International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14, Issue 05, IIRA 5.0 (2026)

Smart Accesibility Device using Raspberry Pi

Kavyansh Tyagi
Master of Computer Applications Ajay Kumar Garg Engineering College
Ghaziabad , India
kavyanshtyagi222@gmail.com

Shakshi Deshwal
Master of Computer Applications Ajay Kumar Garg Engineering College
Ghaziabad , India
sakshi2314157@akgec.ac.in

Yash Keshri
Master of Computer Applications Ajay Kumar Garg Engineering College Ghaziabad , India
theyashkeshri@gmail.com

Aman Gupta
Assistant Professor
Ajay Kumar Garg Engineering College Ghaziabad,India
guptaaman@akgec.ac.in

Abstract - ***Assistive technologies play a vital role in enhancing the quality of life for individuals with physical impairments by providing support and improving daily functionality. This paper discusses the design and development of a Wi-Fi-controlled robot car that utilizes the Raspberry Pi Pico microcontroller and the Raspberry Pi function together, with the Raspberry Pi serving as the central control unit. processing unit, efficiently managing communication between the user interface and the various hardware components of the car. Additionally, the user interface is well-designed as a web application, allowing the user to navigate the car effortlessly. Users can also monitor a live video feed directly from the camera. The robot car offers features such as real-time navigation, obstacle detection, and optional surveillance capabilities, targeting applications in accessibility and mobility assistance. The proposed system provides remote control, flexibility, and user-friendly operation. Key features include obstacle avoidance, voice-based control, and optional surveillance capabilities, making it suitable for various applications. This innovative approach demonstrates the potential for integrating advanced technologies to create versatile and cost-effective solutions for assistive mobility.***

Keywords

Wi-Fi Controlled Car, Raspberry Pi, Real-Time Video Streaming, Web Application, Surveillance and Exploration.

1. Introduction

The demand for assistive technologies has surged in recent years, driven by the growing need to enhance the quality of life for individuals with mobility impairments. Wheelchair users, in particular, often face significant challenges in navigating their environments independently. Despite the availability of various assistive devices, many existing solutions are hindered by high costs, limited scalability, and lack of adaptability to diverse settings. These limitations highlight the need for innovative and affordable technologies that can serve a diverse range of users and settings.

Individuals who use wheelchairs often encounter challenges related to mobility assistance. Many assistive technologies are hindered by their high costs, limited scalability, and lack of adaptability. This highlights the need for innovative, low-cost technologies that can cater to the diverse needs of different users and environments.

This research involves the development of a Wi-Fi-operated robot car designed to improve accessibility for individuals with mobility disabilities. The robot car is controlled remotely using a Raspberry Pi, which can be operated via a smartphone or computer. The web application interface provides simple controls for movement and other functions.

The Wi-Fi-enabled robot car is equipped with obstacle avoidance, allowing it to safely navigate around and detect obstacles. It can be voice-controlled for users with limited dexterity and includes an option for monitoring via real-time video streaming. The Wi-Fi-enabled robot car system integrates advanced features to enhance existing assistive technologies' efficiency. Its open-source and modular design ensures cost-

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14, Issue 05, IIRA 5.0 (2026)

effectiveness and scalability for larger populations.

2. Literature Review

Zaini et al. [1] provides a remote monitoring system using a Wi-Fi-powered RC car and Raspberry Pi and a USB camera. The system seeks to address challenges in surveillance, particularly in huge buildings, dangerous zones, and inaccessible areas.

The study by Yılmaz and Özyer et al. [2] presents a remote-control and autonomous robot car that uses an Arduino Uno, an Android-based control system, and real-time obstacle avoidance and detection systems. The system uses Bluetooth technology for communication between the robot and the Android application to allow users to control the robotic car in a manual mode or transition to an autonomous mode, where the car navigates on its own while avoiding obstacles.

The study conducted by Gule and Ohrun et al. [3] investigates a Wi-Fi remote-controlled car using a Raspberry Pi 3, an L298N motor driver, and an Android remote control application. Unlike traditional wired robotic systems, this study focuses on enhancing mobility and user control through wireless communication supported by Wi-Fi technology.

Shashidhar and Tippannavar et al. [4] offer a smart wheelchair, operated through a Raspberry Pi, that leverages EEG technology specifically designed for paralyzed and elderly people. The system employs Brain-Computer Interface (BCI) technology to decode EEG signals from a wireless headband and enables the paralyzed individuals to control the wheelchair by eye blinks and facial muscle movements. The innovation significantly increases mobility and independence for paralyzed individuals with extreme motor disabilities.

3. System Architecture

The overall system can be subdivided into the software model, hardware model and system workflow. The section will discuss these in detail.

