Autonomous Path Planner Vehicle Using Raspberry PI

Abdul Hady Akash
Department of Computer Science and Engineering
University of Asia Pacific
Email: 17101153@uap-bd.edu
Dhaka, Bangladesh

K.B.M. Tahmiduzzaman
Department of Computer Science and Engineering
Daffodil International University
Email: tahmiduzzaman15-11150@diu.edu.bd
Dhaka, Bangladesh

Md. Thoufiq Zumma
Department of Computer Science and Engineering
Daffodil International University
Email: thoufiq15-10968@diu.edu.bd
Dhaka, Bangladesh

Dipankar Halder
Department of Computer Science and Engineering
Daffodil International University
Email: dipankar15-11137@diu.edu.bd
Dhaka, Bangladesh

Mahmudur Rahman
Department of Computer Science and Engineering
Daffodil International University
Email: mahmudur15-11276@diu.edu.bd
Dhaka, Bangladesh

Naimur Rahman
Department of Electrical and Computer Engineering
North South University
Email: naimur.rahman02@northsouth.edu
Dhaka, Bangladesh

2022 13th International Conference on Computing Communication and Networking Technologies (ICCCNT) | 978-1-6654-5262-5/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICCCNT54827.2022.9984547

***Abstract*— The Autonomous Vehicle (AV) is an intelligent cognitive engineering system that processes streams of observations from various onboard sources, such as cameras, ultrasonic sensors, GPS units, radars, LiDARs, and/or inertial sensors, to guide itself automatically. This allows the AV to avoid collisions, park itself, and navigate to its destination without human intervention. Every day, the most current vehicle series makes little but significant improvements to their vehicles. The most important step in overcoming difficulties with this project is the design of the routes that will be taken. The most important aspect is broken down in this article, and it may be possible to get a superior solution for transportation as well as many other sectors that may help the society in a significant amount with services such as an advanced driving assistance system (ADAS). Within the scope of the project, we will make use of a GPS signal to ascertain the destination in order to enable AV to automatically ascertain the route. An ultrasonic sensor will detect whether or not there is an obstacle in the way of the vehicle. If it senses an impediment in front of the sensor, such as another car, a pedestrian, or another item, the vehicle will be aware of it and make an effort to avoid colliding with it. Using the instructions for the Arduino and the raspberry pi, it will identify the alternate path that leads to its target that is the shortest. There are a variety of techniques that may be used in order to correctly locate the destination. Some of these strategies include simultaneous localization and mapping (SLAM). Planning your route using GPS technology might be one of the most efficient methods to go to any location. If the route planning is successful, there is still a chance that the issue can be resolved. The fundamental objective of this article is to map out the route that AV will take so that we may successfully complete the mission.**

Keywords—Raspberry pi, Arduino, Autonomous vehicle, GPS, LiDAR, sensor.

I. INTRODUCTION

The modern world increasingly depends on self-operating technologies. As opposed to humans, machines can operate nonstop without becoming tired. Repetition may lead to mental disconnection if the mind is not challenged. Finally, my reaction to every unexpectedly difficult scenario. This project's objective is to develop an autonomous self-sailing system that is capable of avoiding obstacles, performing simultaneous localization and mapping (SLAM), and determining its location on a solid surface by using a low-cost inexpensive Lidar sensor. It can also measure soil conditions and outside surface conditions in real time. The design application includes sensing real-time humidity and temperature, as well as outer surface condition with real-time sky mood by 180* angle image. It is a very sophisticated car that can use for both inside and outer surface conditions. Through its own decision making, the design will be able to avoid obstacles and chart a path.

We performed everything, including sensor location and vehicle structure. This research will use light detection and ranging to study the difficulty of obstacle avoidance in autonomous vehicles. It also involves creating smart approaches to acquiring a localized location from processing sensory input in the simplest feasible manner, including the capability of real-time obstacle avoidance, adjusting the route planning algorithm, and providing a justification for the findings. This technology allows the vehicle to go in both familiar and unknown terrain. There's a chance it's reacting to information entered by users. Because of its small size, it would be able to go through incredibly strong surfaces. The use of plastic raw materials and glass fiber wrapping makes it stiffer and more long-lasting

II. RELATED WORK

In [1], the authors defined an autonomous vehicle that uses the open CV method to detect the path using symbols on roads. The pictures taken with the Raspberry Pi camera are processed by OpenCV, and the processed signal is then sent digitally to trigger the next action. In [2] order to speed up the search, the author creates an obstacle-sensitive cost function.

