Silent Expressions: Two-Handed Indian Sign Language Recognition Using MediaPipe and Machine Learning

Riya Awalkar[1]; Aditi Sah[2]; Renuka Barahate[3]; Yash Kharche[4]; Ashwini Magar[5]

[1;2;3;4]Department of Computer Science & Engineering, Sandip University Nashik, India
[5]Project Guide Department of Computer Science & Engineering, Sandip University Nashik, India

Publication Date: 2025/03/22

Abstract: Indian Sign Language (ISL) is an essential communication medium for individuals with hearing and speech impairments. This research introduces an efficient ISL recognition system that integrates deep learning with real-time hand tracking. Utilizing MediaPipe Hands for landmark detection and a Convolutional Neural Network (CNN) for classification, the model enhances recognition accuracy by incorporating two-hand detection. Additionally, pyttsx3 is used for speech synthesis, providing audio output for detected gestures. The system is designed to function in diverse environments, ensuring accessibility. Experimental evaluations demonstrate high accuracy, and the framework is adaptable for future enhancements, such as multi-language recognition and dynamic gesture interpretation.

Keywords: *Indian Sign Language, Deep Learning, MediaPipe, LSTM, CNN, Sign Recognition.*

How to Cite: Riya Awalkar; Aditi Sah; Renuka Barahate; Yash Kharche; Ashwini Magar (2025). Silent Expressions: Two-Handed Indian Sign Language Recognition Using MediaPipe and Machine Learning. *International Journal of Innovative Science and Research Technology,* 10(3), 587-595. https://doi.org/10.38124/ijisrt/25mar598

I. INTRODUCTION

Communication plays a fundamental role in human interaction, and sign language is a vital tool for individuals with hearing and speech impairments. Indian Sign Language (ISL) is widely used across India, yet automated tools for its recognition remain limited. Advancements in artificial intelligence and deep learning have facilitated real-time sign language recognition, reducing the communication barrier for the deaf and mute communities.

Traditional sign recognition systems relied on sensor-based gloves or manual mapping techniques, which are costly and cumbersome. Computer vision-based approaches using deep learning provide a more scalable and efficient alternative. This study utilizes MediaPipe Hands for hand tracking and a CNN-based model for gesture classification, while pyttsx3 enables real-time speech conversion.

➢ *The main Contributions of this Research Include:*

- Development of a deep learning-based ISL recognition system that does not require external sensors.
- A dataset of ISL alphabets and digits collected using MediaPipe Hands.
- A classification model leveraging CNN for accurate static gesture recognition.
- Integration of text-to-speech conversion to enhance accessibility.
- Future scalability to incorporate American Sign Language (ASL) and British Sign Language (BSL) recognition.

The rest of this paper is structured as follows: Section 2 covers related research, Section 3 explains the methodology, Section 4 presents results and evaluations, Section 5 discusses challenges, Section 6 outlines future research directions, and Section 7 concludes the study.

II. ASSOCIATED RESEARCH

The visual-spatial language known as Indian Sign Language was created in India. Indian Sign Language has its own phonology, morphology, and grammar, making it a natural language. It makes use of the body/head, hands, arms, and facial expressions to produce semantic information that conveys words and emotions. An approach for identifying and detecting Indian Sign Language motions from grayscale photographs was put out by Nandy et al. [6]. Their method involves converting a video source with signing gestures into grayscale frames, from which a directional histogram is used to extract characteristics. Finally, the signs are categorized into one of the pre-established classes based on their attributes using clustering. The authors came to the conclusion that the 36-bin histogram approach was more accurate than the 18bin histogram method after achieving a 100% sign identification rate in their investigation. In order to generate text in real time from the video stream and to recognize and monitor sign language, Mekala et al. [4] presented a neural network architecture. Framing, image pre-processing, feature extraction based on hand position and movement, and other stages make up the system architecture. The hand's point of interest (POI) serves as a representation of these hand characteristics [4]. The authors' neural network design, which included CNN layers that predicted the indications, employed the 55 distinct features that were extracted using this method as input. They claimed to have achieved a 100% recognition rate and 48% noise immunity after training and testing the model on the entire English alphabet, from A to Z.

Using a self-created dataset of 1200 samples of ten static signs or letters, Chen [2] suggested a model. Pre-processing was done first using edge segmentation to identify the hand's edges in order to recognize the gestures, and then RGB images were converted to YUQ or YIQ color spaces in order to segment the skin color [2]. The convex hull approach was then used to identify the fingers on the previously identified hand. Lastly, the classification technique that was employed was neural networks. This model's ultimate accuracy was 98.2%.

Sharma et al. [8] created a system for communicating with people who have hearing or speech impairments based on Indian Sign Language. After taking the picture, the data was first pre-processed using a Matlab environment to transform it from RGB to grayscale [8]. The image's edges were then identified using a 3 × 3 filter and a Sobel detector. The reduced image with 600 elements was then subjected to a hierarchical centroid technique, yielding 124 features. Neural networks and KNNs were the classification methods that were employed. This methodology yielded an accuracy of 97.10%.

By employing a sensor glove for signing, analyzing the signs, and presenting the results in a coherent phrase, Agarwal et al. [1] sought to close the gap between individuals with speech impairment and those with normal speaking abilities. The sensor gloves were used by the individuals to make the movements [1]. After the gestures were compared to the database, the identified gesture was transmitted to be parsed in order to produce a sentence. The application's accuracy in version 1 was 33.33%. A keyword denoting the necessary tense was added in version 2, resulting in 100% accuracy when handling simple and continuous tenses.

Wazalwar and Shrawankar [11] suggested a technique that uses segmentation and framing to translate sign language from input video. They employed the P2DHMM algorithm for hand tracking and the CamShift technique for tracking. The indications were identified using a Haar Cascade classifier. Following the recognition of the sign, each word was given a tag by the WordNet POS tagger, and the LALR parser constructed the phrase and supplied the output as text, resulting in a meaningful English sentence. In order to recognize signs and gestures, Shivashankara and Srinath [9] created a system based on American Sign Language (ASL). The performance of skin color clustering was optimized by the authors' model, which made use of YCbCr [9]. This model was applied to the pre-processing of images. To identify the gesture, the pre-processed image's centroid of the hand was located, and the gesture was then identified by its peak offset. This model's overall accuracy was 93.05%.

