FRASER BARKER

1600196

1600196@uad.ac.uk

ASSESSMENT Report

CMP301

Contents

[Overview 2](#_Toc532058782)

[Objects 2](#_Toc532058783)

[Brief Response 2](#_Toc532058784)

[UI & Controls 2](#_Toc532058785)

[Algorithms and Data Structures 3](#_Toc532058786)

[Buffers & Maths 3](#_Toc532058787)

[Vertex Manipulation 3](#_Toc532058788)

[Post Processing 3](#_Toc532058789)

[Lighting & Shadows 3](#_Toc532058790)

[Tessellation 3](#_Toc532058791)

[Geometry Shader Stage 3](#_Toc532058792)

[Shaders 3](#_Toc532058793)

[Depth of Field 3](#_Toc532058794)

[Depth 3](#_Toc532058795)

[Displacement Depth 3](#_Toc532058796)

[Displacement 3](#_Toc532058797)

[Horizontal Blur 3](#_Toc532058798)

[Vertical Blur 3](#_Toc532058799)

[Render Displacement Normals 3](#_Toc532058800)

[Render Tessellated Normals 3](#_Toc532058801)

[Shadow Shader 3](#_Toc532058802)

[Tessellation Depth 3](#_Toc532058803)

[Tessellation 3](#_Toc532058804)

[Texture 3](#_Toc532058805)

[Critical Reflection 4](#_Toc532058806)

[References 5](#_Toc532058807)

# Overview

## Objects

* What objects are in the scene and what techniques do they demonstrate.

[PICTURE OF SCENE]

There are three objects within the scene. Two are spheres and the third is a plane. The two spheres both showcase tessellation and vertex manipulation. Both spheres have their depth calculated from each light and the camera. This allows the spheres to have their shadows cast dynamically onto the plane below them. The plane is more basic and has a plain texture applied to it. It also has its depth calculated from each light and the camera but its main use is to have the spheres shadows cast onto it.

* Water sphere demonstrates vertex manipulation, post processing, lighting and shadows, tessellation and the geometry shader stage.
* Earth sphere demonstrates vertex manipulation, post processing, lighting and shadows, tessellation and the geometry shader stage.
* Plane demonstrates lighting and shadows.

## Brief Response

* How you have responded to the coursework brief.

Initially, I planned to have a cohesive desert island scene but changed over due to being able to demonstrate the same process with much less hassle. Even straying from the proposal submitted in week seven although I came pretty close. Two spheres demonstrating more or less all the shader stages. Just missing the third (and arguably more desired, personally) sphere which would have showcased algorithmic morphing and reflection.

## UI & Controls

* Outline any UI elements and controls beyond the standard controls.

### Standard Controls

* Move the camera with W, A, S, D.
* Alter the camera’s height with E, Q.
* Alter the camera’s rotation with either the Up, Down, Left and Right arrow keys, or hit Spacebar and use the mouse to alter rotation.

### ImGUI Controls

[PICTURE OF IMGUI]

Separated into specific sections to alter the shape that displays a specific graphical technique.

* Tessellation
  + Tessellation Factor
    - Ranges from 1 – 64. Alters the tessellation value that is applied to the insides and edges of each tessellated shape.
    - Is also affected by the position of the camera as the tessellation factor in the shader stages is divided by the distance the camera is away from the mesh.
  + Wave Height
    - Ranges from 0 – 5 as the shape is manipulated along its normal. Alters the height of the wave.
  + Wave Frequency
    - Ranges from 0 – 5. Alters the frequency of the waves appearing on the mesh.
  + Wave Speed
    - Ranges from 0 – 5. Alters how fast the waves moves across the mesh.
* Displacement
  + Displacement Height
    - Ranges from 0 – 1. Alters the vertexes position based on the height map applied to the mesh.
* Depth of Field
  + Range
    - Ranges from 0 – 2. Alters range at which objects around the centre of the depth texture are perceived to be in focus.
* Light Settings
  + Left Directional
    - Ambient
      * Drag on the 4 bars to alter each colour of the ambient accordingly.
    - Diffuse
      * Alter each bar on click on the small square to be presented with a colour picker to set what the diffuse colour will be.
    - Direction
      * Drag on the three bars to alter the direction of the light. There seems to be an issue with the X direction being zero so I tried to implement a rudimentary if statement check to avoid this error. Assert for the X direction being zero being hit.
    - Position
      * Drag on the three bars to alter the position of the light.
  + Right Directional
    - Ambient
      * Drag on the 4 bars to alter each colour of the ambient accordingly.
    - Diffuse
      * Alter each bar on click on the small square to be presented with a colour picker to set what the diffuse colour will be.
    - Direction
      * Drag on the three bars to alter the direction of the light. There seems to be an issue with the X direction being zero so I tried to implement a rudimentary if statement check to avoid this error. Assert for the X direction being zero being hit.
    - Position
      * Drag on the three bars to alter the position of the light.
  + Spot Light
    - Ambient
      * Drag on the 4 bars to alter each colour of the ambient accordingly.
    - Diffuse
      * Alter each bar on click on the small square to be presented with a colour picker to set what the diffuse colour will be.
    - Direction
      * Drag on the three bars to alter the direction of the light. There seems to be an issue with the X direction being zero so I tried to implement a rudimentary if statement check to avoid this error. Assert for the X direction being zero being hit.
    - Position
      * Drag on the three bars to alter the position of the light.
    - Advanced
      * Angle
        + Ranges from 0 – 90. Alters the angle that the spot light is calculated with.
      * Constant Factor
        + Ranges from 0.01 – 1. Alters the constant factor that the spot light is calculated with.
      * Linear Factor
        + Ranges from 0.01 – 1. Alters the linear factor that the spot light is calculated with.
      * Quadratic Factor
        + Ranges from 0 – 0.5. Alters the quadratic factor that the spot light is calculated with.
* Ortho Meshes
  + Render top left ortho mesh
    - Boolean to enable/disable rendering this mesh.
  + Render top right ortho mesh
    - Boolean to enable/disable rendering this mesh.
  + Render bottom left ortho mesh
    - Boolean to enable/disable rendering this mesh.
  + Render bottom right ortho mesh
    - Boolean to enable/disable rendering this mesh.

