

A Hybrid CNN–MLP Model Using Confidence-Gated Fusion Framework for Real-Time Indian Sign Language Recognition

Youddha Beer Singh

youddhabeersingh@gmail.com

Galgotias College of Engineering & Technology

Khyatee Kaushal

Galgotias College of Engineering & Technology

Pratyush Singh

Galgotias College of Engineering & Technology

Research Article

Keywords: Convolutional Neural Network, Hand Gesture Recognition, Indian Sign Language, MediaPipe, Multi-Layer Perceptron, Multimodal Fusion, Temporal Stabilization

Posted Date: October 22nd, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-10537594/v1>

License:   This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Additional Declarations: No competing interests reported.

Abstract

Indian Sign Language (ISL) recognition play a vital role for allowing communication between hearing-impaired people and general public. However, current methods are still suffering with the decreasing performance under the ambiguous or noisy inputs and inconsistent multimodal fusion. This paper proposes a hybrid deep learning based architecture to recognize Indian Sign Language (ISL). The model integrates visual data from images and geometric data from hand landmarks. The image features are extracted using a MobileNetV2-based convolutional neural network while the landmark data are classified by a multilayer perceptron (MLP) based on data generated by MediaPipe. While CNN work well in controlled environments, they have difficulty with changes to lighting or background. Conversely, landmark-based models are unable to capture fine visual details. To resolve these limitations, this research proposes a combined model that uses both CNN and landmark-based methods by using a confidence-based fusion method to select predictions during real time operation based on how reliable each model is. Results demonstrate that the proposed hybrid model produces reliable and stable performance within the context of real-time applications achieving an accuracy of 95% in testing and a confidence of ~ 1.00 in real-time which makes an excellent choice for use in assistive communication applications.

1. Introduction

Individuals who are hard of hearing or have a speech disability rely on sign language to communicate, but there is currently no practical way for sign language users to communicate with people who do not use sign language due to the absence of universally adopted translation systems for sign language. As a result, there is a growing interest in developing automated systems that can identify hand gestures and converts them into meaningful spoken or written words, thus facilitating greater access to public spaces and services [29]. Progress in deep learning have enabled the gesture recognition field to develop automatic models that learn high-level representations of hand gestures from visual data. The adoption of Convolutional Neural Networks (CNN) and other deep learning methods have gained popularity because these methods have the capacity to learn and retrieve relevant, multi-scale features from images [1, 2]. Lightweight architectures such as MobileNet and MobileNetV2 have been proposed to provide real-time processing, while preserving accuracy by decreasing computational load in a gesture recognition application [3, 4]. Gesture recognition systems built using MobileNet or MobileNetV2 can accurately recognize hand shape and the spatial relationship between the hand and the target.

In addition to visual techniques of tracking hands, new landmark techniques are being developed with the creation of effective hand-tracking methods like MediaPipe, which allow for real-time extraction of hand keypoints [8]. The use of these landmarks helps to provide for the representation of gestures geometrically, in a structured format, and to be processed /recognized via lightweight models e.g., Multi-Layer Perceptrons (MLPs). Computational efficiency of such methods may be far more than other approaches; however, they do not have an ample amount of visual detail [19–22]. Traditional CNN-based

systems are able to recognize gestures; however, they also may be highly sensitive to changes in the external environment including lighting, background and hand position [23, 24].

Current research has inspired new directions toward efficient and lightweight methodologies for sign language recognition. The development of recent studies have encouraged the exploration of hybrid and multimodal learning schemes, which will use the combined characteristics of their respective approaches to increase recognition performance [26, 28]. The purpose of this paper is to investigate the creation of a hybrid system that utilizes the combination of visual data created from CNNs and landmark data created from MLPs. The introduction of a confidence-gated fusion mechanism to combine both systems' output will allow adaptive weighting to be placed depending on the reliability of the methods used in recognizing the gestures. Any potential misuse of this technique can be reduced by having an additional temporal stabilization term in the final output, thereby providing for a more consistent recognition process from frame to frame and reducing the variance in the recognition results. The methodology developed in this study will provide for the effective real-time recognition of ISL (Indian Sign Language) gestural signs and has been tested on a multi-class database and demonstrates both increased robustness and stability with the new hybrid approach as compared to recognition achieved with only one technique. As demonstrated by the experimental results, each of the methodologies provides good performance and thus achieves accurate and stable gesture recognition for all tested conditions.

The main contributions of this research are:

- **Hybrid Architecture:** MLP and CNN are combined to produce a parallel combination of visual and geometric data for improved prediction accuracy from each or all data types.
- **Robust Real Time System:** The system was designed to function accurately under conditions varying from lighting to background difference.
- **Confidence Gated Fusion:** Approval of results from the model is governed dynamically based solely on how confident an input image produces results in order to provide increased overall accuracy.
- **Temporal Smoothing:** Reduced, smoothed or stabilized differences between consecutive frame predictions from the model in order to achieve consistency for real-time recognition.

The remaining section of the paper is organized as follows: The past researches of the work are discussed briefly in section 2; section 3 describes the detailed proposed methodology including dataset gathering and preprocessing to model creation. Section 4 consists of the implementation details the results of which are discussed in section 5. The last section concludes the paper which is followed by the references.

2. Literature Review

Deep learning methods are becoming the foundation for modern sign language and hand gesture recognition systems, with the introduction of many new architectures designed to improve speed and

efficiency. New lightweight architectures such as ShuffleNet and EfficientNet are being developed to improve computational costs with performance [5, 6]. Transformer architectures are being applied to visual recognition tasks, which further demonstrate the ability to model global relationships in image data [7]. Many datasets and benchmarks have also been created for large-scale training and evaluation of sign language recognition systems. For example, the MS-ASL dataset is a large-scale benchmark for gesture understanding in natural environments [9]. These advances have contributed to improved generalization and performance across many gesture recognition tasks. The use of real-time hand-tracking frameworks has also led to a resurgence of interest in landmark-based systems. MediaPipe has become one of the most popular ways to extract the key points of a hand so they can be fed into machine learning-based gesture classifiers [10, 11, 12]. There has also been research published on the use of landmark-based approaches with other sign languages, showing the flexibility of the landmark-based approach [13]. However, most of these systems continue to rely heavily on geometric features as well as they do not necessarily represent many of the intricate visual differences found in hand gestures.

Recent advancements in combining spatial and temporal modelling techniques have been developed to tackle these challenges. Transformer-based and hybrid CNN–LSTM architectures have demonstrated improved performance for dynamic gesture recognition [14, 15, 16, 17, 18]; therefore, they exist to extract spatial characteristics and temporal dependencies through modelling both spatial characteristics and temporal dependencies for more accurate recognition of complicated gestures. However, the use of these advanced architectures typically results in increased computational complexity, thus prohibiting the ability to perform in real time. Furthermore, multimodal learning approaches have been applied to improve robustness through combining multiple types of feature representation. The combination of visual, geometric and temporal information has demonstrated robust and stable recognition performance [25, 27, 28–29]; however most of the implementations available to date have either seen considerable complexity in their underlying architectures or do not have efficient fusion mechanisms for real time applications.

