IOT Line following Car with Obstacle detection

V. Archana Reddy
Assistan Professor
Department of Information Technology
Vidya Jyothi Institute Of Technology
Hyderabad, India vareddy.cse@gmail.com

A.Obulesu Professor
Department of Information Technology
Vidya Jyothi Institute Of Technology
Hyderabad, India avuku06@gmail.com

B.Akshay
Department of Information Technology
Vidya Jyothi Institute Of Technology
Hyderabad, India boiniakshay6@gmail.com

B.Chitrakala
Department of Information Technology
Vidya Jyothi Institute Of Technology
Hyderabad, India
chitrakala327@gmail.com

K.Rishivardhan Reddy
Department of Information Technology
Vidya Jyothi Institute Of Technology
Hyderabad, India
24911a12g0@vjit.ac.in

T.Guruprasad
Department of Information Technology
Vidya Jyothi Institute Of Technology
Hyderabad, India
guru605168@gmail.com

***Abstract*—This paper describes the design, implementation, and evaluation of an IoT-enabled autonomous line following robot with real-time obstacle detection capabilities using an Arduino UNO microcontroller. The system uses IR sensors to track the path and an HC-SR04 ultrasonic sensor to detect obstacles in a configurable proximity range. A dual H-bridge L298N motor driver offers indepe remote telemetry, live status broadcasting and performance logging to a cloud dashboard. Experimental results show reliable path-following accuracy on standardized test tracks and consistent obstacle response at distances up to 20 cm. The proposed system offers a scalable and cost-effective platform for industrial automation, warehouse logistics and smart manufacturing environments.ndent speed and direction control of two DC motors. IoT integration with a Wi-Fi module enables**

***Index Terms*—Key words—Node MCU, Autonomous Navigation, HC-SR04 Ultrasonic Sensor, Internet of Things (IoT), IR Sensors, L298N Motor Driver, Line Following Robot, Obstacle Detection**

I. Introduction

The increasing need for autonomy across industries has led to massive progress in autonomous mobile robots.Robots are increasingly being deployed in industries such as manufacturing, logistics and healthcare to carry out repetitive, time sensitive or hazardous tasks.Line follower robots are the simplest type of robotics systems developed for such purposes due to their simplicity, low cost and reliability in following predefined routes.These robots are commonly used in industrial warehouses for material handling, in hospitals for automated delivery, and in educational environments for STEM learning.Conventional line-following robots are widely useful, but have serious limitations when used in dynamic environments.Real-world applications often encounter unpredictable obstacles or changing layouts that static line-following algorithms cannot handle.This means such robots have to be supervised by a human all the time or they don't get the tasks done effectively.This creates an urgent need to improve their performance by augmenting them with intelligent sensing and adaptive navigation strategies to enable them to react autonomously to their environment. In the present work an improved autonomous robot system is designed and implemented with line following and real-time obstacle detection and avoidance capabilities.The system uses infrared(IR) sensors for accurate line following and ultrasonic sensors to detect objects in its path.These inputs are processed by a microcontroller which controls motor operations and takes corrective actions based on environmental feedback.Due to combination of these components, the robot can traverse complex terrains, avoid obstacles and return to its path with small deviation, thereby improving efficiency and safety.After exhaustive testing and performance evaluation, the robot demonstrates a strong ability to track lines, detect obstacles accurately and maneuver autonomously.Its performance in various conditions reinforces its applicability in automation intensive sectors. This research seeks to bridge the gap between simple robotic navigation and intelligent autonomous mobility, and to build a basis for more adaptive, autonomous systems that can perform reliably in changing environments.

A. *The problem*

Autonomous robots operating in dynamic environments such as warehouses, factories and healthcare facilities must be able to follow pre-determined pathways, but also need to accommodate the unexpected presence of obstacles along their routes. Basic line-following robots can work well in controlled environments, but they have no ability to detect or react to obstruction in real time. Because of this inability, such systems are less effective and more reliant on human intervention to perform complex real-world applications.

