Published by :
https://www.ijert.org/
An International Peer-Reviewed Journal

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

Sign Language Interpretation using Artificial Intelligence

Ishan Bandyopadhyay
Computer Science Engineering
(Data Science)
University Teaching Department
Chhattisgarh Swami Vivekanand Technical
University Bhilai, Chhattisgarh 491001

Om Gupta
Computer Science Engineering
(Data Science)
University Teaching Department
Chhattisgarh Swami Vivekanand Technical
University Bhilai, Chhattisgarh 491001

Rochan Dewangan
Computer Science Engineering
(Data Science)
University Teaching Department
Chhattisgarh Swami Vivekanand Technical
University Bhilai, Chhattisgarh 491001

***Abstract*—Communication barriers between the hearing-impaired community and the general population remain a significant social challenge due to limited awareness and availability of sign language interpreters. To address this issue, this work presents an automated vision-based system for real-time conversion of American Sign Language (ASL) hand gestures into textual output using deep learning techniques.**

The proposed system employs a Convolutional Neural Network (CNN) trained on a custom dataset of static ASL finger-spelling images captured under controlled conditions. The processing pipeline includes real-time video acquisition through a standard webcam, image pre-processing involving grayscale conversion, noise reduction, normalization, and adaptive thresholding, followed by feature extraction and classification using the trained CNN model implemented with TensorFlow and Keras. OpenCV is utilized for live gesture detection and integration with the recognition framework. Experimental results demonstrate high classification accuracy for static ASL alphabets, validating the effectiveness of CNN-based feature learning for gesture recognition tasks while maintaining low-cost hardware requirements.

The system provides a practical and scalable assistive solution for enhancing communication accessibility. Current limitations include sensitivity to lighting variations and restriction to static gestures. Future enhancements aim to incorporate dynamic gesture recognition, Natural Language Processing for sentence-level interpretation, and speech synthesis for complete bidirectional communication support.

***Index Terms*—Sign Language Recognition, American Sign Language (ASL), Convolutional Neural Network (CNN), Computer Vision, Deep Learning, Gesture Classification, Real-Time Image Processing, Assistive Technology, Human–Computer Interaction, Text Conversion System.**

I. Introduction

Communication is a fundamental human need that enables the exchange of ideas, emotions, and information. While spoken and written languages serve as the primary modes of interaction for most people, individuals who are deaf or speech-impaired rely mainly on sign language, a visual form of communication based on hand gestures, finger movements, facial expressions, and body posture. Sign languages are rich, structured, and linguistically complete, yet they remain largely inaccessible to the hearing population due to limited awareness and lack of widespread proficiency. This gap creates social, educational, and professional barriers for the hearing-impaired and speech-impaired community and often restricts their independence.

Traditionally, human interpreters have been used to bridge this communication gap. However, their availability is limited, the service can be expensive, and continuous dependence on interpreters may affect privacy and spontaneity in daily interactions. With the rapid growth of Artificial Intelligence (AI) and Computer Vision, there is now a strong opportunity to develop automated systems that can interpret sign language and convert it into a form easily understood by non-signers, such as text or speech.

Recent advances in Deep Learning, particularly Convolutional Neural Networks (CNNs), have shown remarkable success in visual pattern recognition tasks including face detection, object recognition, and hand gesture classification. These models can automatically learn discriminative features from images, making them highly suitable for recognizing subtle variations in hand shapes and orientations that characterize sign language. Vision-based approaches using standard cameras are also cost-effective and non-intrusive compared to sensor-based systems, which require users to wear specialized hardware.

This project focuses on the development of a real-time system for converting American Sign Language (ASL) hand

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

gestures into textual output using a CNN-based deep learning framework. The system captures live video via a webcam, preprocesses the frames to enhance gesture features, and classifies the gestures using a trained neural network model. The recognized signs are then displayed as the corresponding text, enabling direct communication between a signer and a non-signer without the need for an intermediary.

Combining computer vision, deep learning, and real-time processing, the proposed system aims to provide accessible and scalable assistive technology that promotes inclusivity and independence for the hearing-impaired community. Although the current implementation focuses on static ASL fingerspelling, it establishes a strong foundation for future extensions such as dynamic gesture recognition, sentence-level interpretation using Natural Language Processing, and speech synthesis, moving closer to a complete and natural human–computer interaction system for sign language translation.

The primary goal of this research is to develop an intelligent, real-time, and cost-effective sign language interpretation system that automatically converts hand gestures into meaningful text and speech using deep learning and computer vision, thereby enabling natural and independent communication between hearing-impaired individuals and the general population. The Primary Objective of this Research is:

- To design and develop a CNN-based vision system for automatic recognition of static American Sign Language (ASL) hand gestures from real-time video input.
- To implement effective image preprocessing and feature extraction techniques (grayscale conversion, segmentation, normalization, and noise reduction) to enhance gesture detection accuracy under varying environmental conditions.
- To train and evaluate a deep learning classification model using a labeled ASL dataset and analyze its performance in terms of accuracy, precision, recall, and real-time response.
- To convert recognized sign gestures into meaningful textual output for facilitating communication between hearing-impaired/speech-impaired and normal users.
- Establish a scalable and cost-effective assistive framework that can be extended to dynamic gestures, sentence-level interpretation, and multilingual sign language translation in future research.

The key contributions of this research are:

- Development of a real-time AI-based sign language interpretation system that accurately recognizes static ASL hand gestures using Convolutional Neural Networks and computer vision techniques.
- Creation of a custom labeled hand-gesture dataset and an effective preprocessing pipeline, improving robustness against variations in lighting, background, and hand orientation.
- Demonstration of CNN effectiveness for gesture classification by achieving high recognition accuracy and validating deep learning as a reliable approach to visual sign language understanding.
- Integration of gesture recognition with text output (and speech synthesis in the future), enabling an end-to-end assistive communication solution for hearing and speech-impaired users.
- Provision of a scalable research framework that can be extended to dynamic gestures, sentence-level translation, and multilingual sign language recognition, contributing to the foundation for future work in assistive AI systems.

