COSC422 Advanced Computer Graphics

Assignment 1: Terrain Rendering

Due: 11pm, Friday, 24 August 2018

Maximum Marks: 25

I. Task Description:

The tessellation and geometry shader stages of the OpenGL-4 programmable pipeline have found several applications in mesh processing and rendering. In this assignment, you will develop an application that uses tessellation and geometry shaders to render three-dimensional terrain models based on height maps.

A programming exercise (TerrainProgramming.pdf) containing a sample height map (HeightMap1.tga), a description of the method for generating the wireframe model of a terrain, and sample code are provided. You may extend this program to incorporate the additional requirements of this assignment listed below.

1.(a) Minimum requirements [Max marks 16]: Your implementation of the terrain generation algorithm should include the following functions/features:

- Dynamic level of detail: The tessellation level for a patch should depend on its distance from the camera. Your program should be able to demonstrate this by moving the camera over the terrain in wireframe display mode.

- The geometric shader must include lighting calculations (ambient and diffuse terms) to render the terrain under a light source.

- The implementation must use at least three textures for terrain features (eg. water, rock, grass, snow etc.), and use height based texturing with appropriate blending of textures to get a smooth transition from one texture to another. Water regions must have flat surfaces.

- The program should be able to display at least two terrain models (using two different height maps)

- The program should include the following keyboard and mouse functions for user interaction:
  'w': Toggle between wireframe and solid-fill (textured) display
  '1', '2', …: Display terrain model 1, 2, ..
  Arrow keys: Camera motion over the terrain
  You may include additional keyboard/mouse functions as necessary.

1.(b) Extra features [Max marks 5]: Some of the additional features that you could implement to gain extra marks up to a maximum of 5 marks are listed below:

- Curvature based LOD: Planar regions on a terrain can have a low level of tessellation. The tessellation level could be further adjusted based on the local curvature of a patch [1-2 marks]

- Cracking is a common problem in terrain rendering where the tessellation levels of two adjacent patches do not match along a common edge. A simple solution could be devised to solve the problem of cracking. Please provide screenshots in the report showing the working of your solution. [1 mark]

- Adjustable water levels [1 mark]

- Adjustable snow level [1 mark]

- Water features (ripples, colour variation with depth etc.) [1-2 marks]

- Billboarded trees or other objects [1-2 marks]

- Terrain features such as roads, helipads etc [1-2 marks]

- 3D models (bridges, buildings, windmills etc) [1-2 marks depending on rendering quality]

The list given above should not be taken as the complete set of features that can be implemented.

II. Report (Max. 4 pages; Max. marks: 4):

Please prepare a report describing your work, and include the following sections:

- A brief outline of your program including a description of the extra features implemented. You may also describe problems/challenges faced and how you attempted to solve them.

- All relevant equations describing how tessellation levels and any other animation parameters are updated, how vertex positions, texture coordinates and other primitive characteristics are computed.

- A few screenshots showing the outputs of your program.

- The complete list of keyboard/mouse functions defined for user interaction.

- List of references, acknowledgements.

III. Program Development:

You may use math library functions (eg. GLM), mesh models, and images that are available on the Internet or obtained from other sources such as books. Please acknowledge the source in your report. You may also use programs and other supplementary materials provided in this course. If your implementation is based on a method described in a paper, book etc., please give full details of the source in the list of references.

Demo programs found on the Internet and other OpenGL resources should not be submitted as part of the assignment. Please do not use OpenGL Extensions (ARB, EXT etc) or third-party mesh processing libraries in your program.

IV. Assignment Submission

Submit your files using the assignment link on Learn (learn.canterbury.ac.nz) before 11pm on 24 August 2018. Your submission must contain:

1. The source code(s) and all supplementary files (textures, mesh files) needed to run your program. Please do not include freeglut, opengl, glew or glm library files.

2. Your report in Word or PDF format.

Miscellaneous

1. Check regularly on the *Learn* system forums for spec updates and clarifications.
2. You may submit up to one week late for a 15% penalty.

3. This is not a group project. Your assignment must represent your own individual work. In particular, students are not permitted to share program source code in any way. However, you may discuss ideas, implementation issues etc using the class forum on Learn.

4. Standard departmental regulations regarding dishonest practices and late submissions apply.