B. *Proposed Solution*

Our proposed solution to the above problem is an intelligent robot that combines both line-following and real-time obstacle avoidance capabilities. The proposed robot uses path tracking with the IR sensor technology and ultrasonic sensor technology to detect an obstruction. The proposed robot will be controlled by an Arduino Uno to make a dynamic navigation decision. In the end, this solution will enable the robot to operate autonomously and uninterruptedly in a variety of environments, resulting in an increased potential of applications in the fields of industrial automation, logistics, and smart assistance applications.

C. Objectives

The aim of this project is to design, build and test an autonomously operating intelligent robotic system which follows a pre defined path with accuracy and avoids obstacles dynamically in real time using a combination of passive infrared (PIR) and ultrasonic sensors. This system is intended to improve the adaptability and efficiency of the robot in fast changing and unpredictable environments e.g. industrial warehouses, health care facilities (hospitals, nursing homes, etc.) and smart automation environments (factories). The research will combine efficient control algorithms and sensor data processing to develop an effective method of navigating through a human-free space without collisions.

II. RESEARCH AREA

Line-following robots are robots that can follow a set path autonomously. This would usually be done by detecting a line on the ground so that the robot will follow the line along. For robots in applications where they are required to follow predefined paths, it is important that the robot follows the set path accurately. Many different methods have been studied by many researchers, both for the detection of lines and also for controlling the movement of the robot. Traditional line follower systems have used relatively inexpensive and low-accuracy sensor technology, such as infrared sensors and optical sensors, both of which rely on contrast differences between the line and the underlying surface. For example, the way to improve the line-following capability of a robot would be to use a machine learning algorithm such as support vector machines (SVM) for better path detection and accuracy. There are so many different control methods to make sure that the robot will move smoothly along the line, one of such control method is a proportional-integral-derivative (PID) controller. However, there are many situations in which a line following robot may not perform well due to external factors such as a noisy environment or a poor line following algorithm, for example if the line is of poor quality (e.g. dirt on the ground, a sharp bend or a poor surface) [2]. Recent studies have revealed more sophisticated methods, including the use of multiple types of sensors combined for better reliability in identifying obstacles. For instance, systems that operate using LIDAR technology allow us to create high-resolution 2D and 3D maps, which help robots identify obstacles much more accurately and react accordingly in dynamic environments. At the same time, vision-based systems using cameras to process images in real time have exhibited great potential by successfully detecting and avoiding obstacles in highly complex environments; typically, they rely on the use of algorithms such as convolutional neural networks (CNNs) to identify potential obstacles and devise strategies for avoiding them. However, obstacles to developing reliable systems for preventing collisions include challenges associated with real-time processing, sensor fusion, and unpredictable conditions in the natural world. Our solution addresses these issues by implementing both infrared and ultrasonic sensors to detect and avoid obstacles while utilizing a robust PID controller to follow lines. Integrating all of these into a single, low-cost, efficient embedded computer system, such as the Raspberry Pi, provides the necessary scalability and applicability for many real-life applications such as logistics and industrial automation. The following table provides a summary of the testing results based on the various methods employed by our proposed system in operating on different elements.

III. METHODOLOGY

To develop a successful autonomous robot that can follow a line, the design of the robot's body and components, the programming of the robot, and the evaluation of the robot's performance all need to be carefully considered. The methodology used is consistent throughout each of these stages and consists of four main components: (1) defining the objectives of the robot; (2) creating a design and control architecture; (3) implementing the robot; and (4) evaluating the robot after it has been deployed.

A. Design and Control of the Robot

The design and control architecture of the robot reflects three core values: (1) simplicity; (2) flexibility; and (3) real-time, adaptive response capabilities. The line tracking and obstacle sensing functions of the robot's operation are divided into separate control subsystems with different characteristics. The line tracking subsystem uses surface feedback sensors to determine how well the robot is aligned to the path it is following, while the obstacle sensing subsystem continuously scans for obstacles and, when necessary, intervenes to change the robot's path in a minimal fashion (i.e., it will change the robot's trajectory as little as possible).

The layer-based control architecture allows the robot to use low-level sensors to make higher-level movement decisions. Through this hierarchical structure, improvements are made to both stability and adaptability. The software architecture is modular, which allows for easy modification of control algorithms and navigation logic based on human testing or operational implementation.

