INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 05, Issue 04, April 2025, pp : 1179-1187
www.ijprems.com
editor@ijprems.com
e-ISSN : 2583-1062
Impact Factor : 7.001

SMART NAVIGATION DEVICE WITH VOICE AND BLUETOOTH INTEGRATION: OBSTACLE AVOIDANCE IN ACTION

Dornala Sonal[1], Mankala Keerthy[2], Telugola Vaishnavi[3], Mrs. Farhana Tabassum[4]

[1,2,3]Student, Department of Computer Engineering, Stanley College of Engineering and Technology for Women, Hyderabad, Telangana, India.

[4]Assistant Professor, Department of Computer Engineering, Stanley College of Engineering and Technology for Women, Hyderabad, Telangana, India.

DOI: https://www.doi.org/10.58257/IJPREMS39489

ABSTRACT

An avoidance system controlled via Bluetooth and voice commands utilizes an Arduino board to navigate environments effectively. Ultrasonic sensors detect obstacles in real-time, enabling precise navigation and control. Users can operate the system remotely using a mobile device connected via Bluetooth, with voice commands offering enhanced usability and convenience. The design integrates key hardware components such as ultrasonic sensors, an Arduino board, and a Bluetooth module, alongside custom algorithms for obstacle detection and communication protocols. Suitable for automation, robotics, and smart environments, the system addresses critical needs for obstacle avoidance. By combining Bluetooth connectivity with voice command functionality, it provides an efficient, user-friendly solution for improving control and safety in various practical scenarios.

Keywords: Arduino Board, Bluetooth-Controlled, Voice-Controlled, Avoidance, Obstacle Detection.

1. INTRODUCTION

The development of an Arduino-based robot car with obstacle avoidance, Bluetooth control, and voice command capabilities is a significant step in robotics. Robotics influences industries like transportation, healthcare, and manufacturing.

Recent advancements in embedded systems and AI have enhanced robotic autonomy and control. This project involves integrating ultrasonic sensors for obstacle detection, Bluetooth modules for remote control, and voice recognition for commands. It provides hands-on learning in electronics, programming, and mechanical design. Current research focuses on improving autonomous navigation with AI and IoT.

This project serves as an educational tool, enhancing problem-solving skills and technical proficiency while demonstrating the potential of robotics in modern technology.

2. METHODOLOGY

The methodology outlines the hardware and software requirements essential for developing an Arduino-based robot car with obstacle avoidance, Bluetooth control, and voice command capabilities.

The selected components and tools facilitate seamless integration and efficient performance of the robotic system.

2.1 Hardware Requirements

2.1.1 Arduino Uno

The Arduino Uno is a widely used microcontroller board that provides a simple platform for electronics and programming. It includes built-in I/O pins, enabling easy connection and control of sensors, motors, and other components. The Arduino Uno is commonly used for prototyping, education, and interactive projects.

Fig 2.1.1: Arduino Uno

2.1.2 Motor Driver Shield

A motor driver shield is an electronic board that connects to the Arduino, enabling control over the speed and direction of DC motors. It utilizes H-bridge circuits for bidirectional movement and adjustable speed while also offering current sensing and overcurrent protection. These shields are widely used in robotics and automation.

Fig 2.1.2: Motor Driver Shield

2.1.3 Servo Motor

A servo motor is an actuator that provides precise control of rotational or linear motion. It operates within a closed-loop system, adjusting based on position feedback for accuracy. Many servo motors incorporate encoders for feedback, ensuring fast and precise movements.

Fig 2.1.3: Servo Motor

2.1.4 DC Motors

Four DC motors are required to drive the robot car, providing the necessary rotational force to move the wheels.

Fig 2.1.4: DC Motors

2.1.5 Wheels

Four wheels are mounted onto the DC motors, allowing the robot to move efficiently.

Fig 2.1.5: Wheels

2.1.6 Ultrasonic Sensor

An ultrasonic sensor detects obstacles in the robot's path. Common models include the HC-SR05 and JSN-SR04T, which use sound waves to determine object distance.

Fig 2.1.6: Ultrasonic Sensor

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 05, Issue 04, April 2025, pp : 1179-1187
e-ISSN : 2583-1062
Impact Factor : 7.001
www.ijprems.com
editor@ijprems.com

2.1.7 Jumper Wires

Jumper wires establish electrical connections between different components, ensuring proper power and signal flow.

