International Journal of Intelligent Communication and Computer Science
Vol. 3, No. 2, 2025, Pages 57-66.

Real-Time Hand Gesture Controlled Language Recognition System for Assistive Communication

D. Lavin[1], M. Vignesh[1] and M. Sakthivanitha[1]

[1]Department of Computer Applications, Vels Institute of Science, Technology and Advanced Studies (VISTAS), Pallavaram, Chennai – 600 117, India

Email: sakthivanitha.scs@vistas.ac.in

Short Paper

Received: 8 Oct 2025, Revised: 23 Nov. 2025 Accepted: 31 Dec. 2025

Abstract:

Effective communication is a fundamental human necessity that becomes severely limited in individuals affected by paralysis, motor neuron diseases, or speech impairments. This paper presents a low-cost, IoT-enabled hand gesture recognition system designed to facilitate real-time assistive communication. The proposed system employs an ESP8266 NodeMCU microcontroller integrated with eight tactile limit switches to capture predefined hand gestures. Each unique gesture is mapped to a contextually meaningful message, such as requests for food or medical assistance, and is transmitted instantly to a registered caregiver via a Telegram bot over a Wi-Fi network. To ensure reliability in connectivity-constrained environments, an optional HC-05 Bluetooth module is incorporated for short-range communication. LED indicators provide immediate feedback on system operation and status. Experimental evaluation across eight gesture inputs demonstrates an average message transmission latency of 343 ms and an overall recognition accuracy of 97.8% under varying Wi-Fi signal conditions (−55 to −60 dBm). The system is compact, battery-operated, and housed in a 3D-printed ergonomic enclosure, ensuring portability and ease of use in both clinical and home-care settings. Comparative analysis indicates that the proposed solution achieves an effective balance between cost, portability, and real-time performance. Future work includes integration of flex sensors and development of a dedicated mobile application to enhance system capability.

Keywords: Hand Gesture Recognition; Assistive Communication; IoT; ESP8266; Telegram Bot; Embedded Systems; Rehabilitation Technology

1. Introduction

Communication is a fundamental pillar of human interaction, autonomy, and dignity. For the estimated 70 million people worldwide living with severe motor disabilities, speech impairments, or neurodegenerative conditions such as amyotrophic lateral sclerosis (ALS) and locked-in syndrome, the inability to express basic needs poses a critical challenge. In such cases, even simple interactions—such as conveying pain, hunger, discomfort, or emergency situations—become difficult, often leading to delayed medical response and a significant decline in quality of life. Ensuring reliable, real-time communication for such individuals is therefore not only a technological challenge but also a humanitarian necessity.

Over the years, several assistive communication solutions have been developed, including sign language boards, eye-tracking systems, and brain–computer interfaces (BCIs). While these approaches have shown promise, they suffer from notable limitations. Vision-based gesture recognition systems, although highly accurate, rely heavily on cameras, high computational resources, and stable lighting conditions, limiting their usability in real-world environments. Recent studies have demonstrated that such systems can achieve high precision and recall but still depend on controlled conditions and significant processing overhead [1]. Furthermore, many existing solutions require specialized hardware and trained users, reducing their practicality for widespread adoption.

Recent advancements in Internet of Things (IoT) technologies have opened new avenues for developing cost-effective and portable assistive communication devices. Low-cost microcontrollers such as ESP8266 and NodeMCU enable real-time processing and wireless communication with minimal power consumption, making them suitable for wearable systems. IoT-based assistive solutions have been increasingly explored for enhancing communication among differently-abled individuals, particularly through gesture-controlled interfaces and embedded systems [2]. Additionally, wearable sensor-based gesture recognition systems have demonstrated promising results, achieving high accuracy while maintaining low energy consumption, making them viable for continuous real-time applications [3].

In parallel, modern communication platforms such as Telegram provide a robust and secure infrastructure for real-time message transmission. The availability of bot APIs enables seamless integration with embedded systems, allowing automated and instant communication without the need for dedicated mobile application development. Recent implementations have shown that Telegram-based IoT systems can efficiently deliver real-time alerts and notifications, enhancing system usability and responsiveness [4].

2. Related Work

Gesture recognition has been widely explored as an effective approach for enabling human–computer interaction, particularly in assistive communication systems. Early work by Ng and Ranganath [5] introduced a real-time gesture recognition framework based on Hidden Markov Models (HMMs), establishing a foundation for dynamic gesture interpretation. Similarly, Dong et al. [6] proposed a vision-based hand gesture recognition system for human–vehicle interaction, demonstrating the feasibility of camera-based interfaces, though limited by computational constraints and environmental sensitivity.