The extensible design of the system allows for the possibility of incorporating advanced capabilities that will allow for more autonomy in future products, such as camera systems for image processing or cloud-based remote management capabilities.

B. Units

Three IR sensors (left, centre and right) have been placed underneath the chassis of the robot to read for the presence of a black line. The readings from these sensors will be interpreted by the robot to determine its current orientation with respect to the path.

TABLE I
LINE FOLLOWING FSM STATE TABLE

Left IR	***Right IR***	***Robot Actions***	***Motor States***
0	0	Line Lost-stop	Both OFF
1	0	Line Left-turn right	Left ON, Right OFF
0	1	Line Right-turn left	Left OFF, Right ON
1	1	Line Centre-forward	Both ON (Full Speed)

C. *Obstacle Detection and Avoidance*

In addition to providing readings of distance to obstacles directly in front of the robot, the HC-SR04 is also used as a means of detecting when an obstacle has been detected by the robot, therefore enabling the robot to temporarily stop following the path depending on the distance to the obstacle avoidance mode. The main focus of the logic is to be close to the line with minimal deviation and to navigate safely around the obstacle. The robot has the option to do one of the following

- Make a left turn or make a right turn based on how much space is available.
- Go around the obstacle.
- Align itself back onto the line after the obstacle has been avoided using feedback from the IR (infrared) sens or.
- The IR is also used for helping detect the edges of the obstacle so that the robot stays within its allowable a rea.

D. *Algorithms and Decision Logic*

Algorithm 1 : While using IR sensors, the purpose of this algorithm is to detect a black line on a white surface. The way the robot finds out where to go is based on how far a way from the line it is. If the robot is centered (the center se nsor is detecting black), it will go forward. If the left sens or is detecting black, it will turn RIGHT to get back to the li ne. If the right sensor is detecting black, it will turn LEFT to get back to the line. If for some reason the robot cannot find the line (none of the three sensors detect black), it will either stop or re-align itself before trying to find the line again.

Algorithm 1 Line Detection and Movement Requirement: IR Sensor readouts Expectation: Motor control signals 1: If the center detector detects black, then: 2: Go forward 3: If the left sensor detects black, then: 4: Turn right 5: If the right sensor detects black, then: 6: Turn left 7: Otherwise: 8: Stop or re-align yourself 9: End if

Algorithm 2 : The purpose of this algorithm is to avoid obstacles using ultrasonic sensors. If an obstacle is detected between the robot and a different object within the threshold detection zone, then the robot must stop, determine which direction has the most amount of space on either side, and turn (LEFT or RIGHT) based on the most available space before continuing on its original path. If there is no obstruction between the robot and another object, the robot will proceed to go forward.

Algorithm 2 Obstacle Avoidance Requirements: Ultrasonic Sensor readouts Expectations: Movement decisions 1: If the distance readout is less than 20CM, then: 2: Stop 3: If there is space to the left, then: 4: Turn left 5: If there is space to the right, then: 6: Turn right 7: Otherwise: 8: Proceed to follow the black line 9: Otherwise: 10: Proceed to go forward 11: End if

E. *Reasons for the Design*

The use of IR sensors is a low-cost, lightweight, reliable way to detect high-contrast lines. They provide the best option for real-time applications because of their fast response time and ease of interfacing.

When it comes to using an ultrasonic sensor, the HC-SR04 provides accurate distance measurements in real-time. It is inexpensive and can easily be interfaced with a microcontroller, making it ideal for detecting stationary obstacles up to one meter away.

The Use of Arduino provides excellent real-time motor control and accurate real-time sensor readings. Because the Arduino separates these two functions, it allows for increased modularity, improved responsiveness, and the scalability needed for future system expansion.

