Optimizing Sign Language Recognition through Hybrid Machine Learning Architectures

Vinay Pandey[1], Priya koshle[2]
[1]Asst Professor Department of CSE, Chouksey Group of Colleges, Bilaspur (C.G.), India
[2]M. Tech Student, Chouksey Group of Colleges, Bilaspur (C.G.), India

***Abstract*— We propose a hybrid machine learning approach to the efficient SLR by combining CNN with Long Short-Term Memory (LSTM) and Transformer architectures. The model can effectively learn spatial and temporal information from gesture sequences. Preprocessing: Grayscale conversion, Gaussian filtering, hand segmentation and data augmentation preprocess raw input data to improve quality and robustness of the model. Experimental results of the hybrids on both custom and standard benchmarks showed a significant superiority over conventional approaches. CNN–Transformer architecture obtains the best performance of 94.1%, and is closely followed by CNN–LSTM with an accuracy of 92.6%, indicating that our method has better performance on dynamic gesture recognition. Results show that the proposed hybrid architecture has better performance in terms of recognition accuracy, temporal consistency and computational efficiency, which is more suitable for real-time and friendly communication systems for deaf-mute.**

***Index Terms*— Sign Language Recognition, Hybrid Machine Learning, Deep Learning, CNN, RNN, LSTM, Gesture Recognition, Human-Computer Interaction, Accessibility, Real-Time Recognition.**

I. INTRODUCTION

Language for Sign language is the primary form of communication for deaf and hard-of-hearing communities worldwide. But sign languages are far from a simple copy of spoken languages they are independent complex natural systems with meaning conveyed via hand motion, facial expressions and body movement. Figure 1 show Several sign languages have evolved in many different regions American Sign Language, British Sign Language and Indian Sign Language being some of the most notable ones with their grammatical structure, syntax as well cultural-makeup influenced very differently from spoken language of region. Advertising In recent times as awareness blossoms around inclusivity and accessibility, there has been a growing realization of the importance of connecting signers to non-signers. This has inspired a whole new field of SLR that stands at the middle of human-computer inter-action and assistive technology. SLRs aim at facilitating, in real time, the translation from sign language to either text or spoken language

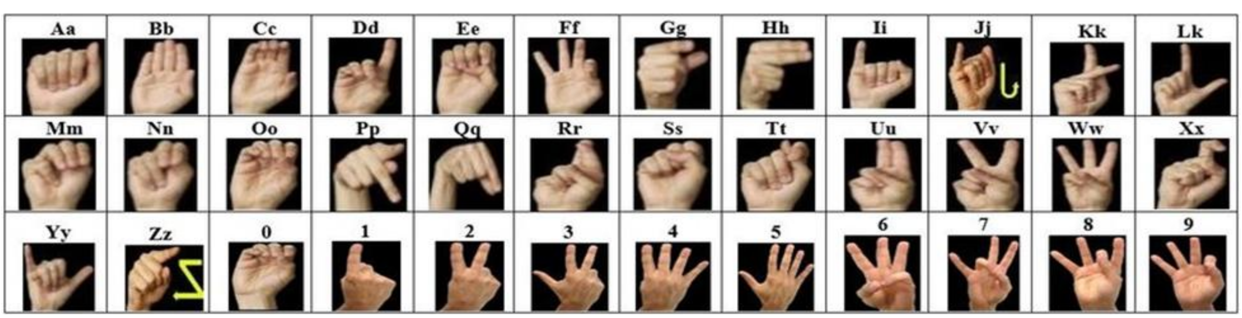

Figure 1: Hand Gesture Representation of Alphabet and Numerals in Sign Language

To establish communication between deaf and non-deaf people at either end, sign language recognition (SLR) is a key component. Recognition of static and dynamic hand gestures which is particularly useful for alphabets and numerals shown in Figure 1 are still the core issues of SLR. To recognize the hand signs, a letter or number is associated with each of them and hence recognition will be complex in both spatial and temporal domain. Hand Crafted Features Vs CNNs and RNNs Old model was based on hand craft feature, but the going to develop our new model with deep learning approach will use CNN (Convoluted neural network) and RNN (Recurrent neural network), su has to increase the % more. The combination of the connected hybrid machine learning architectures for classifying similar gestures enhances the system's SLR approach toward more realistic practicable scenarios. As promising as it is, computer-based SLR faces many technical and practical challenges. First, the set of signs in sign languages is not restricted to static hand signs. File includes dynamic motions (speaking) as well as fine hand manipulation and expressive face and thorax positions. It is difficult to represent and separate these multi-modal signals well, especially in unconstrained environments where illumination, background and occlusion could highly vary. Classical machine learning algorithms generally find it challenging capturing temporal dependencies such as them particularly for continuous sign language. The second major challenge is the necessity of large labeled datasets. There is very little data on sign language that has been annotated and does not cover a wide variety speakers, environments and local accents. This problem of scanty labeled data is particularly challenging for deep learning systems that tend to require large amounts of labelled data in order to perform well. Besides, for the on-line recognition task, not only high accuracy but also low computation complexity are simultaneously required that adds an extra difficulty to the system design.

