

Implementation of Communication through Gesture and Speech Synthesis

Juguntla Srilekha^{1*}, Kammari Akshaya¹, Siragam Venkata Krishna¹, P. Durga Devi², K. Raghu²

¹ UG Student, Department of ECE, Mahatma Gandhi Institute of Technology, Hyderabad, Telangana, India.

² Assistant Professor, Department of ECE, Mahatma Gandhi Institute of Technology, Hyderabad, Telangana, India.

jsrilekha_ece220495@mgit.ac.in, kakshaya_ece220497@mgit.ac.in,

svenkatakrishna_ece2204b9@mgit.ac.in, pdurgadevi_ece@mgit.ac.in.

Abstract - Communication is one of the most essential aspects of human interaction; however, individuals who cannot speak and hear often face significant challenges in expressing themselves and understanding others. This project aims to develop a real-time interactive communication system that enables smooth and natural conversation between speaking and hearing, non-speaking and non-hearing individuals. The system translates spoken language into sign language for the non-hearing user and converts hand gestures into text and speech for the hearing user, thereby establishing a two-way communication link. The proposed model is implemented using a Raspberry Pi as the central processing unit and employs TensorFlow for gesture recognition along with OpenCV for image capture and processing. Hand gestures captured through a webcam are translated into meaningful text or audio output, while speech inputs are processed and displayed as corresponding text or sign animations. This ensures accurate and real-time communication suitable for classrooms, workplaces, and public environments. By integrating artificial intelligence, computer vision, and embedded technology, the project bridges the communication gap and promotes social inclusion for individuals who cannot speak and hear.

Key Words: Sign Language Recognition, Gesture Recognition, Real-Time Communication, Raspberry Pi, Computer Vision, Assistive Technology

1. INTRODUCTION

Communication is one of the most essential aspects of human life, enabling individuals to express ideas, emotions, and information effectively. However, individuals with speech and hearing impairments face significant challenges in communication, leading to social barriers. Sign language serves as an effective medium, using hand gestures, facial expressions, and body movements. Linguistic studies confirm that sign language is a structured language with its own grammar, as highlighted by Klima and Bellugi [27] and Liddell and Johnson [28].

From a non-technical perspective, communication for deaf and mute individuals traditionally relied on interpreters or manual learning, which lacked scalability and real-time accessibility. To address this, assistive technologies such as glove-based systems [2], [3] and Raspberry Pi-based devices

[4] were introduced. Although effective, these systems required wearable hardware, reducing convenience and usability.

The evolution of gesture recognition began with studies on biological motion by Johansson [12], followed by structured gesture models by Bobick and Wilson [13]. Early vision-based systems, such as those by Starner et al. [16], enabled real-time gesture recognition using video data. Probabilistic approaches like Hidden Markov Models (HMMs) [17], [18] and Parallel HMMs [19], [20], along with extensions for large vocabulary recognition [30], further improved the modeling of dynamic gestures.

With advancements in computer vision, traditional techniques such as motion detection [23], segmentation [24], [25], and tracking methods like condensation [21] and recursive estimation [22] laid the foundation for modern systems. Additionally, event classification approaches based on probabilistic modeling [26] contributed to improved interpretation of gesture sequences. These methods enabled effective feature extraction from visual data.

From a technical perspective, gesture recognition involves analyzing spatial and temporal features. Machine learning approaches [11], along with deep learning models such as CNNs [5] and LSTMs [10], have significantly improved recognition accuracy. Modern frameworks like MediaPipe [1] and landmark-based methods [7] enable efficient real-time detection, while multi-view approaches [8] enhance robustness. Research on phoneme-level decomposition of signs [29] has further contributed to scalability in recognition systems. Recent assistive systems integrate gesture recognition with speech processing, enabling bidirectional communication through sign-to-speech conversion [6]. However, challenges such as environmental sensitivity, dataset limitations, and complex gesture handling still exist [9]. The proposed system focuses on a real-time, vision-based approach that eliminates wearable devices. It captures gestures using a camera and converts them into text or speech output, while also supporting speech-to-text mapping. The system follows a standard pipeline of image acquisition, preprocessing, feature extraction, and classification [1].

