COMP 308 Final Project

1. Objective

The main objective of the project is to use your technical knowledge of 3D computer graphics to solve practical problems. You will pick an interesting and challenging topic to improve the computer graphics pipeline that has been built during the three assignments and the mid-term project in this course.

You will then create a real-time interactive demo to demonstrate your new pipeline. You will need to fully utilize the technical content covered in lectures as well as additional issues that were not a part of the course.

The project will require you to:

1. Choose some technically challenging problems in 3D computer graphics and efficiently address and solve them.

2. Implement additional features into the graphics pipeline to improve the quality of the output.

3. Demonstrate your improved pipeline in a real-time interactive demo simulating a possible real world application.

4. Efficiently describe your technical contributions in the form of a technical report.

Due dates:

- Proposal Submission: 23:59 **22nd August** 2014
- Proposal Presentation: 9th September 2014 (lecture time)
- Project Progress Discussion: 7th, 8th October 2014 (lecture time)
- Code Submission: 23:59 17th October 2012
- Individual Reports Submission: 23:59 17th October 2012
- Marking Session: TBA

2. Scenario

It is assumed that you are trying to win a commission from Beta-FX, a large graphics company, to make improvements to their 3D real-time graphics pipeline. While they have had huge international success with their products, their graphics pipeline is still very primitive. In fact, it happens to match exactly the graphics pipeline created by adding together three assignments and the mid-term project from COMP 308! They have decided to use the huge profits from their previous products to spruce up their graphics pipeline, to help them keep up in this quickly changing industry. They have put up an advertisement looking

for small teams of developers to provide a demo of how they can improve the system, with the best demo winning a contract with them. In order to win this contract you will need to form a team of 1 to 3 members, write a proposal of the work you intend to do, and finally give a short live technical demonstration (on site/off line) of your system working in real time.

3. First Stage – Proposal

Submission Due: 23:59 22nd August 2014 (no late penalty until 31st August)

Presentation Due: 9th September 2014 (during lecture)

For the first stage, you will need to form a team and submit a single proposal of the team project. There are a large number of possible additional features that you could implement, and you are able to choose freely. All proposals will be assessed and rated for both feasibility and technical difficulty.

3.1. Forming a Team

A team is a group of students ranging from one to three members. It is expected that all the members will collaborate to build a single complete pipeline, but that each member will have parts for which they are personally responsible. **Note that working alone is allowed, but it is highly recommended that you work with other members;** more learning, more interesting result, high chance of success (risk shared), lower workload (overhead shared). However, 60% of grades will be assigned per student and based on individual contributions, so you cannot rely on your teammates to improve your grade.

One member of the team must be nominated as the "*contact person*". This person is responsible for submitting the proposal and source code for the project.

3.2. Selecting a Topic

All projects must be technical in nature. It is strongly recommended that topics are chosen such that they will look nice in a demo, but the main purpose of the project is to show your technical skills. Each member of the team will work on one or more tasks but cannot work on the same topic as another member. You are free to choose your project topic, but your proposal needs permission by the course coordinator. Some example topics for each stage of the pipeline are:

- **3D geometry modeling and interface**: image based modeling, 3D reconstruction, mesh generation (smoothing), compression, scene graph interface, geometry file interface, procedural modeling (terrain, tree, etc)
- **Animation**: motion capture control (motion blending, motion retargeting), inverse kinematics, physics based animation, trajectory control (camera animation, walking path animation), and particle animation
- **Interaction control**: scene graph control (interactive interfaces for manipulating objects, lights, materials, textures), collision detection and handling, vision based

motion control (e.g. kinect).

- **Skinning and deformation**: advanced skinning algorithms (pose space deformation, multi-weighted skinning), facial animation, morphing, cloth simulation, physics based deformation, and fluid simulation
- **Lighting and materials**: soft shadows, area light, and illumination models (BRDF)
- **Texturing**: multiple textures (bump maps, light maps, projected textures), procedural textures
- **Rendering**: non photorealistic rendering, photo realistic rendering, global illumination, volume rendering, large scene rendering (e.g. level of detail controls)

Please read the course references and journal or conference papers to find ideas. SIGGRAPH, SIGGRAPH Asia, ACM Transactions on Graphics, IEEE Transactions on Visualization and Computer Graphics, Eurographics, Symposium on Interactive 3D Graphics and Games, and Symposium on Computer Animation are places where you can look. The course reference books and SIGGRAPH Course notes are recommended as good sources.

Example Task Allocation

- **Member 1: modeling and texturing**
  - Topic: large scene model interface and light mapping
  - Methods: extend obj parser to adapt multiple texture, light map texturing
  - References: real time rendering 3rd Edition, SIGGRAPH course note 00
- **Member 2: animation and interactive control**
  - Topic: advanced camera and walking path animation
  - Methods: spline based path control, squad interpolation, speed control, motion synthesis.
  - References: SIGGRAPH87 paper (author, title), Graphics Gem IV
- **Member 3: skinning**
  - Topic: example based skin deformation
  - Methods: creating sample poses using Maya, pose space deformation
  - References: data from turbosquid, Eurographics07 paper (author, title),

3.3. Submitting and Presenting the Proposal

The contact person for each team must submit a 1 to 2 pages project proposal before the due date using the ECS submission system. Each proposal will be presented by students and will receive feedback in the lecture. There will be a penalty applied to each team member's grade if it is late or incomplete.

