Development of an Obstacle Avoidance Robot with Edge Detection Capability

John Ogunleye and Kazeem B. Adedeji

Department of Electrical and Electronics Engineering, Federal University of Technology Akure, Ondo State, Nigeria
Email: iamjohnofficial001@gmail.com; kbadedeji@futa.edu.ng

***Abstract*—The growing demand for autonomous robotic systems capable of navigating complex and unstructured environments has brought about significant advancements in the field of robotics. This study focuses on the design and implementation of an autonomous robot equipped with edge detection and obstacle avoidance capabilities to address safety and efficiency challenges in transportation and exploration. Extensive real-world testing and validation were conducted to ensure the robot's reliability and efficacy in edge detection and obstacle avoidance tasks. The result obtained shows that the robot can navigate through unfamiliar environments while autonomously detecting edges and avoiding obstacles in real-time.**

Keywords—edge detection; obstacle avoidance; robot; ultrasonic sensor

I. Introduction

Robotics has emerged as a rapidly growing and intriguing field of research, transforming modern technology and driving innovation. Among its fascinating aspects, autonomous robots stand out for their ability to navigate unstructured and unknown environments without external intervention. These intelligent machines rely on sophisticated software intelligence to sense their surroundings, detect obstacles, and autonomously navigate around them. Today, obstacle avoidance is a common feature in many robots as it is used in a variety of applications, such as warehouse robots, indoor cleaning robots, autonomous vehicles, etc. It makes life easier for individuals and assists in moving goods around warehouses, cleaning floors and other surfaces in homes and offices, driving vehicles autonomously with human input and also in drone technologies. The obstacle avoidance capability is of great importance to avoid collision with obstacles which reduces environmental hazard.

As robots become more sophisticated, the need for improved obstacle avoidance techniques will become increasingly important. This is because robots will be operating in more complex and dynamic environments. One promising area of research is the development of 3D obstacle avoidance algorithms which takes into account the height of obstacles, which is important for robots that need to navigate in cluttered environments. Another area of research is the development of obstacle avoidance algorithms that can learn from experience [1]. These algorithms would be able to improve their performance over time as they encounter new obstacles. In addition to developing new algorithms, researchers are also working on developing new sensors that can help robots better perceive their surroundings. For example, lidar sensors can create a detailed 3D map of the environment which can be used by obstacle avoidance algorithms to plan safe paths [2-5]. The development of improved obstacle avoidance techniques is essential for the future of robotics. As robots become more autonomous, they need to be able to navigate complex environments safely and efficiently. The research in this area is making significant progress, and it is expected to see even more advances in the years to come. This study developed an autonomous obstacle avoidance robot with edge detection features.

II. Literature Review

A. Overview of the Field of Robotics

The field of robotics is a multidisciplinary branch of science and engineering that focuses on the design, construction, operation, and application of robots. A robot can be regarded as a machine capable of carrying out a complex series of actions automatically, especially one programmable by a computer. Robots are autonomous or semi-autonomous machines that can perform tasks and interact with their environment without direct human intervention. Robotics encompasses the study of creating machines (robots) that can manipulate objects, move in various environments, and perform tasks, often with the goal of automating and simplifying complex or repetitive tasks. Robots are designed to act on their decision-making ability or to be controlled by humans. Robots work by trying to mimic human behaviour as they are made to possess the same components of a human being. These components include:

i) Muscle systems that move body systems
ii) Body structure.
iii) Power source used to activate sensors and neurons
iv) Sensory system for acquiring environmental information

v) A brain system that processes the sensed information and gives the muscles information on how to respond.

In recent times, robots are autonomous as they represent a class of machines capable of exhibiting a high degree of self-governance, without the need for external control. Achieving true autonomy involves a sophisticated blend of artificial intelligence, robotics, and information engineering. These robots, much like humans, possess the ability to make independent decisions based on sensory data and subsequently execute actions in response to their surroundings. In terms of mobility, this entails decision-making processes encompassing actions like initiating movement, coming to a halt, and navigating around obstacles obstructing their path. A common feature in the sensory repertoire of these robots is the utilization of ultrasound sensors or infrared technology. These sensors function akin to the echolocation seen in certain animals. By emitting a signal in the form of light or sound and measuring the time taken for it to return after bouncing off objects, these robots can perceive and detect obstacles in their environment [6, 7]. Advanced robotic systems employ stereo vision techniques with two cameras to discern depth and identify objects [8-10]. This enables them to categorize and react to different objects effectively. Some cutting-edge robots even possess the capability to adapt to new and unfamiliar environments, effectively learning and adjusting their behavior. They are capable of tackling challenging terrains, such as rough landscapes. For instance, a rover robot (Fig. 1) constructs a terrain map using its visual sensors, allowing it to respond dynamically to uneven terrain by selecting alternative routes. These systems hold tremendous promise for exploration, including planetary missions.

Fig. 1. *A rover robot.*

One illustrative example of these advanced autonomous systems is NASA's Urbie robot shown in Fig. 2. The robot is designed for various military applications. Urbie showcases impressive capabilities, including the ability to navigate stairs and other challenging paths. When faced with obstacles, Urbie employs a combination of sensory data and force sensors to resolve issues autonomously, minimizing the need for centralized control and programming instructions. This signifies a significant leap in the field of autonomous robotics, where robots exhibit remarkable self-sufficiency and adaptability.

