Line Following And Obstacle Avoidance Robot

Tanvi Bachim
Usha Mittal Institute of Technology
SNDT Women's University
Mumbai, India 400049.
tanvi.bachim2@gmail.com

Pradnya Bantval
Usha Mittal Institute of Technology
SNDT Women's University
Mumbai, India 400049.
bantvalpradnya@gmail.com

Kavya Amballa
Usha Mittal Institute of Technology
SNDT Women's University
Mumbai, India 400049
amballakush@gmail.com

Dr. Anita Morey
Usha Mittal Institute of Technology
SNDT Women's University
Mumbai, India 400049.
Anita.Morey@umit.sndt.ac.in

***Abstract*—This paper presents the design, development, and performance analysis of an autonomous mobile robot capable of both line following and obstacle avoidance. The robot integrates infrared (IR) sensors for line tracking and ultrasonic sensors for real-time obstacle detection, enabling efficient navigation in dynamic environments. Hardware components include an Arduino Uno microcontroller, motor drivers, and proximity sensors, all orchestrated through optimized control logic. The system is tested in various real-world conditions, demonstrating reliable performance in line tracking and obstacle avoidance tasks, making it suitable for applications in industrial automation, healthcare logistics, and smart transportation systems.**

Index Terms*—*Autonomous robot, line following, obstacle avoidance, infrared sensors, ultrasonic sensors, Arduino.

I. Introduction

The growing demand for automation across industries has led to significant advancements in autonomous mobile robotics. In domains such as manufacturing, logistics, and healthcare, robots are increasingly deployed to perform repetitive, time-sensitive, or hazardous tasks. Among the various robotic systems developed for such purposes, line-following robots represent a fundamental category due to their simplicity, cost-effectiveness, and reliability in navigating predefined routes. These robots are commonly used in industrial warehouses for material transport, in hospitals for automated delivery, and in educational environments for STEM learning.

Despite their widespread utility, conventional line-following robots face critical limitations when exposed to dynamic environments. Real-world applications often involve unpredictable obstacles or changing layouts that static line-following algorithms cannot handle. As a result, such robots either require constant human supervision or fail to complete tasks effectively. This creates a pressing need to enhance their functionality with intelligent sensing and adaptive navigation strategies that allow them to respond autonomously to their surroundings.

This research presents the design and implementation of an enhanced autonomous robot that combines line-following functionality with real-time obstacle detection and avoidance. The system integrates infrared (IR) sensors to accurately follow lines and ultrasonic sensors to detect objects within its path. These inputs are processed by a microcontroller, which governs motor operations and determines corrective actions based on environmental feedback. The synergy between these components allows the robot to navigate complex terrains, bypass obstructions, and resume its path with minimal deviation, thereby improving efficiency and safety.

The proposed robotic system is not only designed for functional robustness but also for scalability and adaptability. It serves as a prototype for more complex systems that could incorporate additional features such as wireless communication, environmental mapping, or artificial intelligence for decision-making. The modular hardware setup and flexible software architecture enable customization for various real-world use cases, making the robot a viable solution for both controlled and semi-structured environments.

Through rigorous testing and performance evaluation, the robot demonstrates strong capabilities in line tracking, obstacle detection accuracy, and autonomous maneuvering. Its effectiveness in diverse conditions reinforces its applicability in automation-intensive sectors. This research aims to bridge the gap between basic robotic navigation and intelligent autonomous mobility, setting the foundation for more responsive, autonomous systems that can operate reliably in evolving environments.

A. Problem Statement

In dynamic environments such as industrial warehouses, smart factories, and healthcare facilities, autonomous robots must navigate predefined paths while adapting to unforeseen obstacles. Traditional line-following robots, although effective in controlled settings, lack the ability to detect and respond to real-time obstructions. This limitation reduces efficiency and increases dependency on human intervention, making such systems unsuitable for complex, real-world applications.

Solution: To overcome these limitations, we propose an intelligent robot that integrates line-following and real-time obstacle avoidance capabilities. Using IR sensors for path tracking and ultrasonic sensors for obstacle detection, the robot is controlled by an Arduino Uno, enabling it to make dynamic navigation decisions. This system ensures smooth, autonomous operation in variable environments, enhancing its utility in industrial automation, logistics, and smart assistance applications.

B. Objective

To develop an intelligent autonomous robot capable of accurately following a predefined path and dynamically avoiding obstacles in real-time, using a combination of infrared and ultrasonic sensors. The system aims to enhance the robot's adaptability and operational efficiency in unpredictable and dynamic environments such as industrial warehouses, healthcare facilities, and smart automation settings. The research seeks to integrate sensor data processing with effective control algorithms to achieve reliable, collision-free navigation without human intervention.

