GESTURE DRIVEN SMART CAR USING ARDUINO

Allu Chinni Krishna, Gongati Raghu, Gurujala Bharath, Hitesh Reddy.A
Final year B.E. Student, Final year B.E. Student, Final year B.E. Student, Final year B.E. Student

Dakshayini G.R
Assistant Professor,
Department of Computer Science,
KS School Of Engineering And Management, Bengaluru, India
alluchinnikrishna758@gmail.com, gongatiraghu56@gmail.com, bharathchowdary1560@gmail.com, hiteshreddy30@gmail.com,

ABSTRACT
This project involves designing a gesture-controlled smart car using an Arduino Nano, MPU6050 sensor, L298N motor driver, and DC motors. The MPU6050 detects hand gestures, which are processed by the Arduino to control the car's motion (forward, backward, left, right, and stop). This intuitive system offers hands-free operation, with applications in robotics, assistive technology, and interactive demonstrations. It showcases Arduino-based robotics and sensor integration, with potential for wireless or autonomous upgrades.

Keywords: Gesture-Controlled Smart Car, Arduino Nano, MPU6050 Sensor, L298N Motor Driver, DC Motors, Robotics, Sensor Integration, Hands-Free Control, Assistive Technology, Interactive Systems.

I. INTRODUCTION
This project presents the design and implementation of a gesture-driven smart car controlled through hand movements. The system employs an Arduino microcontroller paired with an MPU6050 accelerometer and gyroscope sensor to detect and interpret gestures. By tilting or moving the sensor, users can command the car to move forward, backward, turn left or right, or stop. The MPU6050 measures acceleration and angular velocity in three axes (X, Y, and Z), and the Arduino processes this data to control the car's DC motors via an L298N motor driver. This gesture-based approach provides hands-free operation, making it suitable for individuals with limited mobility or in situations where traditional control devices are impractical. The intuitive nature of gesture control simplifies the user experience, while the system's interactivity enhances its applicability in robotics, education, entertainment, and assistive technologies.

II. METHODOLOGY
The methodology employed in the Smart Cart System focuses on the integration of various hardware components, real-time data synchronization, and seamless user interaction to provide an efficient and automated shopping experience. The system operates based on the following key steps, ensuring smooth cart movement, item scanning, and checkout, while maintaining real-time synchronization with the cart’s website for user convenience.

Step 1: Accelerometers on the user's hand detect motion and orientation

Accelerometers are mounted on the user’s hand to sense the movement and tilt. These sensors measure the changes in acceleration due to motion or changes in position, converting the physical movement into electrical signals.

Step 2: The Arduino Nano processes the accelerometer data into control commands

The Arduino Nano receives the raw data from the accelerometers. It processes the signals to interpret the direction, speed, and type of motion, converting these into meaningful control commands.

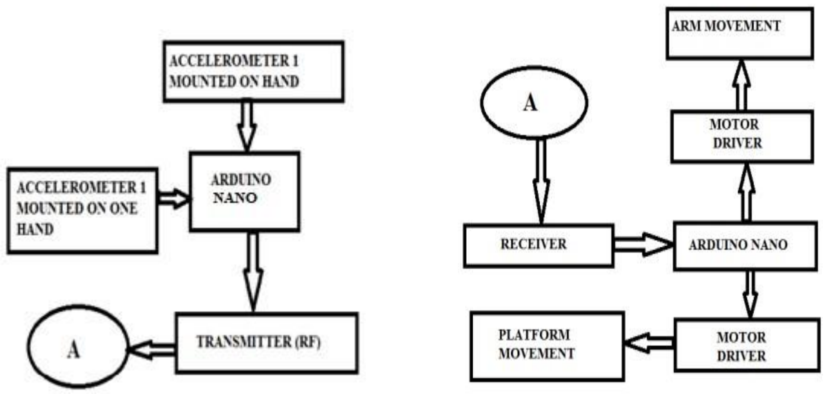
Step 3: An RF transmitter sends these commands wirelessly to the receiving system

Once the commands are generated, they are transmitted wirelessly using an RF transmitter. This enables communication between the controlling unit (user's hand) and the receiver, ensuring seamless transfer of motion data.

