

Autonomous Roaming Robot using LiDAR (ARRL)

EE 416/464 Computer Engineering Senior Lab, Fall 2020

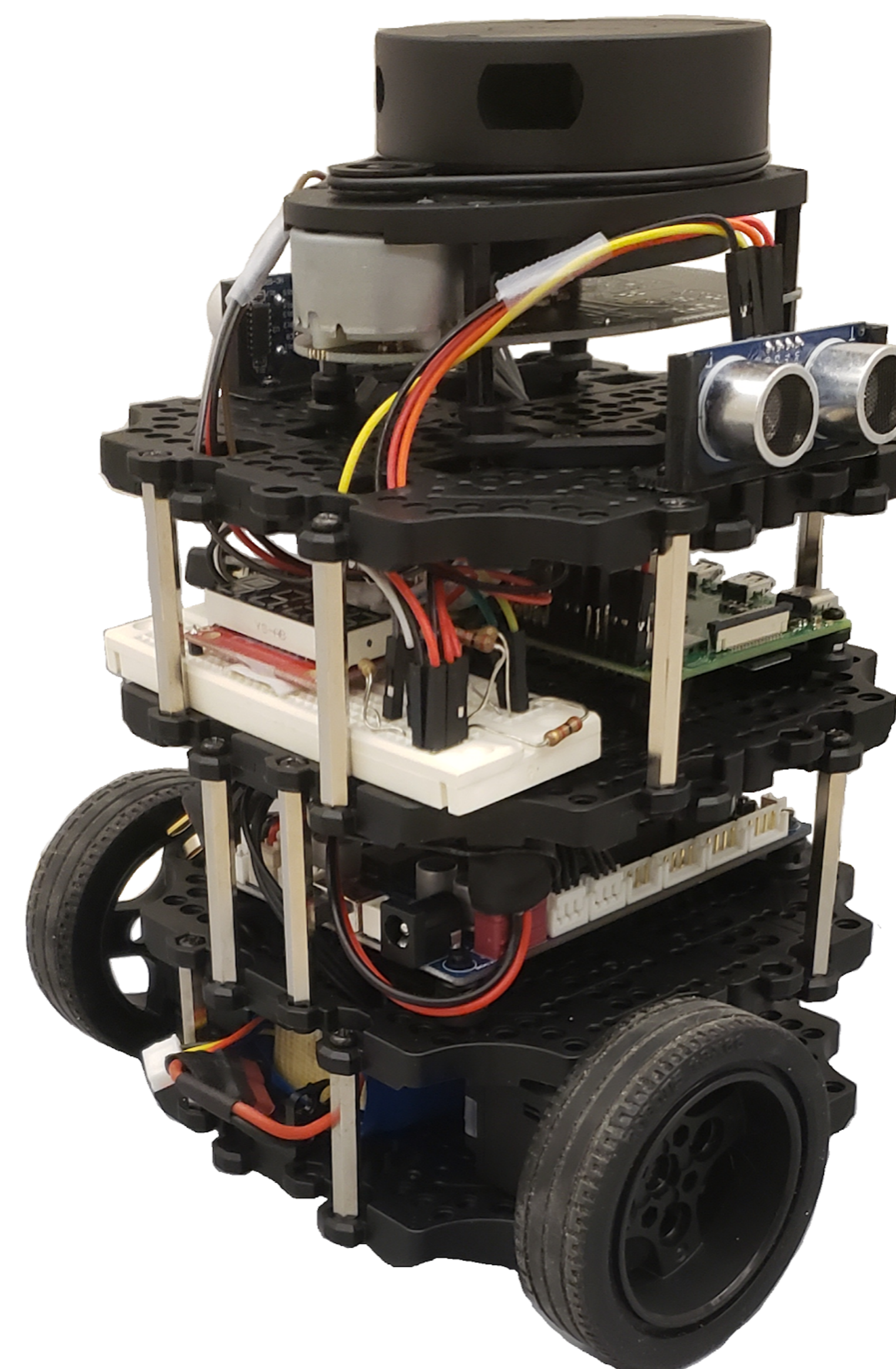
Team A: Joshua Frank, Evan Perry, Evan Nguyen, Rod Izadi. Mentored by Dr. Chen Liu and Zhiliu Yang.

Sponsored by Clarkson University's Computer Architecture and Microprocessor Engineering Lab.



