International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 10 Issue: 04 | April - 2026 SJIF Rating: 8.659 ISSN: 2582-3930

Development of a Web-Based Remote-Controlled Robotic System Embedded Platforms

K. Sai SivaJyothi[1], Ch. Divya [2], D. Jagannadha rao[3], M. Ramya[4], K. Mani varaha deepak[5], Dr. B. RaviChandra[6]
[1,2,3,4,5]B.Tech Students, Department of Electronics and Communication Engineering,
[6]Associate Professor, Department of Electronics and Communication Engineering,
Nadimpalli Satyanarayana Raju Institute of Technology

Abstract**—**This paper presents the design and development of a web-based remote-controlled robotic system using embedded platforms. The proposed system enables users to control and monitor a robotic platform in real time through a web interface accessible via standard browsers. The system integrates embed- ded devices such as the Raspberry Pi and Arduino, where the Raspberry Pi handles high-level processing, communication, and web server functionalities, while the Arduino manages real-time control of motors and sensors.

The web-based interface allows users to send control com- mands and receive feedback over a network using HTTP and Wi-Fi protocols. The system supports real-time operation with minimal latency, ensuring smooth and responsive control of the robotic platform. Experimental results demonstrate reliable com- munication, efficient command execution, and stable performance under different operating conditions.

The proposed system is cost-effective, scalable, and easy to implement, making it suitable for applications such as surveil- lance, industrial automation, and remote monitoring. Future enhancements can include integration of artificial intelligence, video streaming, and IoT-based monitoring to further improve system functionality and intelligence.

I. Introduction

The rapid advancement of embedded systems, robotics, and web technologies has enabled the development of intelligent systems capable of remote monitoring and control. In recent years, there has been a growing demand for robotic sys- tems that can be operated from distant locations, especially in applications such as surveillance, industrial automation, and hazardous environment exploration. Traditional robotic systems often require manual control or dedicated hardware interfaces, which limit flexibility and accessibility.

To overcome these limitations, web-based remote-controlled robotic systems have emerged as an effective solution. These systems utilize web technologies to provide a user-friendly interface that can be accessed through standard web browsers, allowing users to control robotic platforms from anywhere over a network. The integration of embedded platforms such as Raspberry Pi and Arduino further enhances system per- formance by combining high-level processing with real-time control capabilities.

In such systems, the embedded platform acts as the core processing unit, handling communication, data processing, and control logic. The web interface enables users to send commands and receive feedback in real time, ensuring efficient operation. The use of wireless communication technologies such as Wi-Fi allows seamless data transmission between the user and the robotic system.

The proposed system focuses on the design and devel- opment of a web-based robotic platform that enables real- time control and monitoring using embedded systems. This approach offers advantages such as remote accessibility, cost-effectiveness, scalability, and ease of implementation. The system can be applied in various domains, including security surveillance, smart home automation, and industrial inspec- tion.

This paper aims to develop a reliable and efficient web-based robotic system that leverages embedded platforms for real-time operation, providing a flexible and practical solution for modern robotic applications.

II. Literature Survey

Web-based remote-controlled robotic systems have gained significant attention due to their ability to provide real-time monitoring and control through internet technologies [1], [2]. Early robotic systems were limited to wired or short-range communication, which restricted their usability in remote applications. With the advancement of Internet of Things (IoT) and wireless communication technologies, robotic systems can now be controlled from anywhere using web-based interfaces [3], [4].

Embedded platforms such as Raspberry Pi and Arduino have been widely used in the development of such systems due to their low cost, flexibility, and ease of integration [5], [6]. The Raspberry Pi is commonly used for handling web server functionalities, data processing, and networking, while Arduino is used for real-time control of motors and sensors [7], [8]. This combination enhances system performance by efficiently distributing tasks between processing and control units.

