ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/25nov517

Computer Vision-Powered Indian Sign Language Recognition System

Dr. V. Sathiyasuntharam[1]; Harsh Nagar[2]; Divek Jain[3]; Harmanpreet Kaur[4]

[1;2;3;4]SSCSE, CSE Department, Sharda University, Greater Noida, Uttar Pradesh, India

Publication Date: 2025/11/19

Abstract: This systematic review analyzes eight pioneering ISL recognition studies from 2012 to 2024 for tracking the transition from conventional machine learning to advanced deep learning for helping the deaf community in India. Early methods, such as SVM with Hu Moments, attained up to 97.5% accuracy but relied on a small dataset and heavy preprocessing. The latest deep learning models, especially CNN-based, have achieved more than 99% accuracy for the recognition of the static alphabet. The system that represents the current state-of-the-art is a hybrid MF-DNet, which includes VGG-19, MediaPipe, and BiLSTM. MF-DNet recognizes dynamic words with an accuracy of 96.88% for 263 classes, addressing occlusion. However, severe challenges persist: the critical lack of standardized large-scale ISL datasets, extremely minimal studies regarding continuous sentence interpretation, and the absence of real-time deployment on mobile platforms. The future research focus will fall on the creation of the ISL-100K benchmark, application of Vision Transformers, and lightweight end-to-end bidirectional translation systems towards the ultimate goal of achieving more than 95% accuracy in real-time ISL sentence recognition on mobile.

How to Cite: Dr. V. Sathiyasuntharam; Harsh Nagar; Divek Jain; Harmanpreet Kaur (2025) Computer Vision-Powered Indian Sign Language Recognition System. *International Journal of Innovative Science and Research Technology*, 10(11), 814-819. https://doi.org/10.38124/ijisrt/25nov517

I. INTRODUCTION

Communication is a basic human right, but India's 2.8 million deaf and hearing-impaired citizens who use Indian Sign Language (ISL) as their main way of communicating face a huge barrier. The Rights of Persons with Disabilities (RPwD) Act 2016, which recognises ISL as an official language, has been passed in stages. However, this community still has a lot of trouble getting important services like healthcare, education, and jobs. This is mostly because there aren't enough certified sign language interpreters (about 1 in 500) and their services are too expensive (₹500–₹2,000 per hour).

However, research in ISL recognition remains significantly less developed compared to its counterparts, such as American Sign Language (ASL) and British Sign Language (BSL). While current ISL models can achieve over 99% accuracy for constrained static alphabet recognition, the practical challenge of dynamic word recognition still shows a much lower range (87% to 97%), and continuous sentence recognition—the foundation of natural conversation—is largely unexplored.