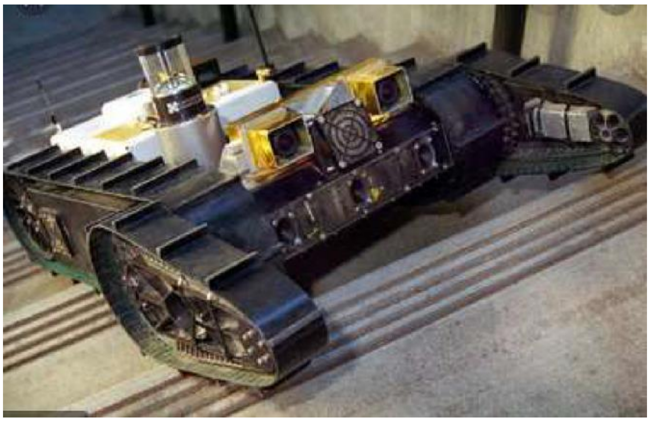

Fig. 2. *Autonomous Urbie by NASA [11].*

B. *Review of Studies on Obstacle Avoidance and Edge Detection*

Several studies have been conducted on the development of obstacle avoidance robot. Vairavan *et al*. [12] developed an intelligent obstacle-avoiding robot car using ultrasonic sensors and motor control systems. The robot autonomously detects obstacles and determines a collision-free path in real-time. An Arduino microcontroller serves as the central control, enabling the robot to alter its course based on sensor inputs. This study advances autonomous navigation by integrating sensor technology and decision-making in robot cars. In Li *et al.* [13], an intelligent Arduino-based wheel robot was developed for obstacle avoidance and path tracking. The robot uses infrared tracking, ultrasonic obstacle detection, motor control, and power management to navigate autonomously. The study addresses challenges in automatic path following and obstacle avoidance. The authors highlight the growing use of such robots in material transport, military operations, and scientific research, contributing to smarter, more practical autonomous robots.

Yasin *et al.* [14] developed a cost-effective collision avoidance algorithm for autonomous vehicles using a single ultrasonic sensor. This method minimizes path deviation by rotating the sensor to detect object shapes and edges. Tested in various indoor scenarios, the algorithm effectively avoids obstacles and reroutes the vehicle with minimal deviation, demonstrating the potential of low-cost sensors for reliable autonomous navigation.

Neloya *et al.* [15] propose a real-time obstacle-avoiding robot utilizing the bug algorithm for edge detection and path finding. Constructed with an ATmega8 microcontroller (Arduino Uno R3) and an ultrasonic sensor, the robot detects obstacles and edges, sending signals to the microcontroller. The microcontroller then directs the robot to an alternative path via motor actuation. This design enables the robot to make spontaneous decisions for obstacle avoidance, showcasing its potential for autonomous navigation in dynamic environments.

Pandey *et al.* [16] reviewed various navigation and obstacle avoidance methods for enhancing mobile robots' autonomous capabilities. The study highlighted the growing importance of these robots in industry, space, defense, transportation, and social sectors, performing tasks such as rescue operations, patrolling, disaster relief, planetary exploration, and material handling. The authors emphasized the need for intelligent navigation techniques for autonomous operation in both static and dynamic environments.

Borenstein and Koren [17] introduced the Virtual Force Field (VFF) method for real-time obstacle avoidance in mobile robots. This method combines certainty grids for obstacle representation and potential fields for navigation, effectively managing inaccurate sensor data and enabling sensor fusion. The VFF allows continuous robot motion without stopping for obstacles and resolves the local minimum trap problem. Experimental results with a robot moving at 0.78 m/s demonstrate the method's effectiveness in dynamic environments.

III. Materials and Methods

Fig. 3 shows the block diagram of the process involved during the development of the obstacle avoidance robot. At first, the necessary materials needed for the project were selected. Thereafter, the hardware design commences. Under this section, the components are placed on the vero board and soldered. The algorithms needed for the edge detection were written on the microcontroller through the Arduino IDE. After completion, the entire system was tested for obstacle avoidance and edge detection.

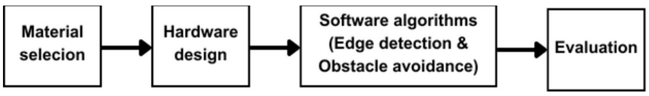

Fig. 3. *Block diagram of the development process.*

A. Material Selection

Table I shows the list of components and their specification used for the design and implementation of this robot.

B. Hardware Desing

During the construction of the hardware framework for the robot, the necessary components such as ultrasonic sensors, infrared sensors, L293D motor driver ICs, LM7805, etc. were placed on the veroboard and connected together through precise soldering techniques while the motors were also attached to the chassis and then connected to the tyres. The block diagram in Fig. 4 shows how each of the hardware is connected. In Fig. 4, four ultrasonic and infrared sensors are connected to the Arduino MEGA microcontroller powered by a 5 V power supply.

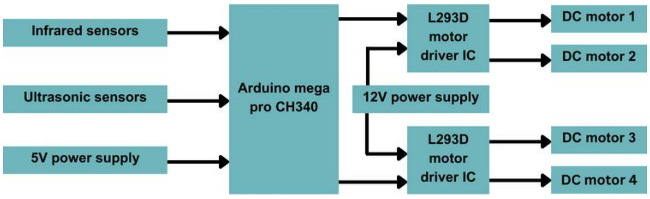

Fig. 4. *Block diagram showing the hardware framework.*

The MEGA 2560 Pro utilizes an onboard CH340G universal asynchronous receiver transmitter (UART). A separate safe driver is needed to allow the host computer to communicate with the MEGA 2560. It is strongly encouraged to download and install the suitable driver for the operating system (windows, mac, linux, etc) so as to allow the mega pro CH340 to properly communicate with the host computer for upload of required codes to the mega pro CH340 microcontroller.

