Human Following Robot

Gunjan Kumar
Institute of Infrastructure Technology Research Management Ahmedabad
Email: anshuman7520gunjan@gmail.com

***Abstract*—The *Human Following Robot* project presents the design and implementation of an autonomous robotic system capable of detecting, tracking, and following human subjects. The system integrates Arduino microcontrollers, ultrasonic sensors, and motor drivers to achieve real-time obstacle detection and adaptive navigation. This work highlights the relevance of low-cost autonomous robots in assisting humans across various domains, including household, industrial, and military applications. Experimental results validate the effectiveness of the system in indoor settings while highlighting areas for improvement in robustness and scalability.**

***Index Terms*—Autonomous Robot, Arduino, Ultrasonic Sensors, Human Following, Obstacle Detection, Robotics.**

I. Introduction

Robotics has emerged as a vital field in modern engineering, providing solutions that enhance automation, reduce manual effort, and improve human quality of life. A human-following robot represents an intelligent system capable of tracking and following a human user while avoiding obstacles. Such robots find applications in industrial environments for transporting goods, in households for assisting the elderly, and in commercial settings such as smart trolleys in shopping malls and airports.

The objective of this project is to design and implement a cost-effective, reliable robotic prototype using widely available electronic components. By leveraging Arduino Uno, ultrasonic sensors, and motor drivers, the robot can autonomously follow a target and make navigation decisions in real time. This paper presents a detailed description of the system's design, working principle, results, and potential scope for future enhancement.

II. Literature Review

Several researchers have explored autonomous robots that can follow humans and navigate safely. Early work primarily focused on vision-based systems, where cameras and image processing algorithms were used to detect and track humans. While effective, these approaches often suffered from high computational requirements and reduced accuracy in low-light or crowded environments.

Low-cost alternatives, such as sensor-based navigation, emerged as an efficient solution. Ultrasonic sensors have been widely applied due to their affordability, reliability, and ease of integration with microcontrollers. They enable distance measurement by calculating the time taken for ultrasonic waves to bounce back from an object. Projects like [1] and [3] demonstrated how multiple ultrasonic sensors can be combined with Arduino to implement simple obstacle detection and line following.

The integration of microcontrollers such as Arduino Uno or ESP8266 further improved decision-making capabilities, enabling real-time control of motors and sensors. More recent studies, such as [2], highlight the use of IoT and wireless communication to enhance mobility and user interaction.

Compared to existing solutions, this project emphasizes simplicity and affordability while still maintaining the core functionalities of human detection, following, and obstacle avoidance. By relying on ultrasonic sensors and Arduino, the proposed system balances performance and cost, making it suitable for educational purposes, small-scale industrial applications, and personal assistance.

III. Methodology

The Human Following Robot was implemented using a combination of hardware and software modules. The Arduino Uno microcontroller serves as the core of the system, interfacing with ultrasonic sensors for distance measurement and an L298 motor driver to control the robot's motors.

A. System Workflow

The robot's operation follows these steps:

1) Ultrasonic sensors measure the distance of objects and the human subject.
2) The Arduino processes these values using threshold-based decision rules.
3) The L298 motor driver regulates the DC motors to move the robot forward, stop, or turn.
4) The robot continuously adjusts its trajectory to follow the human while avoiding collisions.

B. Hardware Implementation

The hardware includes Arduino Uno, ultrasonic sensors, motor driver, BO motors, a chassis, Li-ion battery, and jumper connections. Figs. 1, 2, 3, and 4 illustrate the system design.

