STM JOURNALS

Journal of
Image Processing & Pattern Recognition Progress

ISSN: 2394-1995
Volume 12, Issue 1, 2025
January–April
DOI (Journal): 10.37591/JoIPPRP

https://journals.stmjournals.com/joipprp

Research JoIPPRP

Sign Language-enabled Offline IP-based Video Call Intercom System for the Deaf

Prakrati Bajpai[1,*], Soni M.[2], Nikita Singh[1], Ravi Shankar Kumar[1], Sanjay B.R.[1]

Abstract

Individuals who are deaf or hard of hearing frequently encounter major communication challenges on digital platforms because sign language support is often insufficient. This study presents a real-time, internet-independent sign language-to-text translation system embedded in an offline IP-based video intercom, designed to facilitate seamless communication. Leveraging TensorFlow's Convolutional Neural Network (CNN) for classification and MediaPipe for hand tracking, the system achieved a training accuracy of 94.24% and a validation accuracy of 94.01% within 20 epochs. In contrast to existing solutions that depend on internet connectivity or specialized hardware, this system operates offline, making it scalable, cost-effective, and adaptable for diverse environments. Furthermore, it integrates vibration sensors to provide tactile alerts, enhancing emergency notifications and usability in noisy or visually inaccessible settings. By combining real-time sign language translation with an offline IP-based video intercom, this system addresses the limitations of conventional video calling platforms. It offers an inclusive and practical solution for communication, making it particularly suitable for workplaces, public spaces, and other environments requiring accessible communication tools for the deaf and hard-of-hearing community.

Keywords: Sign language recognition, CNN, video call intercom, offline LAN IP system, sensor, real-time text conversion, WebRTC, tactile feedback, emergency alerts

INTRODUCTION

Effective communication is the cornerstone of successful interactions in any environment, including workplaces, business organizations, and homes. However, traditional video calling systems pose significant challenges for individuals who are deaf or hard of hearing. These systems rely heavily on auditory cues, such as phone calls or intercom systems, which exclude those who primarily communicate through sign language, thereby creating barriers in everyday interactions. Advancements in machine learning, computer vision, and IoT-based systems have paved the way for innovative solutions to bridge this communication gap. Sign language recognition has gained considerable attention in recent years, with methodologies focusing on real-time gesture monitoring, neural networks, and multimodal approaches. Meng *et al*. demonstrated the effectiveness of MediaPipe's real-time hand tracking system in accurately detecting hand gestures [1]. Similarly, Moryossef *et al*. utilized Open Pose's robust human pose estimation for real-time sign language detection, highlighting the potential of lightweight and efficient models for practical deployment [2, 3]. The proposed system

***Author for Correspondence**
Prakrati Bajpai
E-mail: pbajpai091@gmail.com

[1]Student, Department Electrical and Electronics Engineering, Dayananda Sagar College of Engineering, Bangalore, Karnataka, India
[2]Assistant Professor, Department Electrical and Electronics Engineering, Dayananda Sagar College of Engineering, Bangalore, Karnataka, India

Received Date: January 15, 2025
Accepted Date: January 18, 2025
Published Date: February 07, 2025

Citation: Prakrati Bajpai, Soni M., Nikita Singh, Ravi Shankar Kumar, Sanjay B.R. Sign Language-enabled Offline IP-based Video Call Intercom System for the Deaf. Journal of Image Processing & Pattern Recognition Progress. 2025; 12(1): 53–66p.

leverages these advancements by integrating Media Pipe's hand-tracking capabilities with TensorFlow's Convolutional Neural Networks (CNNs) for robust and accurate gesture recognition [4]. MediaPipe's ability to track hand landmarks in real time, combined with CNN's deep learning capabilities, ensures high precision in translating sign language gestures into text.

This feature is critical for bridging the communication gap between deaf and hearing individuals, enabling seamless interactions regardless of the participants' familiarity with sign language. Studies such as those by Camgoz *et al*. and Huang *et al*. have demonstrated the effectiveness of deep learning models, including hybrid CNN HMM systems and attention-based neural networks, in achieving significant accuracy for gesture recognition [5–8]. To enhance accessibility, the system incorporates tactile feedback mechanisms, such as a wearable wristband for notifications, inspired by Miah *et al.*, who discussed the integration of tactile alerts in wearable devices for haptic feedback [9]. This feature is particularly useful in visually or audibly restrictive environments like noisy workplaces, where physical cues are essential for notifications. The system operates on a Local Area Network (LAN) facilitated by a Raspberry Pi, eliminating dependency on external internet connectivity. This offline architecture aligns with the principles outlined by Adaloglou *et al*., for lightweight and resource-efficient models [8].

It ensures scalability and cost-effectiveness, making the system suitable for deployment in diverse environments such as offices, public spaces, and residential areas. By combining gesture recognition, text conversion, and video call functionality, the proposed system serves as a comprehensive platform for inclusive communication. Ultimately, this project aligns with the broader vision of fostering an inclusive society by reducing communication barriers. Recent literature, including studies by Aggarwal *et al*. and Al Abdullah *et al*., underscores the need for real-time, efficient, and user-friendly systems [10, 11]. It empowers individuals with hearing impairments to participate fully and independently in their personal and professional lives. By integrating video call functionality, tactile alerts, and real-time translation, this system not only enhances internal communication efficiency but also provides a transformative tool for enabling equal participation in diverse settings.

SYSTEM DESCRIPTION

This system is designed to facilitate seamless communication by integrating multiple devices interconnected through a Raspberry Pi, which functions as a centralized server [3]. The server establishes a Local Area Network (LAN) via a router, enabling communication without the need for internet access [12]. A key component of the system is a camera module that captures users' hand gestures, which are then processed by a machine learning model trained to recognize and translate these gestures into readable text [13].