Two L293D motor driver ICs were used to control the speed and direction of the four DC motors attached to the robot's chassis. The IC is mounted on a veroboard, soldered and connected to the Arduino Mega Pro CH340 microcontroller board, the DC motors, and power supply as shown in Fig. 4. The pins of the IC and other components were soldered to the copper pads on the veroboard to ensure a strong and conductive connection. The code for the motor control was written using the Arduino programming language and uploaded to the microcontroller board using the Arduino IDE software and a USB cable. The power supply was then connected to the breadboard and turned on. The functionality of the motor driver IC was tested by running the code on the microcontroller board and observing the behavior of the motors. The code was also modified and improved as needed to achieve the desired performance and functionality of the motors. The L293D motor driver IC is a vital component of the robot, as it enables the robot to navigate its environment and perform its tasks.

(a) Pin configurations for L293D motor driver IC used

Table II shows the description of each of the pins on the l293d motor driver IC. The L293D motor driver IC is housed in a 16-pin DIP package. Its pins include enable and input pins for each of its two motor outputs, enabling direction and speed control. Ground and V_{CC} (Vs and V_{ss}) pins provide necessary power connections, with V_{ss} typically set to 5 V. Each output pair (OUT1/OUT2 and OUT3/OUT4) corresponds to a motor terminal, with IN1/IN2 and IN3/IN4 pins controlling the direction of rotation for each motor. Enable pins (EN1, EN2, EN3, EN4) determine whether the motor outputs are active or inactive.

TABLE I. LIST AND SPECIFICATION OF THE COMPONENT USED.

S/N	Component used	Qty	Rating
1	Arduino Mega	1	5 V
2	7805 IC	1	12 V
3	18650 rechargeable battery	3	11.1 V
4	Four Wheel Drive (4WD) Robot Chasis kit	1	6 V
5	Infrared sensor	4	20 V
6	Ultrasonic sensor	4	20 V
7	Switch	1	-
8	Patreas box	1	-
9	L293D motor driver IC	1	12 V
10	TP4056 18650 battery charging module	1	5 V
11	Battery holder	1	-
12	LEDs	4	12 V
13	Pack of connecting cables	1	-

TABLE II. DESCRIPTION OF L293D MOTOR DRIVER IC PINS.

Pin number	Pin name	Description
1	Enable 1,2 (EN1,EN2)	This pin enables the input 1 and input 2 pins
2	Input 1 (IN1)	Directly controls the Output 1 pin. Controlled by digital circuits
3	Output 1 (OUT1)	Connected to one end of Motor 1
4	Ground (GND)	Ground pins are connected to ground of circuit (0V)
5	Ground (GND)	Ground pins are connected to ground of circuit (0V)
6	Output 2 (OUT2)	Connected to another end of Motor 1
7	Input 2 (IN2)	Directly controls the Output 2 pin. Controlled by digital circuits
8	Vcc1 (Vs)	Connected to Voltage pin for running motors (4.5V to 36V)
9	Enable 3,4 (EN3, EN4)	This pin enables the input pin Input 3(10) and Input 4(15)
10	Input 3 (IN3)	Directly controls the Output 3 pin. Controlled by digital circuits
11	Output 3 (OUT3	Connected to one end of Motor 2
12	Ground (GND)	Ground pins are connected to ground of circuit (0V)
13	Ground (GND)	Ground pins are connected to ground of circuit (0V)
14	Output 4 (OUT4)	Connected to another end of Motor 2
15	Input 4 (IN4)	Directly controls the Output 4 pin. Controlled by digital circuits
16	Vcc2 (Vss)	Connected to +5V to enable IC function

Fig. 5 shows how these pins are connected for the motors.

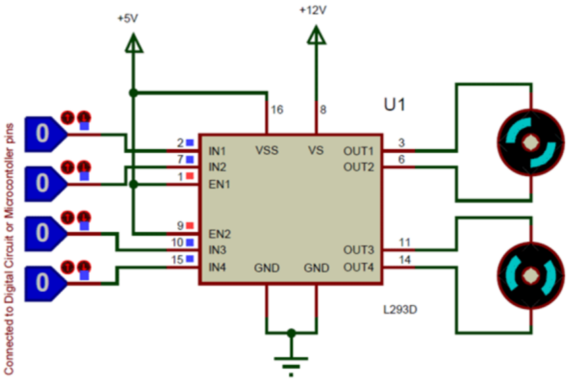

Fig. 5. *L293D motor driver circuit diagram.*

The L293d motor driver IC consists of 16 pins in total, 8 at the left and 8 at the right in which a motor is connected at the output pins (1 and 2) of the IC at its right side and another motor connected to the output pins (3 and 4) of the motor at its left side thereby making it a total of two motors controlled by one of the IC. The input pins of the L293d are connected to the digital I/O pins of the microcontroller while the V_s pin is connected to supply voltage of 12 V. V_{ss}, Enable 1 and Enable 2 pins are all connected to a 5 V supply and the GND pins connected to ground. Fig. 6 shows the image of the soldered L293d motor driver IC on a veroboard.