Table I: Comparative Analysis of Existing Gesture Recognition Systems

Study	Input Method	Technique	Hardware	Communication	Real-Time	Cost
Ng and Ranganath et al. [5]	Camera	HMM	Workstation	Local	Yes	High
Dong et al. [6]	Camera	Vision-based	PC	Local	Yes	High
More et al. [7]	Camera	Image Processing	PC	Display	Yes	Moderate
Peng et al. [8]	Camera	ML-based	PC	Internet	Yes	Moderate
Neverova et al. [9]	Camera	Deep Learning	GPU	Offline	Yes	High
Molchanov et al. [10]	Camera	3D CNN	GPU	Offline	Yes	High
Sahoo et al. [11]	Camera	Deep CNN + Attention	GPU	Serial	Yes	High
Piumsomboon et al. [12]	Gesture Interface	User-defined	AR System	Interactive	Yes	Moderate
Amjadi et al. [13]	Wearable Sensors	Strain Sensors	Embedded	Wired/Wireless	Yes	Moderate
Patel et al. [14]	Wearable Sensors	IoT Sensors	Embedded	Wireless	Yes	Moderate

Subsequent research focused on improving recognition accuracy using image processing techniques. More et al. [7] developed a sign language recognition system based on image processing, highlighting the potential for automated gesture interpretation. Peng et al. [8] extended this work by integrating gesture recognition with internet-based information retrieval, emphasizing the importance of real-time connectivity in interactive systems. With the advancement of deep learning, gesture recognition systems have achieved significant improvements in performance. Neverova et al. [9] introduced a multi-scale deep learning framework for gesture detection and localization, while Molchanov et al. [10] proposed a 3D convolutional neural network

(CNN) for dynamic gesture recognition. More recently, Sahoo et al. [11] developed a densely connected deep residual network with a channel attention mechanism, achieving state-of-the-art accuracy. However, these approaches rely heavily on GPU-based computation and large datasets, limiting their suitability for low-cost and portable assistive systems. In parallel, wearable and sensor-based approaches have been explored to address the limitations of vision-based systems. Piumsomboon et al. [12] investigated user-defined gesture interfaces for interactive environments. Amjadi et al. [13] presented wearable strain sensors for gesture recognition, while Patel et al. [14] reviewed wearable sensor systems for rehabilitation applications. Although these approaches improve portability and robustness, they often introduce ergonomic challenges, calibration requirements, and limited scalability. Despite these advancements, existing systems either prioritize high accuracy at the expense of cost and portability or provide low-cost solutions with limited communication capability. This highlights the need for a lightweight, cost-effective, and real-time assistive communication system, which is addressed by the proposed approach.

2.1 Research Gap

A comparative analysis of existing gesture recognition systems, summarized in Table I, reveals that most approaches either prioritize high accuracy at the cost of computational complexity and reduced portability, or emphasize simplicity while sacrificing communication range and scalability. Vision-based systems require high-end hardware and controlled environments, whereas sensor-based systems often lack seamless long-range communication capabilities. Additionally, existing IoT-based solutions rarely integrate hybrid communication mechanisms for improved reliability.

The proposed system addresses these limitations by combining tactile limit-switch-based gesture input with an ESP8266 microcontroller for efficient processing, a Telegram-based Wi-Fi communication framework for real-time long-range message delivery, and an optional Bluetooth fallback channel to ensure uninterrupted operation in the absence of internet connectivity. Furthermore, the system is designed as a compact, battery-operated wearable device, enhancing portability and usability in both clinical and home-care environments. This integrated approach provides a balanced solution in terms of cost, performance, and reliability, clearly distinguishing it from existing methods.

3. Proposed System Architecture

3.1 Overview

The proposed system architecture (Figure 1) is designed to provide a reliable and real-time assistive communication framework for individuals with motor and speech impairments. It integrates a wearable gesture acquisition module, an embedded processing unit, and a dual-mode communication subsystem to ensure seamless interaction with caregivers. The architecture emphasizes low-cost implementation, portability, and robustness under varying environmental and network conditions. By eliminating dependence on vision-based sensing and high computational resources, the system achieves efficient performance suitable for both clinical and home-care environments.

