GESTURE CONTROLLED ROBOT CAR

Pavani R Sharma

1RVU23CSE332

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING (SoCSE)
RV University
Bengaluru.

OCTOBER-2024

GESTURE CONTROLLED ROBOT CAR

Submitted by

PAVANI R SHARMA

1RVU23CSE332

Under the guidance of

Prof. Kalpana Devi

Centre of IOT and EC

Submitted in partial fulfillment of the requirements for the summer internship conducted during July-August 2024, as part of the Btech (Hons.) program at the School of Computer Science and Engineering.

प्रज्ञा: ★ धीरा:

RV UNIVERSITY

Go, change the world

an initiative of RV EDUCATIONAL INSTITUTIONS

CERTIFICATE

This is to certify that the work presented in this report, entitled ***GESTURE CONTROLLED ROBOT CAR*** has been carried out by ***PAVANI R SHARMA (USN NO: 1RVU23CSE332)*** as part of the summer internship during July-August 2024. The work was completed under my supervision and guidance at the School of Computer Science and Engineering, RV University, Bengaluru, India.

I confirm that the content of this report is the student's original work and has not been submitted elsewhere for the award of any degree or diploma.

Prof. Kalpana Devi
Assistant Professor
School of Computer Science and Engineering,
RV University, Bengaluru.

(Signature)

Center Head	**Program Head**
Prof. Chandramouleeswaran Sankaran	Dr.Sudhakar K. N
School of Computer Science and Engineering, (SoCSE) RV University, Bengaluru.	School of Computer Science and Engineering, (SoCSE) RV University, Bengaluru.
(Signature)	(Signature)

Bengaluru,
October, 2024

CERTIFICATE

I hereby certify that the work presented in this report, entitled ***"GESTURE CONTROLLED ROBOT CAR"*** has been carried out by me during my summer internship in July-August 2024. This work was conducted under the supervision of Prof. Kalpana Devi Assistant Professor, School of Computer Science and Engineering, RV University, Bengaluru, India.

I confirm that the work embodied in this report is my own and has not been submitted for the award of any degree or diploma.

Pavani R Sharma

1RVU23CSE332
School of Computer Science and Engineering,
RV University, Bengaluru.

Bengaluru,

October, 2024

ACKNOWLEDGEMENT

We are deeply grateful to Prof. Chandramouleeswaran Sankaran and Prof. Vidya M J for entrusting us with the opportunity to work on this project. Their encouragement, feedback, and expertise provided us with the foundation to explore and develop our skills throughout this endeavor. We extend special thanks to Prof. Kalpana Devi, whose unwavering guidance and insightful suggestions helped us navigate challenges and enhance the quality of our work.

We would also like to express our appreciation to our classmates and peers for their constructive input and collaboration during various stages of the project. Finally, we are indebted to RV University for creating an academic environment that fosters critical thinking, innovation, and creativity. The resources and support made available to us were pivotal in bringing this project to completion.

INTRODUCTION

Human-computer interface (HCI) and robotics are two interrelated fields that greatly affect technology and human users. HCI focuses on designing and improving interactions between users and computer systems, to improve functionality, accessibility, and user experience. The key areas in HCI are user interface design, functionality, user experience, accessibility, interface design, and psychological thinking. These principles are applied to software design, virtual and augmented reality, game design, and accessibility tools. The design, planning, building, and operating of robots, on the other hand, is known as robotics. It combines mechanical design, control systems, artificial intelligence, human-robot interaction (HRI), sensing, and guidance. Industrial Automation, Robotics Applications. From robots, they progress to social, search, and service robots. The interface between HCI and robotics is reflected in HRI, where HCI principles are applied to create flexible and efficient interactions between humans and robots. User-centered thinking in robotics ensures that robots meet the needs and wants of the user while considering the ethical and social implications of technology in both sectors, such as privacy, security, and job migration and designers, users by combining insights from HCI and robotics to develop flexible and adaptable robotic systems in the context of life.

