CS334/ECE30834: Final Assignment

Out: March 28th, 2022
Back/Due:
Project Selection – March 31st
Fast Forward – April 4-8
Demo Day April 27-29, 2022 (about 4 weeks total time)

Objective

Now that you have an understanding of basic computer graphics, the focus of this final assignment is to give you the freedom to extend your knowledge in a particular area of interest to you. Below are some suggested individual projects. Alternatively, you may also suggest your own project. All projects will be demonstrated to a public audience. The demo includes a live presentation of your project/program and a short PowerPoint presentation explaining how it works – schedule is TBD. Teams of 2 are allowed but only by explicit instructor approval. All pre-determined projects are "single person". If you propose a new project, you can choose to make it an individual project or a team project.

Please choose one of the projects below and fill out

https://docs.google.com/document/d/1bhDqBLayvxuC_XgE9SgWmNvJaNVQVYoO0yNTey1fbow/edit

by March 31st (11:59pm) indicating which project you will be doing – if not, I will randomly assign a project to you.

If you are doing an independent project, please fill out the above sheet AND send me an email with the keywords "CS334 Final Project" and then attached your one page project description that includes: 1) Motivation, 2) Expected Demo, 3) Main Technical Components. If it is a team project, then also 4) What else team member will do. Does not need to be a long verbose description. Bullets and diagrams preferred. Please note that ALL TEAM projects are essentially independent projects.

As a prelude to choosing a project, I recommend you browse through the papers (mostly PDFs) at http://kesen.realtimerendering.com, in particular for the SIGGRAPH, Eurographics, and ACM Symposium on Interactive Graphics conferences. Browsing through the papers might inspire you with regards to options. Most, if not all, of the below projects have previous works published at the aforementioned venues. Depending on your interest and project objectives, I can point you to several specific papers which you can read for background information.

Below are guidelines for the projects – I encourage all to go above and beyond the descriptions given. Grading will be based on how well you *implement* the project and *demonstrate* it (live demo and PowerPoint presentation), your *understanding* of the general concepts, and the *features and functionality* of your program. In other words, if you do something but don't know how it works or how to show it off, you get serious points off. Furthermore, you must implement the project in its entirety and no downloaded or previous implemented code is permitted *unless* explicitly approved by the instructor (I am not trying to discourage using software libraries but rather trying to prevent you from getting credit from what it implements ☺).

During April 4-8, and in class, each of you will describe briefly via PowerPoint slides a fast forward of your project. The purpose is to communicate your project briefly and to the point and to get initial feedback about its suitability, scope, and technical details. Such fast forwards are customary in top-level graphics conferences. This fast forward is a graded part of our project. Slots will be randomly assigned – more details as the date comes closer. You must email slides describing your project to the instructor by 9am on the day of the presentation (so that we can load-up the PC).

This final assignment will take effort and I vehemently encourage you to **start working on the assignment well before the due date** – i.e. today! As with the previous assignments, I prefer a nicely working, well-implemented and demonstrated project as opposed to a half-working buggy project with lots of half-baked complex features. Remember, you have only 1 month so measure your time carefully.

1. Procedural Modeling

This project is to develop a system for procedural modeling. You may choose areas of focus such as:

- procedural methods for 3D plants
- procedural methods for buildings,
- procedural method for roads,
- procedural method for cities, or
- procedural method for terrain.

The project should allow for a "grammar" to be written which can be then be converted to 3D geometry and rendered on the fly. The grammar is to be read, parsed, and applied. The resulting model is to be displayed using project tools developed in previous projects.

In class, we talked about multiple types of grammars and cellular automata. Your task is to choose a simplified version of one of these grammar options and to explore from there what you can do. The result should be something that you provide a "text file" as input with rules, terminals, etc. and thus new content can be generated during demo time. The model should NOT be hardcoded.

For pictures and inspiration about plants, take a look at: http://algorithmicbotany.org/papers.

For the same about urban modeling, take a look at:

http://www.esri.com/software/cityengine/

http://www.cs.purdue.edu/cgvlab/papers/aliaga/egstar09.pdf

http://www.cs.purdue.edu/cgvlab/urban/

http://www.iit.edu/~krawczyk/nks2003/

If you are interested, I have particular papers I could refer you too. In general, this project is very flexible and the objective is for you to focus on creating a way to develop something visually interesting: write some simple text description and generate nice 3D content – don't forget to have fun!

Cities/Buildings: you should generate 3D building-like structures
Roads: you should generate a 2D road network of a variety of styles, such as "New York", "Paris", etc.; you can use Google Maps, or OpenStreetMap.org, to get a view of the streets.

Terrain: you should generate realistic looking terrain, including color high-peaks in white, for example, and low areas in blue, for example, as water.

A key component is to provide some type of "control" of the generated content. Just generating "random" procedural content is NOT sufficient.

2. GPU Programming

This project is rather open and consists of developing a system that exploits GPU usage and programming. It can be a more sophisticated illumination algorithm (e.g., such as some form of Ambient Occlusion). It can be a physically based simulation (e.g., water). Or another idea.

