Jiayu Yang

jarrodyang227@gmail.com | U.S. Permanent Resident | github.com/jarrod227 | linkedin.com/in/jiayuyang227

Education

Carnegie Mellon University | Pittsburgh, PA, USA
M.S. in Biomedical Engineering, Research Track | Expected May 2028

Rutgers University | New Brunswick, NJ, USA
B.S. in Electrical and Computer Engineering | Minor in Mathematics | May 2026
GPA: 3.618 / 4.000 (Major: 3.804 / 4.000)
Relevant Coursework: Digital Signal Processing, Probability & Random Processes, Machine Learning, Deep Learning
Awards: Easton Elmer C. Scholarship (Jan 2025); Dean's List, all eligible semesters

Skills

- **Embedded Systems:** STM32 firmware (HAL, FreeRTOS), bare-metal C/C++, CAN/CAN FD, SPI, peripheral interfacing
- **Signal Processing:** Digital filtering, spectral analysis, sensor fusion, biosignal processing
- **Machine Learning:** PyTorch, deep learning (GRU/CTC), classical ML (SVM/LDA), time-series modeling
- **Robotics & Vision:** ROS 2, MoveIt 2, PD joint control, computer vision (OpenCV)
- **Hardware:** Analog front-end design, PCB design (Altium/KiCad), circuit simulation & debugging
- **Programming & Tools:** Python, MATLAB, C/C++, Git, Linux/Bash

Research Experience

Neuromuscular Recording Hardware
Neuromechatronics Lab (NML), Carnegie Mellon University | Advisor: Douglas Weber | Graduate Researcher | Sep 2026 – Present

- Designed a 32-channel high-density electromyography (HD-EMG) flex PCB in KiCad, integrating an 8×4 electrode array, 300 mm flex lead, and Omnetics connector.
- Automated PCB component and layout generation with Python; verified channel connectivity and took the board through fabricator DFM/DFA review to production (16 units ordered).

Projects

Laika Quadruped Robot: Motor-Control Firmware
CMU Robotics Club | Firmware Team Member | Oct 2026 – Present | GitHub: CMU-Robotics-Club/quads_firmware

- Brought up four CAN buses to ODrive motor controllers on an STM32H723 (three on-chip FDCAN plus an SPI-attached MCP2518FD), configuring the clock tree and 1 Mbps bit timing to match the ODrive's 87.5% sample point.
- Integrated a 1 kHz PD joint torque controller into the FreeRTOS control loop with bumpless start, an ODrive watchdog feed and a zero-torque dry-run mode, then validated it on a two-joint test leg.

Biosignal-Driven Shared-Autonomy Assistive Manipulation
Independent Project | Jun 2026 – Present | GitHub: jarrod227/emg-shared-autonomy-arm

- Built a three-layer shared-autonomy system in ROS 2 (Jazzy) and MoveIt 2, combining three simple user commands with autonomous perception and motion planning.
- Ran a fixed-point LDA intent classifier on an STM32F103 (three surface EMG channels, 20 Hz), reaching 93.3–95.1% offline accuracy on held-out electrode applications.
- Added event gating and markerless YOLO-based perception, recording zero discrete-intent false triggers over 10 minutes of normal activity; ArUco baseline localization within ~1 cm at ≤0.5 m.

Head- and Eye-Controlled Cursor via Electrooculography (EOG) and IMU
Rutgers University | Senior Capstone, Mentor: Simiao Niu | Sep 2025 – May 2026 | GitHub: jarrod227/capstone_project

- Owned firmware and signal processing for a 5-person hands-free cursor: head motion drives pointer movement, eye movements handle clicks and scrolling.
- Designed a dual-channel AD8232 analog front end for EOG and applied Kalman filtering for gyroscope drift correction, bounding pointer drift over 10+ minutes.
- Trained an RBF SVM to classify six eye states, reaching 87.0% cross-validation accuracy and ~77% end-to-end task success across 20 trials per action.

Brain-to-Text Speech Decoding for Brain-Computer Interfaces
Rutgers University | Deep Learning Course Project | Apr 2026 – May 2026 | GitHub: jarrod227/brain2text-skipdiphone

- Built a full PyTorch pipeline for intracortical speech decoding: bidirectional GRU encoder, diphone CTC decoder (DCoND) with skip-diphone and smoothness losses, and Kaldi WFST decoding.
- Ran a five-variant ablation study on the Brain-to-Text '24 benchmark, improving phoneme error rate to 19.75% vs. 20.48% for the baseline.
- Found the phoneme-level gain did not carry over: the baseline retained the lowest word error rate (18.07%) with a 5-gram language model after WFST decoding.