# Algorithms and Data Structures

## Buffers & Maths

* An in-depth explanation and justification (based on complexity and/or hardware architecture) of the algorithms and data structures used in the scene.
* Important calculations used, data passed and shader stages.
* This should focus on the hlsl/shaders written.
* Providing diagrams, code snippets and supporting screenshots as required.

## Vertex Manipulation

[PICTURE OF THE TWO TYPES]

**Tessellation and Displacement Shaders.** Both spheres within the scene showcase a different example of vertex manipulation. The water sphere has its vertexes manipulated along their normal via a sin wave. The earth sphere showcases vertex manipulation via a height map which has been applied to the sphere. The earth spheres vertexes are manipulated based along their normal and the height at which point the height map was sampled from. Displacement shader uses height map. Tessellation shader by default manipulates along meshes normal via a sin wave.

## Post Processing

[PICTURE OF DoF]

**Vertical Blur, Horizontal Blur and Depth of Field Shaders.** The scene utilises the vertical and horizontal Gaussian blur with down/up sampling to apply a final depth of field post processing effect. The depth of field shader takes in the normal scene texture, the combined blur texture after down sampling and the depth texture created from the cameras depth pass. The depth of field shader also takes in three floats, one of which is the range which is user controlled via ImGUI, the other two which are the SCREEN\_NEAR and SCREEN\_DEPTH. These floats plus any padding required are passed into the pixel shader where they are used to calculate the blur factor. The blur factor is used to linearly interpolate between the two textures (normal and blurred), based on what the camera is looking at. The three textures passed in are sampled based on the texture coordinates of the ortho mesh which this shader is applied to. The depth texture is sampled twice, once using the given texture coordinates and another based on the centre of the depth texture. These values are then flipped to equal 1 minus their value to put them in the 0 – 1 range. They are then both multiplied by the result of SCREEN\_DEPTH – SCREEN\_NEAR and passed in to the calculation of the blur factor. **TALK ABOUT REQUIREMENTS.** The result blurs the scene based on what the camera is looking at.

## Lighting & Shadows

[PICTURE OF LIGHTING AND SHADOWS]

**Depth, Displacement Depth and Tessellation Depth Shaders. Also Tessellation and Displacement Shaders.** There are three lights within the scene, two are directional and the third is a spot light. Each light has its own shadow map generated via the three depth shaders based on what object it is. These shadow maps are then used within the shadow, tessellation and displacement shaders. **Talk about similar code in each.** **Talk about weird floor shadow issue with the spotlight.**

## Tessellation

[PICTURE OF TESSELLATION WITH WIREFRAME]

Both spheres within the scene are dynamically tessellated. The tessellation factor which affects the inside and edges of both sphere can be altered via ImGUI. Both spheres are subject to distance based tessellation also which divides the tessellation factor passed into the domain shader by the distance each mesh is away from the camera.

## Geometry Shader Stage

[PICTURE OF NORMALS BEING RENDERED TO TEXTURE]

* Talk about weird rotation issue but otherwise seeming fine.

# Shaders

## ~~Depth of Field~~

## Depth

## Displacement Depth

## ~~Displacement~~

## ~~Horizontal Blur~~

## ~~Vertical Blur~~

## Render Displacement Normals

## Render Tessellated Normals

## Shadow Shader

## Tessellation Depth

## ~~Tessellation~~

## Texture

# Critical Reflection

* Discussion of what you learned, any shortfalls, areas for improvements, how you might extend the application, what you would do differently if doing it over again. Offer possible solutions to the challenges or how it could be taken forward.
* This should refer to the coursework plan done during week 7/8.

# References

* References for any images, models or techniques incorporated in your application. Cases of plagiarism will be taken very seriously.