Thus, there continues to be an opportunity to develop a lightweight and robust hybrid framework for combining complementary feature representation with the ability to achieve real-time performance [31]. The current proposal offers an integration of visual and landmark feature representation using a confidence-guided fusion mechanism to enhance the robustness and consistency in predictions.

A comparative summary of closely related works is presented in Table 1

Table 1
Comparative Analysis of Existing Methods

Ref	SL	Technique Used	Dataset(s)	Sign Identification	Accuracy
[10]	ISL	MediaPipe + CNN	Custom dataset	Gesture classification	99.95%
[11]	ISL	MediaPipe + CNN	Custom dataset	Similar gesture detection	99.12%
[12]	ISL	MediaPipe + CNN	ISL dataset	Alphabet recognition	94%
[15]	ISL	KD-MSLRT	Benchmark dataset	Gesture recognition	91%
[16]	TSL	TSLFormer	Landmark dataset	Sign classification	90.67%
[17]	ISL	CNN + LSTM	Video dataset	Dynamic gestures	92%

The Table 1 illustrates that although cnn and mlp work well under controlled conditions, they lack robustness in realtime environments. These limitations highlight the need for a hybrid framework that effectively integrates visual and geometric features while ensuring stable real-time predictions.

3. Methodology

This section presents the architectural design, acquisition of datasets, and implementation of the proposed hybrid deeel learning based model for real time Indian sign language recognition. The fundamental chalange

The model integrates visual data from images and geometric data from hand landmarks.

The image features are extracted using a MobileNetV2-based convolutional neural network while the landmark data are classified by a multilayer perceptron (MLP) based on data generated by MediaPipe. While CNN work well in controlled environments, they have difficulty with changes to lighting or background. Conversely, landmark-based models are unable to capture fine visual details.

3.1. Dataset Acquisition

The current research leverages publicly available data, which is meant for sign language gesture recognition. The source of the data is an open-source GitHub repository [30]. The dataset contains 29 classes of static images of hand gestures, each representing an alphabetic symbol. It also includes three other classes: del, space, nothing, in addition to the 26 alphabet classes from 'A' to 'Z'. The dataset comprises of 87,000 images in total, with each of the 29 classes having approximately 3,000 images. For the purpose of developing the model, the dataset is divided into three subsets: training, validation, each classes is kept in a separate folder to avoid any overlapping data. The description of the dataset is summarized in Table 2.

Table 2
Dataset Description

Parameter	Value
Total Classes	29
Alphabets	26 (A–Z)
Additional Classes	del, space, nothing
Images per Class	3000
Total Images	~87,000
Dataset Split	Train (80%), Validation (10%), Test (10%)

3.2. Data Preprocessing

The image-based model in this work is built using the MobileNetV2 architecture [4]. Before training, all input images were resized to 224x224 pixels to match the expected input dimensions of the network. To improve numerical stability during training, pixel values were scaled to the range [0, 1]. This normalization step helps in faster convergence of the optimization process and reduces the impact of large intensity variations. The transformation is defined as in Eq. 1:

$$x_{norm} = \frac{x}{255}$$

1

where x denotes the original pixel value and x_{norm} represents the normalized output.

In addition to image preprocessing, a landmark-based representation was also generated using the MediaPipe Hands framework (b8?). This framework extracts 21 key points from each hand, with each point represented using three-dimensional coordinates.

$$L = \{ (x_1, y_1, z_1), (x_2, y_2, z_2), \dots, (x_{21}, y_{21}, z_{21}) \}$$

2

where L denotes the set of all 21 hand landmarks, and (x_i, y_i, z_i) denotes the three-dimensional coordinates of the i -th hand landmark, with $i = 1, 2, \dots, 21$. These points collectively form a 63-dimensional feature vector for each gesture.

To ensure consistency across samples, the landmark coordinates were first translated with respect to the wrist joint:

$$(x'_i, y'_i, z'_i) = (x_i - x_w, y_i - y_w, z_i - z_w)$$

3

where (x'_i, y'_i, z'_i) denotes the converted coordinates of the i -th landmark relative to the wrist, and (x_w, y_w, z_w) represents the coordinates of the wrist joint.

Next, scaling was applied to the dataset minimize the variations caused by the distance between hand size and camera which is defined as in Eq. 4:

$$(x''_i, y''_i, z''_i) = \frac{(x'_i, y'_i, z'_i)}{s}$$

4

where (x''_i, y''_i, z''_i) signifies the scaled coordinates of the i -th landmark, and s represents the scaling factor used for normalization.

Finally, a mirroring operation was used to standardize left-hand and right-hand representations in Eq. 5:

$$x_{mirror} = -x$$

5

where x_{mirror} denotes the mirrored x-coordinate, obtained by negating the original coordinate x to standardize hand orientation.

3.3. Model Creation

3.3.1. Convolutional Neural Network Model (MobileNetV2)

An CNN model was built using transfer learning through MobileNetV2 [4] for image-based classification, which is computationally efficient for real-time purposes.

Transfer learning was used by initializing the MobileNetV2 model with ImageNet pre-trained weights. This enabled the model to leverage knowledge that had been learned in previous tasks on features that were visually discriminative, such as edges, textures, and spatial configurations.

For the gesture classification problem, the original classification layers were replaced. The architecture of the model is defined by the following layers. The first layer is a Global Average Pooling layer which is then followed by a Dense fully connected layer with 256 neurons and ReLU activation. Thereafter a Dropout layer with a dropout rate of 0.3 is used for regularization. The final layer uses Softmax activation with 29 nodes.

Equation 6 define the Softmax function used for classification as given below:

$$P(y_i) = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}}$$

6

where $P(y_i)$ denotes the predicted probability that the input belongs to class i and z_i represents the score for class i and K is the number of classes.

The structure of the CNN model used is illustrated in the Fig. 1.

3.3.2. Landmark-Based MLP Model

For geometric feature classification, a Multi-Layer Perceptron (MLP) network was used. This model works with numerical feature vectors, which are generated based on hand landmark information.

The input to the network is a 63-dimensional vector, which is generated based on the coordinates of the 21 detected landmarks. There are multiple fully connected layers in this network with non-linear activation functions. This helps in understanding the relationship between the positions of different landmarks.

This method focuses on structural hand information, which reduces its dependency on lighting conditions. Regularization is also used in this network.

The architecture of the MLP model is shown in Fig. 2.

3.3.3. Proposed Hybrid Model

To incorporate both visual and geometric features, a hybrid architecture was proposed. The input image goes through a process in two parallel branches.

In the first branch, the input image is passed through a convolutional neural network (CNN), which extracts features from the image and generates predictions. In the second branch, hand landmarks are detected using a library called Mediapipe [8] and then processed through a multilayer perceptron (MLP).