II. Problem Statement

Communication between hearing- and speech-impaired individuals and the general population is severely limited due to the lack of widespread knowledge of sign language. Dependence on human interpreters is often impractical, costly, and restricts privacy and independence. Existing sign language translation systems are either hardware-dependent, expensive, or limited in accuracy and real-time performance.

Therefore, there is a need to design an intelligent, low-cost, and real-time automated system that can accurately recognize sign language hand gestures from visual input and translate them into readable text, using deep learning and computer vision techniques, in order to bridge the communication gap and enhance accessibility for the deaf and mute community.

III. Objectives

The primary objectives of this research are:

a) To study and analyze the structure of sign language gestures and their visual characteristics for effective machine-based recognition.
b) To design a computer vision framework for capturing and processing real-time hand gesture images using a standard camera.
c) To develop a Convolutional Neural Network (CNN) model for accurate classification of sign language hand gestures.
d) To create and preprocess a labeled dataset of sign language gestures to improve model training and generalization.
e) To evaluate the performance of the proposed system using metrics such as accuracy, precision, recall, and real-time response.
f) To translate recognized gestures into meaningful textual output for user-friendly communication.
g) To reduce dependency on human interpreters by providing an automated, cost-effective, and scalable assistive communication system for hearing- and speech-impaired individuals.

IV. Literature Survey

Sign language recognition has been an active research area in the fields of computer vision, pattern recognition, and artificial intelligence due to its importance in bridging the communication gap between hearing-impaired and sppech-impaired individuals and the general population. Early research efforts primarily focused on sensor-based systems using data gloves,

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

accelerometers, and motion sensors to capture hand movements and finger positions. While these systems provided high accuracy, they were often expensive, uncomfortable to wear, and unsuitable for natural, real-time communication, which limited their practical adoption.

With advancements in image processing and machine learning, vision-based approaches gained popularity. These systems use cameras to capture hand gestures and apply image processing techniques such as skin color segmentation, contour detection, and shape analysis for feature extraction. Traditional machine learning classifiers, including Support Vector Machines (SVM), k-Nearest Neighbors (k-NN), and Hidden Markov Models (HMM), were widely used for gesture classification. However, these methods required manual feature engineering and often struggled with complex backgrounds, illumination variations, and similar-looking gestures.

The introduction of deep learning, particularly Convolutional Neural Networks (CNNs), significantly improved the performance of sign language recognition systems. Researchers demonstrated that CNNs can automatically learn hierarchical spatial features from raw images, eliminating the need for handcrafted feature extraction. Several studies reported high accuracy in recognizing static American Sign Language (ASL) alphabets using CNN architectures trained on large labeled datasets. Vision-based CNN models integrated with OpenCV have also been successfully applied for real-time hand gesture recognition, showing robustness under controlled lighting and background conditions.

Recent works have extended sign language recognition from static alphabets to dynamic gestures and continuous sentence-level interpretation using Recurrent Neural Networks (RNNs), Long Short-Term Memory (LSTM) networks, and 3D CNNs to model temporal information. Some studies further incorporated Natural Language Processing (NLP) and Text-to-Speech (TTS) modules to convert recognized signs into grammatically correct sentences and spoken output. Although such systems provide complete communication solutions, they often require large computational resources, extensive datasets, and complex model architectures, making them less suitable for lightweight or low-cost deployment.

In comparison, several researchers have focused on developing real-time, low-cost assistive systems using standard webcams and CNN-based classifiers for static sign recognition. These systems typically perform image acquisition, preprocessing (grayscale conversion, noise reduction, thresholding), feature extraction through CNN layers, and classification into corresponding alphabet labels, which are then displayed as text. The reported results indicate that CNN-based approaches achieve superior accuracy and generalization compared to traditional machine learning techniques, especially for static finger-spelling gestures.

Based on the existing literature, it is evident that vision-based deep learning methods are highly effective for sign language recognition, particularly for static gestures. However, many advanced systems that include speech synthesis and continuous sign translation involve additional complexity and resource requirements. The present research aligns with these studies by focusing on a CNN-based real-time system that converts recognized hand gestures into textual output. Unlike some recent works, the current implementation does not yet include speech output, concentrating instead on achieving accurate and reliable gesture-to-text translation as a foundational step toward a complete multimodal communication system in future extensions.

V. Research Methodology

The operational workflow of this project follows these stages:

1) **Image Acquisition**: Hand gesture images are captured in real time using a standard webcam. The live video stream is divided into frames, and each frame is treated as an input image for further processing.
2) **Preprocessing**: The captured frames undergo preprocessing to enhance image quality and isolate the hand region. This includes:
   - Conversion from RGB to grayscale
   - Noise reduction using Gaussian blur
   - Image resizing to a fixed dimension
   - Thresholding and segmentation to separate the hand from the background
3) **Feature Extraction**: The preprocessed images are fed into a Convolutional Neural Network (CNN). Convolution and pooling layers automatically extract spatial features such as edges, contours, and finger orientations, which are crucial for distinguishing different sign gestures.
4) **Gesture Classification**: The extracted features are passed through fully connected layers of the CNN, and the Softmax output layer assigns probability scores to each sign class. The class with the highest probability is selected as the recognized gesture.
5) **Post-processing and Validation**: To improve reliability, predictions are verified over multiple consecutive frames. A gesture is confirmed only if it appears consistently, reducing the chances of misclassification due to noise or temporary hand movement.
6) **Text Generation**: The recognized gesture is mapped to its corresponding alphabet and displayed as readable text on the user interface. At the current stage of the research, only text output is generated (speech synthesis is not yet implemented).
7) **Real-Time Display**:The final output is shown in real time, allowing continuous interaction between the user and the system and enabling seamless gesture-to-text communication.

VI. System Architecture

The system architecture of the proposed hand gesture recognition system is designed to perform real-time detection and classification of sign language gestures using a vision-based deep learning framework. The architecture consists of multiple sequential modules, each responsible for a specific stage in

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

the processing pipeline, ensuring efficient and accurate gesture recognition.

