

TDT4230: Graphics and Visualisation

Final project

Bart van Blokland

Michael H. Gimle

Peder Bergebakken Sundt

Department of Computer and Information Science
Norwegian University of Science and Technology (NTNU)

March 2022

- **The final project counts towards 30% of your final grade.**
- You may work alone or in a team of two.
- Deliver an archive of your code and assets on *Blackboard* before the deadline.
- Ensure that your project can be built using CMake/Cargo, and that all necessary dependencies are included with your submission.
- If you're not using C++/Rust, make sure you include an easy to follow README to get your project running.
- If working as a team: we also want the git history. You can dump it using this command:
`git bundle create history.bundle HEAD`
The resulting `history.bundle` file includes all files tracked by git.

Introduction

The remaining course work of this course is in the form of a final project. Its intent is to investigate a more advanced or complex visualisation method in detail by implementing it yourself.

This document aims to outline what is expected, how it is graded, and suggests some projects you could take on (although we highly encourage you to come up with your own!).

You're allowed to do this project by yourself or together with someone else. However, it should be noted that in the latter case we expect a project that's approximately twice the amount of work to complete.

Important Deadlines

There are three deadlines related to the project:

13th of March

Submit a short description (2-3 paragraphs) of the project you want to pursue on blackboard. Your project must be approved before you can start to work on it.

18th of April, 23:59

Deadline to submit your code archive and final report.

TBA of April

Presentation day.

Support/feedback from the professor or TA during Easter (10th-17th of April) will be limited.

Grading

The project counts towards 20% of your final grade. If the exam is made digital due to COVID it will be worth 30%. This score is computed by the point sum of five distinct criteria:

- 5%: Originality and Ambition
- 9%: Implementation and Technical Achievement
- 7%: Final Report
- 8%: Final Presentation
- 5%: The “wow-factor”

Scores over 30 are clamped. This ceiling is 40 in the case of an online exam.

The criteria are outlined in detail below.

Originality and Ambition

Because you are allowed to choose your own topic, there can be variations in the complexity and originality of the problem setting across projects. Additionally, those working in pairs inherently have an advantage over those that do not.

This first criterion allows us to account for these differences. Its score is determined by your project topic proposal.

Implementation and Technical Achievement

This criterion measures the extent to which you achieved what you set out to do in your project proposal.

- The code archive you submit with the final report should contain all necessary dependencies.
- Like the assignments, your implementation should use OpenGL 4.0 or higher, and be written in either C++ or Rust (unless you have talked with me about doing something different).
- It should be possible to build the project using CMake/Cargo (it might be useful to use the Gloom project as a starting point here).

Final Report

The final report should document the process that went into implementing your project. It should give an idea of what you did to get to your final product, and what the greatest challenges were on the way. In at most 4 pages it should describe the following:

- Briefly describe your project topic.
- How does your implementation achieve its goal?
- What are some notable problems you encountered on the way? How did you solve them?
- What did you find out about the method in terms of its advantages, its limitations, and how to use it effectively?
- Briefly mention what resources did you used to learn about the technique. No need to include every link to everything you read, but I should get a general idea of how you figured it out, even if the answer ends up being pure experimentation!
- If working as a team: disclose who did what.

Final Presentation

The objective of the presentation is to show off your work to everyone. The exact times for the presentation are not fixed yet, as we're not sure if we have enough people submitting to require multiple days.

For now the plan is to have physical, adequately distanced presentations, but this will depend on the current situation in Trondheim.

The presentation itself should be at most 10 minutes long, and include the following:

- A description of how your chosen method works.
- A good amount of eye candy. Try showing situations where your method works well, and where it does not. A live demo would of course be awesome, but if you want to make sure things work, a video is great too.

How you present your work is entirely up to you. You can for instance use slides, videos, live demos, or combinations thereof. Live demos are highly encouraged, as they are often the most interesting things to look at!

The Wow-Factor

These points are given for things you do to polish your implementation or presentation, which in some way go beyond the bare minimum of getting your method working. This is because we want to incentive you to give this project your best shot, and to have lots of cool eye-candy on Presentation day!

Selecting a Project Topic

When it comes to selecting a project topic, the two most important criteria that we look for are:

- By far the most important criterion is that the project is not trivial. For example, there's no tutorial or ready-made code examples on the internet that tells you step by step how to go about it. It should also take some time to implement.
- It's not *very* important, but try to be original. It makes for a much more enjoyable presentation day when everyone did something different.

When in doubt, feel free to send me a mail and we can talk things over.

Project Ideas

Here are some scenes or subjects that could make for interesting projects:

- A cozy room with a fire burning in the fireplace
- Clouds that cast shadow over the terrain below
- An epic space battle with a hundred ships shooting at each other with lasers
- Portals, as in the game “Portal”
- A tree whose leaves and branches oscillate in the wind
- Large ocean waves hitting a rocky coastline
- A sunset on the beach with a giant lens flare
- A walking character leaving bootprints in the snow
- A black hole in space showcasing spatial distortion

Here are some keywords that could point you in interesting directions:

- Ray Marching and/or Constructive Solid Geometry
- Fractal rendering
- Volumetric rendering or simulation
- Parametric Surfaces
- Subsurface scattering
- Deferred shading
- Physics- or swarm-based particle systems (i.e. boids)
- Ray Tracing

Your imagination is the limit! This project is all about doing something interesting and fun and showing it off to everyone!

You can also find videos showing projects from previous years on Blackboard under “Extras” or here: <https://www.idi.ntnu.no/grupper/vis/teaching/>. These can also give you some idea of the scope of this project, but do not be discouraged by those who went above and beyond.