The predictions from both networks are combined through a fusion mechanism to arrive at a final prediction. This allows the system to utilize both types of information, making it more reliable and accurate even in different environmental conditions.

The overall hybrid framework is illustrated in Fig. 3.

The proposed hybrid CNN–MLP based architecture (Fig. 3) combines visual appearance features with hand landmark-based geometric features to achieve robustness and accurate ISL recognition. By integrating fusion mechanism (confidence-gated fusion) and temporal stabilization, the proposed model

improves the accuracy, and reduces misclassification, and enables stable real-time ISL recognition for the human computer interaction-based applications.

4. Implementation

4.1. Training Environment

The implementation of the proposed gesture recognition model was carried out using a combination of cloud-based and local development environments. The CNN model was trained on a GPU-enabled cloud platform (RunPod), with an NVIDIA RTX A5000 GPU, which provided the necessary computational capability for training deep convolutional networks while the landmark-based MLP model was trained using Google Colab, and the complete hybrid system was integrated and executed locally using Visual Studio Code for real-time inference. The system was developed utilizing Python programming language for all aspects of the project. Model training was accomplished using TensorFlow/Keras as the development platform, and image acquisition and pre-processing was achieved using OpenCV while real-time detection of hand landmarks was accomplished with MediaPipe [8]. The dataset was loaded into the framework using a directory-based approach. Data preprocessing and augmentation were carried out using the ImageDataGenerator utility. Image normalization and augmentation were applied to the input images. Data augmentation techniques include rotation, zoom, and spatial transformations. Such techniques improve data diversity, which in turn enables better generalization of the CNN model.

4.2. Model parameters

Two models, namely a CNN and an MLP, are trained independently and their outputs are fused using confidence gated strategy to form the hybrid model. The main configuration details of the proposed models are listed in Table 3.

Table 3
Model Architecture Parameters

Component	CNN Model (MobileNetV2)	MLP Model (MediaPipe)
Input Type	Image ($224 \times 224 \times 3$)	Landmark vector (63 features)
Feature Extraction	MobileNetV2 (ImageNet pretrained)	MediaPipe hand landmarks (21 keypoints)
Preprocessing	Data augmentation (rotation, zoom, shift)	Landmark extraction from images
Pooling Layer	Global Average Pooling	—
Hidden Layers	Dense (256, ReLU)	Fully connected layers (MLP)
Dropout	0.3	0.3
Output Layer	29 classes (Softmax)	29 classes (Softmax)
Training Strategy	Transfer learning and fine-tuning (last 40 layers)	Supervised training on landmark dataset
Optimizer	Adam ($1e-3$ to $1e-5$)	Adam (standard)

4.3. Proposed Hybrid Approach

The proposed hybrid model integrates the appearance-based features and landmark-based learning to achieve better accuracy and robust for real-time ISL recognition. Initially, each captured frame from the webcam is processed to extract the hand region of interest (ROI). The extracted ROI is simultaneously fed into two parallel Models, one is a Convolutional Neural Network (CNN) Model based on MobileNetV2 for visual feature extraction and second one is MLP based model that utilizes the MediaPipe hand landmarks to capture the geometric features. The output generated by both the models are combined using a fusion mechanism i.e. confidence-gated fusion strategy, where the CNN prediction is selected only when its confidence exceeds a predefined threshold; otherwise, the MLP prediction is used. Finally, a temporal stabilization module employing a sliding window and majority voting is applied to suppress prediction variation and improve the recognition stability. The recognized gesture is then displayed on the screen and converted into speech output. The complete workflow of the proposed architecture is described in Algorithm 1.

Algorithm 1

Hybrid CNN–MLP Architecture for real-time ISL/hand gesture recognition. Each captured frame through webcam are processed through parallel CNN and MLP models. A confidence-gated fusion strategy selects for the final prediction, and refined using temporal majority voting before being displayed and converted into speech.

Input : Webcam video stream V

Output: Recognized gesture G

```
1: Initialize MobileNetV2, MLP, MediaPipe, and webcam
2: Initialize sliding window W
3: while webcam is active do
4:   Capture frame F
5:   Extract hand ROI from F
6:   Resize ROI to 224 × 224
7:   CNN ← MobileNetV2(ROI)
8:   Detect 21 hand landmarks using MediaPipe
9:   if landmarks detected then
10:    X ← Generate 63-dimensional feature vector
11:    MLP ← Predict(X)
12:   else
13:     continue
14:   end if
15:   if CNN.confidence ≥ T then
16:    P ← CNN.prediction
17:   else
18:    P ← MLP.prediction
19:   end if
20:   Insert P into sliding window W
21:   G ← MajorityVote(W)
22:   Display G
23:   Generate speech output
24: end while
25: Release resources
```

During training, a multi-step approach was used. To begin with, the pre-trained MobileNetV2 model [4] was used. The model was frozen, with only updates being applied to the newly added classification layers. This process was carried out for 5 epochs using the Adam optimizer (31) with a learning rate of

1×10^{-3} . Training was continued for 5 more epochs with validation monitoring. This approach was used for 10 epochs in total. The model with the best validation performance was retained for further refinement.

The next step in training involved fine-tuning. To do this, the last 40 layers of the model were made trainable. The model was then trained for 5 epochs with a reduced learning rate of 1×10^{-5} . This approach enables fine-tuning of feature representations in alignment with the gesture recognition problem.

The model was trained using the categorical cross-entropy loss function, which is described in Eq. 7.

$$L = - \sum_{i=1}^K y_i \log(\hat{y}_i)$$

7

where y_i denotes the ground truth label, $\hat{y}_i$ represents the predicted probability for class i , and K is the total number of classes.

This training methodology allows the model to first learn task-specific features in the classification level and then proceed to refine the deeper layers. This improves generalization and reduces the chances of overfitting. In addition, validation-based checkpointing is used to ensure the best-performing model is retained. Also, having separate sets for training, validation, and testing prevents data leakage and ensures proper testing.

The MLP model is trained separately using the 63-dimensional landmark feature vectors. These vectors represent the spatial relationships of keypoints in the hand and are a compact representation of gesture structure for classification.

For real-time implementation of the proposed hybrid model, video frames are captured through a webcam. These frames are then passed through two parallel streams. In the first stream, frames are passed through the CNN model for predictions using visual features. In the second stream, the frames are passed through the MediaPipe framework [8] to obtain hand landmark keypoints. These keypoints are then converted into vectors and passed through the MLP model.