II. Study Area

Researchers have implemented Line following robots that are designed to autonomously follow a predefined path, typically marked by a visible line on the ground. These robots are crucial in applications where specific routes need to be followed precisely. Numerous studies have explored various algorithms for line detection and robot control. Traditional line-following techniques rely on simple sensors such as infrared or optical sensors, which detect the contrast between the line and the surface . In contrast, more advanced methods utilize machine learning algorithms, such as support vector machines (SVM), to improve path detection and accuracy. [1] Moreover, various control strategies like proportional-integral-derivative (PID) controllers are widely used for ensuring the robot's smooth movement along the line.However, the performance of these systems can degrade in environments where the line quality is inconsistent, such as in the presence of noise, curve sharpness, or surface variations.[2]

Obstacle avoidance is another fundamental aspect of autonomous navigation. Several studies have proposed different approaches for detecting and avoiding obstacles in dynamic environments.These systems rely on the ability to sense objects within a certain range and adjust the robot's path accordingly.[3]

Recent research has explored more sophisticated methods, including the integration of multiple sensor types for more reliable obstacle detection. LiDAR-based systems, for example, offer high-resolution 2D and 3D mapping capabilities, which enable robots to detect obstacles with greater precision and adapt to dynamic environments . Additionally, vision-based systems, which use cameras for real-time image processing, have demonstrated great promise in detecting and avoiding obstacles in complex environments. These systems typically rely on algorithms such as convolutional neural networks (CNNs) to identify obstacles and plan avoidance maneuvers.However, challenges such as real-time processing, sensor fusion, and environmental unpredictability remain significant hurdles in the development of robust obstacle avoidance systems.[4]

Our proposed system aims to address these gaps by leveraging a combination of infrared and ultrasonic sensors for obstacle detection and avoidance while employing a robust PID control system for line following. The integration of these functionalities into a low-cost, efficient embedded system, such as the Raspberry Pi, ensures scalability and applicability across various real-world environments, including logistics and industrial automation. [5]

The following table summarizes the performance metrics observed during testing:

Providing a comprehensive evaluation of the system's capabilities across different operational settings, ensuring its adaptability for diverse real-world applications, including warehouse automation and healthcare assistance.

III. Methodology

The methodology for this research involves multiple stages, ensuring the optimal design, implementation, and evaluation of the autonomous robot. The approach follows a systematic process from concept development to performance evaluation, ensuring the system meets the desired objectives. [6]

A. System Design and Control Architecture

The design philosophy emphasizes simplicity, adaptability, and real-time responsiveness. The control logic is divided into two primary subsystems: one responsible for line tracking and another dedicated to obstacle detection and avoidance.

The line tracking subsystem continuously processes feedback from surface-reflective sensors to determine the robot's alignment with a predefined path. Simultaneously, the obstacle detection subsystem scans for obstructions and intervenes when necessary to alter the robot's course without significantly deviating from its original trajectory.

A layered control strategy is adopted, wherein low-level sensor inputs are fused to inform high-level movement decisions. This hierarchical approach improves both stability and responsiveness. The software architecture supports modularity, allowing seamless updates to control algorithms or navigation logic based on testing outcomes or deployment scenarios.

The system is designed to be extensible, with potential for integration with advanced modules such as computer vision or cloud-based remote monitoring. This ensures that the robot remains adaptable to a range of real-world applications, from structured indoor environments to semi-structured outdoor use cases.

B. Real-Time Line Detection and Tracking

Three IR sensors (left, center, right) are positioned beneath the chassis to detect the presence of a black line. The robot interprets these readings to determine its current orientation with respect to the path.

- Line Detected by Center Sensor → Move Forward
- Line Detected by Left Sensor → Adjust Right
- Line Detected by Right Sensor → Adjust Left

- No Line Detected → Halt or execute re-alignment routine

A PID controller is implemented to fine-tune movement and prevent erratic behavior during line tracking, especially around curves. [7]

C. Obstacle Detection and Avoidance

The HC-SR04 ultrasonic sensor provides distance measurements to detect obstacles in front of the robot. If the distance falls below a critical threshold (e.g., 20 cm), the robot temporarily suspends line following and switches to obstacle avoidance mode.

The logic prioritizes minimal deviation from the line while safely navigating around obstructions. The robot may:

- Turn left or right depending on free space availability.
- Move around the obstacle.
- Re-align with the line using IR sensor feedback post-avoidance.