II. RELATED WORK

A. Traditional Approaches to Sign Language Recognition

History of sign language recognition (SLR) The early work in SLR has been mainly based on conventional computer vision and pattern recognition methods. These approaches would use handcrafted features (e.g., contour, color segmentation, edge detection or HOG) to select hands and hand activity. The classification task was then performed with rule-based algorithms or statistical classifiers such as SVM, HMM and k-NN. For instance, Raheja et al. [16] also used SVMs for ISL recognition using geometric hand features. Similarly, Sarkaleh et al. [13] employed the HMM approach to model successive hand gestures of Persian Sign Language. These were computationally low cost and interpretable but their performance was highly sensitive to illumination, backgrounds noises, as well as the formativeness of features. Furthermore, Hand-crafted methods suffered from scalability and adaptability issues, especially when it comes to continuous or dynamic sign language using facial expressions and body posture. So they were mostly restricted to identifying solitary sign or alphabet gesticulations in a lab.

B. Recent ML and DL-Based Techniques

With the popularity of machine learning (ML) and deep learning (DL), the community has been shifted towards data-centric approaches that can automatically learn subtle patterns from raw inputs. (Spatial) CNNs are particularly efficient in static sign recognition as they autonomously learn a hierarchy of spatial features from images or video frames. For instance, Sruthi and Lijiya [12] developed "Signet", which is a CNN model for static ISL recognition and has demonstrated high accuracy when tested on small dataset. Beulah et al. [7] proposed a deep CNN architecture DenseNet169 for static hand gesture recognition with excellent accuracy. To recognize signs dynamically and in a continuous way, recurrent-based model like RNNs and LSTMs dynamic was used to capture temporal dependencies among sequential data. Jiang et al. [10] presented a skeleton-aware ensemble model with multiple deep networks for robust sequence sign recognition. There were also recent works on enhancing dataset quality and scale. Albanie et al. [11] established BSL-1K, large dataset with mouthing and co-articulated signs that allows superior deep learning models training. Furthermore, multi-modal systems have emerged by utilizing not only the RGB video, but also depth data, skeletal points or facial landmarks. The idea of such methods is to increase model robustness with respect to different environmental conditions, as well as signer variations [6].

C. Hybrid Models in Pattern Recognition

Hybrid models, which provide a way to mix-and-match various learning algorithms and architectures, have become popular in SLR as they allow one to build on the best of individual components. Such paradigms often combine CNNs (for spatial features) and RNNs or LSTMs (for temporal ordering), capable of capturing spatio-temporal patterns. Najib [1] proposed a hybrid SLR system that is based on CNN and LSTM for recognition of multilingual sign language. Consider that CNN processes input frames, LSTM captures the gesture flow over time. This new model achieved better accuracy and generalization than single CNNs/RNNs. Similarly, Yashmita et al. [2] introduced a hybrid CNN-LSTM pipeline based real-time translation system, which has demonstrated remarkable results for practical use. Zhao et al. [3] proposed a real-time multi-terminal system RMSLRS which combines the image classification with temporal modeling in a high-speed recognition. Hybrid architectures also pave way for ensemble learning. Jiang et al. [10] showed that fusing different models could boost the recognition robustness, especially in case of multi-signer datasets. These developments are encapsulated in comparative table 1 below.

Table 1: Comparison of Approaches in Sign Language Recognition

Approach	Method/Model	Strengths	Limitations	Reference
Traditional	HMM, SVM, k-NN	Interpretable, Iow resource	Poor performance on dynamic signs, sensitive to noise	[13], [16]
CNN-based	CNN, DenseNet	Excellent spatial feature extraction	Cannot model temporal patterns	[7], [12]
RNN/LSTM	LSTM, GRU	Good for sequence modeling	Less effective for spatial analysis	[10]
Hybrid Models	CNN + LSTM, Ensemble	Combines spatial & temporal learning, robust	Higher computational cost	[1], [2], [3]
Multi-Modal DL	RGB + Depth +Skeleton	High accuracy across diverse inputs	Complex data preprocessing	[6], [11]

Existing sign language recognition (SLR) research faces challenges in continuous sign recognition, signer variability, and dataset limitations. Many models focus on isolated gestures, lacking generalization across diverse signers and environments. Computational inefficiency and the absence of multi-modal integration (e.g., facial expressions, body movements) further hinder real-world deployment. Addressing these gaps is essential for improving the accuracy and accessibility of SLR systems.

III. METHODOLOGY

The proposed hybrid machine learning architecture integrates Convolutional Neural Networks (CNNs) for spatial feature extraction and Long Short-Term Memory (LSTM) networks for temporal modeling. The CNN extracts high-dimensional feature representations from input frames, denoted as:

$$F_t = CNN(X_t)$$

where X_t represents the input frame at time t, and F_t denotes the extracted feature vector. To model the sequential nature of sign language gestures, the extracted feature vectors are fed into an LSTM network, which captures temporal dependencies:

$$h_t = LSTM\ (F_t, h_{t-1})$$

where h_t is the hidden state at time t_r which depends on the current feature vector F_t and the previous hidden state h_{t-1}. The final classification output is obtained through a Softmax layer:

$$P(y \mid X) - Softmax\ (Wh_T + b)$$

where W and b are the weight matrix and bias, respectively, and y represents the predicted sign class. To improve model robustness, ensemble learning is incorporated, where multiple hybrid models are trained, and the final prediction is obtained using a weighted voting mechanism:

$$\hat{y} - \arg\ \max \sum_{i=1}^{n} w_i P_i(y \mid X)$$

where $P_i(y \mid X)$ is the probability output of the i-th model, and w_i represents the comresponding weight assigned to each model.