Human-computer interaction (HCI) and robotics are interrelated industries that together enhance the way humans interact with technology. HCI is dedicated to developing and improving the interface between users and computer systems, with an emphasis on usability, accessibility, and user experience.This covers topics including usability testing, interface design, accessibility, user experience (UX) design, and user interface design. These ideas have been used in many different contexts, including virtual and augmented reality systems, interactive installations, game design, software development (including mobile apps and web interfaces), and the use of tools to guarantee that all program users will be productive and at ease. Conversely, robotics encompasses the creation, building, functioning, and utilization of robots from fields like computer science, mechanical engineering, electrical engineering, vision, control, and so forth. The applications of robots are large and diverse, ranging from industrial applications (such as assembly lines) and medical robots (such as surgical assistants) to industrial robots (e.g. robots for storage and delivery), exploration robots (such as those used in space or underwater missions), and social robots (or for caring for the elderly). Or as if designed for companionship). The interface between HCI and robotics is most evident in human-robot interaction (HRI), where principles from HCI are applied to make robots more flexible and efficient for human users This includes designing interfaces to control robots, enabling natural human interaction mechanisms plus ensuring safety and reliability for human-robot interaction.

User-centric approaches from HCI in robotics are critical to understanding the user's needs and wants, ensuring that a robot is designed with the end user's intellect. This integration is not only practical but user-friendly and integrates easily in human contexts is important for the development of feasible robots. Both HCI and robotics address the broader ethical and social implications of the technology. They consider privacy, security, the potential for job displacement due to automation, the social impact of increased use of robotics, etc. By integrating an application-oriented HCI approach with advanced robotic capabilities, engineers and designers can develop robotic systems that are not only technically advanced but also socially responsible Supportive technologies Improvements enhance human capabilities and improve lives, not create new challenges or differences.

OBJECTIVE

The project Gesture controlled robot car aims to control a robot car with the hand gestures via wireless communication. The proposed system has sensors and arduino programming.

METHODOLOGY

The development of the gesture-controlled robot car involves a structured methodology as outlined below:

1. **System Design**
   The project is divided into two main parts: the **transmitter** and the **receiver**.
   - **Transmitter Section Components**:

     ADXL335 Accelerometer: A 3-axis accelerometer is used to capture hand gestures. The sensor measures both static and dynamic acceleration, providing data on hand movements.

     RF+ Nano Board: The accelerometer's output is sent wirelessly to the receiver using the NRF24L01 RF module.

   - **Receiver Section Components**:

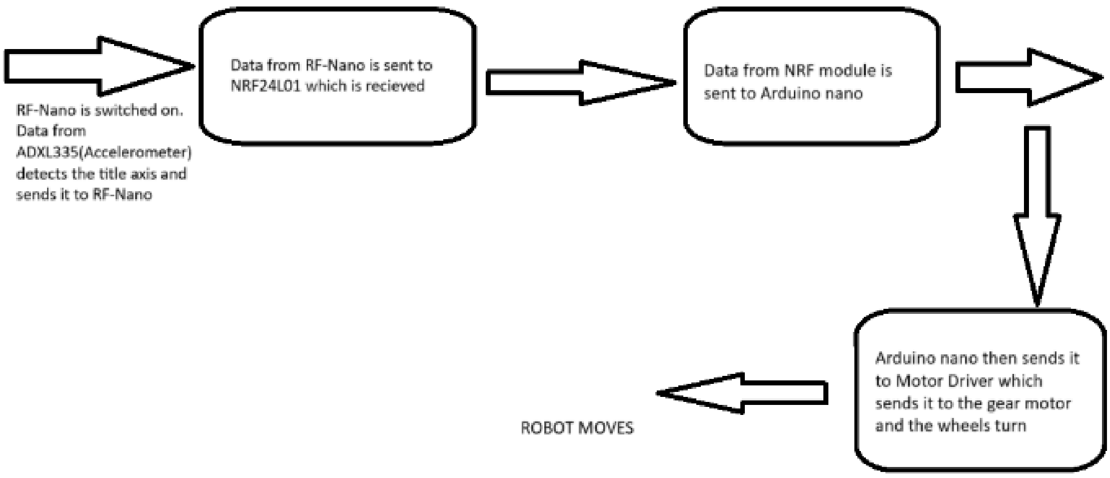
     Arduino Nano: This microcontroller receives data from the NRF24L01 module, processes it, and controls the robot's movement. Arduino Nano was chosen for its compact size and ease of programming.