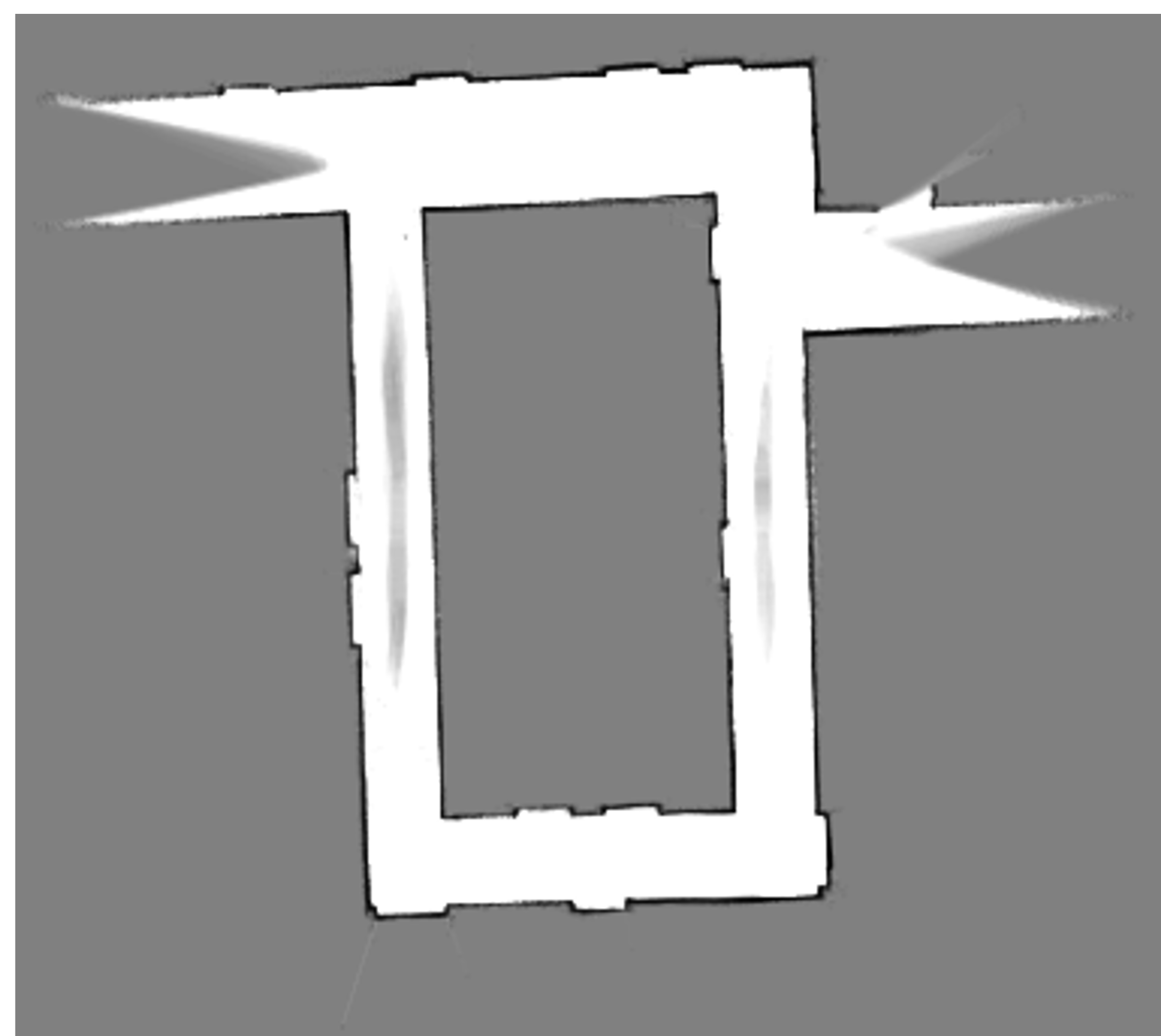
Abstract

The primary goal of this project was to create a system that was able to collect data to generate a map, and then be able to autonomously navigate this generated map. The system we have created is able to record LiDAR and Ultrasonic sensor data while manually piloted, as well as autonomously navigate a given map. Alongside this the system has active object avoidance using the LiDAR sensor, allowing the system to detect objects not portrayed on the map image.



Autonomous Navigation

Using the ROS navigation package, the robot can be assigned 2D navigation goals on a map and calculate a path to the destination. This path is constantly re-assessed and react to unexpected objects. Using the LiDAR sensor, the robot will react to any unmapped objects that enter the path. The active object avoidance feature is essential for safe autonomous navigation.



Generated 2D Map of 2nd Floor CAMP Annex

Map Generation

Using the recorded data within the .bag file, a map is generated using the pi-Map algorithm or Google's Cartographer. These maps represent the 2D environment with a maximum error of 1.3%.