Several researchers have developed web-controlled robotic systems for applications such as surveillance, home automa- tion, and industrial monitoring [9], [10]. These systems utilize HTTP protocols and Wi-Fi communication to enable real-time command transmission and feedback. The use of web-based interfaces eliminates the need for specialized applications, improving accessibility and user convenience.

Recent studies have also focused on integrating video streaming and computer vision into robotic systems to enhance monitoring capabilities [11], [12]. Live video feedback allows users to observe the environment remotely and make informed decisions. Additionally, the incorporation of cloud computing and IoT platforms has improved system scalability and data management [13], [14].

Advanced research includes the use of machine learning and artificial intelligence to enable autonomous decision-making and intelligent control [15], [16]. These technologies allow robotic systems to perform tasks such as object detection, path planning, and environment analysis.

Despite these advancements, challenges such as network latency, security issues, and reliability of communication remain critical concerns [17], [18]. Researchers are working on improving communication protocols, optimizing system architecture, and enhancing security mechanisms.

Ongoing developments in edge computing, 5G communication, and smart embedded systems are expected to further improve the performance and efficiency of web-based robotic systems, making them more suitable for real-world applications [19], [20].

III. Proposed Methodology

The proposed system is developed to enable real-time remote control of a robotic platform using web technologies and embedded systems. The methodology focuses on integrating hardware and software components to achieve efficient communication, processing, and control.

A. System Design

The system is designed with a modular architecture that combines a web-based interface with embedded platforms. The Raspberry Pi acts as the central processing unit responsible for communication, data processing, and hosting the web server, while the Arduino handles real-time control of motors and sensors. This design ensures efficient task distribution and improved system performance.

B. Component Integration

The hardware components are integrated to form a complete robotic system. The Raspberry Pi is connected to the Arduino through serial communication, enabling data exchange between the two units. The Arduino is interfaced with the motor driver, which controls the movement of the robot. Additional sensors can be integrated to provide environmental feedback. Proper wiring and configuration ensure reliable system operation.

C. Web Interface Implementation

A web-based interface is developed using HTML, CSS, and JavaScript, with Flask used as the backend framework. The interface allows users to send control commands and monitor system behavior in real time. The web server running on the Raspberry Pi processes user inputs and communicates with the Arduino to execute corresponding actions.

D. Communication Mechanism

The communication between the user interface and the robotic system is established using HTTP and Wi-Fi protocols. Commands from the user are transmitted over the network to the Raspberry Pi, which processes and forwards them to the Arduino via serial communication. This ensures real-time control with minimal delay.

E. Control Implementation

The control mechanism involves processing user commands on the Raspberry Pi and transmitting them to the Arduino. The Arduino generates control signals to drive the motor driver, enabling the robot to move in different directions. The system ensures smooth and accurate movement through efficient coordination between processing and control units.

F. Testing and Evaluation

The system is tested under various operating conditions to evaluate performance parameters such as response time, communication reliability, and control accuracy. The results confirm that the system operates efficiently with low latency and stable performance, making it suitable for real-time applications.

IV. System Architecture

The system architecture of the proposed web-based remote-controlled robotic system is designed to enable real-time monitoring and control through efficient integration of web technologies and embedded platforms. The architecture consists of interconnected modules including the user interface, communication layer, processing unit, control unit, and robotic platform. This structured design ensures seamless interaction between hardware and software components for reliable system operation.

