2015-12-15

BTH

3D Programming – DV1541/DV1542

Project – 8 hp

Description

In this project you will implement from scratch an Interactive 3D Application.

Rules:

- All GPU code (shaders) must be 100% of your authorship, and you should be able to explain everything in them.
- You can reuse other libraries, such as sound or window management (SDL, SFML, etc)
- The application side should be written using C/C++
- The APIs you can use are OpenGL and Direct3D
- You can work in groups, up to 3 participants.
    - Everybody in a group should understand all the code submitted.

Mandatory techniques or implementation details:

- You have to use a “perspective” camera,
- You have to use textures in your 3D models,
- You have to implement ambient, diffuse AND specular lighting in your shaders,
- You have to implement a first-person camera movement
    - Use “w-a-s-d” for moving and the mouse for looking at different directions
- You have to have Vertex and Fragment shaders **at least (Geometry Shader** is not mandatory, but recommended)
- At the end, you will submit the project and also a written report in English describing each of the techniques you have implemented. The English

language is not graded in any way, so do not worry but use a spellchecker before submitting the report. The report should explain each technique implemented in detail.

Additional Techniques: additionally to the mandatory techniques, you will implement a number of techniques from each category in the list below. The parenthesis determine how many of each category have to be implemented. For example, for the "Core Techniques" you have to implement **ONE (1)** of the options.

- Core techniques (**1**):
  - **Deferred Rendering**
  - **Skeletal Animation** (M3D, FBX, Collada, etc)
- Geometry (**2**):
  - **Parsing and rendering an existing model format** (OBJ, FBX, Collada, etc). This includes parsing and using the material defined in the format (textures)
  - **Displacement mapping using Tessellation hardware**
  - **Height-map terrain rendering, user can walk on the terrain.**
  - **Level of Detail using Tessellation**
  - **Morph-based vertex animation**
- Texturing and Lighting (**1**):
  - **Normal mapping**
  - ~~**Dynamic Ambient Occlusion**~~ **– Screen Space Ambient Occlusion**
  - **Blend mapping (multi-texturing)**
- Projection Techniques (**1**):
  - **Shadow mapping**
  - **Shadow volumes**
  - **Dynamic cubic environment mapping**
  - **Dynamic paraboloid mapping**
- Acceleration Techniques (**2**):
  - **View frustum culling against a quadtree**
  - **Occlusion culling**
  - **Back face culling using Geometry Shader**

- Other techniques (**2**):
    - **Particle systems with billboarded particles**
    - **Gaussian filter**
    - **Bilateral filter**
    - **Mouse Picking**
    - **Screen-Space Antialiasing**
    - **Water-effect**
    - **Glow-effect**
    - **Solve any technique listed in this category using Compute Shaders**
    - **Make a small game in your application**

Guidelines:

- We do not assess on how nice the models or the textures look, but on the correctness of the techniques. As explained in the lectures, many techniques have variations, and it is ok to implement a variation as long as you understand it.
- It should be clear when presenting the project, who did what in each group
- The project presentations are going to be individual, and do not affect the other members of the group.
- If you have worked on Linux or MacOS you will have to bring your computer to present your project, it is not enough with showing the code, the application has to be ran in front of the teacher/s.
- More information will be published in ITs Learning when necessary to support some of the techniques, and you will be notified about it.

Submission, deadline and presentation

You should submit your solution to Its Learning.

Before submitting, please “clean the solution (both Debug and Release modes), and delete the files with extension “.sdf” from the project folder. After this, make a ZIP file of the project folder.

IMPORTANT:

The deadline for submission is two weeks after we finish the lectures, on the 3rd of April 2016.

If you do not meet the deadline, the resubmission deadline will be four weeks later, 3rd of May.

We will establish a date for presentations in the future.

The assessment of your code is very important, please do not take this lightly. You should be able to explain your code (not the code provided), why it works, and what happens if you make changes to it. You should be able to explain the math that you use in your code as well.

Work hard, good luck and happy coding!

The teachers