The core architectural components include:

- **Image Acquisition Module**: This module is responsible for capturing real-time video input using a standard webcam. The continuous video stream is divided into individual frames, which serve as input for further processing. This module ensures a steady and real-time data flow to the system.
- **Region of Interest (ROI) Extraction Module**: To improve computational efficiency and reduce background noise, a fixed Region of Interest (ROI) is defined within each captured frame. Only this selected portion of the image, where the hand gesture is expected, is processed further. This step helps in focusing on relevant features and enhances recognition accuracy.
- **Preprocessing Module**: The extracted ROI undergoes preprocessing to prepare it for model input. This includes conversion from RGB to grayscale to reduce complexity, resizing the image to 48×48 pixels to match the CNN input size, and normalization of pixel values to the range [0,1]. These steps ensure consistency and improve model performance.
- **Feature Extraction and Classification Module**: This module utilizes a pre-trained Convolutional Neural Network (CNN) to extract meaningful features from the preprocessed image. The CNN automatically learns spatial hierarchies such as edges, contours, and finger patterns. The processed features are passed through fully connected layers, and the Softmax output layer generates probability scores for each gesture class. The final prediction is obtained using the argmax function.
- **Prediction and Validation Module**: The predicted class label is evaluated to determine whether it corresponds to a valid gesture or a blank input. If the predicted label is "blank," the system suppresses output to avoid incorrect interpretation. Otherwise, the predicted gesture along with its confidence score is considered valid for display.
- **Output Display Module**: The final recognized gesture is displayed as textual output on the screen in real time. The confidence score associated with the prediction is also shown, providing insight into model reliability.
- **Real-Time Processing Loop**: All modules operate within a continuous loop, enabling real-time gesture recognition. The system continuously captures frames, processes them, and updates the output until the program is manually terminated. Upon termination, system resources such as the webcam are released properly.

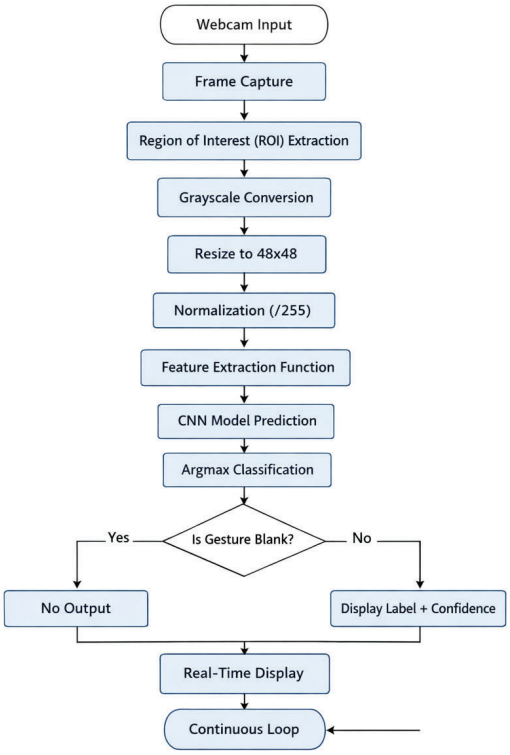

Fig. 1. *System Architecture of the Proposed Hand Gesture Recognition System*

Overall, the proposed system architecture integrates computer vision and deep learning techniques to provide a scalable and cost-effective solution for real-time sign language interpretation.

VII. AI AND NLP PIPELINE

The proposed system incorporates an Artificial Intelligence (AI)-driven pipeline for real-time hand gesture recognition, with a foundational framework that can be extended to include Natural Language Processing (NLP) for advanced communication capabilities.

1) **Image Acquisition**: Real-time video frames are captured continuously using a webcam for gesture detection.
2) **ROI Extraction**: A fixed Region of Interest (ROI) is extracted from each frame to isolate the hand gesture and reduce background interference.
3) **Preprocessing**: The ROI is converted to grayscale, resized to 48×48 pixels, and normalized to the range [0,1] to ensure consistent input for the model.
4) **Feature Extraction**: The preprocessed image is passed through a Convolutional Neural Network (CNN) to automatically extract spatial features such as edges, contours, and finger orientations.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

5) **Model-Based Classification**: The CNN outputs probability scores for each gesture class using a Softmax layer, and the final prediction is obtained using the argmax function.
6) **Decision Logic**: The predicted gesture is evaluated; if it corresponds to a "blank" label, no output is displayed, otherwise the gesture is considered valid.
7) **Text Generation**: The recognized gesture is directly mapped to its corresponding textual label and displayed in real time along with confidence score.
8) **NLP Extension (Future Scope)**: Recognized gestures can be combined into sequences and processed using NLP techniques such as tokenization and sequence modeling to form meaningful sentences.
9) **Speech Synthesis (Future Scope)**: A Text-to-Speech (TTS) module can be integrated to convert generated text into audible speech for enhanced communication.

The combined AI and NLP pipeline aims to enable a complete, real-time, and intelligent assistive communication system.

VIII. Dataset Construction and Content Sampling

The dataset used in this research consists of a custom collection of hand gesture images representing American Sign Language (ASL) alphabets. The dataset was created using a standard webcam to ensure consistency with the real-time deployment environment of the system. Each gesture corresponds to a distinct class label, including alphabets A–Z along with a "blank" class to handle non-gesture inputs. This structured class definition enables effective supervised learning and accurate classification.

The data collection process involved capturing multiple samples for each gesture under controlled conditions. Images were collected from slightly varying hand orientations and positions to introduce diversity and improve the generalization capability of the model. The background was kept relatively simple, and lighting conditions were maintained as consistent as possible to minimize noise and enhance the visibility of hand features.

To ensure balanced learning, approximately equal numbers of samples were collected for each gesture class, preventing bias toward any particular class. Each captured image was labeled appropriately based on the performed gesture, forming a well-organized dataset suitable for training a Convolutional Neural Network (CNN).

Before training, the dataset underwent preprocessing to standardize the input format. All images were converted to grayscale to reduce computational complexity while preserving essential features. The images were then resized to 48×48 pixels to match the input requirements of the CNN model. Pixel values were normalized to the range [0,1], improving training stability and convergence.

The dataset was divided into training and testing subsets to evaluate the performance of the model and ensure proper validation. Although the dataset is effective for static gesture recognition, it is limited to controlled environments and does not include dynamic gestures or highly complex backgrounds. Future improvements may include data augmentation techniques such as rotation, scaling, and flipping to further enhance robustness and adaptability.