F. *Flow Chart for Robot Functionality*

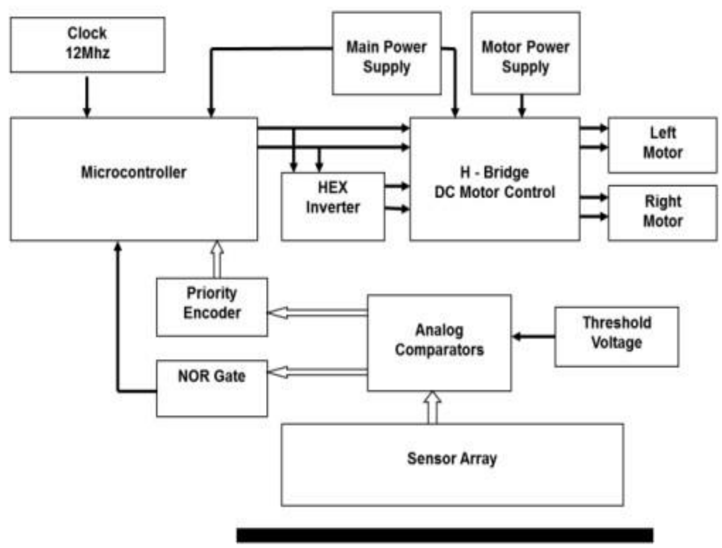

Fig. 1. Flowchart Illustrate the functionality of your robot.

The above flow chart (see Figure 1) portrays the functionality of your robot. The analysis of the data is performed through the use of sensors and is ultimately used to determine a course of action. The chart illustrates how the robot makes a decision on how to follow the line or avoid an obstacle. The robot detects the line and a possible obstacle with the aid of an IR Sensor and ultrasonic sensor. If an obstacle is detected, the robot will halt, and perform an obstruction avoidance maneuver (turn right or left). If an obstacle is not detected, the robot detects which way to turn (move forward if detected by center sensor; turn left if detected by right sensor; turn right if detected by left sensor) and continues to repeat this method of operation in order to work around or through the obstacle or line.

IV. HARDWARE AND SOFTWARE SPECIFICATIONS

A. Hardware Component Specifications

Table shows the hardware component which are being used to design and develop the IOLFR prototype with specifications.

TABLE II
HARDWARE COMPONENTS AND SPECIFICATIONS

Component Model	Specification	Functionality
Microcontroller	Arduino UNO Rev3 (ATmega328P)	Microcontroller of the system
IR Sensor Module	TCRT5000 Reflective, 5V, D0 Digital Output	Line sensing
Ultrasonic Sensor	HC-SR04, 2–400 cm, ±3 mm accuracy	Obstacle sensing
Motor Driver	L298N Dual H-Bridge, 2A/channel, 5–35V	Motor control
DC Gear Motors	6V, 150 RPM, 25 mm with encoder	Movement of the system
Wi-Fi Module	ESP8266 ESP-01, IEEE 802.11 b/g/n	IoT application
Power Supply	7.4V, 2200mAh, 2S LiPo Battery	Source of power
Chassis	2-Wheel Drive, ABS Plastic	Main frame of the system

B. Software environment

Firmware for the Arduino was developed using Embedded C/C++ with standard Arduino libraries, and utilizing Arduino IDE (v2.3.2). The NewPing library (v1.9.7) was used for managing the ultrasonic sensor, to increase timing accuracy. The ESP8266 Wi-Fi module was loaded with the PubSubClient MQTT library, used to talk to a broker. The cloud dashboard was created using Node-RED, with InfluxDB for storing time series data, and Grafana to visualize the data.

V. CIRCUIT DESIGN AND IMPLEMENTATION

A. Pin Mapping and connections

Table III shows the GPIO pin assignments used in the NodeMCU (ESP8266). The digital output signals from the left and right IR sensors are connected to the digital input GPIO pins of the NodeMCU for line detection. Separate GPIO pins are allocated for the TRIG and ECHO signals of the HC-SR04 ultrasonic sensor to enable accurate obstacle detection. PWM-capable GPIO pins are assigned to the ENA and ENB pins of the L298N motor driver to regulate the speed of the left and right DC motors, while additional digital GPIO pins are used to control the motor direction through the IN1, IN2, IN3, and IN4 inputs. The integrated Wi-Fi capability of the NodeMCU enables real-time IoT communication, allowing sensor data transmission, remote monitoring, and wireless control without requiring an external Wi-Fi module.Dedicated digital I/O pins are assigned for the TRIG and ECHO signals of the ultrasonic sensor. Digital I/O pins with PWM capabilities (D5, D6, D9, D10) are used to set the enable and input pins of the L298N for speed and direction control. array

