[1]

Autonomous Vehicle

Christian Rodriguez, Fengbo Yang, Yihua Zhou, Zhijun Chen, Advisor: Arnold Swindlehurst

***Abstract*—This project is about the Autonomous Vehicle, our goal is to build an autonomous car which it can avoid obstacles automatically, recognize and follow roads automatically, recognize traffic lights and signs automatically. In order to achieve the above requirements, we need the following materials: a modified RC car, battery, driver board, raspberry pi, lidar, and camera. Battery-powered, driver board can control moving and stop of car, raspberry pi used to control lidar and camera, lidar recognizes and avoids obstacles, and cameras are responsible for identifying traffic lights, signals and making vehicles follow the road. The reason for designing autonomous cars is because autonomous driving technology is the development direction of future driving modes. Mature applications of autonomous driving technology will bring more convenience to people's lives. At the end of the first quarter, we completed the movement and stop of the car as well as the lidar part as planned which could stop the car when it near obstacle and when we remove the obstacle, the car will re-run. In our plan, we will do the camera part in the next quarter, after that, the car will truly achieve autonomous driving on the road, and our projects are all completed.**

***Index Terms*—autonomous, vehicle, lidar**

I. Introduction

WITH the development of technology, the driving mode has gradually changed from human driving to autonomous driving, so autonomous driving will be the mainstream of future driving modes. Therefore, our project is about Autonomous Vehicle. To make an autonomous car requires a combination of hardware and software, for hardware, we need a modified RC car, battery, driver board, raspberry pi, lidar, and camera. For software, we use Python to write code for lidar and OpenCV for the camera. Now we have the power system of the car that can make car moving and stop, and we have done for lidar part which can stop the car when is near obstacle and the car will re-run when we remove the obstacle. In the next quarter, we will add a camera on the RC car and use the camera to identify traffic lights, signals and make vehicles follow the road.

II. Main

A. *Materials Used*

For the first quarter of the senior design of autonomous vehicle simulation, we mounted the A2M8 Lidar, showed in Figure 1.

There are many kinds of lidars. On this project, we need the lidar able to detect 360 degrees environment to ensure safety distance between the VR car and the objects around. We originally look for cheaper one but not spinnable, and place a spinning driver at the bottom, so the static lidar can be rotated, but it is hard to keep track of the distance data from lidar to the corresponding angle data from the spinning driver. We then looking for a spinnable, great-embedded lidar, and A2M8 lidar is a great fit. A2M8 Lidar is a small, self-error detect, with build-in function lidar [1].

Figure 1. A2M8 Lidar

We have two quarters for doing the autonomous vehicle simulation project. In our planning, we should finish the lidar with RC car in the 1st quarter (We reached the 1st quarter goal!) and should finish the camera, lidar with RC car in the 2nd quarter. Raspberry pi has a strong CPU and enough memory to run object detection in the following quarter. Although we should use Arduino for the first quarter but considering the whole project through out 2 quarters, we decide to use Raspberry pi [2], showed in Figure 2.

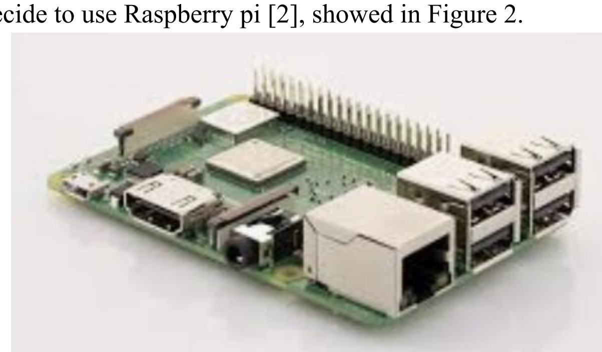

Figure 2. Raspberry pi 3 b+

B. *High-level hardware and method*

For the hardware, we can see from the picture above, there are two boards we use, the upper one is the car driver board, the lower one is the Raspberry Pi 3+. They are connected through all the digital pins, 5v source, and ground. We indirectly control the car through the Raspberry Pi 3+ which will directly send instruction signals to the driver board to control the rotation of the front wheel and the motor of the back wheel. For the front wheel, it needs a digital pin, a ground pin, and a 5v pin. For the back wheel, it needs a voltage difference in two pins to enable the motor to drive. For the LIDAR, it directly connects to the Raspberry Pi 3+. Originally, the LIDAR was placed two times more than the height as it is right now in the picture. We cut down the supporting board to half the size, and place the hardware board off to the side. Now the LIDAR can detect at an appropriate

[1]

height.