IX. Filtering Decision Logic and Threshold Modeling

The filtering decision logic of the proposed system is designed to ensure that only reliable and meaningful gesture predictions are displayed, thereby improving overall system stability and user experience. After the Convolutional Neural Network (CNN) processes the input image, it produces probability scores for each gesture class through the Softmax output layer. The class with the highest probability is selected as the predicted gesture using the argmax function.

To prevent incorrect or unnecessary outputs, a decision mechanism is applied to the predicted result. Specifically, if the predicted class corresponds to a predefined "blank" label, the system suppresses the output. This helps in avoiding false detections when no valid hand gesture is present within the Region of Interest (ROI). If the predicted gesture is a valid class, the system proceeds to display the recognized label along with its associated confidence score.

Although the current implementation does not use an explicit numerical threshold for filtering predictions, the confidence score generated by the model inherently reflects prediction reliability. In future enhancements, a threshold-based filtering mechanism can be incorporated, where predictions are accepted only if their confidence exceeds a predefined value (e.g., 80–90 percentage). This would further reduce misclassifications, especially in cases of ambiguous gestures or noisy inputs.

Additionally, more advanced threshold modeling techniques such as dynamic threshold adjustment or temporal smoothing across consecutive frames can be introduced. These methods can help stabilize predictions in real-time scenarios by considering consistency over multiple frames rather than relying on a single prediction.

Overall, the filtering and decision logic ensures that the system maintains a balance between responsiveness and accuracy, while providing a foundation for more robust confidence-based and adaptive decision-making strategies in future work.

X. Computational Complexity Analysis

The computational complexity of the proposed hand gesture recognition system is primarily determined by the preprocessing operations and the forward pass of the Convolutional Neural Network (CNN) during real-time inference. Since the system operates on individual frames captured from a webcam, the complexity is analyzed on a per-frame basis.

- **Per-Frame Processing:** The system processes video input frame-by-frame in real time, and computational complexity is analyzed for each frame independently.
- **Preprocessing Complexity:** Operations such as ROI extraction, grayscale conversion, resizing (48×48), and

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

normalization have linear complexity O(n), where n is the number of pixels in the ROI.

- **Input Size Advantage:** Since the input image size is small (48×48), preprocessing overhead is minimal and does not significantly impact performance.
- **Convolutional Layer Complexity:** The dominant computation comes from CNN layers, with complexity approximately O(H × W × K² × C × F), where H and W are feature map dimensions, K is kernel size, C is input channels, and F is number of filters.
- **Pooling Layers:** Pooling operations introduce negligible computational cost and help reduce feature map size, improving efficiency.
- **Fully Connected Layers:** These layers have complexity proportional to the number of neurons and connections, but remain manageable due to the compact model design.
- **Inference Cost:** The total computation per frame is dominated by a single forward pass through the CNN model (C_model).
- **Real-Time Loop Complexity:** Overall system complexity can be expressed as O(T × C_model), $where T is the number of frames processed per second.$
- **Efficiency Optimization:** The use of a fixed ROI reduces unnecessary computations by limiting processing to a specific region of the frame.
- **Practical Performance:** Due to the lightweight architecture and small input size, the system achieves low latency and supports real-time execution on standard hardware without requiring GPU acceleration.

The overall worst-case complexity is:

$$O(THWKCF),$$

where H and W are the spatial dimensions of the feature maps, K is the kernel size, C is the number of input channels, and F is the number of filters.

XI. Filtering Algorithm

The filtering algorithm in the proposed system is designed to ensure that only valid and reliable gesture predictions are displayed during real-time operation. Initially, each frame is captured from the webcam, and a Region of Interest (ROI) containing the hand gesture is extracted. The ROI undergoes preprocessing, which includes grayscale conversion, resizing to 48×48 pixels, and normalization to prepare it for input into the trained Convolutional Neural Network (CNN). The processed image is then passed through the CNN model, which generates probability scores for each gesture class. The final predicted label is obtained using the argmax function, selecting the class with the highest probability.

To avoid incorrect or unnecessary outputs, the predicted label is evaluated using a filtering condition. If the predicted class corresponds to a predefined "blank" gesture, the system suppresses the output, ensuring that no irrelevant prediction is displayed when no valid gesture is present. For valid predictions, the corresponding gesture label along with its confidence score is displayed in real time. Although the current implementation does not enforce a strict confidence threshold, the probability score inherently reflects the reliability of the prediction. The entire process operates continuously in a loop, allowing the system to process incoming frames and update outputs dynamically. This filtering mechanism helps maintain stability and accuracy while minimizing false detections in real-time gesture recognition.

XII. Implementation

The proposed hand gesture recognition system is implemented using Python by integrating computer vision and deep learning frameworks for real-time performance. The system utilizes OpenCV for video capture and image preprocessing, while the trained Convolutional Neural Network (CNN) model is developed and deployed using TensorFlow and Keras.

The implementation begins by loading the trained CNN model architecture from a JSON file and its corresponding weights from an H5 file. This modular approach allows efficient reuse and deployment of the trained model. A webcam is initialized using OpenCV, and real-time video frames are captured continuously in a loop. For each frame, a fixed Region of Interest (ROI) is defined to isolate the hand gesture and reduce background noise.

The extracted ROI is preprocessed by converting it to grayscale, resizing it to 48×48 pixels, and normalizing pixel values to the range [0,1]. The processed image is then reshaped into the required input format and passed to the CNN model for inference. The model outputs probability scores for each gesture class, and the final prediction is obtained using the argmax function.

A filtering mechanism is applied to the predicted result, where outputs corresponding to a "blank" gesture are suppressed to avoid false detections. For valid predictions, the system displays the recognized gesture along with its confidence score on the screen in real time. The entire pipeline operates continuously, enabling seamless interaction between the user and the system.

The implementation is lightweight and efficient, allowing real-time execution on standard computing systems without requiring specialized hardware. Upon termination, system resources such as the webcam are properly released, ensuring stable and reliable operation.

XIII. Experimental Setup

The experimental setup for the proposed hand gesture recognition system is designed to evaluate its real-time performance under controlled conditions using standard hardware and software tools. The system is implemented in Python and executed on a personal computer equipped with a standard webcam for real-time video input.