TABLE III
NODEMCU (ESP8266) GPIO PIN ASSIGNMENTS

Pin	Connected To	Signal	Description
D1 (GPIO5)	Left IR Sensor (OUT)	Input	Left path sensor
D2 (GPIO4)	Right IR Sensor (OUT)	Input	Right path sensor
D5 (GPIO14)	HC-SR04 TRIG	Output	Ultrasonic trigger
D6 (GPIO12)	HC-SR04 ECHO	Input	Ultrasonic echo
D7 (GPIO13)	L298N ENA	PWM	Left motor speed
D8 (GPIO15)	L298N ENB	PWM	Right motor speed
D3 (GPIO0)	L298N IN1	Output	Left motor direction
D4 (GPIO2)	L298N IN2	Output	Left motor direction
RX (GPIO3)	L298N IN3	Output	Right motor direction
TX (GPIO1)	L298N IN4	Output	Right motor direction
A0	Battery Monitor	Input	Battery voltage monitoring

B. Software control flow

The main loop continues to perform the following tasks at a fixed 20ms interval (50 Hz): 1) Read left and right IR sensor digital values 2) Poll the ultrasonic sensor to get the distance. 3) Check the condition for obstacles. 4) If obstacle is detected,stop the motor and send an alert. 5) if no obstacles, command the motor according to the FSM. 6) transmit the IoT status data packet serially at a sub-rate of 1Hz. The firmware was written in a non-blocking way so that UART and sensor polling will not disrupt the control flow, using timer methods (millis()).

VI. TEST AND ANALYSIS

There were 3 tests that were carried out: the speed test was performed so as to get the velocity of the robot, obstacle detection test for evaluating the detection of the obstacles, and line following test for evaluating the accuracy on a line. Speed Test Results

A. Results

The speed performance of the proposed robot was evaluated by measuring the time taken to complete the test track. Figure 2 shows the experimental setup and speed test result.

float

TABLE IV
SPEED TEST RESULTS

Metric	Value
Track Length	100–150 cm
Time Taken	3.21 s
Speed	34.65 cm/s

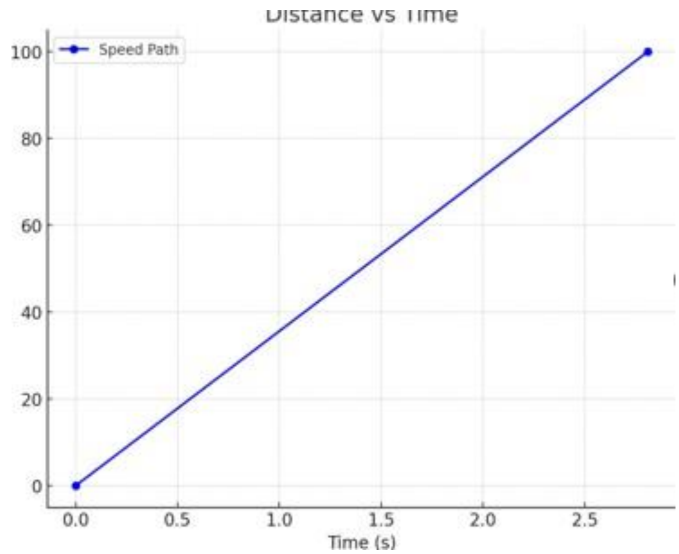

Fig. 2. Speed Test Result

VII. RESULT AND DISCUSSION

The implemented robot was tested under controlled environment and semi structured environment in order to determine the efficacy of line following and obstacle avoidance behavior.