In addition to gesture recognition, the system incorporates speech recognition technology to further enhance communication between deaf and hearing individuals. This feature transcribes spoken words from the other party into text, making verbal communication accessible to deaf users by displaying the transcribed text on their side. The system is integrated with an IP-based video intercom, equipped with sign language recognition capabilities, ensuring inclusive and effective communication [14]. To improve accessibility, vibration sensors, embedded in a wearable wristband, provide tactile alerts during emergencies. This functionality ensures that users receive critical notifications such as call alerts even when visual or auditory cues are unavailable (Figure 1).

COMPONENT DESCRIPTION

Web Camera Feed

This web-cam feed as real-time gestures performed by the user, providing input data for the machine learning model hosted on the Raspberry Pi. It acts as the primary sensor for interpreting sign language gestures, enabling seamless communication for deaf and mute individuals [15].

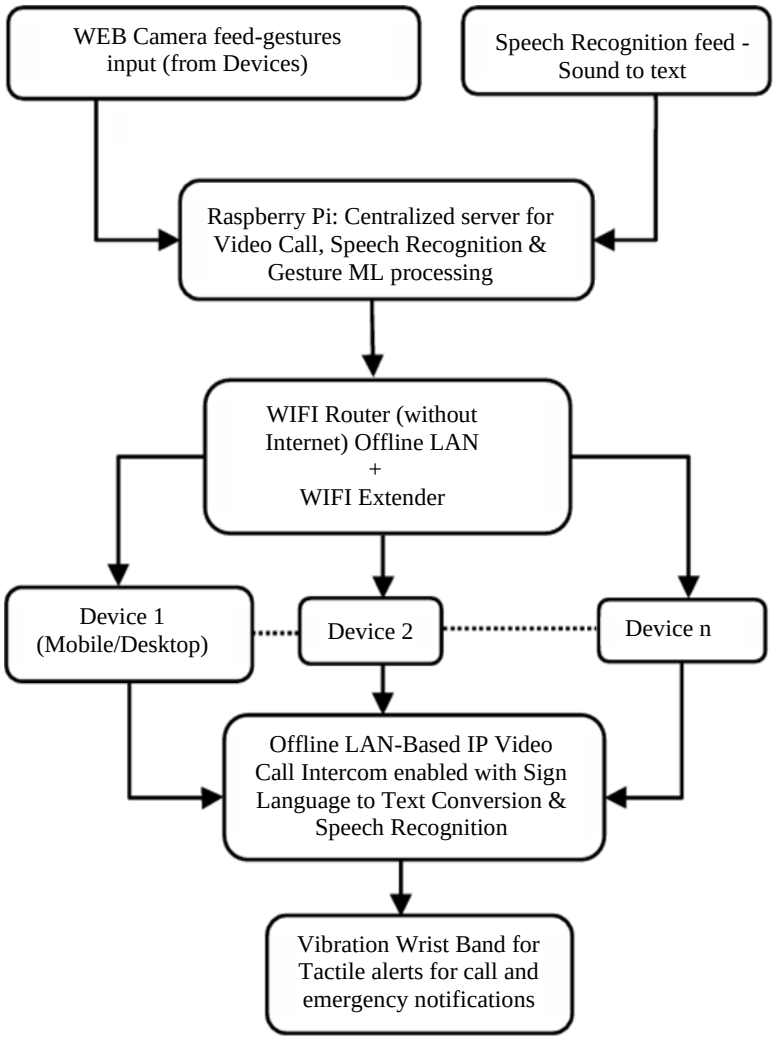

Figure 1. Block diagram of the system.

Raspberry Pi (Centralized Server)

The Raspberry Pi (Model 4B+) serves as the centralized server for the system. It hosts the machine learning model for gesture recognition and manages the communication between devices on the network [16]. The Raspberry Pi processes the data from the camera module and translates recognized gestures into readable text, which is displayed on connected devices. It connects to the router for network management, ensuring all devices on the same network can access the system offline, without requiring internet connectivity [17].

Router (Network Extender)

The router is configured to create a local area network (LAN) and provides extended coverage for seamless communication. Raspberry Pi is connected to it and acts as a server where all the central works is done [18]. The router ensures scalability, allowing multiple devices, including smartphones and computers, to connect to the system even in larger areas like offices or residential complexes [19].

Indoor Range: 100–200 m
Outdoor Range: 300–500 m

Offline LAN-Based IP Video Call Intercom

The Offline LAN-Based IP Video Call Intercom enables seamless communication without internet by using a Raspberry Pi as a server and a router for extended LAN coverage. It supports video calls

within the network and integrates a sign language recognition model that translates gestures into text for deaf users [20]. Speech recognition transcribes spoken words into text, enabling two-way communication between deaf and hearing individuals. The system includes tactile notifications through a vibration sensor for alerts. This solution ensures inclusive, reliable, and scalable communication in offline environments. It allows deaf users to communicate with hearing individuals through:

1. *Sign Language to Text Conversion:* The machine learning model translates gestures into text during video calls.
2. *Two-Way Interaction:* Deaf individuals communicate via gestures, and their messages are displayed as text on the receiver's device. Conversely, speech from hearing users is transcribed into text for the deaf user, ensuring accessible and inclusive communication.

Tactile Alert System

This system uses wearable technology to provide tactile notifications to users in emergencies or for incoming call alerts. Key components include:

- *Bluetooth Module:* It receives incoming signals when a call notification or emergency signal is triggered to the wearable device.
- *Transistor (Switch):* Acts as a control mechanism for the vibration motor. When the Bluetooth module receives a signal, it activates the transistor, allowing current to flow (Figures 2 and 3).
- *Vibration Motor:* Generates physical vibrations to alert the user to incoming calls or emergencies.
- *Vibration Sensor:* Amplifies the detected signals to ensure the motor produces a strong tactile response, even in noisy environments.