A. User Interface

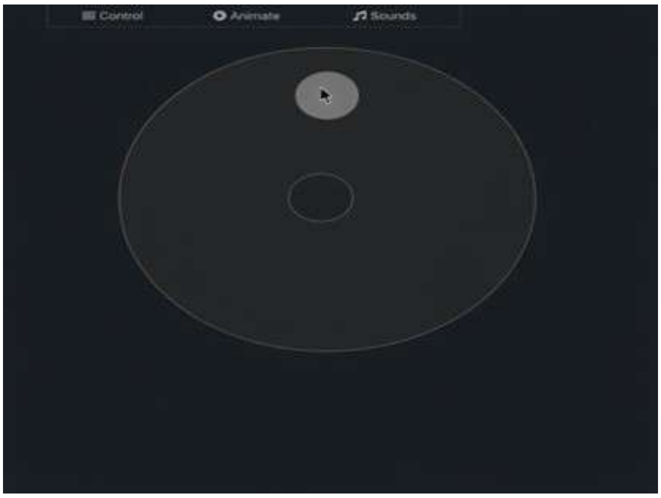

Fig. 1. Web-based control interface with virtual joystick for robotic navigation

The user interface serves as the primary point of interaction between the user and the robotic system. It is developed using web technologies and can be accessed through standard web browsers on devices such as laptops, smartphones, or tablets. The interface allows users to send control commands and monitor system status in real time, ensuring ease of use and accessibility from remote locations.

B. Communication Layer

The communication layer facilitates data exchange between the user interface and the robotic system. It utilizes HTTP and Wi-Fi protocols to transmit control commands and feedback over the network. This layer ensures real-time communication and enables remote operation of the system with minimal delay.

C. Processing Unit (Raspberry Pi)

The Raspberry Pi acts as the central processing unit of the system. It hosts the web server, processes incoming user commands, and manages communication with the Arduino. It also handles data processing and can support additional functionalities such as video streaming and system monitoring.

D. Control Unit (Arduino)

The Arduino microcontroller functions as the control unit responsible for executing hardware-level operations. It receives commands from the Raspberry Pi via serial communication and generates appropriate control signals for the motor driver and sensors. This ensures precise and real-time control of the robotic platform.

E. Motor Driver

The motor driver acts as an interface between the Arduino and the motors. It regulates the speed and direction of the motors based on control signals, enabling the robot to perform movements such as forward, backward, and turning operations.

F. Robotic Platform

The robotic platform consists of motors, sensors, a camera module (optional), and a power supply system. It performs physical operations and interacts with the environment. Sensors provide feedback that can be transmitted back to the Raspberry Pi for monitoring and further processing.

G. System Operation

The system operates through coordinated interaction among all components. The user sends commands through the web interface, which are transmitted via the communication layer to the Raspberry Pi. The Raspberry Pi processes the commands and forwards them to the Arduino. The Arduino controls the motors and sensors, enabling the robot to perform the desired actions. Feedback from the system is sent back to the user interface, ensuring continuous monitoring and control.

Fig. 2. Block Diagram of Web-Based Remote-Controlled Robotic System Using Embedded Platforms

V. Block Diagram

The block diagram illustrates the architecture of the proposed web-based remote-controlled robotic system. The system begins with the web interface, where the user sends control commands through a browser. These commands are transmitted via the communication layer using HTTP and Wi- Fi protocols to the Raspberry Pi, which acts as the processing unit.

The Raspberry Pi processes the received commands and forwards them to the Arduino through serial communication. The Arduino functions as the control unit, generating appropriate signals for the motor driver, which controls the movement of the robotic platform. The robot performs actions such as forward, backward, and directional movement based on the commands received.

Additionally, the system supports feedback mechanisms, where sensor data from the robotic platform is sent back to the Raspberry Pi and displayed on the web interface for monitoring. This ensures real-time control and continuous interaction between the user and the robotic system.

VI. Working Principle

The working principle of the proposed web-based remote-controlled robotic system is based on real-time interaction between the user interface, embedded processing unit, communication layer, and the robotic platform. The system converts user inputs into control signals and executes them through coordinated hardware and software operations.

A. Command Input and Transmission

The operation begins when the user accesses the web-based interface through a browser and sends control commands such as movement directions. These commands are transmitted over the network using HTTP and Wi-Fi protocols to the Raspberry Pi, enabling remote operation of the system.