     NRF24L01 Module: Acts as the receiver for hand gesture data from the transmitter.

     L298N Motor Driver: This driver controls the motors that dictate the robot's movement in response to the gestures.

TP5100: Used for power management and battery charging.

2. **Gesture Recognition Algorithm**
   A gesture recognition algorithm is implemented in the Arduino programming environment. The algorithm translates accelerometer data into motion commands, such as forward, backward, left, and right. Each hand gesture corresponds to a specific motion of the robot car.
3. **Wireless Communication Setup**
   The NRF24L01 module is set up to enable wireless communication between the transmitter and receiver. The module operates at 2.4 GHz and can transmit data over a range of 100 meters in an open space. SPI protocol is used for communication between the RF module and the Arduino Nano.
4. **Control Logic**
   The Arduino Nano processes the gesture data from the accelerometer and sends corresponding signals to the motor driver. Depending on the hand movement, the motor driver controls the speed and direction of the motors.
5. **Testing and Calibration**
   - Calibration of the accelerometer was done to ensure accurate gesture recognition. Various hand movements were tested and mapped to robot commands.
   - Wireless communication was tested for range and reliability.
   - The system was fine-tuned to eliminate noise or erroneous commands caused by unintended gestures.
6. **Power Management**
   The TP5100 chip was used to manage the battery charging process, ensuring that the system has efficient power use without interruptions during operation.

