



(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 8.699

Volume 14, Issue 5, May 2025

⊕ www.ijirset.com 🖂 ijirset@gmail.com 🖄 +91-9940572462 🕓 +91 63819 07438





www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 5, May 2025 |DOI: 10.15680/IJIRSET.2025.1405494|

Deep Learning-Based Real-Time Sign Language Recognition and Text Conversion

Prof. Swarupa Kamble, Tapaswi Jadhav, Aditya Bhosale, Tanishka Ghate, Aaisha Baig

Assistant Professor, Dept. of Computer Engineering, Dr. D.Y. Patil School of Engineering & Technology, Pune, India

UG Student, Dept. of Computer Engineering, Dr. D.Y. Patil School of Engineering & Technology, Pune, India

UG Student, Dept. of Computer Engineering, Dr. D.Y. Patil School of Engineering & Technology, Pune, India

UG Student, Dept. of Computer Engineering, Dr. D.Y. Patil School of Engineering & Technology, Pune, India

UG Student, Dept. of Computer Engineering, Dr. D.Y. Patil School of Engineering & Technology, Pune, India

ABSTRACT: This research introduces a novel approach to real-time Indian Sign Language (ISL) recognition that converts gestural communication into textual format. Our methodology combines advanced computer vision algorithms with deep neural networks to deliver superior recognition precision. The framework utilizes MediaPipe for gesture detection and landmark identification, coupled with a MobileNetV2-derived architecture for classification, developed through Google Colab infrastructure. The solution comprises a Flask-powered application featuring three core components: static image interpretation, textual-to-gestural visualization, and live webcam-based gesture analysis. Evaluation results validate the system's capability in eliminating communication barriers for individuals with auditory and speech disabilities. This work details the technical execution, architectural design, and comprehensive performance assessment of our developed solution.

KEYWORDS: Indian Sign Language Recognition, MobileNetV2, Media Pipe, Hand Landmark Detection, Flask Web Application, Deep Learning, Computer Vision, Real-Time Recognition.

I. INTRODUCTION

Effective communication represents a cornerstone of human interaction, yet remains significantly challenging for India's estimated 63 million individuals with hearing disabilities. Sign language constitutes their predominant communication medium, with Indian Sign Language serving as the primary gestural system across the Indian subcontinent. Nevertheless, the broader community's insufficient familiarity with sign language establishes substantial communicative obstacles.

This communication gap necessitates technological interventions that can translate sign language into forms comprehensible to the general population. While several efforts have been made in this direction, most existing solutions suffer from limitations such as restricted vocabulary, inability to function in real-time, or dependence on specialized hardware.

This paper presents a comprehensive implementation of a sign language recognition system that addresses these limitations through:

- 1. Advanced Neural Architecture: A deep learning framework constructed on MobileNetV2 foundations for optimal recognition efficiency.
- 2. MediaPipe Integration: Incorporation of MediaPipe technology for precise gesture landmark identification
- 3. Comprehensive Web Platform: A multi-component Flask-based application providing three core services:
 - Static image-to-text interpretation for uploaded gesture photographs
 - Textual-to-image transformation for sign demonstration
 - Live webcam-based gesture recognition and translation

IJIRSET©2025





www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 5, May 2025

|DOI: 10.15680/IJIRSET.2025.1405494|

Our system leverages cutting-edge computer vision methodologies and neural networks to accomplish superior accuracy while preserving real-time functionality on conventional computing platforms. This implementation seeks to enhance sign language recognition accessibility and practicality for daily applications.

II. LITERATURE SURVEY

- Hand and Pose-Based Feature Selection for Zero-Shot Sign Language Recognition
 Authors: Giray Sercan Özcan, Yunus Can Bilge, Emre Sümer
 This study proposes a new approach to Zero-Shot Sign Language Recognition (ZSSLR) using hand and landmark
 data streams. The method integrates Multiscale Vision Transformers, ResNeXt, and Spatial-Temporal Graph
 Convolutional Networks to enhance feature extraction and visual embedding. Sign descriptions are used alongside
 visual data for Zero-Shot Learning. The approach addresses the lack of training data for sign languages and is
 validated on two established benchmarks.
- 2. Sign Language Recognition Using Graph and General Deep Neural Network Based on Large Scale Dataset Authors: Abu Saleh Musa Miah, Md. Al Mehedi Hasan, Satoshi Nishimura, Jungpil Shin

This paper introduces a two-stream multistage graph convolutional network (GCAR) to improve Sign Language Recognition (SLR) by focusing on hand skeleton, body motion, and facial expressions. By processing spatial-temporal data in two separate streams, the model generates better feature representations, improving classification accuracy. The method achieves high performance on large datasets, setting a new benchmark for SLR systems.

