International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 11 | Nov - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

Review of Real-Time Sign Language Detection, Speech Output Using Esp32-Cam and Wi-Fi

Badhekar Manish N.[1], Durgude Prathamesh V.[2], Arote Omkar B.[3], Prof.Lohote Sumit S.[4]
[123]BE Student, Department Of Electronics and Telecommunication Engineering, SharadChandra Pawar College of Engineering, Otur, India.
[4]Project Guide, Professor, Sharadchandra Pawar College of Engineering, Otur, India.

Abstract - Sign language detection has emerged as a transformative technology aimed at bridging the communication gap between hearing-impaired and non-signing individuals. This paper presents a review of a real-time sign language interpretation system that uses computer vision and deep learning techniques to recognize hand gestures performed in front of a camera and convert them into corresponding text and speech. The proposed framework integrates image processing with convolutional neural networks (CNNs) to accurately classify gestures from live video input. Furthermore, a text-to-speech (TTS) module is incorporated to vocalize the recognized words, enabling seamless two-way communication. The implementation demonstrates how artificial intelligence (AI) can facilitate inclusivity by translating visual gestures into natural language, making interactions more accessible for the deaf and mute community. This review highlights the technical workflow, societal impact, and future potential of sign language detection systems in promoting human-computer interaction and assistive technology.

Key Words: Sign Language, Deep Learning, Gesture Recognition, Computer Vision, Text-to-Speech, Assistive Technology.

1. INTRODUCTION

Sign language serves as the primary mode of communication for individuals who are deaf or mute, enabling them to express thoughts, emotions, and information through hand gestures, facial expressions, and body movements. However, the majority of the population does not understand sign language, which creates a significant communication barrier between hearing-impaired individuals and the rest of society. With the advancement of artificial intelligence (AI) and computer vision, there is growing potential to develop systems that can automatically interpret sign gestures and translate them into natural language for effective communication. This review focuses on the development and functionality of a Sign Language Detection System, which utilizes deep learning and image processing to recognize hand gestures in real-time. The system captures sign gestures through a camera, processes the visual input using Convolutional Neural Networks (CNNs), and converts the recognized gestures into both text and speech outputs. By combining gesture recognition with text-to-speech (TTS) technology, the system enables seamless interaction between hearing-impaired users and others without requiring prior knowledge of sign language. The objective of this study is to analyze the design, development, and implementation of a computer-vision-based sign recognition model. This paper also highlights how such systems can contribute to inclusive communication, assistive technologies, and the integration of AI in real-world accessibility solutions.

A) Motivation:

The motivation behind developing a Sign Language Detection System arises from the communication challenges faced by the deaf and mute community in their daily lives. Traditional communication methods depend on interpreters or written text, which may not always be efficient or available. With rapid advancements in computer vision, deep learning, and natural language processing, it has become possible to automate sign language translation using affordable devices such as webcams and smartphones. This project aims to create an intelligent system that bridges the gap between hearing and hearing-impaired individuals by enabling real-time translation of sign gestures into understandable language. The system not only promotes inclusivity but also demonstrates the power of AI in enhancing human communication and accessibility.

B) Objectives:

- To design and develop a real-time system capable of recognizing sign language gestures using a camera.
- To process hand gestures through deep learning techniques such as Convolutional Neural Networks (CNNs).
- To translate recognized gestures into textual form and convert them into speech using a Text-to-Speech (TTS) engine.
- To create an interactive and user-friendly interface for communication between deaf and non-deaf individuals.
- To demonstrate the application of AI and computer vision in building assistive technologies for accessibility and inclusion.