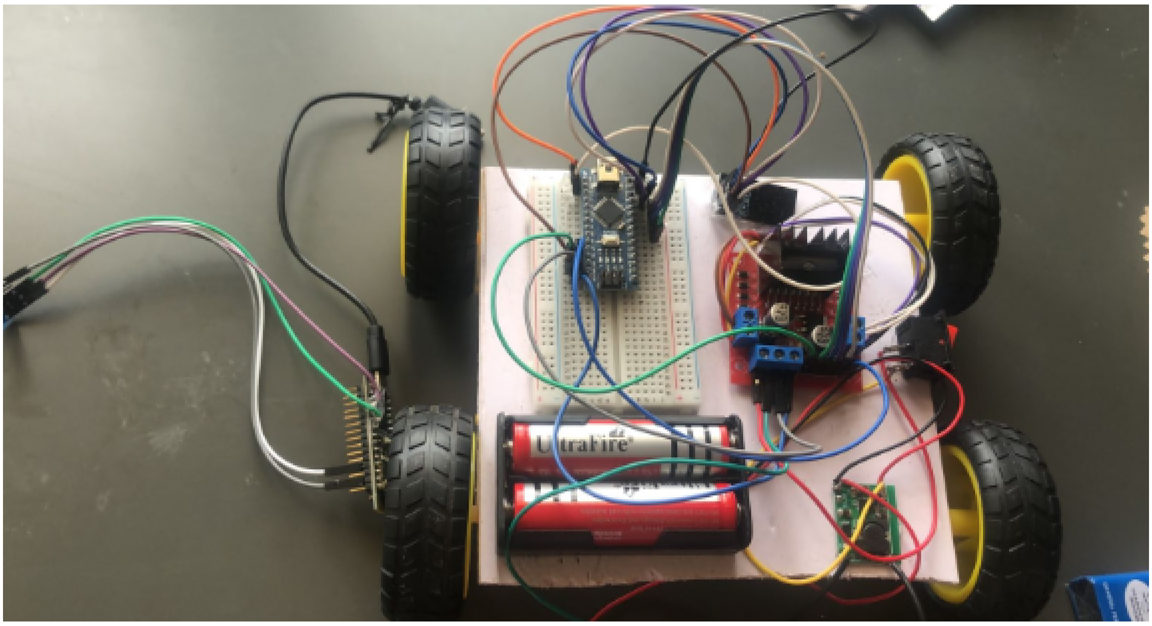


IMPLEMENTATION

For implementing the **Hand Gesture Controlled Robot Car using Arduino Nano and NRF24L01**, the project can be broken down into several steps, including setting up the transmitter (gesture sensing) and receiver (robot control) sections.

Hardware Components Required:

1. **Transmitter Section:**
    - ADXL335 Accelerometer
    - NRF24L01 RF Module (Transmitter)
2. **Receiver Section (Robot Car):**
    - Arduino Nano
    - NRF24L01 RF Module (Receiver)
    - L298N Motor Driver
    - 4 DC Motors (for driving the car)
    - Power Supply (Li-ion battery or any suitable battery pack)
    - Chassis for the robot (with wheels and motors)
    - Jumper Wires



1. Transmitter Section Setup (Gesture Sensing)

This part of the system will read the hand gestures and transmit them wirelessly to the robot. The hand gestures are captured using the ADXL335 accelerometer, which measures the tilt (acceleration) in the X, Y, and Z axes.

Code for Transmitter section:

```
#include <SPI.h>
```

```
#include <nRF24L01.h>

#include <RF24.h>


// RF24 radio object

RF24 radio(10, 9); // CE, CSN


// ADXL335 pin definitions

const int xPin = A0; // Analog pin for X axis

const int yPin = A1; // Analog pin for Y axis


// Define a struct to hold the sensor data

struct SensorData {

  int xAxis;

  int yAxis;

};


void setup() {

  Serial.begin(9600);

  radio.begin();

  radio.openWritingPipe(0xF0F0F0F0E1LL); // Set the address

  radio.setPALevel(RF24_PA_LOW); // Set power level

  radio.setDataRate(RF24_250KBPS); // Set data rate

  radio.stopListening(); // Set to transmit mode

  Serial.println("Transmitter setup complete");

}
```

```
void loop() {
 SensorData data;
 data.xAxis = analogRead(xPin); // Read X axis value
 data.yAxis = analogRead(yPin); // Read Y axis value

 bool success = radio.write(&data, sizeof(data)); // Send sensor data
 if (success) {
  Serial.println("Data sent successfully");
  Serial.print("X: ");
  Serial.print(data.xAxis);
  Serial.print(" | Y: ");
  Serial.println(data.yAxis);
 } else {
  Serial.println("Failed to send data");
 }

 delay(1000); // Wait 1 second before sending the next data
}

void loop() {
 SensorData data;
 data.xAxis = analogRead(xPin); // Read X axis value
 data.yAxis = analogRead(yPin); // Read Y axis value
```

```
  delay(1000); // Wait 1 second before sending the next data
}
```

2. Receiver Section Setup (Robot Control)

The robot car receives the gesture commands via the NRF24L01 module and controls the DC motors using the L298N Motor Driver based on the hand movements.

