

Building Budget-Friendly Obstacle-Avoiding Robots Using Arduino and Ultrasonic Technology

Jaffino G, Padma S, Tanaya Singh, Sapitha P, Ishika Agarwal,
Harrshinishri.S.S

*School of Electronics Engineering, Vellore Institute of Technology, Vellore, Tamil Nadu,
India*

Email: jaffino.g@vit.ac.in

In contemporary society, robotics is rapidly emerging and expanding across various domains, including manufacturing, healthcare, and transportation, demonstrating its ubiquitous influence in diverse industries. One such application is obstacle avoiding robots, which detect obstacles in their path and adjust their routes based on the distance to the obstacle. This research presents the design of an obstacle-avoiding robot using an L298N motor driver, ultrasonic sensor and Arduino embedded system. Initially, the threshold distance and motor speed of the robot are set. In the subsequent stage, ultrasonic sensor data is processed to detect any obstacles in the robot's path. If an obstacle is detected, the servo motor is activated to turn left or right based on the distance threshold value. The schematic view of the proposed method was developed using Tinker cad, and the components were chosen to ensure cost-effectiveness and enhance efficiency in obstacle detection. The proposed method enables the robot to make more precise and effective decisions, improving its ability to avoid potential collisions in dynamic environments.

Keywords: robot, obstacles, ultrasonic sensor, Arduino, normalization.

1. Introduction

Mobile robots are becoming more prevalent in various tasks such as transportation and surveillance. In these scenarios, the robot must autonomously navigate around obstacles to reach its destination, necessitating the development of path planning algorithms for obstacle avoidance. One such robotics application is obstacle avoiding robots. These robots detect obstacles in their way and then change their paths according to the distance between the robot and the obstacle. In this robot Arduino Uno R3 is used for programming, ultrasonic

sensor is used for sensing and measuring the distance between the obstacle and the robot, servo motor is used to rotate the direction of ultrasonic sensor so that it can measure the distance in left and right direction and the robot can move accordingly. Further, the L298N motor driver is used to control the motors and to provide power, voltage needed in order to operate the device. L298N is especially used in this project because of its dual H-bridge design which helps in controlling the direction and speed of two motors independently. It is a fully autonomous wheeled robot which avoids collisions and has various applications in the current era.

Tianye et al. [1] proposed the MVO (Model-based Velocity Obstacle) method to help robots avoid collisions with high-speed objects. This method incorporates holonomic kinematic constraints to enhance the robot's performance in navigating its environment safely and efficiently. Meng et al. [2] introduced the use of Support Vector Machine (SVM) technology for enabling robots to avoid obstacles in complex dynamic environments. Their approach also incorporated the hunting method to optimize the robot's movements in scenarios involving pursuers and evaders. This combination of SVM and the hunting method represents a significant advancement in the field of robotic navigation and obstacle avoidance, offering robust performance in highly dynamic and unpredictable setting.

Yang et al. [3] presented a learning approach for human-robot integration platforms, focusing on the use of the Conditional Variational Autoencoder (CVAE) framework for temporary target detection. Their work details how the CVAE framework can be effectively applied to improve the interaction between humans and robots, enhancing the robot's ability to detect and respond to transient targets in its environment. Zhiwei et al. [4] described a hybrid technique that combines the Time Elastic Band (TEB) algorithm with Optimal Reciprocal Collision Avoidance (ORCA) to predict obstacles. This method was validated using the Gazebo and Rviz simulation environments, and further tested on the Little Ant robot. Their experimental results demonstrate the effectiveness of this hybrid approach in improving obstacle prediction and navigation performance in robotic systems. Shiliang et al. [5] proposed an artificial potential field method for determining inhibiting velocity based on dynamic characteristics. This method demonstrated notable safety and stability, providing an effective approach for enhancing the performance and reliability of robotic navigation systems.

Jaffino et al. [6-11] discussed various optimization techniques for deep learning networks aimed at enhancing model accuracy. Their comprehensive review covers multiple methods, providing insights into how these optimizations can significantly improve the performance of deep learning models across different applications. Johnrose et al. [12] explained rider algorithm for improving the performance of various classification models which can be used for embedded applications. Sundaram et al.[13] proposed c means based feature analysis techniques for data classification. Jamie et al. [14] described a method for predicting future trajectories using the velocities of other robots. This approach effectively reduces oscillations in the robots' movements and aids in adjusting their trajectories, enhancing overall navigation efficiency and stability in multi-robot systems. Van Dang et al. [15] elaborated on the SOAR cognitive architecture combined with a speed planning system for obstacle avoidance. This method has been shown to enhance performance, particularly when navigating turns and corners, thereby improving the overall efficiency and agility of robotic

systems. The key contributions of this research work include: (i) Designing a schematic view of the proposed method using thinker card for evaluating the robot's performance. (ii) Developing an obstacle-avoiding robot utilizing an ultrasonic sensor, servo motor, and Arduino embedded system.