WORKING METHODOLOGY

The system enables seamless offline communication by using a LAN-based IP video intercom managed by a Raspberry Pi server as described in Figure 4. Real-time hand gestures are captured through a camera and tracked using ML model. The translated text is displayed on the other side, bridging the communication gap for deaf and hard-of-hearing users. Additionally, vibration sensors provide tactile alerts for calls or emergencies, ensuring accessibility even in noisy environments. This efficient, internet-independent solution is practical for workplaces, public spaces, and other settings requiring inclusive communication tools. The flow of complete working methodology of the proposed system is clearly demonstrated in Figure 4.

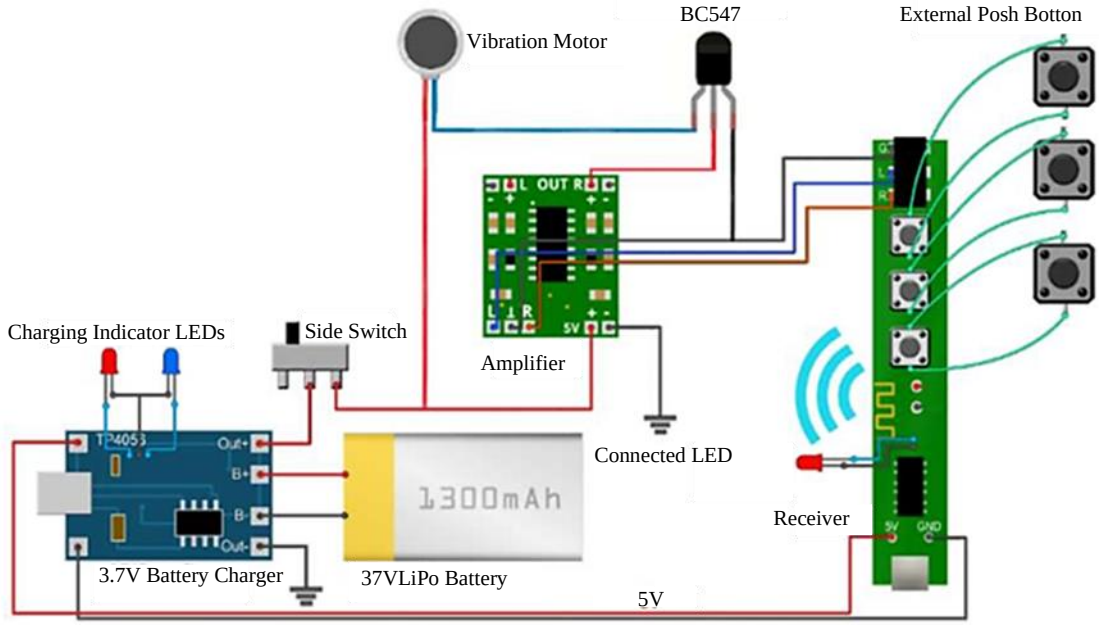

Figure 2. Tactile alert system circuit diagram.

Figure 3. Tactile alert system hardware circuit.

System Overview

The application operates in two primary modes, catering to distinct user groups: Sign Language Detection which is designed for normal users to read textual interpretations of sign language gestures made by deaf users. Speech-to-Text Recognition which is designed for deaf users to read textual transcriptions of spoken words from normal users.

Sign Language Detection

For normal users, sign language detection is activated to translate hand gestures into readable text. The system performs the following steps:

- Video Frame Capture: Video frames are captured from the user's camera.
- Frame Processing: The frames are sent to a backend server implemented in Python using Flask. MediaPipe extracts 3D hand landmarks from the captured frames. TensorFlow processes the landmarks to classify them into corresponding gestures.
- Gesture Prediction: The backend server predicts gestures and returns the results to the frontend.
- UI Update: The frontend displays the translated text to the normal user, facilitating effective communication.

Speech-to-Text Recognition

For deaf users, speech-to-text recognition is activated to convert spoken words into text. The process follows these steps:

- Speech Input Capture: The system captures speech input from the microphone of the normal user.
- Speech Processing: Using the Web Speech API, the speech is processed in real-time to extract meaningful content.
- Text Conversion: The recognized speech is converted into text and displayed on the deaf user's interface.

WebRTC Video Call Setup

The application enables real-time video calling between users through WebRTC. The session is initialized using Node.js backend which handles signaling and session establishment via Socket.io. Video and audio streams are exchanged between users after a successful connection. Optional features like sign language detection and speech recognition are seamlessly integrated into the video call, allowing users to enable or disable them based on their communication needs.

Unified Output

Both sign language detection and speech-to-text recognition outputs are combined and displayed on the user interface. Normal users can read the textual interpretations of gestures made by deaf users. Deaf users can read transcriptions of speech from normal users, enhancing accessibility during the video call.

Transmission to Central Server (Raspberry Pi)

The captured video feed or still images are transmitted to the Raspberry Pi, which acts as the centralized server where all the backend processing is performed, including sign language conversion to text and speech recognition. The Raspberry Pi is connected to a router via offline LAN, enabling it to process gesture inputs and communicate within the local network. This ensures real-time data handling without the need for external servers or internet connectivity. The system operates within a defined LAN range: Indoor: 30–50 m without any additional repeaters; and Outdoor: 100 m. If extended range is needed, the router with a repeater enhances the LAN coverage, allowing for a larger operating distance (indoor: 100–200 m, outdoor: up to 500 m).