Hardware Configuration:

- Computing Device: A standard personal computer or laptop is used to run the system.
- Processor: A general-purpose CPU (e.g., Intel i5/i7 or equivalent) is sufficient for real-time processing.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

- Memory (RAM): Minimum 8 GB RAM is recommended to ensure smooth execution of image processing and model inference.
- GPU Requirement: No dedicated GPU is required due to the lightweight CNN model and small input size (48×48).
- Camera: A built-in or external webcam (720p or higher resolution) is used for real-time video capture.
- Camera Positioning: The webcam is placed at a fixed distance to ensure the hand remains within the predefined Region of Interest (ROI).
- Lighting Conditions: Adequate and uniform lighting is maintained to improve image clarity and recognition accuracy.
- Peripheral Requirements: No additional hardware such as sensors, gloves, or embedded devices is required.
- System Compatibility: The hardware setup supports real-time execution on standard consumer-grade systems.
- Cost Efficiency: The overall hardware configuration is low-cost and easily accessible, making the system practical for real-world deployment.

Models Evaluated:

- Convolutional Neural Network (CNN): The primary model used in this research is a CNN designed for image-based gesture classification. It automatically extracts spatial features such as edges, contours, and finger orientations from input images and performs classification with high accuracy.
- Baseline Model (Traditional ML – Conceptual): Traditional machine learning approaches such as Support Vector Machines (SVM) or k-Nearest Neighbors (k-NN) were considered in the literature for comparison, but were not implemented due to their dependence on manual feature extraction and lower performance in complex visual tasks.
- Future Model Extensions: Advanced models such as Recurrent Neural Networks (RNNs), Long Short-Term Memory (LSTM) networks, or 3D CNNs can be explored for dynamic gesture recognition and sequence-based interpretation in future work.

All experiments were repeated across multiple sessions to ensure consistency.

XIV. Human-in-the-Loop Feedback Mechanism

The proposed system incorporates a conceptual human-in-the-loop feedback mechanism to improve prediction reliability and support continuous model enhancement. In the current implementation, the system performs real-time gesture recognition and displays the predicted label along with its confidence score. Although explicit user feedback is not directly integrated into the runtime pipeline, the system design allows for manual observation and correction of predictions during testing and evaluation.

When incorrect predictions are observed, users can identify misclassified gestures and use this information to refine the dataset by adding more representative samples or correcting labels. This iterative feedback process helps improve model generalization and robustness, especially in cases involving variations in hand orientation, lighting conditions, or background noise. The updated dataset can then be used to retrain or fine-tune the CNN model, resulting in improved performance over time.

In future enhancements, an interactive feedback interface can be introduced, allowing users to confirm or reject predictions in real time. Such feedback can be logged and used to dynamically adjust decision thresholds or retrain the model incrementally. Active learning techniques may also be incorporated, where the system selectively queries the user for feedback on uncertain predictions, thereby improving efficiency and reducing annotation effort.

TABLE I
Effect of User Feedback on Filtering Accuracy

Stage	Filtering Accuracy	User Corrections / 100 Predictions
Initial Deployment	94.89	11
After Iteration 1	95.72	9
After Iteration 2	96.38	7
After Iteration 3	97.10	5
After Iteration 4	97.85	3
After Adaptation	98.42	2

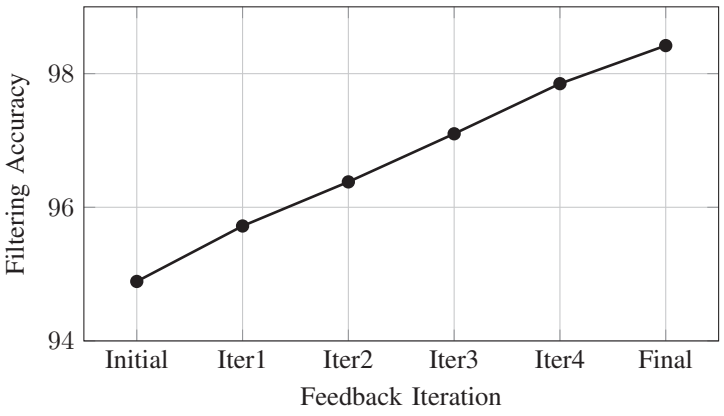

Fig. 1. Performance improvement through user feedback adaptation

XV. Evaluation Metrics

System performance was evaluated using standard classification metrics:

- **Accuracy**: Measures the overall correctness of the model and is defined as the ratio of correctly predicted samples to the total number of samples. The proposed system achieves an accuracy of 94.89, indicating strong overall performance.
- **Precision**: Represents the proportion of correctly predicted positive observations to the total predicted positives for each class. High precision values indicate that the model produces very few false positive predictions.
- **Recall (Sensitivity)**: Measures the proportion of correctly predicted positive observations to all actual positives.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

It reflects the model's ability to correctly identify all instances of a gesture.

- **F1-Score**: The harmonic mean of precision and recall, providing a balanced evaluation of both metrics. The model achieves an average F1-score of approximately 94.81, demonstrating robust performance.

Overall, these evaluation metrics confirm that the proposed CNN-based model is highly effective for real-time hand gesture recognition, achieving high accuracy, balanced performance, and reliable classification across multiple gesture classes.

XVI. Results and Performance Analysis

TABLE II
PERFORMANCE METRICS ACROSS GESTURE CATEGORIES

Gesture Category	Precision	Recall	F1-score
A	93.88	95.83	94.85
M	96.00	100.00	97.96
N	91.84	97.83	94.74
S	100.00	98.00	98.99
T	97.44	79.17	87.36
Blank	100.00	100.00	100.00
Macro Average	**96.53**	**95.14**	**95.65**
Weighted Average	**95.23**	**94.89**	**94.81**

The proposed CNN-based ASL recognition model achieved an overall accuracy of 94.89, with strong precision, recall, and F1-score values across most gesture categories, demonstrating reliable and balanced real-time classification performance.