2. Methodology

The proposed methodology for the decision-making process of the designed obstacle-avoiding robot is illustrated in Fig.1. The schematic view of the proposed robot is shown in Fig.2. Initially, the robot initializes by connecting its power supply, sensors, and motors. It then continuously reads data from the ultrasonic sensor to detect obstacles. When an obstacle is detected directly in front and within turning range, the servo motor rotates to face the obstacle and measures the distance. If the obstacle is too close, the robot takes evasive action by turning left or right. If no obstacle is detected in front, the robot checks for obstacles on the left and right sides using the servo motor. Upon detecting an obstacle on either side, the robot measures the distance and takes evasive action by turning. If no obstacles are detected in any direction, the robot moves forward. The flow chart concludes with the robot stopping if there are no obstacles and no further tasks to complete.

The robot is designed to be small and lightweight, which enhances its portability and reduces battery consumption. The compact size also positions the sensors lower, increasing responsiveness to small obstacles. The robot's chassis forms the main body and houses various components, including wheels, batteries, a battery holder, screws, a caster wheel for stability.

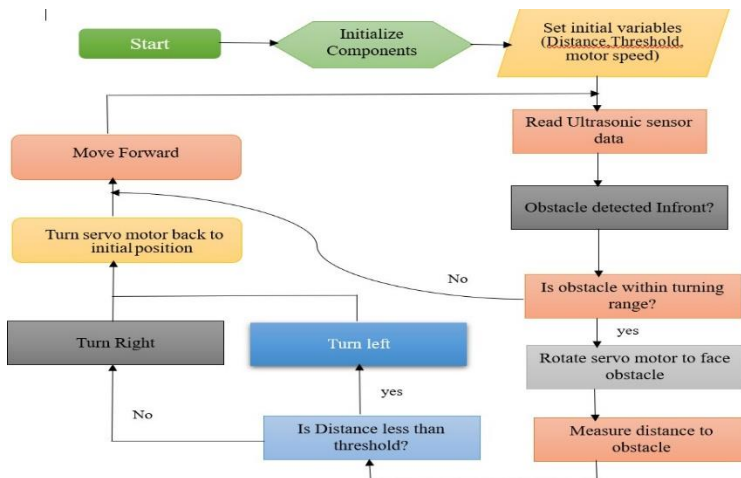


Fig.1 Flow chart of the Proposed method

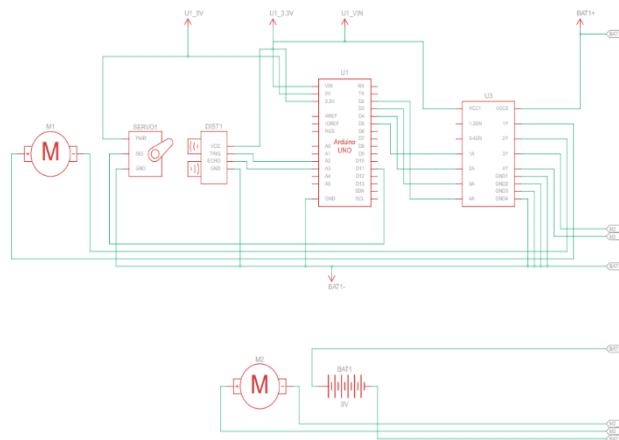


Fig.2 Schematic view of the Proposed Robot

Embedded system

In this work, Arduino embedded system is used to design the robot. Arduino is an open-source platform that offers an Integrated Development Environment (IDE) supported process, facilitating easy programming. An Arduino board reads inputs and converts them into outputs. The Arduino Uno board, in particular, is a low-cost, flexible, and open-source microcontroller suitable for integration into various projects. It features 14 digital input/output pins, 6 analog inputs, a 16MHz ceramic resonator, a USB connection, a power jack, an In-Circuit Serial Programming (ICSP) header, and a reset button.