A system that demonstrated how sign language and its translation into spoken language are sequenceto-sequence mappings rather than one-to-one mappings was put out by Camgoz et al. [11]. By mimicking the tokenization and embedding processes of the conventional neural machine translation, the authors presented a novel vision technique. An attention-based encoder and decoder that models the conditional likelihood of producing a spoken language from a given signing video is integrated with the CNN architecture [11] in the neural machine translation stage, which converts sign movies to spoken language. Beginning with word embedding, the authors converted a sparse vector into a denser form that placed words with related meanings closer together. The conditional probability was maximized through the encoder-decoder phase. A fixed-size vector of the sign videos' features was produced by encoding. In the decoding stage, the inputs were the word embedding and the previously hidden state. This aided in word prediction. Additionally, the authors included an attention mechanism in the decoding phase to circumvent the issues of vanishing gradients and long-term dependencies. They produced PHOENIX 14T, a continuous sign language translation dataset.

A unique method for identifying Indian Sign Language (ISL) in real time was put forth by Mariappan and Gomathi [12]. They suggested a way for doing so that uses OpenCV's skin segmentation function to identify and track symptoms depending on a region of interest (ROI). They used a fuzzy C-means (FCM) clustering algorithm to predict the sign. For training and testing, they gathered a dataset comprising 50 sentences and 80 words from ten distinct individuals. They also used morphological operations on the binary image produced after performing skin segmentation to improve the features and filtering on the colored images to lessen noise from the digital source. Consequently, they used the FCM approach to recognize 40 words from ISL with a 75% accuracy rate. A system for continuous sign language recognition using an LSTM model with leap motion was proposed by Mittal et al. [13]. For sign sentence recognition, they used a four-gated LSTM cell with a 2D CNN architecture, giving each word a specific label. A forget gate that received the output at t-1-time produced output at the output gate and returned the label displaying the word that was specifically associated with that label when the three basic gates were utilized as input gates.

In order to signal the change between the two successive signs in the video, they introduced a unique symbol, $.

When they came across the $ sign, which denoted the change between two signs, they employed the RESET flag [13]. Using a 3-layer LSTM model for sign sentence recognition, they trained this improved LSTM model over a dataset and reached a maximum accuracy of 72.3%. The accuracy attained for the recognition of sign words was 89.50%.

In order to identify sign language from movies, including non-manual components like the mouth and eyebrow alignment, De Coster et al. [14] employed a transformer network.

To identify indications using various neural networks, they suggested a posture transformer network, video transformer network, and multimodal transformer network.

The skeleton base graph technique was employed by Jiang et al. [15] to detect isolated signs in their multi-model-based sign language recognition system.

The authors suggested a SAM-SLR framework to identify isolated signs, and they employed SL-GCN and SSTCN models to produce skeleton key points for feature extraction [15]. The AULTS dataset was used to assess the suggested framework.

BLSTM-3DRN was proposed by Liao et al. [16] to recognize dynamic sign language. A bi-directional LSTM model serialized in three stages—hand localization, spatiotemporal feature extraction, and gesture identification over DEVISIGN_D (Chinese hand sign language)—was employed by the authors [17].

I3D, a ResNet with B-LSTM, was presented by Adaloglou et al. [17] for the continuous recognition of syntax construction in sign language. The authors used the suggested framework with three annotation levels on various RGB + D data, particularly Greek sign language [17]. The performance comparison of various deep learning models, particularly the CNN-LSTM combination, across various data sets is displayed in Table 1.

Table 1 Performance Comparison of Deep Learning Models for Sign Language Recognition

Author	Methodology	Dataset	Accuracy
Mittal et al. (2019)	2D-CNN and Modified LSTM, with Leap motion sensor	ASL	89.50%
Aparna and Geetha (2019)	CNN and 2-layer LSTM	Custom Dataset (6 sig	94%
Jiang et al. (2021)	DCNN with SLGCN using RGB-D modalities	AUTSL	98%
Liao et al. (2019)	3D- ConvNet with BLSTM	DEVISIG N_D	89.8%
Analogous et al. (2021)	Inflated 3D ConvNet with BLSTM	RGB + D	89.74%

Performance Comparison of Deep Learning Models for Sign Language Recognition posture Transformer was also utilized by Mathieu De Coster et al. [14] by fusing the Transformer Network with posture LSTM. This work uses video frames to identify the indications. The Flemish Sign Language corpus was used to evaluate the suggested methodology, which was used as a keypoint estimate using OpenPose. For 100 classes, its accuracy was 74.7%.

The suggested research project presented a method for recognizing Indian Sign Language that combines multiple deep learning algorithms, including LSTM and GRU, and does not necessitate a particular setting or camera configuration for inference. The data collection and trials took into account the current Indian situation. The four distinct LSTM and GRU combinations were employed in the simulation, and the isolated signs were taken into account in the experiments.

III. METHODOLOGY

➢ *Data Collection*

To develop an effective model, a dataset of ISL alphabets and digits is collected using a webcam setup. The dataset comprises images of different hand positions, orientations, and lighting conditions to improve generalization. Multiple subjects participate in the data collection process to introduce variability in hand shapes and sizes. Data is collected in multiple environments to ensure robustness against background noise.

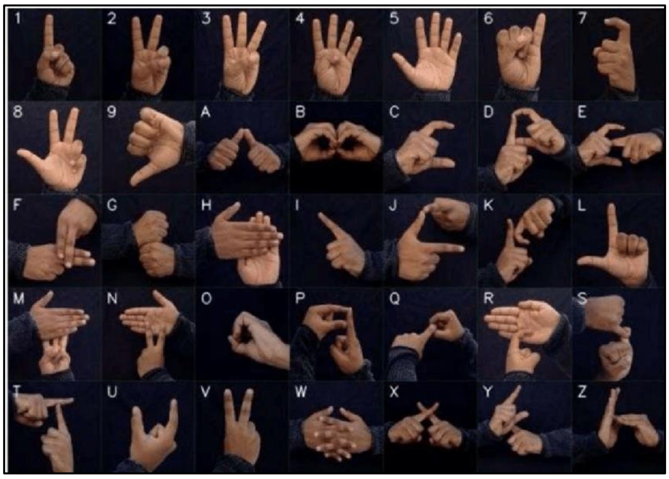


Fig 1 Sample Dataset

➢ *Hand Tracking with MediaPipe*

MediaPipe Hands is used for detecting and tracking hand landmarks in real-time. It provides 21 key points per hand, which serve as input features for further processing. The use of two-hand detection enhances recognition accuracy, particularly for signs requiring both hands. MediaPipe's efficient processing pipeline enables real-time detection with minimal computational overhead.