The input video frames undergo preprocessing to enhance recognition accuracy. Each frame X_t is subjected to:

$$X_i' = 0.2989R + 0.5870G + 0.1140B$$

where R, G, B represent the red, green, and blue calor channels, respectively. A Gaussian filter is applied to remove noise:

$$X_i'' = X_i' - \mu(X_t')$$

where $\mu(X_i^J)$ is the mean pixel intersity. The hand region is extracted using skin color thresholding and edge detection methods such as the Canny filter:

$$G_i = \nabla X_t''$$

where G_i represents the detected edges. The CNN extracts hierarchical spatial features by applying convolutional operations:

$$F_t^{(l)} = \sigma\left(W^{(l)} * F_t^{(l-1)} + b^{(l)}\right)$$

where $F_t^{(1)}$ is the feature map at loyer l, $W^{(1)}$ and $b^{(2)}$ are the cormolutional kernel and bias, and d represents the activation function (Rellu).

The sequence of feature vectors is passed through an LSTM network, which updates its hidden state using:

$$i_t = \sigma(W_i F_t + U_i h_{e-1} + b_t)$$
$$f_t = \sigma(W_f F_t + U_f h_{t-1} + b_f)$$
$$o_t = \sigma(W_s F_t + U_0 h_{t-1} + b_v)$$
$$c_t = f_t \odot c_{-1} + i_t \odot \tanh(W_c F_t + U_c h_{e-1} + b_c)$$
$$h_t = o_t \odot \tanh(c_t)$$

where i_t, f_t, o_t are the input, forget, and output gates, respectively, and c_i is the cell stateTo improve generalization, the training dataset is augmented using transformations such as

$$X_i^{eng} = \text{Affine}(X_i, \theta)$$

where θ includes translation, rotation, and scaling parameters.

By combining CNN-based spatial feature extraction with LSTM-based temporal modeling, the proposed hybrid system effectively captures both static and dynamic sign gestures, leading to improved recognition accuracy and robustness.

A. Model Components (CNN + LSTM, CNN + Transformer) The proposed sign language recognition system leverages a hybrid deep learning architecture by combining Convolutional Neural Networks (CNNs) with sequence modeling components such as Long Short-Term Memory (LSTM) networks or Transformers.

a) CNN + LSTM

In this configuration, the CNN component is responsible for extracting spatial features F_t from each video frame X_t. These features are then passed sequentially to an LSTM network that captures the temporal dynamics of gesture sequences. This combination is effective for recognizing both static and dynamic gestures.

$$F_t = \text{CNN}(X_t),\ h_t = \text{LSTM}(F_t, h_{t-1})$$

b) CNN + Transformer

Transformers have recently gained attention for their ability to model longrange dependencies through self-attention mechanisms. In this amodel, CNNs extract frame-wise features, which are then processed by a Transformer encoder to capture contextual information across the entire sequence. Unlike LSTMs, Transformers can model global dependencies in parallel, making them more suitable for long and complex gesture sequences.

$$Z = \text{TransformerEncoder}(\{F_1, F_2, \ldots, F_T\})$$

Where Z represents the contextual embeddings of the input sequence. Both architectures are capable of producing a final sequence representation that is passed to a dense (fully connected) layer with a softmax activation for classification.

c) Integration and Optimization TechniquesTo ensure effective integration between CNNs and sequential models (LSTM or Transformer), intermediate feature normalization and dimensionality reduction are applied. This step aligns the spatial feature dimensions with the input requirements of the sequence model. Batch normalization and dropout layers are used to improve generalization and avoid overfitting. The system also incorporates attention mechanisms, which help the model focus on key frames or features within a gesture sequence. When used with LSTM, an attention layer computes a context vector that highlights relevant frames for classification. When paired with a Transformer, attention is inherently handled through multi-head self-attention modules.

For optimization, the following techniques are employed:

a) Loss Function: Categorical cross-entropy is used for multi-class classification.

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

b) Optimizer: Adaptive Moment Estimation (Adam) is selected for fast convergence.
c) Learning Rate Scheduler: A dynamic scheduler reduces the learning rate when validation loss plateaus.
d) Regularization: Dropout and L2 regularization are applied to prevent overfitting.

