

To: Professor Pisano

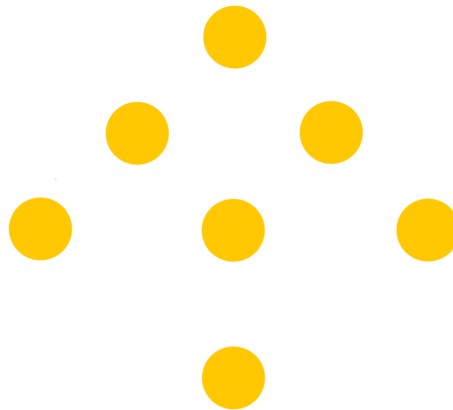
Team: 10

Date: 11/3/16

Subject: First Prototype Testing Plan

First Prototype Test Plan

Draper Convoy



By

Team 10

Team Members

Sadman Kabir: kabirs@bu.edu

Benjamin Hsu: bhsu25@bu.edu

Nikhil Krishna: nikhilk@bu.edu

Hugo Silmberg: hsilmber@bu.edu

Michael Moran: mmoran3@bu.edu

Required Materials

Hardware:

- Desktop computer running Ubuntu and access to Gazebo simulation software

Software:

- Gazebo software
 - Simulates TurtleBot sensor information and output
- ROS2 installed on the desktop computer
 - Using ROS2 Humble distribution
 - Allows data to be accessed through TurtleBot sensors and communicate with other software
- Person detection node
 - YOLO object detection algorithm
 - Pretrained model from ultralytics
 - Identifies human beings detected within camera view and outputs information about their position relative to the robot
 - Highly accurate, even from long distances and figure partially cut off
- Motor control node
 - Accepts data from the person detection node and outputs information to control the TurtleBot's motors

Set-Up

1. Open a terminal window
2. Enter this in to source it correctly:
 - `cd ~/ros2_ws`
 - `source /opt/ros/humble/setup.bash`
 - `source install/setup.bash`
3. To launch the simulation enter this:
 - `ros2 launch turtlebot4_ignition_bringup turtlebot4_ignition.launch.py`
 - Want to launch the simulation with nav2, slam and, rviz? Enter this:
 - `ros2 launch turtlebot4_ignition_bringup turtlebot4_ignition.launch.py`
`nav2:=true slam:=true rviz:=true`
 - Want to launch the simulation with a preexisting map for nav2:
 - `ros2 launch turtlebot4_ignition_bringup turtlebot4_ignition.launch.py`
`nav2:=true slam:=true rviz:=true`
4. Open another terminal window and source it like in **step 2**
 - Enter this to launch the motor control node:
 - `ros2 run my_teleop_pkg teleop_twist_keyboard`
5. Open another terminal window and source it like in **step 2**
 - Enter this to launch the perception node for person detection:
 - `ros2 run yolo_person_detector yolo_person_detector`
6. When you get everything launched, make sure to unpause the simulation and then click on the terminal with the motor control to put it in focus so you can use the keyboard to control it

(Node files should be located in ~/ros2_ws/src.)

Pre-Testing Set-Up Procedure

1. Ensure Gazebo simulation is running correctly.
2. Check that all nodes are running and responding.
3. Ensure control functionality is working correctly.

Testing Procedure

- Place a human model in the simulation and turn the TurtleBot to identify the human.
- The person detection node identifies the human and sends data to the motor control node.
- The TurtleBot begins to move toward the human model.
- Move the human model around the simulation to demonstrate real-time identification and data processing.
- Move the human model outside of the camera's frame and test relocation functionality.
- Controls:
 - P: Autonomous driving mode + relocation
 - U I O; J K L; M , . ; Remote control driving
 - anything else: stop

Measurable Criteria

- I. The person detection node should be able to correctly identify a human being in the Gazebo simulation, display a bounding box around the detected person, and place a centroid in the center of the bounding box.
- II. The person detection node should display the confidence level of the detection near the bounding box, and publish data about the person's distance and angle relative to the center of the camera's frame.
- III. The motor control node should successfully subscribe to the person detection node and accurately retrieve the data published from it. The motor control node should respond to this incoming data and output data to the TurtleBot's motors to move toward the detected human being, with the robot stopping when it gets within a defined distance.
- IV. If the person moves out of the camera's view, a button can be pressed and the TurtleBot should rotate in an attempt to relocate the human being.

Score Sheet

Functionality	Working? (Y/N)
Identifies a human model	
Bounding box around model w/ confidence level and centroid	
Seek human target	
Stop when close enough to the human model	
Auto relocate -> continue seeking humans	