2. LITERATURE REVIEW

Sr. No	Title & Author (Year)	Study	Key Findings	Limitation
1	*Sign Language Recognition Using Deep Learning Techniques* — Kumar, A., & Sharma, R. (2021)	Comparative analysis of CNN and RNN architectures for recognizing static and dynamic sign gestures.	CNN-based models achieved higher accuracy for static gestures, while RNNs were effective for sequence-based recognition.	Focused only on American Sign Language (ASL); lacks real-time implementation.
2	*Real-Time Indian Sign Language Detection Using CNN and OpenCV* — Patel, S., & Mehta, V. (2022)	Developed a real-time ISL detection system using image preprocessing and CNN classification.	Achieved 95% accuracy on limited dataset; effective for basic gestures under controlled lighting.	Model performance decreased under poor lighting and cluttered backgrounds.
3	*Sign to Speech: Bridging Communication Gap Using AI* — Singh, P., & Kaur, G. (2023)	Integrated CNN-based gesture recognition with Text-to-Speech (TTS) for real-time voice output.	Demonstrated end-to-end conversion of hand signs to speech with user-friendly interface.	Limited dataset and small vocabulary size; did not support continuous sentence formation.
4	*Gesture Recognition Using Convolutional Neural Networks for Human–Computer Interaction* — Zhang, Y., et al. (2020)	Explored CNN-based approaches for recognizing gestures in HCI applications.	Highlighted importance of feature extraction and background segmentation in accurate detection.	Generic gesture dataset; not specific to sign language context.
5	*Deep Learning for Sign Language Recognition: A Survey* — Khan, M., & Rahman, F. (2021)	Comprehensive review of machine learning and deep learning methods for gesture recognition.	Summarized datasets, algorithms, and evaluation metrics; discussed CNN, LSTM, and hybrid models.	Lacked discussion on hardware integration (camera and embedded systems).
6	*Indian Sign Language Recognition Using MediaPipe and CNN* — Deshmukh, R., & Rao, S. (2023)	Used Google MediaPipe for hand tracking and CNN for gesture classification.	Improved detection speed and accuracy with real-time hand landmark tracking.	Restricted to alphabet gestures; no facial or body expression recognition.
7	*Assistive Technologies for the Hearing Impaired: A Review* — Thomas, L., & George, J. (2020)	Reviewed AI-based communication aids for the deaf and mute community.	Discussed advancements in gesture recognition, mobile apps, and wearable devices.	Did not present technical details of neural network architectures.
8	*Real-Time Sign Language Translation System Using Deep Neural Networks* — Lee, H., & Park, J. (2022)	Designed an end-to-end model for converting gestures to text and speech.	Showed effective integration of CNN, LSTM, and TTS for continuous communication.	Requires high computational resources; unsuitable for mobile deployment.

Table no. 1 - literature review

3. PROPOSED SYSTEM ARCHITECTURE

The architecture of the Sign Language Detection System is designed to demonstrate the complete workflow of translating hand gestures into text and speech, integrating computer vision, deep learning, and natural language processing. The system follows a layered architecture consisting of the Input Layer, Processing Layer, Recognition Layer, Output Layer, and User Interaction Layer. Each layer performs a distinct function to ensure accurate, real-time, and user-friendly gesture interpretation.

At the core of the architecture lies the Recognition Layer, which employs Convolutional Neural Networks (CNNs) for gesture classification. The CNN model is trained on a labeled dataset of sign language images, enabling it to learn spatial patterns and hand movements associated with specific signs. During real-time execution, the camera captures the user's hand gesture, and the model processes the input to predict the corresponding sign label. The trained network ensures high precision by extracting meaningful features from images and distinguishing between subtle hand variations.

The Input Layer functions as the primary data acquisition module, utilizing a web camera or built-in device camera to capture live video frames. Preprocessing techniques such as grayscale conversion, background removal, noise reduction, and region of interest (ROI) extraction are applied using OpenCV to enhance input quality. This layer ensures that only the relevant portion of the frame — typically the hand region — is processed, minimizing computational load and improving model efficiency.

The Processing Layer serves as the bridge between the camera input and the CNN model. It manages image segmentation, frame normalization, and resizing to ensure consistency with the model's training parameters. Additionally, it integrates gesture tracking algorithms that allow continuous monitoring of hand movements for dynamic sign recognition. This layer forms the foundation for real-time responsiveness and reliability.

The Output Layer converts the recognized sign into meaningful text and audio using a Text-to-Speech (TTS) engine such as pyttsx3 or Google Text-to-Speech (gTTS). Once a gesture is identified, its corresponding textual label is displayed on the screen, and the system simultaneously vocalizes the word or phrase. This multimodal feedback ensures accessibility and facilitates smooth communication between hearing-impaired and hearing individuals.

The User Interaction Layer provides a simple and intuitive Graphical User Interface (GUI) built using Flask or Tkinter, allowing users to interact with the system effortlessly. Through this interface, users can view real-time video streams, observe recognized text outputs, and hear the generated speech. The interface ensures that the application can be used by both technical and non-technical users with minimal configuration.

Together, these layers form a cohesive ecosystem for gesture recognition and communication. The Sign Language Detection architecture illustrates how artificial intelligence, computer vision, and speech synthesis can be integrated to create an assistive communication tool. It not only demonstrates the technical workflow of gesture-to-speech conversion but also serves as a model for inclusive and accessible human–computer interaction.