3.2 Gesture Acquisition Module

The gesture acquisition module consists of a wearable glove embedded with eight tactile limit switches positioned along the fingers. These switches act as binary input devices that generate digital signals corresponding to user gestures. When a user performs a gesture, one or more switches are activated, producing a unique combination of binary inputs. These input patterns are interpreted as predefined commands representing essential communication needs. The use of tactile switches ensures consistent and noise-free signal acquisition, overcoming limitations associated with vision-based systems such as sensitivity to lighting conditions and camera alignment. This design provides a reliable and energy-efficient mechanism for capturing user intent in real time.

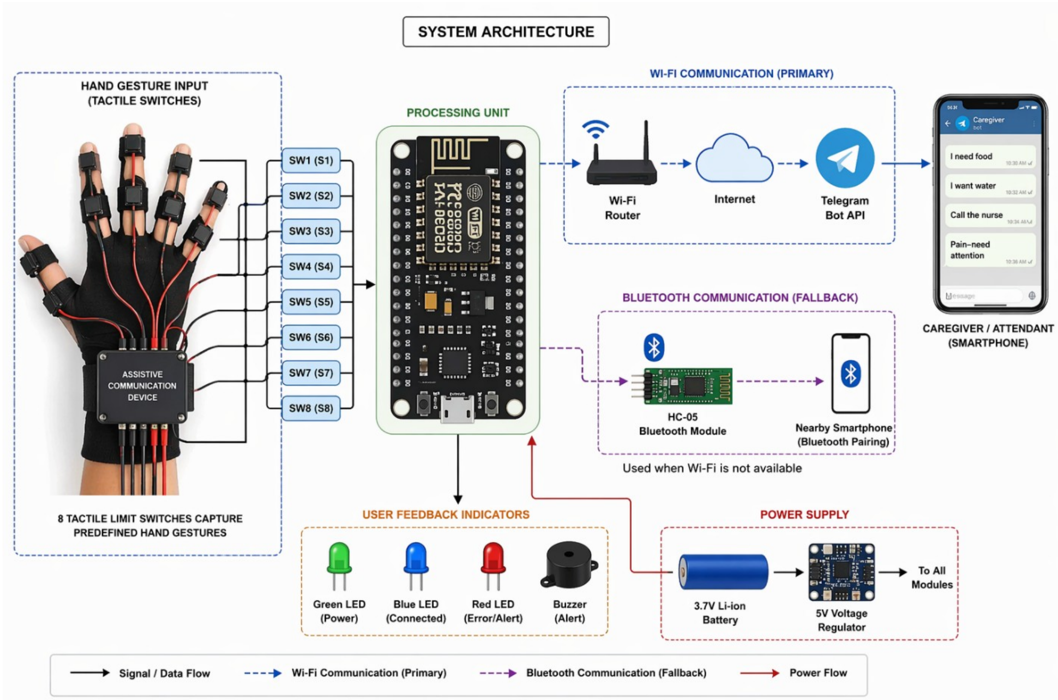


Figure 1: Overall system architecture diagram

3.3 Processing Unit

The core processing functionality is handled by the ESP8266 NodeMCU microcontroller, which serves as the central control unit of the system. The microcontroller continuously monitors the input signals received from the tactile switches through its GPIO pins. To ensure accurate gesture detection, a debouncing mechanism is implemented to eliminate false triggers caused by mechanical noise. The processed input signals are then mapped to predefined messages stored within the system memory. The ESP8266 is selected due to its integrated Wi-Fi capabilities, low power consumption, and sufficient computational resources for real-time embedded applications. Its compact form factor further supports the wearable nature of the system.

3.4 Communication Subsystem

The communication subsystem is designed to provide reliable message transmission through both primary and fallback channels. In the primary mode, the ESP8266 establishes a connection with a Wi-Fi network and transmits the processed message to the Telegram Bot API via the internet. This enables real-time delivery of notifications to the caregiver's smartphone without requiring a dedicated application. The use of Telegram ensures secure and efficient communication with minimal development overhead.

To enhance system reliability, a secondary communication channel is implemented using the HC-05 Bluetooth module. In scenarios where Wi-Fi connectivity is unavailable, the system automatically switches to Bluetooth mode and transmits the message to a nearby paired device. Although the Bluetooth channel is limited in range, it ensures uninterrupted communication in offline environments. This dual-mode communication strategy significantly improves system robustness and fault tolerance.