Hardware Components

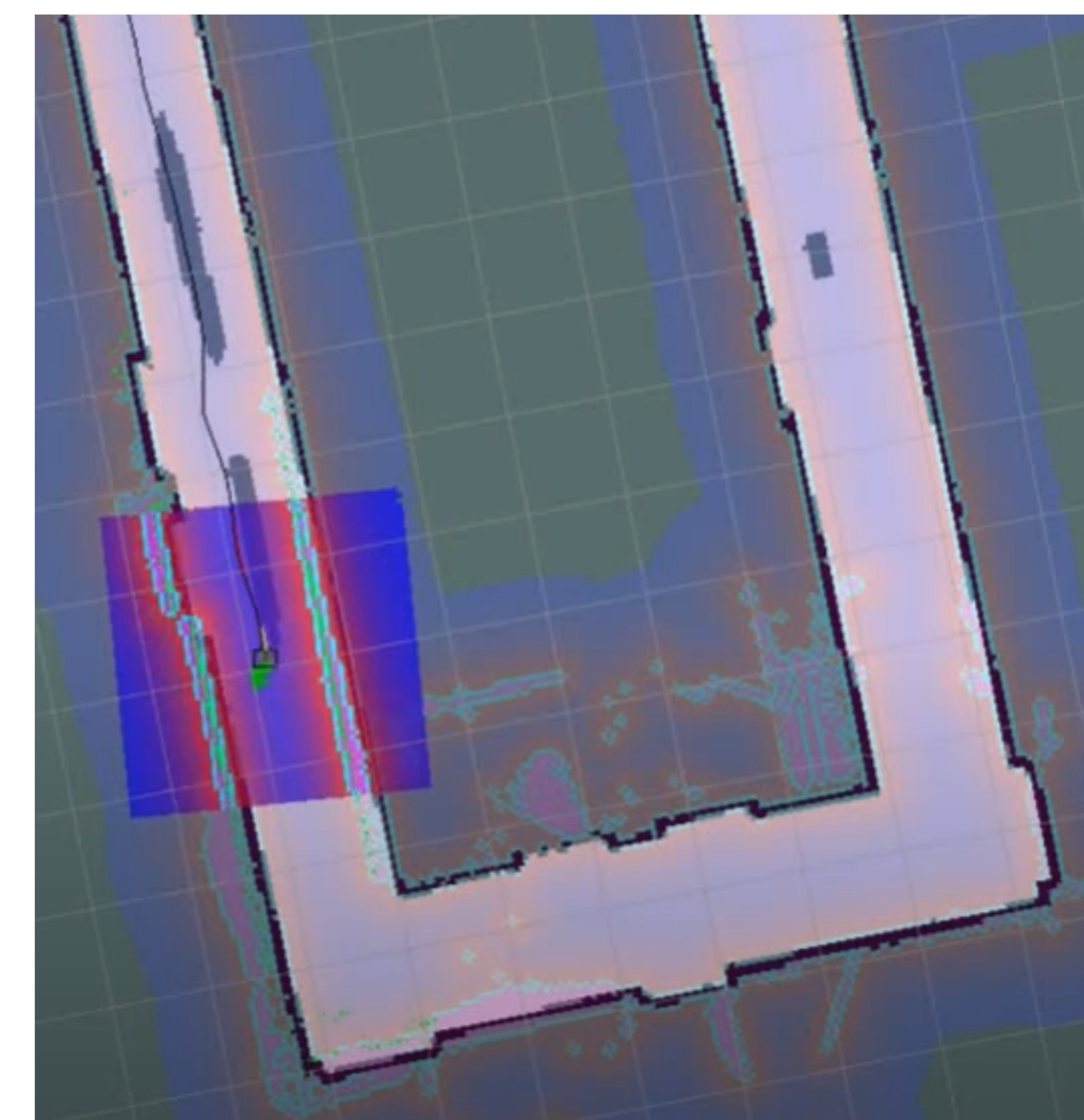
- Turtlebot3 Platform
- Raspberry Pi 4 (8GB)
- OpenCR Board
- RPLiDAR-A1 Sensor
- 2x HC-SR04 Ultrasonic Sensors

Software Components

- Ubuntu 18.04
- ROS Melodic
- VNC Viewer
- Cartographer
- pi-Map

Data Collection

Movement and sensor data is recorded into a ROS .bag file. The data points from both the LiDAR and Ultrasonic sensors are collected at a rate of 10 Hz. While the data is recorded, an operator is manually controlling the robot using the teleoperation ROS node.



Autonomous Navigation and Localization. Path prediction is determined by black line. Localization is depicted by green point cloud around robot.