International Journal of
Human Computations & Intelligence
Vol. 04, Issue. 05, 2025

RESEARCH ARTICLE OPEN ACCESS

Indian Sign Language Understanding Through Deep Transfer Learning and Vision Models

Vishal Kumar Jaiswal

Sr. Manager Software Engineering,
OPTUM, Ashburn, Virginia - 20148, USA.

DOI: **10.5281/zenodo.15745216**
Received: 22 May 2025 / Revised: 19 June 2025 / Accepted: 26 June 2025

Abstract – Communication barriers remain a significant challenge for individuals with hearing and speech impairments, especially in regions where sign language literacy among the general population is limited. The gap in accessible communication tools for the deaf and hard-of-hearing population remains a pressing issue, particularly in countries like India, Pakistan and Bangladesh, where public awareness and proficiency in sign language are minimal. This paper introduces EfficientSign-ISL, a robust and lightweight deep learning model for recognizing Indian Sign Language (ISL) gestures. The model is built upon the EfficientNetB0 architecture, which employs compound scaling to optimize accuracy and computational efficiency. To achieve this, we collected data from various individuals' work, personal gatherings and augmented a dataset of several ISL gestures of 10 distinct classes. To verify our methodology's efficacy, we thoroughly compared well-known transfer learning models, such as ResNet50, MobileNetV2, and InceptionV3. With an excellent overall accuracy of 99.38%, an F1-score of 96.96%, a precision of 97.54%, and a recall of 97.10%, our suggested model fared better than these baselines. These results demonstrate the model's strong classification capability while maintaining low complexity, making it highly suitable for mobile applications or edge device deployment.

Index Terms – Indian Sign Language (ISL), Sign Language Recognition, Hindi Sign Language (HSL), EfficientNetB0, Deep Learning, Transfer Learning, Gesture Classification, Computer Vision, Deaf Communication.

I. INTRODUCTION

Hearing loss is a serious illness that affects millions of individuals globally and impairs their ability to perceive and understand auditory stimuli [1]. The World Health Organization projected that 538 million people were deaf and that over 1.1 billion people had some form of hearing loss by 2013 [2]. Deaf and Mute (D&M) people with severe hearing and speech difficulties frequently use sign language instead of

MILESTONE RESEARCH.IN OPEN ACCESS

ISSN (Online): 2583-5696
Int. Jr. of Hum Comp. & Int.

verbal communication as their primary form of expression [3]. Sign language enables D&M persons to communicate ideas, emotions, and commands using a methodical system of hand gestures, facial expressions, and body movements [4]. Significant communication barriers exist in daily encounters since the general population still does not recognize sign language despite its success in the D&M field. Individuals who use sign language face challenges such as social exclusion, reduced employment opportunities, and difficulty accessing essential support [5]. Researchers are increasingly adopting automated Sign Language Recognition (SLR) systems, including computer vision and artificial intelligence, to convert signs into written or spoken language to bridge this communication gap [6].

There are significant differences between sign languages in various linguistic and cultural situations. There are differences in the grammatical structures and gesture vocabularies of American Sign Language (ASL), British Sign Language (BSL), Japanese Sign Language (JSL), and Arabic Sign Language (ArSL) [7]. The independently created Indian Sign Language (ISL) is very different from other international sign systems and captures the sociocultural subtleties of India. Despite its unique relevance, ISL remains understudied in computer sign identification, mainly because there aren't enough standardized datasets or advanced recognition frameworks tailored to its structure. In deep learning, transfer learning has appeared as an effective model that enables models that have already been trained on large datasets, such as ImageNet, to be reused for specific applications with less training data. In light of the paucity of annotated data for ISL identification, this is highly beneficial. Promising outcomes in sign language studies have been observed with VGG16, ResNet50, InceptionV3, and MobileNetV2 transfer learning models. Unfortunately, many of these models need much processing power or don't provide the best accuracy-to-efficiency ratio for real-time or mobile applications.

To overcome these constraints, we present EfficientSign-ISL, an innovative framework for sign language recognition that uses the EfficientNet architectural family, namely EfficientNetB0, as a backbone for transfer learning. EfficientNet presents a compound scaling approach that balances network depth, breadth, and resolution to achieve high accuracy with little computing cost. With an emphasis on speed, generalization, and scalability, our suggested model is made to identify static hand movements in ISL. EfficientSign-ISL is a lightweight solution for real-time applications that captures fine-grained gesture information by combining EfficientNetB0 with a specially designed classification head.

The contribution of our work is as follows:

- We introduce EfficientSign-ISL, a novel transfer learning-based framework for Indian Sign Language (ISL) recognition that leverages the EfficientNetB0 architecture for high accuracy with minimal computational cost.
- Despite limited training data, we effectively adapt a pre-trained EfficientNet model to the Indian Sign Language domain, demonstrating its ability to generalize well to culturally unique gesture sets.
- Our work targets the recognition of static ISL hand gestures, optimizing the model for real-time classification tasks relevant to alphabetic and symbolic communication.
- We perform extensive experiments comparing EfficientSign-ISL with other state-of-the-art transfer learning models such as ResNet50, MobileNetV2, and InceptionV3, demonstrating superior performance in terms of accuracy and efficiency.