When the car loses its way, an obstacle blocks its path, or accurate localization inside the predefined road space is impossible. In [3], the author displays and ITS, which is a system that uses DSRC to enable two-way data communication between vehicles (V2V), infrastructure (V2I), and other objects (V2X) to enhance road safety and efficiency. In [4], the authors construct a prototype autonomous vehicle with monocular vision utilizing Raspberry Pi to collect data from the outside environment through an ultrasonic sensor and a camera. Line-detection and abstract detection algorithms allow the automobile to arrive at its destination without endangering the passengers or the general public and without the need for human intervention.

In [5], the authors explain the background history of the self-driving car, and future uses, and also built a prototype self-driving car. The writers of [6] create an automated robot called an Integrated Autonomous Vehicle (IAV) that can go to any part of the security zone without raising suspicion among the adversary. It has potential uses in both bomb disposal and the identification of suspicious devices. In [7], the authors present a robotic agent platform that uses Arduino and RaspBarry pi to automate hardware functions and Jason to provide intelligence resourcing. Programmers don't need to simulate the environment, just upload the robotic agent and it will make a communication path between them.

Previously, researchers used ultrasonic sensors to build automated cars with Arduino and Raspberry Pi. We integrate our project's ultra-sensors and a camera module, which are always on. When the ultrasonic sensor suddenly stops working, the camera module switches in and the vehicle rides perfectly.

III. METHODOLOGY AND WORKING PRINCIPLE

A. Methodology

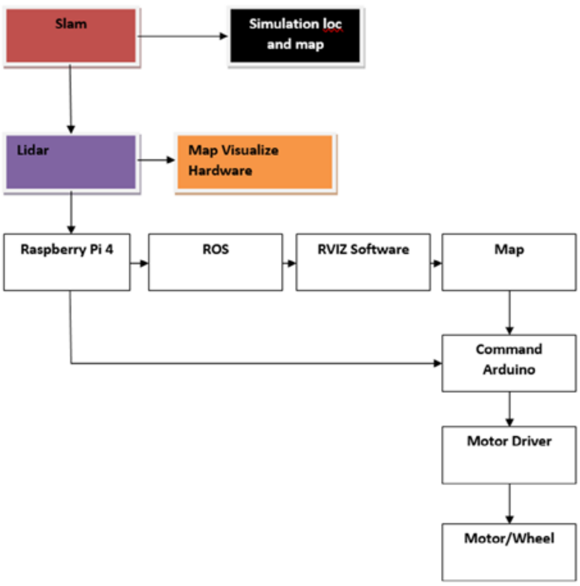

Figure 1: Flow-chart of methodology

The universe is undergoing a scientific revolution. Some people lose their vision, while others are born blind. To carry out this type of endeavor would represent a significant social revolution. The sensory handicapped should be able to communicate more effectively for social and educational objectives. Following the project and the addition of its many characteristics, the block diagram reflects the operation of the project, and the flowchart elaborately illustrates the operation of the project. The chapter explains and describes the project's methodology and models.