3. Hybrid InceptionNet Based Enhanced Architecture for Isolated Sign Language Recognition

Authors: Deep R. Kothadiya, Chintan M. Bhatt, Hena Kharwa, Felix Albu

This research introduces a hybrid architecture for isolated sign language recognition combining InceptionV4 and convolutional networks. The model improves feature extraction using optimized stem and inception layers, and processes both spatial and temporal information. Tested on multiple datasets, including Indian Sign Language, it achieves high accuracy (98.46%), demonstrating its potential for automated sign language interpretation.

III. PROPOSED METHODOLOGY

Our project will employ state-of-the-art computer vision techniques and machine learning algorithms to achieve accurate sign language recognition. The methodology will follow these key steps:

- 1. **Data Acquisition**: Compile a comprehensive dataset of Indian Sign Language gestures, ensuring diversity in hand shapes, movements, and signers.
- 2. **Image Preprocessing**: Enhance and standardize the collected images to optimize them for model training, including techniques such as normalization, noise reduction, and background removal.
- 3. **Model Architecture**: Utilize advanced deep learning techniques, with a focus on MobileNetV2, Media Pipe, to construct a robust gesture recognition model.
- 4. **Real-time Video Processing**: Develop a system capable of capturing and processing live video input, extracting relevant frames for gesture analysis.
- 5. Gesture Classification: Apply the trained model to classify gestures from processed video frames, implementing techniques to handle continuous signing.
- 6. **Text Generation**: Convert classified gestures into corresponding textual output, considering context and grammar rules of Indian Sign Language.
- 7. User Interface Development: Design and implement an intuitive, accessible interface for users to interact with the system effectively.

IV. SYSTEM ARCHTECHTURE

The implemented system follows a modular architecture comprising three main components: the frontend interface, backend processing, and the deep learning model.





www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 5, May 2025

|DOI: 10.15680/IJIRSET.2025.1405494|

A. Frontend Interface: The frontend interface is developed using HTML, CSS, and JavaScript with Flask templating. It provides three distinct functionalities:

- 1. Image Upload Interface: Allows users to upload images containing sign gestures for recognition.
- 2. Text Input Interface: Enables users to enter text that gets converted to corresponding sign language images.
- 3. Webcam Interface: Provides real-time sign language recognition through the user's webcam.

The interface is designed to be intuitive and responsive, with minimal latency between user actions and system responses.

■ SignEase Breaking Communication Barriers Through Technology				
Bridgi Gaps SignEase u gestures in accessible Try It No	In g Communication ses advanced AI to recognize sign language real-time, making communication more or everyone.			
Information of Sign Language				
Alphabets and Words in Sign Language Dictionary of Sign Language	Sign Language to Text Analyze sign language gestures from images			
Our Project Contact Us	T Text to Sign Language Convert text into sign language gestures.			
	Live Detection Real-lime sign Language detection using your camera			
0	Sign Language Dictionary Learn common sign language gestures and their meanings Open Dictionary			

B. Backend Processing: The backend is implemented using Flask, a lightweight WSGI web application framework. It handles:

- 1. Image Processing: Pre-processing uploaded images or webcam frames using OpenCV
- 2. Hand Detection and Landmark Extraction: Using MediaPipe to detect hands and extract key landmarks
- 3. Model Inference: Passing processed data to the trained model for prediction
- 4. Response Formatting: Converting model predictions into user-friendly output

The backend routes are structured to handle different types of requests efficiently, with separate endpoints for each functionality module.

IJIRSET©2025

14695





www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 5, May 2025

|DOI: 10.15680/IJIRSET.2025.1405494|

C. Neural Network Component: Our system's foundation comprises a neural network based on MobileNetV2 architecture, chosen for its optimal balance between precision and computational requirements. The network underwent training on an Indian Sign Language gesture dataset utilizing transfer learning methodologies.