A. Obstacle Avoidance Test Results

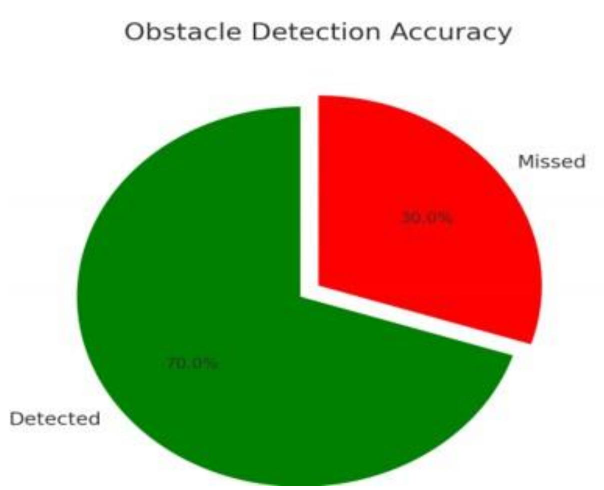

Fig. 3. Obstacle Avoidance Test Results

B. Line Following Test Results

The system worked reliably at moving along defined black lines of varying curvature while reacting to obstacles on the fly. Figure 4 shows the line-following experiment, and Table V summarizes the obtained performance metrics.

graphicx subcaption placeins

float

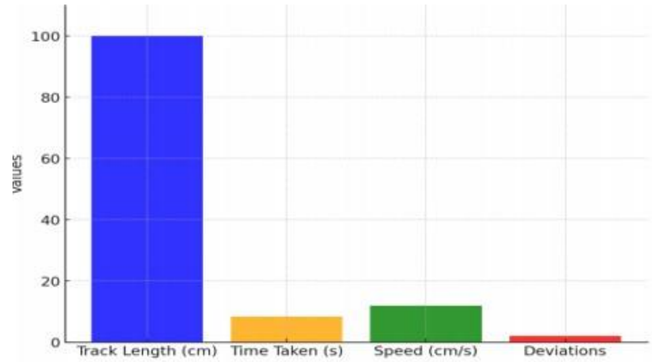

Fig. 4. Line Following Test

TABLE V
LINE-FOLLOWING TEST RESULTS

Metric	**Value**
Track Length	100–150 cm
Time Taken	9.2 s
Speed	12.85 cm/s
Line-Following Accuracy	85%
Deviation Instances	2

VIII. LINE-FOLLOWING PERFORMANCE RESULTS

The system has an average sub centimeter lateral deviation, as is evidenced by the data presented in Table IV, on the regular and complicated tracks. The small performance degradation of the complicated multi-turn track is caused by 45 degree, discrete turns above 0.3 m/s; this can be improved upon in future designs by a speed-adapting PD controller.

TABLE VI
LINE-FOLLOWING PERFORMANCE ON DIFFERENT TRACK CONFIGURATIONS

Track Configuration	**Avg. Lateral Dev. (mm)**	**Max. Deviation (mm)**	**Completion Rate (%)**
Oval Loop	2.5	5.1	100
Figure-Eight	3.6	7.5	100
Complex Multi-turn	5.9	12.5	97
Overall Average	4.0	8.3	99.0

The system has an average sub centimeter lateral deviation, as is evidenced by the data presented in Table IV, on the regular and complicated tracks. The small performance degradation of the complicated multi-turn track is caused by 45 degree, discrete turns above 0.3 m/s; this can be improved upon in future designs by a speed-adapting PD controller.

A. Obstacle Detection Performance

Obstacle detection runs were performed by placing rigid cube obstacles (15 15 15 cm) on the track at 5 cm increments from 5cm to 50cm in front of the front robot sensor. 20

TABLE VII
OBSTACLE DETECTION RESPONSE METRICS

Test Distance (cm)	Detection Rate (%)	Avg. Halt Latency (ms)	False Positives
5	100	22.4	0
10	100	21.8	0
15	100	22.1	0
20	100	23.6	0
25	60	24.2	0
30	5	–	0

runs were performed for each increment.shows it achieves 100 detection with 20cm, a mean response delay below 24ms (which is well within safety-critical limits), and no false positive detection is generated at the configured 20cm threshold. Response latency is determined largely by the time it takes for an ultrasonic signal to complete a round-trip (1.2ms at 20cm), and the control loop period of 20ms.