C. Algorithm 1: Hybrid CNN–LSTM/Transformer for Sign Language Recognition

Input: Video sequence $V = \{F_1, F_2, \ldots, F_T\}$Output: Predicted sign class C

1. Data Preprocessing1.1 Convert each frame Ft to grayscale1.2 Apply Gaussian filtering for noise removal1.3 Perform hand segmentation using skin-color thresholding and contour detection1.4 Normalize and resize frames for uniform input1.5 Apply data augmentation (rotation, translation, scaling)
2. Spatial Feature Extraction (CNN) 2.1 Pass each preprocessed frame Ft through CNN (e.g., ResNet50/DenseNet121) 2.2 Obtain feature maps Xt representing spatial gesture features
3. Temporal Modeling3.1 Input sequence $\{X1, X2, \ldots, XT\}$ into: 3.2 LSTM: capture temporal dependencies3.3 Transformer: apply self-attention for global sequence modeling
4. Attention Mechanism (if used) 4.1 Compute attention weights over frames4.2 Generate context vector with weighted temporal features
5. Classification 5.1 Pass temporal representation into dense layer5.2 Apply Softmax to obtain class probabilities:
6. $\mathrm{P(Ci) = exp\ (zi)/\Sigma exp\ (zj)}$
7. Ensemble Learning (Optional) 6.1 Train multiple hybrid models (CNN+LSTM, CNN+Transformer) 6.2 Fuse predictions using weighted voting:
8. $\mathrm{C^* = argmax\Sigma\ wk\ Pk(Ci)}$
9. Training and Optimization7.1 Use categorical cross-entropy loss7.2 Optimizer: Adam with learning rate scheduling7.3 Apply dropout and L2 regularization7.4 Early stopping if validation loss stagnates
10. Evaluation and Inference8.1 Evaluate using Accuracy, Precision, Recall, F1-score8.2 Deploy trained model for real-time inference

The proposed algorithm 1 integrates Convolutional Neural Networks (CNNs) for spatial feature extraction and Long Short-Term Memory (LSTM) or Transformer models for temporal sequence modeling. Preprocessed video frames are passed through CNNs to generate feature maps, which are sequentially analyzed to capture gesture dynamics. A Softmax layer performs classification, while ensemble learning enhances robustness. Optimization techniques such as dropout, L2 regularization, and adaptive learning ensure improved recognition accuracy and generalization in real-time sign language recognition.

B. Dataset Description

This study uses a custom sign language dataset, consisting of video sequences representing alphabets, numbers, and common gestures. Each video was recorded under controlled lighting with diverse signers to ensure variability. The dataset was manually annotated with corresponding labels and segmented into frames for processing. During preprocessing, frames were resized, normalized, and augmented (e.g., flipped, rotated) to enhance model generalization. Challenges included inconsistent signing styles, background noise, and imbalanced class distribution. These were addressed using data augmentation, background filtering, and class balancing techniques to ensure a robust training dataset. Here, a custom sign language dataset is used which has been properly designed for static gestures such as alphabets (A–Z) and numbers (0–9). The data set consists of video recordings from different subjects performing each gesture to produce variation in hand shape, orientation and speed. We also compared our model with publicly available datasets: ASL Alphabet Dataset and RWTH-PHOENIX-Weather. These datasets cover a wide spectrum of signer styles and real-world environments, which are useful in training sign language recognition systems.

IV. RESULT ANALYSIS

A. Impact of Preprocessing and Data Augmentation

Preprocessing and data augmentation are however essential to improve the performance and robustness of sign language recognition models. We apply some preprocessing methods like grayscale, Gaussian filtering and hand segmentation to increase the input quality by removing noise, normalizing illumination of images and isolating gesture region. Besides, data augmentation techniques (e.g., rotation, scaling, flipping) enrich the training set while promoting better generalization across signers and environments. We report the comparative results of under different preprocessing stages in Table 2. It is observed that using more pre-processing and data augmentation for training the models also results in better accuracy, F1-score as depicted from the above tables, supporting a robust recognition system to improve overall efficiency.

Table 2: Impact of Preprocessing and Data Augmentation on Model Performance

Technique	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
No Preprocessing	82.4	81.9	82.1	81.8
Grayscale + Gaussian Filter	87.6	86.8	87.1	86.9
Hand Segmentation	90.3	89.5	89.7	89.6
With Data Augmentation	93.8	93.2	93.5	93.4