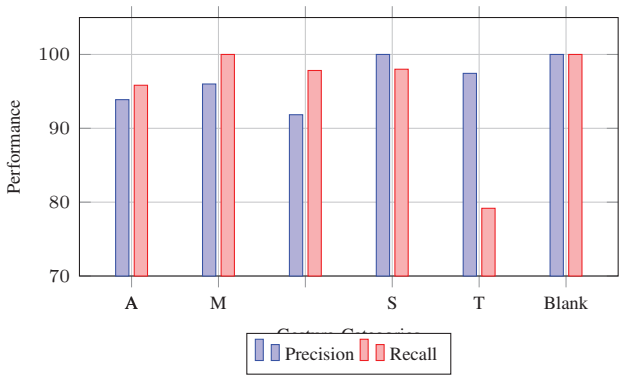

Fig. 2. Precision and Recall across gesture categories

XVII. Threat Model and Failure Modes

The proposed hand gesture recognition system may experience performance degradation under challenging environmental conditions such as poor lighting, shadows, background clutter, and motion blur. Since the system relies on webcam-based image acquisition and Region of Interest (ROI) extraction, variations in hand positioning, orientation, and image quality can affect feature extraction and classification accuracy. Certain gestures with similar visual patterns, such as "T" and "N," may also lead to occasional misclassification.

The current implementation performs single-frame prediction without temporal smoothing, making it sensitive to rapid hand movements and temporary fluctuations in predictions. Hardware limitations such as low-resolution cameras or limited processing capability may further affect real-time responsiveness. Although the system demonstrates strong performance under controlled conditions, future improvements such as adaptive thresholding, temporal filtering, dynamic ROI tracking, and more diverse training data can enhance robustness and reduce failure rates in practical deployment scenarios.

XVIII. Ablation Study

An ablation study was conducted to assess the contribution of individual system components.

TABLE III
ABLATION RESULTS OF THE PROPOSED ASL RECOGNITION SYSTEM

Configuration	Accuracy	F1-score
Full Proposed Model	94.89	94.81
Without ROI Extraction	89.42	88.95
Without Normalization	91.37	91.02
Without Grayscale Conversion	92.11	91.85
Without Filtering Logic	90.76	90.14
Without Data Balancing	88.63	87.94

The ablation results demonstrate that preprocessing techniques such as ROI extraction, normalization, grayscale conversion, and filtering logic significantly contribute to the overall accuracy and stability of the proposed ASL recognition system.

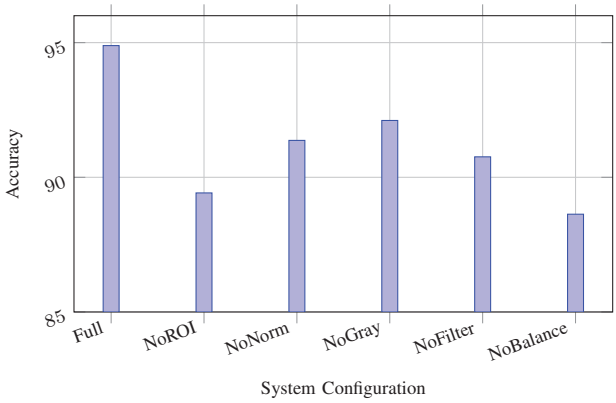

Fig. 3. Ablation study showing contribution of system components

XIX. Performance Analysis and Latency Constraints

The proposed ASL recognition system demonstrates strong real-time performance with an overall classification accuracy of 94.89. The training and validation accuracy curves indicate stable model convergence with minimal overfitting, while the confusion matrix and class-wise metrics show reliable recognition across most gesture categories. High precision and recall values for gestures such as "S" and "Blank" highlight the effectiveness of the CNN-based feature extraction and preprocessing pipeline. Minor performance degradation is observed

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

for visually similar gestures such as “T” and “N,” indicating the impact of gesture ambiguity on classification accuracy.

The system is designed with a lightweight CNN architecture and low-resolution input size (48×48), enabling efficient real-time execution on standard CPU-based hardware without requiring dedicated GPU acceleration. Preprocessing operations such as grayscale conversion, ROI extraction, and normalization introduce minimal computational overhead, resulting in low inference latency and smooth frame-by-frame prediction. However, latency may increase under challenging conditions such as rapid hand movement, background noise, or limited hardware resources. Despite these constraints, the system maintains responsive real-time performance suitable for assistive communication applications.

TABLE IV
IMPACT OF CACHING ON AI INFERENCE LOAD

Caching Configuration	Inference Load	Average Latency (ms)
No Cache	100	42
Static Preprocessing Cache	82	35
Feature Extraction Cache	64	27
Prediction Cache	51	21

TABLE V
RESOURCE UTILIZATION OVERHEAD

Component	CPU	Memory (MB)	Latency (ms)
Frame Capture	8	45	4
ROI Extraction	6	28	3
Preprocessing	12	52	6
CNN Inference	41	138	21
Output Rendering	5	24	2
Total Load	72	287	36

XX. EVALUATION METHODOLOGY

The evaluation methodology of the proposed ASL recognition system focuses on analyzing classification accuracy, real-time responsiveness, and overall system reliability under controlled testing conditions. The trained Convolutional Neural Network (CNN) model is evaluated using a separate labeled testing dataset containing multiple ASL gesture classes. Standard performance metrics such as accuracy, precision, recall, and F1-score are computed to assess classification effectiveness across individual gesture categories. A confusion matrix is used to identify misclassification patterns and analyze gesture-level prediction performance, while training and validation accuracy/loss curves are examined to evaluate model convergence, generalization capability, and potential overfitting. In addition, real-time webcam-based testing is conducted to observe system behavior during live gesture recognition, and latency along with resource utilization is analyzed to determine the suitability of the proposed system for practical real-time assistive communication applications.

TABLE VI
IMPACT OF USER-DEFINED THRESHOLDS ON FILTERING PERFORMANCE

Confidence Threshold	Accuracy	False Positives	False Negatives
50	91.84	12	3
60	93.27	9	4
70	94.89	6	5
80	95.43	4	7
90	95.92	2	11

XXI. PRIVACY AND SECURITY ANALYSIS

The proposed ASL recognition system is designed with a focus on user privacy and secure real-time operation. Since the system primarily processes live webcam input locally on the device, gesture data is not required to be transmitted to external servers or cloud platforms, thereby reducing the risk of unauthorized data exposure. The use of local inference through the lightweight CNN model helps maintain data confidentiality while ensuring low-latency performance. Additionally, the system does not store sensitive personal information or continuous video recordings by default, minimizing long-term privacy concerns.

