans and Sun are for the Sun directional light source.

textTime is for the User Interface

**Shaders**

These parameters are for all in Game Objects in the scene.

Color Texture allows for a texture\_COLOR to be applied to the Game Object. This also allows for tiling and offset on the x and y planes.

Bump Map allows for a texture\_NRM to be applied to the Game Object. This also allows for tiling and offset on the x and y planes.

Bump Depth allows for a user to modify the bump mapping depth from -2.0 to 2.0.

Color allows for the user to modify the lighting color that appears on the Game Object (With exception to Specular and Rim color).

Specular Color allows for the user to modify the specular color that appears on the Game Object.

Shininess allows the user to adjust how much shine is being applied to the Game Object. A higher value leads to interesting blending effects from two spotlights.

Rim Color allows for the user to modify the rim color effect that appears on the Game Object.

Rim Power allows for the user to modify the rim lighting that appears on the Game Object. A lower value will lead to brighter lighting. A higher value will lead to dimmer lighting.

1. **Source Code Comments**

For custom C# scripts and shader refer to source code in Unity Project Assignment\_1. Commenting explains the process. The following files and their location is shown below.

|  |  |
| --- | --- |
| **File name** | **Location** |
| Spotlight\_1.cs | Assignment\_1\Assets\Lightning\Light Source Scripts |
| Spotlight\_2.cs | Assignment\_1\Assets\Lightning\Light Source Scripts |
| SunOrbit.cs | Assignment\_1\Assets\Lightning\Light Source Scripts |
| FloorShader.cs | Assignment\_1\Assets\Objects\Floor\Materials |
| FloorShader.shader | Assignment\_1\Assets\Objects\Floor\Materials |
| ChairShader.cs | Assignment\_1\Assets\Objects\HQ Classic Armchair\Materials |
| ChairShader.shader | Assignment\_1\Assets\Objects\HQ Classic Armchair\Materials |
| LampShader.cs | Assignment 1\Assignment\_1\Assets\Objects\Lamps pack\Materials |
| LampShader.shader | Assignment 1\Assignment\_1\Assets\Objects\Lamps pack\Materials |
| LionShader.cs | Assignment 1\Assignment\_1\Assets\Objects\Lion\_Statue\Materials |
| LionShader.shader | Assignment 1\Assignment\_1\Assets\Objects\Lion\_Statue\Materials |
| EthanShader.cs | Assignment 1\Assignment\_1\Assets\Objects\Standard  Assets\Characters\ThirdPersonCharacter\Materials |
| EthanShader.shader | Assignment 1\Assignment\_1\Assets\Objects\Standard Assets\Characters\ThirdPersonCharacter\Materials |
| BearShader.cs | Assignment 1\Assignment\_1\Assets\Objects\Statue Bear Rough Bronze\Materials |
| BearShader.shader | Assignment 1\Assignment\_1\Assets\Objects\Statue Bear Rough Bronze\Materials |

1. **Statement of Completeness**

Below refers to the criteria from the IGB381 Assignment 1. Below is what has been completed. Everything required should have been implemented.

“SOFTWARE”

* A well designed Object Oriented C# implementation off a 3D scene using Unity scripting

“CONTENT”

* Human figure, representing the user’s avatar.
* Movable sun is implemented for the scene.
* Two pixel shader-based spot lights
* All the objects in the scene are textured, normal mapped and lit by shaders.
* The whole scene should have high quality antialiasing applied as a final pass

“DOCUMENTATION”

* A description of the shader techniques you used to model the vision, written as a report to Steve Binkman and the game development team
* A description of the parameters to the vision shaders that have been made available to be tuned by level designers
* Source code comments
* Statement of completeness enumerating the assignment sections completed
* References to any content or source code that you used from other sources

1. **References**

[1] “Cg Programming/Unity/Cookies – Wikibooks, open books for an open world”, En.wikibooks.org, 2017. [Online]. Available: <https://en.wikibooks.org/wiki/Cg_Programming/Unity/Cookies#Unity.27s_Cookies>. [Accessed: 08-Sep-2017].

[2] R. Brown, “INB382 Texture Antialising”, YouTube, 2017. [Online]. Available: <https://www.youtube.com/watch?v=8mxmiPnWjIA>. [Accessed: 07-Sep-2017].

[3] A. Telford, “Textures – 6 Theory - Noob to Pro Unity Shader Writing in Unity 4 beginner”, YouTube, 2017. [Online]. Available <https://www.youtube.com/watch?v=iUl0WQDuiuc>. [Accessed: 06-Sep-2017].

[4] A. Telford, “Multiple Lights – 5 Theory – Noob to Pro Unity Shader Writing in Unity 4 beginner”, YouTube, 2017. [Online]. Available <https://www.youtube.com/watch?v=yoRNGZh29To>. [Accessed: 06-Sep-2017].

[5] A. Telford, “Normal Maps – 7 Practical – Noob to Pro Unity Shader Writing in Unity 4 beginner””, YouTube, 2017. [Online]. Available <https://www.youtube.com/watch?v=61ltaf-MNas>. [Accessed: 07-Sep-2017].

[6] A. Hibberd, “Unity 5 Tutorial – Day/Night Cycle”, YouTube, 2017. [Online]. Available <https://www.youtube.com/watch?v=DmhSWEJjphQ>. [Accessed: 02-Sep-2017].

[7] A. Telford, “Rim Lighting – 4 Theory – Noob to Pro Unity Shader Writing in Unity 4 beginner”, YouTube, 2017. [Online]. Available <https://www.youtube.com/watch?v=d5VcYE-3xLw>. [Accessed: 06-Sep-2017].