C. Software

For software, we were using python and run it on Raspberry Pi. The most important software technique we use in this project is multi-threading programming [3]. For Lidar detection data, we separate a file only for gating valid data. Get_Lidat_data file will get valid data from lidar in every single degree, and store in into a 2D Global array. Delete one of them every half second to ensure the data is most recent. For sharing the Lidar 2D Global array, we had a global variable file. For moving the car which include front wheel rotation and back wheel motoring, we have the test_angle file that contains the __main__ function. The test_angle file is able to use the 2D Lidar global data, and make action base on the environment data. For example, in this file, when circular radius distance is less than 50CM, the car will stop, Or make any rotation. More importance, this file will initially call a threading for lidar to run before moving the car. For moving the back wheel, we control GPIO pins 21, 26, 13, 20.

D. Method

We have designed the following structure figure, this figure shows the main methods and structures of the autonomous vehicles we made.

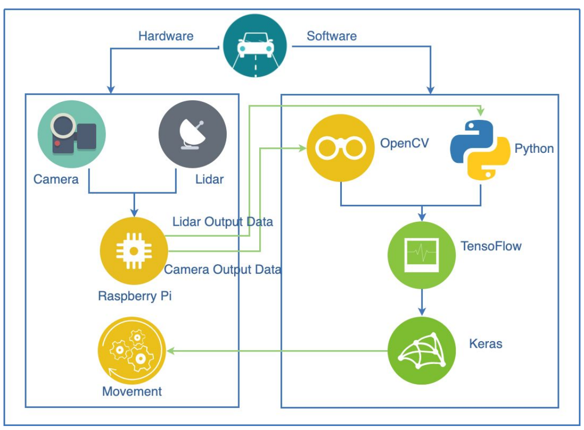

Figure 3. Diagram

E. Result and Performance

To sum it all up, we eventually made the car work and reach our goal. Although the car has some delay, the LIDAR can detect an object within 12 meters. We can set the stop distance larger to ensure the car has enough time to stop.

Also, we have the video to show the results and performance, the link is below: https://www.youtube.com/watch?v=1xQseyqKeCY&feature=youtu.be

III. Summary and Conclusions

A. Summary

Our autonomous vehicle uses a variety of parts to get it moving. Such parts are: a LIDAR, Raspberry pi, a driver board, and a modified RC car. These parts are interconnected and placed on key points in order to ensure that the functionality of the vehicle was excellent. The LIDAR was placed low enough to be able to detect short objects, the raspberry pi and the driver board are placed on top of each other to save space.

We used python to write our code quickly so we can have a general idea of how to structure our code for the quarter. In order to speed up our code, we used a technique called multi-threading to separate different tasks. We kept our data in a file that is in the form of two 2-D arrays that are used for the information that is received by the LIDAR. The main function is left with the responsibility of deciding how to rotate the wheels of the vehicles depending on how data is recorded from the LIDAR which shows objects in the way of the vehicle.

B. Conclusion

The autonomous vehicle is able to turn its wheels left and right and has a working motor and control that can drive the car forward and backward. It has a functioning LIDAR that is able to gather information about solid objects (wall or obstacle) at any degree of rotation. So far halfway through working on this project, we were able to get the basic functionalities of the vehicle up and running. The main features are already set and the car is able to move and stop on its own when detecting an obstacle or a wall.

IV. Appendix

A. Technical Standards

In our project, we use LIDAR for obstacle detection. Direct and high-control lasers are possibly dangerous in light of the fact that they can consume the retina of the eye or even the skin. In order to control the risk of injury, we choose American National Standards Institute (ANSI) Z136 and Code of Federal Regulations (CFR) to be our technical standards. CFR defines "classes" of laser depending on their power and wavelength and ANSI provides users with control measures for laser hazards, as well as various tables helpful in calculating maximum permissible exposure (MPE) limits and accessible exposures limits (AELs) [4].

