

A5 Project Proposal
Title: Confined
Name: Alex Klen
Student ID: 20372654
User ID: ayklen

Final Project:

Purpose :

To create a game that utilizes and exhibits modern OpenGL shader techniques.

Statement :

You find yourself confined to a room and need to escape. The focus of this project is on the graphical techniques, and some amount of gameplay and interactivity will be implemented if time permits. The game will take place in a scene which exhibits many of the modern techniques expressed in the objectives. It will extensively use lights and (dynamic) shadows and run at an interactive framerate.

The goal is to implement several algorithms using shaders and OpenGL 3.3+ (that all work together). These include shadow mapping, deferred rendering (to support many dynamic light sources), texture and bump mapping, an approximation algorithm for ambient occlusion, and planar mirror reflection.

This project is interesting and challenging because it involves using several advanced shader techniques at the same time, and the game itself will hopefully look good and be interesting to play.

I will learn how to effectively use GLSL on modern hardware to achieve quality light and shadow rendering at interactive framerates.

Technical Outline :

This project will require implementing several algorithms, all of which have to do with using shaders written in GLSL. OpenGL 3.3+ will be used for its efficient buffers, custom rendering pipeline, and multiple render targets (required for deferred shading).

One of the most important objectives is to implement deferred shading. Deferred shading is a method of shading that speeds up subsequent lighting and shadow depth passes by letting them perform per-pixel lighting calculations only instead of computing them for all geometry and all fragments [2]. When forward-rendering, the vertex shader will compute lighting parameters (incident, normal, colours) which will be interpolated for each fragment. Deferred shading, however, computes geometric parameters for each pixel and writes them to a textures for later use. Subsequent shading passes can then sample these textures and operate per-fragment only. The benefit of this is clear when you consider shadow mapping with many lights, since when forward rendering you would need to run the shadow mapping algorithm over all geometry for each light and each fragment, whereas with deferred shading you only need to do it for each fragment. Although there are many shader passes the memory bandwidth of each is quite small.

The other crucial objective is shadow mapping. This algorithm is well-known and consists of a first pass where geometry is rendered from the perspective of a dynamic light source, and only a depth buffer is written [6]. This is done for each light. Then a final pass can perform lighting calculations based off of whether the camera's perspective sees pixels that are farther or closer than the first surface each light sees is. This algorithm is compatible with shadow-mapping and in fact allows you to use many dynamic light sources because they only have to pass over each fragment. There are also extensions to shadow mapping that eliminate visual artifacts present in the basic implementation and allow more realistic penumbras [1] [3].

Another interesting objective is to apply an ambient occlusion (approximation) algorithm in shaders. There is more than one fast algorithm for this, including Screen Space Ambient Occlusion [4][5].

A few other objectives:

- Bump/normal mapping, implemented in a way compatible with deferred shading (perturbed normals can be written in the first pass).
- Planar mirror reflection - rendering to texture the scene once from the perspective of the reflected ray from the camera's perspective, and then rendering again from the camera's perspective after applying the rendered texture to the mirror.

- Basic keyframe animation - implemented by using multiple mesh files with the same corresponding data (can be exported by modelling applications) and performing interpolation between them.

Bibliography :

- [1] Rouslan Dimitrov. Cascaded shadow maps. *Developer Documentation, NVIDIA Corp*, 2007.
- [2] Michal Ferko. Real-time lighting effects using deferred shading.
- [3] Randima Fernando. Percentage-closer soft shadows. In *ACM SIGGRAPH 2005 Sketches*, page 35. ACM, 2005.
- [4] V KAJALIN. Shaderx7. charles river media, march 2009, ch. *Screen Space Ambient Occlusion*, pages 413–424.
- [5] Martin Mittring. Finding next gen: Cryengine 2. In *ACM SIGGRAPH 2007 courses*, pages 97–121. ACM, 2007.
- [6] Lance Williams. Casting curved shadows on curved surfaces. In *ACM Siggraph Computer Graphics*, volume 12, pages 270–274. ACM, 1978.

Objectives:

Full UserID: ayklen

Student ID: 20372654

- ___ 1: Unique scene that shows off dynamic shadows with many light sources.
- ___ 2: Interactive first-person controls and interaction with environment (UI).
- ___ 3: Deferred Shading pipeline.
- ___ 4: Texture Mapping.
- ___ 5: Bump Mapping.
- ___ 6: Shadow Mapping.
- ___ 7: Ambient Occlusion approximation in shader.
- ___ 8: Sound synchronized with game.
- ___ 9: Planar mirror reflection using rendering to texture.
- ___ 10: Keyframe Animation.