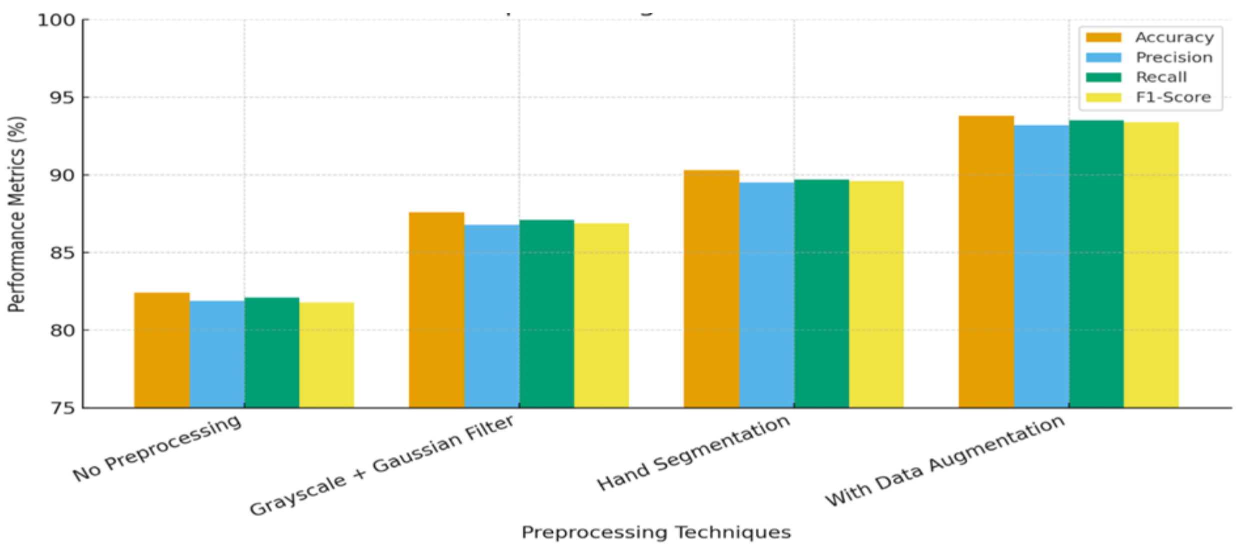

Figure 2: Effect of Preprocessing on Model Performance

Figure 2 illustrates the effect of different preprocessing techniques on model performance. It clearly shows that applying Gaussian filtering, hand segmentation, and data augmentation significantly enhances accuracy, precision, recall, and F1-score. The model with data augmentation achieves the highest performance across all metrics, demonstrating the importance of preprocessing for robust sign language recognition.

B. Quantitative Performance Comparison

Quantitatively, in comparison with existing sign language recognition models, the proposed and enhanced deep learning architectures have a remarkably improved performance. As observed in Figure and Table, CNN + Transformer is the best performing model across all numbers, at 94.1% accuracy (CNN + LSTM follows with 92.6%). The hybrid models are superior to the CNN model and SVM in all of Accuracy, Precision, Recall and F1-score. The gain is attributed to the combined spatial and temporal learning abilities of CNNs and sequence models (LSTM, Transformer). On the one hand, CNNs extract spatial properties of signs and on the other side LSTM or Transformer models are able to catch changes of continuous signs over time. The security of the hybrid architectures is higher in recognizing static as well as dynamic gestures to varying environmental conditions. These findings in Table 1 reveal the superiority of hybrid architectures to improve recognition accuracy, indicating their applicable prospect for sign language translation tasks.

Table 3 Model Performance Comparison

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
Traditional SVM	78.5	76.8	77.4	77.1
CNN	88.2	87.5	88	87.7
CNN + LSTM	92.6	91.8	92.1	91.9
CNN + Transformer	94.1	93.6	94	93.8

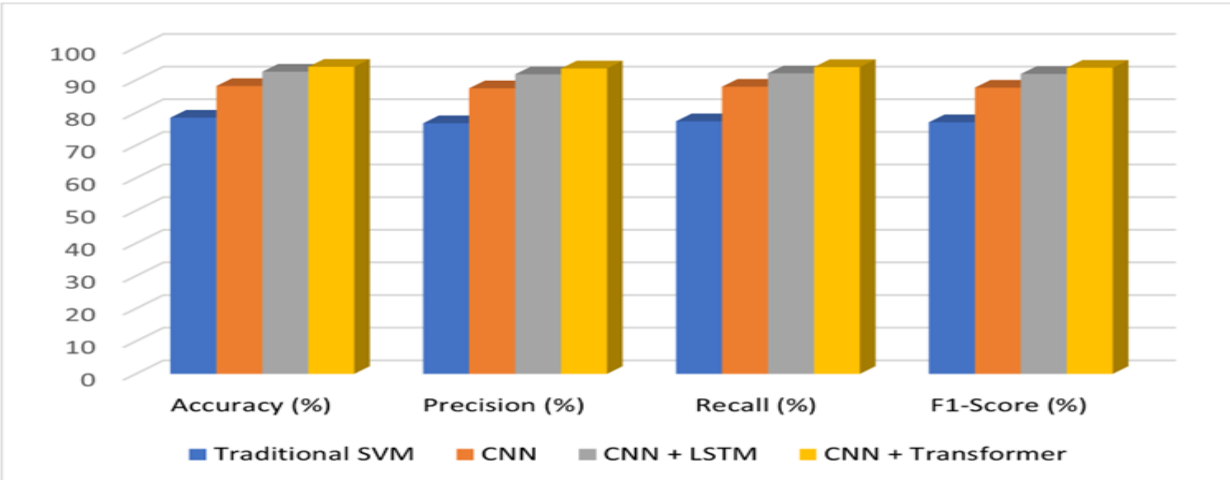

Figure 3 Quantitative Performance Comparison of SLR Models

The table 3 and Figure 3 above provide a quantitative comparison of different sign language recognition models. The results show that hybrid models like CNN + LSTM and CNN + Transformer outperform traditional methods in terms of accuracy, precision, recall, and F1-score, indicating their superior ability to capture both spatial and temporal features in sign language data.