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 05 | May - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

Step 4: The RF receiver captures the transmitted commands
On the receiving end, the RF receiver collects the wireless signals sent by the transmitter. It ensures the commands are accurately captured without signal loss or interference.

Step 5: The Arduino Nano processes the received data and generates motor control signals
The Arduino Nano decodes the received signals and translates them into control signals that are understandable by the motor drivers. It serves as the intermediary between the RF receiver and the actuators. **Step 6: Motor drivers execute the signals to control the robotic arm and platform movements**
The motor drivers take the control signals from the Arduino Nano and drive the motors accordingly. These motors control both the robotic arm's movement (based on the user's hand gestures) and the platform's movement (for locomotion).
Step 7: The system allows real-time, gesture-based control of the robotic arm and platform
By integrating accelerometers, wireless communication, and motor control, the system enables the user to intuitively control a robotic arm and platform in real-time using natural hand gestures. This setup has applications in remote operations, rehabilitation devices, and industrial automation.



III. BLOCK DIAGRAM

Figure 2: Block Diagram

This system is designed for wireless motion-based control of motors, commonly used in robotics or remotecontrolled devices. The MPU6050 sensor module, which combines an accelerometer and gyroscope, collects motion and orientation data. This data is processed by the first Arduino Nano, which acts as the controller for interpreting the sensor readings. The processed information is then transmitted wirelessly using an nRF transmitter module. On the receiving end, the nRF receiver passes the data to a second Arduino Nano, which interprets the incoming signals and generates appropriate commands for the motor driver (L293D). The motor driver acts as the interface between the second Arduino and the motors, enabling precise control of motor operations such as speed and direction. This system facilitates a seamless and efficient way to control motors based on real-time motion inputs without requiring wired connections.

IV. WORKING

This system demonstrates a hand-gesture-controlled motorized car using an accelerometer, Arduino Nano, RF modules, and a motor driver. It starts with an accelerometer detecting hand gestures and sending motion and orientation data to an Arduino Nano. The Arduino processes this data and converts it into angle values ranging from 0° to 450°, representing different movement directions. These processed signals are then transmitted wirelessly using an RF transmitter. On the receiving end, an RF receiver collects the data and sends it to another Arduino Nano for further analysis. The second Arduino checks the angle ranges to determine the corresponding movement commands, such as forward, backward, left, or right, based on pre-defined angle thresholds. The appropriate command is then sent to a motor driver (L293D), which controls the car's motors to execute the movement. This setup enables wireless, intuitive control of the car's movement using simple hand gestures detected by the accelerometer.