Ultrasonic Sensor

An ultrasonic sensor is an electronic device that measures the distance between an obstacle and the device by converting reflected waves into an electrical signal. It provides non-contact detection and high-accuracy readings within a range of two to four hundred centimeters. The sensor comprises two components: a transmitter and a receiver. The distance to a target is determined by measuring the time lapses between sending and receiving an ultrasonic pulse. The working principle involves the transmitter emitting ultrasonic sound waves, which reflect off obstacles within the specified range. These reflected waves are then converted into electrical signals. The distance detected by the sensor is calculated as follows

$$d = \frac{t * v}{2} \quad (1)$$

Where d is the distance one way and t is the time taken at high echo for round trip, and v is the speed of sound.

Servo Motor

A servo motor is a type of motor that enables precise linear or angular rotation. It consists of a motor, a potentiometer, a gear assembly, and a control circuit. Essentially, it operates through a servo mechanism. Along with the motor, the ultrasonic sensor mounted on it also rotates. The servo motor is a closed-loop system that uses positive feedback to control the

motion and final position of the shaft. The device is controlled by a feedback signal generated by comparing the output signal with the input signal. The servo motor control is mathematically expressed as

$$\theta(t) = K_p \cdot e(t) + K_i \int e(t)dt + K_d \frac{de(t)}{dt} \quad (2)$$

Where $\theta(t)$ is the angular position of the motor at time t and $e(t)$ is the error signal (the difference between the desired and actual position), K_p , K_i K_d are the proportional, integral, and derivative gains, respectively. The Motor torque is expressed as

$$\tau = K_t \cdot u(t) \quad (3)$$

Here τ is the motor torque, K_t is the motor torque constant and $u(t)$ is the control signal.

Motor Driver

A motor driver is an integrated circuit (IC) chip used to control motors in robotic applications. It is responsible for supplying the necessary power, voltage, current, and protection required to operate the motor and its components. The motor driver is directly connected to the microcontroller, allowing it to transmit control signals from the microcontroller to the motors.

Development

In the design of the robot chassis, the first step involved securely attaching two motors to the chassis. Ensuring a firm attachment while avoiding contact with metal parts is crucial to prevent potential short circuits. Following this, a caster wheel was installed at the front of the chassis to provide proper alignment and stability. The next step involved integrating the wheels with the motors. Ensuring a secure and stable fit was essential to enable smooth movement of the robot. A double battery holder was then added to the rear section of the chassis, firmly securing it to support the robot's power requirements. For ease of operation, a power button was incorporated to allow convenient on/off control. To enable specific functionalities, such as movement or sensing, a servo motor was securely mounted onto the chassis. Careful positioning of the servo was essential for optimal performance. Additionally, the ultrasonic sensor was mounted onto the servo holder, ensuring proper alignment for accurate readings. To prevent potential short circuits, the motor driver was insulated with double-sided tape before being affixed to the chassis. Motor wires were securely soldered to the motor driver, and proper wire connections were made to the power switch. The model of the robot with its fully attached components is shown in Fig.3. For the battery wiring, the red wire was connected to the VCC of the motor driver and the black wire to the ground pin. Specific wiring instructions for the ultrasonic sensor included connecting VCC to 5V, GND to ground, ECHO to A2, and TRIG to A3. The motor driver input connections involved connecting the four input pins to digital pins 2, 3, 4, and 5. Finally, for the servo wiring, digital pin 11 was connected to the servo, with a jumper attached to the ground pin and its VCC connected to 5V for proper functioning. The detailed connections are summarized in Table 1.

Table 1: Components and their connections to the Arduino

Components	Connections
Ultrasonic Sensor	VCC - 5V Gnd - Gnd Echo - A2 Trig - A3
Motor driver	IN1 - Pin5 IN2 - Pin4 IN3 - Pin3 IN4 - Pin2
Servo	Digital Pin 11
Arduino Shield	VCC - 5V (Motor driver) Gnd – Gnd (Motor driver)