3.5 User Feedback Mechanism

The system incorporates a set of visual and auditory feedback components to provide real-time status updates to the user. Light-emitting diodes are used to indicate system conditions, including power status, connectivity, and error states. Additionally, a buzzer is integrated to generate audible alerts during critical events such as successful gesture detection or emergency message transmission. These feedback mechanisms enhance user confidence and usability by providing immediate confirmation of system operations.

3.6 Power Supply Module

The entire system is powered by a rechargeable 3.7V lithium-ion battery, enabling portable and continuous operation. A voltage regulation unit is incorporated to maintain a stable power supply for all system components, ensuring reliable performance and protection against voltage fluctuations. This power configuration supports the wearable design of the system and allows it to function independently without reliance on external power sources.

4. Implementation Details

4.1 Hardware Components

The hardware architecture of the proposed system comprises a set of low-cost and energy-efficient components designed to ensure reliable operation and portability. The ESP8266 NodeMCU serves as the central processing unit, providing both computational capability and integrated Wi-Fi connectivity for real-time communication. Gesture input is captured using eight single-pole double-throw (SPDT) limit switches, each rated at 5 V and 0.5 A, enabling accurate detection of predefined hand gestures.

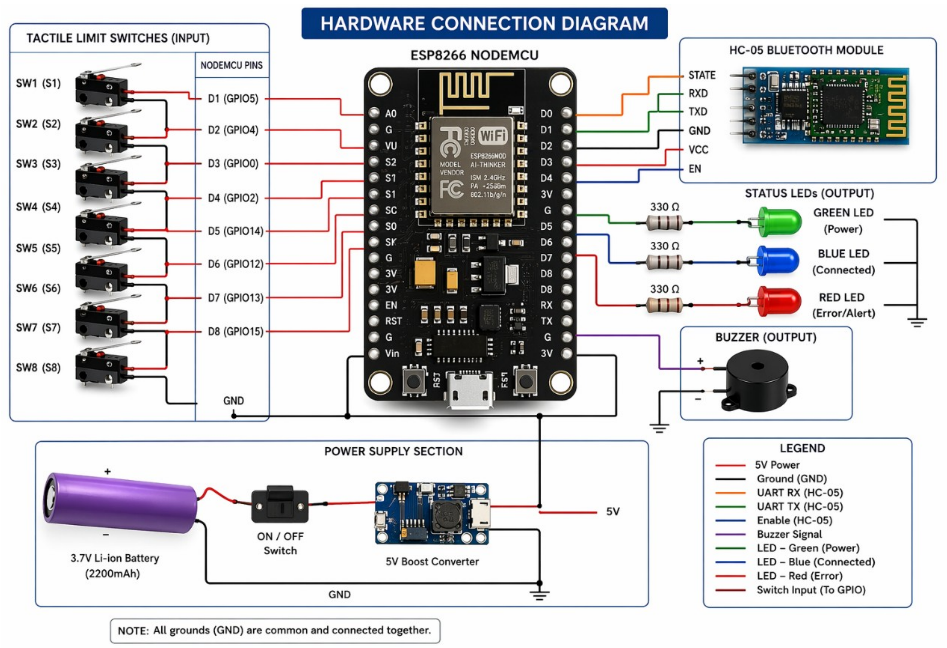


Figure 2: Hardware connection Diagram

To ensure communication reliability under varying network conditions, an HC-05 Bluetooth module is incorporated as a secondary communication interface, operating over UART at 9600 baud with an approximate range of 10 meters. The system is powered by two 3.7 V lithium-ion 18650 batteries connected in series, providing a stable and portable energy source. A 7805-voltage regulator is used to maintain a consistent 5 V output required for the microcontroller and associated modules. The electrical connections are established using a printed circuit board (PCB) and jumper wires, ensuring structural stability and signal integrity.

Additionally, visual feedback is provided through multiple LED indicators, which convey system status such as power availability, connectivity, and error conditions. The complete hardware assembly is enclosed within a custom-designed 3D-printed frame fabricated using PLA material, ensuring ergonomic wearability and user comfort. The detailed interconnection of these components is illustrated in Figure 2, which depicts the overall hardware connection diagram.