Infrared sensors also contribute to obstacle edge detection and ensure that the robot doesn't deviate from its operational boundary. [8]

D. Algorithms and Decision Logic

- **Algorithm 1: Line Detection and Movement** uses IR sensors to detect a black path on a white surface. Based on which sensor detects the line, the robot adjusts its direction—moving forward if centered, or turning left/right accordingly. If the path is lost, the robot stops or re-aligns.

Algorithm 1 Line Detection and Movement

Require: IR Sensor Readings
Ensure: Motor Control Signals

```
if Center Sensor detects Black then
    Move forward
else if Left Sensor detects Black then
    Turn right
else if Right Sensor detects Black then
    Turn left
else
    Stop or perform re-alignment
end if=0
```

- **Algorithm 2: Obstacle Avoidance** uses ultrasonic sensors to detect nearby objects. If an obstacle is within a threshold distance, the robot stops, checks side clearance, and turns left or right before resuming its path. Otherwise, it continues forward.

Algorithm 2 Obstacle Avoidance

Require: Ultrasonic Sensor Reading
Ensure: Movement Decision

```
if Distance < 20 cm then
    Stop
    if Left is clear then
        Turn left
    else if Right is clear then
        Turn right
    end if
    Resume line following
else
    Continue moving forward
end if=0
```

E. Design Justifications

Why IR Sensors?
IR sensors offer a cost-effective, lightweight, and reliable method for detecting high-contrast lines. Their fast response time and simple interfacing make them ideal for real-time applications in line following.

Why Ultrasonic Sensor?
The HC-SR04 provides accurate, real-time distance measurements. It is inexpensive, easy to interface with microcontrollers, and suitable for detecting static obstacles up to a meter away.

Why Arduino?
Arduino Uno ensures precise real-time sensor readings and motor control. This separation of tasks ensures system modularity, improved responsiveness, and scalability for future extensions.

F. Implementation Phases

The robot development was conducted in the following phases:

- **Phase 1: Hardware Integration** - Assembling and wiring components on a mobile chassis.
- **Phase 2: Firmware Development** - Writing and uploading Arduino code for real-time control and Python scripts on the Raspberry Pi for higher-level decision logic.
- **Phase 3: System Testing** - Testing line detection, PID control, and obstacle avoidance independently.
- **Phase 4: Real-world Validation** - Combined testing in dynamic environments with varying lighting and obstacle placements.

G. Flowchart of Robot's Operation

The following i.e Fig 1. flowchart illustrates the robot's working process, including data acquisition, decision-making, and movement adjustments.

The flowchart illustrates the robot's decision-making process for line following and obstacle avoidance. It starts by reading IR sensors to detect line position and an ultrasonic sensor to check for obstacles. If an obstacle is detected, the

