Gerard Geer

gerarddgeer@gmail.com
gerardgeer.com
github.com/gerard-geer
(817) 372-8286
8641 San Joaquin Trl, Fort Worth, Texas 76118

Seeking a software development co-op or internship for Spring, Summer or Fall; any location.

Education

Tarrant County College
(Fort Worth, TX)
2014-2016

Attending for an Associate of Science, with aim to transfer to a four-year institution.

Rochester Institute of Technology
(Rochester, NY)
2011-2014

Attended for B.S. in Computer Science. Have since transferred to TCC. I remain an active Computer Science House member.

St. Hugh's College
(Oxford, Oxfordshire, UK)
2010

St. Hugh's College Summer Programme with a focus topic comparing global illumination effectiveness and practicality in traditionally rasterized and ray-traced computer graphics paradigms.

Select Coursework

Computer Graphics	Calculus / Differential Equations	CS Theory
Data Structures	Linear Algebra	Object Oriented Programming

Skills

OpenGL + WebGL	GLSL	Ray-Marching/Tracing	Procedural Texturing + Content Generation
Web Development (frontend/backend)	Python	Java	C/C++

Projects/Hobbies

Fragment Shader Lunar Lander Clone - http://spacelander.gerardgeer.com
I wrote a clone of Lunar Lander that runs entirely in four fragment shaders. All content (including text glyphs) is generated procedurally or with distance fields, and state is stored in a framebuffer that is spared from vblank.

Procedural *Gimmick!* Scene - http://gimmick.gerardgeer.com
For SIGGRAPH 2015's "Favorite Game/Movie moment" WebGL shader hackathon, a friend and I decided to recreate a scene from the Famicom game *Gimmick!.* The shader procedurally generated and animated the art from the scene, as well as emulated its music. For the soundtrack my friend transcribed the score to GLSL using functionality I wrote to emulate the 2A03 and Sunsoft 5B.

Other Fragment Shader Debauchery - http://gerardgeer.com/shaders
I've written a variety of other shaders as well. Many are real-time ray-tracers, exploring procedural texturing and distance field generation, and secondary ray applications (refraction and hard/soft reflection, as well as secondary lighting). Others work in 2D, such as generating procedural text, and efficiently storing state between frames in framebuffers.

CSH 3D Map - http://map.gerardgeer.com/
A web application written in JavaScript using WebGL and a bit of JQuery UI that functions as an interactive 3D map of RIT's Computer Science House (CSH) dormitory floor.

CSH LUMA (LED User Manipulated Apparatus) - https://github.com/gerard-geer/LUMA
An in-progress distributed LED ambient lighting system for use in hallways, under furniture, etc. Provides users with the ability to fine tune lighting patterns through a responsive web interface. Leverages Angular.js for user interaction, and uses Flask for its backend and provide a RESTful API. Lights themselves are controlled via Raspberry Pis.

RenderSprite - https://github.com/gerard-geer/RenderSprite
An OpenGL sprite library I've been working on to provide several hard-to-find features such as render-to-sprite, palette functionality (color replacement, per-scanline palette-splitting), and PNG support.

SOAP (Shower Oriented Audio Player) - https://github.com/gerard-geer/Detergent
A project that uses a Node.JS server, a Redis cache, and Python to store and play playlists of music over a P.A. system installed in the showers of CSH.

ASCIIGL
A simple command line graphics library that allows tailored and positioned drawing of text and various shapes to a terminal window. Akin to NCurses with structure naively similar to OpenGL, this project was done in Python as part of a silly ruse and remains one.