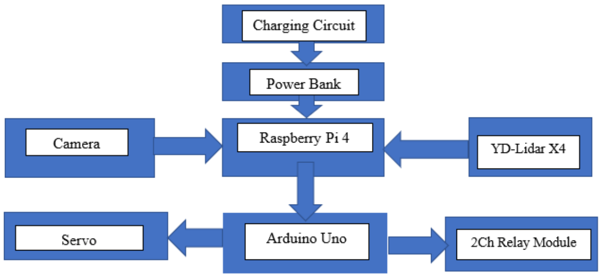

Figure 2: Block diagram of the system

It is extremely likely that a microcontroller is used to do the task. The Raspberry Pi is used as a microcontroller that is gracefully connected to a force. Within the context of this project, the primary responsibility of the Raspberry Pi is to gather the data and transmit it to the Arduino Mega. The servo is operated by the Arduino Mega, and the hand-off completes the following tasks. The raspberry pi's python code is used to do this task. The Raspberry Pi has a camera as one of its source inputs. YD-Lidar x4 contributes to the raspberry pi 4 by drawing energy from the force bank. Finally, a device that customers use will display the yield on its screen. In this manner, it completes the entire task for a person with hearing loss. A camera is used as information for someone who is blind. The information sent to the raspberry-pi is captured by the camera.

B. Software working

The element of the software implementation system that uses microcontroller programming is really basic. First, make sure that the host and vehicle controlling unit devices have all the required wires connected. Launch the necessary programs next in order to run the code on both devices. The vehicle is finally prepared for use when the admin enters the host command. The crucial software, in this case, is the VNC server, Arduino IDE, and Python IDLE.

Implementation of Vehicle Controller Unit

Mapping strategy, real-time location detection, and simultaneous command progression to the LIDAR sensor are all implemented using simultaneous localization and mapping (SLAM). The Raspberry Pi 4 computer gets the governing algorithm from the single-board Lidar sensor, which also promotes the map Visualization hardware modules for obstacle detection. The Raspberry Pi acts as the command center, sending orders to the ROS and the microcontroller.

Robot Operating System receives instructions from Raspberry Pi computer (ROS). ROS is made up of code and tools that allow the project's code to be executed, data to be assembled and sent to the 3D visualization tool RVIZ for modeling robot models, sensing data to be recorded from the sensors, replayed, and used to help in data collecting for the map. The map shows the current position and sends that information to the Arduino that is in charge of giving commands to other devices. Following the analysis of all the gathered data and information, the controlling Arduino sends the necessary instructions to the motor driver, who in turn controls the motor and the wheel section.

The Vehicle Controller Unit (VCU) is connected to the host computer utilizing a socket connection. We are first attempting to connect a socket with this line.

```
s=socket.socket(socket.AF_INET,socket.SOCK_STREAM)
```

and we are printing when the connection is complete. The Arduino is connected statement was written after binding the post and host to the socket and adding the serial connection. Next, we tried to listen to the socket by supplying 1 as an argument, and when it accepted the connection, it printed the connected address. Data has been received when the data is received, and if not, a break statement is printed. If it gets w, we're trying to get the VCU to move ahead; if it gets b, we're trying to get them to go backward; similarly, if it gets l, we're instructing them to go left; r signifies right, and z means stop.

Host Controller Implementation:

While you're working on the automobile, connect all of your networks to the same Wi-Fi. The next step is to activate the VNC server and establish a connection between the host PC and a Raspberry Pi desktop interface. Launch the program after that on both computers when Python is idle. Everything is now prepared to depart. Now, a user may instruct the host unit to drive the car.

When we hit the up key after connecting to VCU, we actually pass VCU and instruct you to proceed ahead, and when we press the left and right keys, we truly pass left and right.

Our ambitious project operates through the process of installation and obstacle removal. The host controller unit and the vehicle moving component are the two components that make up the entire system. When the two parts are put together, our prototype car transforms into a completely autonomous vehicle. Part smart executed the framework performance. After each portion was finished, it was tested a few times before different sections were given. Before usage, every component, including the battery and others, underwent testing. After finishing the system hardware consumption, the experiment's convincing conclusion was obtained. A number of enhancements are added, and acceptable feedback is received, which shows that our project was successful. The host controller unit and the vehicle moving portion are the two components that make up the entire system. Part savvy had completed the framework's implementation. After each portion was completed, it was tested a few times before various parts were carried out. Before usage, every component, including the battery and others, was tested. Finally, system hardware use was completed, and we received the experiment's conclusive outcome.