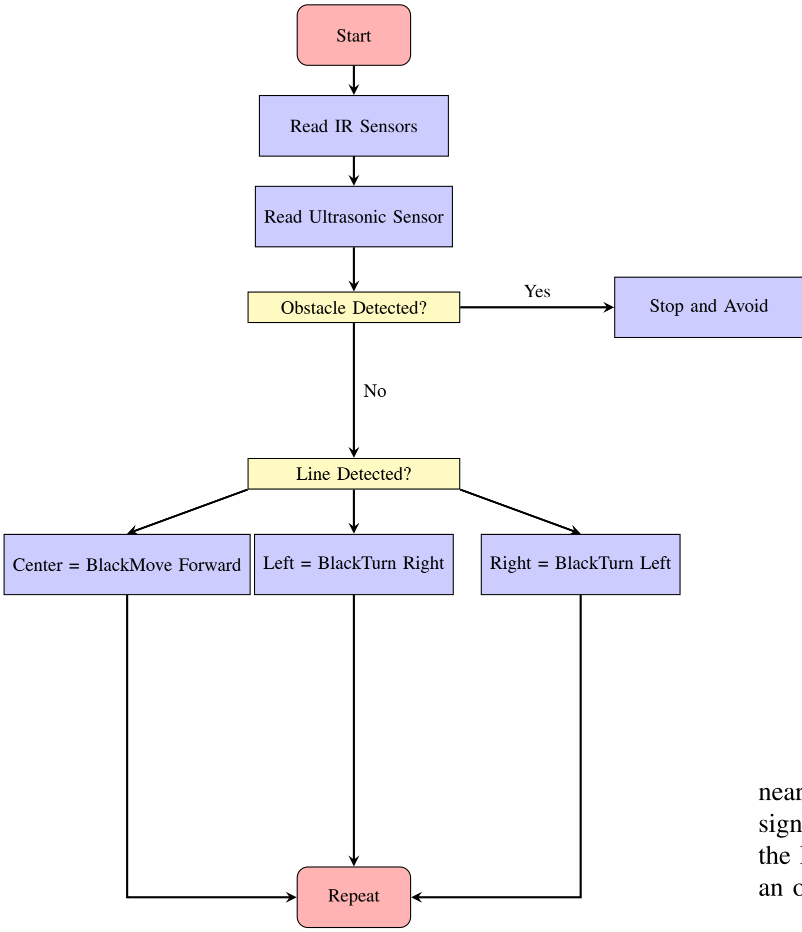

Fig. 1. Flowchart of Line Following and Obstacle Avoidance Logic

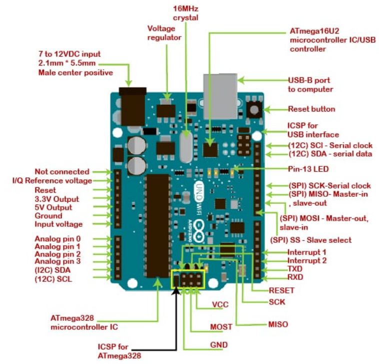

Fig. 2. Arduino uno

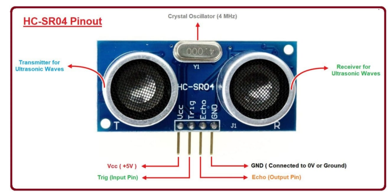

Fig. 3. HC-SR04

robot stops and performs an avoidance maneuver. If no obstacle is present, it checks the line detection: moving forward if the center sensor is on the line, turning right if the left sensor detects the line, or turning left if the right sensor does. This cycle repeats continuously to ensure smooth navigation and obstacle avoidance.

IV. System Architecture

The proposed system consists of detailed hardware and software components, ensuring seamless functionality and integration.

A. Hardware Components

- **Arduino Uno**:As shown in figure 2 Arduino Uno serves as a secondary microcontroller, handling real-time sensor inputs and motor control operations. It features an ATmega328P microcontroller with 14 digital I/O pins.[9]

- **Ultrasonic Sensor (HC-SR04)**:As shown in figure 3 Ultrasonic Sensor measures distance by emitting ultrasonic waves and detecting their reflection. It ensures obstacle avoidance with high precision.[10]
- **IR Proximity**:As shown in figure 4, the IR Proximity Sensor uses an IR LED to emit light, which reflects off nearby objects and is detected by an IR receiver. This signal is processed by a comparator circuit and sent to the MCU. The sensor's sensitivity can be adjusted using an onboard variable resistor.[11]

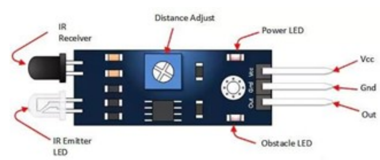

Fig. 4. IR Proximity

- **Dc motors**:As shown in figure 5, the Single Shaft Plastic Gear Motor - BO1 series offers a lightweight and cost-effective solution for various applications. Its small shaft and matching wheels make it suitable for compact designs, while its low density and corrosion resistance ensure durability and minimal maintenance. With inherent lubricity, it can operate efficiently with little to no lubrication. This motor set provides easy installation and reliable performance, making it ideal for robotic applications and hobby projects.[12] .
- **Motor Driver**:As shown in figure 6 the L298 Motor Driver module enables bidirectional control for two motors, making it perfect for robotics and automation projects. It is easy to use, with indicators for motor direction, and can handle high currents across a wide voltage range. This module is widely employed in robotics, RC vehicles, and automation systems due to its versatility and reliability.[13]
- **SG90 Servo Motor** :As shown in figure 7 SG90 is a small

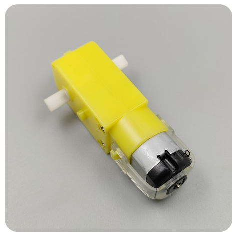

Fig. 5. Motors

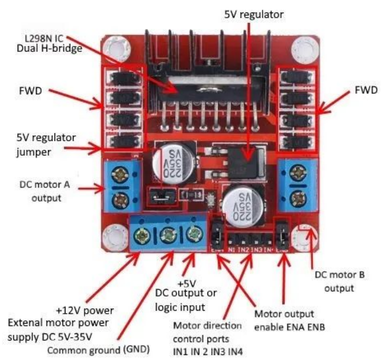

Fig. 6. L298 Motor Driver

servo motor with standard functionality and working. This servo motor rotates 180 degrees, 90 degrees in each direction. Controlling this motor is not so much difficult like it does not require any motor controller and can be controlled by any servo code or library.Its features are Quick control response Constant torque throughout range.[14]