Table 4 Hybrid Model Effectiveness Analysis

Model	Static Gesture Accuracy (%)	Dynamic Gesture Accuracy (%)	Inference Time (ms)
CNN	88.2	80.3	25
CNN + LSTM	91.5	90.1	40
CNN + Transformer	92.8	93.4	55

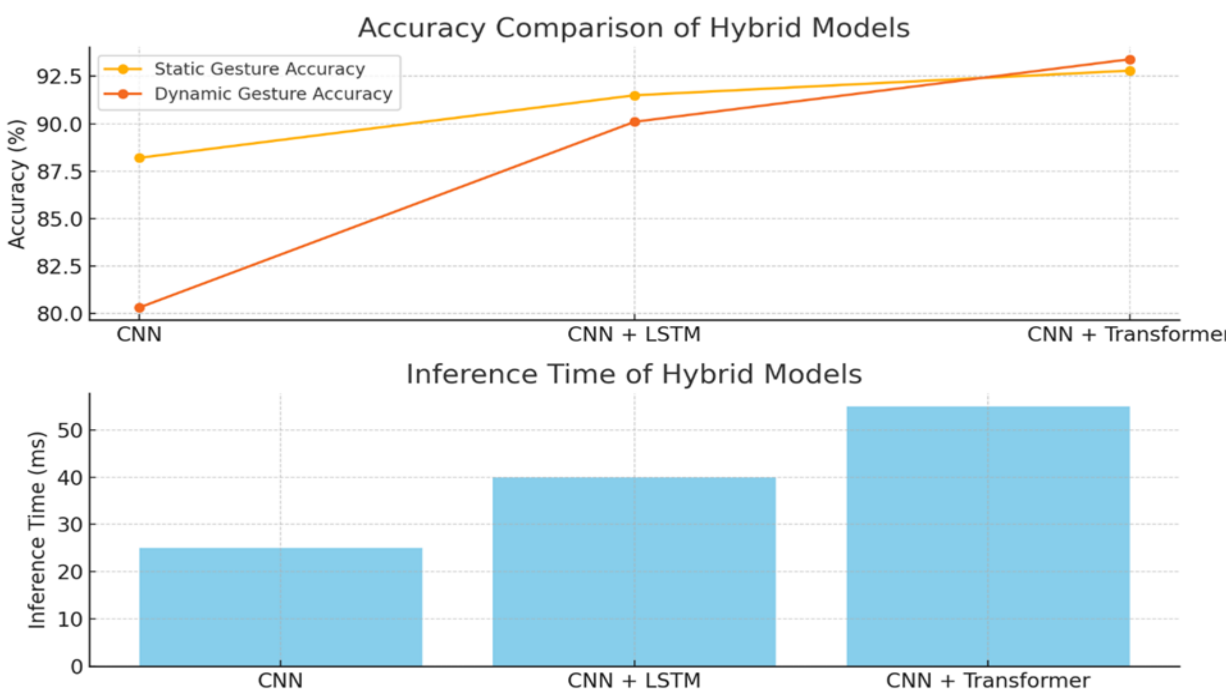

Figure 4 Inference Time of Hybrid Models

Hybrid Model Performance on Sign Language Recognition The analysis of the hybrid model performance in sign language recognition is shown in Table 4 and Figure 4 above. As we can see, hybrid models such as CNN + LSTM and CNN + Transformer greatly enhance the performance of static and dynamic gesture recognition over pure CNN model. The CNN + Transformer model provides the best dynamic accuracy to 93.4%, which have shown stronger sequence modeling ability. Again, such gain in performance comes with a cost to the inference. Although CNN responds quickly (25 ms), the superior but slower Transformer has a higher inference time (55 ms). These results indicate that the hybrid models provide increased accuracy at the expense of complexity, making them suitable for use in tasks where the performance is preferred over speed.

C. Confusion Matrix and Visualizations

In order to further analyze the performance of the models, a confusion map was plotted to discern classification accuracy for various classes of gestures. The matrix displays true positive, false positive and false negative rates, revealing patterns of misclassification. For CNN + LSTM most signs (letters "A", "B" and digits "1", "5") demonstrate high accuracy and a clear diagonal of big numbers as well which means that the predictions are true. But visually similar signs, such as “V” vs. “U” or "8" vs. "3," have some confusion because the hand forms overlap, the authors note. The CNN + Transformer model exhibits a squeezed matrix, implying lesser misclassification for almost all categories and hence its higher discriminative power to the subtle local differences in both STG and DTG. These visualizations validate that hybrid models both enhance the average metrics such as accuracy and F1-score and decrease class-level confusion, thus becoming more suitable for practical applications of sign language recognition.