The predictions of both models are then combined using a confidence-based mechanism to arrive at the gesture prediction. Confidence Gating Fusion can be defined as one of the decision level fusion methods that involves making a choice between the outputs of models based on higher confidence levels, using a predefined threshold parameter. This is done to utilize the complementary nature of visual and geometric representations of gestures, which improves robustness during real-time prediction. Let y_{CNN} and y_{MLP} denote the predicted class labels from the CNN and MLP models, respectively, and let C_{CNN} represent the confidence score associated with the CNN prediction. The final prediction is determined as in Eq. 8:

$$y = \begin{cases} y_{CNN}, & \text{if } C_{CNN} > \theta \\ y_{MLP}, & \text{otherwise} \end{cases}$$

8

where θ represents a predefined confidence threshold. In this framework, the MLP model is prioritized due to its robustness to variations in lighting conditions and background noise, while the CNN model serves as a complementary component that provides visual context when landmark information is unavailable.

To further enhance prediction stability in real-time scenarios, a temporal stabilization mechanism was incorporated. Instead of relying on predictions from a single frame, the system maintains a sliding window of recent predictions. A gesture is considered valid only when it appears consistently across multiple frames within the window. This can be expressed as in Eq. 9:

$$\hat{y} = \text{mode}(y_1, y_2, \dots, y_n)$$

9

where $y_1, y_2, \dots, y_n$ represent predictions over consecutive frames. This approach reduces prediction fluctuations and improves the reliability of the final output in continuous gesture recognition.

5. Experimental Results and Performance comparison

CNN and MLP and hybrid model were implemented and tested on the

5.1. CNN and MLP Model Performance

The plots of accuracy and loss curves of both the models are shown in Fig. 4.

The CNN model demonstrated exceptionally accurate classification performance, achieving an overall accuracy approximately 99.98% on the test dataset. The training and validation accuracy curves as shown in Fig. 4(a) show rapid convergence, while the corresponding loss curves as shown in Fig. 4(b) indicate stable optimization with no significant divergence between training and validation loss. This reflects the model's strong capability in learning distinct visual features from input images.

The high performance of the CNN model can be attributed to the use of transfer learning with MobileNetV2, balanced dataset distribution, and relatively controlled imaging conditions. These factors collectively enable effective feature extraction and classification across all gesture classes.

The MLP model, trained on hand landmark features, achieved an overall accuracy of approximately 97.1%. The training and validation curves Fig. 4(c) and (d) show consistent learning behavior, although the performance is slightly lower as compared to the CNN model, this is expected, as the MLP relies

solely on geometric features derived from hand landmarks, which may not fully capture ultra-fine visual variations.

The confusion matrices of both models as shown in Fig. 5(a) and Fig. 5(b) further highlight their performance characteristics.

The CNN model exhibits highly accurate predictions across all gesture classes, whereas the MLP model shows minor confusion among gestures with similar hand configurations, particularly for classes such as M, N, R, and U. Despite this, the MLP model demonstrates strong robustness to variations in lighting and background conditions due to its reliance on structural features.

5.2. Hybrid Model Performance

The hybrid model integrates predictions from both the CNN and MLP models using a confidence-gated fusion strategy. The confusion matrix of the hybrid model Fig. 6 shows strong diagonal dominance, indicating accurate classification across the majority of gesture classes achieving an overall accuracy of approximately 95%, with macro-averaged precision, recall, and F1-score values of approximately 0.96, 0.95, and 0.95, respectively.

While the numerical accuracy is slightly lower than that of the CNN model, the hybrid approach provides improved robustness and consistency in challenging conditions. Evidently, the hybrid model reduces misclassifications observed in the MLP model while maintaining stable predictions under varying environmental conditions. This demonstrates the effectiveness of combining visual and landmark-based features for gesture recognition.

This demonstrates that the proposed hybrid framework prioritizes robustness and real-time reliability over peak accuracy, making it more suitable for practical deployment.

5.3. Classification Performance Comparison

The classification performance of the three models on validation and test dataset is summarized in Fig. 7.

The results indicate that the CNN model achieves the highest accuracy due to its ability to capture detailed visual features. The MLP model, while slightly less accurate, provides stable performance based on geometric hand representations. The hybrid model balances these strengths, offering a more robust and reliable solution.

5.4. Comparative Analysis in Training and Real-Time Scenarios

To further analyze the effectiveness of the proposed approach, a comparison was conducted considering both training performance and real-time applicability. Figure 8 shows that the comparison

between validation accuracy with real time accuracy which shows that the proposed hybrid model gives better result than the CNN and MLP models.

The overall classification performance of the three models is summarized in Table 4.

Table 4
Training and Real-Time Performance Comparison

Model	Validation/Test Accuracy	Real-Time Confidence	Robustness
CNN	99.8%	0.70	Sensitive to environment
MLP	97.1%	0.98	Robust to variations
Hybrid Model	95%	1.00	Highly robust

The CNN model, in particular, was able to display exceptional performance within a controlled environment, though it was likely to degrade in the presence of illumination and background changes in real-time scenarios. The MLP model, on the other hand, provided better stability in such scenarios, though it was unable to display finer visual distinctions. As such, the hybrid model was able to effectively incorporate the desirable characteristics of both models, thus ensuring better stability in prediction.

Apart from the accuracy of classification, the prediction confidence of the models was also monitored in real-time scenarios. The CNN model was able to display a confidence level between 0.6 and 0.7, thus displaying significant levels of uncertainty in real-time scenarios due to changes in environmental factors. However, the MLP model was able to display higher levels of confidence, with most scenarios having confidence levels higher than 0.98 due to the reliance on landmark features. However, the model was unable to display accurate predictions for some of the alphabets due to their similarities in gesture. The hybrid model, on the other hand, was able to display consistent confidence levels of ~ 1.00 or 100% for almost all alphabets, thus displaying better reliability in real-time scenarios compared to the CNN model. The real time results of hybrid model are shown in Fig. 9.

The use of confidence-gated fusion, coupled with temporal stabilization, also enhances the reliability of gesture recognition in real-time scenarios. The hybrid model, in particular, was able to display better robustness in real-time scenarios compared to the accuracy of classification, thus making it a better solution for real-time scenarios.

5.5 Comparison with the Existing Approaches

The performance of the proposed hybrid model is compared with existing state-of-the-art sign language recognition approaches reported in recent literature, is presented in Table 5.

Table 5
Comparison of Various Sign Language Recognition Approaches

Ref / Year	Technique Used	Dataset Description	Accuracy	Signs Identified	Output
[12]/2023	MediaPipe + CNN	ISL dataset using real-time webcam input for gesture recognition.	94%	ISL gestures (alphabets + translation)	Text + Speech
[15]/2026	MediaPipe + CNN (Hybrid)	Custom dataset with 24 classes. ~120k images including static and dynamic gestures.	91%	Alphabets + dynamic gestures	Text
[17]/2025	CNN + LSTM (MediaPipe-based)	Landmark-based dataset using video sequences for temporal gestures.	92%	Dynamic gestures	Text
Proposed Model	CNN + MLP (Fusion)	ASL dataset with 29 classes. ~87k images used.	95%	Alphabets (A–Z)	Text

As observed from Table 5, existing approaches such as MediaPipe-based CNN models achieve high accuracy under controlled conditions but often lack robust-ness in real-time environments. Hybrid approaches incorporating emporal modeling improve dynamic gesture recognition but introduce additional complexity. The pro-posed CNN–MLP hybrid model achieves competitive accuracy while offering improved stability and reliability in real-time scenarios through confidence-based fusion and temporal smoothing.