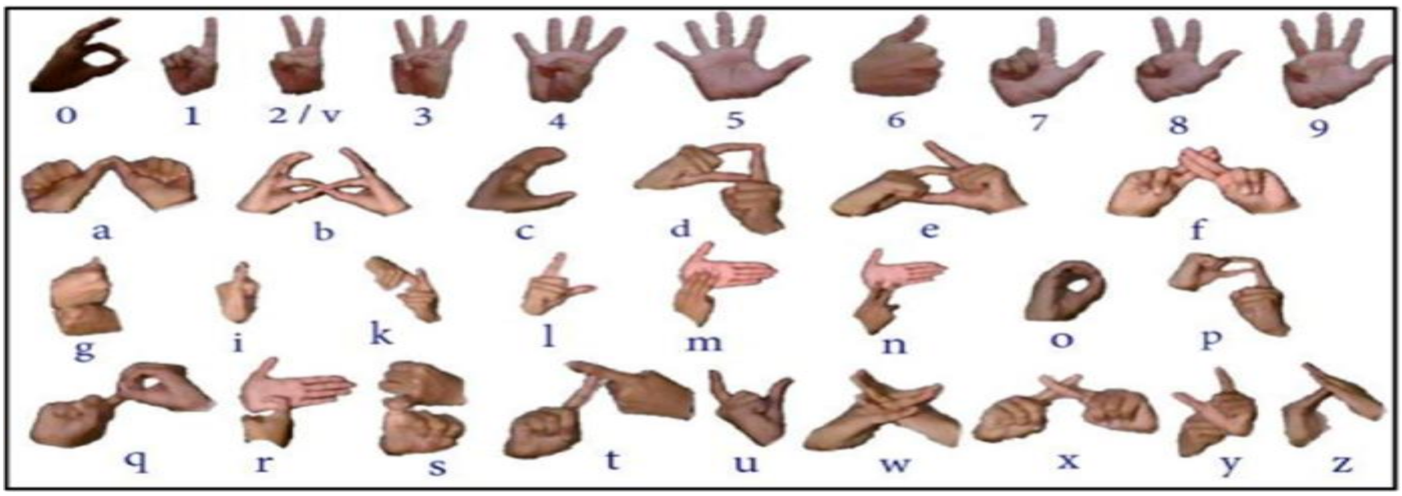

Fig 1 Indian Sign Language

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/25nov517

II. CHALLENGES IN INDIAN SIGN LANGUAGE RECOGNITION SYSTEM

Recent breakthroughs in computer vision and deep learning have provided unprecedented possibilities for SLRS development. The technology has advanced from glove-based systems with dedicated sensors to vision-based methods using off-the-shelf cameras and deep neural networks. Convolutional Neural Networks (CNNs), Long Short-Term Memory (LSTM) networks, and transfer learning methods using pre-trained models like VGG-19 and ResNet-50 have been extremely successful in image and video perception tasks. Pose estimation pipelines such as MediaPipe now offer real-time skeletal tracking functionality, allowing for hybrid methods combining appearance-based and keypoint-based features.

III. DATA ACQUISITION AND DATABASE

From employing small, proprietary datasets (80–540 images) gathered in extremely controlled environments for static signs, data acquisition for ISL recognition has changed to more recent efforts that concentrate on large-scale video-based datasets (such as ISLTranslate or iSign Benchmark). The main obstacle, though, is still the absence of a large-scale, standardised benchmark similar to ASL. The scale and demographic diversity required to properly train deep learning models for reliable, generalised, continuous sentence recognition are still lacking in existing datasets.

IV. HISTORY OF INDIAN SIGN LANGUAGE

With roots in Indo-Pakistani Sign Language (IPSL), Indian Sign Language (ISL) has developed organically as an indigenous language throughout the Indian subcontinent. It was suppressed for decades by oralism-focused educational policies. The first ISL Dictionary was produced in the early 1980s as a result of groundbreaking research, particularly by Dr. Madan Vashishta, demonstrating the language's status as a separate one. The Indian Sign Language Research and Training Centre (ISLRTC) was formally established in 2015 as the result of this endeavour. Importantly, ISL was given official legal recognition by the Rights of Persons with Disabilities (RPwD) Act 2016, sealing its place and requiring its use to meet the educational and communication needs of the deaf community in India.

V. LITERATURE SURVEY

Table 1 Literature Survey

No.	Authors	Title of the Paper	Benefits	Challenges
[1]	P.K. Athira, C.J. Sruthi, A. Lijiya [2019]	A Signer Independent Sign Language Recognition with Coarticulation Elimination from Live Videos: An Indian Scenario	Economical can be implemented with a mobile camera which makes it user-friendly	Not efficient under cluttered backgrounds and different illumination conditions
[2]	Mohito Jaiswal, Vaidehi Sharma, Abhishek Sharma, Snadeep Saini, Raghuvir Tomar [202]	An Efficient Binarized Neural Network for Recognizing 2 Hands Indian Sign Language Gesture in Real-Time Environment	This architectureachieves overall accuracy of 98.8% which is higher than other existing models	Misclassiffication of some sign of M, N, E
[3]	Sruthi C.J., and Lijiya [2019]	A Deep Learning Based Indian Sign Language Recognition System	Training accuracy of 99.93%	Facial expression and context analysis are the other part not included
[4]	Muneer AlHammadi, Ghulam Mahummad and Mahomed Amine Mektichke [2020]	Hand Gesture Recognition for Sign Language Using JDCNN	This outperformed four methods and showed comparable performance to the other two	Does not work for a live video feed
[5]	Sevgi Z. Gurbiz, Ali Cafer Gurbuz, Evie A. Malaia, Darrin J. Griffin, Chris S. Crawford, Mohammad Mahbubur Rahman [2020]	American Sign Language Recognition Using RF Sensing	Results demonstrate the eventuality of RF Sensing to give contactless ASL recognition capabilities in aid of ASL sensitive smart surroundings while surviving effectively in the dark and guarding user privacy	A massive intermodal database of connected native signing would be required to make meaningful interpretations for technology and algorithm correlation.