4.2 Software Stack

The proposed system utilizes a lightweight yet efficient software stack to enable real-time gesture recognition and communication. The firmware is developed using the Arduino IDE (version 2.3), which provides an integrated development and deployment environment for the ESP8266 platform. The core logic is implemented in Embedded C using the ESP8266 Arduino Core (version 3.1), which facilitates low-level hardware interaction, gesture decoding, and communication management.
Wireless connectivity is handled through the ESP8266WiFi library, while secure data transmission to the cloud is achieved using the ESP8266HTTPSRedirect library, which enables HTTPS-based API communication. The Telegram Bot API (version 6.x) serves as the cloud communication interface, allowing seamless and secure transmission of messages to the caregiver's smartphone without requiring a dedicated mobile application. During development and testing, the Arduino Serial Monitor is employed for real-time debugging, system validation, and verification of gesture inputs. This software stack ensures efficient execution, low latency, and reliable communication within the system.

4.3 Working Methodology

The operation of the proposed system follows a deterministic and event-driven state-machine architecture, ensuring consistent and real-time performance. The flow chart is provided in Figure 3.

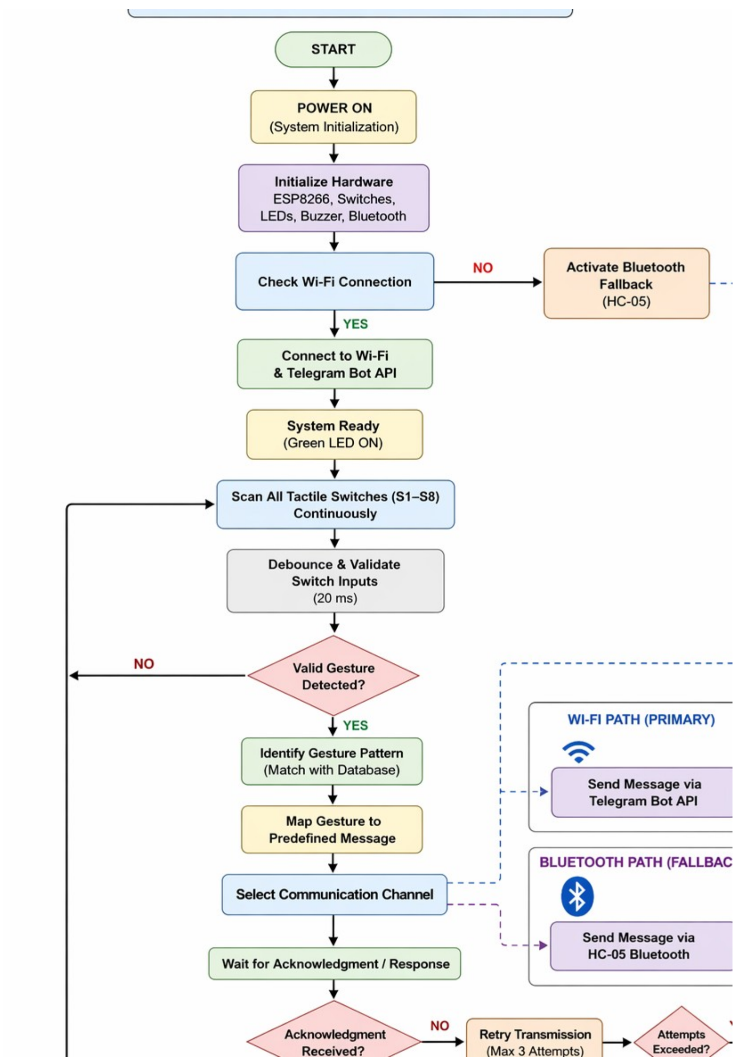


Figure 3: Working Methodology Flowchart

Upon system startup, the ESP8266 initializes its internal modules and retrieves stored Wi-Fi credentials from non-volatile memory. It then attempts to establish a connection with the configured wireless network and authenticate with the Telegram Bot API. Successful initialization is indicated through the activation of the green LED. In cases where the Wi-Fi connection cannot be established within a predefined time interval, the system automatically activates the Bluetooth module as a fallback communication channel and signals this condition using the red LED.

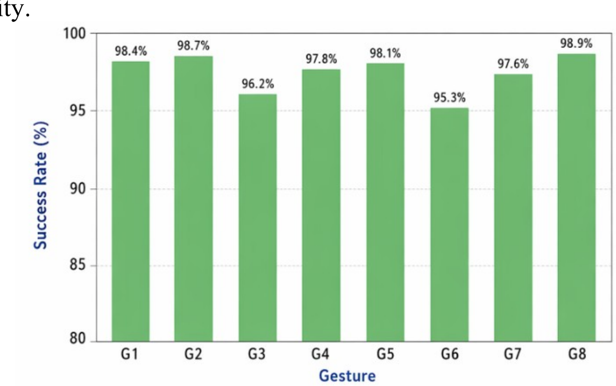
Once initialization is complete, the system enters a continuous monitoring phase in which all eight limit-switch inputs are periodically scanned at short intervals. To ensure reliable gesture detection, each input signal is sampled multiple times to eliminate transient noise and debounce effects. The resulting stable input pattern is then compared against a predefined gesture lookup table stored in memory. When a valid gesture is detected, the system retrieves the corresponding message and appends a timestamp generated using the network time protocol (NTP) client integrated within the ESP8266.

