Université d'Ottawa
Faculté de génie

École de science informatique
et de génie électrique

uOttawa

L'Université canadienne
Canada's university

University of Ottawa
Faculty of Engineering

School of Electrical Engineering
and Computer Science

CSI4130 Computer Graphics
Winter 2020: Assignment 4

Due: Demo: Wednesday, April 1st, 2020, 8:30 in Class
Code and Documentation: Monday, April 6th, 2020, 23:00 in Virtual Campus
University of Ottawa - Université d'Ottawa

Jochen Lang

1 Project Description [20 in total]

Choose one of the following for your course project:

- THRRE or WebGL: Extend one of the assignment with an additional feature of similar difficulty than the original assignment.
- WebGL: Implement various mapping techniques without THREE. Example: Procedural bump maps in the fragment shader.
- WebGL: Use sphere mapping or cube mapping to generate texture coordinates for meshes loaded from file. Demonstrate your code by loading an image as a sphere map, or six images arranged in a cube map for cube mapping with various meshes.
- THREE or WebGL: Create a Perlin noise demo. Give the user control over the effects and let the user decide how to use it (directly, with turbulence, for color selection, etc.)
- THREE: Explore the ammo.js physics engine in THREE (see THREE examples). Note that ammo.js is very poorly documented, however, the original C/C++ code called Bullet is one of the most popular open source engines. One idea is to port a simpler example from Bullet in C++ or even python to js in THREE.
- WebGL: Simple RayTracer. A good (and not simple) example is the Cornell box demo by Evan Wallace at http://madebyevan.com/webgl-path-tracing/
- THREE or WebGL: Your own idea of similar difficulty but you must consult with me first.

You can use whatever you like in this project assignment from the labs. If you are looking for meshes, there are many commercial sites that provide sample meshes without charge. One source for numerous 3D models is Google Sketchup warehouse but the native sketch up format is not commonly supported by other tools (Sketch-up pro has appearently exporter to more useful formats). Some models however support another format, e.g., dae, which you may be able to load in MeshLab or Blender and convert to obj wavefront. The quality of the meshes on Sketchup varies greatly. You can find MeshLab at http://www.meshlab.net/, or you can try MeshLabJS at http://www.meshlabjs.net/. Blender is a full-fledged opens source modelling tool, see https://www.blender.org but be warned it is not only very powerful but the user interface has a steep learning curve. Two academic sources of free meshes are http://vcg.isti.cnr.it/nexus/ and http://people.csail.mit.edu/tmertens/textransfer/data/. In any case, make sure to only use meshes legally obtained and which do not depict violence. In summary: Your video should not need an age restriction and you should be able to show it to a future employer; use your judgement.

2 Presentation

You need to present your project in the last class of CSI4130 (April 1st, 2020). The presentation will be a 2-3 minute demo followed by Q&A. The best demo will receive an award.

3 Submission

Please organize your project in a zip archive and upload it to Virtual Campus. Your code must run by loading the corresponding web page in your submission. You can assume that the same libraries as in the labs are available at the same locations. Avoid additional libraries beyond what was used in the lab but if you absolutely need it, than it must be part of your submission. Please include in your submission a short document (preferably pdf) which describes the following:

- Provide a one-page (max) description of your project (goals, achievements)
- Provide a short description of the usage of your program (keys, mouse, steps to perform).
- Acknowledge any external resources used, i.e., *anything that is not your own.*