Fig 2.1.7: Jumper Wires

2.1.8 Bluetooth Module

A Bluetooth module enables wireless communication, connecting the robot to a mobile device for manual control. Common modules include the HC-05 and HC-06, which are compatible with Arduino. These modules allow users to send movement commands and activate obstacle avoidance functions.

Fig 2.1.8: Bluetooth Module

2.1.9 Battery

A battery supplies power to the robot. A 9V or 6V rechargeable battery can be used to ensure continuous operation.

Fig 2.1.9: Battery

2.2 Software Requirements

2.2.1 Arduino IDE

The Arduino Integrated Development Environment (IDE) is used to write, compile, and upload programs to the Arduino board. It supports C/C++ and includes libraries for controlling various sensors and actuators.

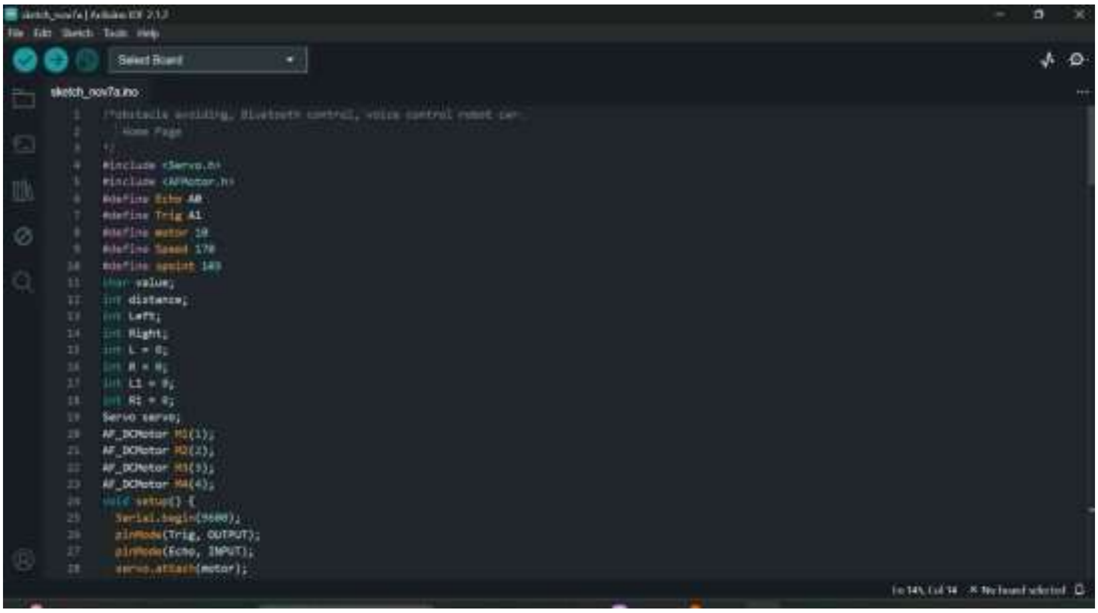

Fig 2.2.1: Arduino IDE

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 05, Issue 04, April 2025, pp : 1179-1187
www.ijprems.com
editor@ijprems.com
e-ISSN : 2583-1062
Impact Factor : 7.001

2.2.2 Bluetooth Control App

A custom or prebuilt Android application is used to control the robot's movements via Bluetooth. The app provides an interface with buttons for movement control and supports voice commands.

Fig 2.2.2: Bluetooth Control App

2.2.3 Mobile Device with Bluetooth Capability

A smartphone or tablet running the Bluetooth control app is used to remotely operate the robot by sending movement commands.