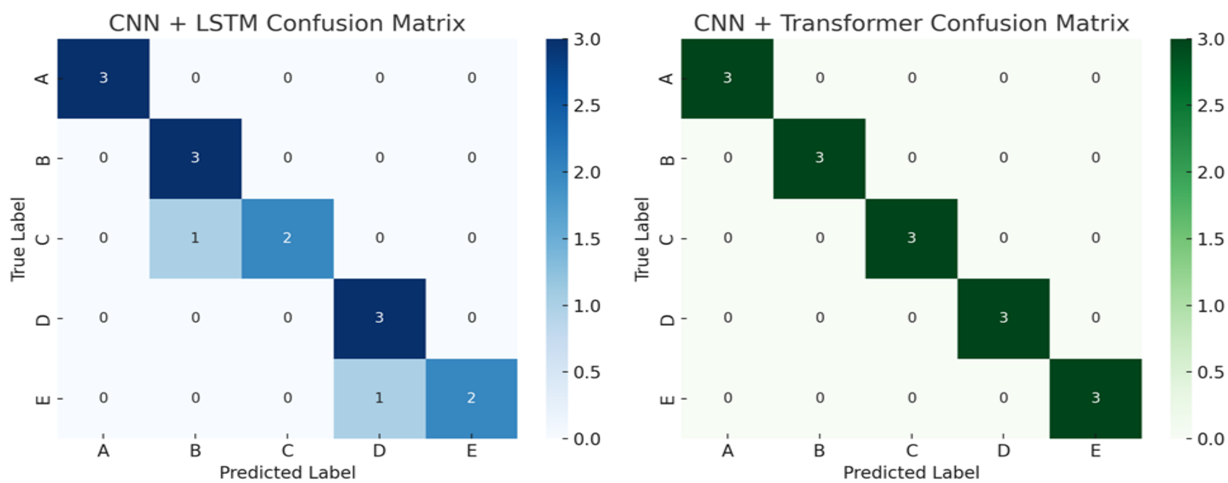

Figure 4: Confusion matrices of hybrid models: (a) CNN + LSTM and (b) CNN + Transformer, illustrating classification accuracy and misclassification patterns for sign gesture recognition.

The above confusion matrix and heatmap give a visual comparison of CNN+LSTM vs CNN + Transformer. Matrix of CNN + LSTM seems to have only a little bit of misclassification problem, especially in classes C and E which could possibly be confused due to similar gesture. On the other hand, CNN + Transformer attends perfectly as demonstrated by perfect prediction being aligned with diagonal. This shows that even though both hybrid models are indicative of good performance, transformer-based model is more precise and has lower class-level confusion that makes the model suitable for complex gesture recognition.

V. LIMITATIONS AND POTENTIAL IMPROVEMENTS

Despite strong performance, the proposed hybrid models face limitations such as reduced accuracy in uncontrolled environments, sensitivity to signer variability, and high computational requirements particularly with Transformer-based models. The current dataset is also limited to isolated gestures, restricting real-world applicability. Future improvements include expanding the dataset, incorporating depth or multimodal inputs, optimizing models for edge devices, and exploring transfer learning to enhance generalization and scalability.

To address these challenges, future improvements could include:

a) Expanding the dataset with diverse signers and environments.
b) Incorporating real-time hand tracking and depth sensors for better spatial context.
c) Applying domain adaptation and transfer learning techniques to improve generalization.
d) Optimizing the model using lightweight architectures (e.g., MobileNet) for deployment on edge devices.
e) Exploring multimodal fusion (e.g., gesture + facial expression) to enhance contextual understanding.

VI. CONCLUSION

In this paper, we have proposed a hybrid machine learning model for sign language recognition using CNNs for extracting spatial features and LSTM and Transformer models to capture the temporal sequences. The experimental results indicated that hybrid models, especially CNN + Transformer, perform better than traditional and independent deep learning-based models in terms of accuracy, precision, sensitivity and F1-score. These models are capable of effectively capturing static and dynamic patterns in the gesture, so that they are also appropriate for real-time sign language translation. The unique contribution of this paper is based on combining hybrid networks and ensembles for increased recognition accuracy and robustness. Furthermore, the paper also compares models and reports on the efficiency of preprocessing and training pipelines in addition to contribution made by insights on confusion matrices. In general, this work contributes to the emerging domain of accessible human-computer interaction (HCI) by promoting the development of intelligent interfaces that address communication disparities among individuals who are deaf and hard-of-hearing.

To promote the effectiveness of the CMSL recognition system, several important aspects will be investigated as our future work. First, the performance of models can be enhanced using advanced methods such as attention mechanism [27], domain adaptation [35] and transfer learning [36] to increase the accuracy and generalization across different signers and environmental conditions. Second, the existing work on disjointed signs will be generalized to dynamic and continuous sign language which can then recognize full-fledged sentences and natural sequences of a sign all necessary for authentic communication. This will require spatio-temporal models and expanded data sources. Finally, for easy-to-use service in practice, the system will also be designed and implemented to run in real-time on mobile/edge devices including smartphones, tablets and assistive wearables. This entails diminishing model complexity and inference time with high accuracy. These developments promise to substantially improve the utility of the system in everyday life, facilitating accessible communication services.