VI. EXPERIMENTAL APPROACH

Experimental approach explains the methods used for the accomplishment of the recognition of signs of Indian Sign Language. The steps are (1) segmentation, (2) finger tip finding algorithm and (3) PCA.

A. Segmentation

Color-based segmentation for ISL recognition begins by converting the image from RGB to HSV to define precise color ranges for the red and blue gloves. A binary mask is then created, isolating all pixels within these ranges. This mask is refined using morphological operations (like erosion/dilation) to remove noise. Finally, contour detection locates the

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/25nov517

boundaries of the cleaned mask, effectively segmenting the gloves for subsequent feature analysis.

➤ *Step1: Separating all the Components*

The system can examine the intensity contribution of each primary colour throughout the entire image thanks to this decomposition. In order to find the red and blue glove components for segmentation, this is a first step that is frequently carried out prior to switching to a more efficient colour space like HSV or occasionally used directly to define thresholds.

➤ *Step 2: Setting Threshold Levels*

The Red band of the red glove has a fixed, broad threshold that ranges from 0 to 255. Otsu's technique is applied to the Green and Blue bands. By automatically determining the ideal threshold to reduce the difference between the foreground (glove) and background pixels, this adaptive technique guarantees a clear separation from the other colour components of the image.

➤ *Step 3: Creating Red, Green, and Blue Masks*

- Binary Image Generation: Step 1 separates the colour bands into red, green, and blue masks.
- Mask Logic: Each mask is a binary image (containing only 0s and 1s):

✓ Value 1 (White): Indicates the presence of that color component where the pixel intensity falls within the defined threshold range.

✓ Value 0 (Black): Indicates the absence of that color component outside the threshold range.

- Red Mask Equation: The Red Mask (RM) is created by comparing the Red Band (RB) against its low (RTL) and high (RTH) thresholds:

$$RM = (RB \geq RTL) \wedge (RB \leq RTH)$$

- Purpose: The masks visually isolate the intensity contributions of each color channel. This is a crucial simplification step used to combine and refine the components later, leading to the final segmentation of the red and blue gloves.

➤ *Step 4: Creating the Object Mask*

The Red, Green, and Blue masks from the previous step are subjected to a logical 'AND' operation to produce the final Object Mask. Only when the desired colour, such as red, is present within its thresholds AND the other colours, such as green and blue, are sufficiently suppressed does this process produce a single binary image with white pixels (1). The target glove's clean silhouette is created by this action, making it suitable for contour analysis.

➤ *Step 5: Obtain Masked Component Images*

In this step, the target object's original colour information is isolated. The Object Mask made in Step 4 and each of the original colour bands (Red, Green, and Blue) are multiplied element-wise (an "AND" operation) to accomplish this. Three new images are produced as the result, with the background turned black (zeroed out) and only the pixels associated with the masked object (the glove) retaining their original colour intensity values. This draws attention to the segmented object's actual colour components.

➤ *Step 6: Concatenating All the Components*

The full-color segmented image is reconstructed by combining the final masked colour components. This completes the segmentation by producing the red object (glove) perfectly isolated against a pure black background.

Fig 2 RGB Image

Fig 3 a) Red Mask

Fig 3 b) Green Mask

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/25nov517

Fig 3 c) Blue Mask

Fig 4 a) Red Mask Component

Fig 4 b) Green Mask Image Component Image

Fig 4 c) Blue Mask Component

Fig 5 Segmented Image after Concatenating Masked Component Images

B. Fingertip Finding Algorithm

The fingertip finding algorithm, which emphasises the quantity of visible fingertips as the primary characteristic, is essential for reaching high ISL recognition efficiency. Once the hand has been segmented, the method finds the points of maximum outward protrusion using geometric principles and contour analysis. Although this method initially has problems, such as detecting incorrect fingertips at incorrect locations, which calls for additional correction steps, it successfully extracts a straightforward yet effective feature for classification.

