Liam Strand

github.com/liam-strand | strandliam@gmail.com | (408) 438-1125

Education

Master of Science in Computer Science, Northwestern University, GPA: 3.95/4.00 **2024 – 2026**
- Coursework: Compilers, CUDA, Distributed Systems, Internet-Scale Experimentation, and Kernel Programming
- Spring 2026: Parallel Architectures
- Thesis in progress: *Adaptive Profiling via ML-Driven Hardware Counter Orchestration*

Bachelor of Science in Computer Science, Tufts University School of Engineering, GPA: 3.97/4.00 **2020 – 2024**
- Graduated **summa cum laude**; Dean's List all 8 semesters
- Concentrations: Operating Systems, Concurrent Programming, Parallel Computing, and Computer Architecture

Academic/Research Awards
- Computing Research Association **Outstanding Undergraduate Researcher Award**, Honorable Mention
- **David Krumme Award for Experimental Computer Science** for meritorious research and projects of practical worth
- **TUAA Senior Award** for outstanding leadership, significant academic achievement, and extraordinary service

Skills

Languages: Assembly, C, C++, Chisel, Erlang, Go, Java, LaTeX, Python, RISC-V, Rust, Swift, Unix/Shell Scripting
Technologies: AWS, Docker, `gdb`, `git`, GitHub, GitLab, GPU/CUDA, Jira, Jupyter, Linux/Unix, PyTorch, TCP/IP, `valgrind`
Methods: Agile, Concurrent/Parallel Programming, Hardware Bringup, Operating Systems, Profiling, Test Development

Experience

Distributed Systems Engineering Intern | Bosch Robotics | Sunnyvale, CA **Summer 2025**
- Transformed FogROS into FogX, a production fog robotics platform for swarms and cloud compute offloading
- Rearchitected app with pluggable protocols, observability, and CI/CD stacks to scale fleets and resolve security vulnerabilities
- Integrated SCReAM congestion control to optimize network utilization and reduce cloud offload latency
- Delivered AI-ready interfaces and regression suites enabling autonomous agents to propose and validate changes
- Implemented identifier-based addressing and ROS2 and TCP adapters to expose topics across networks without app changes
- Approved all changes to the Rust codebase, conducted code reviews, and mentored interns on parallel projects

Researcher | Prescience Lab, Northwestern University | Evanston, IL **Oct 2024 – Present**
- Architected complete synthesis pipeline to translate high-level parallel programs (NESL) into custom hardware accelerators
- Reduced floating-point virtualization overhead by 600× via redesigned Linux exception handling with Chisel
- Implemented cross-stack changes to hardware, firmware, bootloader, Linux kernel, and C runtime for correctness and performance
- Peer Mentor for ≈100 Operating Systems students; taught batch and real-time scheduling policies and driver development

Software Engineering Intern | AeroVironment, Inc. | Wilmington, MA **Summer 2023**
- Demonstrated cost of an aerial robotics platform can be dramatically reduced (≈30%) by transitioning to COTS hardware
- Ported large (>100k LOC), high-performance C++ applications to new distributed hardware environment using `systemd`
- Led hardware bringup and software integration effort. Created containerized automated development pipelines
- Refactored host/device communication protocol to modernize design and improve resiliency
- Debugged complex radio kernel driver compatibility issues; collaborated with hardware vendor engineers to resolve

Researcher | Bioinformatics and Computational Biology Group, Tufts University | Medford, MA **Jan 2021 – Oct 2023**
- Presented published manuscript (first author) at the Metadata and Semantics Research Conference in Milan, Italy
- Led multi-institutional team to deliver language support features, dramatically improved developer ergonomics for end-users
- Implemented client/server architecture in Python, with a well-documented API to support new language features and editors
- Leveraged beta tester feedback and automated regression testing frameworks to ensure a polished and resilient product
- Awarded Fowler Family Summer Scholar Grant to support independent, full-time research

Teaching Assistant | Department of Computer Science, Tufts University | Medford, MA **Sep 2021 – Jun 2024**
- Collaborated with faculty to develop course infrastructure, assessments, and assignments, impacting >1000 students
- Co-wrote course materials and conducted oral exams for concurrent and parallel programming courses
- Designed and led labs and review sessions for C and C++ machine structure and assembly language courses
- Taught performance analysis and debugging strategies using tools including `gdb` and `valgrind`

Hither Creek Boatyard | Data Collection/Analysis | Madaket, MA **Summer/Winter 2018 – 2021**
- Designed Yamaha outboard engine data collection methodology. Gathered, analyzed, and synthesized data for 18 engine models.
- Created accessible Yamaha outboard engine diagnostic charts. Recovered and transferred GPS chartplotter data.

Publications

Liam Strand et al. (2024). "Context-Sensitive Editing for the MEDFORD Metadata Language". In: *Metadata and Semantic Research*. Springer Nature Switzerland, pp. 278–283. ISBN: 978-3-031-65990-4.

Karl Hallsby, Liam Strand, and Peter Dinda (2026). "Hardware-based Kernel-Bypass Exceptions to Accelerate Floating Point Tracing and Virtualization". In review for HPDC '26.

Projects

Tactile Spatial Audio System: CyberGrape
- Led development of audio mixing tool using Rust, Raspberry Pi microcontrollers, and directional Bluetooth antennae
- Created flexible build system tooling to manage seamless cross-compilation and foreign library binding generation
- Developed hardware, firmware, and software solutions; managed scoping, planning, and team delivery of all MVP targets

Text Sentiment ML Classifier: Senti-Mental
- Designed neural network AI model to classify product and film reviews as positive or negative with ≈90% accuracy
- Project scope included using `numpy`, `scipy`, and `sklearn` for preprocessing, training, and hyperparameter search

RISC-V Processor
- Designed single-cycle RISC-V core in SystemVerilog, including control, datapath, and scaffolded memory subsystems
- Simulated and tested design for adherence to the RV32I specification using Verilator

Assembly Emulator: VM Profiling
- Built a simulated architecture emulator in C, and vastly improved performance using modern profiling techniques (Demo)