Gesture Processing and Recognition

Dataset Preparation

The real-time gesture recognition model is trained on the Indian Sign Language dataset which contains over 36,000 labelled images (1000 images per class for 36 classes). Each image is pre-processed, resized and augmented (using flipping, rotation, zooming) to improve model's ability of recognition even in challenging conditions.

Data Preprocessing

The system uses MediaPipe's Hand Landmarks Model to detect and track key points on the user's hands in real-time. It segments the hand from the background by locating palm and predicting the position of 21 key landmarks on each hand which corresponds to important points such as finger joints, fingertips and base of the palm.

Normalization

The extracted 3D coordinates of the landmarks are normalized relative to the size of the image, making the data independent of the hand's position or scale in the frame. This allows for better generalization across different users, hand sizes, and distances from the camera.

Flattening the Coordinates

The 3D coordinates for all 21 landmarks are flattened into a vector, which becomes the input for the model. For example, if using two hands, you may have 42 coordinates (21 per hand) that describe the spatial configuration of both hands.

Training

Instead of feeding raw images to the machine learning model, these landmark vectors are fed into a dense neural network (DNN) or a CNN that has been specifically trained on such structured data. This model is trained to classify these landmarks into specific gesture classes (A–Z and 0–9).

The Convolutional Neural Network (CNN) is trained using the pre-processed dataset which is designed to learn both low-level (edges, corners) and high-level features (hand shape, postures). The dataset is split into training and testing subsets in the ratio of 70:30, to evaluate the model's performance and ensure it generalizes well to unseen data.

Optimization and Accuracy

Adam optimizer is employed to minimize the loss function (categorical cross-entropy), ensuring the model learns the correct mapping between input gestures and output classes. Early Stopping is applied during training to prevent overfitting. If the model's performance on validation data does not improve over a few epochs, training is halted, and the best-performing model is saved.

Deployment

The final trained model is deployed on the web application, hosted on a server accessible within the Offline LAN. The web application, including this model, is then deployed on the Raspberry Pi for on-site, networked use, ensuring efficient real-time recognition and text conversion.

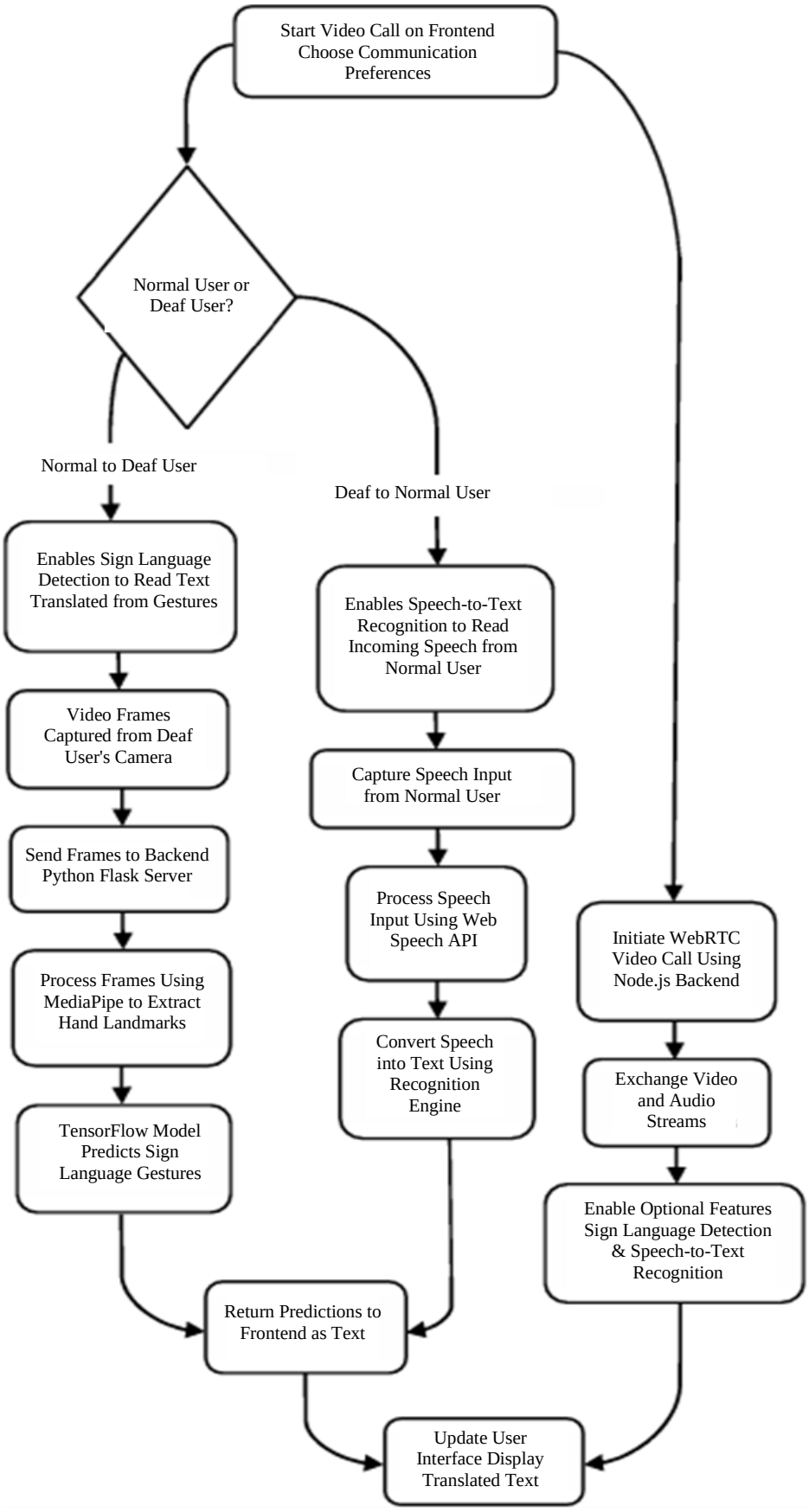

Figure 4. Working methodology flowchart.