- The lightweight nature of EfficientSign-ISL enables deployment in mobile and embedded systems, paving the way for accessible assistive technologies for the Deaf and Mute (D&M) community in India.
- We curated and preprocessed a structured dataset of Indian Sign Language images to train and validate our model, contributing a foundation for future research in this underrepresented linguistic domain.

The rest of the article is organized as follows: Section II provides a thorough literature overview of earlier research on sign language recognition. Section III describes our suggested model architecture and training process. Section IV covers performance assessments and experimental findings, and Section V concludes with conclusions and suggestions for further research.

II. LITERATURE SURVEY

Researchers have developed an Asian region such as Indian, Bengali, and Pakistani sign language detection system using a variety of approaches and techniques. Some of the current research projects in the literature have been discussed in this section. Navin et al. [8] introduced a dual-language sign recognition model using the YOLOv11 object detection framework, trained on a composite dataset of 9,556 labeled images from American Sign Language (ASL) and BdSL. Their model achieved a high precision of 99.12% and an average recall of 99.63% over 30 epochs, demonstrating its real-time detection capabilities.

In another effort, Rubaiyeat et al. [9] presented a novel embedding technique called Relative Quantization Encoding (RQE), which aligns motion trajectories with anatomical landmarks to improve spatial consistency. Their framework notably reduced word error rates by 44.3% on WLASL100 and demonstrated strong performance across multiple Bangla-centric datasets, such as SignBD-200 and BdSLW60. Although RQE and its extension RQE-SF enhance attention and landmark stability, the rigid quantization strategy becomes less effective on large-scale sign databases, highlighting the need for more flexible encoding mechanisms. Ahmed et al. [10] explored a real-time static sign recognition approach using YOLOv10. Their system was trained on a curated dataset of 1,949 images representing 14 unique BdSL gestures. By integrating object detection with gesture classification, the model achieved an average accuracy of 90.67%, with nearly perfect precision across all classes. While promising in terms of performance, the limited vocabulary restricts the model's usability in practical communication scenarios that require dynamic sign interpretation.

Hossen et al. [11] employed a Deep Convolutional Neural Network (DCNN) based on VGG16, pre-trained on ImageNet, to recognize 37 static BdSL signs. During testing, the model's accuracy fell to 84.68% from its peak of 96.33% on the training set, suggesting that it may have overfitted and had limited generalization. In a parallel effort, a group [11] worked with the National Federation of the Deaf to create a library of more than 12,000 BdSL pictures using all 38 alphabets. Utilizing a VGG19-based CNN, they achieved an overall testing accuracy of 89.6%, though the evaluation lacked comprehensive metrics such as precision and recall. Another study [12] proposed a comprehensive approach combining segmentation, augmentation, and CNN-based classification to enhance performance across three benchmark BdSL

datasets: "38 BdSL," "KU-BdSL," and "Ishara-Lipi." In order to partition gesture regions, their preprocessing pipeline used a watershed technique in conjunction with YCbCr and HSV color models. The advantage of combining accurate segmentation with reliable feature extraction was shown by the remarkable 99.60% accuracy of the final classification, which employed a modified CNN named BenSignNet.

Communication between them is minimized by Patel et al. [13]. Hand motions are recorded in this paper, processed using MATLAB, and then translated into text and voice. Two languages—English and Hindi—are selected for voice and text, and the moment approach is used to assess the feature value of the pictures. Two classification methods—PNN and KNN—are employed, and the two classifiers' performances are contrasted. The simplest possible communication between normal and deaf persons will be made possible. Using three distinct models and the You Only Look Once version 5 (YOLOv5) algorithm, Biyani et al. [14] suggest real-time sign language identification models for deaf and mute people. Three models are identified: the first recognizes American Sign Language, the second acknowledges Hindi/Marathi Sign Language, and the third recognizes static motions. The unique data set was used to train the YOLOv5 models, which produced excellent accuracy. A camera was used to record live sign language motions to assess the model's real-time performance; on average, the detection time was 0.05 seconds per frame. The custom data collection comprises 1892 pictures for the models' implementation, 829 training photos for American sign language that represent various sign language motions, 611 training images for Hindi/Marathi sign language, and 452 training images for static gestures. Regarding accuracy and real-time performance, the suggested models outperformed existing deep learning models, such as Faster R-CNN and Mask R-CNN.

