

# Kasina Jyothi Swaroop Chicago, IL | (773)-250-4095

Portfolio : [kjyothiswaroop.github.io](https://github.com/kjyothiswaroop) | Email: [jyothikasina2026@u.northwestern.edu](mailto:jyothikasina2026@u.northwestern.edu) | LinkedIn: [linkedin.com/in/kjswaroop-robotics](https://www.linkedin.com/in/kjswaroop-robotics)

## EDUCATION

---

### Northwestern University

*M.S in Robotics*

Evanston, IL  
Sept 2025 – Dec 2026(Expected)

### Indian Institute of Technology (ISM) Dhanbad

*B.Tech in Electrical Engineering*

Dhanbad, India  
July 2018 – May 2022

## SKILLS

---

**Programming:** C, C++, Python, Embedded Programming, Linux, Git, Unit testing, Behavior Trees, Docker

**Robotics:** ROS 2, SLAM, Perception, Path Planning, Multi-Agent systems, Motion Planning, Control theory, Manipulation

**Machine Learning:** Supervised, Unsupervised, MLP, CNN, YOLO, Reinforcement Learning

**Libraries:** OpenCV, MoveIt 2, Nav2, RTABMap, ORBSLAM3, g2o, realsense2, zed-sdk, Tensorflow, Scikit-learn, Pandas

**Hardware:** ESP32, Arduino Uno, RealSense camera, ZED2i camera, Roboteq controller

**Software Tools:** Gazebo, Mujoco, CoppeliaSim, Rviz, MATLAB, SQL, MongoDB, RabbitMQ, REST API

## EXPERIENCE

---

### Telexistence

*Robotics Engineer*

Tokyo, Japan  
October 2024 – July 2025

- Engineered a Fleet Management System (FMS) in C++ for Autonomous Forklifts (AGF) handling depalletization tasks.
- Implemented task-planning using behavior trees and inter-process communication between AGFs and robot arms in ROS 1.
- Designed recovery pipelines that reduced AGF errors by 93%, cutting failures from 15 per day to just 1.
- Integrated external AGFs into the FMS by developing REST API interfaces in C++ within 2 weeks, accelerating validation.

### Addverb Technologies

*Robotics Engineer*

Noida, India  
August 2022 – June 2024

- Implemented multi-agent path planning algorithms in C++, using behavior trees for task and charge management.
- Independently developed FMS for two greenfield systems: a 28-robot tugger fleet and a vertical sortation robot demo.
- Dockerized code to easily deploy and manage 150+ robots across 4 countries, achieving more than 95% throughput.
- Constructed dashboards for customers and maintenance teams, automating error & throughput reports via Webhooks.
- Mentored interns and coordinated across product, hardware, and mechanical teams to deliver products and deployments.

## PROJECTS

---

### Autonomous Humanoid Exploration [C++, Python, ROS 2, CycloneDDS, RTABMap, Nav2]

Jan 2026 – Present

- Built a reliable bridge from low-level CycloneDDS topics on the Unitree G1 to FastDDS topics in ROS 2.
- Developed a locomotion interface using the G1 high-level control API for teleoperation of the humanoid.
- Implemented RTABMap SLAM on the humanoid for real-time mapping in an indoor lab environment.
- Integrated the Nav2 stack for autonomous navigation of the robot.

### Catch the ball - Franka Robot Arm [Python, ROS 2, MoveIt 2, OpenCV, Machine Learning]

Dec 2025

- Worked in a team of four to design and implement a ball-catching robotic system using the Franka Emika Panda robot arm.
- Built a custom MoveIt 2 API wrapper for joint-space and Cartesian planning with gripper control.
- Developed real-time ball detection and tracking pipelines using OpenCV and YOLO in Python.
- Integrated the `easy_handeye2` package for extrinsic calibration of the realsense/zed camera.
- Coded a custom ROS 2 Kilted wrapper for interfacing with the ZED2i camera.
- Implemented modular ROS 2 nodes for perception, trajectory prediction, and control.

### Unscented Kalman Filter [Python, Git]

Oct 2025

- Implemented the Unscented Kalman Filter from scratch in Python using NumPy on the MRCLAM dataset.
- Fused control inputs and measurements using UKF to correct dead-reckoned trajectories towards the ground truth.
- Resolved asynchronous data rates by building a time-ordered event list and applied correction at measurement timestamps.

### Robotic Pen Grabber - PincherX Robot Arm [Python, ROS 2, OpenCV, Rviz, Git]

Sep 2025

- Developed a vision and manipulation pipeline in Python for a robot arm to detect and pick a pen using a RealSense camera.
- Implemented color-based detection in OpenCV (HSV filtering, contour extraction) and computed centroids to locate the pen.
- Calibrated the camera-robot transform for accurate pick-and-place execution.

### Saviour - Epileptic Seizure Detection Device [Python, ESP32, TensorFlow, Arduino IDE]

Mar 2020

- Prototyped an ESP32 + EMG sensors on a glove to detect seizures and alert caretakers via an app with a team of 4.
- Achieved 89% accuracy on a custom EMG signals dataset.