Code for Receiver section:

```
#include <SPI.h>
#include <nRF24L01.h>
#include <RF24.h>

// Motor pins
const int ENA = 3; // Motor A speed control
const int ENB = 8; // Motor B speed control
const int MotorA1 = 4; // Motor A direction control
const int MotorA2 = 5; // Motor A direction control
const int MotorB1 = 6; // Motor B direction control
const int MotorB2 = 7; // Motor B direction control

RF24 radio(10, 9); // CE, CSN

// Define a struct to match the data sent from the transmitter
struct SensorData {
```

```
  int xAxis;

  int yAxis;

};


SensorData receiveData;


void setup() {

  Serial.begin(9600);

  radio.begin();

  radio.openReadingPipe(1, 0xF0F0F0F0E1LL); // Address

  radio.setPALevel(RF24_PA_LOW); // Try RF24_PA_HIGH for better range

  radio.setDataRate(RF24_250KBPS); // Try RF24_1MBPS or RF24_2MBPS

  radio.startListening(); // Set to receive mode

2.

  // Set motor pins as outputs

  pinMode(ENA, OUTPUT);

  pinMode(ENB, OUTPUT);

  pinMode(MotorA1, OUTPUT);

  pinMode(MotorA2, OUTPUT);

  pinMode(MotorB1, OUTPUT);

  pinMode(MotorB2, OUTPUT);


  Serial.println("Receiver setup complete");

}
```

```
void loop() {
 if (radio.available()) {
  radio.read(&receiveData, sizeof(receiveData)); // Read data into struct


  // Process received data to control motors
  int xAxis = receiveData.xAxis;
  int yAxis = receiveData.yAxis;


  Serial.print("X Axis: ");
  Serial.print(xAxis);
  Serial.print(" | Y Axis: ");
  Serial.println(yAxis);


  // Motor control logic
  if (yAxis > 400) {
   // Move Forward
   Serial.println("Moving Forward");
   digitalWrite(MotorA1, LOW);
   digitalWrite(MotorA2, HIGH);
   digitalWrite(MotorB1, HIGH);
   digitalWrite(MotorB2, LOW);
   analogWrite(ENA, 150);
   analogWrite(ENB, 150);
  } else if (yAxis < 320) {
   // Move Backward
```

```
  Serial.println("Moving Backward");
  digitalWrite(MotorA1, HIGH);
  digitalWrite(MotorA2, LOW);
  digitalWrite(MotorB1, LOW);
  digitalWrite(MotorB2, HIGH);
  analogWrite(ENA, 150);
  analogWrite(ENB, 150);
} else if (xAxis < 320) {
  // Turn Left
  Serial.println("Turning Left");
  digitalWrite(MotorA1, HIGH);
  digitalWrite(MotorA2, LOW);
  digitalWrite(MotorB1, HIGH);
  digitalWrite(MotorB2, LOW);
  analogWrite(ENA, 150);
  analogWrite(ENB, 150);
} else if (xAxis > 400) {
  // Turn Right
  Serial.println("Turning Right");
  digitalWrite(MotorA1, LOW);
  digitalWrite(MotorA2, HIGH);
  digitalWrite(MotorB1, LOW);
  digitalWrite(MotorB2, HIGH);
  analogWrite(ENA, 150);
  analogWrite(ENB, 150);
```

```
    } else {
      // Stop
      Serial.println("Stopping");
      digitalWrite(MotorA1, LOW);
      digitalWrite(MotorA2, LOW);
      digitalWrite(MotorB1, LOW);
      digitalWrite(MotorB2, LOW);
      analogWrite(ENA, 0);
      analogWrite(ENB, 0);
    }
  } else {
    Serial.println("No data received");
  }

  delay(1000); // Short delay to avoid flooding the serial monitor
}
```

Results

```
Moving Backward
X Axis: 348 | Y Axis: 316
Moving Backward
X Axis: 313 | Y Axis: 326
Turning Left
```

```
X Axis: 375 | Y Axis: 439
Moving Forward
```

```
X Axis: 424 | Y Axis: 332
Turning Right
```

Conclusion and Future Work:

To sum up, the Hand Gesture Controller Robot Car with Arduino Nano offers a novel and practical method of utilizing hand gestures to control a robot car's movement. The system is appropriate for a number of uses, including education, entertainment, and accessibility since it is user-friendly, wireless, and accessible. However, the technology can have drawbacks such a restricted range, a lack of available hand gestures, and the requirement for calibration. Notwithstanding these drawbacks, the technology has shown promise for intuitive, hands-free robotic device operation. Subsequent investigations may go deeper into the potentialities and constraints of the system, as well as possible enhancements to augment its efficiency and intuitiveness.