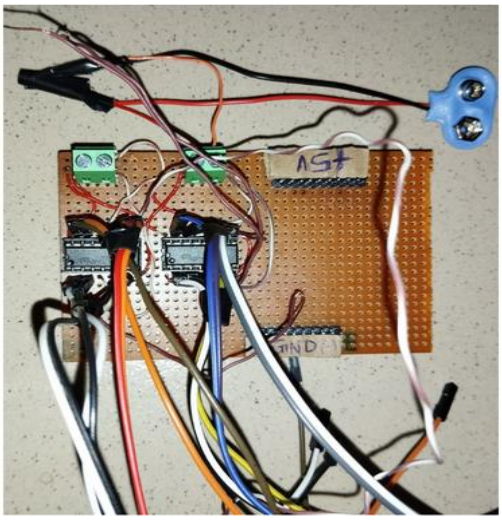

Fig. 6. *Image of the soldered L293D motor ICs on a veroboard.*

(b) Installing the Robot's Chassis

A robot chassis (shown in Fig. 7) is the physical frame or structure of a robot that supports its components and provides stability and mobility. It can be made of various materials, such as metal, plastic, or cardboard, depending on the design and purpose of the robot. The chassis usually includes the drive system, such as wheels, tracks, or legs that allows the robot to move on different terrains. The chassis also provides mounting points for other parts, such as sensors, actuators, or controllers, which enable the robot to perform its tasks.

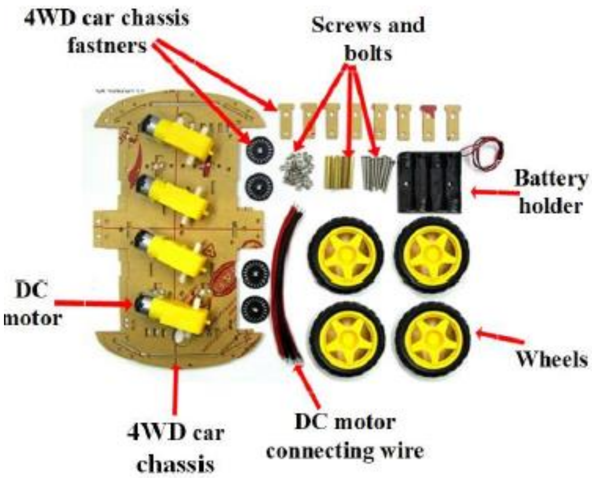

Fig. 7. *Robot chassis components.*

The motors are attached to the chassis using screws, bolts, or other fasteners as shown in Fig. 8. The motors are aligned with the drive system and have enough clearance from the chassis. The motor terminals are also soldered with wires and fed through the chassis holes or slots. The Arduino Mega pro CH340 is mounted to the chassis using standoffs, screws, or other methods. If a metal chassis is used, the microcontroller is prevented from touching the metal chassis or other components by using cardboard, tape, or other insulators. The motor wires are also to the l293d motor driver IC. Thereafter, the power source, such as a battery pack or a portable charger is then connected to the Arduino Mega pro CH340 microcontroller and the motor driver IC. A switch is also connected to turn on and off the power. The power source is secured to the chassis using tape, or other methods. Also, the Infrared sensors, ultrasonic sensors and other necessary components such as LEDs are then mounted on the chassis and connected to the microcontroller. Finally, the robot's chassis was tested after uploading the code to the controller and running the robot. Also the wiring, connections, and the components were checked for any errors or malfunctions.

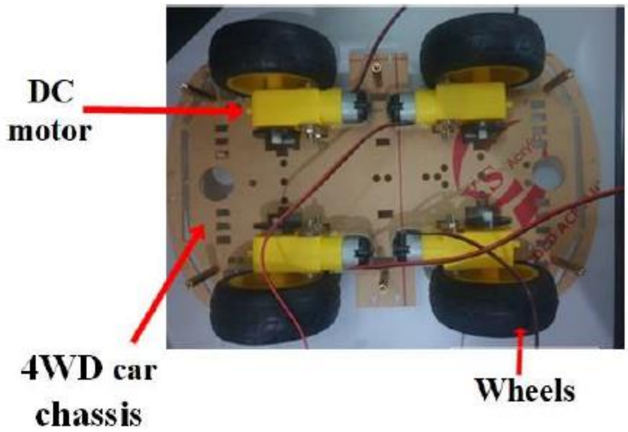

Fig. 8. *Wheels and motors attached to the chassis.*

(c) Programming the Robot

The programming of the robot is done using an Arduino Mega Pro CH340 microcontroller board, which is a compact version of the Arduino Mega 2560 board. The board is connected to the computer via a micro USB cable and a CH340 USB-UART adapter module, which enables the communication between the board and the Arduino IDE software. The Arduino IDE software is used to write the code for the robot's logic, control, and communication using the Arduino programming language, which is based on C/C++. The code is then uploaded to the board's memory using the Upload button in the IDE. After uploading the code, the USB cable is disconnected and the board then connected to the robot's chassis, sensors, actuators, and power source using wires, breadboards, or veroboards. The robot's functionality is then tested by running the code on the board and observing the robot's behavior. The code is also modified and improved as needed to achieve the desired performance and functionality of the robot.