Overall, gesture recognition has evolved from basic motion analysis to advanced AI-based systems. Despite significant

progress, there remains a need for cost-effective, scalable, and robust solutions for real-world applications.

1.1 Related Works

Sign language recognition has been explored using sensor-based, vision-based, and machine learning approaches. Early studies by Johansson [12] and Bobick and Wilson [13] laid the foundation for gesture analysis and structured representation. Probabilistic models such as Hidden Markov Models (HMMs) were widely used for modeling temporal gestures [17], [18], with further improvements using parallel and scalable approaches [19], [29], [30].

Sensor-based systems like Glove-Talk II [15] and modern smart gloves [2], [3] provided accurate gesture-to-speech conversion but required wearable hardware. Raspberry Pi-based assistive devices [4] improved portability, though usability remained limited.

Vision-based methods gained popularity due to their non-intrusive nature. Early work by Stamer et al. [16] demonstrated real-time recognition using video, while recent approaches using MediaPipe and machine learning classifiers have improved efficiency and accuracy [1]. Deep learning models such as CNNs [5] and LSTMs [10], along with landmark-based [7] and multi-view techniques [8], further enhanced performance.

Recent systems integrate gesture recognition with speech processing for assistive communication [6]. However, challenges such as environmental sensitivity, hardware dependency, and difficulty in recognizing complex dynamic gestures still persist [9], [11].

In addition, traditional computer vision techniques such as motion detection [23], segmentation [24], [25], and tracking methods like condensation [21] and recursive estimation [22] have played a crucial role in the development of modern gesture recognition systems. Event classification approaches [26] further improved the interpretation of gesture sequences. These foundational methods continue to support current AI-based models by enabling effective feature extraction and robust performance in real-time environments.

Overall, existing methods show good progress but still face issues like hardware dependency and environmental sensitivity. This highlights the need for a simple, real-time, and more reliable vision-based solution. Additionally, improving accuracy for dynamic and continuous gestures remains an important area for further research.

2. DATA AND METHODS

2.1 Data Acquisition (Input Layer)

The system begins by capturing real-time video input using a camera interface through OpenCV. The camera is initialized at a resolution of 640×480 to balance performance and processing speed. Once initialized, the system continuously captures frames at approximately 15–20 frames per second on Raspberry Pi 4.

Each captured frame is converted from BGR to RGB format, which is required for MediaPipe processing. This stage ensures that the input data is properly formatted and ready for further analysis in subsequent stages.

2.2 Visual Processing (Hand Tracking Layer)

In this stage, the system performs hand detection and landmark extraction using the MediaPipe Hands model (Lite version for CPU efficiency). Each RGB frame is processed to detect hand presence and extract 21 three-dimensional landmark points representing key positions of the hand.

If no hand is detected, the system resets the input buffer and continues capturing frames. If a hand is detected, the (x, y, z) coordinates of all landmarks are extracted and forwarded to the next stage. This step forms the foundation for accurate gesture recognition.

Table 1 : Feature Engineering Process

Step	Operation	Description
1	Translation Invariance	Subtract wrist coordinates (Point 0) to center data
2	Scale Normalization	Normalize using maximum distance from wrist
3	Flattening	Convert landmarks into 63-dimensional vector

This normalization ensures that the system is invariant to position and size variations, thereby improving recognition accuracy.

2.3 Prediction Logic (Inference Layer)

The system uses a dual recognition approach to handle both static and dynamic gestures efficiently.

Static Gesture Recognition

The current feature vector is compared with stored gesture data using Euclidean distance. If the computed distance is less than a predefined threshold (0.35), the gesture is classified as a static gesture. This method is fast and suitable for simple hand poses.