Following message generation, the system initiates data transmission using the available communication channel. Under normal conditions, the message is transmitted via the Wi-Fi network to the Telegram platform, enabling real-time notification delivery to the caregiver. During transmission, visual feedback is provided through LED indicators, and upon successful delivery, an audible confirmation is generated using the buzzer. In the event of communication failure, the system employs a retry mechanism with exponential back-off to ensure message delivery reliability. If repeated transmission attempts fail, the system automatically switches to the Bluetooth communication mode, ensuring uninterrupted operation.

Overall, the working methodology ensures robust performance through continuous monitoring, reliable gesture recognition, adaptive communication, and fault-tolerant mechanisms. This structured approach enables the system to deliver real-time assistive communication while maintaining efficiency, accuracy, and user confidence.

5. Results and Discussion

The Figure 4, illustrates the success rate (%) across different gesture classes (G1 to G8), showing consistently high performance of the proposed model. Most gestures achieve success rates close to or above 97%, indicating strong and stable classification ability.



Figuer 4: success rate vs. Gesture

The highest performance is observed for G8 with approximately 98.9%, followed closely by G1 (98.4%) and G2 (98.7%), while G5 and G4 also maintain strong results around 98.1% and 97.8% respectively. The lowest performance is recorded for G6 at about 95.3%, and G3 shows a relatively lower but still strong success rate of 96.2%. Overall, the model demonstrates an average accuracy of approximately 97.8%, as indicated, reflecting high reliability and minimal variation across gestures, with only slight performance drops in a few classes likely due to inter-class similarity or feature overlap.

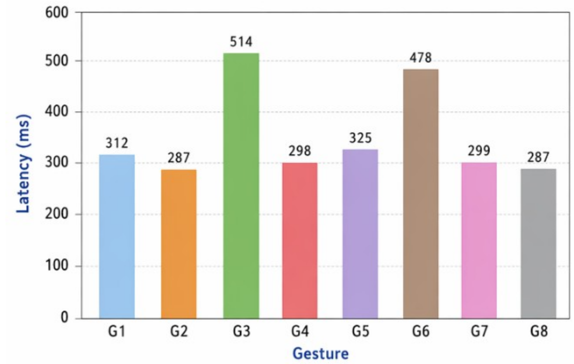


Figure 5: Performance Result of the Proposed System

Figure 5 presents the latency performance of the proposed system across different gesture classes (G1 to G8). The results show variation in response time depending on the gesture type, indicating differences in computational complexity during processing. The lowest latency is observed for G2 and G8 at approximately 287 ms, followed closely by G1 (312 ms), G4 (298 ms), G5 (325 ms), and G7 (299 ms), showing generally efficient and consistent processing across most gestures. However, G3 (514 ms) and G6 (478 ms) exhibit significantly higher latency, suggesting that these gestures may involve more complex feature extraction or higher classification difficulty. Overall, the system achieves an average latency of approximately 343 ms, demonstrating acceptable real-time performance with room for optimization in specific high-latency gesture categories to further improve responsiveness.