C. Collision Avoidance and Edge Detection Algorithm

The flowchart in Fig. 9 provides a visual representation of the process or workflow of the robot showing how it makes its decision based on the input received from the infrared and ultrasonic sensors and what it is programmed to do after receiving the input. First, when the robot is switched on, it initializes all the sensors connected to the Arduino Mega pro CH340

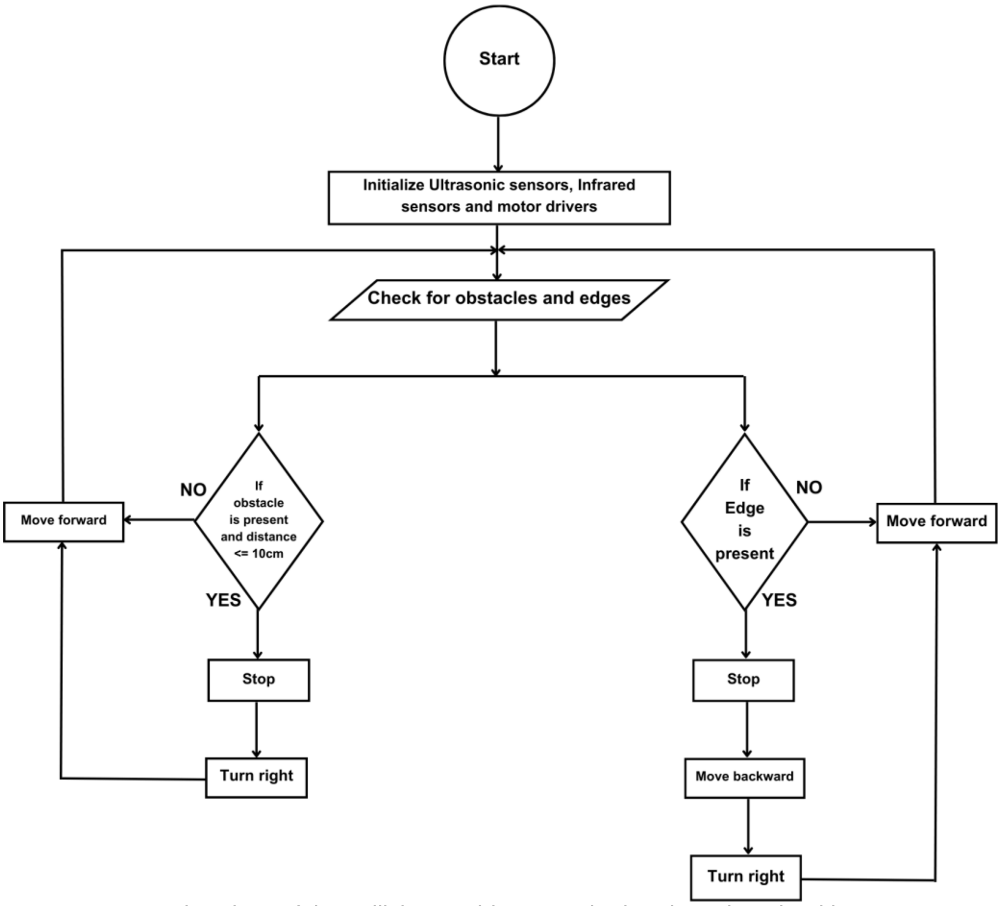

Fig. 9. *Flowchart of the collision avoidance and edge detection algorithm.*

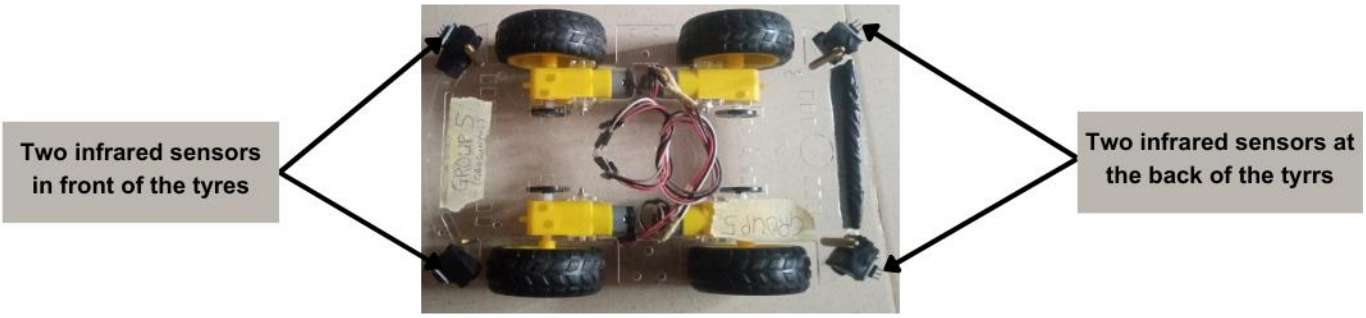

Fig. 10. *Top view showing the mounted infrared sensors on the chassis.*

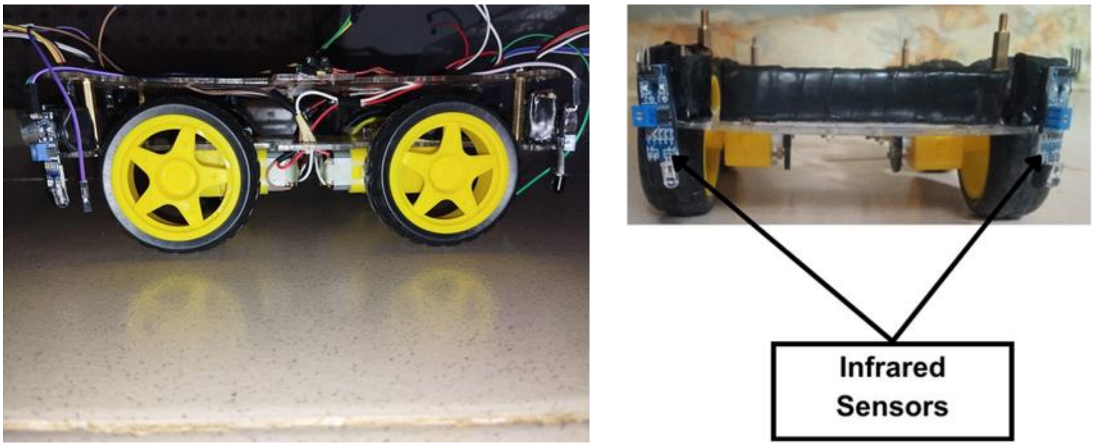

Fig. 11. *Side and front view of the mounted infrared sensors.*

whereby the ultrasonic sensor checks for objects in its path and the infrared sensors detect for edges while the robot is in motion. If an obstacle is present, the robot stops and then changes its direction to either left, right or back based on the code which is uploaded to the microcontroller. Also if any edge is detected, the robot stops and moves backward before changing its direction to either left or right to prevent it from falling off. These algorithms work together, allowing the robot to navigate safely and intelligently.

IV. Results And Discussions

To assess the effectiveness of the robot, rigorous testing was conducted. The robot underwent scenarios involving various obstacles, both static and dynamic. Its ability to detect and avoid obstacles accurately was also measured. Additionally, edge detection was evaluated by simulating different edges and drop-offs. The robot's responsiveness, reliability, and overall performance were scrutinized, ensuring that it meets the intended objectives.

A. Edge Detection Results

This robot also utilizes four infrared sensors in which two of them are placed in front of the tyres at the front of the chassis as shown in Fig. 10 and the remaining two placed at the rear of the tyres at the back of the chassis all facing downwards as shown in Fig. 11. The infrared sensor works almost the same way as the ultrasonic sensor whereby it emits infrared waves and receives the reflected waves to detect distance of obstacles. Using the Arduino Mega and tuning potentiometer located on the IR sensor to control its sensitivity, the IR sensor is programmed in such a way that if it emits wave to a certain distance and it doesn't hit an obstacle that means the robot is probably at an edge and it moves backwards and then take another direction.

