Nathan Hutton

801-560-3611 | nathan.d.hutton@proton.me | linkedin.com/in/nathanhutton | github.com/nathan-hutton | YouTube

Education

University of Utah — Dec. 2025
M.S. Computer Graphics & Data Visualization — *GPA: 4.0*

Westminster University — May 2024
B.S. Cohputer Science, Minor in Applied Mathematics — *GPA: 4.0*
- Awarded outstanding computer science student of the year

Salt Lake Community College — May 2021
A.S. General Education — *GPA: 4.0*
- Earned degree while in high school

Experience

Software Engineer Intern — May 2025 – Aug. 2025
Flight Safety International — *Broken Arrow, OK*
- Integrated Phidgets API with RACKtangle hardware for real-time flight simulation I/O in C++
- Collaborated in an AGILE team environment. Worked exlusively in Windows and Visual Studio

Software Engineer Intern — Feb. 2023 – May 2024
Idaho National Laboratory — *SLC, UT*
- Debug and reformat C++, C, and Python code
- Convert Matlab test harness to Python
- Configure CI/CD pipelines
- Implement, refactor, and debug GUIs
- Worked exclusively on Linux

Computer Science Tutor/TA — Aug. 2022 – Dec. 2023
Westminster University — *SLC, UT*

IT Technician — Aug. 2021 – May 2024
Westminster University — *SLC, UT*

Projects

Boids Flocking Simulation | *C++, OpenGL, ImGui* — GitHub | Video
- A simple ruleset by Craig Reynolds leads to complex flocking behavior
- Extensive customization and debugging implemented with ImGui

Solar System | *C++, OpenGL* — GitHub | Video
- Dynamic physics, shadow maps, bloom, and verlet numerical integration

Volme Renderer | *C++, OpenGL, Glui* — GitHub | Video
- Ray marching in GLSL with modifiable transfer functions

AQI Visualization | Javascript, D3, JSON — GitHub | Video
- Interactive visualization for Utah's air quality index
- I made 63% of all commits

Squibblets | C#, Unity, Firebase, AGILE — GitHub
- 4-person team project. I made the gameplay loop, online leaderboard, UI, and 54% of all commits

Mass Spring System | C++, OpenGL, Eigen — GitHub
- Simulate a mass spring system made up of 8,000 tetrahedrons in real time

Ray Tracer | Python — GitHub
- No reliance on libraries, everything made from scratch

Technical Skills

Languages: C/C++, Python, Java, C#, JavaScript, HTML/CSS, Latex
Developer Tools: Linux, Git, Docker, Vim, VS Code, Visual Studio, PyCharm, IntelliJ
Libraries: OpenGL, NumPy, pandas, ImGui, Matplotlib, Eigen