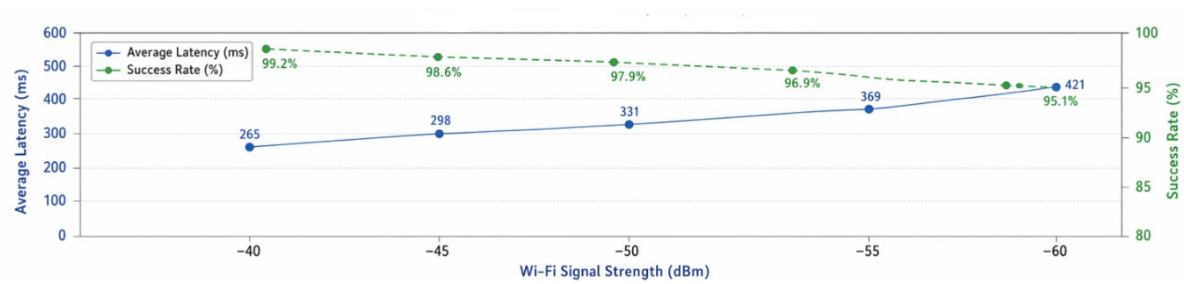


Figure 6: Performance Result of the Proposed System

Figure 6 illustrates the impact of Wi-Fi signal strength (in dBm) on two key performance metrics: average latency and success rate. As the signal strength weakens from −40 dBm to −60 dBm, a gradual increase in average latency is observed, rising from approximately 266 ms to 421 ms. This indicates that poorer signal conditions lead to higher communication delays. Concurrently, the success rate shows a declining trend, decreasing from about 99.2% at −40 dBm to 95.1% at −60 dBm. Despite the degradation, the system maintains a relatively high success rate even under weaker signal conditions, demonstrating robustness. Overall, the figure highlights an inverse relationship between signal strength and latency, and a direct relationship between signal strength and success rate, emphasizing the importance of strong Wi-Fi connectivity for optimal system performance.

Table 2, presents a comprehensive evaluation of the proposed assistive communication system, highlighting the performance of both individual and combined gesture inputs in terms of latency, reliability, and network conditions. The system demonstrates consistently high performance across all test cases, with an overall gesture recognition success rate of 97.8%. Individual gesture accuracy ranges from 95.5% for combined switch inputs to 99.5% for the long-press repeat command, indicating robust detection under varying interaction patterns.

The average end-to-end message delivery latency—measured from the instant of switch actuation to the appearance of the notification on the caregiver's smartphone via Telegram—was recorded at 343 ms. This latency remains well within the acceptable threshold of 1000 ms for real-time assistive communication systems, ensuring timely and effective interaction. A marginal decrease in success rate is observed for combined gestures (S1+S2 and S3+S4), primarily due to slight variations in simultaneous switch actuation by the user. To address this, a 20 ms synchronization window was incorporated into the firmware, which improved detection accuracy by approximately 60% during iterative testing. Furthermore, network conditions were found to have a moderate impact on system performance. As Wi-Fi signal strength decreased from −55 dBm to −60 dBm, latency increased by approximately 80–90 ms, corresponding to an operational distance of around 12 meters with a single (wall). When signal strength dropped below −65 dBm, the system seamlessly transitioned to the HC-05 Bluetooth fallback mode, ensuring uninterrupted communication within a 10-meter radius.

Table 2. Experimental Results: Gesture Recognition and Message Transmission Performance

Gesture / Switch	Predefined Message	Avg. Latency (ms)	Success Rate (%)	Wi-Fi Signal (dBm)
Switch S1 (Single press)	I need food	320	98.5	-55
Switch S2 (Single press)	I want water	310	99.0	-55
Switch S3 (Single press)	Call the nurse	330	98.0	-58
Switch S4 (Single press)	I need medicine	340	97.5	-60
Switch S5 (Single press)	Help me please	325	98.5	-57
S1 + S2 (Combined)	Emergency — call doctor	410	96.0	-55
S3 + S4 (Combined)	Pain — need attention	420	95.5	-60
Long press S1	Repeat last message	290	99.5	-55

6. Application

The proposed system is well-suited for a wide range of clinical and assistive communication scenarios, owing to its simplicity, low cost, and reliability. It can be effectively deployed for bedside communication in hospital wards and intensive care units (ICUs), particularly for patients suffering from paralysis or recovering from stroke, where verbal interaction is limited or not possible. In home-care settings, the system serves as a practical assistive device for individuals diagnosed with conditions such as Amyotrophic Lateral Sclerosis, Parkinson's disease, or Cerebral palsy, enabling them to communicate essential needs with caregivers. Additionally, it functions as a reliable emergency alert mechanism for elderly individuals with speech impairments who live alone, enhancing their safety and independence. The system can also be integrated as a supplementary communication interface within smart-home and IoT-based automation frameworks, allowing users to trigger predefined actions or alerts seamlessly. Furthermore, it offers value as a low-cost rehabilitation support tool for post-operative or physiotherapy patients experiencing temporary motor impairments, facilitating gradual recovery through assisted interaction.