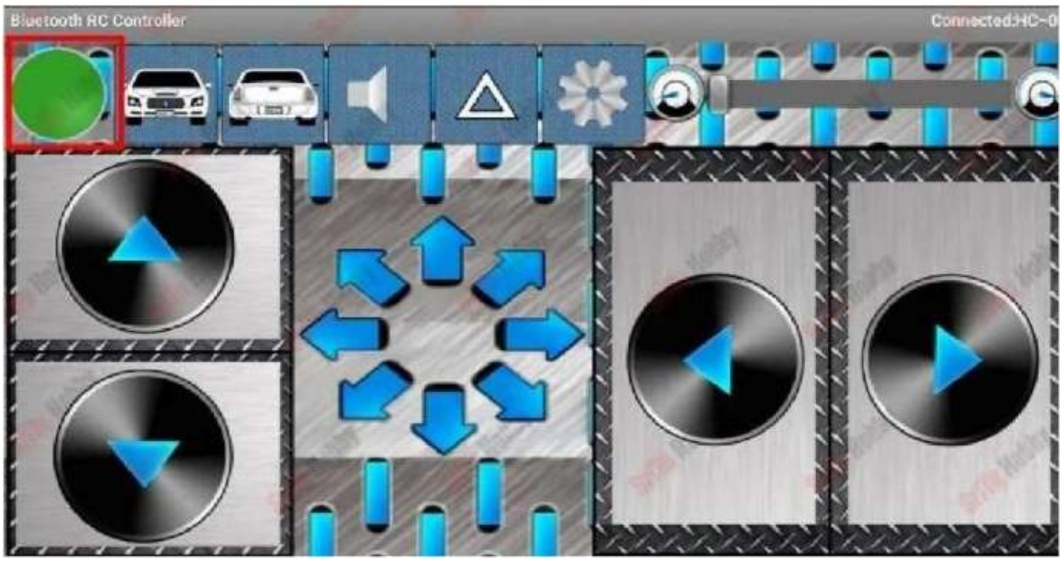

Fig 2.2.3: Mobile Device with Bluetooth Capability

2.2.4 Voice Command Processing Software

Voice recognition software, such as Google's Speech API or custom voice command processing, is used to interpret and execute spoken instructions for robot movement.

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 05, Issue 04, April 2025, pp : 1179-1187
e-ISSN : 2583-1062
Impact Factor : 7.001
www.ijprems.com
editor@ijprems.com

Fig 2.2.4: Voice Command Processing Software

3. MODELING AND ANALYSIS

3.1 Assemble the Hardware Components:

Begin by collecting all necessary components, including the robot chassis, DC gear motors, wheels, motor driver shield, ultrasonic sensor, Arduino board, and power source (battery) according to the circuit diagram.

3.2 Install the Motor Driver Shield:

Connect the DC gear motors to the motor driver shield and then attach the motor driver shield for the Arduino board. This will allow for the control of motor speed and direction.

3.3 Set Up the Ultrasonic Sensor:

Attach Connecting the ultrasonic sensor to the Arduino board, ensuring the sensor is positioned to detect obstacles in the robot's path.

3.4 Write the Arduino Code:

Develop the Arduino code that will enable the robot to detect obstacles using the ultrasonic sensor, avoid them, and control its movement based on the input from the sensors.

Code explanation

First, the necessary libraries are imported

```
#include <Servo.h>
#include <AFMotor.h>
```

Secondly, ultrasonic sensor pins, servo motor pins, motor speed, and servo motor starting point are defined

```
#define Echo A0
#define Trig A1
#define motor 10
#define Speed 170
#define spoint 103
```

Thirdly, some variables have been created to help the program

```
char value;
int distance;
int Left; int
Right; int L
= 0; int R =
0; int L1 =
0; int R1 =
0;
```

Then, objects are created for the Servo Library and the AFMotor Library

```
Servo servo;
AF_DCMotor M1(1);
AF_DCMotor M2(2);
AF_DCMotor M3(3);
AF_DCMotor M4(4);
```

In the setup function, Ultrasonic pins are set to INPUT and OUTPUT. Also, the gear motor speeds have been included

```
void setup() {
Serial.begin(9600);
pinMode(Trig, OUTPUT);
pinMode(Echo, INPUT);
servo.attach(motor);
M1.setSpeed(Speed);
M2.setSpeed(Speed);
M3.setSpeed(Speed);
M4.setSpeed(Speed);
}
```

In the loop function, the three main functions are included. we can run these functions one by one. These are described below

```
void loop() {
 //Obstacle();
 //Bluetoothcontrol();
 //voicecontrol();
}
```

