

# ADITYA PATIL

📞 9011036943 ✉ [patiladitya1309@gmail.com](mailto:patiladitya1309@gmail.com) 🔗 [www.linkedin.com/in/aditya-patil-13207b201/](https://www.linkedin.com/in/aditya-patil-13207b201/) 🛠 [maker-ATOM](#)

## ROBOTICS ENGINEER

Robotics Engineer specialized in designing control software for **Autonomous Mobile Robots** and **Manipulators** utilizing ROS / ROS 2 framework. My expertise lies in Sensor-Integration and Perception for Autonomous Robot and Motion Control for Manipulators.

## EDUCATION

**Pune Institute of Computer Technology, Pune**

**Aug 2020 – July 2024**

*Bachelor of Engineering in Electronics and Telecommunication*

*GPA: 9.04/10*

## TECHNICAL SKILLS

**Languages** : Python, MATLAB

**Technologies / Framework** : ROS 2, Sensor integration, Swarm Intelligence, Gazebo, Rviz

**Computing Environments** : Linux, Raspberry Pi

**Design & Manufacturing Tools** : KiCAD, Fusion 360

## EXPERIENCE

### PICT Robotics

**Apr 2021 – Aug 2023**

*Lead Robotics System Designer*

*Pune, Maharashtra, India*

- Led a team of 4 to design and develop modular code-base to semi-automate ABU Robocon 2023 Robots using **ROS**.
- Introduction of **custom ROS message** to tackle unacknowledged methodology in transmitting motor parameters in ROS.
- Insertion of sync packets in serial communication between ESP32 and Raspberry Pi for synchronization.
- Utilized **sensor integration** to enable precise robot **localization**, integrating data from an Inertial Measurement Unit and Optical Mouse Sensor for accurate odometry.
- Resolved ESP32 pin count limitations by crafting a **Motor Controller HAT** using Atmega328p micro-controller, communicating via I2C protocol for each Swerve pod.
- Implemented rotary encoder-based PID control on **STM32** micro-controller to evaluate both three-wheel holonomic and four-wheel mecanum drives.

### Kanan Park

**Apr 2022 – Jun 2022**

*Electronics Engineer (Intern)*

*Pune, Maharashtra, India*

- Designed control software, utilizing **Teensy** micro-controller as the master to interface with the RC transmitter, and employing Arduino Nano as a slave to manage PID systems for individual swerve pods.
- Established communication protocols between Arduino Nano and Teensy using MAX485 (TTL to RS485), implementing an acknowledge symbol for synchronized transmission.

## PROJECTS

**Decentralized Mapping and Navigation using Swarm of Robots** | ROS 2, Multi-Robot Control

**Aug 2023 – Present**

- Integration of **Mapping and Navigation** Capabilities into renowned swarm algorithm - Swarm Gradient Bug Algorithm (SGBA).
- Real-time Cartographic Data Generation and Propagation by the swarms through a localized mesh network.
- **Adaptive Task Assignment** via a Health State Dependent mechanism for the Robotic entities.

**Optical Odometry** | ROS 2, USB-Driver, Python | [GitHub](#)

**Oct 2023 – Present**

- Experimental project to generate **odometry** data for Autonomous Mobile Robots using through the utilization of Optical Flow Sensor.

**Hologlyph Bots** | ROS 2, Computer Vision, Inverse Kinematics, Python | *Active Development*

**Sept 2023 – Present**

- Implementation of a Proportional Control Driver for Autonomous Mobile Robots employing a **Finite State Machine**.
- Robot Localization through the utilization of **Aruco Markers**, along with the implementation of Fail-Safe Mechanisms for cases in which Not All Markers are detectable.

## CERTIFICATIONS

**Modern Robotics: Foundations of Robot Motion** | [Link](#)

**Aug 2023**

**Modern Robotics: Robot Kinematics** | [Link](#)

**Sept 2023**

**Python Classes and Inheritance** | [Link](#)

**Sept 2023**

**Trees and Graphs** | [Link](#)

**Oct 2023**

## OPEN SOURCE

**Cytron Motor Driver** | [Link](#)

**May 2023**

- A custom library to interface with Cytron Motor Drives which utilizes PWM-DIR and PWM-PWM technique to drive DC motors.
- Optimized codebase for Espressif boards, notably ESP32, addressing **multiplexed PWM channels**.
- Introduced an additional parameter to fulfill ESP32's channel demands, while ensuring seamless **backward compatibility** by setting the parameter to its default state.