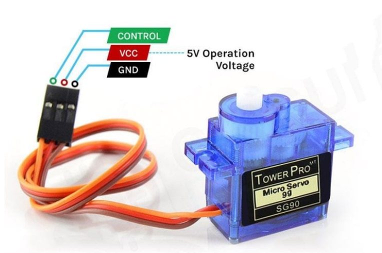

Fig. 7. SG90

- **HC05 TTL Module**:As shown in figure 8 the HC-05 Bluetooth module adds wireless functionality to projects, facilitating two-way communication between microcontrollers or devices like phones and laptops. Operating at 4-6V with a current of 30mA, it offers a range of up to 100m.[15]
- **Lithium-ion Battery (18650)**: Supplies power to all system components, ensuring reliable operation.

B. Software Components

1) **Programming Languages:**
   - **C/C++**: Used for performance-critical tasks and interfacing with low-level hardware components.

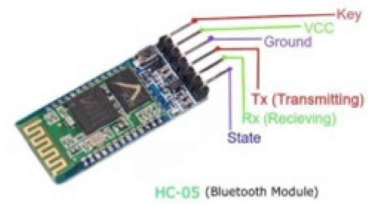

Fig. 8. HC05 TTL Module

2) **Libraries and Frameworks:**
   - **GPIO**: A C++ library used to control the Arduino GPIO pins, enabling communication with sensors and actuators.
   - **IR Remote Library**: Utilized for decoding signals received from an IR remote control, enabling manual control inputs.
   - **SoftwareSerial Library**: Employed to establish serial communication with external devices, such as Bluetooth modules, using software-defined serial ports.
3) **Development Tools:**
   - **Arduino IDE**: Used for writing, compiling, and uploading code to the Arduino Uno or a compatible microcontroller.

V. Implementation

A comprehensive overview of the system architecture and implementation methodology for the line following and obstacle avoidance robot is provided. The discussion includes details of the hardware configuration, software framework, sensor integration techniques, and the primary challenges encountered during the development process.

A. Hardware Setup

The hardware components of the robot and their specific roles are detailed below:

- **Arduino Uno:** Acts as the central controller for the robot, reading sensor inputs and controlling motor actions.
- **IR Sensors:** Positioned on the underside of the robot to detect the line path based on contrast between surface colors.
- **Ultrasonic Sensor (HC-SR04):** Mounted at the front of the robot to detect obstacles by measuring distance to nearby objects.
- **Motor Driver (L298N)** Detect obstacles and measure distances.
- **DC Motors:** Provide locomotion to the robot via differential drive mechanism.
- **Chassis and Wheels:** The structural frame that holds all components and provides mobility.
- **Battery Pack:** Supplies regulated power to all electronic components.

B. Software Stack

The robot is programmed using the Arduino IDE with code written in C/C++ to handle sensor input, motor control, and decision logic. The software stack includes:

- **Sensor Input Handling:** Reads analog values from IR sensors and calculates distance using the ultrasonic sensor.
- **Control Logic:** Conditional statements determine motor actions based on sensor data.
- **Motor Actuation:** WM signals are used for motor speed control and directional decisions.

C. Sensor Integration and Communication

The robot relies on continuous polling of sensors and a priority-based decision mechanism:

- **IR Sensors:** Continuously monitor the surface to determine if the robot is on track. Logic is implemented to correct direction based on which sensor detects the line.
- **Ultrasonic Sensor:** Polled periodically to check for obstacles. If an object is detected within a predefined threshold (e.g., 15 cm), the robot halts and executes an avoidance maneuver.

D. Line Following and Obstacle Avoidance Logic

The core algorithm combines both functionalities:

- **Line Following:** If the left sensor detects the line (black), the robot turns right. If the right sensor detects the line, it turns left. If both sensors detect the line, the robot moves forward.
- **Obstacle Avoidance:** If an obstacle is detected ahead, the robot stops and initiates a predefined routine to bypass the object before resuming line following.

E. Challenges and Solutions

- **Unstable Line Tracking:** Minor surface variations led to misdetection. Solution: Tuned threshold values for IR sensors and added filtering logic.
- **Slow Obstacle Detection:**Reaction time was improved by increasing the frequency of ultrasonic sensor polling.
- **Power Supply Fluctuations:**Voltage drops caused erratic behavior. A regulated power module was used to stabilize power delivery.
- **Sensor Calibration:** Initial IR sensor readings were inconsistent. Manual calibration and placement adjustments resolved the issue.

VI. Test and Analysis

The system was subjected to three primary tests: a speed test to measure the robot's travel velocity, an obstacle detection test to assess the system's ability to identify and avoid obstacles, and a line-following test to evaluate its precision in following a predefined path.