Fig 6 Segmentation Results

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/25nov517

- *Step 1: Thinning Using Distance Transform*

Distance Transform (DT) is the first step in fine-tuning the hand shape. DT tracks the time it takes for a fire to burn inward by mapping the shortest distance between each hand pixel and the boundary. The segmented hand silhouette is successfully thinned down to a medial axis or skeleton that is only one pixel wide, simplifying the geometry while maintaining the hand's structure.

- *Step 2: Finding Perimeter Pixels of the Image*

The algorithm then determines the hand's skeleton's perimeter pixels after thinning. If a non-zero pixel touches one or more zero-valued (background) pixels directly, it forms the shape's perimeter.

- *Step 3: Finding Corner Points*

The first step in identifying possible fingertips is Perimeter Pre-filtering, which isolates points on the edge of the thinned image with an associated less value. This helps concentrate the detection on the hand features that are closest to the edge. Based on the idea that image intensity varies dramatically in all directions around a true corner, the Harris Corner Detector forms the basis of the procedure. Equation 2 illustrates how the Autocorrelation Function C(x, y ; Δx, Δy), which measures the intensity change when a window W(x, y) is shifted by (Δx, Δy), mathematically formulates this idea. Then, by excluding any points that are within 40 pixels of the image edge, a constraint is applied to stop spurious corners from being detected.

C. Recognition with PCA

Data Centering is the first step in the Principal Component Analysis (PCA) process. In this step, the training data matrix. $X(composed\ of\ d\ diensions\ and\ n\ samples)$, is normalized by deducting the mean from each dimension. The Covariance Matrix,C_x, which measures the correlations and differences among all pairs of dimensions, is then computed as $C_x = X \cdot X^T$. The eigenvalues (λ) stored on the main diagonal of the diagonal matrix C_y are obtained by solving the characteristic equation, $C_x V = V C_y$, using Eigen-Decomposition on the covariance matrix. These eigenvalues are important because they show how much variance is captured by the corresponding eigenvectors (λ).

The Feature Vector is produced after the Eigen-Decomposition by arranging the eigenvectors in decreasing order based on their corresponding eigenvalues (Equation 4). This is because the eigenvectors that are chosen capture the most variance, resulting in a smaller set of principal components. The Final Data is then obtained by multiplying the original data matrix, X^T, by this Feature Vector (Equation 5). The classification problem for the ISL recognition system is made simpler by this new, lower-dimensional dataset, which expresses the high-dimensional original data only in terms of the selected, most informative principal components.

- *Feature Vector*

$$= (\mathrm{Eigen}_1, \mathrm{Eigen}_2, \mathrm{Eigen}_3, \ldots, \mathrm{Eigen}_N))$$

$$\text{Final data} = X^T \cdot \text{Feature vector}$$

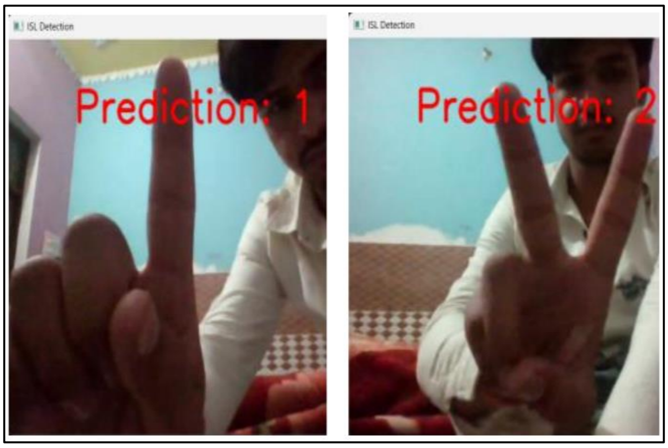

Fig 7 Recognized Signs

VII. RESULTS

Using color-based segmentation on red and blue gloves, this ISL Recognition framework showed a 94% recognition rate for live sign frames. Every 20th video frame is processed by the system, which uses PCA for classification. By introducing a fingertip finding algorithm to build a feature-based database, PCA was able to increase efficiency by comparing input images only with pertinent data subsets. Although the system works well for isolated signs, it has trouble recognizing signs that involve motion (dynamic signs) and hand overlap (occlusion).