Dynamic Gesture Recognition

For dynamic gestures, the system analyzes motion over time. A sliding window buffer stores feature vectors for approximately 45 frames (around 3 seconds). The system computes motion levels based on landmark variations.

Table 2: Dynamic Gesture Processing

Step	Operation	Description
1	Buffering	Store feature vectors in sliding window
2	Motion Analysis	Compute variation of landmarks over time

3	Threshold Check	Detect significant motion
4	Subsampling	Reduce 45 frames to 10 keyframes
5	Matching	Compare with trained motion dataset

This approach enables recognition of complex gestures involving movement.

2.4 Decision and Output (Action Layer)

Before generating output, the system applies validation mechanisms to ensure reliability. A stability check ensures that the predicted gesture remains consistent for at least three consecutive frames. Additionally, state management prevents repeated outputs when the same gesture is continuously detected.

Once validated, the system produces output in two forms:

- **Audio Output:** The recognized gesture is converted into speech using a text-to-speech engine.
- **Visual Output:** The gesture label and landmarks are displayed on the screen in real time.

The system then loops back to the input stage, enabling continuous real-time operation.

2.5 System Workflow

The system operates as a continuous pipeline from gesture capture to output generation. Gestures are processed and classified using both static and dynamic methods, while validation ensures accuracy and stability. The final output combines audio and visual feedback to improve usability.

Overall, the system achieves efficient real-time gesture recognition with reliable performance on embedded platforms such as Raspberry Pi.

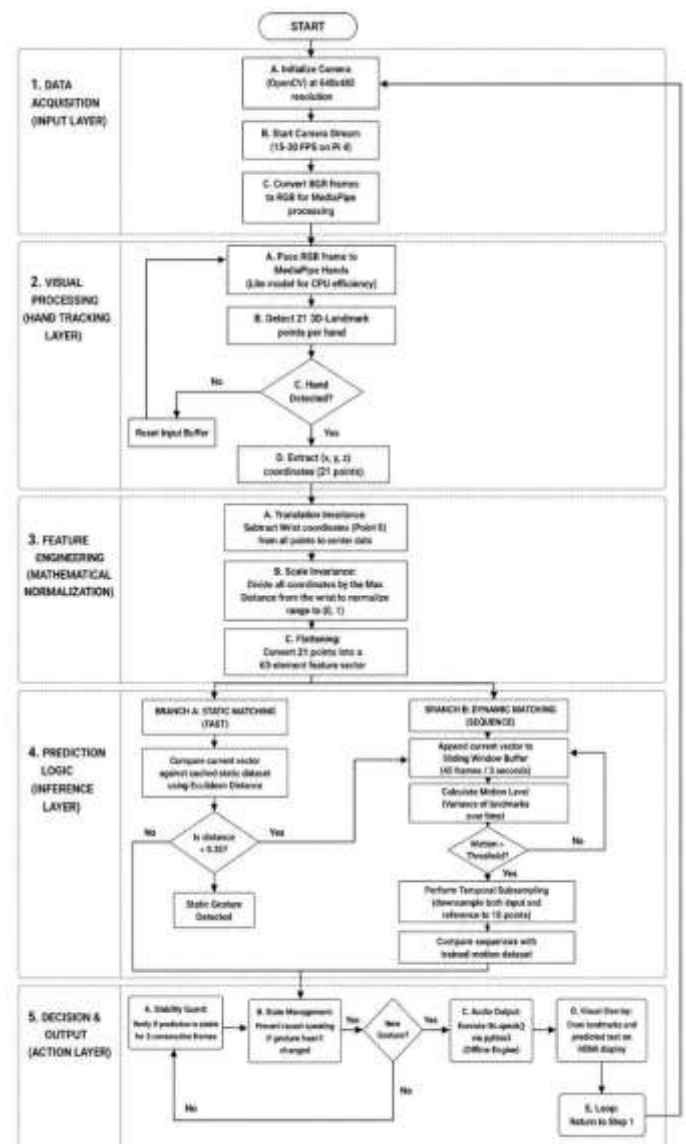


Fig 1: Sign Language Conversion Cycle

The fig1 illustrates the overall working of the proposed sign language gesture recognition system in a step-by-step manner. It represents the complete pipeline starting from data acquisition to final output generation. Each stage in the flowchart corresponds to a specific functional layer of the system, including input capture, hand landmark detection, feature normalization, gesture prediction, and output generation.

