# Delivery Robot (Tennis Prototype)


**Team Members** Jayson Castro, William Corella, Daniel Del Rio, John El Hosni, David Pacheco, Irene Zhang

**Faculty Advisor** Mo Zhang
**Technical Advisor** Sergio Marquez

**Texas Instruments Liaison** Mark Easley

**Department of Electrical Engineering, Department of Mechanical Engineering**
**College of Engineering, Computer Science, and Technology**
**California State University, Los Angeles**


## Project Background

Autonomous robot delivery is an advancement in technology, a given purpose for our team to assemble for our robot's prototyping phase. Our project consists of using the Robotic Operating System (ROS), BeagleBone Blue, and Raspberry Pi. Our system design will contain multiple IDE software and CAD design for our team to build an autonomous tennis ball retriever robot.

## Objectives

- Design and build an autonomous tennis ball retriever.
- Robot will detect and locate tennis balls.
- Robot will autonomously maneuver around obstacles while in route to collect and store tennis balls.

## Requirements

- Must have capability to be autonomous, with an additional manual control mode
- Autonomously navigate to a predetermined destination
- The robot must avoid obstacles in a stationary environment autonomously

## Navigation Stack Setup

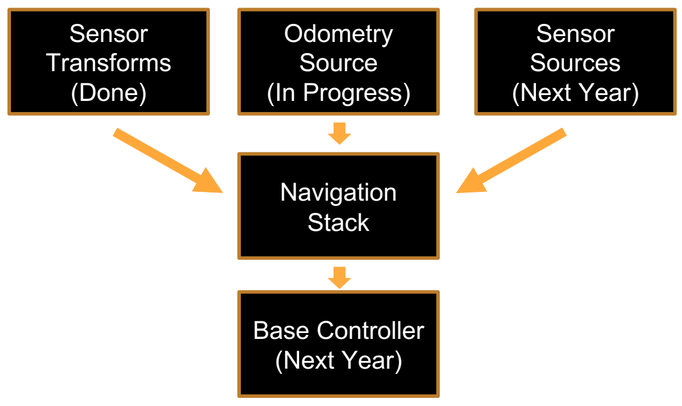


**Figure 1**. Platform Specific Nodes for The Navigation Stack

## System Design Overview

### Chassis Design

- Design and build an autonomous tennis ball retriever.
- Prototype will be constructed majority of 3D printed ABS plastic.
- Retrieve and store up to three tennis balls.
- Design will be constructed majority of 3D printed ABS plastic.

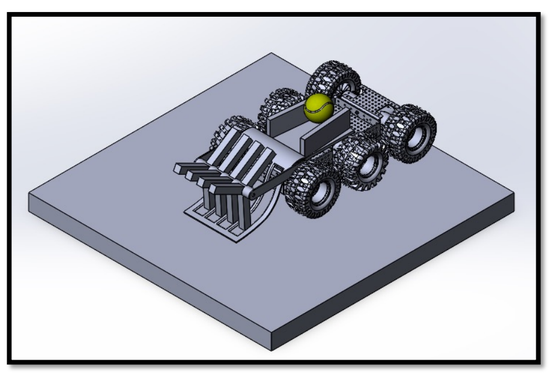

**Figure 2.** Tennis Ball Robot

### Component Design

- Raspberry Pi 4 is a microprocessor capable of running ROS packages.
- Using publisher and subscriber nodes, important data can be sent to different components of a robot.
- With sensors as a publisher, and motors as a subscriber, motor output can automatically change depending on sensor input.
- The concept is pushed further when there are multiple sensors and motors working together.

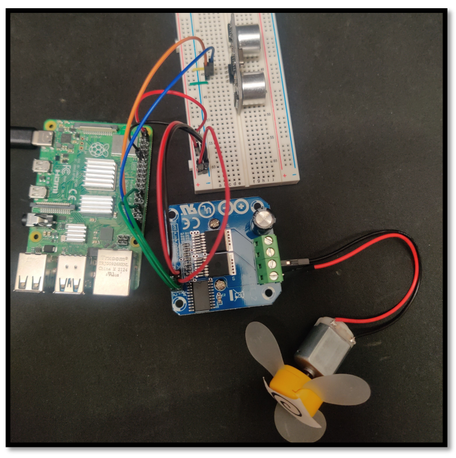

**Figure 3**. Sensor/Motor Control

### Sensor Integration Codes

- Four ultrasonic sensors (HC-SR04) are in the depiction below will measure distance of the object in centimeters Providing 400 cm as its maximum range for ranging accuracy.
- Arduino Sketch and Processing Software (IDE) are the communication coding programs for RADAR usage of the ultrasonic sensors

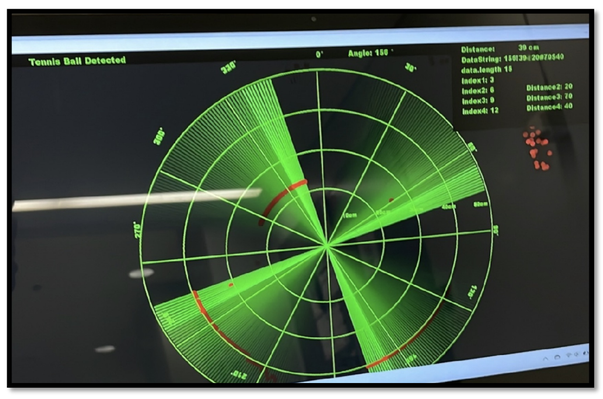


**Figure 4**. Sensor Depiction

### Object Detection System

- All programs have been created using Python with OpenCV on the Visual Studio Code.

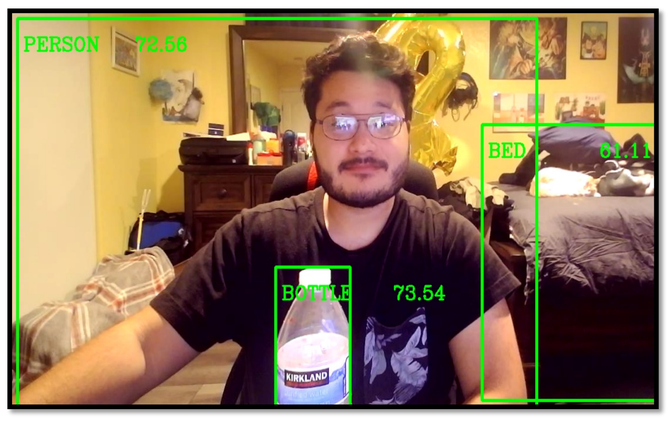


**Figure 5**. General Object Detection (Item Name and Confidence Level Percentage)


**Figure 6**. Tennis Ball Detection (BLUE: Confidence Percentage) / Distance Readings (RED: Inches)

### Motors

- Controlled by a PWM signal
- Depending on sensor values and location, direction of rotation will be determined
- Able to traverse quickly to destination

## Conclusion

Although we were unable to complete the full integration of each of the sections of the robot, the advancements made in most of the sections were of great significance for the completion of a delivery robot. Whether it be the creation of the object detection code, the ROS navigation stack setup, or the robot CAD/prototype, these will be the foundation for the future teams.