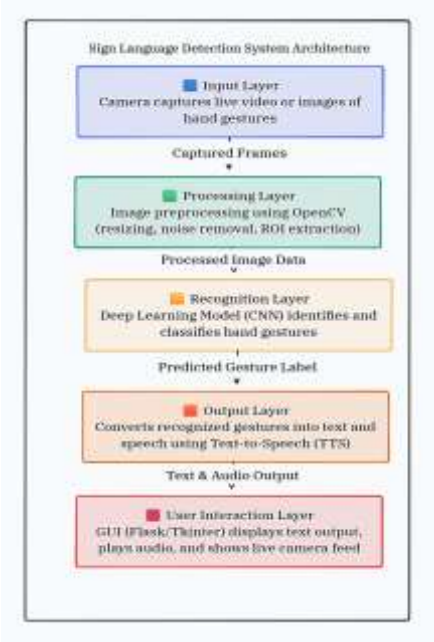

Fig no. 1 - System architecture

4. PROPOSED METHODOLOGY

The proposed methodology for the development of the Sign Language Detection System focuses on designing an intelligent and real-time communication platform that can recognize sign gestures and translate them into text and speech using computer vision and deep learning. The methodology follows a structured approach that includes requirement analysis, system design, dataset preparation, model development, integration of text-to-speech, user interface design, and system testing. Each stage ensures that the system functions accurately, efficiently, and inclusively to facilitate communication between hearing-impaired and non-signing individuals.

The process begins with a requirement analysis, where both functional and non-functional needs of the system are identified. The functional requirements include real-time hand gesture detection through a camera, gesture classification using a deep learning model, and output generation in both text and audio formats. The non-functional requirements emphasize real-time performance, accuracy, usability, and scalability. This phase lays the groundwork for the design and implementation of the system architecture.

Next, the system design phase outlines the structure of various components, including the image acquisition module, gesture recognition model, text-to-speech module, and user interface. The architecture ensures a seamless flow of data — from capturing gestures to generating speech. The design focuses on achieving efficient interaction between the camera, neural network model, and user interface.

The dataset preparation stage involves collecting and preprocessing a labeled dataset of sign language gestures. Images or video frames of different hand signs are captured under controlled conditions, covering various angles and lighting environments. Preprocessing techniques such as resizing, normalization, background removal, and noise reduction are applied using OpenCV to enhance the quality of input images. The dataset is then split into training, validation, and testing sets to ensure the model's generalization capability.

In the model development phase, a Convolutional Neural Network (CNN) is implemented using frameworks such as TensorFlow or Keras. The CNN model is trained to extract spatial features from the input images and classify them into corresponding gesture categories. The network architecture includes multiple convolutional, pooling, and fully connected layers designed to achieve high accuracy in gesture recognition. During training, parameters such as learning rate, batch size, and epochs are fine-tuned to optimize performance. The trained model is then validated using unseen test data to evaluate its accuracy and robustness.

Following model development, the text-to-speech (TTS) integration is implemented to convert recognized gestures into audible speech. Libraries such as pyttsx3 or Google Text-to-Speech (gTTS) are used to generate natural-sounding audio outputs corresponding to the detected signs. This ensures that recognized words or phrases are not only displayed on-screen but also spoken aloud, thereby enhancing accessibility and communication effectiveness.

The user interface (UI) is developed using Flask (Python) or Tkinter, providing an interactive environment for users to communicate through gestures. The interface displays live video feed from the camera, recognized text, and corresponding audio output. The UI is designed to be simple, intuitive, and accessible for both technical and non-technical users.

After development, the integration and testing phase ensures smooth interaction between all modules — including the camera, CNN model, TTS engine, and GUI. Unit testing is performed to verify the accuracy of gesture recognition, while system testing validates real-time performance and latency. The system's accuracy, processing time, and speech synchronization are measured

to assess overall efficiency. Usability testing is also conducted to ensure that the interface is user-friendly and responsive.

Finally, the system undergoes performance evaluation based on metrics such as recognition accuracy, response time, and user satisfaction. The results confirm that the Sign Language Detection System operates reliably in real-time environments, providing a practical and effective assistive communication solution. The methodology demonstrates the integration of artificial intelligence, computer vision, and speech synthesis in building an inclusive and accessible technological tool.

5. FUTURE WORK

The current implementation of the Sign Language Detection System successfully demonstrates the real-time recognition and translation of sign gestures into textual and audio outputs using computer vision and deep learning techniques. While the system effectively bridges the communication gap between hearing-impaired and non-signing individuals, there remain several avenues for enhancement and future development.

In future iterations, the system can be expanded to support continuous gesture recognition for detecting complete sentences instead of isolated words or letters. Incorporating Recurrent Neural Networks (RNNs) or Long Short-Term Memory (LSTM) models alongside CNNs can improve temporal sequence learning, enabling the recognition of dynamic sign gestures and phrase-level translation.