This function includes the Bluetooth control code. The code lines are described one by one in the code

```
void Bluetoothcontrol() {
//gets the serial communication values and puts them into the char variable. if (Serial.available() > 0) {
value = Serial.read();
Serial.println(value);
}
//Next, these values are checked using the IF condition.
//Then, if the char value is 'F', the car moves forward. if (value == 'F') {
forward();
//If the char value is "B", the car moves backward.
 } else if (value == 'B') { backward();
//If the char value is "L", the car moves left.
 } else if (value == 'L') { left();
```

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 05, Issue 04, April 2025, pp : 1179-1187
www.ijprems.com
editor@ijprems.com
e-ISSN : 2583-1062
Impact Factor : 7.001

```
  //If the char value is "R", the car moves right.
} else if (value == 'R') { right();
  //If the char value is "S", the car is stopped.
} else if (value == 'S') { Stop()
}
}
```

This function includes the obstacle-avoiding code. The code lines are described one by one in the code

```
void Obstacle() {

//gets the ultrasonic sensor reading and puts it into the variable.   distance
= ultrasonic();

//then, these values are checked using the IF condition.
//If the value is less than or equal to 12,
//the robot is stopped and the servo motor rotate left and right.
// Also, gets both side distance.
if (distance <= 12) {     Stop();
backward();          delay(100);
Stop();          L  = leftsee();
servo.write(spoint);
delay(800);    R = rightsee();
   servo.write(spoint);

//After, if the left side distance less than the right side distance. The robot turns right
if (L < R) {
    right();
delay(500);
Stop();
    delay(200);

//After, if the left side distance more than the right side distance. The robot turns left.
   } else if (L > R) {
left();
delay(500);
Stop();
    delay(200);
   }

//Otherwise, the robot moves forward.
  } else {
forward();
  }
}
```

This function includes the voice control code. The code lines are described one by one in the code

```
void voicecontrol() {
//gets the serial communication values and puts them into the char variable.
 if  (Serial.available()  >  0)  {
value = Serial.read();
   Serial.println(value);
//If the char value is "^", the car moves forward.
```

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 05, Issue 04, April 2025, pp : 1179-1187
www.ijprems.com
editor@ijprems.com
e-ISSN : 2583-1062
Impact Factor : 7.001

```
    if (value == '^') {      forward();
//If the char value is "-", the car moves backward.
    } else if (value == '-') {
      backward();
//If the char value is "<", the car moves left.
    } else if (value == '<') {
      L = leftsee();     servo.write(spoint);
      if (L >= 10 ) {       left();       delay(500);
        Stop();
      } else if (L < 10) {
        Stop();      }
//If the char value is ">", the car moves right.
    } else if (value == '>') {
      R = rightsee();      servo.write(spoint);      if
(R >= 10 ) {       right();       delay(500);
        Stop();
      } else if (R < 10) {
        Stop();      }
//If the char value is "*", the car is stopped.
    } else if (value == '*') {
      Stop();
    }
  }
}
```

3.5 Test the Robot Car:

Power on the robot and observe its response to the ultrasonic sensor's obstacle detection. Test how it reacts to the input from the user, whether it’s automated or manually controlled.

3.6 Fine-tune the Code:

Adjust the Arduino code to enhance performance. This can include optimizing the movement patterns, adjusting obstacle detection sensitivity, or fine-tuning the motor control for smoother navigation.

3.7 Incorporate a User Interface:

Add a user interface for easier control. This could involve a Bluetooth module to enable remote control via a smartphone or computer.

3.8 Enclose the Robot Car:

Design a protective casing to encase the robot’s components, helping protect sensitive parts like the Arduino board and motor drivers from physical damage, while also increasing the robot’s durability.

3.9 Test and Debug:

Conduct thorough testing to ensure the robot operates as expected. Troubleshoot and debug any issues that arise, ensuring the robot can perform obstacle avoidance, respond to commands, and navigate effectively in its environment.

Fig 3.9: Trimodular Vehicle

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 05, Issue 04, April 2025, pp : 1179-1187
www.ijprems.com
editor@ijprems.com
e-ISSN : 2583-1062
Impact Factor : 7.001

4. CONCLUSION

A summary of various methods used to enhance obstacle-avoiding robots highlights the integration of Bluetooth and voice control systems. Recent studies have explored combining ultrasonic sensors, Bluetooth for remote operation, and voice commands for user interactivity, making robots more versatile and adaptive. These advancements emphasize the potential of multi-modal systems in applications such as education, industrial automation, and real-time navigation, pointing to a future where robots are both autonomous and user-friendly. The integration of these different approaches opens new opportunities for enhancing robot functionality and accessibility in dynamic environments.