C. Working Principle

When the input parameters are individually configured to observe all potential outputs, the output of the individual observation is described in this section. A microcontroller and other components helped us find our results. Below are the results for each part:

Process:

Our robot follows its programming to function as an obstruction-avoiding robot. When it encounters an obstruction on its course, it identifies it before 10 cm, and the survey shifts the Sonner to the right or left depending on whether there is another obstruction there. It goes backward if there are obstructions in its path. This vehicle uses SLAM to build a local map. To exchange its data, this car use a local network. With VNC, we can view all data.

Ultrasonic Sensor Reading:

A cm/m/mm reading is taken by the sonar sensor. Here, we use cm. So, here is the sonar sensor's reading. In order to process the readings, the Sonar sensor transmits them to an Arduino Uno. The motor driver received the data processing command from Arduino, which it accepted.

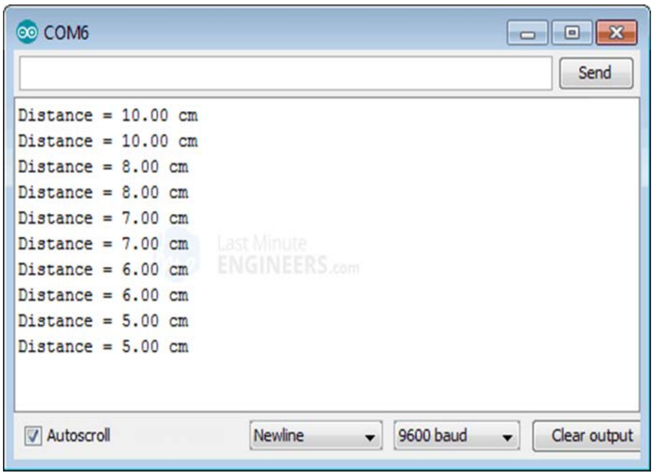

Figure 3: Ultrasonic sensor reading

Lidar Output:

Here are some outputs we saw when turning on the automobile. SLAM is a technique that simultaneously reconstructs the geometry of an area using sensor data and the motion of its sensors. Vehicles collect LIDAR data to create a local map. The data from the 360-degree Lidar camera is sent to a Raspberry Pi 4 computer. After that, the Raspberry Pi receives the raw Lidar data and processes it using the RVIZ software from the ROS operating system before sending it on to the Arduino Uno. The driver of the vehicle is then contacted by text when the information has been received. Next, the driver begins to spin the motor's wheel.

Subject	Minimum value	Typical value	Maximum value	Unit	Remarks
Range sample frequency	4000	9000	9000	Hz	9000 ranging times per second
Scan frequency	5	7	12	Hz	Software speed regulation
Ranging range	0.10	-	16	m	Range frequency=4KHz
	0.22	-	16	m	Range frequency=8KHz
	0.26	-	16	m	Range frequency=9KHz
Angular Range	-	0-360	-	Deg	-
Distance Resolution	-	<0.5	-	mm	Ranging range <2m
		< 1% of actual distance			Ranging range >2m
Angular resolution	0.26	0.28	0.30	Deg	Scan frequency =7 Hz

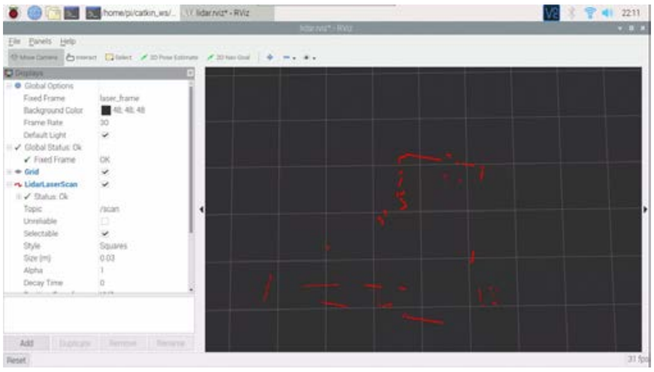

Figure 4: Lidar Outputs

Obstacle Detection:

The Lidar module we used has a run-up of 10 meters and can record from 0 to 360 degrees. These estimates are returned by LIDAR as a point cloud. Every piece of information has three numbers: degrees, separate, and flag quality, and it reflects an obstruction. We divide estimates into 4 classes based on the degrees: forward, cleared out, right, and back. Long Run (LR) and Short Range (SR) estimates are the two categories into which we split measurements for each session. Measurements between cm and 110 cm are classified as SR, whereas those between 111 cm and 1000 cm are classified as LR. In comparison to estimates in LR, SR estimates are given more weight. Preprocessing sifting is used to remove guesses that are no longer needed, hence accelerating calculations. The crucial estimations, for instance, are located in the front category while the vehicle is traveling forward. In order to increase the calculation's processing speed, we should direct estimates and obstructions that are not used in other bearings.

Decision-Making Algorithm:

The calculation's goal is to provide the car the optimal course of action at any given time so that it can keep moving in the direction of its intended destination. It takes into account things like the route taken, the engine's health, the GPS's trajectory, and any surrounding obstacles. Until a deterrent is found, the computation steers the vehicle in the direction of the general principle. The algorithm then evaluates cleared out and right to determine whether a turn is plausible when the vehicle cannot continue moving forward owing to a recognized obstruction. The algorithm prefers the heading that is closest to the objective position when both the cleared-out and right bearings are free of obstructions. If all routes are blocked, the algorithm sends the car on a turnaround track while simultaneously looking for other possible turns.

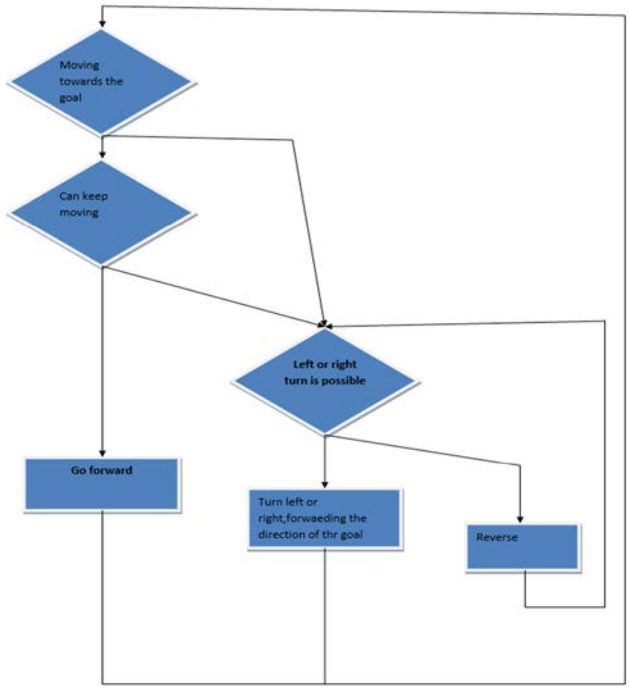

Figure 5: Decision-making algorithm

IMU Data Handling:

Triaxial accelerometers detect the absolute three-dimensional rising speed around the body shape. Since the tri-axial accelerometers, while the versatile robot is stationary offer very exact data, the tilt of the portable robot is assessed using three orthogonal accelerometers. Additionally, the speed and location are given by the single and double acceleration integration. Exact rates for the three tomahawks are provided by the three orthogonal gyroscopes. The introduction of the portable robot is made possible by the integration of accurate rates. When the versatile robot is moving, as well as when it is motionless, the integration of the 26 tri-axial gyroscopes expands the introduction. The gyroscope's precise rates are rectified using the magnetometer's information.

Travel Trajectory:

Unknown are the routes that the car will take. However, the path of the next car is shown below while routing the current one.