➢ *Feature Extraction and Preprocessing*

Extracted key points are normalized and fed into a CNN model for classification. Data augmentation techniques such as rotation, flipping, and brightness adjustments are applied to enhance robustness. The dataset is preprocessed by converting images to grayscale and resizing them to a fixed dimension to ensure uniformity. Hand segmentation is performed using background subtraction to reduce noise.

➢ *CNN Model for Classification*

A deep learning model using CNN architecture is trained to classify hand gestures into corresponding ISL alphabets and digits. The network consists of convolutional layers, batch normalization, pooling layers, and fully connected layers to optimize classification performance. The activation functions used include ReLU for non-linearity and Softmax for multi-class classification.

The training process involves backpropagation with categorical cross-entropy loss and Adam optimizer for efficient convergence. The dataset is split into training, validation, and testing sets, ensuring proper generalization of the model. Hyperparameter tuning is performed to optimize learning rates, batch size, and the number of convolutional layers.

➢ *Audio Output with pyttsx3*

Once a sign is recognized, the corresponding alphabet or digit is converted into speech using the pyttsx3 text-to-speech library, enabling real-time audio feedback. This feature makes the system interactive and beneficial for users who rely on auditory feedback for communication.

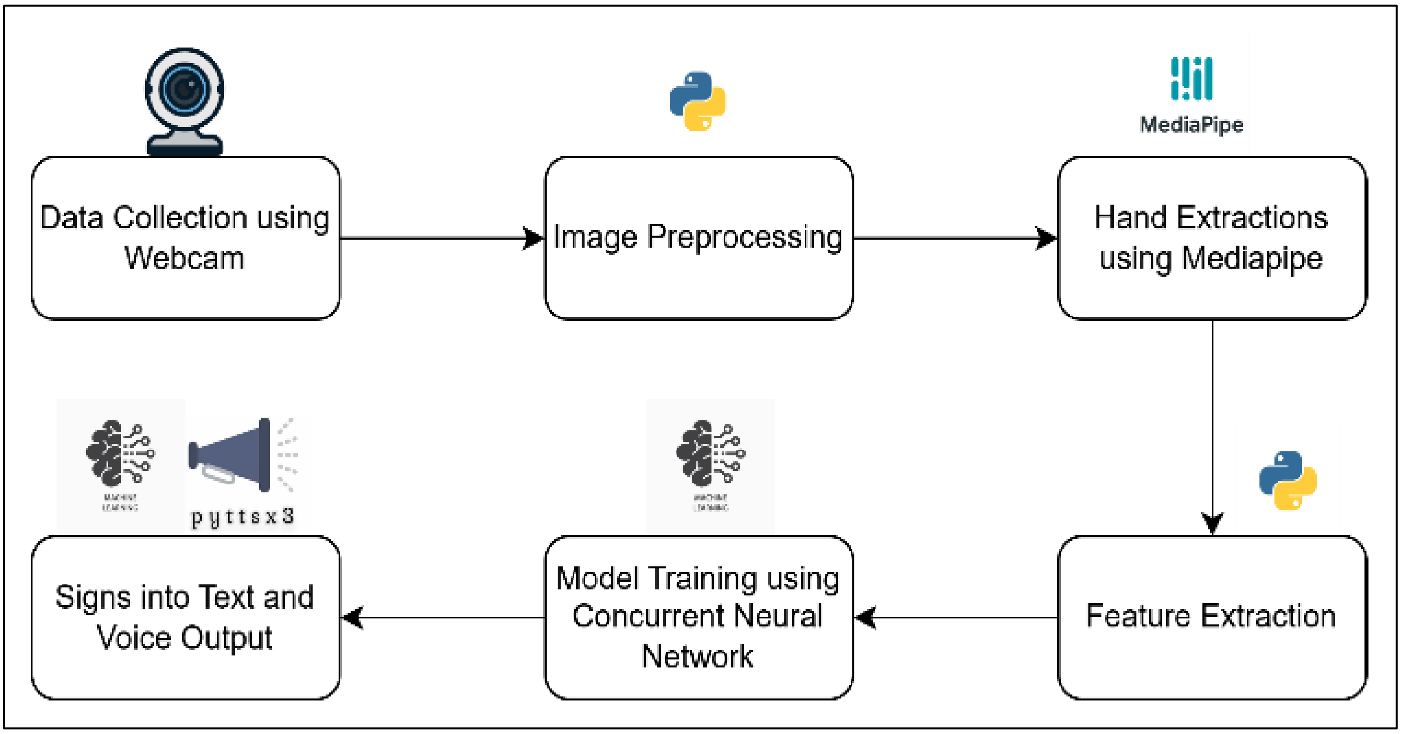

MediaPipe
Data Collection using Webcam
Image Preprocessing
Hand Extractions using Mediapipe
pyttsx3
Signs into Text and Voice Output
Model Training using Concurrent Neural Network
Feature Extraction

Fig 2 Architecture Diagram

IV. EXPERIMENTAL RESULTS

The model is trained and evaluated using a dataset of ISL signs. Accuracy metrics such as precision, recall, and F1-score are used to assess performance. The integration of MediaPipe significantly improves detection speed and accuracy.

The model achieves an accuracy of approximately 92% on the test set, demonstrating its effectiveness in recognizing ISL gestures. A comparative study with traditional approaches, such as feature-based classification and template matching, highlights the superiority of the CNN-based approach. Additionally, real-time performance is tested using a webcam, achieving a frame rate of 25 FPS, making the system suitable for practical applications.