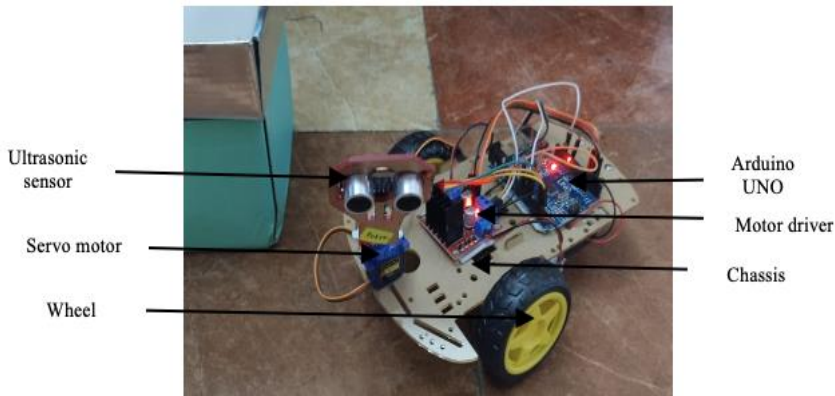


Fig.3 Model of the robot with its components

Working Model

Upon powering on, the robot begins to move forward while continuously calculating the distance between itself and the obstacles using the ultrasonic sensor. The ultrasonic sensor sends electromagnetic waves to the obstacle in front of it and then the distance is being calculated using speed of the wave it takes to hit the object and comes back multiplied by the time it takes. If the distance is found to be less than the threshold distance, then the robot the robot stops and scans in right and left directions to find a clear path using ultrasonic sensor. The ultrasonic sensor first moves to the left direction with the help of servo motor, measures the distance and then the robot turns left and starts moving. If there is any obstacle present in left direction, then the ultrasonic sensor moves in the right direction to measure the distance between the obstacle and the robot and then turns in the direction with more distance and avoids colliding with obstacles as shown in the below scenarios.

3. Experimental Section and discussion

After the completion of the obstacle-avoiding robot model, various experimental conditions were tested to evaluate its performance:

Scenario 1: An object was placed directly in front of the robot at a distance of 7cm, since the threshold distance was set as 10 cm. Upon detecting the obstacle, the robot stopped moving and turned left, where it detected no obstacle and then resumed moving in the left direction as shown in Fig.4.

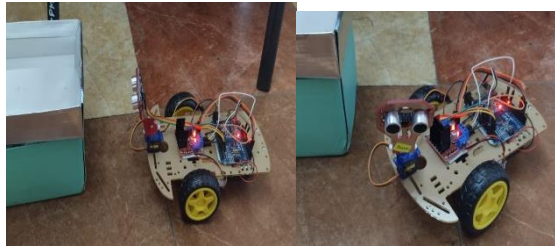


Fig.4 Movement of the robot in scenario 1

Scenario 2: An object was again placed in front of the robot. After detecting the obstacle, the robot stopped, turned left, and detected another obstacle at a distance of 7 cm. The ultrasonic sensor then turned right to scan if there is any obstacle or not, there was no obstacle placed so it detected no obstacle, and the robot turned right and resumed movement as shown in Fig.5.



Fig.5 Movement of the robot in scenario 2

Scenario 3: Obstacles were placed in all directions around the robot within 10 cms. Initially, upon detecting an obstacle, the robot stopped, then the ultrasonic sensor turned left, measured the distance, turned right, and measured the distance again. Since the distances in all directions were equal, the robot stopped to avoid collision until an obstacle was removed as shown in Fig.6.



Fig.6 Movement of the robot in scenario 3

Scenario 4: Obstacles were placed in all directions but at varying distances. The robot turned right, where the distance to the obstacle was 20 cm, and resumed moving until it detected the next obstacle. These experiments demonstrated the robot's ability to navigate and avoid obstacles effectively under different conditions. The operational details and responses of the robot in each scenario have been illustrated in the above figures.

The obstacle-avoiding robot holds significant potential across various applications, showcasing its versatility and impact in different fields. One of the primary applications is in automatic vehicles where it plays a crucial role in collision avoidance. Equipped with obstacle detection capabilities, the robot enables vehicles to halt upon detecting obstacles or humans in their path. It can maneuver by turning left or right based on the proximity of obstacles in both directions, thus enhancing safety and preventing accidents. In space applications, obstacle avoidance is essential for satellites to protect against collisions with space debris. By detecting and avoiding debris, these robots ensure the integrity and longevity of satellite missions. Various mobile and automatic industrial machines benefit from obstacle avoidance to prevent collisions during operation. This capability enhances operational safety and efficiency in industrial settings. Obstacle-avoiding robots are employed in automated floor cleaning equipment, ensuring thorough cleaning without collisions or disruptions. In healthcare settings, these robots can significantly impact operational efficiency and infection control. They can be used to deliver patients' belongings or medical supplies, reducing the risk of spreading communicable diseases, especially in wards with infectious patients. In hazardous environments like mining sites, where human entry poses risks, obstacle-avoiding robots can navigate and perform tasks safely and efficiently. Another critical application is in the development of navigation aids for the visually impaired. These robots can be integrated into smart walking sticks or similar devices, providing real-time obstacle detection and alerting users to obstacles in their path.