A. Software Model

The software program version is a essential element of the system, ensuring seamless interplay

among hardware and software program additives. It is designed to offer efficient actual-time processing, user-pleasant get entry to, and dependable conversation among specific device factors. The software program model includes several key additives:

i. Micro Python Firmware

- Micro Python firmware runs at the Raspberry Pi Pico, serving as the centre software program layer liable for coping with hardware operations and peripheral communique. This firmware allows the microcontroller to execute essential responsibilities such as managing motor controls, sensor integration, and command processing.
- Micro Python is lightweight and optimized for microcontrollers, making it a really perfect choice for jogging efficiently at the Raspberry Pi Pico whilst enabling clean improvement and debugging.

ii. Web Interface

- The web interface acts as the user's manipulate panel for interacting with the device. Designed to be reachable from any tool with a web browser or through a devoted mobile app, it affords an intuitive and responsive revel in for customers.
- The web interface capabilities a dashboard that presentations real-time sensor readings, and controls for the device. Users can control movement, modify settings, and track the sensor data in a centralized location.
- The web interface is built with the use of current web technologies like HTML, CSS, and JavaScript.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14, Issue 05, IIRA 5.0 (2026)

iii. Communication Protocol

- A custom conversation protocol is applied using Wi-Fi and a WebSocket server to ensure low-latency, bidirectional communique among the net interface and the Raspberry Pi Pico.
- The ESP8266 Wi-Fi Module establishes a direct Wi-Fi link between the Raspberry Pi Pico and the user device, enabling a secure and seamless bi-directional communication system.
- A WebSocket server is hosted on the Raspberry Pi to facilitate continuous, real-time communication between the user device and hardware.
- Unlike traditional methods like HTTP, Web Sockets keep a non-stop connection, decreasing latency and improving responsiveness.
- This permits instantaneous command execution and instant comments transmission.

B. Hardware Model

The hardware components of the robotic automobile play an important position in ensuring easy capability, unique manipulate, and efficient verbal exchange. The machine accommodates several key modules, every performing a selected task to make contributions to the general overall performance of the robotic. Below is an in-depth breakdown of the vital hardware components:

i. Raspberry Pi Pico:

- The Raspberry Pi Pico serves as the central microcontroller responsible for processing commands and controlling peripheral components.
- It interfaces with sensors, motors, and communication modules through GPIO pins.
- The Raspberry Pico features a RP2040 microprocessor, providing two ARM Cortex-M0+ processor, enabling efficient and fast multitasking.

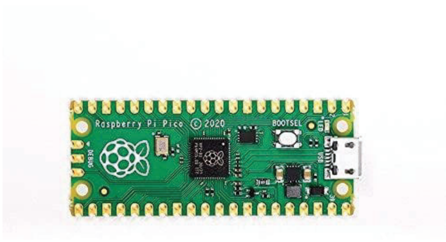

Figure 1. Raspberry Pi Pico

ii. ESP8266 Wi-Fi Module:

- ESP8266 Wi-Fi module is an essential component to enable wireless communication between robot car and user interfaces.
- This module ensures real-time remote control and feedback transmission on a stable Wi-Fi network.
- The ESP8266 is an affordable solution with excellent processing capabilities, enabling reliable connectivity without significantly increasing power consumption.

Figure 2. ESP8266 Wi-Fi Module

iii. L298N Motor Driver:

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14, Issue 05, IIRA 5.0 (2026)

- The L298N Motor Driver is responsible for controlling the movement of the robot's wheels by managing the speed and direction of DC motors.
    - The L298N can handle a significant amount of current, making it suitable for driving motors in robotic applications.

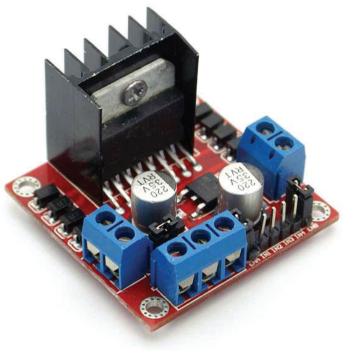

Figure 3. L298N Motor Driver

iv. Rechargeable Battery:

- It powers all components of the robot, ensuring portability and extended operation.
- The battery provides stable voltage levels required for microcontroller operation, motor movement, and wireless communication.
- Rechargeable batteries reduce operational costs and are environmentally friendly.
- The battery chosen for this project is rated for 1500 mAh at 3.7V.

Figure 4. Rechargeable Battery (1500mAh at 3.7V)

C. System Workflow

The user accesses the control interface via a web browser or mobile app. The navigation commands are sent over Wi-Fi to the ESP8266 module. The Raspberry Pi Pico processes these commands and drives the motor controller for movement.

The system workflow of the developed device using various components is shown in Figure 1.