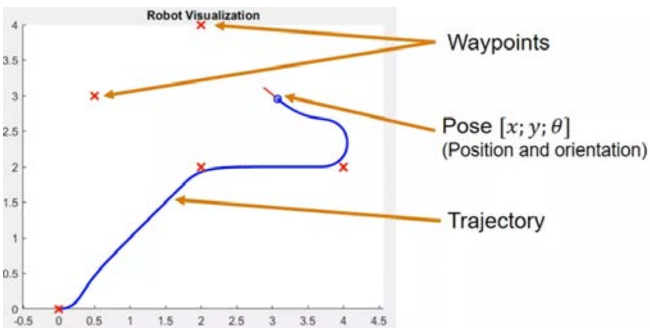

Figure 6: Travelled Trajectory

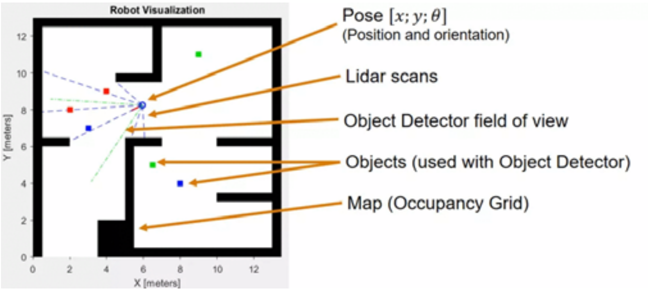

Figure 7: Travelled Trajectory

An autonomous vehicle running on a Raspberry Pi and LIDAR is shown. To avoid possible dangers, the automobile might utilize the existing Wi-Fi network's location services to go to its destination. Creating a MAP of the region around us as we go might greatly facilitate our work if we decide to use this mapping method. In order to predict hindrance growth and explore more effectively in high-energy circumstances, adjustments to the computation may also be performed.

A project's quality can be distinguished in part by how effectively it was analyzed. Recognizing the systematic approach to examining the complete result is crucial. In this chapter, many parameters that were specified as inputs to the project's output were observed. The outcome demonstrates the success of the entire project and will have a big effect on the user. However, despite the air surface posing some challenges to sensor readings, our project is functioning as intended. It is evident from the overall project's outcome analysis that the outputs are accurate and operating as intended.

IV. HARDWARE COMPONENTS

A. Raspberry Pi

Figure 8: Raspberry Pi

A tiny single-board PC powered by a Raspberry Pi microcontroller is seen in Figure 4.5 above. It is functional without a mouse or console. Python is the designated programming language for the Raspberry Pi. [13] It is capable of everything a PC is. The project uses the Raspberry Pi as a microcontroller and the Python programming language to convert speech to text, images to text, and text to sound.

B. Raspberry Pi Camera Module

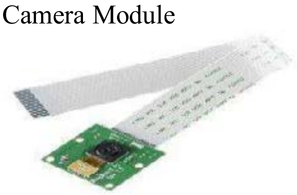

Figure 9: Raspberry Pi Camera Module

The Raspberry Pi camera was mentioned. A lightweight and versatile camera, the Pi Camera Module is compatible with the Raspberry Pi. It uses the standard MIPI camera sequential interface to talk to Pi [14]. It sees regular use in endeavors involving the manipulation of visual content, AI, and surveillance. Since the camera's payload is so low, it is frequently used in observation drones. In addition to these modules, Pi may make use of regular USB cameras that are connected to a PC. The camera was used to take photographs, and the python programming language was used to operate it. It captures the image for handling to turn the image to a message through image preparation.

C. Ultra-Sonic Sensors

Figure 10: Ultra-Sonic Sensors

With regard to an ultrasonic sensor, see Figure 10. This device measures an article's separation using sound waves. In order to prevent deterrents from being overlooked by a small sensor bar, they also provide lower least separations and wider margins of identification. A visually challenged person can use the sensor to identify the objects, making movement easier. [12]

D. Camera Module

Figure 11: Camera Module

With a fixed central focus point and a 5-megapixel resolution, the Raspberry Pi Camera Module is a specially designed add-on for the Raspberry Pi [15]. It supports static images with a resolution of 2592 x 1944 pixels as well as video in 1080p30, 720p60, and 640x480p60/90.

E. Infrared Sensor

Figure 12: Infrared Sensor

An electrical device called an infrared sensor emits light to pick up some ambient elements [16]. An IR sensor can detect movement as well as measure how warm something is. These sensors are referred to as passive IR sensors since they do not send infrared radiation but instead just measure it.