Networked Communication and Device Interaction

All devices in the system are connected to the same offline LAN network as the Raspberry Pi server through a local Wi-Fi router. The Raspberry Pi functions as the centralized server, handling all data processing and routing it through the web application to enable smooth communication and coordination between devices. This ensures that video and gesture recognition data are transmitted efficiently within the local network, without relying on external internet connectivity. In cases where network coverage needs to be expanded, such as in large installations or multi-building environments, a network repeater can be used to extend the LAN's reach, maintaining reliable and uninterrupted communication. This setup guarantees seamless interaction and high data transfer speeds even in more extensive network configurations, ensuring the system's performance remains stable without lag or signal loss (Figure 5).

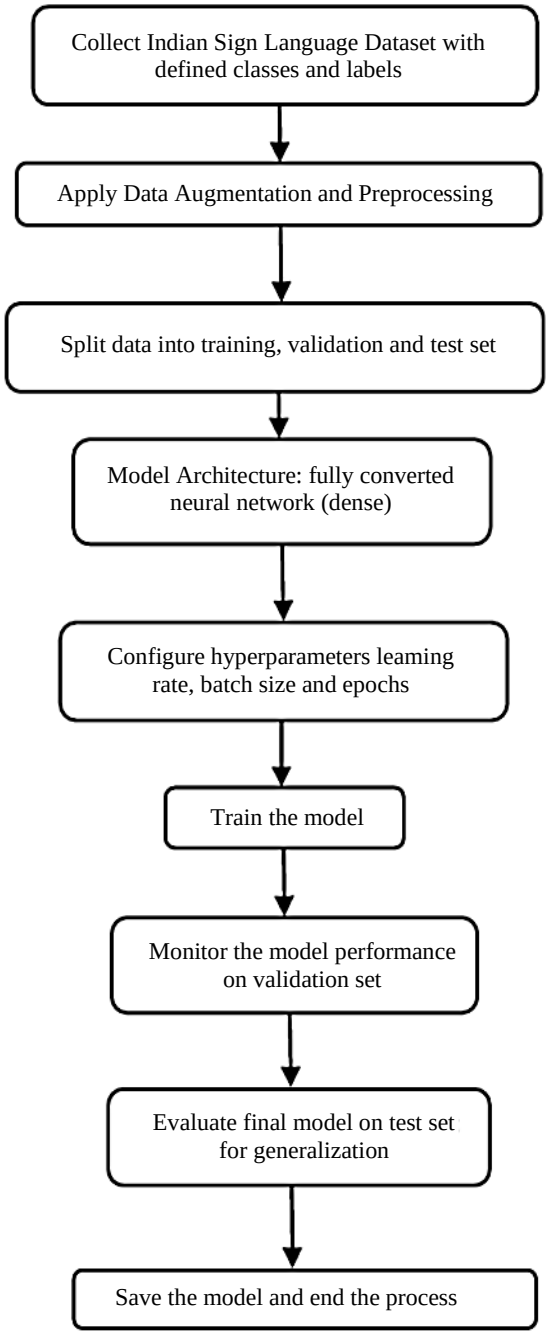

Figure 5. Model Architecture.

Tactile Alerts and Emergency Notifications

The wristband is equipped with a vibration sensor that provides tactile alerts for various notifications, including incoming calls, system notifications, and emergency alerts. This functionality ensures that deaf or hard-of-hearing users receive critical alerts through physical vibrations. The vibration sensor is activated by emergency signals or other important notifications, allowing users to respond promptly to time-sensitive situations.

RESULTS AND DISCUSSION

The machine learning model for gesture recognition developed using TensorFlow's Convolutional Neural Network, was trained on a dataset of Indian Sign Language (ISL) hand gestures. The dataset used in this study was sourced from the Indian Sign Language Research and Training Centre (ISLRTC) [21], which provides a comprehensive collection of hand gestures corresponding to various words and phrases in Indian Sign Language (Figure 6).

The training vs. accuracy plot shown in the Figure 7 indicates a steady increase in accuracy over 20 epochs, reaching 94.24% for training and 94.01% for validation. This demonstrates the effectiveness of the CNN model for recognizing and translating sign language gestures in real time.

It can be clearly observed in Figure 8 that there is a steady decrease in loss, indicating that the model is learning efficiently and not overfitting to the training data. These results suggest that the model is well-suited for deployment in real-world scenarios, where it can accurately translate sign language gestures into text with minimal error.

The confusion matrix as shown in Figure 9 statistically validates the algorithm's performance across 36 classes (digits 0–9 and letters A Z). The majority of predictions lie along the diagonal, indicating a high true positive rate for most classes. Quantitatively, the diagonal values demonstrate that over 94.24% of the total samples are correctly classified, reflecting the model's strong classification capability. Off-diagonal values, representing misclassifications, are minimal, with most classes having fewer than 6% misclassifications, indicating a low false positive rate. These statistical findings confirm the algorithm's effectiveness and robustness.

Figure 6. Indian sign language.