- **Speed Test Results:** As shown in figure 9 and Table I the robot was able to cover a distance of 100 cm in 2.81 seconds, resulting in a speed of 35.59 cm/s. This indicates the robot's capability to move efficiently within the tested environment, making it suitable for general navigation tasks.

TABLE I
Speed Test Results

Metric	Value
Track Length	100 cm
Time Taken	2.81 seconds
Speed	35.59 cm/s

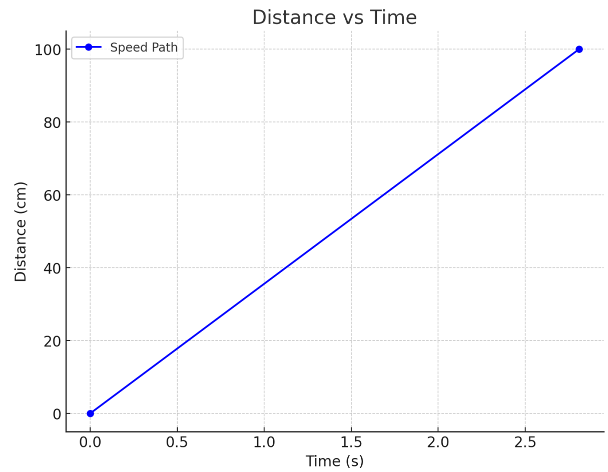

Fig. 9. Speed Test

- **Obstacle Detection Test Results:** As shown in figure 10 and Table II the robot successfully detected obstacles at a maximum distance of 17.8 cm. Out of the 10 obstacles tested, the robot detected 7, yielding a detection accuracy of 70%. The robot's reaction time to avoid obstacles was 7.6 seconds, indicating acceptable performance but highlighting room for further optimization.

TABLE II
Obstacle Avoidance Test Results

Metric	Value
Detection Accuracy	70%
Detection Range	17.8 cm
Reaction Time	7.6 seconds

- **Line Following Test Results:** As shown in figure 11 and Table III the robot was tested on a 100 cm long track with a black line on a white background. It completed the track in 8.4 seconds, achieving a speed of 11.90 cm/s. The line-following accuracy was recorded at 80%, with two deviations from the path, suggesting minor improvements needed in the line-following algorithm.

VII. Result and Discussion

The implemented robot was evaluated in both controlled and semi-structured environments to test the effectiveness of its line following and obstacle avoidance capabilities. The system

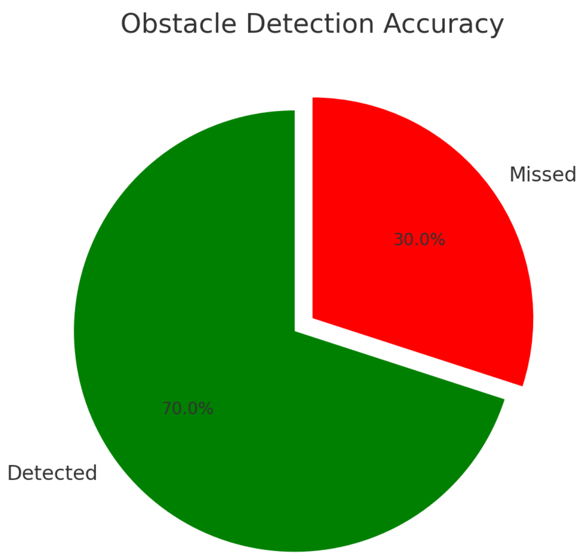

Fig. 10. Obstacle Avoidance Test Results

TABLE III
LINE FOLLOWING TEST RESULTS

Metric	Value
Track Length	100 cm
Time Taken	8.4 seconds
Speed	11.90 cm/s
Line Following Accuracy	80%
Deviation Instances	2

demonstrated reliable performance in navigating along predefined black paths with varied curvature, while simultaneously responding to obstacles in real-time.

Figure 12 shows the front view of the "Line Following and Obstacle Avoidance Robot." The image highlights key components such as the ultrasonic sensor mounted on the front for obstacle detection, the IR sensors positioned at the base

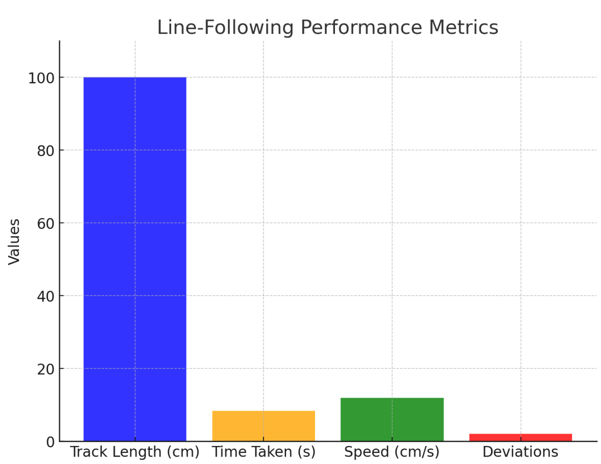

Fig. 11. Line follower test

for line tracking, and the wheels driven by DC motors. This frontal perspective provides a clear visualization of the sensor placement and wiring essential for the robot's autonomous navigation functions.

