

# NICHOLAS DRIAN

ndrian@berkeley.edu | github.com/nicholasdrian | +1 (530) 386-8800 |

## WORK EXPERIENCE

---

### DPS Telecom

Fresno, CA

*Embedded Systems Programmer*

Aug 2023 - Oct 2023

- Developed custom products for monitoring alarm systems within remote networks.
- Updated front end and back end web applications to support newer generations of servers and remote telemetry.
- Trained clients to use our servers and monitoring devices.

### BMW

Munich Germany

*Software Engineering Intern*

Nov 2022 - Apr 2023

- Explored the graphics pipeline of BMW's driving simulator for possible performance and fidelity optimizations.
- Moved skeletal mesh animation onto the GPU using bone animation textures to help alleviate a CPU bottleneck.
- Wrote an Unreal Engine utility for automating the generation of high performance MetaHuman animations.
- Created an application for generating, previewing and benchmarking GPU based animation from FBX files.
- Achieved 2x framerate improvement on some animation intensive benchmarks.

### UC Berkeley

Berkeley CA

*Computational Design Instructor*

Spring 2021

- Created and taught a graduate course on parametric design using the plugin Grasshopper for Rhino.
- Presented a series of 10 lectures demonstrating a streamlined architectural workflow through CAD scripting.

## EDUCATION

---

### University of California, Berkeley

Graduated May 2022

*Bachelor of Architecture*

- **Computer Science GPA:** 3.85/4.0
- **Relevant Coursework:** Data Structures (A+), Logic (A+), Computer Architecture, Computer Graphics, Computational Structure in Data Science, Computer Security (audited), Operating Systems (audited)

## PROJECTS

---

### CAD - 3D NURBS Modeling Application

*Languages: Rust, Typescript, C++, GLSL, WGSL*

- Wrote 3D NURBS modeling software, first in C++ using OpenGL, then in TypeScript using Web GPU, then in Rust using wgpu.
- Check out <https://nicholasdrian.github.io/Web-NURBS/> for more about the project.

### Path Tracing

*Language: C++ | APIs: OpenGL, glm*

- Created path tracing software that supports global illumination, reflection, refraction, dispersion, textures sampling with trilinear mip-map interpolation, and material properties such as specular, diffuse, roughness, and more.
- Achieved unbiased results through monte carlo integration while importance sampling both the BRDF and lighting.
- Used photon mapping and statistical error metrics to support dispersion with a parametric error threshold.

### Home Slice - 3D Printing Software

*Language: Java | APIs: Swing, AWT*

- Implemented a desktop application that builds printable .gcode from an .stl file and various printing parameters.
- Features a fully maneuverable, real time rendering of the print path and source mesh
- Capable of rasterizing a 20,000 face mesh at 20 fps on the CPU. (From before I learned APIs such as OpenGL)
- Optimized the program to process large mesh inputs using kd-trees.

## SKILLS:

---

C++, C, Python, Javascript, Java, GLSL, WGSL, RISC-V, OpenGL, WebGL, Vulkan, Lua, 3D-Modeling, Rendering, 3D-Printing, Animation, Algorithms, Competitive Programing, Rock Climbing, Linear Algebra.