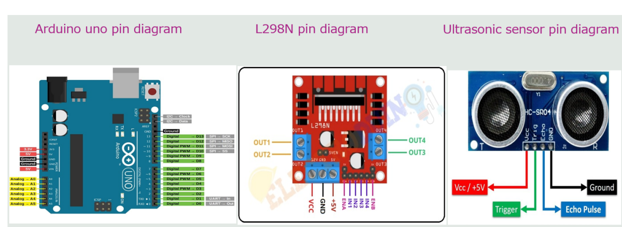

Fig. 1: Components required for the Human Following Robot

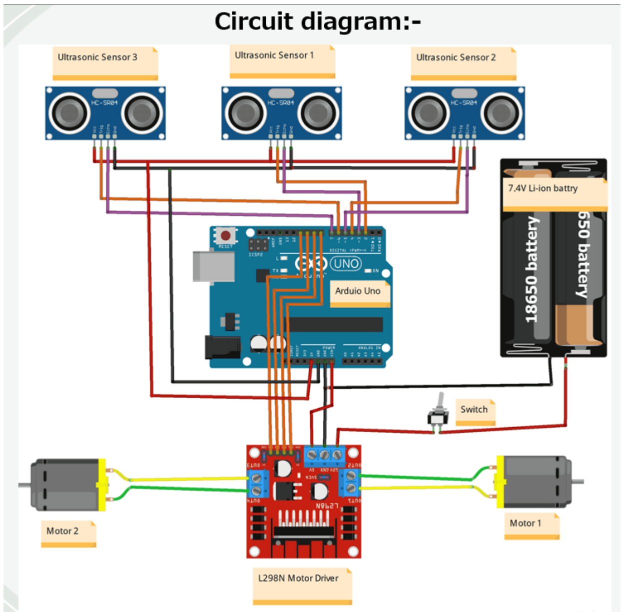

Fig. 2: Ultrasonic sensors used for detection

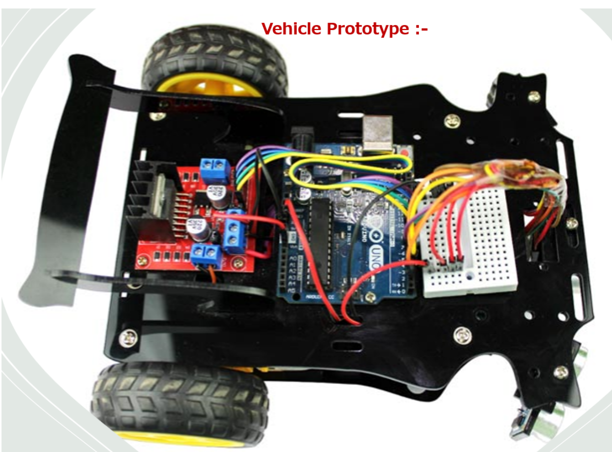

Fig. 3: Robot chassis and motor assembly

C. Software Implementation

The Arduino program reads sensor data, applies logic for distance evaluation, and controls the motors. A simplified snippet is shown in Listing 1.

```
void loop() {
  int frontDistance = sensorOne();
  if (frontDistance < 10) {
    stop();
  } else {
    moveForward();
  }
}
```

Listing 1: Simplified Arduino loop function

IV. Results

The robot was tested in an indoor environment. It successfully detected human presence and maintained a consistent following distance of up to 40 cm. Obstacle avoidance worked reliably, preventing collisions and ensuring safe navigation. The robot demonstrated smooth movement, accurate turns, and stable operation.

However, outdoor testing revealed limitations due to sensor noise in bright sunlight and reduced accuracy on reflective surfaces. Payload capacity was also limited due to the use of lightweight BO motors.

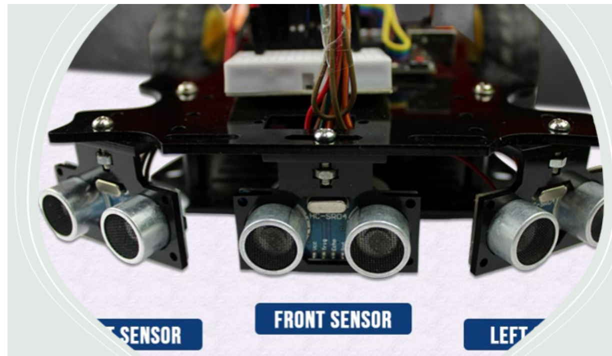

Fig. 4: Circuit diagram of the Human Following Robot

V. Discussion

The results validate that ultrasonic sensors combined with Arduino microcontrollers can provide reliable performance for basic human-following applications. The project highlights the balance between affordability and functionality, making it suitable for real-world scenarios where advanced robotics solutions may be cost-prohibitive. Nevertheless, reliance solely on ultrasonic sensing restricts adaptability in complex environments. Incorporating vision or infrared sensors could significantly enhance robustness.

VI. Conclusion

The Human Following Robot demonstrates an effective design for autonomous human tracking and obstacle avoidance. It provides a cost-effective solution for applications such as industrial logistics, elderly assistance, and smart carts. The project successfully meets its objectives, establishing a foundation for more advanced robotic systems.

VII. Future Scope

Potential advancements to this system include:

- Integration of computer vision for enhanced human recognition.
- IoT connectivity for real-time monitoring and control.
- AI-based learning algorithms for dynamic environments.
- Larger chassis and powerful motors for industrial payloads.