6. Conclusion

In this work, we have developed a hybrid approach for recognizing gestures of Indian Sign Language (ISL) in real-time by combining two models: one that uses CNN-based visual features and another that uses geometric features extracted from hand landmarks using MLPs. The overall goal of this work is to leverage appearance-based feature representations and geometric feature representations in order to achieve greater reliability in gesture recognition systems in real time. The performance of the CNN model for recognizing gestures by extracting detailed visual features from images (that is, its ability to learn detailed visual patterns from images) was demonstrated to be very strong, while the MLP model demonstrated great robustness in recognizing gestures by extracting hand landmark features (that is, by extracting geometric location-based features) because these types of features are relatively less affected by environmental uncertainties. By applying a confidence-based on approach to both output from both models, the proposed gesture recognition system is capable of providing more dependable predictions than either individual model would provide. The results of this work indicate that while the CNN model was able to provide extremely high accuracy levels for gesture recognition when tested under controlled experimental conditions, the use of both the CNN and MLP models provides

significantly greater stability and consistency (and therefore increases reliability) for gesture recognition in real-time applications. Further, the application of additional stabilizing algorithms to provide temporal stabilization reduces variations in the predictions provided by the system over successive frames. To conclude, this research shows that by combining multiple methods of gesture recognition through the use of various methods for representing feature data, it is possible to create a much more equally distributed and functional gesture recognition method than through using only one method of gesture recognition, thereby making the hybrid model developed herein suitable for identifying ISL gestures in real-time applications.

Declarations

This research does not receive any specific funding, and the authors declare that there is no conflict of interests. This approval, consent to participate, consent for publication, and materials availability were not applicable. The dataset has been taken from a publicly available GitHub repository [30], and the code is available from the corresponding author on the request. All authors contributed equally to this work and approved the final manuscript.

Ethics, Consent to Participant, and Consent to Publish

This research does not include human or animal participation, or clinical data requiring ethical approval so Ethics, Consent to Participant, and Consent to Publish not applicable.

Funding

Declaration: This research work does not receive any specific funding.

Author Contribution

Proposed hybrid approach and manuscript writing is done by Youddha Beer Singh and Pratyush Singh and implementation section done by Khyatee Kaushal. All authors contributed equally to this work and approved the final manuscript.

Data Availability Statement:

This study uses a publicly available dataset for Real Time Indian Sign Language gesture recognition. The dataset was obtained from the open-source GitHub repository cited in Reference [30] and is available at: <https://github.com/sudiptasarkar011/sign-bridge>. No additional or secondary datasets were used in this study. The proposed model was trained and evaluated using this publicly available dataset, along with real-time Indian Sign Language recognition experiments conducted to validate its practical applicability.

References

1. He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770–778).
2. Simonyan, K., & Zisserman, A. (2015, April). Very deep convolutional networks for large-scale image recognition. In *3rd international conference on learning representations (ICLR 2015)*. Computational and Biological Learning Society. <https://doi.org/10.48550/arXiv.1409.1556>
3. Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., & Adam, H. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*. <https://doi.org/10.48550/arXiv.1704.04861>
4. Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L. C. (2018). Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4510–4520). <https://doi.org/10.48550/arXiv.1801.04381>
5. Zhang, X., Zhou, X., Lin, M., & Sun, J. (2018). Shufflenet: An extremely efficient convolutional neural network for mobile devices. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 6848–6856). <https://doi.org/10.48550/arXiv.1707.01083>
6. Tan, M., & Le, Q. (2019, May). Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning* (pp. 6105–6114). PMLR. <https://proceedings.mlr.press/v97/tan19a.htm>
7. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., ... Houlsby, N. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*. <https://doi.org/10.48550/arXiv.2010.11929>.
8. Zhang, F., Bazarevsky, V., Vakunov, A., Tkachenka, A., Sung, G., Chang, C. L., & Grundmann, M. (2020). Mediapipe hands: On-device real-time hand tracking. *arXiv preprint arXiv:2006.10214*. <https://doi.org/10.48550/arXiv.2006.10214>
9. Joze, H. R. V., & Koller, O. (2018). Ms-asl: A large-scale data set and benchmark for understanding american sign language. *arXiv preprint arXiv:1812.01053*. <https://doi.org/10.48550/arXiv.1812.01053>
10. Kumar, R., Bajpai, A., & Sinha, A. (2023). Mediapipe and cnns for real-time asl gesture recognition. *arXiv preprint arXiv:2305.05296*. <https://doi.org/10.48550/arXiv.2305.05296>
11. Verma, A. R., Singh, G., Meghwal, K., Ramji, B., & Dadheech, P. K. (2024). Enhancing sign language detection through mediapipe and convolutional neural networks (CNN). *arXiv preprint arXiv:2406.03729*. <https://doi.org/10.48550/arXiv.2406.03729>
12. Deshpande, S., & Shettar, R. (2023, November). Hand gesture recognition using mediapipe and cnn for indian sign language and conversion to speech format for indian regional languages. In *2023 7th International Conference on Computation System and Information Technology for Sustainable Solutions (CSITSS)* (pp. 1–7). IEEE. <https://doi.org/10.1109/CSITSS60515.2023.10334218>
13. Pilare, L. K. M., Mahinay, J. C. A., Degamo, A. C. C., & Piner, B. N. C. (2024). Filipino Sign Language hand gesture recognition using MediaPipe and machine learning. *International Journal of Computing*