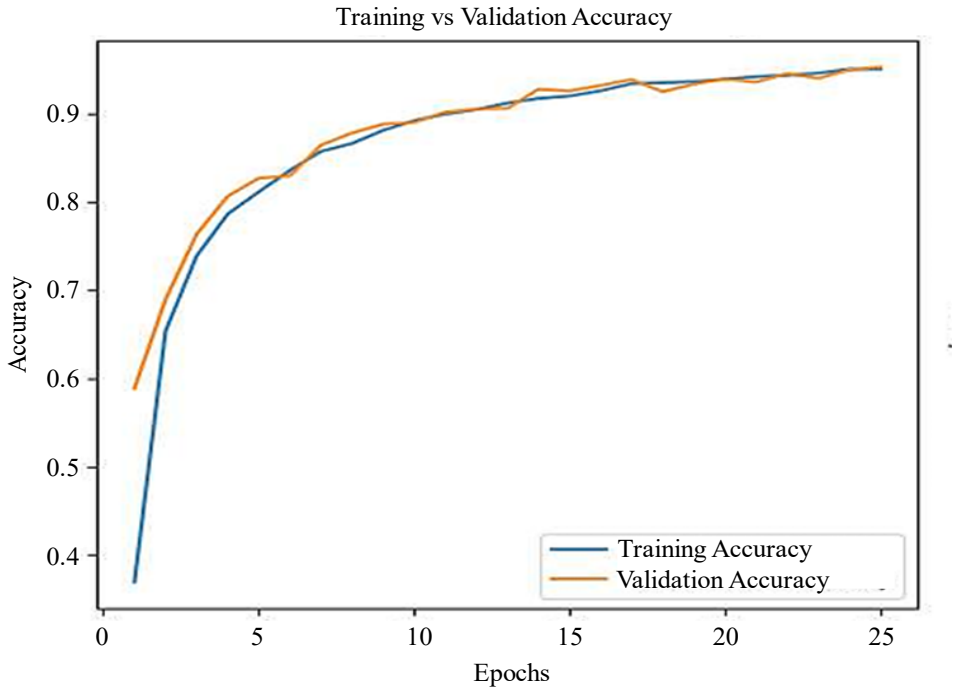

Figure 7. Training v/s accuracy plot.

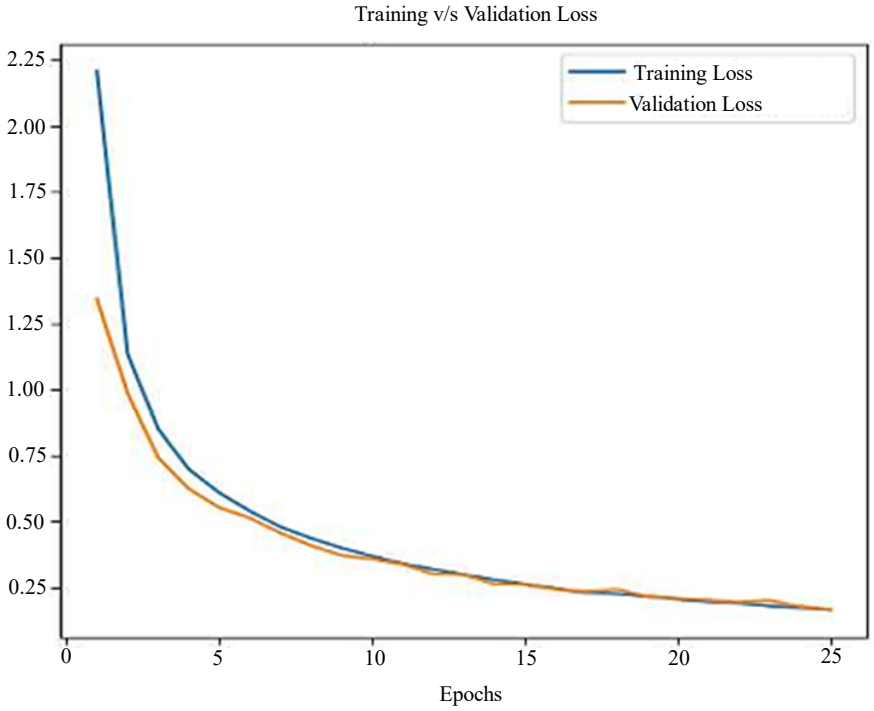

Figure 8. Training vs. validation loss plot.

The system interface for the LAN Video Call with Sign Language Detection demonstrated in Figure 10 features two video panels: one displaying the local user (normal user) and the other showing the remote user (deaf user) performing sign language gestures. It includes control buttons for initiating or ending the call, starting captioning, and enabling or disabling the sign language detection feature. A dedicated output section displays the recognized gestures, showing the current letter, dynamically forming words, and constructing meaningful sentences in real-time.

