

Taylor Tam

(650) 269-0964

taylor52@stanford.edu

Palo Alto, CA

Education

Stanford University — B.S. Computer Science (AI Track), Minor: Electrical Engineering August 2024 - present | 3.7 GPA
Courses: Deep Learning for Computer Vision (grad course), Machine Learning (grad course), AI-Enabled Robotics, Operating Systems and Principles, Computer Organization and Systems, Infrastructure at Scale, Linear Algebra and Multivariable Calculus, Applied Matrix Theory, Probability & Statistics

Skills: C++, Python, C, Rust Swift, TypeScript, SQL | PyTorch, Metal Shading Language, CUDA (familiar), OpenCV, RLib, GCP, Docker, Git

Y Combinator AI Startup School & Summer Conference Invitee, June 2025 - July 2025

Experience

Aida — SWE Intern, June 2025 - September 2025 | Python, SQL, GCP, TypeScript

- Designed, developed scalable, **distributed backend services** in Python for CRM data pipelines, meeting prep automation, and email entity search, enabling seamless data flow across Google Calendar, internal systems, and microservices
- Improved backend reliability by **resolving production-critical issues** (timestamp drift, calendar crawl redundancy), implementing field-stage dependency handling, and automating workflows for deal tracking and pipeline filtering
- Oncall lead responsible for **deployments, hotfixes, canary releases**; monitored cloud logs/queues, triaged user-facing issues

The Shinozaki Lab, Stanford University — Machine Learning Researcher, April 2023 - August 2024 | Python, PyTorch, ML, Swift

- Led research on EEG-based **delirium detection CNN** for novel medical device, cleaned and preprocessed EEG data
- Integrated trained model into **iOS app** for clinical testing and real-time inference

Stanford Center for Artificial Intelligence and Imaging — Software Intern, May 2022 - September 2023 | Python, PyTorch, ML

- Trained ResNet-50 CNN via transfer learning to detect medical devices in chest X-rays, achieving **97% accuracy**
- Presented at [Stanford AI Symposium](#), authored publication

The Melcher Lab, Stanford University — Software Engineer, May 2020 - September 2025 | Python, GCP, Swift, Firebase, ML, C#

- Co-authored [publication](#) for GPU-accelerated **embedded device** automating liver tissue analysis, **100x latency reduction**
- Co-developed CVAI model for steatosis quantification from liver images, Python-based **image preprocessing algorithm**
- Built **Jetson GPU inference system** with RESTful JSON API and iOS frontend for clinical use; scaled processing with **Google Cloud Functions** (currently in legal review with Stanford University for distribution)
- Developed virtual reality suturing simulation on Oculus and Vision Pro using Unity

Projects & Research

Benchmarking Multi-Agent Coordination in Flow-SUMO Vehicle Platoons — Current | Python, RLib, Flow-SUMO, PyTorch

- Implemented **centralized training-decentralized execution (CTDE)**, independent PPO agents w/ RLib, Flow-SUMO simulating multi-lane platooning and V2X communication
- Engineered **continuous-control policies** for acceleration and lane selection; evaluated emergent stability
- Built a **PyTorch-based analysis pipeline** for synchronized rollouts, statistical testing (t-tests, Levene's), and trajectory visualization

Rust Based Shape-Safe Computer Vision Library for Apple Silicon — January 2025 - Current | Rust, Metal Shading Language

- Building open-source computer vision library in Rust, focused on compile-time shape safety and hardware acceleration
- Designed a **type-level tensor system** using const generics for compile-time shape enforcement
- Integrated **Metal Performance Shaders (MPS)** on Apple Silicon to implement fused GPU kernels; near-peak throughput
- Developed a **backend-agnostic architecture** via Rust traits, hot-swappable CPU/GPU execution, no runtime overhead

SoleMate (Stanford TreeHacks) — January 2025 | Swift, GCP, LiDAR, 3D modeling

- [Built app in Swift](#) using LiDAR and ARKit to capture 3D foot scans and generate custom-fit orthotic insoles
- Implemented **3D modeling pipeline**, including mesh reconstruction, alignment, and Laplacian smoothing for fit

Gesture-Controlled Mario Kart — January 2024 - June 2024 | YOLO, OpenCV, Python, ML, Bluetooth HID Emulation

- Designed and deployed a real-time computer vision + ML pipeline replacing Nintendo Switch JoyCons with hand-gesture control
- Developed a **YOLOv8, ORB pipeline** for skeleton detection, rotation/scale-invariant gesture matching, achieving **165ms latency, 30FPS**
- Reverse-engineered Nintendo Switch **bluetooth HID protocols**, implemented JoyCon emulation on Nvidia Jetson Orin
- Optimized control flow w/ **custom HID report packets**, supporting simultaneous joystick, button input for improved gameplay