In our next phase, we plan to implement a hand gesture that triggers pellet shooting, extending the system's use to military applications. Additionally, this technology could be adapted for wheelchair control, offering mobility solutions for injured individuals.

References

1)Dr. C.K. Gomathy, Mr. G. Niteesh, Mr K. Sai Krishna,"The Gesture Controlled Robot",IRJET,Volume: 08 Issue: 04 | Apr 2021

2)1Riyaz Mansuri, 2Sandesh Vakale, 3Ashish Shinde, 4Tanveer Patel,"Hand Gesture Control Robot Vehicle",IJECT Vol. 4, IssuE spl - 2, Jan - MarCh 2013
3)] Aswath S, Chinmaya Krishna Tilak, Amal Suresh and GaneshaUdupa, "Human Gesture Recognition for Real-Time Control of Humanoid Robot"

4)W.T. Freeman, C.D. Weissman, Television control by hand gesture, Proceedings of the International Workshop on Automatic Face-and Gesture-Recognition, Zurich, Switzerland, June 1995, pp. 179-183.

5)J.S. Kim, C.S. Lee, K.J. Song, B. Min, Z. Bien, Real-time hand gesture recognition for avatar motion control, Proceedings of HCI'97, February 1997, pp. 96-101.

6)Ashutosh zagade1, Vishakha Jamkhedkar2, Shraddha Dhakane3, Vrushali Patankar"A STUDY ON GESTURE CONTROL ARDIUNO

ROBOt"May 2018 IJSDR | Volume 3, Issue 5

7)Naik, G.R., Kumar, D.K., Singh, V.P., & Palaniswami, M. Hand gestures for HCI using ICA of EMG. HCSNet Workshop on the Use of Vision in HCI, 2006, 67-72.

8)Dubost, G. and Tanaka, A. A wireless, network-based biosensor interface for music, In Proceedings of International Computer Music Conference (ICMC), 2002.

*9)Anita Shukla*1, Ankit Jain2, Prajjwal Mishra3 and Rahul Kushwaha4 "Human Gesture Controlled Car Robot "S-JPSET : Vol. 11, Issue 2 2019*

*10)Mr. Kunal Appasaheb Patil*1, Mr .Laxman Baban Ardale*2, Mr. Rohit Ravsaheb Malgave*3, Mr. Shivanand Manohar Mujgonda*4, Ms. R.S. Ladag "HAND GESTURE CONTROLLED ROBOT "Volume:05/Issue:03/March-2023*

11)Milind Patwardhan Harsh Lanjewar, Harshit Rathore, Harsh Ambekar, Harsh Laddha, Harshal Sharma"Robotic Car with Wireless Hand Motion Control" VOLUME 21 : ISSUE 12 (Dec) - 2022

12)Ms. Vidhi Kalpesh Sheth,Ms. Bhakti Kiran Rathod ,Ms. Vrushali Kalpesh Sheth,Dr. Kranti Vithal Ghag "AI Power Hand Gesture Control Vehicle"Vol. 9 Issue 06, June-2020

13) Namira Khan1, Sharmin siddiqui2, Rutuj Meshram 3Shefali Meshram4,Shahbaz ahmad5, Prof. Pranali langde6, HOD Gayatri padole7 " Gesture Control Robot"July 2022 | IJIRT | Volume 9 Issue 2

14)Gaurav Kamat1Harsh Kaushik 1Praveen Kumar Singh1 RajneshKumar Singh2"Gesture Controlled Robotic Arm Using IoT" Volume 7, Issue 5, May – 2022

15)AFREEN BANO, MITUN KUNDU ,AYAN JANA, SUMAN KUMAR PAL"GESTURE CONTROL INTERFACE OF A ROBOT-CAR USING RASPBERRY PI" may 2019