3. Non-Photorealistic Rendering (NPR)

This project entails developing and implementing one of the NPR techniques discussed in class. The details can be found in the relevant papers. The tentative, but not all inclusive list, is the following:

- An overview: http://www.red3d.com/cwr/npr/

- Exaggerated Shading: implement something akin to the Exaggerated Shading by Rusinkiewicz et al. 2006 [http://gfx.cs.princeton.edu/pubs/Rusinkiewicz_2006_ESF/exaggerated_shading.pdf]

- Pen-and-Ink: implement some form of pen-and-ink using a precreated tonal map (e.g., textures of different density/darkness), Computer Generated Pen-and-Ink Illustration by Winkenbach et al. 1994 [ftp://ftp.cs.washington.edu/tr/1994/01/UW-CSE-94-01-08b.d/UW-CSE-94-01-08b.pdf]

- Contour/Silhouette: write a system to render the silhouettes/contours of arbitrary 3D models – you could require fine polygonal tessellation but the contours should be smooth; Coherent Stylized Silhouettes by Kalnins et al. 2003, Artistic Silhouettes: A Hybrid Approach by Markosian et al. 2000. [http://gfx.cs.princeton.edu/pubs/Kalnins_2003_CSS/kalnins2003css.pdf], or [http://graphics.cs.brown.edu/research/art/artistic-sils/artistic-sils-300dpi.pdf].

- Point-based Stippling (or Stipple Drawing). See the work of Floating Points: A Method for Computing Stipple Drawings by Deussen et al. in Eurographics 2000. [http://www.cs.princeton.edu/courses/archive/fall00/cs597b/papers/stipple_deussen.pdf]

- Drawing 3D Curves: See the work of Cohen et al. in I3D 1999, An Interface for Sketching 3D Curves. [http://graphics.cs.brown.edu/research/art/3dcurve.pdf]

4. Sand Appearance Editing

In this project, you will develop an appearance editing system for a sandbox. Details described in class. e.g., http://idav.ucdavis.edu/~okreylos/ResDev/SARndbox.

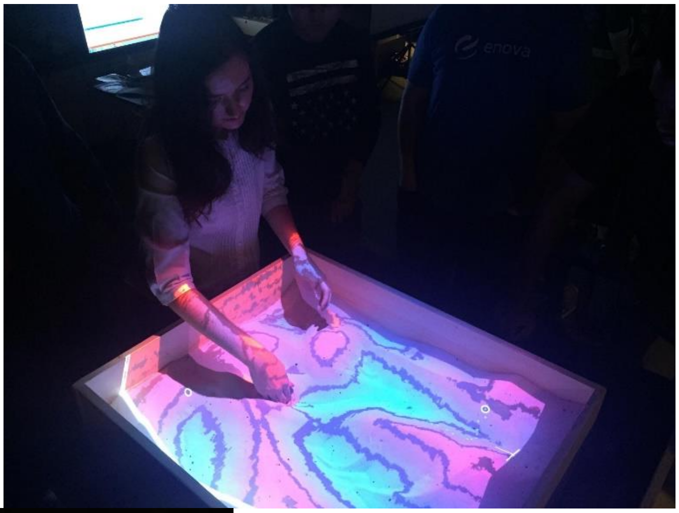
The project hardware (all provided by the instructor, no cost to you) is:

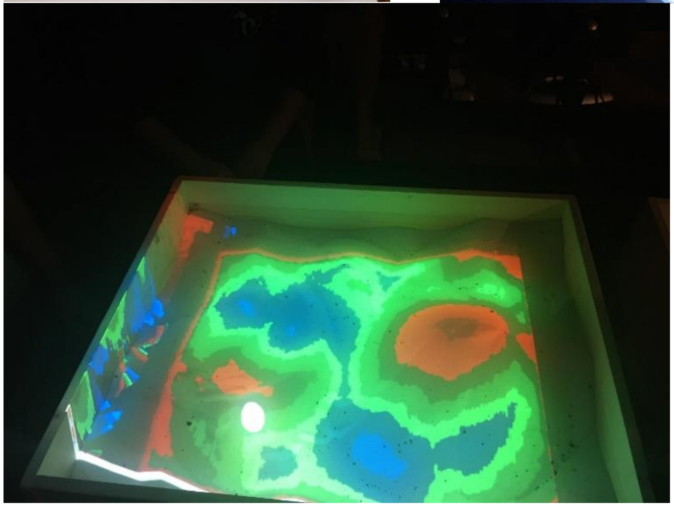
- a small sandbox with sand in the lab
- a Kinect
- a projector
- a camera, and
- a computer

The main tasks are:

- use the camera to register the relative locations of the projector, sandbox, and Kinect (this would be done during a setup phase)
- an OpenGL program that draws content to a window reacting to the different depth values you get from the Kinect
- extending the above component to some application: terrain drawing, beach ball game, etc, and
- having fun!

The instructor would assign you a graduate student experienced in this topic and in the used hardware to very quickly get it setup. It would require starting ASAP with the setup so that you can spend time on the application you choose. If lots of people are interested, we would decide on a suitable single or multiple team arrangement.





5. Your Own Project

If you wish to propose and present your own final assignment, that is fine, **but you must provide a project description by March 31st**. Your project description should be one-page long and should include the following sections:

a. title
b. motivation
c. what your demo will consist of
d. main technical components
e. (what each team member will do, if it is a team project)

If you make it a team project, each person should have a clear task in the project and students will receive independent grades. Please plan the project accordingly and include the distribution of tasks in the project proposal. Team projects can be at most 2 (two).

Let your imagination roar!!!

Demo Day

Final Assignment demo day is April 27 and 29, as discussed in class. Closer to that date we will setup a schedule via a democratic algorithm. Essentially, we will put up department-wide announcements, and have a demo-fest. Each student will present and demonstrate her/his assignment. You must also provide on that same day a key/link containing your source code, binaries, project and presentation. No extensions, no late penalties, no late passes, and no exceptions to this due date will be given.

If you have more questions, please see myself or the TAs. Good luck!