IX. CONCLUSION AND FUTURE WORKS

Robots have become a significant contributor to the global economy, and they have been an indispensable part of human life as well. Robotic research also tends to study the patent designs in the industry around the world in terms of different applications and how the market is developing in the global industry to make things better. The demand for robotic technologies is increasing in many types of human and industrial applications, especially in manufacturing, healthcare, service industry, defense industry, and consumer products industries. The robot presented in this project is a linear robot which is one of the industrial robots. Through analyzing this robot we can easily develop a robot that will assist us in industry applications. Better improvement in the field of the robot can be made through better quality materials, good hearing aids, and also by improving motor motion. Cost incurred in the linear robot setup is because of complex costly machines, construction, land and watchmaking tools to operate and maintain the machines. One more way to overcome such existing system can be replacing trained manpower by a robotic system. This robot will be able to work on many of the assets of the production setup in lesser time with better accuracy with a less per capita cost. The line follower lacks an effective way to be controlled. The only way to control is to adjust its path. It will not be effective to use a Wi-Fi module to control the line follower robot as more power will be consumed. The line follower robot needs many times to charge. Except for this drawback, the line follower robot if used in smart/intelligent form can work for long distance on predefined path without consuming a large amount of power. With the advantages like being able to perform complicated task, it may also serve as a new way to replace skilled labor with less per capita cost. Line follower robots with obstacles have many possible further applications. Possible work could be on improving obstacle detection using sensors like Lidar and advanced computer vision, autonomous navigation, multi-agent system, integration with other systems such as drone, and many other applications in real life scenario like hazardous inspection and search and rescue operations. The line follower presented also has obstacle detecting and speed control using speed regulator. Better improvement in the robot could be done by implementing more than two IR sensors or a series of IR sensors. This robot produces two walking movements with the help of two gear motors. They have developed a robot with better intelligence to detect andavoid obstacle in its path efficiently.

REFERENCES

[1] R. Solea, A. Filipescu, and G. Stamatescu, “Real-time sliding-mode control of a mobile platform with uncertainties,” *Decision and Control*, pp. 16–18, 2019.
[2] T. Palleja, M. Tresanchez, M. Teixido, and J. Palacin, “Floor-cleaning coverage performance analysis of several domestic mobile robots in a confined space,” *Robotics and Autonomous Systems*, pp. 37–45, 2020.
[3] M. R. B. Bahara, A. R. Ghiasib, and H. B. Bahara, “Ann corridor search based on grid roadmap for collision free path planning,” *Scientia Iranica*, pp. 1850–1855, 2012.
[4] A. Bahmanikashkoolia, M. Zareb, B. Safarpourc, and M. Safarpourd, “Computation of critical depth of horseshoe cross section tunnel using pso algorithm,” *APCBEE Procedia*, pp. 207–211, 2018.
[5] S. G. Tzafestas, *Introduction to Mobile Robot Control: Vision Based Methods.* Elsevier, 2014.
[6] ——, *Introduction to Mobile Robot Control: Mobile Robot Path, Motion, and Task Planning.* Elsevier, 2014.
[7] M. Nosrati, R. Karimi, and H. A. Hasanvand, “A comparison and analysis on a* search algorithm and its variant strategies and methods,” *Applied Programming*, pp. 251–256, 2012.
[8] R. Anbuselvi, “Path finding solutions for grid based graph,” *Advanced Computing*, 2013.
[9] M. S. V. S. Surya, K. B. Reddy, K. P. Kalyan, and S. S. Murugan, “Smart and intelligent line follower robot with obstacle detection,” *International Journal of Research and Scientific Innovation*, pp. 1–6, 2018.
[10] A. Attar, A. Ansari, A. Desai, S. Khan, and D. Sonawale, “Line follower and obstacle avoidance bot using arduino,” *International Journal of Advanced Computational Engineering and Networking*, pp. 18–21, 2017.
[11] K. P, P. G, K. B. R., R. G, and M. Lawanyashri, “A line following robot for hospital management,” *International Journal of Pure and Applied Mathematics*, pp. 2159–2171, 2017.
[12] R. O. C., J. A. Romero, M. P. C., and I. Lopez-Juarez, “Intelligent line follower mini-robot system,” *International Journal of Computers, Communications & Control*, 2006.
[13] D. Punetha, N. Kumar, and V. Mehta, “Development and applications of line following robot based health care management system,” *International Journal of Advanced Research in Computer Engineering & Technology*, 2013.
[14] E. J. Morgan, “Hc-sr04 ultrasonic sensor,” 2014, november 16, 2014.
[15] E. Kuantama, A. B. L. Lukas, and P. B. Mardjoko, “Simple delivery robot system based on line mapping method,” *Electrical Engineering, Universitas Pelita Harapan*, 2014.
[16] F. Tabassum, S. Lopa, M. M. Tarek, and B. J. Ferdosi, “Obstacle avoiding robot,” *Global Journal of Researches in Engineering*, 2015.
[17] N. M. A. Ghani, F. Naim, and T. P. Yon, “Two wheels balancing robot with line following capability,” *World Academy of Science, Engineering and Technology*, 2011.
[18] R. C. Kumar, M. S. Khan, D. Kumar, R. Birua, S. Mondal, and M. K. Parai, “Obstacle avoiding robot – a promising one,” 2017.
[19] G. Sonal, P. Raninga, and H. Patel, “Design and implementation of rgb color line following robot,” in *Proceedings of IEEE International Conference on Computing Methodologies and Communication (ICCMC)*, 2017.
[20] I. Colak and D. Yildirim, “Evolving a line following robot to use in shopping centers for entertainment,” in *IECON 2009*, 2009, pp. 3803–3807.