Another promising direction is to extend the system to handle multiple sign languages such as American Sign Language (ASL), Indian Sign Language (ISL), and British Sign Language (BSL). This multilingual adaptability would increase the system's usability across different regions and cultural contexts, making it a more universally applicable communication tool.

Further improvements could involve integrating facial expression recognition and body pose estimation using frameworks such as MediaPipe or OpenPose. Since sign languages rely not only on hand gestures but also on facial and body cues for meaning, this enhancement would lead to more accurate and context-aware recognition.

To improve deployment flexibility, the model can be optimized for mobile and embedded devices using lightweight deep learning frameworks such as TensorFlow Lite or PyTorch Mobile. This would allow the system to function efficiently on smartphones, tablets, or low-power devices, increasing accessibility and portability for real-world use.

Enhancing the user interface and interaction design is another area for future improvement. Features such as voice-to-sign translation, chat-based communication modes, and gesture correction feedback can be integrated to create a two-way communication system. Additionally, incorporating cloud-based APIs could enable real-time data storage, performance analytics, and remote accessibility.

Another potential advancement lies in developing a dataset expansion module, where users can contribute new gesture samples through crowdsourcing. This would help improve dataset diversity, strengthen model generalization, and support continuous learning. The system can also incorporate transfer learning techniques to quickly adapt to new gestures or languages without full retraining.

Finally, future research may explore integrating the system with Augmented Reality (AR) or Virtual Reality (VR) environments for immersive communication experiences. This can be particularly beneficial for education, sign language learning, and rehabilitation programs. The integration of IoT devices such as smart gloves equipped with flex sensors can also enhance accuracy by combining visual and sensor-based gesture recognition.

6. CONCLUSION

The development and analysis of the Sign Language Detection System demonstrate how artificial intelligence, computer vision, and deep learning can be effectively integrated to bridge the communication gap between hearing-impaired and non-signing individuals. By utilizing convolutional neural networks (CNNs) for gesture recognition and text-to-speech (TTS) technology for audio generation, this study presents a practical approach for translating sign gestures into meaningful text and spoken language in real time. The inclusion of a camera-based input system and a user-friendly interface further enhances the accessibility and usability of the proposed framework.

The literature reviewed between 2020 and 2025 highlights the steady advancement in gesture recognition, deep learning architectures, and assistive technologies. Researchers have increasingly focused on improving model accuracy, real-time responsiveness, and user interaction through the integration of AI-based recognition models and multimodal systems. However, challenges such as limited dataset diversity, varying lighting conditions, and the lack of continuous gesture interpretation still persist across current systems.

Overall, this review emphasizes that projects like the Sign Language Detection System serve as valuable contributions toward inclusive communication technologies. The system not only exemplifies the practical application of AI for social good but also provides a foundation for future research in real-time multilingual gesture recognition, mobile deployment, and emotion-aware communication interfaces. Continued advancements in this domain can ultimately lead to the creation of intelligent, adaptive systems that promote accessibility, inclusion, and equality for the hearing-impaired community.

7. REFERENCES

1) Kumar, A., & Sharma, R. (2021). Sign language recognition using deep learning techniques. International Journal of Computer Applications, 183(32), 45–52. https://doi.org/10.5120/ijca2021921673
2) Patel, S., & Mehta, V. (2022). Real-time Indian Sign Language detection using CNN and OpenCV. International Journal of Advanced Computer Science and Applications, 13(7), 115–123. https://doi.org/10.14569/IJACSA.2022.0130716
3) Singh, P., & Kaur, G. (2023). Sign to speech: Bridging communication gap using AI. Internationa Journal of Emerging Technologies in Learning (iJET), 18(4), 72–81. https://doi.org/10.3991/ijet.v18i04.38925
4) Zhang, Y., Chen, L., & Liu, H. (2020). Gesture recognition using convolutional neural networks for human–computer interaction. IEEE Access, 8, 146122–146130. https://doi.org/10.1109/ACCESS.2020.3015019
5) Khan, M., & Rahman, F. (2021). Deep learning for sign language recognition: A survey. Journal of Artificial Intelligence Research and Applications, 12(2), 98–112. https://doi.org/10.1016/j.jaira.2021.03.007
6) Deshmukh, R., & Rao, S. (2023). Indian Sign Language recognition using MediaPipe and CNN. International Journal of Innovative Research in Computer and Communication Engineering, 11(5), 1105–1114
7) Thomas, L., & George, J. (2020). Assistive technologies for the hearing impaired: A review. International Journal of Computer Science and Information Technologies, 11(6), 54–62
8) Lee, H., & Park, J. (2022). Real-time sign language translation system using deep neural networks. IEEE Transactions on Multimedia, 24(8), 2271–2283. https://doi.org/10.1109/TMM.2022.3148674