Command Control Dashboard

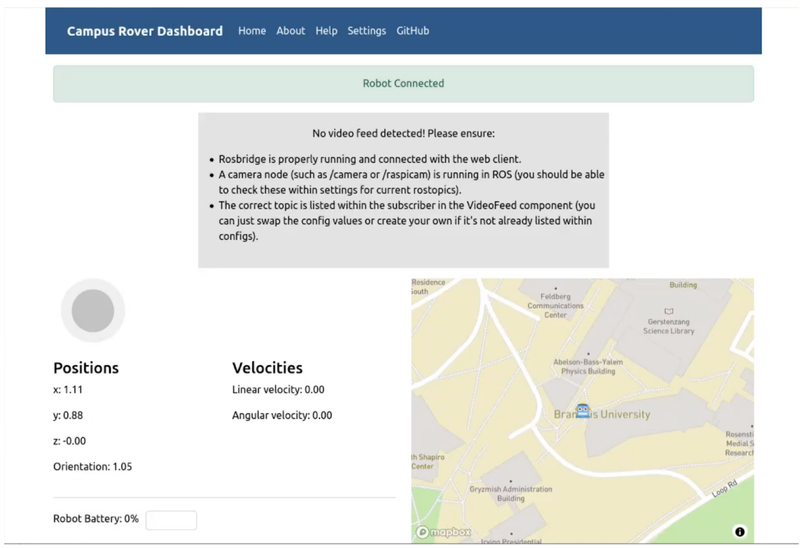
Naimul Hasan, James Kong, Brandon J. Lacy
Brandeis University
COSI-119: Autonomous Robotics

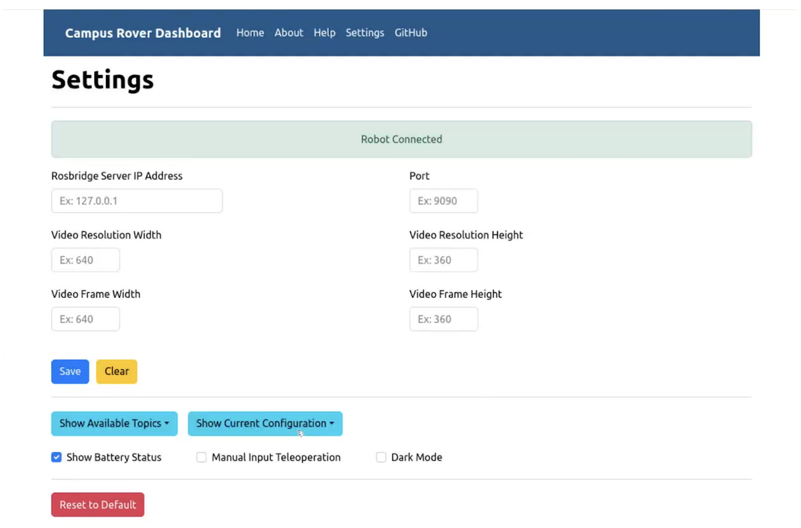
Overview

The command and control dashboard, otherwise known as the campus rover dashboard, is a critical component to the development of the campus rover project at Brandeis University as it's the medium in which remote control will take place between the operator and the robot. It is the culmination of three components: a web client, robot, and GPS. Each component is a separate entity within the system which leverages the inter-process communications model of ROS to effectively transmit data in messages through various nodes. There are various directions in which the campus rover project can progress, whether it be a tour guide for prospective students or a package delivery service amongst faculty. The intention of our team was to allow the command and control dashboard to serve as the foundation for future development regardless of the direction enacted upon in the future.

Web-Client

The front-end and back-end of our web client is built with React.js. It uses ROSLIBJS to establish a WebSocket connection to the ROSBridge server, through specifying a specific IP address and port number. ROSBridge is a package for the Robot Operating System (ROS) that provides a JSON-based interface for interacting with ROS through WebSocket protocol (usually through TCP). It allows external applications to communicate with ROS over the web without using the native ROS communication protocol, making it easier to create web-based interfaces for ROS-based robots. ROSLIBJS is a JavaScript library that enables web applications to communicate with ROSBridge, providing a simple API for interacting with ROS. It allows developers to write web applications that can send and receive messages, subscribe to topics, and call ROS services over websockets. ROSBridge acts as a bridge between the web application and the ROS system. It listens for incoming WebSocket connections and translates JSON messages to ROS messages, and the other way around. ROSLIBJS, on the other hand, provides an API for web applications to interact with ROSBridge, making it easy to send and receive ROS messages, subscribe to topics, and call services.





GPS

GPS, or Global Positioning System, is a widely-used technology that enables precise location tracking and navigation anywhere in the world through trilateration, a process which determines the receiver's position by measuring the distance between it and several satellites. Though it has revolutionized the way we navigate and track objects, its accuracy can vary depending on a variety of factors that are beyond the scope of this paper. However, there are various companies which have embraced the challenge of a more accurate navigation system that incorporates other data points into the algorithm responsible for the determination of the receiver position. Apple is notorious for the pin-point accuracy available within their devices which incorporate other sources such as nearby Wi-Fi networks and cell towers to improve its accuracy and speed up location fixes. The utilization of a more sophisticated location technology is critical for this project to be able to navigate routes within our university campus. Therefore, we've chosen to leverage an iPhone 11 Pro placed on our robot with the iOS application GPS2IP open to leverage the technology available in Apple's devices to track our robot's movement.

Result

We sought a deeper understanding of the inter-process communications model of ROS to be able to allow our web application to take advantage of this model and we've successfully developed this knowledge. Additionally, it allowed us to integrate our software development knowledge with ROS to create a complex system which will draw attention to our respective resumes. We are proud of the product created in this course and see it as a major improvement from the previous implementation of the dashboard. The only aspect in which we are not content is the lag present on the camera feed and joystick. However, it is an area for improvement in which we recommend to be addressed by the next team to tackle this project. There are a variety of improvements to be made to this application to continue to shape campus rover into an implemented product. It's up to the next generation of students to decide where to take it next.