The fig1 also highlights the decision-making process involved in distinguishing between static and dynamic gestures, along with validation steps such as stability checking and motion analysis. This visual representation helps in understanding the sequential data flow, system logic, and interaction between different modules, ensuring clarity in the implementation of the real-time recognition system.

3. RESULTS & DISCUSSION

The system detects a hand gesture and displays the corresponding sign name using a trained recognition model. If the same gesture is maintained continuously for 5 seconds, it confirms the input and produces the final output .

3.1 Performance Metrics

The system performance is evaluated using standard metrics:

Accuracy: Percentage of correctly classified gestures

Confusion Matrix: Shows classification performance across classes .

Latency: Time taken to predict gesture

Frame Rate (FPS): Number of frames processed per second.

3.2 Accuracy Analysis

The trained Random Forest model achieved high accuracy in recognizing predefined gestures under controlled conditions.

Table 3: Accuracy Performance

Parameter	Value
Training Accuracy	~95–98%
Testing Accuracy	~90–95%
Real-Time Accuracy	~88–92%

The slight drop in real-time accuracy is due to environmental variations and hardware constraints.

3.3 Confusion Matrix Analysis

The confusion matrix shows how well the system distinguishes between different gestures. Most gestures are correctly classified, while minor misclassifications occur between visually similar gestures.

Observations:

- High diagonal values indicate correct predictions
- Errors occur in gestures with similar hand shapes
- Overall classification performance is satisfactory

3.4 Real-Time Performance

The system was tested in real-time conditions to evaluate responsiveness and usability.

Table 4 : Real-Time Performance

Parameter	Laptop	Raspberry Pi
Frame Rate (FPS)	25–30 FPS	15–20 FPS
Latency	Low (~100 ms)	Moderate (~200–300 ms)

Response Time	Fast	Slightly delayed
---------------	------	------------------

The system performs smoothly on both platforms, with minor delays on Raspberry Pi due to limited processing power.

3.5 Outputs

The system successfully detects and displays gesture labels in real time.



[a] Recognition of the sign “thankyou” [b] Recognition of the sign “help”

The experimental results indicate that:

- Real-time gesture classification is achieved using normalized features.
- Hand landmarks are accurately detected and utilized for spatial modeling.
- The system operates with low latency through continuous frame processing.
- Feature engineering enhances robustness and generalization.

The output includes:

- Gesture label displayed on screen.
- Hand landmark visualization.

The results confirm that the system provides accurate and continuous gesture recognition.

3.6 Hardware Implementation Results

The Raspberry Pi-based system was tested for real-world usability.

Observations:

- System runs successfully as a standalone device
- Provides real-time gesture recognition with acceptable speed
- Suitable for portable assistive applications
- Performance slightly affected under low-light conditions

4. CONCLUSION & PERSPECTIVES

This paper presents a real-time, vision-based sign language gesture recognition system designed to assist communication for individuals with speech and hearing impairments. The system captures hand gestures using a camera, extracts hand landmarks through MediaPipe, and converts them into feature vectors for classification using a Random Forest model.

A complete real-time gesture recognition pipeline was developed, achieving accurate hand landmark detection and

classification. The system enables low-latency gesture-to-text conversion and is designed to be scalable for efficient deployment on embedded hardware platforms..

The system currently supports only a limited set of static gestures and is sensitive to variations in lighting conditions, background complexity, and occlusions. Additionally, limited dataset diversity affects the model's generalization capability across different users and environments. Furthermore, the system does not fully support dynamic gestures or continuous sign language communication.

