# SHARWIN PATIL

sharwinpatil@u.northwestern.edu  925-389-8466  sharwinpatil.info  Chicago, IL

## EDUCATION

**Northwestern University | Master of Science in Robotics**

*Expected Graduation: 12/2025*  *Chicago, IL*

**Northeastern University | B.S. in Computer Engineering & Computer Science, Minor in Robotics**

*Graduated: 05/2024*  *Boston, MA*

## SKILLS

**Robotics:** ROS/ROS2, Nav2, MoveIt, RViz2, Gazebo, OpenCV, Robot Kinematics, Dynamics, & Manipulation, Kalman Filtering
**Software:** Embedded C, C++, Python, C#, .NET, Java, Matlab, Linux, Git, Unity, Docker, SolidWorks (CSWA)
**Machine Learning:** Computer Vision, PyTorch, CNNs, Reinforcement Learning (DQN, PPO), Mujoco

## EXPERIENCE

**Fulfil Solutions | Robotics Software Engineer**

*10/2025 – Present*  *Mountain View, CA*

- Writing C# behavioral code for automated fulfilment systems focusing on grocery packing robots

**Locus Robotics | Planning & Controls Intern**

*06/2025 – 09/2025*  *Remote*

- Optimizing fleet behavior with new ROS services bridging navigation stack (C++) and the robot behavior stack (Python)
- Performance-tested ROS services on large-scale maps with hundreds of data points, ensuring scalability for fleets of hundreds of robots
- Democratized access to navigation behaviors by integrating new features into the existing ROS-based architecture

**GreenSight | UAV Robotics Engineer Co–op**

*06/2023 – 12/2023*  *Boston, MA*

- Developed RTOS firmware for communications between a swarm of drones and GCS over LoRa for collecting real-time weather data
- Implemented a Hardware-Abstraction-Layer (HAL) in C for the ESP32 platform to interface with a custom LoRa chipset
- Created a custom SPI driver for the ESP32 to connect to peripherals for the drone's data collection

**Fulfil Solutions | Robotics Software Controls Co–op**

*07/2022 – 12/2022*  *Redwood City, CA*

- Developed planning code in C# for high-level behavior and task assignment for up to 50 robots in a warehouse automation setting
- Composed data fetching functions to bridge C# sequencing code to MongoDB
- Optimized AGV planning and curated heuristics for maintaining the factory's health while improving performance
- Deployed factory-wide alerts and notifications for operators to react with relevant safety measures

**Doble Engineering | Software Engineering Co–op**

*07/2021 – 12/2021*  *Marlborough, MA*

- Developed an external data persistence mechanism in C# running on the .NET framework for various Doble software products
- Designed and deployed a firmware installation wizard using Windows Presentation Foundation (WPF) for Doble instruments

## PROJECTS

**Open Sourced ROS2 Delta Robot**

*01/2025 – 03/2025*

- Designed and fabricated a 3-DOF delta robot using 3D printed parts and Dynamixel motors
- Derived Jacobian for planning and executing both position and velocity trajectories
- Authored a configurable ROS 2 package to interface with any delta robot and provide motion planning and sensor telemetry

**Toasting Bread with Franka Robot Arm**

*11/2024 – 12/2024*

- Implemented a ROS2 package to interface with the MoveIt API for sending motion requests to the Franka robot arm
- Utilized an Intel Realsense camera to identify april tag markers for the robot to interact with the scene
- Collaborated within a group of students to sequence the camera and robot with the scene elements to autonomously toast bread

## KUKA Mobile Manipulator Pick and Place Simulation

*11/2024 – 12/2024*

- Implemented a task-space feed-forward controller for the KUKA YouBot in Python for pick and place tasks
- Derived Jacobian for omni-directional mobile base and robotic arm to generate velocity trajectories for robot arm and the mobile base

## Extended Kalman Filter for 6-Wheel Mars Rover

*07/2025 – Present*

- Derived and implemented an Extended Kalman filter in C++ for state estimation, fusing data from wheel encoders, an IMU, and a LiDaR
- Simulated the rover's kinematics in Python to tune process and measurement noise covariances and visualize planned trajectories
- Architected a ROS2 Package to interface with sensors and handle real-time data processing