F. Servo Motor

Figure 13: Servo Motor

A spinning actuator or a straight actuator that takes into account exact control of direct or rakish position, speed, and quickening is referred to as a servo engine [17]. It has a decent engine connected to a sensor that inputs position.

G. Wi-Fi Module

Figure 14: Wi-Fi Module

Any microcontroller may connect to your wireless network thanks to the ESP8266 Wi-Fi Module, which is a standalone SOC with an integrated TCP/IP protocol stack. Assisting an application or taking over full Wi-Fi management duties, the ESP8266 is a powerful microcontroller [18].

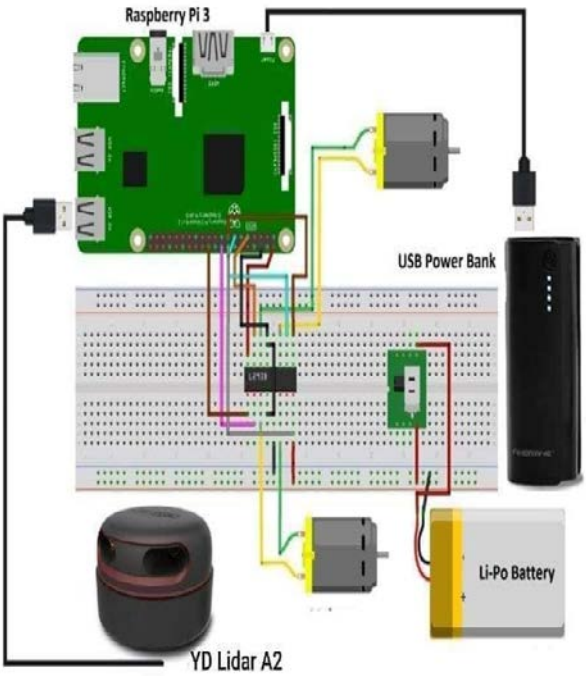

Figure 15: Simulation of the Vehicle

V. EXPERIMENTAL RESULTS

It demonstrates a Raspberry Pi and LIDAR-powered autonomous vehicle. The car may use the existing Wi-Fi network to pinpoint its position, navigate between predetermined points, and maintain a safe distance from obstacles while it does so. Our efforts may be improved by engaging in a mapping activity, such as making a MAP of the immediate region while we navigate. To predict the emergence of obstacles and conduct exploration more effectively in high-energy conditions, adjustments to the computation may also be performed.

An important factor in judging a project's quality is how effectively it was analyzed. Recognizing the systematic approach to taking the full outcome into consideration is crucial. This chapter examined the project's outcome after setting various settings as inputs. The outcome shows how well the entire effort was accomplished and will have a big effect on the user. Although our project is functioning well, the air surface makes it difficult for the sensor to obtain a reading. It is clear from the project's overall outcome analysis that the outputs are accurate and functioning as instructed.

Figure 16: Prototype of the model vehicle

VI. CONCLUSION

The results of this plan suggest that, under the given circumstances, a development will, on the one hand, generate a sufficient number of inhabitants, grant free parking to parcels in inactive town zones, and offer access to public transportation systems without a driver's license. On the other side, the autonomous car will probably result in less vehicle gathering. In the mid-to long-term, the developing countries' demand will continue to grow, while the developed countries' demand will progressively decline. The AV technology is hoping to be a game-changer in the competition to stop market shares that competes with the most important automobile bunches worldwide, much as the ability in the electric vehicle. However, the method that the main on-screen actors might use to assert their dominance in this upcoming advertisement must take into account morality just as much as it does business-related issues. This work's limitations remain that it cannot collect data and store it in memory. In future work, we will be focusing on computer vision and deep learning approaches.

REFERENCES

[1]B. Sahu, B. Kumar Sahu, J. Choudhury and A. Nag, "Development of Hardware Setup of an Autonomous Robotic Vehicle Based on Computer Vision Using Raspberry Pi", *2019 Innovations in Power and Advanced Computing Technologies (i-PACT)*, 2019. Available: 10.1109/i-pact44901.2019.8960011 [Accessed 8 August 2022].