This robot incorporates four infrared sensors dedicated to edge detection, strategically positioned on its chassis. In the front, two of these sensors are placed in proximity to the tires, while the remaining two are situated at the back, all facing downward. These infrared sensors operate on a principle similar to that of ultrasonic sensors, emitting infrared waves and measuring the reflected waves to determine the distance to obstacles. The Arduino Mega, in collaboration with tuning of potentiometers on each IR sensor, facilitates the control of IR sensors sensitivity.

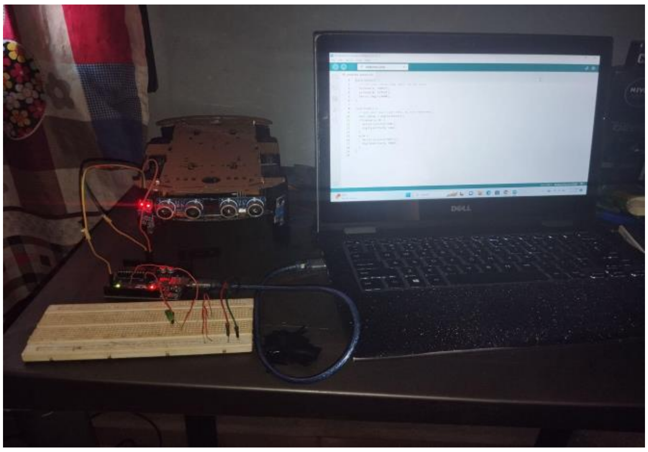

Fig. 12. *Testing the infrared sensors for edge detection.*

The IR sensors are programmed as shown in Fig. 12 such that they emit waves to a predetermined distance; if no obstacle is detected within that range, the robot interprets this as being at an edge. Consequently, it initiates backward movement and adjusts its direction, effectively navigating away from the perceived edge. This sophisticated integration of infrared sensors and the Arduino Mega enhances the robot's capability to discern and respond to the presence of edges in its environment, contributing to a more nuanced and adaptive navigation system.

B. Testing the Ultrasonic Sensor for Obstacle Detection

The effectiveness of the developed obstacle avoidance robot heavily relies on the accuracy and reliability of the ultrasonic sensors. Here, the rigorous testing process conducted was outlined to assess the performance of these sensors in detecting obstacles and preventing collisions. The first test involved placing obstacles of various sizes and shapes (cubes, cylinders, irregular objects) at different distances (close, medium, far) from the robot's ultrasonic sensors. The distance readings reported by the sensors and compared them to the actual distances measured with a tape measure. Table III shows the comparison between the ultrasonic sensors' measured reading to the actual reading. The data obtained was observed for consistency, repeatability, and any potential outliers or sources of error.

TABLE III. Comparison of ultrasonic sensor's measured reading to the actual reading.