We choose RPLIDAR A2 model as our LIDAR. The RPLIDAR A2 system uses a low power infrared laser (wavelength between 775 - 795 nanometer and max laser power at 5 milliwatt) as its light source and drives it by using modulated pulse (between 60-90 microsecond). The laser emits light in a very short time frame which can ensure its safety to humans and pets, and it reaches Class I laser safety standard. Complies with 21 CFR 1040.10 and 1040.11 except for deviations pursuant to Laser Notice No. 50, dated June 24, 2007.

B. Constraints

Although our autonomous car project is a simplified simulation of the real-world self-driving car, it must meet the fundamental requirements of the self-driving car, such as safety, reliability, privacy.

Safety is the major concern of autonomous cars. Although the standers of autonomous cars are still under development, we must follow the guidelines of the road vehicle during design and test experiments. For example, the car should drive between the lines, only change lines between dashed lines, and should not pass the yellow line. In our project, we use a digital camera to detect lines, because a digital camera has a higher resolution than a webcam. During testing, we make sure the car follows the path and drives between the lines in agreement with the output of the camera. Driving wobbly is not acceptable. Following the path avoids conflicts with other driving vehicles, and reduces the probability of accidents.

Since the autonomous car is controlled by the software, software reliability is paramount important. Failure in software reliability may lead to inoperative LIDAR and camera, uncontrollable system. It eventually results in car accidents and death. To ensure software reliable, we create test cases to test each functionality of the autonomous car. For example, we have test cases to test LIDAR could recognize obstacles between -60˚ and 60˚ in front of the car; test cases for the car that can read the output of LIDAR and react when there is an obstacle; test cases for the camera to recognize different lines such as dashed line, yellow line.

Finally, privacy should also be taken into consideration. While training the autonomous car, a lot of data needs to be recorded, such as images of people in front of the car. Recording images of people without their consent may cause an ethical dilemma. In our project, we ensure all the images, data, information that recorded are only for academic learning purposes, and will not be reported to other unrelated organizations. Some data such as images will be deleted after a period of time.

C. Hardware and Software Security Issues

In our RC car, we use lithium-ion batteries as a power source. Lithium-ion batteries have a remarkable feature: less self-discharge -- the charges of the battery are unlikely to lose while the battery is not in used. It means lithium-ion batteries can contain more charges and have longer lifetimes. However, lithium-ion batteries have an insecurity issue -- lithium battery fire.

The lithium battery may result in overheating, explosion or fire when it near a heat source, contacts with water or exposes to direct sunlight. To ensure lithium battery safety, we have serval hardware designs. We place the lithium battery at the lowest level of the RC car to avoid direct sunlight. We place the raspberry pi one level above the lithium battery so that they would not physically in touch. We are planning to add a waterproof case on our RC car to shun the water [5].

In software, we use lidar and camera to identify an object and use the raspberry pi to process all the data. There may be an insecure issue that our car can be hacked through the raspberry pi because it has Bluetooth and wifi. All the data from the sensors can be obtained and can potentially show the location of the car and invade privacy and control over the vehicle. To avoid this happened, our image data and lidar data will be clear after a period of time. And also we plan to have a security system to protect our program inside the raspberry pi, such as user login to access program, admin to access data.

References

[1] FierceElectronics. (2019). LiDAR vs. RADAR. [online] Available at: https://www.fierceelectronics.com/components/lidar-vs-radar [Accessed 1 Dec. 2019].

[2] Opensource.com. (2019). Which Raspberry Pi should you choose for your project?. [online] Available at: https://opensource.com/life/16/10/which-raspberry-pi-should-you-choose-your-project [Accessed 9 Dec. 2019].

[3] Docs.oracle.com. (2019). Understanding Basic Multithreading Concepts (Multithreaded Programming Guide). [online] Available at: https://docs.oracle.com/cd/E19455-01/806-5257/6je9h032e/index.html [Accessed 4 Dec. 2019].

[4] Rli.com. (2019). Laser Standards and Classifications. [online] Available at: https://www.rli.com/resources/articles/classification.aspx [Accessed 1 Dec. 2019].

[5]Viceprovost.tufts.edu. (2019). [online] Available at: https://viceprovost.tufts.edu//ehs/files/The-Hazards-of-Lithium-Batteries.pdf [Accessed 11 Dec. 2019].