14. Li, H. H., & Hsieh, C. C. (2025). Dynamic hand gesture recognition using MediaPipe and Transformer. *Engineering Proceedings*, 108(1), 22. <https://doi.org/10.3390/engproc2025108022>
15. Li, Y., Ren, B., Hu, K., Liu, C., Jiang, Z., Dang, K., & Su, J. (2025, April). KD-MSLRT: Lightweight sign language recognition model based on mediapipe and 3d to 1d knowledge distillation. In *Proceedings of the AAAI Conference on Artificial Intelligence* (Vol. 39, No. 27, pp. 28177–28185). <https://doi.org/10.1609/aaai.v39i27.35037>
16. Ertürk, K., Altınışık, F., Sarıaltın, İ., & Gerek, Ö. N. (2025). Tslformer: A lightweight transformer model for turkish sign language recognition using skeletal landmarks. *arXiv preprint arXiv:2505.07890*. <https://doi.org/10.48550/arXiv.2505.07890>
17. Takouchouang, F. S. (2025). Automatic Sign Language Recognition: A Hybrid CNN-LSTM Approach Based on Mediapipe. <https://doi.org/10.48550/arXiv.2510.22011>
18. Rehman, M. U., Ahmed, F., Khan, M. A., Tariq, U., Alfouzan, F. A., Alzahrani, N. M., & Ahmad, J. (2022). Dynamic hand gesture recognition using 3D-CNN and LSTM networks. *Computers Materials & Continua*, 70(3), 4675. <https://doi.org/10.32604/cmc.2022.019586>
19. Abd Al-Latief, S. T., Yussof, S., Ahmad, A., & Khadim, S. (2024). Deep Learning for Sign Language Recognition: A Comparative Review. *Journal of Smart Internet of Things*, 2024(1), 77–116. <https://doi.org/10.2478/jsiot-2024-0006>
20. Simion, G., Gui, V., & Ottesteanu, M. (2012). Vision based hand gesture recognition: A review. *international journal of circuits systems and signal processing*, 6(4), 275–282.
21. Wang, T., Li, Y., Hu, J., Khan, A., Liu, L., Li, C., & Ran, M. (2018, August). A survey on vision-based hand gesture recognition. In *International Conference on Smart Multimedia* (pp. 219–231). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-030-04375-9_19
22. Franslin, N. M. F., & Ng, G. W. (2022, March). Vision-based dynamic hand gesture recognition techniques and applications: A review. In *Proceedings of the 8th International Conference on Computational Science and Technology: ICCST 2021, Labuan, Malaysia, 28–29 August* (pp. 125–138). Singapore: Springer Singapore. https://doi.org/10.1007/978-981-16-8515-6_11
23. Ahmad, A., Migniot, C., & Dipanda, A. (2019). Hand pose estimation and tracking in real and virtual interaction: A review. *Image and Vision Computing*, 89, 35–49. <https://doi.org/10.1016/j.imavis.2019.06.003>
24. Mahdikhanelou, K., & Ebrahimnezhad, H. (2023). 3D hand pose estimation from a single RGB image by weighting the occlusion and classification. *Pattern Recognition*, 136, 109217. <https://doi.org/10.1016/j.patcog.2022.109217>
25. Ren, B., Liu, M., Ding, R., & Liu, H. (2024). A survey on 3d skeleton-based action recognition using learning method. *Cyborg and Bionic Systems*, 5, 0100. <https://doi.org/10.34133/cbsystems.0100>
26. Jabeen, S., Li, X., Amin, M. S., Bourahla, O., Li, S., & Jabbar, A. (2023). A review on methods and applications in multimodal deep learning. *ACM Transactions on Multimedia Computing Communications and Applications*, 19(2s), 1–41. <https://doi.org/10.1145/3545572>

27. Baltrušaitis, T., Ahuja, C., & Morency, L. P. (2018). Multimodal machine learning: A survey and taxonomy. *IEEE transactions on pattern analysis and machine intelligence*, 41(2), 423–443. <https://doi.org/10.1109/TPAMI.2018.2798607>

28. Jiao, T., Guo, C., Feng, X., Chen, Y., & Song, J. (2024). A comprehensive survey on deep learning multi-modal fusion: Methods, technologies and applications. *Computers Materials & Continua*, 80(1), 1–35. <https://doi.org/10.32604/cmc.2024.053204>

29. Ingoley, S., & Bakal, J. (2025). Interpretation of Indian sign language to text and speech to communicate with speech and hearing impaired community. *Procedia Computer Science*, 258, 1980–1992. <https://doi.org/10.1016/j.procs.2025.04.449>

30. Sarkar, S. Sign-Bridge: Sign language recognition dataset and system, GitHub repository. [Online]. Available: <https://github.com/sudiptasarkar011/sign-bridge>

31. Kinga, D., & Adam, J. B. (2015, May). A method for stochastic optimization. In *International conference on learning representations (ICLR)* (Vol. 5, No. 6). <https://doi.org/10.48550/arXiv.1412.6980>