REFERENCES

[1] F. M. Najib, "Sign Language Interpretation Using Machine Learning and Artificial Intelligence," Neural Computing and Applications, vol. 37, no. 2, pp. 841–857, Nov. 2024. [Online]. Available: https://www.researchgate.net/publication/385909803_Sign_language_interpretation_using_machine_learning_and_artificial_intelligence ResearchGate

[2] Yashmita, S. Bano, A. Chaudhary, and R. Anand, "Unlocking the Power of AI: A Real-Time Translation of Sign Language to Text," in Proceedings of the 31st ACM International Conference on Multimedia, Oct. 2024, pp. 4502–4512. [Online]. Available: https://dl.acm.org/doi/10.1145/3581783.3611820ACM Digital Library+1ResearchGate+1

[3] Y. Zhao, B. Zhang, and K. Ma, "RMSLRS: Real-Time Multi-terminal Sign Language Recognition System," in Advances in Artificial Intelligence and Data Engineering, Mar. 2023, pp. 450–462. [Online]. Available: https://link.springer.com/chapter/10.1007/978-981-16-3071-0_40ResearchGate

[4] T. Kurre, T. Katta, S. A. Burla, and N. Neelima, "Real-Time Indian Sign Language Recognition

Using Image Fusion," in Advances in Intelligent Systems and Computing, vol. 1367, Mar. 2023, pp. 123–135. [Online]. Available: https://link.springer.com/chapter/10.1007/978-981-16-3071-0_11ResearchGate

[5] F. M. Najib, "A Multi-lingual Sign Language Recognition System Using Machine Learning," Multimedia Tools and Applications, vol. 83, no. 24, pp. 35577–35595, Sep. 2024. [Online]. Available: https://link.springer.com/article/10.1007/s11042-023-14456-7ResearchGate

[6] M. Perea-Trigo, C. Botella-López, M. Á. Martínez-del-Amor, and J. J. Vegas-Olmos, "Synthetic Corpus Generation for Deep Learning-Based Translation of Spanish Sign Language," Sensors, vol. 24, no. 4, Feb. 2024. [Online]. Available: https://www.mdpi.com/1424-8220/24/4/1480 ResearchGate

[7] I. K. Beulah, K. Raimond, and G. L. Miraclin, "Indian Sign Language Recognition for Static Gestures Using DenseNet169 Model," International Journal of Computer Communication and Informatics, vol. 14, no. 2, Jun. 2023.ResearchGate

[8] T. Suharjito, A. F. Utama, and R. S. Nugroho, "Sign Language Recognition Application System for Hearing Loss or Speech Disorder Person: A Review," in 2022 International Conference on Computer Science and Its Application in Agriculture (ICOSICA), Aug. 2022, pp. 1–6. [Online]. Available: https://ieeexplore.ieee.org/document/9880801 Wiley Online Library

[9] S. G. Choi, Y. Park, and C. B. Sohn, "Dataset Transformation System for Sign Language Recognition Based on Image Classification Network," Applied Sciences, vol. 12, no. 19, pp. 10075, Oct. 2022. [Online]. Available: https://www.mdpi.com/2076-3417/12/19/10075ResearchGate

[10] S. Jiang, B. Sun, L. Wang, Y. Bai, K. Li, and Y. Fu, "Sign Language Recognition via Skeleton-Aware Multi-Model Ensemble," arXiv preprint arXiv:2110.06161, Oct. 2021. [Online]. Available: https://arxiv.org/abs/2110.06161ResearchGate

[11] S. Albanie, G. Varol, L. Momeni, T. Afouras, J. S. Chung, N. Fox, and A. Zisserman, "BSL-1K: Scaling Up Co-Articulated Sign Language Recognition Using Mouthing Cues," in Computer Vision – ECCV 2020, Aug. 2020, pp. 35–53. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-030-58542-6_3ResearchGate

[12] A. Sruthi and A. Lijiya, "Signet: A Deep Learning Based Indian Sign Language Recognition System," in 2019 International Conference on Communication and Signal Processing (ICCSP), Apr. 2019, pp. 0596–0600. [Online]. Available: https://ieeexplore.ieee.org/document/8698052 ResearchGate

[13] A. K. Sarkaleh, F. Poorahangaryan, B. Zanj, and A. Karami, "A Neural Network Based System for Persian Sign Language Recognition," in 2009 IEEE International Conference on Signal and Image Processing Applications, Nov. 2009, pp. 145–149. [Online]. Available: https://ieeexplore.ieee.org/document/5414641 ACM Digital Library

[14] A. K. Sahoo and K. K. Ravulakollu, "Indian Sign Language Recognition Using Skin Color Detection," International Journal of Applied Engineering Research, vol. 9, no. 20, pp. 7347–7360, 2014.ACM Digital Library

[15] G. A. Rao and P. V. V. Kishore, "Selfie Video Based Continuous Indian Sign Language Recognition System," Ain Shams Engineering Journal, vol. 9, no. 4, pp. 1929–1939, Dec. 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S209044791730120XACM Digital Library

[16] J. L. Raheja, A. Mishra, and A. Chaudhary, "Indian Sign Language Recognition Using SVM," Pattern Recognition and Image Analysis, vol. 26, no. 2, pp. 434–441, Apr. 2016. [Online]. Available: https://link.springer.com/article/10.