The network processes hand landmark coordinates as input, generating probability distributions across supported sign categories. Our trained model is preserved in .h5 format and integrated into the Flask application for inference operations.

V. EXPERIMENTAL RESULTS

A. Dataset Development: Our model training utilized a customized Indian Sign Language gesture dataset including:

- Images of alphabets (A-Z) in ISL
- Common words and phrases in ISL
- Images captured in various lighting conditions
- Multiple subjects performing the signs

Data enhancement techniques including rotation, scaling, and brightness modifications were implemented to strengthen model resilience. Dataset division comprised training (70%), validation (15%), and testing (15%) segments.

B. Gesture Detection and Landmark Extraction: MediaPipe Hands, a precision hand and finger tracking solution, was incorporated for gesture detection and landmark analysis. MediaPipe delivers 21 three-dimensional landmarks per hand.

The implementation follows these steps for landmark extraction:

- 1. Detect hand in the image using MediaPipe detector
- 2. Extract the 21 key landmarks with their x, y, z coordinates
- 3. Normalize the coordinates to make them invariant to image size and hand position
- 4. Create a feature vector from these normalized coordinates

This approach allows the system to work with any hand size and position within the frame, making it robust to variations in user positioning and camera distance.

C. Model Architecture and Training: The core recognition model employs MobileNetV2 architecture, chosen for its efficiency and suitability for deployment on systems with limited computational resources. The model architecture includes:

- 1. Base MobileNetV2 (pretrained on ImageNet): Used for feature extraction
- 2. Global Average Pooling: To reduce spatial dimensions
- 3. Dense Layers: For classification with dropout regularization
- 4. Softmax Output Layer: Producing probability distribution across sign classes.





www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|



Volume 14, Issue 5, May 2025 |DOI: 10.15680/IJIRSET.2025.1405494|

The model was implemented using TensorFlow and Keras, and trained on Google Colab to leverage GPU acceleration. Training configuration included:

- Optimizer: Adam (learning rate:0.0001)
- Loss function: Categorical Cross-Entropy
- Batch size: 32
- Epochs: 100 with early stopping mechanism

After training, the model was converted to .h5 format for integration into the Flask application.

D. Flask Application Development: The web application was developed using Flask framework, with three main routes corresponding to the system's functionalities:

- 1. /sign-to-text: Handles image uploads for sign-to-text conversion
- 2. /text-to-sign: Processes text input and returns corresponding sign images
- 3. /webcam: Manages real-time webcam feed processing and sign recognition

The application uses asynchronous processing where appropriate to maintain responsiveness. For real-time recognition, frames are processed at an optimized rate to balance between performance and accuracy.

E. Sign-to-Text Conversion Module: The sign-to-text conversion module processes static images containing sign language gestures and translates them into text. As shown in Image 2, the implementation includes:

- 1. Image Input Handling: The system processes uploaded image files containing hand gestures
- 2. Hand Detection and Landmark Extraction: Using MediaPipe to identify hand positions and extract the 21 key landmarks
- 3. Preprocessing: Normalizing landmarks to create uniform input for the model regardless of image conditions
- 4. **Prediction and Confidence Scoring**: Passing the processed landmarks to the trained model, which outputs both the predicted sign and a confidence score (as shown by the "Predicted: L, Confidence: 100.00%" display)
- 5. **Performance Metrics**: Displaying processing speed (280ms/step in the example) to monitor system performance.





www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 5, May 2025

|DOI: 10.15680/IJIRSET.2025.1405494|

Indian Sign Language Recognition Bridging communication gaps through technology					
Upload Sign Language	Gesture	Recognition Result			
Select Recognition Mode:		Upload an image to see the recognition result.			
Alphabets (A-Z)	Numbers (1-9)	0			
Upload Image		Processing your image			
Capture using Webcam		Α			
Preview of uploaded sign language gesture		Letter A			

F. Text-to-Sign Conversion Component: The text-to-sign conversion module operates by correlating textual input with corresponding sign language images maintained in a database. Implementation involves:

- 1. Tokenizing the input text into words and characters
- 2. Querying the database for matching sign images
- 3. Returning the sequence of images corresponding to the input text

For words not found in the database, the system falls back to character-by-character spelling using alphabet signs.