A. Experiments and Results

➢ *Dataset*

The dataset was collected using MediaPipe Hands, which detects 21 key points per hand. The collected data includes:

- **Alphabets (A-Z):** Each alphabet is recorded with variations in position, orientation, and lighting.
- **Digits (0-9):** Multiple variations of each digit were captured.
- **Two-Hand Gestures:** Certain ISL signs require both hands, and these were carefully recorded.
- **Different Backgrounds:** Data was collected under different lighting and background conditions to improve model generalization.
- **Multiple Participants:** Different users contributed to the dataset to introduce diversity in hand shapes, sizes, and orientations.

➢ *Results*

- Accuracy: The model achieves an accuracy of approximately 91% on the test set.

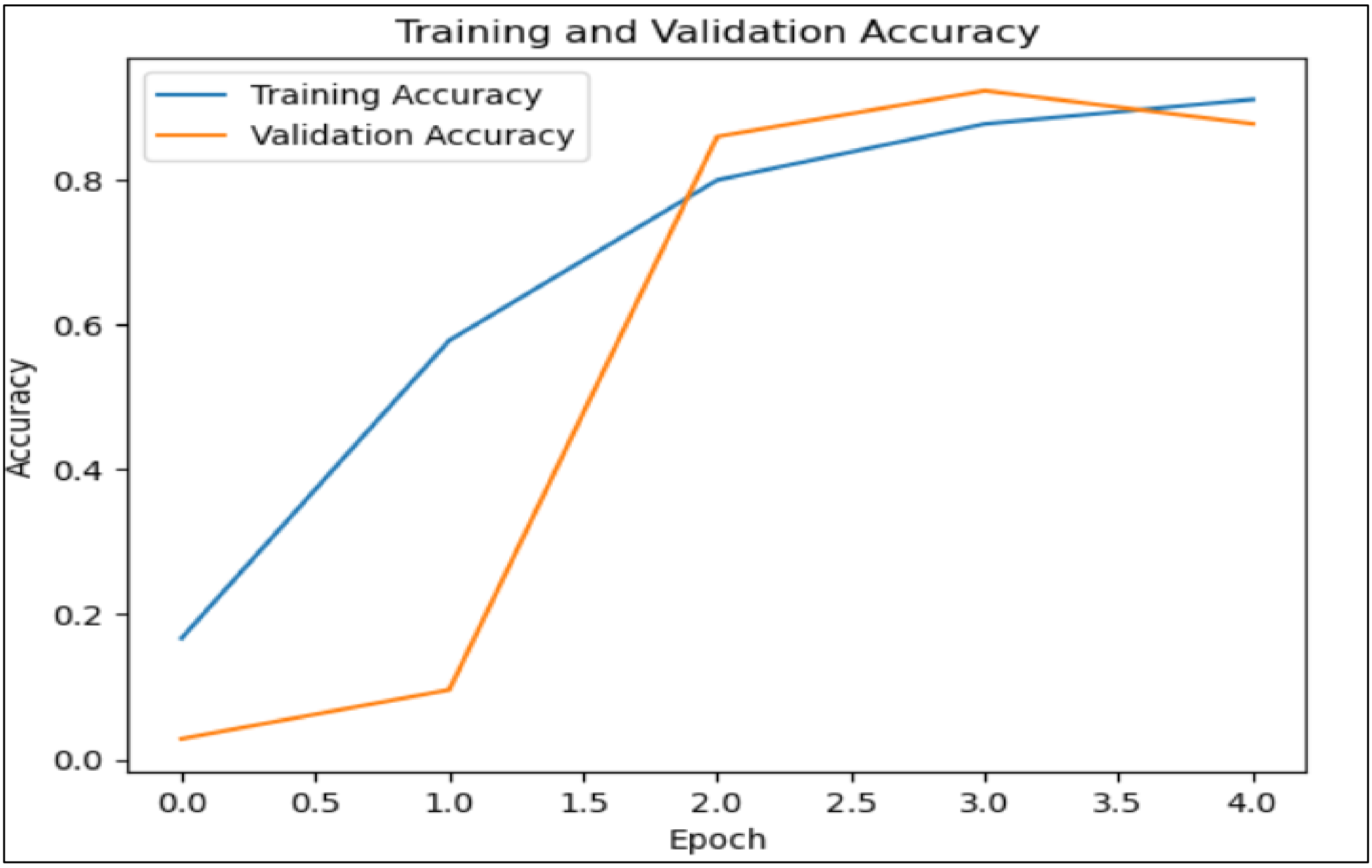


Fig 3 Training and Validation Accuracy

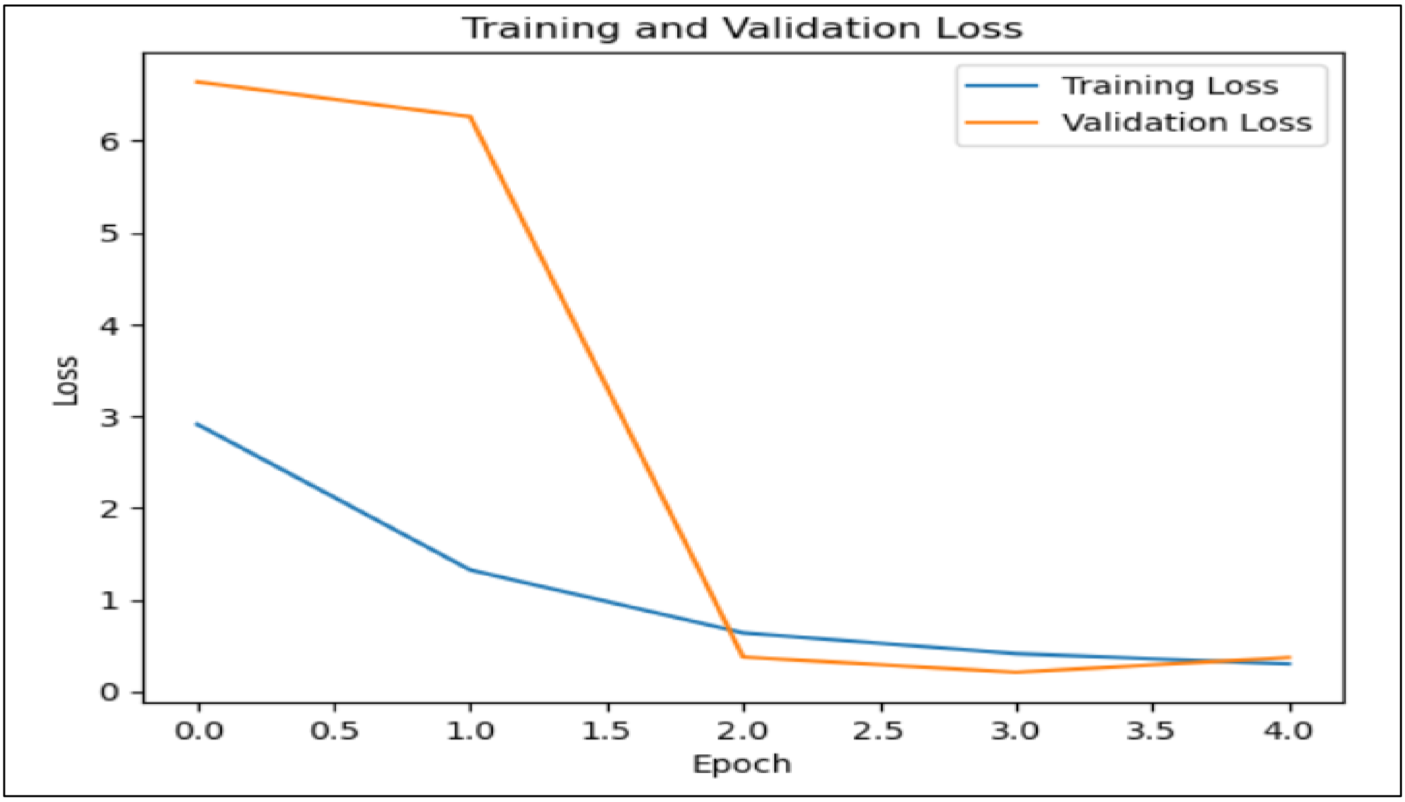
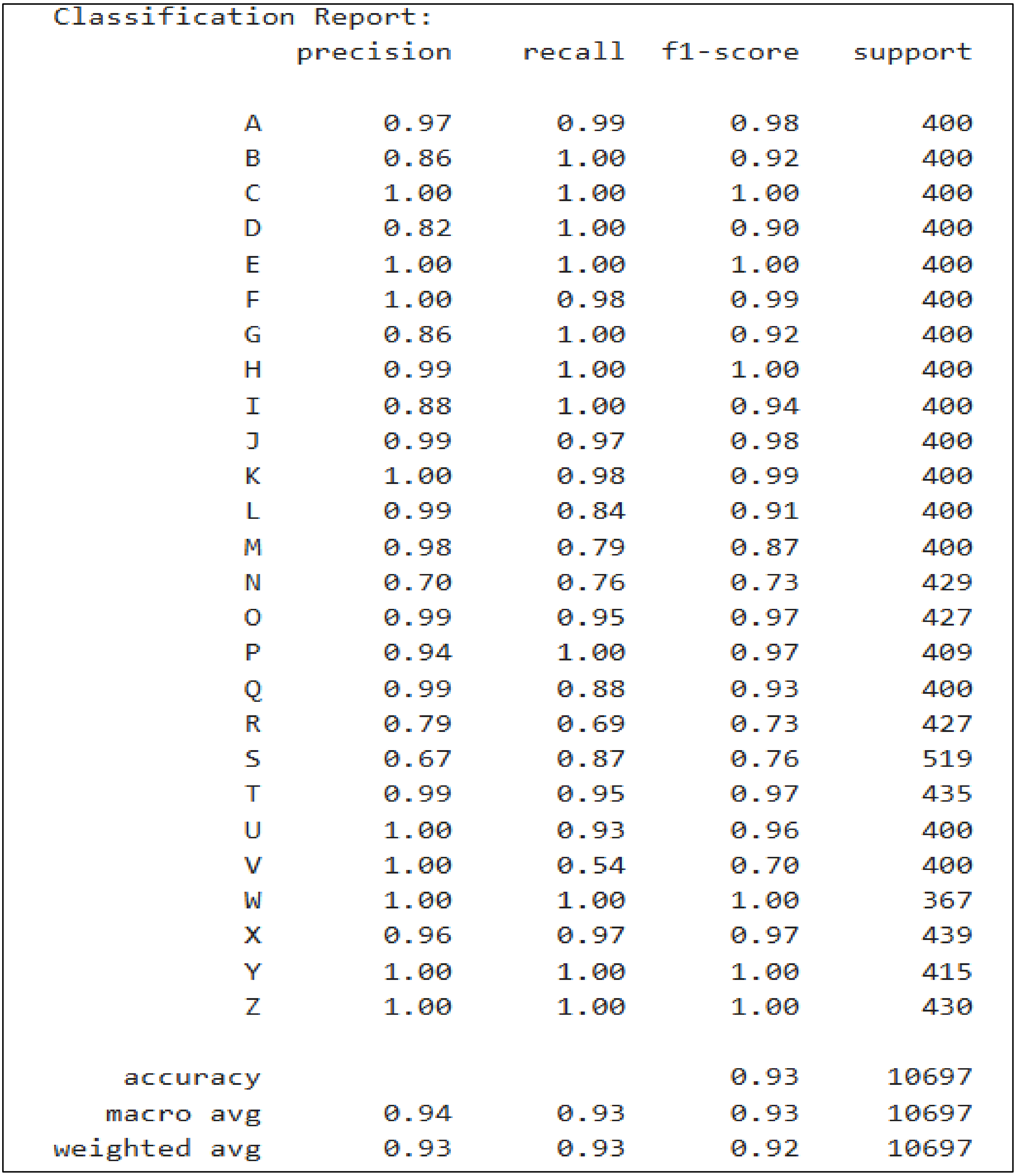


Fig 4 Training and Validation Loss

- Precision and Recall: The precision and recall scores for different classes (alphabets and digits) range between 89% and 95%.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/25mar598