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 10 Issue: 04 | April - 2026 SJIF Rating: 8.659 ISSN: 2582-3930

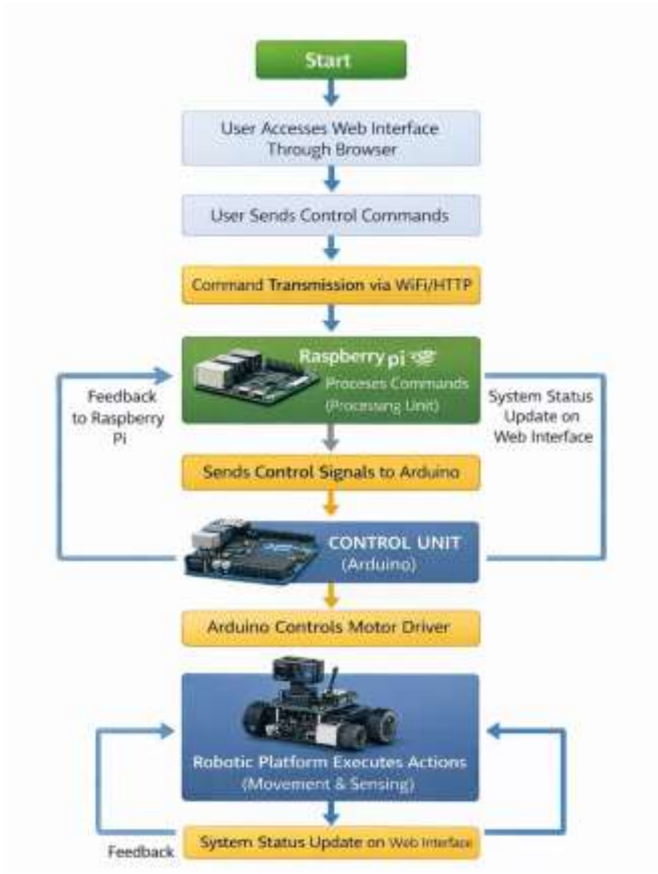

Fig. 3. Flowchart of Web-Based Remote-Controlled Robotic System Operation

B. Command Processing

The Raspberry Pi receives the user commands and processes them as the central control unit. It interprets the inputs and converts them into appropriate control signals. The processed commands are then transmitted to the Arduino microcontroller through serial communication.

C. Control Execution

The Arduino receives the control signals and executes them by generating appropriate outputs for the motor driver. The motor driver regulates the speed and direction of the motors, allowing the robotic platform to move according to user commands. This ensures accurate and real-time control of the system.

D. Feedback Mechanism

The system supports feedback through sensors and monitoring data. Sensor information can be sent back to the Raspberry Pi and displayed on the web interface, allowing users to observe system behavior and make necessary adjustments.

E. Real-Time Operation

The continuous interaction between the user interface, Raspberry Pi, Arduino, and robotic platform ensures real- time operation. The system maintains synchronization between command input, processing, and execution, resulting in minimal delay and efficient performance.

VII. RESULTS

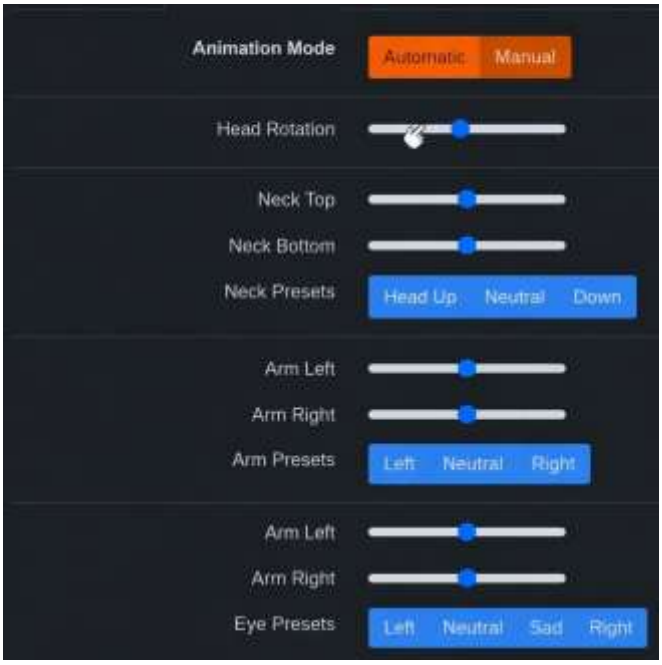

Fig. 4. Web-Based Robotic Control Interface with Real-Time Servo and Preset Controls