Ultrasonic sensor reading (cm)	Actual reading (cm)
0	1
1	1
5	5
7	7.2
10	10
13	13.1
16	16.3

C. Testing the Robot's Obstacle Avoidance and Edge Detection Algorithm

A controlled testing environment is prepared as shown in Fig. 13, incorporating various obstacles, barriers, and edges arranged on a tabletop. It was also ensured that the robot's fully charged and the connection to the power source is tight. The next step involved uploading the obstacle avoidance and edge detection algorithm to the microcontroller using an Arduino IDE so that the robot can be put in operation. The robot's behavior was monitored during testing, noting its reactions when approaching obstacles or edges.

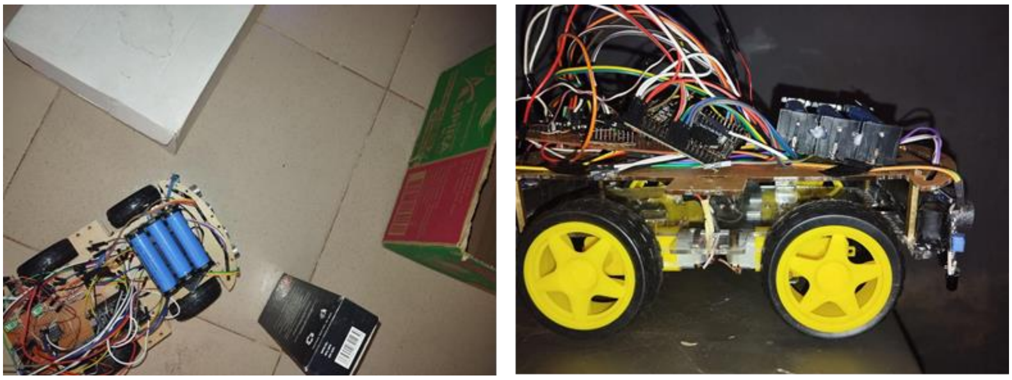

Fig. 13. *Testing the robot for obstacle avoidance and edge detection.*

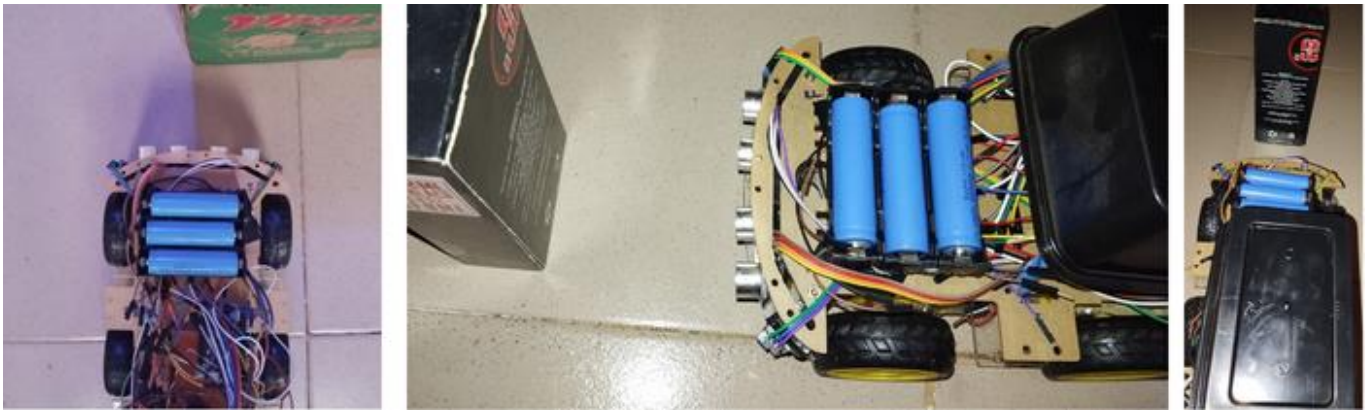

Fig. 14. *The robot successfully stopped as soon as it detects an obstacle.*

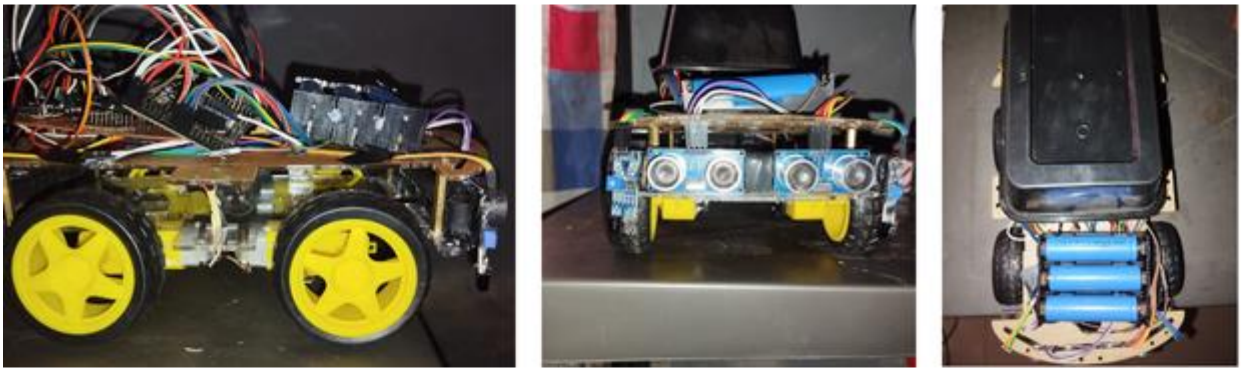

Fig. 15. *The robot successfully stopped and re-navigate as soon as it detects an edge.*

The robot stops whenever it detects an obstacle or an edge as shown in Figs. 14 and 15 respectively before deciding on its next direction to avoid hitting obstacles or falling off an edge. Multiple tests were performed with different objects as obstacles and also placed on different table heights to ensure the consistency and reliability of the robot's performance. Necessary adjustments were also made on the code to improve the robot's algorithm based on the feedback from the testing.

