Zachary Klein

(803) 226-1009 • Aiken, SC • ztklein@lavabit.com • https://ztklein.dev • linkedin.com/in/ztklein • github.com/ZTKlein

References available upon request

EDUCATION

Clemson University | Master of Science in Digital Production Arts | GPA 3.64 August 2020 - May 2022
Clemson University | Bachelor of Science in Computer Science | GPA 3.57 August 2016 - May 2020
Relevant Coursework: Production Pipeline Development | Studio Production (Pipeline lead) | Computer Graphics Images | Computer Graphics | Physically Based Modeling and Animation | Advanced Computer Graphics | Physically Based Visual Effects | GPGPU | Rendering and Shading | Human-Computer Interaction | Surfacing | Visual Effects & Compositing |Virtual Reality Systems | Data Structures and Algorithms | Software Development Foundations | Network Programming | Operating Systems | Programming Systems

WORK EXPERIENCE

Software Engineer August 2020 - December 2021
Clemson University | Charleston, SC

- Worked with Dr. Tessendorf to develop a C++ API for Project Gilligan (software primarily used to generate and simulate ocean surfaces and waves) to allow transfer of ocean surface and cloud volume information to other software for rendering (Navy contract)

Web Developer May 2019 - May 2020
Clemson University | Clemson, SC

- Worked with a team developing tools to visualize and analyze millions of data points from sensors around campus (utilities, air quality, etc.)
- Designed and implemented a front-end web dashboard, including tools such as a campus heatmap and alerts system
- API design and implementation
- Containerization of the backend

Engineering Intern - Debugger May 2018 - Aug. 2018
Parsons | Aiken, SC

- Debugged complex training & simulation software built on multiple interacting systems
- Identified and proposed solutions for over twenty bugs

Data Structures and Algorithms Teaching Assistant Jan. 2017 - Dec. 2017
Clemson University School of Computing | Clemson, SC

- Coordinated with professors in the department for the data structures and algorithms course to help students in the lab
- Explained the language (C++) and fundamental concepts to students during lab and office hours
- Graded and debugged students' code

Software Development Intern - Frontend Web Developer May 2017 - Aug. 2017
Savannah River Nuclear Solutions | Aiken, SC

Developed web applications to replace legacy software for internal use.

- Front-end programming with Angular, Bootstrap/CSS, and HTML
- Some modification of the existing API built with SpringBoot when necessary
- Optimized the code for one API endpoint to be more than 10x faster (page load time went from minutes to seconds)

NOTABLE PROJECTS

Pipeline Lead

- Clemson University DPA Student Production – Charleston Campus
- Set up pipeline for the team (ShotGrid for asset management, Tractor for the renderfarm)
- Migrated lab machines from Ubuntu to Rocky Linux, including development of automation for headless setup and installation/testing of all production software
- Set up user scripts for team members

Complex BxDF Models

- Used PBRTv3 framework for the code
- Compared several microfacet models rendered using geometry created using a tool I wrote vs microfacet BSDF's I coded
- Modeled single layer thin-film interference (outer medium, layer, substrate). Also created a model that allowed for multiple layers of varying thickness.
- Modeled diffraction using a custom tool to generate texture for orientation of tangent vector on surface and spectral BRDF code

Implementing Volumetric Renderers

- Implemented ray marching algorithms using a variety of methods and languages – offline C++ renderer, OpenCL renderer, OpenGL renderer
- Basic SDF editor program written in C++
- Real-time SDF viewer in Unreal using custom nodes (HLSL)

Machine Learning Models for Programming Language Classification

- Compared several custom models for identifying code languages from snippets (no need for file extension)
- Primarily used Python with Scikit-Learn and Tensorflow
- Best model had ~96% accuracy across 55 languages (used IBM's CodeNet dataset)

SKILLS

Programming languages/libraries/frameworks: C | C++ | C# | OpenGL | GLUT | GLSL | HLSL | Java | HTML | CSS | JavaScript | TypeScript | Angular 2+ | Prolog | React | RxJS | Node.js | Python | Tensorflow | Spring Boot | Express | PHP

Programming concepts: Software design patterns | Data structures | Object-oriented, functional, rule-based, and declarative programming paradigms | Real-time and offline rendering techniques

Operating Systems: Windows | Linux | macOS

Technology/Software: AWS | Git | Unix toolkit | GCC | Clang | Valgrind | GDB | SVN | Mercurial | Maya | Unity | Unreal Engine | MS Office | Nuke