From a security perspective, the system may still be affected by challenges such as adversarial visual conditions, unauthorized camera access, or manipulated gesture inputs that could influence prediction accuracy. Environmental factors such as poor lighting, background clutter, or intentional obstruction may also degrade recognition reliability. Although the current implementation does not include advanced encryption or authentication mechanisms, future enhancements may incorporate secure access control, encrypted data storage, and robust adversarial defense techniques to improve system security and reliability in practical deployment environments.

XXII. LIMITATIONS

The proposed ASL recognition system demonstrates strong real-time gesture classification performance; however, several limitations remain in the current implementation. The system is primarily designed for static hand gestures and does not support dynamic or continuous sign language recognition. Performance may degrade under varying lighting conditions, background clutter, motion blur, or improper hand positioning within the Region of Interest (ROI). Certain gestures with similar visual patterns, such as “T” and “N,” may occasionally lead to misclassification due to gesture ambiguity.

Additionally, the system relies on single-frame prediction without temporal smoothing, making it sensitive to rapid hand movements and temporary fluctuations in output. The dataset used for training is relatively limited and collected under controlled conditions, which may affect generalization in diverse real-world environments. The current implementation also focuses only on gesture-to-text conversion and does not include Natural Language Processing (NLP), sentence-level interpretation, or speech synthesis capabilities. Future improvements involving larger datasets, dynamic gesture modeling, temporal filtering, and multimodal communication features can help overcome these limitations and improve overall system robustness.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

TABLE VII
FALSE POSITIVE AND FALSE NEGATIVE ANALYSIS

Gesture Class	False Positives	False Negatives
A	3	2
M	1	0
N	4	1
S	0	1
T	2	10
Blank	0	0

XXIII. REPRODUCIBILITY AND OPEN SCIENCE CONSIDERATIONS

The proposed ASL recognition system is developed using widely accessible tools and frameworks, including Python, OpenCV, TensorFlow, and Keras, which supports reproducibility and ease of implementation across different computing environments. The system architecture, preprocessing pipeline, model configuration, and evaluation methodology are documented in detail to enable replication of experimental results. Standard hardware components such as webcams and CPU-based systems are used, ensuring that the proposed approach remains cost-effective and practically reproducible without specialized equipment.

From an open science perspective, the project can be further strengthened by publicly releasing the source code, trained model weights, dataset structure, and experimental configurations through open repositories. Sharing these resources would promote transparency, encourage collaborative improvements, and facilitate comparative research in sign language recognition and assistive AI systems. Future work may also include standardized benchmarking and cross-dataset evaluation to improve reproducibility and support broader research adoption.

XXIV. PRACTICAL IMPLICATIONS AND DEPLOYMENT SCENARIOS

The proposed ASL recognition system has significant practical implications as an assistive communication tool for hearing- and speech-impaired individuals. By converting hand gestures into textual output in real time, the system can help reduce communication barriers in educational institutions, workplaces, healthcare environments, and public service interactions. The lightweight CNN architecture and low-cost hardware requirements make the system accessible and suitable for deployment on standard consumer devices without the need for specialized equipment.

The system can be deployed in various real-world scenarios such as smart classrooms, customer support kiosks, hospitals, and human–computer interaction systems where real-time gesture interpretation is beneficial. Since the current implementation operates using a webcam and local processing, it can function as a portable and privacy-preserving solution. Future deployment possibilities include integration with mobile applications, embedded edge-AI devices, and speech synthesis modules to provide a complete multimodal communication platform for everyday assistive use.

XXV. COMPARATIVE ANALYSIS WITH EXISTING SOLUTIONS

The proposed ASL recognition system demonstrates competitive performance compared to existing vision-based sign language recognition approaches while maintaining low computational and hardware requirements. Traditional sensor-based systems often require specialized devices such as data gloves or motion sensors, which increase system cost and reduce user convenience. In contrast, the proposed system utilizes a standard webcam and computer vision techniques, making it more accessible and practical for real-time deployment.

Compared to conventional machine learning methods that rely on handcrafted feature extraction, the CNN-based architecture automatically learns hierarchical gesture features and achieves improved classification accuracy and robustness. The proposed model attains an overall accuracy of 94.89 with efficient real-time inference on CPU-based systems, demonstrating balanced performance across multiple ASL gesture categories. Although advanced deep learning systems incorporating dynamic gesture recognition, Natural Language Processing (NLP), and speech synthesis may provide broader functionality, they typically require larger datasets, higher computational resources, and more complex architectures. The current implementation offers a lightweight, cost-effective, and scalable solution that establishes a strong foundation for future extensions toward more advanced assistive communication systems.

TABLE VIII
COMPARISON WITH BASELINE CLASSIFICATION TECHNIQUES

Technique	Accuracy	F1-score	Real-Time Support
k-NN	84.72	83.95	Limited
SVM	88.43	87.80	Moderate
Traditional CNN	91.26	90.74	Yes
Proposed CNN Model	94.89	94.81	Yes

XXVI. RESULTS AND DISCUSSION

The experimental results demonstrate that the proposed CNN-based ASL recognition system achieves reliable and efficient real-time gesture classification performance. The model obtained an overall accuracy of 94.89, with strong precision, recall, and F1-score values across most gesture categories. Training and validation accuracy curves indicate stable convergence with minimal overfitting, while the corresponding loss curves show consistent reduction in training and validation error throughout the learning process. The confusion matrix further confirms effective gesture recognition, with most predictions concentrated along the diagonal, indicating correct classification of gesture classes.

Class-wise analysis reveals particularly strong performance for the gestures “S” and “Blank,” which achieved near-perfect classification results. However, minor misclassification was observed between visually similar gestures such as “T” and “N,” highlighting the impact of gesture ambiguity and similarity in finger orientation. The ablation study demonstrates

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

that preprocessing techniques such as ROI extraction, normalization, grayscale conversion, and filtering logic significantly contribute to model accuracy and stability. Resource utilization and latency analysis show that the lightweight CNN architecture enables smooth real-time execution on standard CPU-based systems without requiring GPU acceleration. Overall, the proposed system provides a cost-effective and scalable assistive communication solution while establishing a strong foundation for future enhancements involving dynamic gesture recognition, NLP integration, and speech synthesis.