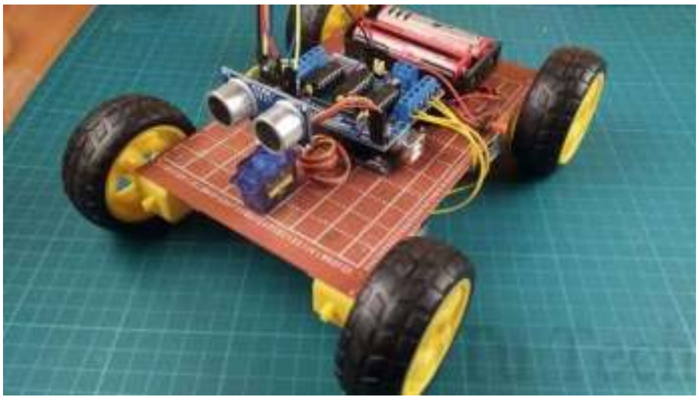

Fig 4: Obstacle avoidance robot car

5. REFERENCES

[1] V. Teeda, K. Sujatha, and R. Mutukuru, "Robot Voice: A Voice Controlled Robot using Arduino," *International Journal of Engineering and Advanced Technology (IJEAT)*, 2016.

[2] A. N. Khan, K. Priya, and S. Kumar, "Implementation of Voice-Controlled Robotic Vehicle with Automatic Braking and Obstacle Avoidance," 2018.

[3] A. Tiwari, R.K. Ghanwar, and M. Red S, "Gesture Controlled Robot with Obstacle Avoidance using Arduino," *International Research Journal of Engineering and Technology (IRJET)*, 2019.

[4] S. Srivastava and R. Singh, "Voice Controlled Robot Car Using Arduino," *International Research Journal of Engineering and Technology (IRJET)*, vol. 07, no. 05, May 2020, ISSN 2395-0056.

[5] S. Caroline, Y. Bekuma, and J. S. Alaric, "Obstacle Avoiding Voice Controlled Bluetooth Car Using Arduino," *International Journal of Innovative Research in Science, Engineering and Technology (IJIRSET)*, vol. 9, no. 11, November 2020, ISSN 2319-8753.

[6] S. Marne and D. Bomble, "Human Voice Controlled Robot Embedded with Real-Time Obstacle Detection and Avoidance," *International Journal of Engineering Research & Technology (IJERT)*, 2020, ISSN 2278-0181.

[7] R. A. Narhare, M. G. Pawar, A. M. Nagare, L. D. Jadhav, and D. S. Thosar, "Smart Voice Controlled Vehicle with Obstacle Detection Using IoT," *International Journal of Innovative Research in Technology*, 2021.

[8] S. Muruganantham, S. Dhivya, S. Nandhini, K. Narmatha, and K. Ramya, "An Obstacle Avoiding Robot Vehicle," *International Journal of Research*, 2021.

[9] A. Ananth and A. S. C, "Bluetooth Based Obstacle Avoiding Robot," *International Advanced Research Journal in Science, Engineering and Technology*, 2021.

[10] S. Venkatesh, S. Kundan, and A. Srija, "Obstacle Avoidance Robotic Vehicle Using HC-SR04 Ultrasonic Sensor," *Turkish Journal of Computer and Mathematics Education*, 2021.

[11] A. Bhikule and S. Mankame, "Obstacle Avoidance and Voice Control Unit for Autonomous Car," *International Journal of Scientific Research in Engineering and Management (IJSREM)*, vol. 05, no. 03, March 2021, ISSN 2582-3930.

[12] T. S, S. M. S, B. K, J. S. S, B. H. S, and P. K. S, "Obstacle Sensing Assistance for Visually Impaired Person Using Arduino," *IEEE*, 2021.

[13] H. Shabani, M. R. M. Razif, and M. F. Muhsin, "Smart Car Control System-Based Arduino Uno for Obstacles Avoidance and Engine Temperature Control," *IEEE*, 2021.