Appendix

```
#define S1Trig 2
#define S2Trig 4
#define S3Trig 6
#define S1Echo 3
#define S2Echo 5
#define S3Echo 7

#define LEFT_MOTOR_PIN1 8
#define LEFT_MOTOR_PIN2 9
#define RIGHT_MOTOR_PIN1 10
#define RIGHT_MOTOR_PIN2 11

#define MAX_DISTANCE 40
#define MIN_DISTANCE_BACK 5
#define MAX_SPEED 150
#define MIN_SPEED 75

void setup() {
  pinMode(LEFT_MOTOR_PIN1, OUTPUT);
  pinMode(LEFT_MOTOR_PIN2, OUTPUT);
  pinMode(RIGHT_MOTOR_PIN1, OUTPUT);
  pinMode(RIGHT_MOTOR_PIN2, OUTPUT);
  pinMode(S1Trig, OUTPUT);
  pinMode(S2Trig, OUTPUT);
  pinMode(S3Trig, OUTPUT);
  pinMode(S1Echo, INPUT);
  pinMode(S2Echo, INPUT);
  pinMode(S3Echo, INPUT);
  Serial.begin(9600);
}

void loop() {
  int frontDistance = sensorOne();
  int leftDistance = sensorTwo();
  int rightDistance = sensorThree();

  if (frontDistance < MIN_DISTANCE_BACK) {
    moveBackward();
  } else if (frontDistance < leftDistance &&
             frontDistance < rightDistance &&
             frontDistance < MAX_DISTANCE) {
    moveForward();
  } else if (leftDistance < rightDistance &&
             leftDistance < MAX_DISTANCE) {
    turnLeft();
  } else if (rightDistance < MAX_DISTANCE) {
    turnRight();
  } else {
    stop();
  }
  delay(100);
}

int sensorOne() {
  digitalWrite(S1Trig, LOW); delayMicroseconds(2);
  digitalWrite(S1Trig, HIGH); delayMicroseconds(10);
  digitalWrite(S1Trig, LOW);
  long t = pulseIn(S1Echo, HIGH);
  return t / 29 / 2;
}

int sensorTwo() {
  digitalWrite(S2Trig, LOW); delayMicroseconds(2);
  digitalWrite(S2Trig, HIGH); delayMicroseconds(10);
  digitalWrite(S2Trig, LOW);
  long t = pulseIn(S2Echo, HIGH);
  return t / 29 / 2;
}

int sensorThree() {
  digitalWrite(S3Trig, LOW); delayMicroseconds(2);
  digitalWrite(S3Trig, HIGH); delayMicroseconds(10);
  digitalWrite(S3Trig, LOW);
  long t = pulseIn(S3Echo, HIGH);
  return t / 29 / 2;
}

void moveForward() {
  analogWrite(LEFT_MOTOR_PIN1, MAX_SPEED);
  analogWrite(LEFT_MOTOR_PIN2, LOW);
  analogWrite(RIGHT_MOTOR_PIN1, MAX_SPEED);
  analogWrite(RIGHT_MOTOR_PIN2, LOW);
}

void moveBackward() {
  analogWrite(LEFT_MOTOR_PIN1, LOW);
  analogWrite(LEFT_MOTOR_PIN2, MAX_SPEED);
  analogWrite(RIGHT_MOTOR_PIN1, LOW);
  analogWrite(RIGHT_MOTOR_PIN2, MAX_SPEED);
}

void turnRight() {
  analogWrite(LEFT_MOTOR_PIN1, LOW);
  analogWrite(LEFT_MOTOR_PIN2, MAX_SPEED);
  analogWrite(RIGHT_MOTOR_PIN1, MAX_SPEED);
  analogWrite(RIGHT_MOTOR_PIN2, LOW);
}

void turnLeft() {
  analogWrite(LEFT_MOTOR_PIN1, MAX_SPEED);
  analogWrite(LEFT_MOTOR_PIN2, LOW);
  analogWrite(RIGHT_MOTOR_PIN1, LOW);
  analogWrite(RIGHT_MOTOR_PIN2, MAX_SPEED);
}

void stop() {
  analogWrite(LEFT_MOTOR_PIN1, LOW);
  analogWrite(LEFT_MOTOR_PIN2, LOW);
  analogWrite(RIGHT_MOTOR_PIN1, LOW);
  analogWrite(RIGHT_MOTOR_PIN2, LOW);
}
```

Listing 2: Full Arduino program for Human Following Robot

References

[1] Circuit Digest. Human Following Robot using Arduino and Ultrasonic Sensor. Available: https://circuitdigest.com/microcontroller-projects/human-following-robot-using-arduino-and-ultrasonic-sensor

[2] Project Hub. Human Following Bot. Available: https://projecthub.arduino.cc/mohammadsohail0008/human-following-bot-f139db

[3] Techatronics. Human Following Robot using Arduino. Available: https://techatronic.com/human-following-robot-using-arduino