V. CONCLUSIONAND FUTURE STUDIES

This study presents the development and evaluation of an indoor navigation aid for the visually impaired, utilizing an ultrasonic glove system. The system has demonstrated significant potential in assisting visually impaired individuals by providing timely and effective feedback for obstacle detection. Through user feedback and performance evaluations, the glove's simplicity and forward-facing sensor orientation have proven to be effective in real-world scenarios, enabling users to navigate their surroundings with reduced dependence on external assistance. The system's reliance on a single vibration mode and its challenges in crowded environments highlight areas where further refinement is necessary. Specifically, users reported difficulty in interpreting overwhelming feedback in environments with multiple obstacles, which suggests a need for adaptive sensitivity control. Despite these limitations, the glove has shown promise as an innovative solution for indoor navigation.

REFERENCES

[1] D. Budakova, G. Pavlova, R. Trifonov, and I. Chavdarov. “Obstacle avoidance algorithms for mobile robots,” In Proceedings of the 20th International Conference on Computer Systems and Technologies, pp. 78-83, 2019.

[2] L. De Filippis, G. Guglieri, and F. Quagliotti. “Path planning strategies for UAVS in 3D environments,” Journal of Intelligent & Robotic Systems, vol. 65, pp. 247-264, 2012.

[3] P.Y. Leong, and N.S. Ahmad. “LiDar-based Obstacle Avoidance with Autonomous Vehicles: A Comprehensive Review.” IEEE Access, vol. 12, pp. 164248-164261, 2024.

[4] C. Han, W. Wu, X. Luo, and J. Li. “Visual navigation and obstacle avoidance control for agricultural robots via LiDAR and camera.” Remote Sensing, vol. 15, no. 22, p.5402, 2023.

[5] L. Liu, X. Wang, X. Yang, H. Liu, J. Li, and P. Wang. “Path planning techniques for mobile robots: Review and prospect.” Expert Systems with Applications, vol. 227, p.120254, 2023.

[6] N.K. Chandran, M.T.H. Sultan, A. Łukaszewicz, F.S. Shahar, A. Holovatyy, and W. Giernacki. “Review on type of sensors and detection method of anti-collision system of unmanned aerial vehicle,” Sensors, vol. 23, no. 15, p.6810, 2023.

[7] M.R. Watts, C. Poulton, M. Byrd, and G. Smolka. “Lidar on a chip enters the fast lane: sensors for self-driving cars and robots will be tiny, reliable, and affordable.” IEEE Spectrum, vol. 60, no. 9, pp.38-43, 2023.

[8] D. Kragic, M. Björkman, H.I. Christensen, and J.O. Eklundh. “Vision for robotic object manipulation in domestic settings,” Robotics and autonomous Systems, vol. 52, no. 1, pp. 85-100, 2005.

[9] U. Ulusoy, O. Eren, and A. Demirhan. “Development of an obstacle avoiding autonomous vehicle by using stereo depth estimation and artificial intelligence based semantic segmentation.” Engineering Applications of Artificial Intelligence, vol. 126, p.106808, 2023.

[10] Y. Bai, B. Zhang, N. Xu, J. Zhou, J. Shi, and Z. Diao. “Vision-based navigation and guidance for agricultural autonomous vehicles and robots: A review.” Computers and Electronics in Agriculture, vol. 205, p.107584, 2023.

[11] Uribe (2024). Urban robot-JPL robotics-NASA. Onle. Available from: https://www-robotics.jpl.nasa.gov/what-we-do/research-tasks/urbie-urban-robot/ [Accessed: 30/12/2024].

[12] R. Vairavan, S.A. Kumar, L.S. Ashiff, and C.G. Jose. “Obstacle avoidance robotic vehicle using ultrasonic sensor, Arduino controller.” International Research Journal of Engineering and Technology, 5(02), 2140-2143, 2018.

[13] Z.F. Li, J.T. Li, X.F. Li, Y.J. Yang, J. Xiao, and B.W. Xu. “Intelligent tracking obstacle avoidance wheel robot based on arduino,” Procedia Computer Science, 166, 274-278, 2020.

[14] J.N. Yasin, S.A.S. Mohamed, M.H. Haghbayan, J. Heikkonen, H. Tenhunen, and J. Plosila. “Low-cost ultrasonic based object detection and collision avoidance method for autonomous robots,” International Journal of Information Technology, VoL. 13, pp.97–107, 2021.

[15] A.I. Neloya V. Baruab, M. Dasc, P. Barua, S.U. Rahate, and A. Pathakf. An intelligent obstacle and edge recognition system using Bug algorithm,” American Scientific Research Journal for Engineering, Technology, and Sciences, vol. 64, no. 1, pp. 133-143, 2020.

[16] A. Pandey, D. Parhi, and S. Pandey. “Mobile Robot navigation and obstacle avoidance techniques: A review,” International Robotics & Automation Journal, vol. 2, no. 3, pp. 1-12, 2017.

[17] J. Borenstein, and Y. Koren. “Real-time obstacle avoidance for fast mobile robots,” IEEE Transactions on Systems, Man, and Cybernetics, vol. 19, no. 5, pp. 1179-1187, 1989.