[2]J. Ziegler and M. Werling, "Navigating car-like robots in unstructured environments using an obstacle sensitive cost function", *2008 IEEE Intelligent Vehicles Symposium*, 2008. Available: 10.1109/ivs.2008.4621302 [Accessed 8 August 2022].

[3]S. Tayeb, M. Pirouz and S. Latifi, "A Raspberry-Pi Prototype of Smart Transportation", *2017 25th International Conference on Systems Engineering (ICSEng)*, 2017. Available: 10.1109/icseng.2017.25 [Accessed 8 August 2022].

[4]G. SinghPannu, M. Dawud Ansari and P. Gupta, "Design and Implementation of Autonomous Car using Raspberry Pi", *International Journal of Computer Applications*, vol. 113, no. 9, pp. 22-29, 2015. Available: 10.5120/19854-1789.

[5]D. T., "Self Driving Car", *International Journal of Psychosocial Rehabilitation*, vol. 24, no. 5, pp. 380-388, 2020. Available: 10.37200/ijpr/v24i5/pr201704.

[6]K. Swain, S. Dash and S. Gouda, "Raspberry PI based Integrated Autonomous Vehicle using LabVIEW", *2017 Third International Conference on Sensing, Signal Processing and Security (ICSSS)*, 2017. Available: 10.1109/ssps.2017.8071567 [Accessed 8 August 2022].

[7]N. Lazarin and C. Pantoja, "A Robotic-agent Platform for Embedding Software Agents Using Raspberry Pi and Arduino Boards", 2015. [Accessed 8 August 2022].

[8]*Raspberrypi.org*, 2022. [Online]. Available: https://www.raspberrypi.org/learn/. [Accessed: 08- Aug- 2022].

[9]"What is Arduino?", *Arduino.cc*, 2022. [Online]. Available: https://www.arduino.cc/en/Guide/Introduction. [Accessed: 08- Aug- 2022].

[10]"Tinkercad | From mind to design in minutes", *Tinkercad*, 2022. [Online]. Available: https://www.tinkercad.com/. [Accessed: 20- Aug- 2022].

[11]"Software", *Arduino.cc*, 2022. [Online]. Available: https://www.arduino.cc/en/software. [Accessed: 20- Aug- 2022].

[12]"What is an Ultrasonic Sensor? | Sensor Basics: Introductory Guide to Sensors | KEYENCE", *Keyence.com*, 2022. [Online]. Available: https://www.keyence.com/ss/products/sensor/sensorbasics/ultrasonic/info/. [Accessed: 21- Aug- 2022].

[13]"What is a Raspberry Pi?", *Opensource.com*, 2022. [Online]. Available: https://opensource.com/resources/raspberry-pi. [Accessed: 21- Aug- 2022].

[14]"Raspberry Pi Documentation - Camera", *Raspberrypi.com*, 2022. [Online].Available:https://www.raspberrypi.com/documentation/accessories/camera.html. [Accessed: 21- Aug- 2022].

[15]"What is Camera Module | The ultimate introduction", *Supertek*, 2022. [Online]. Available: https://www.supertekmodule.com/what-is-camera-module/. [Accessed: 21- Aug- 2022].

[16]"Infrared Sensors Overview: Types, Functioning and Use Cases | Kisi", *Getkisi.com*, 2022. [Online]. Available: https://www.getkisi.com/guides/infrared-sensors. [Accessed: 21- Aug- 2022].

[17]"How Servo Motors Work", *Kollmorgen.com*, 2022. [Online]. Available: https://www.kollmorgen.com/en-us/blogs/how-servo-motors-work/. [Accessed: 21- Aug- 2022].

[18] T. Agarwal, "ESP8266 Wi-Fi Module Datasheet : Working & Its Applications", *ElProCus - Electronic Projects for Engineering Students*, 2022. [Online]. Available: https://www.elprocus.com/esp8266-wi-fi-module/. [Accessed: 21- Aug- 2022].