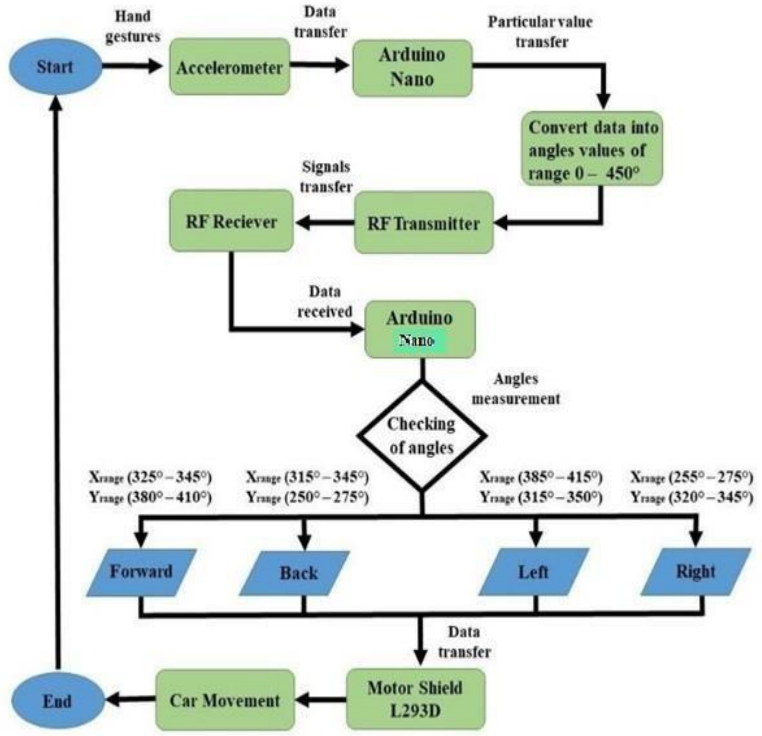


Figure 3: Flow Chart (a)

V. DESIGN AND CIRCUIT DIAGRAM

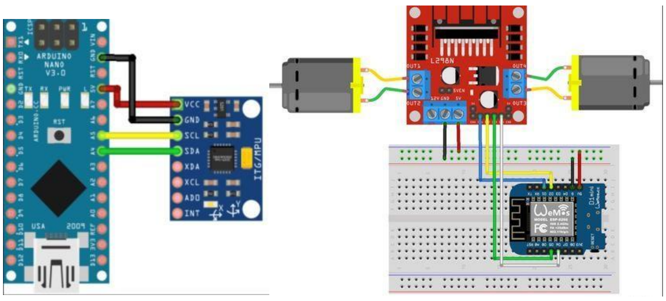


Figure 5: Circuit Diagram

VI. RESULT

Gesture Recognition: The MPU-6050 accurately detects hand movements, such as tilting forward, backward, left, or right, which the Arduino Nano processes to identify specific gestures.

Real-Time Communication: The WeMos D1 Mini modules enable seamless wireless communication with minimal latency for near-instantaneous responses.

Car Movement: The car responds to gestures effectively, with actions like moving forward, reversing, or turning left or right based on hand tilts.

Power Efficiency: The system operates efficiently on battery power, with the motor driver managing high current demands and microcontrollers running on low voltage.

Ease of Use: Gesture-based control simplifies operation, allowing users to navigate the car effortlessly with hand movements.

Stability and Robustness: The system ensures stability and precise control even in varying environmental conditions due to the MPU-6050's sensitivity.

Range and Responsiveness: The wireless system functions effectively within a 30-meter range, ensuring prompt motor responses for smooth operation.

VII. CONCLUSION

The Gesture-Driven Smart Car Using Arduino revolutionizes vehicle control by leveraging advanced gesture recognition for a seamless and intuitive driving experience. It enables touchless operation of essential functions like headlights and entertainment systems, enhancing safety and convenience. By reducing distractions, it promotes a safer driving environment. Future advancements, such as machine learning and integration with autonomous technologies, will further enhance accuracy and personalization. This innovation sets a new standard for vehicle interaction, paving the way for more intuitive and efficient automotive technology.

VIII. REFERENCES

[1] Xing-Han Wu, Mu-Chun Su, Pa-Chun Wang, "A Hand-Gesture-Based Control Interface for a Car-Robot" IEEE Research paper, 2010.

[2] "Accelerometer-Based Hand Gesture Control Robot Using Arduino and 3-Axis Accelerometer" – N. M. Sundaram, R. S. R. Krishnan, T. S. Kamal, J. Rajeswari, 2018

[3] "Arduino-based Wireless Hand Gesture Controlled Robot" - P. Chanda, P. Mukherjee, S. Modak, A. Nath, 2016

[4] Dr Sachin Kumar Gupta January 2023 - Arduino-based wireless hand gesture controlled robot

[5] C K Gomathy April 2021 - The Gesture Controlled Robot