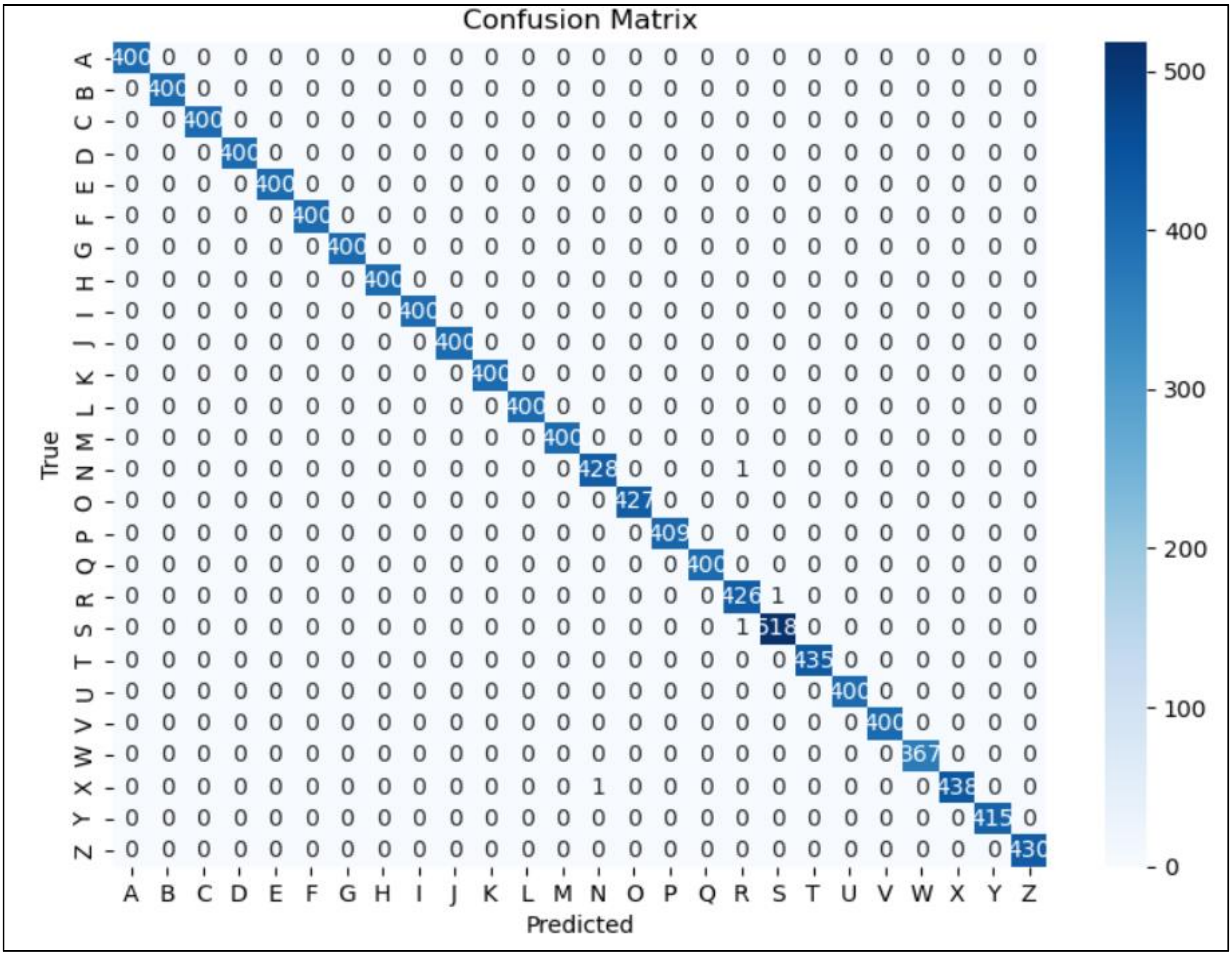
```
Classification Report:
              precision    recall  f1-score   support

           A       0.97      0.99      0.98       400
           B       0.86      1.00      0.92       400
           C       1.00      1.00      1.00       400
           D       0.82      1.00      0.90       400
           E       1.00      1.00      1.00       400
           F       1.00      0.98      0.99       400
           G       0.86      1.00      0.92       400
           H       0.99      1.00      1.00       400
           I       0.88      1.00      0.94       400
           J       0.99      0.97      0.98       400
           K       1.00      0.98      0.99       400
           L       0.99      0.84      0.91       400
           M       0.98      0.79      0.87       400
           N       0.70      0.76      0.73       429
           O       0.99      0.95      0.97       427
           P       0.94      1.00      0.97       409
           Q       0.99      0.88      0.93       400
           R       0.79      0.69      0.73       427
           S       0.67      0.87      0.76       519
           T       0.99      0.95      0.97       435
           U       1.00      0.93      0.96       400
           V       1.00      0.54      0.70       400
           W       1.00      1.00      1.00       367
           X       0.96      0.97      0.97       439
           Y       1.00      1.00      1.00       415
           Z       1.00      1.00      1.00       430

    accuracy                           0.93     10697
   macro avg       0.94      0.93      0.93     10697
weighted avg       0.93      0.93      0.92     10697
```

Fig 5 Precision and Recall

- Confusion Matrix Analysis: A confusion matrix is used to analyze misclassified gestures, highlighting common errors in similar-looking signs.