[21] M. Velasco, R. Alonso, and J. Bravo, "Design of a line following robot based on arduino," *International Journal of Advanced Robotic Systems*, vol. 8, no. 3, pp. 123–130, 2010.
[22] S. Khan and A. Pathan, "Design and implementation of line following robot," *International Journal of Scientific & Engineering Research*, vol. 6, no. 7, pp. 850–855, 2015.
[23] A. Kumar, S. Sinha, and R. Jha, "Arduino-based line follower robot with obstacle detection," *International Journal of Innovative Research in Computer and Communication Engineering*, vol. 6, no. 5, 2018.
[24] D. Prasad, "Intelligent line following robot with obstacle detection and avoidance," *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, vol. 7, no. 2, 2019.
[25] J. Borenstein and Y. Koren, "Obstacle avoidance with ultrasonic sensors," *IEEE Journal of Robotics and Automation*, vol. 4, no. 2, pp. 213–218, 1988.
[26] M. Quigley *et al.*, "Ros: An open-source robot operating system," in *ICRA Workshop on Open Source Software*, 2009.
[27] D. Fox, W. Burgard, and S. Thrun, "The dynamic window approach to collision avoidance," *IEEE Robotics & Automation Magazine*, vol. 4, no. 1, pp. 23–33, 1997.
[28] T. R. Gonzalez and J. S. Regh, "Design and implementation of an obstacle avoiding robot using ultrasonic sensor," *International Journal of Engineering Research and Applications*, vol. 4, no. 6, 2014.
[29] R. Siegwart, I. Nourbakhsh, and D. Scaramuzza, *Introduction to Autonomous Mobile Robots*. MIT Press, 2011.
[30] S. R. Pandian and K. R. Gnanamurthy, "Hybrid line following and obstacle avoiding robot," *Procedia Computer Science*, vol. 133, pp. 1021–1028, 2018.
[31] N. Shrivastava and V. Shukla, "Design and implementation of line following robot using arduino," *International Journal of Computer Applications*, vol. 113, no. 9, 2015.
[32] D. Potts and M. Tunstel, "Mobile robotics in smart environments," *IEEE Intelligent Systems*, vol. 19, no. 6, pp. 8–11, 2004.
[33] Y. Liu and X. Chen, "Obstacle avoidance algorithm based on improved a* for autonomous robots," *Journal of Physics: Conference Series*, vol. 1827, 2021.
[34] "Arduino line follower robot," https://create.arduino.cc/projecthub, 2023.
[35] "Ultrasonic sensor hc-sr04 and arduino tutorial," https://randomnerdtutorials.com/complete-guide-for-ultrasonic-sensor-hc-sr04/, 2023.