Figures

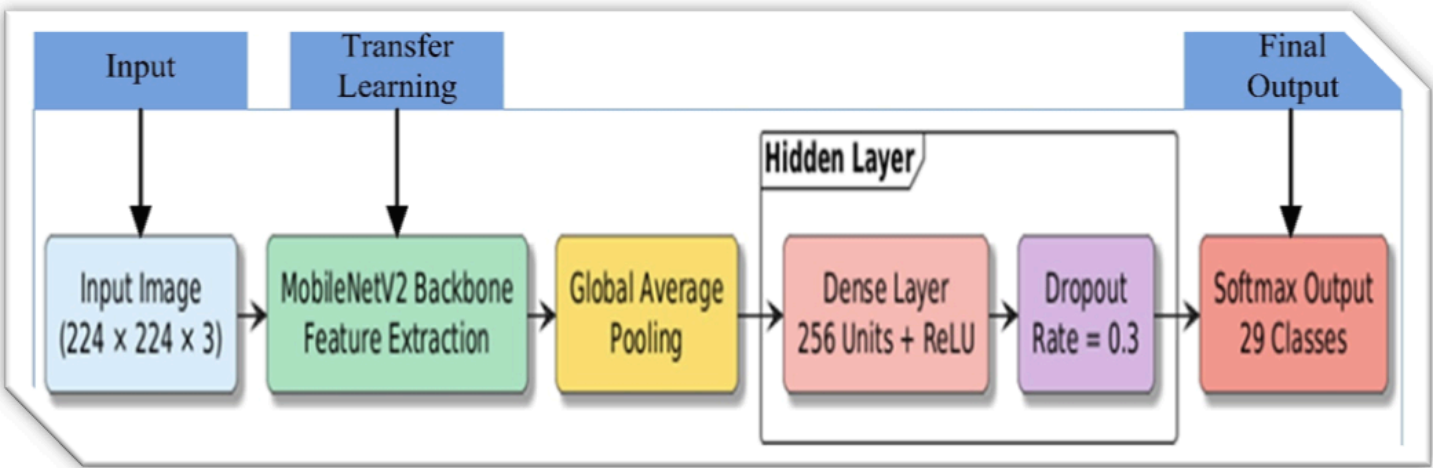


Figure 1

CNN Model Architecture

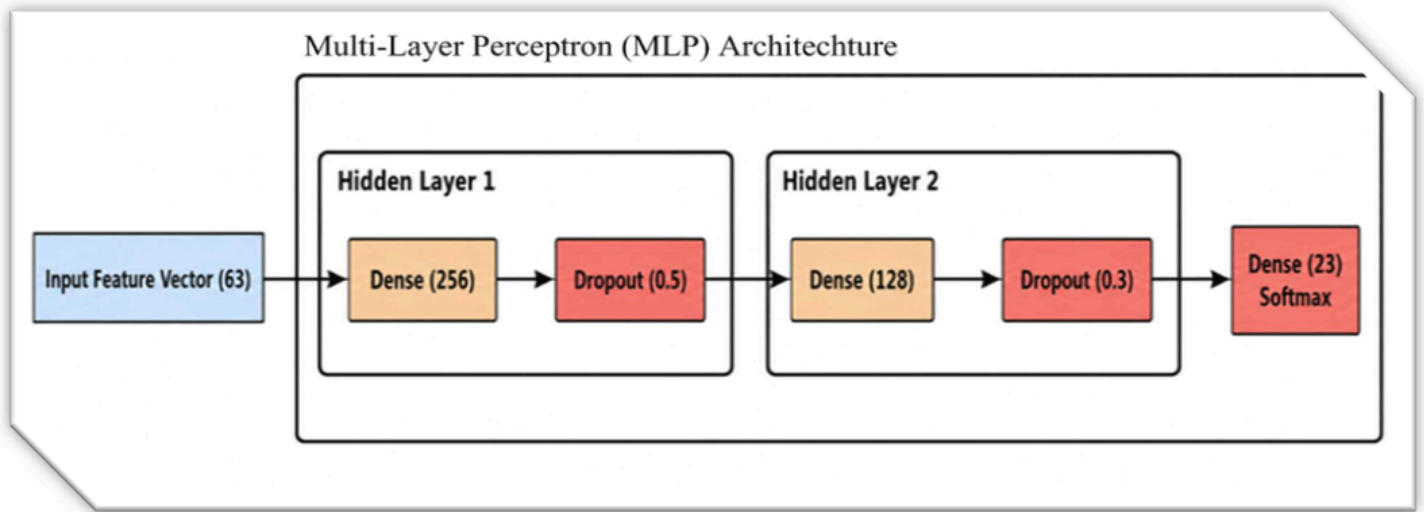


Figure 2

MLP Model Architecture



Figure 3

Hybrid Model Architecture

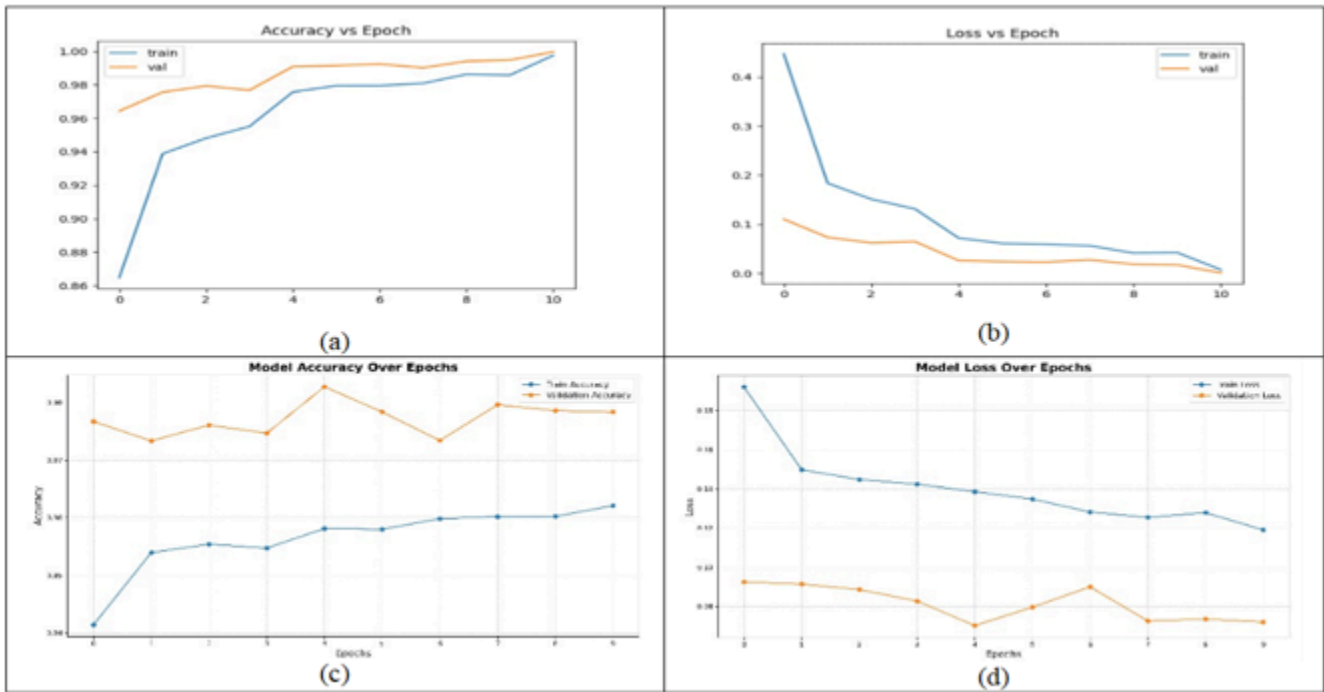


Figure 4

(a) CNN accuracy, (b) CNN loss, (c) MLP accuracy, (d) MLP loss

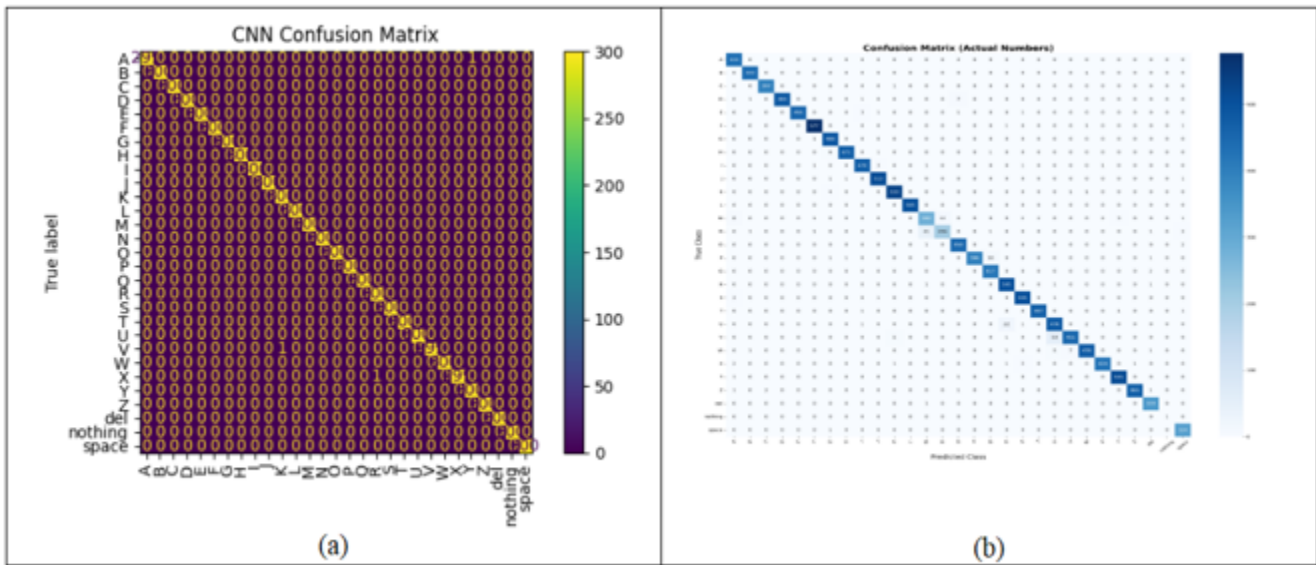


Figure 5

(a) CNN confusion matrix, (b) MLP confusion matrix