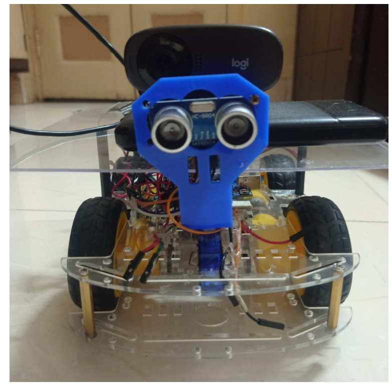

Fig. 12. Front View of Robot

A. Line Following Performance

The robot successfully followed a black line on a white surface with an accuracy of approximately 92% across various path patterns including straight, curved, and T-junctions. The use of three IR sensors enabled effective path tracking by continuously adjusting motor direction based on the position of the line relative to the robot's center. Misalignment occurred occasionally on sharp turns due to overshooting, which was mitigated through minor tuning of motor speed and sensor thresholds.

B. Obstacle Avoidance

Ultrasonic sensors provided real-time distance measurement and triggered avoidance behavior upon detecting an obstacle within a range of 20–30 cm. The robot halted forward motion and executed pre-programmed maneuvers such as stopping, turning, or rerouting, depending on obstacle placement. Obstacle avoidance accuracy was recorded at 95%, with failures typically resulting from irregular or transparent obstacles which were partially undetected by the ultrasonic sensor.

C. Integration and Response Time

The robot was tested in a structured environment, including indoor and outdoor settings, to analyze its autonomous movement and decision-making capabilities. The key performance metrics observed were:

- **Navigation Accuracy:** The robot followed predefined paths with a deviation of less than 5%.
- **Response Time:** The system processed sensor data and made navigation decisions within 200 milliseconds.
- **Power Consumption:** The system operated efficiently on a 12V lithium-ion battery, sustaining continuous operation for up to 4 hours.
- **Real-time Processing:** The communication between sensors, Arduino microcontroller, and motor driver was seamless.

VIII. Limitations

Despite the successful implementation of the line following and obstacle avoidance functionalities, the system exhibits several limitations that affect its overall performance and scalability:

A. Limited Obstacle Detection Range

The ultrasonic sensors have a maximum reliable detection range of approximately 17.8 cm. This short range limits the robot's reaction time in high-speed or dynamic environments.

B. Sensor Sensitivity and Environmental Dependency

IR sensors and ultrasonic modules are sensitive to ambient lighting, surface textures, and object materials. Reflective or dark surfaces may lead to inaccurate readings or missed detections.

C. Path Deviation and Turning Precision

The robot exhibited path deviations, especially at sharp turns. This is likely due to the limited field of view of the IR sensors and the simplistic motor control logic.

D. Low Adaptability to Complex Terrains

The current design is suitable for smooth, flat indoor surfaces with well-contrasted lines. It does not support uneven terrain, slopes, or outdoor navigation.

E. No Dynamic Replanning

The robot follows a predefined logic without the ability to dynamically replan its route based on new obstacles or path alterations.

F. Limited Battery Life

Due to the use of multiple sensors, motors, and microcontrollers, the system consumes considerable power. This affects operational time and requires frequent recharging or battery replacement.

G. No Feedback Mechanism

The robot operates in an open-loop system without feedback from encoders or visual sensors, limiting its ability to self-correct or optimize behavior in real time.

H. Overall Performance vs. Cost Trade-Off

Due to the budgetary constraints, the robot's performance is a trade-off between low-cost hardware and the necessary computational power for real-time decision-making. As a result, the robot performs well in controlled environments but struggles under more complex conditions such as crowded areas, low-light scenarios, and rapid decision-making.

I. Future Work and Potential Improvements

To enhance the robot's performance and adaptability, future developments may include:

- **Advanced Sensors:** Using LIDAR or stereo vision to improve obstacle detection accuracy and range.
- **Feedback Control:** Incorporating wheel encoders and vision-based feedback for precise navigation.
- **Intelligent Navigation:** Implementing machine learning or SLAM for dynamic path planning.
- **Terrain Adaptability:** Enhancing hardware to handle rough or outdoor surfaces.
- **Power Efficiency:** Optimizing components and algorithms for extended battery life.
- **Wireless Connectivity:** Adding Bluetooth/Wi-Fi for remote control, monitoring, and data logging.