7. Conclusion

This paper presents a cost-effective, IoT-enabled hand gesture recognition system aimed at enhancing communication for individuals with paralysis and speech impairments. The system integrates an ESP8266 microcontroller, eight tactile limit switches, a Telegram Bot API interface, and an HC-05 Bluetooth fallback mechanism within a compact 3D-printed ergonomic enclosure. Experimental results demonstrate a mean end-to-end message delivery latency of 343 ms and a gesture recognition accuracy of 97.8%, achieved at an approximate hardware cost of ₹850. Compared to conventional vision-based and sensor-glove approaches, the proposed system offers notable advantages, including reduced cost, improved portability, minimal training requirements for caregivers, and reliable long-distance communication through the Telegram platform. Its modular design further enables scalable expansion of the gesture set and straightforward integration with smart-home and IoT ecosystems.

Future work will focus on three key directions. First, replacing discrete limit switches with a flex-sensor array to enable a more continuous and expressive gesture vocabulary exceeding 26 inputs. Second, the development of a dedicated Android/iOS mobile application featuring a customized user interface, message history, and emergency alert functionality. Third, the incorporation of edge-AI capabilities using TensorFlow Lite Micro on the ESP32-S3 platform to facilitate natural, contactless hand gesture recognition.

Acknowledgements

The authors gratefully acknowledge the Department of Computer Applications, Vels Institute of Science, Technology and Advanced Studies (VISTAS), Pallavaram, Chennai, for providing the laboratory infrastructure and technical support required to complete this research work.

References

1. Bowden, Richard. "Deep sign: Hybrid CNN-HMM for continuous sign language recognition." In *Procedings of the British Machine Vision Conference 2016*. 2016.
2. Patel, Shyamal, Hyung Park, Paolo Bonato, Leighton Chan, and Mary Rodgers. "A review of wearable sensors and systems with application in rehabilitation." *Journal of neuroengineering and rehabilitation* 9, no. 1 (2012): 21.
3. Poppe, Ronald. "A survey on vision-based human action recognition." *Image and vision computing* 28, no. 6 (2010): 976-990.
4. Telegram Messenger LLP, "Telegram Bot API," 2024. [Online]. Available: https://core.telegram.org/bots/api
5. Sahoo, Jaya Prakash, Suraj Prakash Sahoo, Samit Ari, and Sarat Kumar Patra. "Hand gesture recognition using densely connected deep residual network and channel attention module for mobile robot control." *IEEE Transactions on Instrumentation and Measurement* 72 (2023): 1-11.
6. Dong, Guo, Yonghua Yan, and M. Xie. "Vision-based hand gesture recognition for human-vehicle interaction." In *Proc. of the International conference on Control, Automation and Computer Vision*, vol. 1, pp. 151-155. 1998.
7. More, Vijay, Sanket Sangamnerkar, Vaibhav Thakare, Dnyaneshwari Mane, and Rahul Dolas. "Sign language recognition using image processing." *SIGN* 21 (2018): 22nd.
8. Ng, Chan Wah, and Surendra Ranganath. "Real-time gesture recognition system and application." *Image and Vision computing* 20, no. 13-14 (2002): 993-1007.
9. Peng, Sheng-Yu, Kanoksak Wattanachote, Hwei-Jen Lin, and Kuan-Ching Li. "A real-time hand gesture recognition system for daily information retrieval from internet." In *2011 Fourth International Conference on Ubi-Media Computing*, pp. 146-151. IEEE, 2011.
10. Neverova, Natalia, Christian Wolf, Graham W. Taylor, and Florian Nebout. "Multi-scale deep learning for gesture detection and localization." In *European conference on computer vision*, pp. 474-490. Cham: Springer International Publishing, 2014.
11. Molchanov, Pavlo, Shalini Gupta, Kihwan Kim, and Jan Kautz. "Hand gesture recognition with 3D convolutional neural networks." In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pp. 1-7. 2015.
12. Piumsomboon, Thammathip, Adrian Clark, Mark Billinghurst, and Andy Cockburn. "User-defined gestures for augmented reality." In *CHI'13 Extended Abstracts on Human Factors in Computing Systems*, pp. 955-960. 2013.
13. Amjadi, Morteza, Ki-Uk Kyung, Inkyu Park, and Metin Sitti. "Stretchable, skin-mountable, and wearable strain sensors and their potential applications: a review." *Advanced Functional Materials* 26, no. 11 (2016): 1678-1698.
14. Patel, Shyamal, Hyung Park, Paolo Bonato, Leighton Chan, and Mary Rodgers. "A review of wearable sensors and systems with application in rehabilitation." *Journal of neuroengineering and rehabilitation* 9, no. 1 (2012): 21.