The developed web-based remote-controlled robotic system demonstrates efficient real-time performance with reliable communication over a wireless network. As shown above, the user interface provides multiple control options, including automatic and manual modes, slider-based servo adjustments, and predefined motion presets for different robotic components such as head, neck, arms, and eyes.

The system successfully executes user commands with minimal latency, ensuring smooth and precise movement of the robotic platform. The integration of Raspberry Pi and Arduino enables effective task distribution, where high-level processing and communication are handled by the Raspberry Pi, while real-time hardware control is managed by the Arduino.

Experimental observations indicate stable system performance under various operating conditions, with consistent and accurate responses to user inputs. The use of HTTP protocol over Wi-Fi ensures efficient data transmission, contributing to low communication delay and improved responsiveness.

Furthermore, the interface enhances user interaction by providing intuitive controls such as sliders and preset buttons, allowing easy manipulation of multiple degrees of freedom in the robot. This improves usability and reduces control complexity.

Overall, the results confirm that the proposed system is a cost-effective, scalable, and reliable solution for real-time remote robotic control and surveillance applications.

VIII. Conclusion

This paper presented the design and development of a web- based remote-controlled robotic system using embedded plat- forms. The system enables real-time control and monitoring through a web interface, ensuring ease of access and flexibility. The integration of Raspberry Pi and Arduino provides efficient processing and reliable hardware control, resulting in smooth and accurate system operation.

The experimental results demonstrate that the system achieves low latency, stable performance, and effective com- munication over the network. The proposed solution is cost-effective, scalable, and suitable for applications such as surveillance, automation, and remote monitoring. Overall, the system offers a practical and efficient approach for modern robotic control applications.

Acknowledgment

We would like to thank Design Space lab, Department of Electronics and Communication Engineering at Nadimpalli Satyanarayana Raju Institute of Technology (NSRIT) for pro- viding the resources and support required for this research work.

References

[1] A. Bahga and V. Madisetti, *Internet of Things: A Hands-On Approach*, 2015.
[2] O. Vermesan and P. Friess, *IoT Technologies*, 2013.
[3] K. Ashton, "Internet of Things," RFID Journal, 2009.
[4] D. Evans, "The Internet of Things," Cisco, 2011.
[5] E. Upton and G. Halfacree, *Raspberry Pi User Guide*, 2014.
[6] M. Margolis, *Arduino Cookbook*, 2011.
[7] S. Monk, *Programming the Raspberry Pi*, 2016.
[8] J. Igoe, *Making Things Talk*, 2014.
[9] A. Raj and A. Raman, "Web-Based Control Systems," IJERT, 2018.
[10] S. Saha *et al.*, "Web Controlled Robot," IEEE, 2019.
[11] R. Szeliski, *Computer Vision*, 2010.
[12] T. Bouwmans *et al.*, *Video Surveillance Techniques*, 2014.
[13] M. Collotta and G. Pau, "IoT Smart Systems," *IEEE Internet of Things Journal*, 2015.
[14] A. Sehgal and N. Kumar, *Embedded Systems for IoT*, 2019.
[15] I. Goodfellow *et al.*, *Deep Learning*, 2016.
[16] Y. LeCun *et al.*, "Deep Learning," *Nature*, 2015.
[17] "Security in IoT Systems," *IEEE Transactions*, 2018.
[18] "Real-Time Communication Systems," *IEEE*, 2017.
[19] W. Shi *et al.*, "Edge Computing," *IEEE Internet of Things Journal*, 2016.
[20] "5G and IoT Integration," Elsevier, 2020.