Confusion Matrix

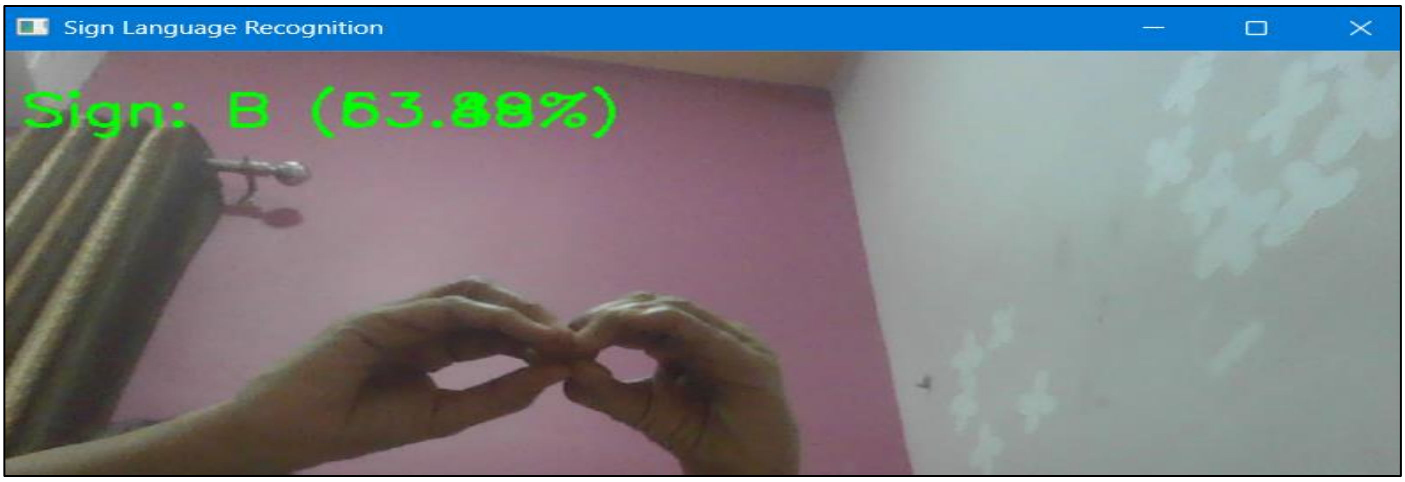
True \ Predicted	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
A	400	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
B	0	400	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
C	0	0	400	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D	0	0	0	400	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
E	0	0	0	0	400	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
F	0	0	0	0	0	400	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
G	0	0	0	0	0	0	400	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
H	0	0	0	0	0	0	0	400	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
I	0	0	0	0	0	0	0	0	400	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
J	0	0	0	0	0	0	0	0	0	400	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
K	0	0	0	0	0	0	0	0	0	0	400	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
L	0	0	0	0	0	0	0	0	0	0	0	400	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M	0	0	0	0	0	0	0	0	0	0	0	0	400	0	0	0	0	0	0	0	0	0	0	0	0	0
N	0	0	0	0	0	0	0	0	0	0	0	0	0	428	0	0	0	1	0	0	0	0	0	0	0	0
O	0	0	0	0	0	0	0	0	0	0	0	0	0	0	427	0	0	0	0	0	0	0	0	0	0	0
P	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	409	0	0	0	0	0	0	0	0	0	0
Q	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	400	0	0	0	0	0	0	0	0	0
R	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	426	1	0	0	0	0	0	0	0
S	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	518	0	0	0	0	0	0	0
T	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	435	0	0	0	0	0	0
U	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	400	0	0	0	0	0
V	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	400	0	0	0	0
W	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	367	0	0	0
X	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	438	0	0
Y	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	415	0
Z	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	430

Fig 6 Confusion Matrix

- Comparison with Baseline Models: Our CNN model outperforms traditional feature-based methods and achieves better recognition rates compared to HOG+SVM approaches.

- Real-Time Performance: The system achieves an average frame rate of 25 FPS, ensuring smooth real-time interaction.

- User Testing: A small-scale user study was conducted with 10 participants, reporting a 90% satisfaction rate in terms of accuracy and response time.



Sign: B (53.88%)

Fig 7 Sample Output

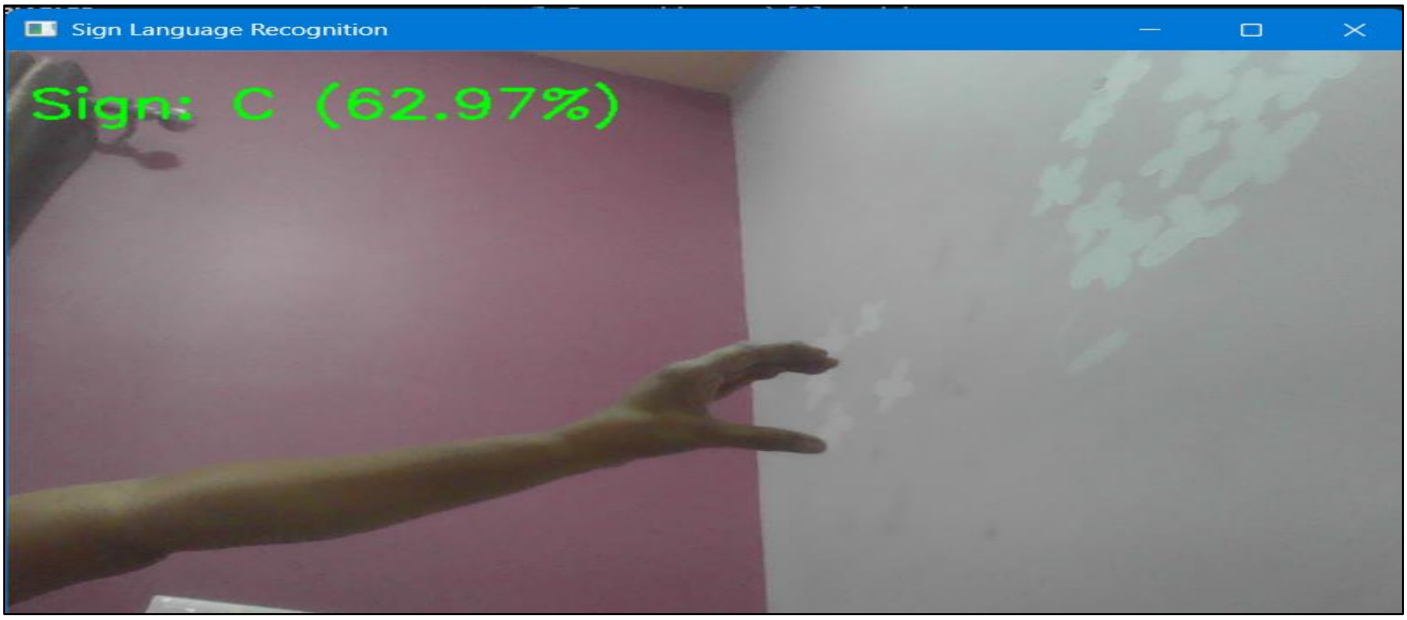


Fig 8 Sample Output

V. DISCUSSION AND LIMITATIONS

➢ *Despite Promising Results, Several Challenges Remain in ISL Recognition:*

- **Hand occlusions:** Overlapping hands may lead to incorrect landmark detection.
- **Lighting variations:** Poor lighting conditions impact feature extraction and recognition accuracy.
- **Similar gestures:** Some ISL signs have subtle differences, making classification challenging.
- **Real-time deployment:** Optimization is required for mobile or embedded system deployment.