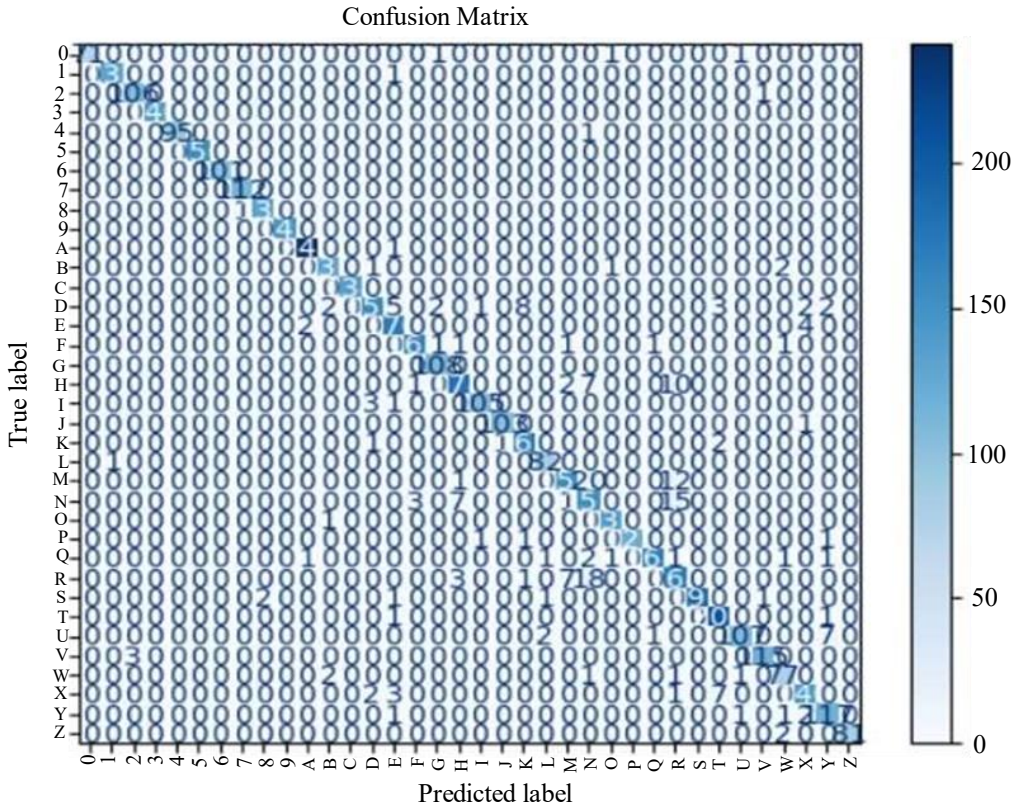

Figure 9. Confusion Matrix for true label vs. predicted label.

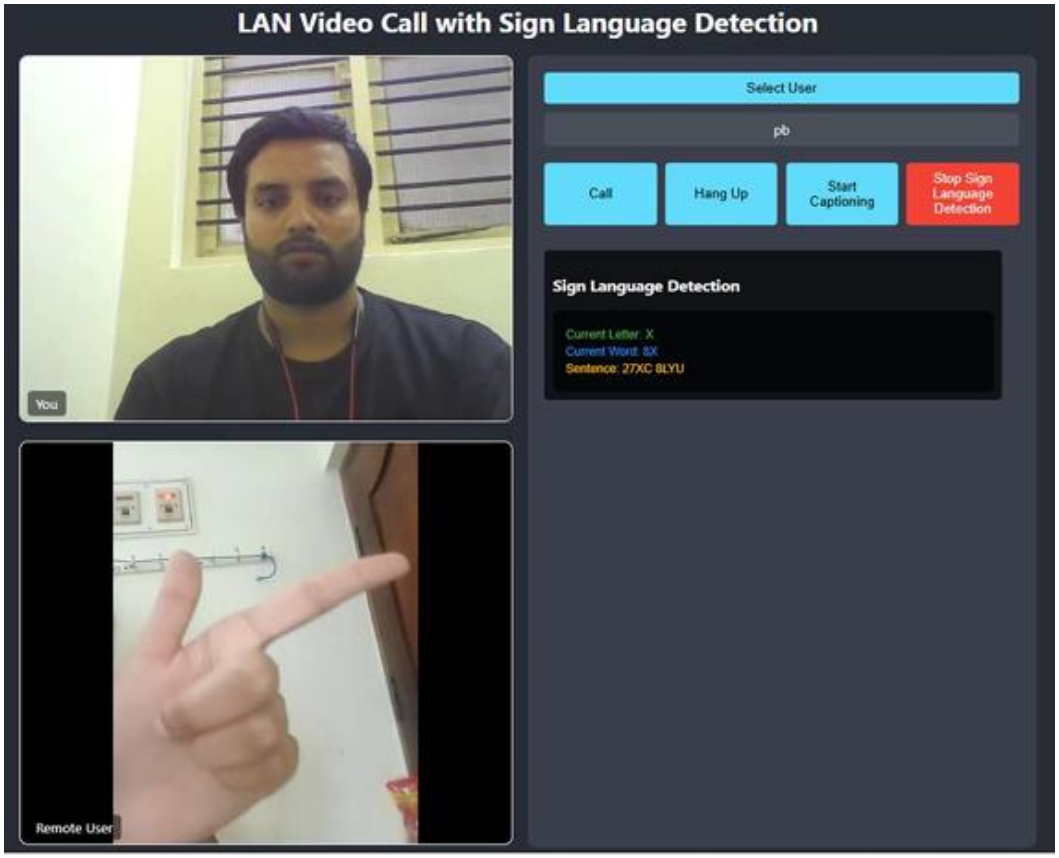

Figure 10. Real-time gesture recognition in LAN Video Call.

Figure 11. Real-time speech recognition in LAN Video Call.

As shown in Figure 11, the interface demonstrates the speech recognition functionality, where the speech provided by the local user (normal user) is converted into text in real-time for the remote user (deaf user). Below the panels, the recognized speech is displayed in text format, allowing the deaf user to read and comprehend the spoken communication.

CONCLUSION

The suggested system effectively tackles the communication barriers encountered by individuals with hearing impairments. By leveraging TensorFlow's CNN for gesture classification and MediaPipe for efficient hand tracking, the system achieves high accuracy, as evidenced by a training accuracy of 94.24% and a validation accuracy of 94.01%. The portable wristband device further enhances usability by providing tactile alerts for emergency notifications, ensuring accessibility even in noisy or visually inaccessible settings.

Unlike existing solutions reliant on internet connectivity or specialized hardware, this system operates independently of external networks, making it a cost-effective and scalable solution for diverse environments such as workplaces, public spaces, and educational institutions. By enabling smooth communication through sign language translation, it helps connect the deaf community with hearing individuals, fostering inclusivity and making services more accessible.