4. Conclusion

In this work an obstacle-avoiding robot is developed for autonomous navigation and obstacle detection. The robot's functionality was demonstrated through various experiments, confirming its operational effectiveness under different conditions. The robot operates autonomously when no obstacles are present and it moves forward. Upon detecting an obstacle in its path, the robot stops and activates its sensors. Using a servo motor, the sensors scan left and right to identify clear paths. Once a clear path is identified, the robot maneuvers in that direction to continue its movement. This work showcases the feasibility and reliability of autonomous systems in real-world applications. Once activated, the robot operates independently without requiring human intervention, demonstrating its potential for enhancing safety and efficiency in diverse environments.

References

1. Xu, T., Zhang, S., Jiang, Z., Liu, Z., & Cheng, H. (2020). Collision avoidance of high-speed obstacles for mobile robots via maximum-speed aware velocity obstacle method. *IEEE Access*, 8, 138493-138507.
2. Zhou, M., Wang, Z., Wang, J., & Cao, Z. (2024). Multi-robot collaborative hunting in cluttered environments with obstacle-avoiding voronoi cells. *IEEE/CAA Journal of Automatica Sinica*, 11(7), 1643-1655.
3. Hong, Y., Ding, Z., Yuan, Y., & Chi, W. (2023). Obstacle Avoidance Learning for Robot Motion Planning in Human-Robot Integration Environments. *IEEE Transactions on Cognitive and Developmental Systems*, 15(4), 2169-2178.

4. Wang, Z., Li, P., Li, Q., Wang, Z., & Li, Z. (2023). Motion Planning Method for Car-Like Autonomous Mobile Robots in Dynamic Obstacle Environments. *IEEE Access*, 11, 137387-137400.
5. Shao, S., Zhang, J., Wang, T., Shankar, A., & Maple, C. (2023, December). Dynamic Obstacle-Avoidance Algorithm for Multi-Robot Flocking Based on Improved Artificial Potential Field. *IEEE Transactions on Consumer Electronics*, 70(1), 4388-4399.
6. Ashok, R., Sundaram, M., Jaffino, G., & Jose, J. P. (2023). Weighted Moth-Flame Optimization Algorithm for Edible Oil Quality Detection Using Microwave Technologies. *Food Analytical Methods*, 16(9), 1487-1497.
7. Jaffino, G., & Jose, J. P. (2021). Contour-and Texture-based analysis for victim identification in forensic odontology. *Data Technologies and Applications*, 56(1), 146-160.
8. Jaffino, G., Sundaram, M., & Jose, J. P. (2022, April). Weighted 1D-local binary pattern features and Taylor-Henry gas solubility optimization based Deep Maxout network for discovering epileptic seizure using EEG. *Digital Signal Processing*, 122, 103349.
9. Jaffino, G., and J. Prabin Jose. "Classification of Ayurvedic Medicinal Plant Leaf using KNN Classifier and FPGA." In 2024 10th International Conference on Advanced Computing and Communication Systems (ICACCS), vol. 1, pp. 2255-2259. IEEE, 2024.
10. Jaffino, G., Palaniappan, T., MK, S. S., Gugapriya, G., manikya Kumar, M., & Jose, J. P. (2022, March). IoT Based Frame work for Automatic accident Intimation System. In 2022 8th International Conference on Advanced Computing and Communication Systems (ICACCS) (Vol. 1, pp. 1568-1572).
11. Jaffino, G., Jose, J. P., & Sundaram, M. (2021, September). Deer Hunting Optimization with Deep Neural Network for Automated Arrhythmia diagnosis in ECG Signals. In 2021 Fourth International Conference on Electrical, Computer and Communication Technologies (ICECCT)
12. Johnrose, P. J., Muniasamy, S., & Georgepeter, J. (2021). Rag-bull rider optimisation with deep recurrent neural network for epileptic seizure detection using electroencephalogram. *IET Signal Processing*, 15(2), 122-140
13. Sundaram, M., Jose, J. P., & Jaffino, G. (2014, December). Welding defects extraction for radiographic images using C-means segmentation method. In 2014 International Conference on Communication and Network Technologies (pp. 79-83
14. Snape, J., Beg, J. V. D., Guy, S. J., & Manocha, D. (2011). The hybrid reciprocal velocity obstacle. *IEEE Transactions on Robotics*, 27(4), 696-706.
15. Dang, C. V., Ahn, H., Kim, J., & Lee, S. C. (2023). Collision-free navigation in human-following task using a cognitive robotic system on differential drive vehicles. *IEEE Transactions on Cognitive and Developmental Systems*, 15(1), 78-87.