

Due Date Part 1: 23:59 pm on Thursday, December 12th, 2024

Due Date: 23:59 pm on Thursday, January 2nd, 2025

WebGL2 Texture and Lighting

In this experiment, you are expected to learn creating textured objects and use per-shader based lighting. Also, you are expected to create controllable camera from the previous experiment.

Part 1

In this part, you are expected to download the Experiment5_Part1_base.zip (see Figure 1a) and umbrella.png from the resources in Piazza. Then, you will cover the cube with texture. You will use buttons to change light's position as in Figure 1b.



Figure 1: Part 1.

Part 2

In this section, you will explore textures and lighting in WebGL2. You will learn two different methods for implementing textures and use directional lighting to illuminate your scene effectively. The steps are explained in the following statements:

1. You will create a scene by placing a sphere and an indoor plant (see Figure 2).
2. Create a sphere with radius 15 units in center.
 - (a) Cover the sphere with sand images (albedo, ambient occlusion, height, metallic, normal and roughness) using planar projection mapping [2] and physically based rendering (PBR) [3] shaders.
 - (b) The texture should be seen as Figure 3.
3. Read the plant object and material file in your code and place it on sphere on the given figure 2.
 - (a) You will implement textures using material (.mtl) file [4].
 - (b) The texture should be seen as Figure 4.
4. You will position a directional light above the sphere, aiming it toward the center of the world.



Figure 2: The scene.

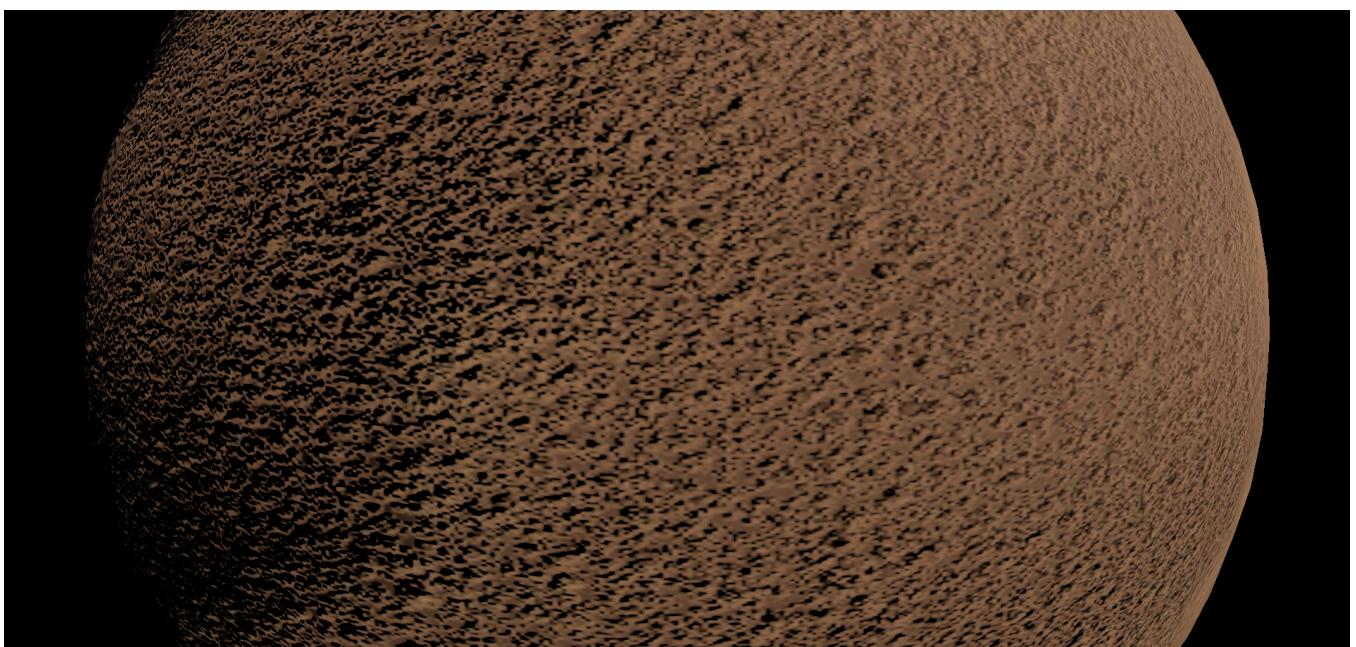


Figure 3: Sand texture.



Figure 4: Indoor plant.

- (a) The directional light will rotate around the ~~x axis~~ while remaining focused on the center of the world, causing the illuminated areas of the sphere and plant to shift dynamically.

- ✓ 5. You will create a controllable camera from the previous assignment.
✓ 6. Sphere and plant will rotate around y axes.

The Implementation Details

1. Implement your homework using **WebGL2**. All programming assignments must use the shader-based functionality of **WebGL2**: at least one vertex shader and one fragment shader.
2. The assignment must be original work. Turning in someone else's work, in whole or in part, as your own will be considered a violation of academic integrity. Please note that the former condition also holds for the material found on the Web as everything on the Web has been written by someone else. **Detection of such plagiarism in a submission will automatically void the submission and establish grounds to trigger an official disciplinary investigation.** General discussion of assignment among peers is allowed, but do not share answers, algorithms or source codes. **Also using other resources (example source code, book, webpage etc.) as a code and javascript libraries (except jquery, Angel's book) are not allowed.**
3. Do not write the scripts into the html file. Reference your scripts in html.
4. You should use Netbeans or Webstorm as an IDE for your projects.

The Report

You will write a report on latex for this assignment. You will explain the code parts and algorithm for part 1 and part 2.

What to Hand In

You should submit the entire Netbeans or Webstorm project directory, including javascript files and html file, in a zip file extracted from IDE. The submission file structure is as given below:

- b<studentNumber>.zip
 - |–Experiment5_2024
 - |—Part 1(**The whole Netbeans or Webstorm project**)
 - |—Part 2(**The whole Netbeans or Webstorm project**)
 - |–report.pdf

Achieve this folder as **b<studentNumber>.zip** and send via submit system.

Grading

The assignment will be graded out of 100:

- PART 1 - CODE: 0 (no implementation)
20 (correct solution).
- PART 2 - CODE: 0 (no implementation)
10 (sphere with rotation)
10 (sphere texture)
10 (plant with rotation)
10 (plant texture)
10 (directional light with rotation)
10 (controllable camera)
- REPORT: 20

Academic Integrity

All assignments must be done individually, unless otherwise indicated. You are encouraged to discuss with your classmates about the assignments given, but these discussions should be carried out in an abstract way. That is, discussions related to a particular solution to a specific problem (either in actual code or in pseudocode) will not be tolerated. In short, turning in someone else's work, in whole or in part, as your own will be considered as a violation of academic integrity. Please note that the former condition also holds for the material found on the web as everything on the Web has been written by someone else.

References

- [1] <https://github.com/esangel/WebGL>
- [2] <https://webgl2fundamentals.org/webgl/lessons/webgl-planar-projection-mapping.html>
- [3] <https://blog.demofox.org/2017/07/10/webgl-pbr-implementation/>
- [4] <https://webgl2fundamentals.org/webgl/lessons/webgl-load-obj-w-mtl.html>