Sign Transcriber					
	Enter Text : [A	<u>]</u>			
	Convert				
	A 1 (
	Back to Home				

The module also displays the source code in the application window, providing transparency about the implementation and facilitating debugging during development. This can be seen in the right side of Image 2, where the path handling and image processing code is visible.





www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 5, May 2025

|DOI: 10.15680/IJIRSET.2025.1405494|

G. Real-Time Webcam Recognition Module: The webcam-based real-time recognition module, demonstrated in Image 1, is the most interactive component of the system. This module:

- 1. Captures Live Video: Accesses the user's webcam to receive a continuous stream of frames
- 2. Processes Frames in Real-Time: Each frame is analyzed for hand gestures
- 3. Visualizes Hand Landmarks: Displays detected hand landmarks as green overlay lines on the detected hand
- 4. Character Recognition: Identifies individual signs and displays the recognized character.
- 5. Sentence Building: Accumulates recognized characters into a sentence (building "HELLO")
- 6. Word Suggestions: Provides intelligent word completion suggestions based on the characters recognized so far.
- 7. **Interactive Controls**: Offers user interaction buttons like "Clear" to reset the sentence and "Speak" to utilize text-to-speech functionality.



VI. MODEL EVALUATION

Our implemented model underwent evaluation based on recognition precision, processing velocity, and user experience assessment.

A. Recognition Accuracy: The model achieved the following precision metrics on the testing dataset:

- Overall accuracy: 94.5%
- Alphabet recognition accuracy: 96.8%
- Common word recognition accuracy: 92.3%

Confusion matrix analysis revealed that most errors occurred between visually similar signs, such as 'M' and 'N' in the ISL alphabet. Fig. 5 shows the confusion matrix for alphabet recognition.

B. Processing Speed: Processing speed is critical for real-time applications. The implementation achieved:

- Average processing time per frame: 42ms
- Frame rate for real-time recognition: ~23-24 FPS
- Recognition latency: <100ms

These metrics demonstrate real-time system operation with minimal perceptible delay, establishing suitability for practical applications.

C. User Interface Performance: The interface was evaluated based on responsiveness and user experience, with an emphasis on:

- Load time for different modules
- Response time for image uploads

IJIRSET©2025

| An ISO 9001:2008 Certified Journal |

14699





www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 5, May 2025

|DOI: 10.15680/IJIRSET.2025.1405494|

• Fluidity of real-time recognition

All interface components performed within acceptable parameters, with most operations completing in less than 500ms.

VII. CONCLUSION

This research presented comprehensive implementation details of a deep learning-based Indian Sign Language recognition system. MediaPipe integration for gesture landmark detection with MobileNetV2-based classification provides an effective real-time sign language recognition solution. Our Flask-based web application offers multiple functionalities, making technology accessible through standard web browsers.

The system achieves high accuracy while maintaining real-time performance, demonstrating its potential for practical applications. While limitations exist, the modular architecture allows for incremental improvements and extensions.

This implementation represents a significant step toward bridging the communication gap faced by hearing and speechimpaired individuals, with potential applications in educational, public service, and personal communication contexts.

REFERENCES

[1] World Health Organization, "Deafness and hearing loss," WHO, 2021.

[2] M. K. Bhuyan, D. Ghosh, and P. K. Bora, "A framework for hand gesture recognition with applications to sign language," in IEEE India Conference, 2006.

[3] 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.

[4] M. Zahedi, D. Keysers, T. Deselaers, and H. Ney, "Combination of tangent distance and an image distortion model for appearance-based sign language recognition," in DAGM Symposium, 2005.

[5] O. Koller, H. Ney, and R. Bowden, "Deep hand: How to train a CNN on 1 million hand images when your data is continuous and weakly labelled," in IEEE Conference on Computer Vision and Pattern Recognition, 2016.

[6] J. Huang, W. Zhou, H. Li, and W. Li, "Sign language recognition using 3D convolutional neural networks," in IEEE International Conference on Multimedia and Expo, 2015.

[7] N. C. Camgoz, S. Hadfield, O. Koller, H. Ney, and R. Bowden, "Neural sign language translation," in IEEE Conference on Computer Vision and Pattern Recognition, 2018.

[8] H. Zhou, W. Zhou, Y. Zhou, and H. Li, "Spatial-temporal multi-cue network for continuous sign language recognition," in AAAI Conference on Artificial Intelligence, 2020.









INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY





www.ijirset.com