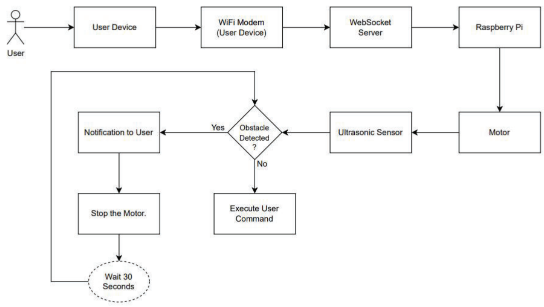

Figure 5. System Workflow of the Device

The device and the user communicate with each other with the help of Wi-Fi technology. The user sends navigation commands via web interface. The commands are transmitted to the device through the WebSocket server. The Raspberry Pi then processes the commands and send operational commands to the motors through the GPIO pins. The sensors relay real-time data back to the Raspberry Pi which then avoids the obstacles dynamically.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14, Issue 05, IIRA 5.0 (2026)

4. Implementation

A. Hardware Integration

The following hardware was chosen to create a compact and efficient working system:

1. Raspberry Pi: It is the brains of the device, managing all Input-Output operations. The GPIO pins are used to connect sensors, motor driver and other components.
2. Ultrasonic Sensor: Ultrasonic Sensors are placed at the front of the device for obstacle detection.
3. Motor Driver: Motor is connected to the Raspberry Pi to control the movement of the device. The Raspberry Pi varies the power going to the motor to control the speed of the device.
4. Wheels: The front wheels have a turn functionality to change the direction of movement of the device.
5. Power Supply: The entire system is powered by a rechargeable battery pack. The battery pack was chosen ensuring sufficient voltage and runtime for the device.

B. Software Development

The software of the system consists of multiple layers to handle user commands, feedback from sensors and the user interface integration.

1. Raspberry Pi Firmware: A custom Micro Python code was written to handle motor control, process sensor data and communication via Wi-Fi.
2. WebSocket Server: A custom WebSocket runs locally on the Raspberry Pi to ensure a connection between the user interface and the device. It is used to transmit user commands and feedback.
3. User Interface: A simple web interface was developed using HTML, CSS and JavaScript. The users can access this interface from any device (smartphone, tablet, PC) as long as both devices are connected to the same Wi-Fi network.
4. Ultrasonic Sensors: An algorithm was used to read sensor data and process it to detect any obstacles.
   When an obstacle is detected within a predefined range, the device halts all movement and sends a notification to the user via the interface.

5. Challenges and Solution

1. Battery Optimisation: The system load changes drastically upon usage in different scenarios, causing the battery pack to discharge quicker. Various optimisation techniques can be used to maximize runtime for the device.
2. Network Latency: In situations where the Wi-Fi network used was crowded there was an increase in latency. This was minimised by limiting the number of devices on the network.
3. Obstacle Detection: Since the obstacle detection works on a predefined range, the sensor can give false data when the device is operated in a confined space. This was solved by changing the range parameters in the obstacle detection algorithms.

6. Conclusion

The Smart Accessibility Device using Raspberry Pi demonstrates the potential of IoT and Robotics in enhancing the accessibility of individuals with disabilities, particularly those in wheelchairs. The device build using off-the-shelf components like Raspberry Pi, offer remote control, obstacle detection enabling users to navigate their surrounding areas with ease and safety. For a person with disabilities, this serves as a helpful companion, assisting in tasks which normally would be more effort consuming like exploring inaccessible spaces, therefore reducing reliance on caregivers. It’s simple and affordable design make it easy and user-friendly. Future Enhancements like AI navigation

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14, Issue 05, IIRA 5.0 (2026)

and voice control could expand the features and benefits of this device.

7. References

1. Nazirah Ahmad Zaini, Norliza Zaini, Md. Fuad Abdul Latip, Nabilah Hamzah - ***"Remote Monitoring System based on a Wi-Fi Controlled Car Using Raspberry Pi"*** 2016 IEEE Conference on Systems, Process and Control (ICSPC 2016).

2. Esra Yılmaz, Sibel T. Özyer - ***"Remote and Autonomous Controlled Robotic Car based on Arduino with Real Time Obstacle Detection and Avoidance"*** Universal Journal of Engineering Science, 2019.

3. Mehmetcan Güleç'i, Murat Orhun – ***"Android-based Wi-Fi Controlled Robot Car using Raspberry Pi"***

4. Chandana, Dr. S. A. K. Jilani, Mr. S. Javeed Hussain, et al. *"* ***Smart Surveillance System using Thing Speak and Raspberry Pi*** *"* *IEEE Communications Surveys & Tutorials*, vol. 17, no. 4, 2015.

5. Shashidhar R., Sanjay S. Tippannavar., et al. ***"EEG-based Smart Wheelchair Using Raspberry Pi for Elderly and Paralysed Patients,"*** IEEE 2nd Mysore Subsection International Conference (MysuruCon), 2022..

6. Varsha Vishwa Kiran, S Santhanalakshmi, et al. " **Raspberry Pi based Remote Controlled Car using Smartphone Accelerometer** " F*ourth International Conference on Communication and Electronics Systems*, 2019.