TABLE IX
LATENCY ANALYSIS ACROSS GESTURE CATEGORIES

Gesture Category	Average Inference Time (ms)	Response Stability
A	34	Stable
M	36	Stable
N	38	Moderate
S	33	Stable
T	41	Moderate
Blank	29	Highly Stable

XXVII. EXTENDED FUTURE RESEARCH DIRECTIONS

Future research directions for the proposed ASL recognition system include extending the current framework from static gesture recognition to dynamic and continuous sign language interpretation using sequence-based deep learning models such as Long Short-Term Memory (LSTM) networks, Recurrent Neural Networks (RNNs), or 3D CNNs. Integrating Natural Language Processing (NLP) techniques can enable sentence-level interpretation and contextual understanding, while Text-to-Speech (TTS) modules can provide complete multimodal communication support. Additional improvements may involve expanding the dataset with more diverse lighting conditions, backgrounds, and user variations to improve generalization and robustness in real-world environments. Future work may also explore deployment on mobile and embedded edge-AI platforms for portable assistive applications, incorporation of temporal smoothing and adaptive thresholding for improved prediction stability, and implementation of privacy-preserving and secure inference mechanisms to support reliable large-scale deployment.

XXVIII. SCOPE AND FUTURE WORK

Future enhancements include:

- Extend the current system from static gesture recognition to dynamic and continuous sign language interpretation using sequence-based deep learning models such as LSTM and RNN architectures.
- Integrate Natural Language Processing (NLP) techniques for sentence-level interpretation and contextual understanding of recognized gestures.
- Incorporate Text-to-Speech (TTS) functionality to convert recognized text into audible speech for complete multimodal communication support.
- Improve system robustness by expanding the dataset with diverse users, lighting conditions, hand orientations, and complex backgrounds to enhance generalization capability.
- Deploy the system on mobile and embedded edge-AI platforms to enable portable, low-cost, and real-time assistive communication applications.

XXIX. ETHICAL CONSIDERATIONS

The proposed ASL recognition system is developed with the objective of promoting accessibility and inclusive communication for hearing- and speech-impaired individuals. Ethical considerations primarily involve ensuring user privacy, fairness, transparency, and responsible use of AI-based assistive technologies. Since the system processes webcam-based gesture data, maintaining confidentiality and preventing unauthorized access to visual information are important concerns. The current implementation performs local processing without requiring cloud-based data transmission, which helps reduce privacy risks. Additionally, care must be taken to ensure that the training dataset represents diverse users, hand shapes, and environmental conditions to minimize bias and maintain fair performance across different individuals. The system is intended solely for assistive and supportive communication purposes and should not be used for surveillance, unauthorized monitoring, or discriminatory applications. Future enhancements should incorporate stronger security measures, informed user consent mechanisms, and transparent model evaluation practices to ensure ethical and trustworthy deployment in real-world environments.

XXX. CONCLUSION

The proposed ASL recognition system presents a lightweight and effective real-time hand gesture classification framework using computer vision and deep learning techniques. By combining preprocessing operations such as ROI extraction, grayscale conversion, normalization, and CNN-based feature extraction, the system achieves an overall accuracy of 94.89 while maintaining low computational overhead suitable for real-time execution on standard hardware. Experimental evaluation through performance metrics, confusion matrix analysis, ablation studies, latency measurements, and resource utilization assessment demonstrates reliable and balanced classification performance across multiple gesture categories. The system offers a cost-effective assistive communication solution for hearing- and speech-impaired individuals and establishes a strong foundation for future enhancements including dynamic gesture recognition, Natural Language Processing (NLP), speech synthesis, and deployment on mobile or embedded edge-AI platforms.

REFERENCES

[1] P. Verma and K. Badli, 2022 – "Real-Time Sign Language Detection using TensorFlow, OpenCV and Python" (International Journal for Research)
[2] A. Thakur et al., 2020 – "Real Time Sign Language Recognition and Speech Generation" (Journal of Engineering and Applied Sciences)
[3] A. Das et al., 2018 – "Sign Language Recognition Using Deep Learning on Custom Processed Static Gesture Images" (IEEE Conference)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

[4] A. Rao Gondu et al., 2018 – "Deep Convolutional Neural Networks for Sign Language Recognition" (Springer)
[5] D. Golekar et al., 2022 – "Sign Language Recognition using Python and OpenCV" (International Journal of Scientific Research)
[6] A. Kumar et al., 2022 – "Sign Language Recognition Using Convolutional Neural Network" (Springer)
[7] R. Cui et al., 2017 – "Recurrent Convolutional Neural Networks for Continuous Sign Language Recognition by Staged Optimization" (IEEE CVPR)
[8] Y. Zhao and L. Wang, 2018 – "The Application of Convolution Neural Networks in Sign Language Recognition" (International Conference on Intelligent Human-Machine Systems)
[9] C. C. de Amorim et al., 2019 – "Spatial-Temporal Graph Convolutional Networks for Sign Language Recognition" (Springer)
[10] F. Yasir et al., 2017 – "Bangla Sign Language Recognition using Convolutional Neural Network" (International Conference on Intelligent Computing)
[11] S. Chavan et al., 2021 – "Convolutional Neural Network Hand Gesture Recognition for American Sign Language" (IEEE Access)
[12] L. Pigou et al., 2014 – "Sign Language Recognition using Convolutional Neural Networks" (ECCV Workshops)
[13] N. Pugeault and R. Bowden, 2011 – "Spelling it Out: Real-Time ASL Fingerspelling Recognition" (IEEE ICCV Workshops)
[14] K. L. Cheng et al., 2020 – "Fully Convolutional Networks for Continuous Sign Language Recognition" (arXiv)
[15] OpenRouter AI, "API Documentation," https://openrouter.ai/docs.
[16] OpenAI, "Moderation API," https://platform.openai.com/docs/guides/moderation.