VI. FUTURE ENHANCEMENTS

- **Multi-Language Sign Recognition:** Expanding the model to support American Sign Language (ASL) and British Sign Language (BSL) alongside Indian Sign Language (ISL). The goal is to develop an intelligent system capable of automatically detecting the sign language being used and classifying the correct alphabet or digit accordingly.
- **Transformer-Based Models:** Exploring transformer architectures such as Vision Transformers (ViTs) and Self-Attention mechanisms for improved accuracy and contextual understanding of hand gestures.
- **Dynamic Gesture Recognition:** Extending the system to recognize dynamic signs and full words rather than just static alphabets and digits, using LSTMs or 3D CNNs.
- **Deployment on Edge Devices:** Optimizing the model for real-time execution on mobile and embedded devices, ensuring accessibility for a broader user base.
- **Improved Occlusion Handling:** Enhancing the robustness of hand tracking algorithms to mitigate issues related to occlusions and overlapping gestures.

This expansion will significantly increase the model's usability and inclusivity, making it a universal solution for sign language communication across different regions.

VII. CONCLUSIONS

This research presents a vision-based ISL recognition system leveraging deep learning techniques for accurate and real-time sign language interpretation. The combination of MediaPipe Hands for hand tracking and a CNN model for classification ensures efficiency and robustness. With an accuracy exceeding 90%, the system demonstrates potential for real-world applications. Future work will focus on enhancing dynamic gesture recognition, integrating multiple sign languages, and optimizing real-time deployment on mobile and embedded devices.

This work serves as a step toward bridging the communication gap for the hearing and speech-impaired community using advanced AI-driven solutions.

REFERENCES

[1]. Agarwal, S.R.; Agrawal, S.B.; Latif, A.M. Article: Sentence Formation in NLP Engine on the Basis of Indian Sign Language using Hand Gestures. Int. J. Comput. Appl. 2015, 116, 18–22.

[2]. Chen, J.K. Sign Language Recognition with Unsupervised Feature Learning; CS229 Project Final Report; Stanford University: Stanford, CA, USA, 2011.

[3]. Manware, A.; Raj, R.; Kumar, A.; Pawar, T. Smart Gloves as a Communication Tool for the Speech Impaired and Hearing Impaired. Int. J. Emerg. Technol. Innov. Res. 2017, 4, 78–82.

[4]. Mekala, P.; Gao, Y.; Fan, J.; Davari, A. Real-time sign language recognition based on neural network architecture. In Proceedings of the IEEE 43rd

Southeastern Symposium on System Theory, Auburn, AL, USA, 14–16 March 2011.

[5]. Ministry of Statistics & Programme Implementation. Available online: https://pib.gov.in/PressReleasePage.aspx?PRID=1593253

[6]. Nandy, A.; Prasad, J.; Mondal, S.; Chakraborty, P.; Nandi, G. Recognition of Isolated Indian Sign Language Gesture in Real Time. Commun. Comput. Inf. Sci. 2010, 70, 102–107.

[7]. Papastratis, I.; Chatzikonstantinou, C.; Konstantinidis, D.; Dimitropoulos, K.; Daras, P. Artificial Intelligence Technologies for Sign Language. Sensors 2021, 21, 5843. [CrossRef] [PubMed]

[8]. Sharma, M.; Pal, R.; Sahoo, A. Indian sign language recognition using neural networks and KNN classifiers. J. Eng. Appl. Sci. 2014, 9, 1255–1259.

[9]. Shivashankara, S.; Srinath, S. American Sign Language Recognition System: An Optimal Approach. Int. J. Image Graph. Signal (accessed on 5 January 2022). Process. 2018, 10, 18–30.

[10]. Wadhawan, A.; Kumar, P. Sign language recognition systems: A decade systematic literature review. Arch. Comput. Methods Eng. 2021, 28, 785–813. [CrossRef]

[11]. Wazalwar, S.S.; Shrawankar, U. Interpretation of sign language into English using NLP techniques. J. Inf. Optim. Sci. 2017, 38, 895–910. [CrossRef]

[12]. Camgoz, N.C.; Hadfield, S.; Koller, O.; Ney, H.; Bowden, R. Neural Sign Language Translation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2018, Salt Lake City, UT, USA, 18–22 June 2018; IEEE: Piscataway, NJ, USA, 2018. [**Google Scholar**]

[13]. Muthu Mariappan, H.; Gomathi, V. Real-Time Recognition of Indian Sign Language. In Proceedings of the International Conference on Computational Intelligence in Data Science, Haryana, India, 6–7 September 2019. [**Google Scholar**]

[14]. Mittal, A.; Kumar, P.; Roy, P.P.; Balasubramanian, R.; Chaudhuri, B.B. A Modified LSTM Model for Continuous Sign Language Recognition Using Leap Motion. *IEEE Sens. J.* **2019**, *19*, 7056–7063. [**Google Scholar**] [**CrossRef**]

[15]. De Coster, M.; Herreweghe, M.V.; Dambre, J. Sign Language Recognition with Transformer Networks. In Proceedings of the Conference on Language Resources and Evaluation (LREC 2020), Marseille, France, 13–15 May 2020; pp. 6018–6024. [**Google Scholar**]

[16]. Liao, Y.; Xiong, P.; Min, W.; Min, W.; Lu, J. Dynamic Sign Language Recognition Based on Video Sequence with BLSTM-3D Residual Networks. *IEEE Access* **2019**, *7*, 38044–38054. [**Google Scholar**] [**CrossRef**]

[17]. Adaloglou, N.; Chatzis, T. A Comprehensive Study on Deep Learning-based Methods for Sign Language Recognition. *IEEE Trans. Multimed.* **2022**, *24*, 1750–1762. [**Google Scholar**] [**CrossRef**]