Gregory **Du**

2444 Virginia Street Apt. A, Berkeley, CA 94709

📱 (978)-846-2178 | ✉ 17gregoryd@gmail.com | 🏠 gregoryd2017.github.io/home/

Education

University of California, Berkeley *Berkeley, California*

Expected Degree: B.A. Computer Science (Spring 2021) *August 2017 - Present*

Relevant Coursework: Computer Graphics, Operating Systems (FA '20), Algorithms, Data Structures, Discrete Mathematics and Probability Theory, Computer Security, Computer Architecture, Intro to Computer Vision and Computational Photography (FA '20), Structure and Interpretation of Computer Programs, Linear Algebra

Skills & Abilities

- C++, Python, C, Java, GLSL | OpenGL, Nvidia Nsight, Qt Framework, HOOPS Visualize | LaTeX, Source Control (Git & Perforce), Windows and Linux development.
- Industrial Software Engineering (functional & design spec., coding, debugging & regression testing).

Work Experience

Cadence Design Systems, Inc. *May 2020 - August 2020*

Software Engineering Intern

- Returned to Cadence to work on an overhaul of Allegro's 3D Canvas, focusing on augmenting performance in 3D.
- Implemented support for the Reversed-Z method with a high precision 32-bit depth buffer to solve the Z-plane fighting issues present with PCB data (C++ and OpenGL).
- Improved frame render time during selection operations by adding support for pixel buffer objects to allow for asynchronous data transfer with DMA (C++ and OpenGL).
- Implemented an Arcball Camera for more intuitive CAD style navigation in 3D space (C++).
- Reworked default shader to include diffuse and specular light calculations for more natural scene lighting (GLSL).

Cadence Design Systems, Inc. *May 2019 - August 2019*

Software Engineering Intern

- Developed a variable highlighting mechanism for 3D Canvas in Allegro, Cadence's proprietary PCB design software. New functionality allows the end user to choose from various visual effects to apply to elements in the 3D Canvas. Created an accompanying GUI (Qt in C++ & HOOPS Visualize 3D rendering).
- Developed a machine learning procedure in Python to predict various performance metrics in Allegro's 3D Canvas. Implemented a dense neural net using Keras, a data collection mechanism, and a storage database to accurately predict the load time using the 3D Canvas.

Projects

Please visit my website (https://gregoryd2017.github.io/home/) for my complete portfolio, including in-depth write-ups and walkthroughs of many of these projects.

- Implemented a pathtracer in C++ to support ray intersection computation, BVH construction, direct illumination, global illumination with Monte Carlo estimation, and adaptive sampling. Additionally, encoded depth of field simulation and material BSDFs including diffuse, mirror, and glass.
- Extended the pathtracer to handle volumetric scattering in homogenous media based on the paper by Lafortune and Willems, as well as the PBRT participating media implementation guide.
- Worked on an APIC fluid simulation extension on the PIC/FLIP fluid simulation originally written by Robert Bridson, and later augmented by Ante Qu. Rendered frames in Pygame, and stitched together an animation using OpenCV.
- Implemented features in a rasterizer for scalable vector graphics including in-triangle tests for basic triangle coloration, antialiasing, SVG transformations, color interpolation using barycentric coordinate calculation, pixel sampling methods for texture mapping including nearest pixel sampling and bilinear sampling, and level sampling methods using mipmaps.