

Programming Languages

scifi6546

March 2020

1 Introduction to Open Shading Language

The purpose of Open Shading Language (OSL) is to create programs that are used by a raytracing rendering engines to draw scenes. Open Shading Language is used in many rendering engines such as blender's cycles engine and V-Ray. OSL has a different focus then a traditional shading languages such as glsl because OSL outputs materials rather than directly calculating RGB values. The rendering engine then uses the generated materials to calculate the trajectory of rays of light in order to render the correct color for the object.

2 Features

OSL has a c like syntax.

3 Execution

OSL is executed with a SIMD architecture. Each shader outputs a closure that can be run in parallel. The rendering engine then executes the closures to calculate the trajectory of rays of light when they hit a object with the current shader. Rendering is commonly accelerated using General Purpose Graphics Processing Units (GPGPU). For example Blender uses OpenCL and CUDA to accelerate rendering [1]. The output of shader programs are used as input of other programs. In blender the shaders can be combined using a GUI node based system as shown in the figure below.

The GUI system is then used by artists to generate the desired graphics, such as a wood material for a floor.

4 Example Code

All of the below examples are shaders that are placed onto a sphere with a single sun light and no global illumination. The code sample below is a simple white diffuse material.

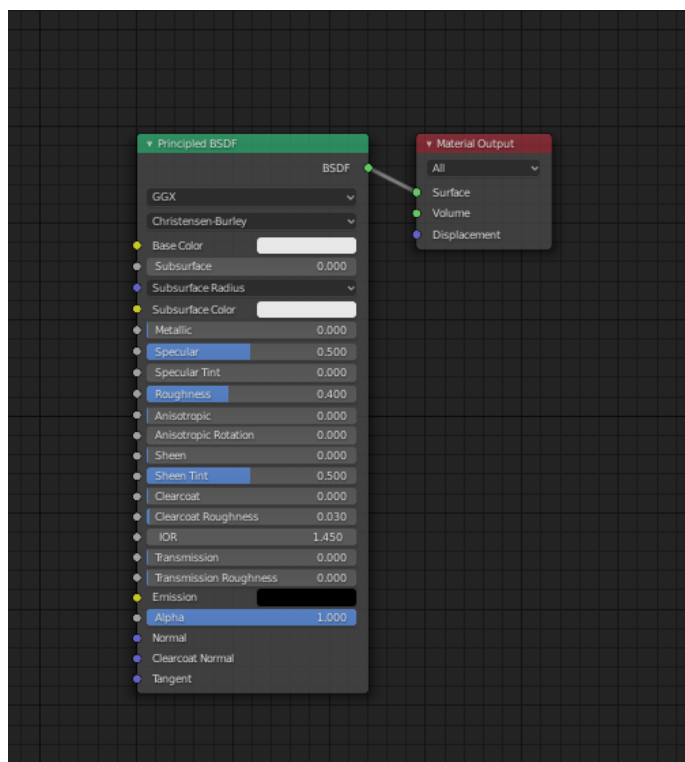
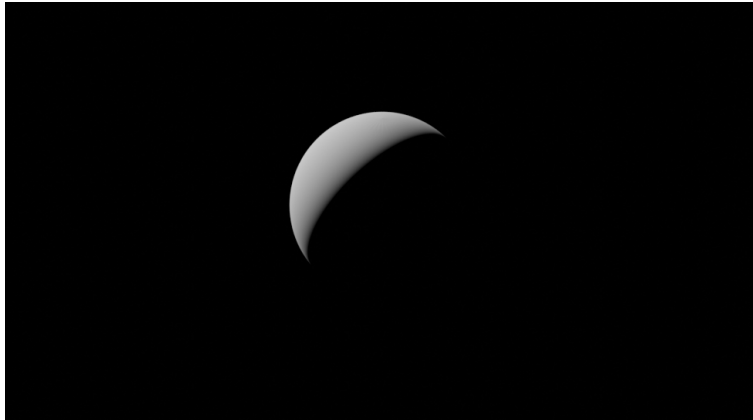


Figure 1. Blender Node System



```
shader simple_material(  
    output closure color BSDF = diffuse(N)){  
    BSDF = diffuse(N);  
}  
}
```

References

- [1] Blender Manual, https://docs.blender.org/manual/en/latest/render/cycles/gpu_rendering.html,