Programming Languages

Nicholas Alexeev

March 2020

1 Introduction to Open Shading Language

The purpose of Open Shading Language (OSL) is to create programs that are used by a raytacing 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 then 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

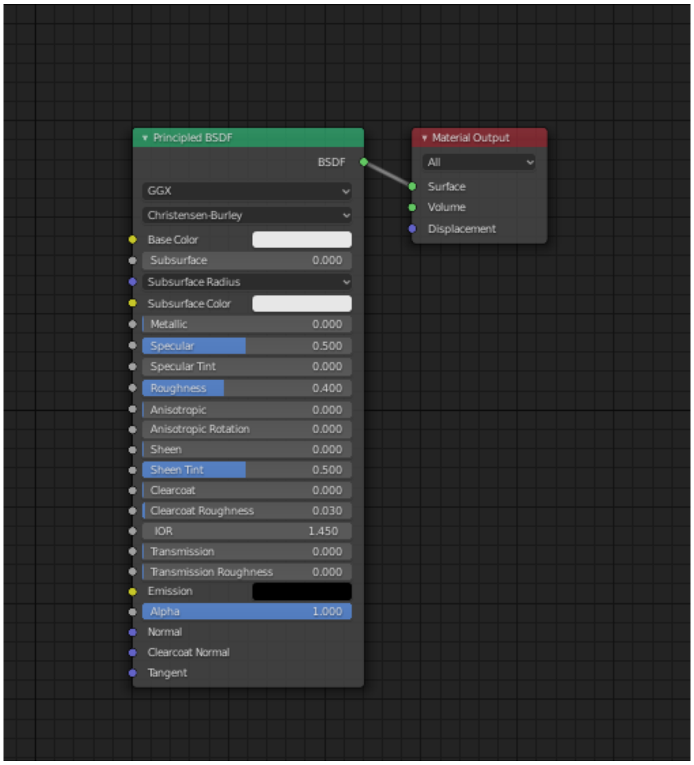
OSL supports parallel execution across many different computation platforms. OSL has manifest typing as types are declared by the programmer. OSL primary form of data output is a closure.

3 Execution

OSL is ececuted 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 shader programs. In blender the shaders can be combined using a GUI node based system as shown in the figure below.

4 Intended Use Case

OSL shaders are intended to be used by artists to shade their 3d models. The shaders can be created by an expert programmer and then tey are imported



Principled BSDF

BSDF

GGX

Christensen-Burley

Base Color

Subsurface 0.000

Subsurface Radius

Subsurface Color

Metallic 0.000

Specular 0.500

Specular Tint 0.000

Roughness 0.400

Anisotropic 0.000

Anisotropic Rotation 0.000

Sheen 0.000

Sheen Tint 0.500

Clearcoat 0.000

Clearcoat Roughness 0.030

IOR 1.450

Transmission 0.000

Transmission Roughness 0.000

Emission

Alpha 1.000

Normal

Clearcoat Normal

Tangent

Material Output

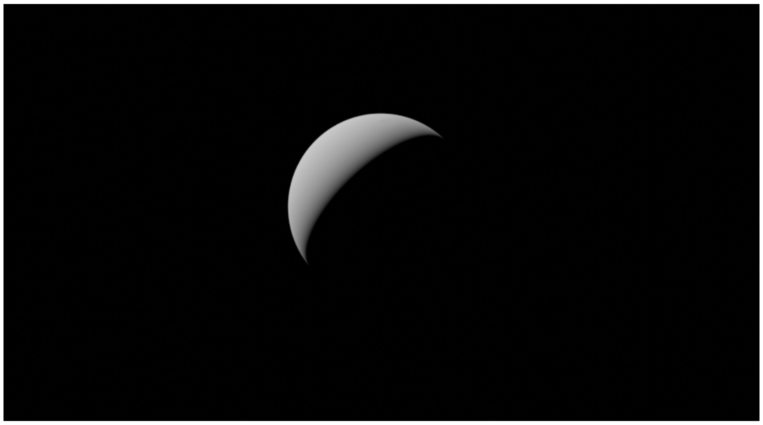
All

Surface

Volume

Displacement

Figure 1. Blender Node System



with a relativly easy to use GUI and they are then combined with other shaders and input sources such as textures and preset constants. OSL is not intended to be used to generate images on its own and instead it is used to inform a rendering engine how materials applied to objects should behave.

5 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.

```
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,