The proposed approach successfully achieves real-time gesture recognition and displays outputs as text using OpenCV, demonstrating the effectiveness of a non-intrusive, camera-based solution that eliminates the need for wearable devices. The system is cost-effective, portable, and modular, allowing integration with speech processing and embedded platforms.

Overall, the system demonstrates a practical and scalable assistive communication solution, with strong potential for real-world deployment and future advancement.

Future enhancements include expanding the gesture dataset, enabling dynamic and sentence-level recognition using advanced models, improving robustness under varying conditions, and integrating mobile/cloud platforms. Additional features such as text-to-speech and speech-to-sign conversion can further enable complete bidirectional communication.

REFERENCES

- [1] G. Sundar, M. J. Paulraj, and S. Palanisamy, "Real-Time Recognition of American Sign Language Using MediaPipe and Machine Learning Techniques," *Procedia Computer Science*, vol. 172, 2020, Article ID S1877050920300880. doi:10.1016/j.procs.2020.03.266. Full text: <https://www.sciencedirect.com/science/article/pii/S1877050920300880?via%3Dihub>
- [2] Shravani Jadhav, Siddhi Mhatre, Riddhi Patil, and Prof. Prachi Sorte, "Smart Hand Gloves for Deaf and Dumb People," *International Journal for Research in Applied Science & Engineering Technology (IJRASET)*, vol. 12, issue III, March 2024. Available: <https://www.ijraset.com/best-journal/smart-hand-gloves-for-deaf-and-dumb-people.ijraset.com>
- [3] H. Almshjary, A. Alghamdi, A. Barsheed, D. Alahmadi, O. Alzamzami, H. Bitar & A. Babour, "Intelligent Gloves: An IT intervention for deaf-mute people," *Journal of Intelligent Systems*, vol. 32, no. 1, 2023. Available: https://www.researchgate.net/publication/368831568_Intelligent_gloves_An_IT_intervention_for_deaf-mute_people
- [4] Anusha B. Ballary and Ganashree T. S., "Raspberry Pi based Assistive Device for Communication between Blind, Deaf, and Dumb People," *International Journal for Research Trends and Innovation (IJRTI)*, vol. 7, issue 9, 2022. Available: <https://ijrti.org/papers/IJRTI2209032.pdf>
- [5] Salunke, D., Joshi, R., Ranjan, N., Tekade, P., & Panchal, G. (2023). Sign Language Recognition System Using Customized Convolution Neural Network. *Proceedings of International Conference on Data Science and Applications, Lecture Notes in Networks and Systems*, 552, 825-837. Springer Nature Singapore. PDF available at: https://www.researchgate.net/publication/368396468_Sign_Language_Recognition_System_using_Customized_Convolution_Neural_Network
- [6] Kaur, B., Chaudhary, A., Bano, S., Yashmita, Reddy, S. R. N., & Anand, R. (2024). Fostering inclusivity through effective communication: Real-time sign language to speech conversion system for the deaf and hard-of-hearing community. *Multimedia Tools and Applications*, 83, 45859–45880. Available: <https://link.springer.com/article/10.1007/s11042-024-18667-3>
- [7] Gil-Martín, M., Marini, M. R., Martín-Fernández, I., Esteban-Romero, S., & Cinque, L. (2025). Hand Gesture Recognition Using MediaPipe Landmarks and Deep Learning Networks. In *Proceedings of the 17th International Conference on Agents and Artificial Intelligence (ICAART 2025)*, pp. 24–30. DOI: 10.5220/0013053500003890. PDF available at: <https://www.scitepress.org/Papers/2025/130535/130535.pdf>
- [8] Rastgoo, R., Kiani, K., & Escalera, S. (2020). Hand sign language recognition using multi-view hand skeleton. *Expert Systems with Applications*, 150, 113336.
- [9] A Comprehensive Analysis Of Sign Language Recognition System 2025 PDF: <https://rjpn.org/jetnr/papers/JETNR2505045.pdf> rjpn.or
- [10] A. Navendu, Word-Level Indian Sign Language Recognition using MediaPipe and LSTM, 2024. PDF Available: https://www.techrxiv.org/users/798477/articles/1151368/master/file/data/Navendu_iSeS_2024-2/Navendu_iSeS_2024-2.pdf
- [11] R. Z. Khan and N. A. Ibraheem, "Hand Gesture Recognition: A Literature Review," *International Journal of Artificial Intelligence & Applications (IJAlA)*, vol. 3, no. 4, pp. 161–174, 2012.
- [12] G. Johansson, "Visual Perception of Biological Motion and a Model for Its Analysis," *Perception & Psychophysics*, vol. 14, no. 2, pp. 201–211, 1973.
- [13] A. F. Bobick and A. D. Wilson, "A State-Based Approach to the Representation and Recognition of Gesture," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 12, pp. 1325–1337, 1997.
- [14] C. Vogler and D. Metaxas, "A Framework for Recognizing the Simultaneous Aspects of American

