

Project 3 – Shading

In this assignment, I understood how to implement simple lighting and shading with Blinn–Phong reflection model.

What you have implemented & How to use your implementation:

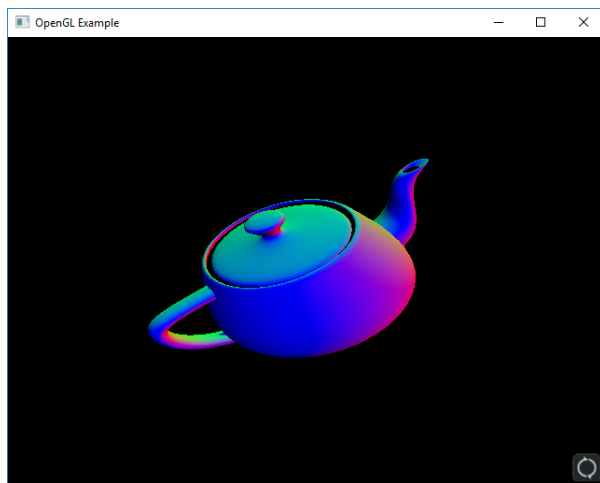
Step 1: Display triangle vertices:

In this assignment, we are not drawing `GL_POINTS` anymore. Instead, we use `GL_TRIANGLES` to display our teapot. To do this, we can create index buffer saving the index order of the triangles. The main reason is we can have lower memory usage when we use the index buffer.

Step 2: Normal buffer:

For shading the triangles, we need surface normals to calculate angles. This step is similar to how we generate the vertex buffer but for just for normal vertices. Then we can use the normal buffer as the color for our teapot. My implementation would be available in my fragment shader, and it looks like:

```
vec3 N = normalize( Normal_cameraspace );  
color = N;
```



Step 3: Blinn shading:

The last step is to implement the Blinn–Phong reflection model.

In my project, I just hard code my AmbientColor being `vec3(0.1, 0.1, 0.1)` and DiffuseColor being `vec3(1, 0, 0)`. Based on the model, my implement would be

```
vec3 MaterialAmbientColor = vec3(0.1, 0.1, 0.1);
```

```
float diffuseTerm = clamp( dot(N, L), 0, 1 );
```

```
vec3 MaterialDiffuseColor = vec3(1, 0, 0);
```

```
float Shininess = 50;
```

```
vec3 H = normalize(L + V);
```

```
float specularTerm = pow(clamp(dot(N, H), 0, 1), Shininess);
```

```
vec3 MaterialSpecularColor = vec3(1, 1, 1);
```

```
color = MaterialAmbientColor +  
        MaterialDiffuseColor * diffuseTerm +  
        MaterialSpecularColor * specularTerm;
```

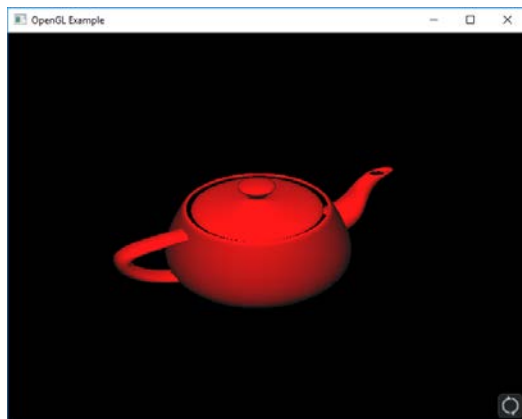
N: normal vertex

L: lighting direction

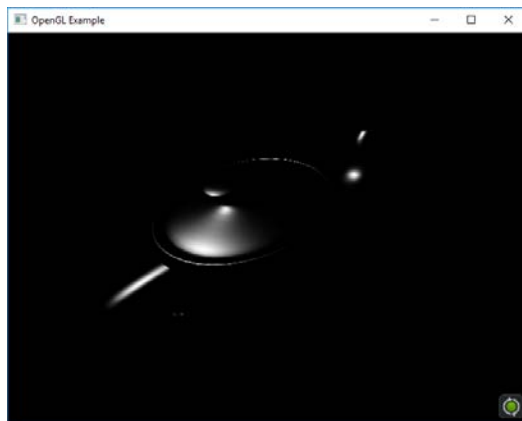
V: view direction

We can use **Left-Ctrl** to rotate the lighting position.

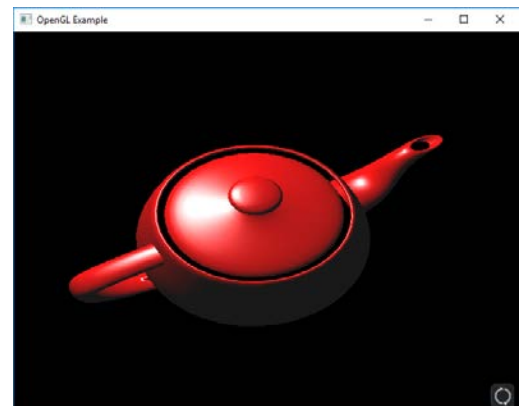
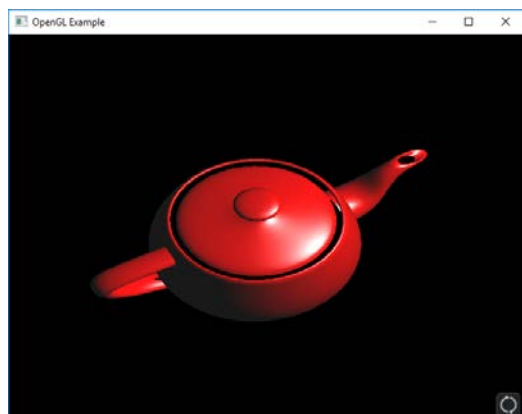
Ambient + Diffuse:



Specular:



Ambient + Diffuse + Specular and lighting rotation:



What you could not implement:

None

Additional functionalities beyond project requirements:

None

What operating system and compiler you used:

I work and compile my code in Visual Studio 2017 in Windows 10.

External libraries and additional requirements to compile your project:

cyCodeBase

GLEW

Ref:

<http://www.cemyuksel.com/cyCodeBase/index.html>

<http://www.opengl-tutorial.org/>