Aditya Patil

Brooklyn, NY | aditya.a.patil2@gmail.com | Portfolio | GitHub | LinkedIn

Robotics M.S. student with hands-on embedded systems experience across MCU bring-up, sensor/actuator integration, and communication protocols.

Education

New York University, Tandon School of Engineering — Brooklyn, NY
M.S. in Robotics, Mechatronics Track — Sept 2025 – May 2027 (Expected)

- Relevant Coursework: Robot Localization & Navigation, Advanced Mechatronics, Control Systems, Machine Learning

Symbiosis Institute of Technology — Pune, India
B.Tech in Mechanical Engineering, Minor: IoT & Smart Manufacturing — Jul 2021 – Jun 2025

Selected Projects

STM32 MCU Bring-Up: Motor, Steering, and Sensor Control System
STM32 (C, ARM Cortex-M), Raspberry Pi (Linux) – Ackermann-steered mobile robot

- Brought up an undocumented STM32 (ARM) board from scratch – reverse-engineered motor, steering, and electrical-polarity mappings with no board-level documentation, consulting the STM32 datasheet and reference manual to configure timer peripherals (alternate functions, prescalers, PWM channels)
- Wrote safety-critical C drivers for motor/servo PWM and encoder reading, gated behind an arm-before-move sequence; calibrated motion primitives to measured tolerances ($\sim$15.24 cm straight-line increments, $\sim$59.73° turns, near-zero drift)
- Diagnosed and corrected a timer-prescaler bug (from 83 to 15) blocking servo response, then isolated an intermittent serial/USB CDC communication fault from the mechanical layer using reproducible evidence of corrupted command strings
- Built a Linux-based (Raspberry Pi) control and route-execution stack that safely armed, executed, and logged multi-segment routes, completing 3 consecutive full runs in a 5-run reliability stress test
- Managed the firmware toolchain (STM32CubeCLT, CMake/Ninja, SWD flashing) across iterative versions – removing an overly restrictive arming step and replacing pulse-burst motion with smoother continuous-hold commands – while preserving safety behavior

Stereo Visual-Inertial SLAM with Optimization and Loop Closure
Python, OpenCV, EuRoC MAV

- Built a stereo visual-inertial SLAM pipeline – stereo depth estimation, LK feature tracking, IMU fusion, bundle adjustment, loop closure – reducing absolute trajectory error from 0.192 m to 0.031 m
- Evaluated trajectory accuracy via alignment and multi-sequence ATE benchmarking, generating sparse 3D landmark maps

Closed-Loop Temperature Control System
Arduino, discrete-time control

- Contributed to designing and implementing a discrete-time feedback controller (Tustin discretization) for closed-loop temperature control on Arduino-based embedded hardware

Experience

Student Researcher — May 2026 – Present
AI4CE Lab, NYU Center for Robotics and Embodied Intelligence (CREO)

- Supporting the platform's data-collection infrastructure – multi-camera teleoperation dataset pipelines and hardware and firmware diagnostics on Linux – to keep experiments running
- Patched the platform's arm-control library's PD + feedforward-torque command path – correcting torque encoding, mode-activation codes, and CAN frame structure – restoring safe joint actuation on updated firmware, now used in ongoing manipulation research
- Co-authoring an in-preparation manuscript on vision-language-action (VLA) and vision-language (VLM) models, evaluated on YOR, a dual-arm mobile manipulator with onboard embedded GPU compute

Technical Skills

Programming: C (embedded firmware), C++, Python, MATLAB
Embedded & Protocols: STM32 (ARM Cortex-M), Arduino, UART/Serial, CAN, USB CDC, PWM/Timer Configuration, GPIO
Embedded Toolchains: STM32CubeCLT, CMake/Ninja, SWD/ST-Link Flashing
Systems & Tools: Linux (Raspberry Pi, embedded debugging), ROS2, OpenCV, Git
Robotics & Perception: SLAM, Visual Odometry, Stereo Vision
Controls & Estimation: Discrete-Time Control Systems, Sensor Fusion, State Estimation