The proposal must include all followings:

- The name, student ID and email of each member of the group
- Who has been designated as the contact person for this group (note this must be the same person who is submitting the proposal)
- The project title
- The objective of the project
- What technical problems each member has been assigned
- How each of the problems will be solved, with a brief description
- Plan and milestone (possible fortnight output)
- References

Note that if the reference is not clear, you cannot get a good mark

4. Implementing the System

- **Progress Discussion: 7th, 8th October 2014**
- **Code Submission: 23:59 17th October 2014**

In this part, your team will implement the system you have proposed (including any changes as a result of feedback). We provide an example 3D scene and avatar model. These are just examples that have been checked against the *G308_Geometry* class and there are no special benefits to using them. Note that the Sponza scene has more than one associated texture, meaning that it cannot be fully used with the default OBJ parser. Teams can choose to use other scenes and models (e.g. from http://graphics.cs.williams.edu/data/meshes.xml or http://www.turbosquid.com) if they desire.

The provided scene and avatar are:

- Scene: Sponza.zip (obj, mtl, textures (color, normal))

- Avatar: Avatar.zip (obj, asf, mtl, textures, weight map)

The system should work reliably and not crash. The allocated contact person for the team should submit the completed code and data to the ECS submission system with a Makefile, readme, and all the necessary data and instructions for testing. If the systems fails to build on the ECS machines marking penalties will apply.

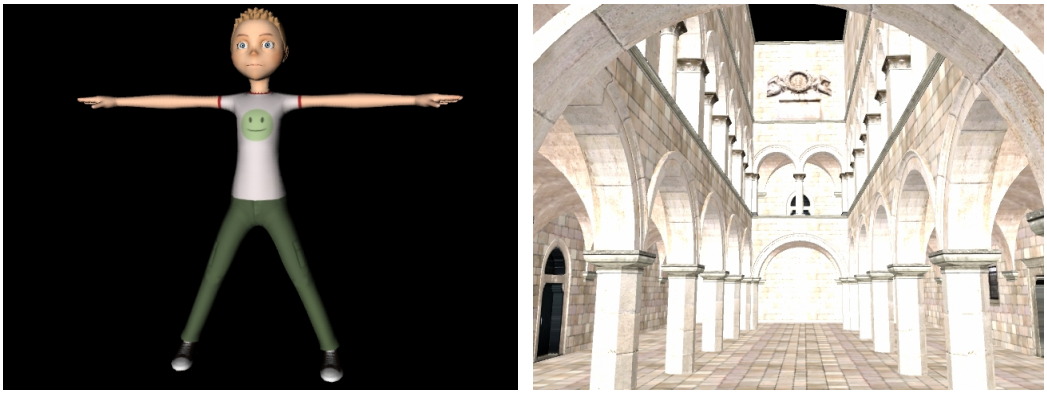

Figure 1: Renders of the provided files: avatar is the left and the sponze is the right

5. Individual Report

Due: 23:59 17th October 2013

Report: Every team member must submit an individual project report about what they did using the ECS submission system. The report should be about **maximum four A4 pages**, in 10 points font, pdf format including all images. It should detail what your team did, and specifically what you did and why it was technically difficult.

Each report must contain:

- Project title
- List of members
- Overall goal of the project
- A table showing a list of technical contributions and which member was responsible for each.
- Theoretical backgrounds and references
- Problem definition and technical challenges
- Description of solution and implementation; specifics on what you individually have contributed
- Results (snapshot of the important output)
- Conclusion and discussion
- Limitation and future works

Supporting Video: a supporting video that captures and showcase your real-time demo is highly recommended. The video should be less than 1min in proper format (avi, mpeg, or mov). The video file can be submitted to ECS submission system; accessible video link (e.g. YouTube) is also acceptable.

6. Marking

Total Marks: 100

Your marks for the project will be based on the following:

- **Group mark: 40**
  - **Proposal and progress (10)**
  - **System as a whole (30):** of which 10 is for integration if the team has more than one member
- **Individual mark: 60**
  - **Final project report**

Marking Session: we will set up marking sessions for your final project. Each team will come to show live demo. Each team will get exactly 10 minutes for three-person group, 8 minutes for two-person group, and 6 minutes for one-person group to setup, run demo, and take questions. The schedule will be announced during lecture after the mid-term break. Please note that attending marking session is not mandatory. However, it would be highly recommended for accurate marking. The marking session will be open to other students. If you want to see others work, welcome to visit whole session.