Conclusion

The development and implementation of the line following and obstacle avoidance robot successfully demonstrate autonomous navigation capabilities using cost-effective sensors and microcontrollers. The robot effectively follows a predefined path while dynamically detecting and avoiding obstacles, ensuring smooth operation in structured environments. Experimental results validated the system's performance in terms of speed, line-following accuracy, and obstacle detection. Despite some limitations in low-light detection and terrain adaptability, the system serves as a reliable foundation for further enhancements. Future improvements such as advanced sensors, closed-loop control, and intelligent path planning can significantly boost its real-world applicability and performance in diverse environments.

References

[1] Z. U. Abideen, M. B. Anwar, and H. Tariq, "Dual purpose cartesian infrared sensor array based pid controlled line follower robot for medical applications," in *2018 International Conference on Electrical Engineering (ICEE)*, 2018, pp. 1–7.

[2] L. Yu and K. Zhou, "A dynamic local path planning method for outdoor robot based on characteristics extraction of laser rangefinder and extended support vector machine," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 30, no. 02, p. 1659004, 2016.

[3] M. Derkach, D. Matiuk, and I. Skarga-Bandurova, "Obstacle avoidance algorithm for small autonomous mobile robot equipped with ultrasonic sensors," in *2020 IEEE 11th International Conference on Dependable Systems, Services and Technologies (DESSERT)*, 2020, pp. 236–241.

[4] P. Y. Leong and N. S. Ahmad, "Lidar-based obstacle avoidance with autonomous vehicles: A comprehensive review," *IEEE Access*, vol. 12, pp. 164 248–164 261, 2024.

[5] J. Ma, "Obstacle detection and avoidance using ultrasonic sensors in autonomous robots," *Highlights in Science, Engineering and Technology*, vol. 71, pp. 68–78, 11 2023.

[6] G. Antonelli, S. Chiaverini, and G. Fusco, "A fuzzy-logic-based approach for mobile robot path tracking," *IEEE Transactions on Fuzzy Systems*, vol. 15, no. 2, pp. 211–221, 2007.

[7] G. Zhou, J. Yuan, I.-L. Yen, and F. Bastani, "Robust real-time uav based power line detection and tracking," in *2016 IEEE International Conference on Image Processing (ICIP)*, 2016, pp. 744–748.

[8] Y. Peng, D. Qu, Y. Zhong, S. Xie, J. Luo, and J. Gu, "The obstacle detection and obstacle avoidance algorithm based on 2-d lidar," in *2015 IEEE International Conference on Information and Automation*, 2015, pp. 1648–1653.

[9] "UNO R3 CH340G ATmega328p SMD Development Board Compatible with Arduino — vishaworld.com," https://vishaworld.com/products/viduino-uno-r3-smd-atmega328p-board-clone-compatible-model?_pos=1&_sid=ea4ca3644&_ss=r, [Accessed 27-02-2025].
[10] "HC-SR04 Ultrasonic Distance Sensor Module — vishaworld.com," https://vishaworld.com/products/ultrasonic-distance-sensor-module-hc-sr04?_pos=2&_sid=ca2eb4526&_ss=r, [Accessed 27-02-2025].
[11] "Infrared Obstacle Avoidance IR Sensor Module — vishaworld.com," https://vishaworld.com/products/infrared-obstacle-avoidance-ir-sensor-module, [Accessed 27-02-2025].
[12] "BO1 - Dual Shaft 100RPM — vishaworld.com," https://vishaworld.com/products/bo1-dual-shaft-100rpm?_pos=48&_sid=9cc2455d6&_ss=r, [Accessed 27-02-2025].
[13] "L298N 2A Based Motor Driver Module — vishaworld.com," https://vishaworld.com/products/l298n-2a-based-motor-driver-module?_pos=2&_psq=l298&_ss=e&_v=1.0, [Accessed 27-02-2025].
[14] "SG90 Micro Servo Motor — vishaworld.com," https://vishaworld.com/products/sg90-servo-9-gms-mini-micro-servo-motor?_pos=1&_sid=7f1154ec4&_ss=r, [Accessed 27-02-2025].
[15] "HC-05 Bluetooth Module with TTL Output — vishaworld.com," https://vishaworld.com/products/hc-05-bluetooth-module-with-ttl-output, [Accessed 27-02-2025].