- Sign Language,” Computer Vision and Image Understanding, 2001.
- [15] S. S. Fels and G. E. Hinton, “Glove-Talk II: A Neural Network Interface which Maps Gestures to Speech,” IEEE Transactions on Neural Networks, vol. 9, no. 1, pp. 205–212, 1998.
- [16] T. Starner, J. Weaver, and A. Pentland, “Real-Time American Sign Language Recognition Using Desk and Wearable Computer-Based Video,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 20, no. 12, pp. 1371–1375, 1998.
- [17] A. D. Wilson and A. F. Bobick, “Parametric Hidden Markov Models for Gesture Recognition,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 21, no. 9, pp. 884–900, 1999.
- [18] R. H. Liang and M. Ouhyoung, “A Sign Language Recognition System Using Hidden Markov Model and Context Sensitive Search,” ACM Symposium on Virtual Reality Software and Technology, pp. 59–66, 1996.
- [19] C. Vogler and D. Metaxas, “Parallel Hidden Markov Models for American Sign Language Recognition,” IEEE International Conference on Computer Vision, pp. 116–122, 1999.
- [20] F. Lefebvre-Albaret, “Automatic Processing of French Sign Language: Modeling and Phonological Constraints,” PhD Thesis, University of Toulouse, 2010.
- [21] M. Isard and A. Blake, “CONDENSATION—Conditional Density Propagation for Visual Tracking,” International Journal of Computer Vision, vol. 29, no. 1, pp. 5–28, 1998.
- [22] J. Schlenzig, E. Hunter, and R. Jain, “Vision-Based Hand Gesture Interpretation Using Recursive Estimation,” Proceedings of Asilomar Conference, 1994.
- [23] S. Haynes and R. Jain, “Detection of Moving Edges,” Computer Vision, Graphics, and Image Processing, vol. 21, no. 3, pp. 345–367, 1980.
- [24] A. Leonardis, A. Gupta, and R. Bajcsy, “Segmentation as the Search for the Best Description of the Image,” IEEE International Conference on Computer Vision, 1990.
- [25] D. Marshall, G. Lukacs, and R. Martin, “Robust Segmentation of Primitives from Range Data,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 23, no. 3, pp. 304–314, 2001.
- [26] J. M. Siskind and Q. Morris, “A Maximum-Likelihood Approach to Visual Event Classification,” European Conference on Computer Vision, pp. 347–360, 1996.
- [27] E. S. Klima and U. Bellugi, “The Signs of Language,” Harvard University Press, 1979.
- [28] S. K. Liddell and R. E. Johnson, “American Sign Language: The Phonological Base,” Sign Language Studies, vol. 64, pp. 195–277, 1989.
- [29] C. Vogler and D. Metaxas, “Toward Scalability in ASL Recognition: Breaking Down Signs into Phonemes,” Gesture Workshop, Springer, 1999.
- [30] M. Jebali, P. Dalle, and M. Jemni, “Extension of Hidden Markov Model for Recognizing Large Vocabulary of Sign Language,” Research Lab LaTICE, University of Tunis.