Abstract

According to a National Health Interview Survey in 2018, 32.2 million American adults (10% of the entire US population) aged 18 and older reported experiencing vision loss. Infrastructures and assistance for visually impaired people are thus in great need. Among them, guide dogs are one of the most popular blind assistants due to their friendliness and long-term accompany. However, their issues like high training costs and entrance restrictions in certain transportations and public areas have urged the exploration of other guidance options. In our project, we designed a robotic guide dog and implemented some basic functionalities using a robotic car, a passive buzzer, and an ultrasonic sensor.

Part I: Project Description

The “guide dog” works as follows: it walks alongside the user. When it detects an obstacle ahead, it plays a note to notify the user to stop. Then, it looks to the left, if there is an obstacle, it plays a different note and looks to the right; if no obstacle on the left is detected, it moves toward that direction. If the robot looks to the right and detects an obstacle, it plays a note that is different from the previous two and turns around; if it does not see an obstacle on the right, it moves toward that direction. We believe that, with bugs fixed and user experience improved, our robotic guide dog would greatly benefit the visually impaired community by providing more people with guiding service, for our robot has little to no training cost and is able to be produced in a massive amount.

Part 2: Design and Implementation Details

The implementation of a robot guide dog has the following parts: obstacle avoidance, passive buzzer, and communication through ROS. First, we add multiple delays to the original design of obstacle avoidance to make both the detection and movement of the robot car slower. Then within each obstacle detection (front, left, right), we use a variable named ‘int_msg’ as a publisher. It only publishes code (0, 1, 2) to represent different scenarios. For code ‘0’, this means there is an obstacle ahead; for code ‘1’, there is an obstacle to its left; for code ‘2’, there is an obstacle to its right. While the car publishes the code, the other Arduino board will serve as a subscriber that reads the code. According to different codes, the buzzer will play different notes corresponding to different instructions to the user who are following the car. To achieve this, in the subscriber callback function, the program first reads the data inside ‘int_msg’ and has three ‘if/else if’ conditions to play corresponding notes to three different codes. Since we are still using two Arduino boards for our project, the communication between two boards can be handled by the launch file from lab 11.

Part 3: Relevance in the Course

For the robot car part, we used the ultrasonic sensor; servo; and chassisMove; for the buzzer part, we used the passive buzzer.

We mostly relied on the knowledge that we learned. Nonetheless, during the design of the project, unlike previous projects that solely focus on robot performance or outcome, we incorporated many human-centered ideas. For example, add significant delays between each movement, so that the blind person has time to react, judge, and move according to the car’s instructions.

Part 4: Review and Reflection

In this final project, we really solidified our understanding of ROS as we had to flip the roles of publisher and subscriber from previous labs. Additionally, we were able to take a piece of hardware, a buzzer, that we had only used once before and successfully integrate it into the project. In terms of soft skills, we also improved on our levels of communication, because for the final project we broke it up into different pieces and assigned individuals different parts. Previously for the labs, we had taken the approach of everyone working on the same thing off of one computer, but for the final project, there were many different moving pieces. So, being able to communicate and adapt was much more important.

To build this project, we relied on a lot of prior knowledge. A key piece of the setup was lab 11, which involved controlling the car using ROS and a joystick. Using lab 11 as a model allowed us to build out the framework for the final project. As for new information, we had to learn how to make the buzzer act as the subscriber and understand more about playing tones. Originally, we got the subscriber code to work but picked notes that were too similar sounding. This resulted in us believing that the subscriber code was incorrect, and that led us to try to troubleshoot for errors there instead.

Part 5: Limitations

One of the limitations we had was that we weren't able to set up the code for the publisher and subscriber properly. There were some errors when we were trying to create the subscriber in the very beginning, as it wasn't picking up any signals that were sent from the publisher - and later the right signal but too similar. However, this was anticipated. As we had similar problems when we were working on lab 11. We tried using the same techniques that Professor Biswas and Yifan

showed us, but that didn’t really work, so we had to call Yifan again to resolve the issue. Another issue we encountered was properly setting up the publisher so that it sends a different message depending on the obstacle. We had to change the code on the car a bit to fit the criteria - which took a bit longer than expected. Another limitation we had was with the buzzer. Since we used the buzzer in one of our labs, we were somewhat more comfortable using it. However, since we never used tones before, we had to learn a bit more about different tones and how to properly modify them. This resulted in one of our biggest setbacks, which as mentioned earlier, made us believe that the subscriber wasn’t working as intended.

Originally, our idea was to create a voice-controlled car. However, after a few days of trying to tinker with the new part we got, we realized that we are missing a lot of libraries and technicalities needed to make the project work. We believe that that would've been a very cool project to do in the future, just not with the equipment and knowledge we have on hand.