Future advancements in this system could include incorporating multiple sign languages to make the system versatile and globally applicable. The development of real time speech-to-sign translation by generating sign animations or visual gestures can make the system more adaptive. Furthermore, the system's performance can be significantly enhanced through Edge AI optimization, enabling efficient

operation on edge devices. This approach reduces latency and improves real-time responsiveness, ensuring seamless communication in environments with a dense network of interconnected devices. This work demonstrates the potential of combining machine learning and embedded systems to create impactful, user-centric solutions for underserved communities, paving the way for further innovations in assistive communication technologies.

REFERENCES

1. Meng Y, Jiang H, Duan N, Wen H. Real-Time Hand Gesture Monitoring Model Based on MediaPipe's Registerable System. Sensors. 2024 Sep 27; 24(19): 6262.
2. Moryossef A, Tsochantaridis I, Aharoni R, Ebling S, Narayanan S. Real-time sign language detection using human pose estimation. In Computer Vision ECCV 2020 Workshops: Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16. Springer International Publishing; 2020; 237–248.
3. 11Cao Z, Simon T, Wei SE, Sheikh Y. Realtime multi-person 2d pose estimation using part affinity fields. In Proceedings of the IEEE conference on computer vision and pattern recognition. 2017; 7291–7299.
4. Amutha S, Shanmukh N, Naidu AP, Kumar PV, Narayana GS. Real-Time Sign Language Recognition using a Multimodal Deep Learning Approach. In 2023 IEEE International Conference on Advances in Computing, Communication and Applied Informatics (ACCAI). 2023 May 25; 1–8.
5. Borg M, Camilleri KP. Sign language detection "in the wild" with recurrent neural networks. In ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). 2019 May 12; 1637–1641.
6. Huang J, Chouvatut V. Video-Based Sign Language Recognition via ResNet and LSTM Network. J Imaging. 2024 Jun; 10(6): 149.
7. Camgoz NC, Hadfield S, Koller O, Ney H, Bowden R. Neural sign language translation. In Proceedings of the IEEE conference on computer vision and pattern recognition. 2018; 7784–7793.
8. Adaloglou N, Chatzis T, Papastratis I, Stergioulas A, Papadopoulos GT, Zacharopoulou V, Xydopoulos GJ, Atzakas K, Papazachariou D, Daras P. A comprehensive study on deep learning-based methods for sign language recognition. IEEE Trans Multimedia. 2021 Apr 1; 24: 1750–62.
9. Miah AS, Hasan MA, Nishimura S, Shin J. Sign language recognition using graph and general deep neural network based on large scale dataset. IEEE Access. 2024 Mar 1; 12: 34553–34569.
10. Aggarwal D, Ahirwar S, Srivastava S, Verma S, Goel Y. Sign Language Prediction using Machine Learning Techniques: A Review. In 2023 IEEE 2nd International Conference on Electronics and Renewable Systems (ICEARS). 2023 Mar 2; 1296–1300.
11. Al Abdullah B, Amoudi G, Alghamdi H. Advancements in Sign Language Recognition: A Comprehensive Review and Future Prospects. IEEE Access. 2024 Sep 10; 12: 128871–128895.
12. Simon T, Joo H, Matthews I, Sheikh Y. Hand keypoint detection in single images using multiview bootstrapping. In Proceedings of the IEEE conference on Computer Vision and Pattern Recognition. 2017; 1145–1153.
13. Soni M, Bhat A, Aralikatti S, Pasha A, Niranjan L. An Efficient Digital Cluster Scheme to Improve the Lifetime Ratio of Wireless Sensor Networks. In 2023 IEEE International Conference on Smart Systems for applications in Electrical Sciences (ICSSES). 2023 Jul 7; 1–5.
14. Smilkov D, Thorat N, Assogba Y, Nicholson C, Kreeger N, Yu P, Cai S, Nielsen E, Soegel D, Bileschi S, Terry M. Tensorflow.js: Machine learning for the web and beyond. Proceedings of Machine Learning and Systems. 2019 Apr 15; 1: 309–21.
15. Fareed AI, Ramanathan M, Yeswanth R, Devi S. Translation Tool for Alternative Communicators using Natural Language Processing. In 2024 IEEE 5th International Conference on Electronics and Sustainable Communication Systems (ICESC). 2024 Aug 7; 842–848.
16. Miah AS, Hasan MA, Shin J, Okuyama Y, Tomioka Y. Multistage spatial attention-based neural network for hand gesture recognition. Computers. 2023 Jan 5; 12(1): 13.

17. Abdallah MS, Samaan GH, Wadie AR, Makhmudov F, Cho YI. Light-weight deep learning techniques with advanced processing for real-time hand gesture recognition. Sensors. 2022 Dec 20; 23(1): 2.
18. Koller O, Zargaran S, Ney H, Bowden R. Deep Sign: Hybrid CNN-HMM for Continuous Sign Language Recognition. In Proceedings of the British Machine Vision Conference (BMVC). 2016 Sep 19; 136.1–136.12.
19. Premsai I, Thiyagu TM. IoT based Wireless Alert System for Individuals with Impaired Hearing. In2024 3rd International Conference on Sentiment Analysis and Deep Learning (ICSADL). 2024 Mar 13; 662–666.
20. Monteiro CD, Mathew CM, Gutierrez-Osuna R, Shipman F. Detecting and identifying sign languages through visual features. In 2016 IEEE International Symposium on Multimedia (ISM). 2016 Dec 11; 287–290.
21. Joshi A, Agrawal S, Modi A. ISLTranslate: Dataset for Translating Indian Sign Language. arXiv preprint arXiv:2307.05440. 2023 Jul 11.