VIII. CONCLUSION

This study introduced a framework for the recognition of Indian Sign Language (ISL) that uses Principal Component Analysis (PCA) for classification and color-based segmentation of red and blue gloves. With a sample taken every 20th frame, the system's recognition rate for live sign frames was 94%. By using a fingertip finding algorithm to build a feature-rich database, PCA was able to compare input images more effectively by counting the number of fingers that were extended. The intricacies of ISL, specifically the problems of hand overlap (occlusion) and dynamic signs (motion), pose a challenge to the accuracy of the system despite this successful feature integration, highlighting the need for further research on sophisticated temporal and spatial modelling.

REFERENCES

[1]. Box, G. E. P., Jenkins, G. M., Reinsel, G. C., & Ljung, G. M. *Time Series Analysis: Forecasting and Control*. Wiley, 2015.

[2]. Fan, S., & Hyndman, R. J. "Short-Term Load Forecasting Based on a Semi-Parametric Additive Model." *IEEE Transactions on Power Systems*, vol. 27, no. 1, pp. 134–141, 2012.

[3]. Hochreiter, S., & Schmidhuber, J. "Long Short-Term Memory." *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[4]. Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. "Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation." *arXiv preprint arXiv:1406.1078*, 2014.

[5]. Zhang, G., Eddy Patuwo, B., & Hu, M. Y. "Forecasting with Artificial Neural Networks: The State of the Art." *International Journal of Forecasting*, vol. 14, no. 1, pp. 35–62, 1998.

[6]. UCI Machine Learning Repository. "Individual Household Electric Power Consumption Dataset." University of California, Irvine. Available: https://archive.ics.uci.edu/ml/datasets/individual+household+electric+power+consumption.

[7]. Weron, R. "Electricity Price Forecasting: A Review of the State-of-the-Art with a Look into the Future." *International Journal of Forecasting*, vol. 30, no. 4, pp. 1030–1081, 2014.

[8]. Marino, D. L., Amarasinghe, K., & Manic, M. "Building Energy Load Forecasting Using Deep Neural Networks." *2016 International Joint Conference on Neural Networks (IJCNN)*, pp. 4394–4401.

[9]. Wang, Y., Chen, Q., Kang, C., Xia, Q., & Zhang, M. "Load Profiling and Its Application to Demand Response: A Review." *IEEE Transactions on Smart Grid*, vol. 10, no. 3, pp. 3125–3143, 2019.

[10]. Amasyali, K., & El-Gohary, N. M. "A Review of Data-Driven Building Energy Consumption Prediction Studies." *Renewable and Sustainable Energy Reviews*, vol. 81, pp. 1192–1205, 2018.

[11]. Smyl, S. "A Hybrid Method of Exponential Smoothing and Recurrent Neural Networks for Time Series Forecasting." *International Journal of Forecasting*, vol. 36, no. 1, pp. 75–85, 2020.

[12]. Kong, W., Dong, Z. Y., Jia, Y., Hill, D. J., Xu, Y., & Zhang, Y. "Short-Term Residential Load Forecasting Based on LSTM Recurrent Neural Network." *IEEE Transactions on Smart Grid*, vol. 10, no. 1, pp. 841–851, 2019.

[13]. Ahmad, T., & Chen, H. "A Review on Machine Learning Forecasting Growth Trends and Their Real-Time Applications in Different Energy Systems." *Sustainable Cities and Society*, vol. 54, p. 102010, 2020.

[14]. Li, C., Ding, Z., Zhao, D., Yi, J., & Zhang, G. "Building Energy Consumption Prediction: An Extreme Deep Learning Approach." *Energies*, vol. 10, no. 10, p. 1525, 2017.

[15]. Mocanu, E., Nguyen, P. H., Gibescu, M., & Kling, W. L. "Deep Learning for Estimating Building Energy Consumption." *Sustainable Energy, Grids and Networks*, vol. 6, pp. 91–99, 2016.