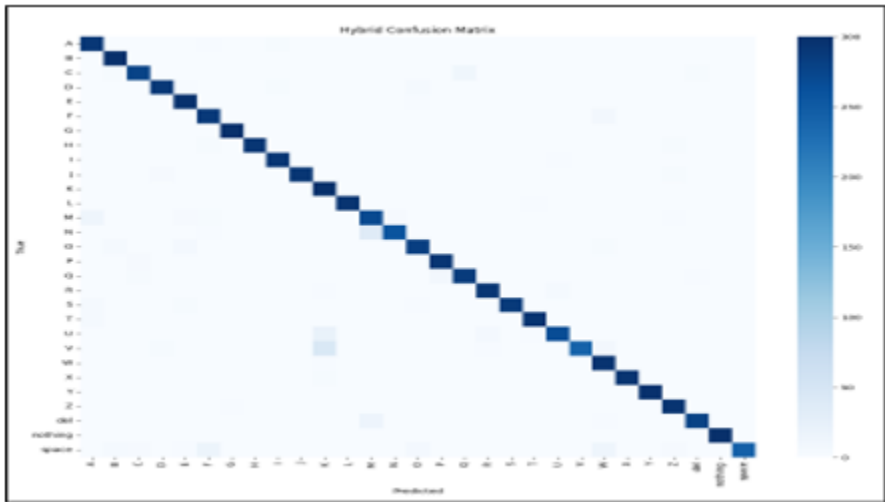


Figure 6

Hybrid confusion matrix

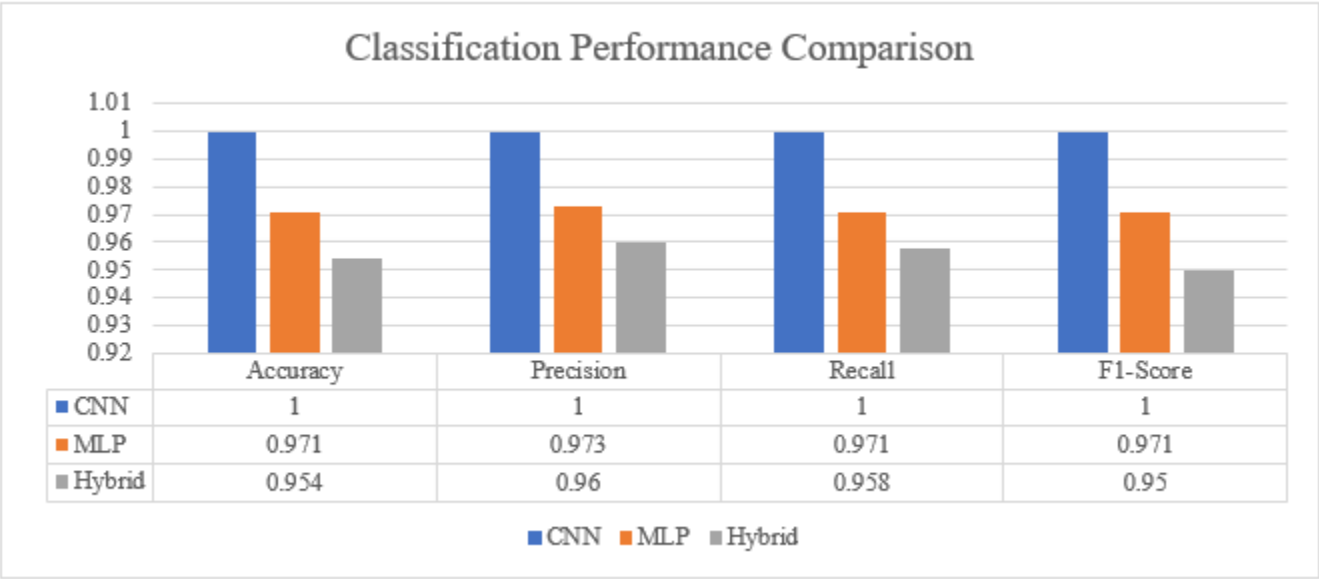


Figure 7

Classification Report

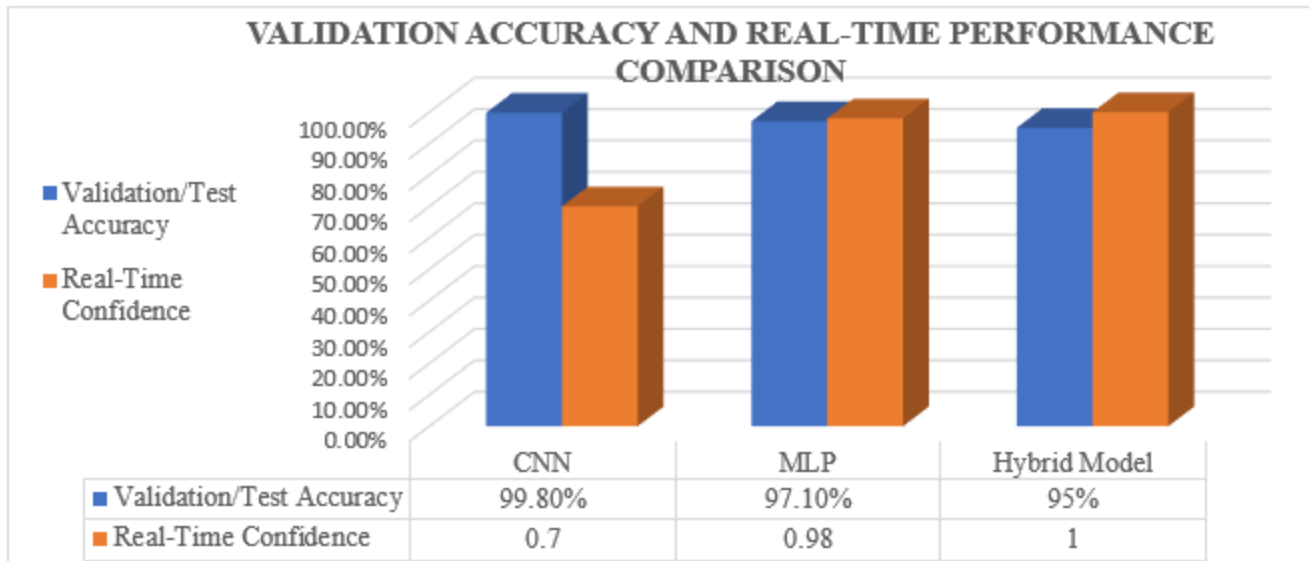


Figure 8

Comparison between validation accuracy with real time ISL accuracy



Figure 9

Real-time gesture recognition results showing predicted ISL alphabets with confidence scores using MediaPipe-based landmark detection.