The Hindi vowel static gesture dataset was extracted by Singh et al. [15] from various age groups, including children, adults, and elderly citizens. Depending on the area, each sign language is distinct. Indian Sign Language, in contrast to American Sign Language, is signed with both hands, which makes interpretation more difficult due to hand and finger occlusion. Nevertheless, considering all these difficulties, a state-of-the-art technique in this paper uses a deep learning technique called YOLO to interpret the static Hindi vowel gestures in Indian Sign Language. We have achieved a 94.60% accuracy rate with good precision and recall values. A suggested pipeline for the Pakistan Sign Language recognition system that includes an augmentation unit was presented by Hamza et al. [16]. C3D, I3D, and TSM are three deep-learning models used to verify the efficacy of the suggested pipeline. According to the results, the two most effective augmentation methods for the Pakistan Sign Language dataset are translation and rotation. Compared to alternative approaches that employed the original data, the models trained to utilize the data-augment-supported pipeline perform better. The most appropriate model is C3D, which has a minimal training time compared to other models and yielded an accuracy of 93.33%.

Mirza et al. [17] focused on building a vision-based recognition system for both static and dynamic Pakistani Sign Language (PSL) alphabets using the Bag-of-Words (BoW) model in combination with Support Vector Machines (SVM). The dataset comprised 5,120 images representing 36 static PSL signs and 353 video sequences (totalling 45,224 frames) for three dynamic PSL signs, all captured from 10 native PSL users. RGB images were first resized to various resolutions and converted to grayscale to process the data. A thresholding segmentation process was applied, followed by feature extraction with the Speeded Up Robust Features (SURF) method. The extracted descriptors were then grouped using K-

means clustering to construct visual codebooks. BoW representations were generated for classification by calculating Euclidean distances between SURF descriptors and cluster centres. Fivefold cross-validation was employed to evaluate model performance. The system achieved a peak accuracy of 97.80% for static sign recognition at an image resolution of 750 × 750 using 500 visual words. In comparison, the dynamic sign recognition achieved 96.53% accuracy at a video resolution of 480 × 270 with 200 visual words.

While these studies underscore notable progress in ISL recognition, most rely on small datasets or computationally expensive models. Additionally, only a few addresses the challenge of real-time, scalable performance suitable for mobile or edge deployment. This motivates the need for a more efficient and generalized solution. Our work builds upon these foundational studies by proposing EfficientSign-ISL, which leverages the compound scaling advantages of EfficientNetB0 to achieve high recognition accuracy with reduced computational overhead. Unlike prior models, EfficientSign-ISL is optimized for real-time applications and demonstrates strong generalization on static ISL alphabets with fewer resources, marking a significant step toward deployable and accessible sign language recognition systems for the Bangladeshi context.

III. METHODS & MATERIALS

This section contains the methodologies and approaches that we have proposed. Here are thorough explanations of every step. Figure 1 displays the general framework of our study graphically.

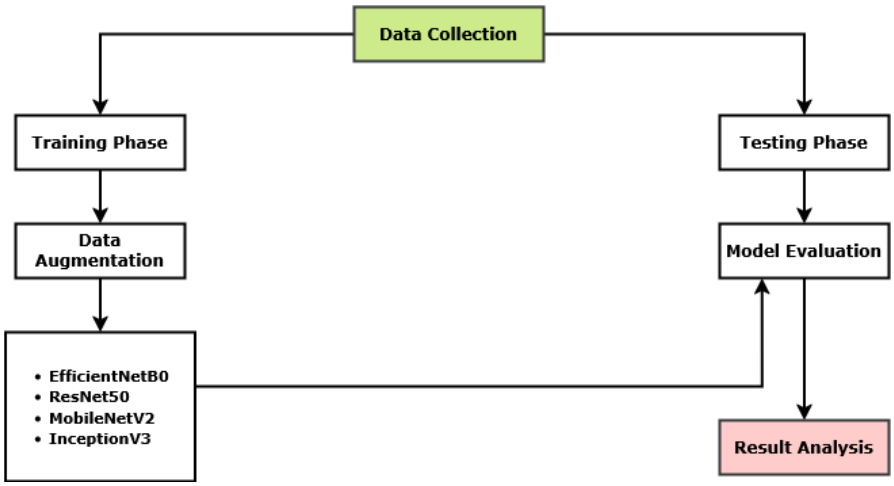

Fig. 1: Overall Framework of our research

A. Dataset Description

In this work, we developed a custom dataset for Hindi Sign Language (HSL) recognition because publicly accessible datasets in this field are noticeably lacking. We manually collected all image samples used in the model training and evaluation phases to address this gap. The dataset includes hand gesture images representing 43 distinct Hindi alphabets, and we categorized them into 10 distinct classes. A Samsung smartphone with a 48MP camera was used to take these pictures. To increase flexibility and resilience, the photos were taken in various lighting circumstances and from several perspectives. Images for signs were reviewed to ensure uniformity and authenticity in the rendering of gestures. Figure 2 displays a visual representation of the dataset.

Fig 2: Sample of the dataset

B. Data Preprocessing and Augmentation

We employed a series of data augmentation techniques to improve generalization and mitigate overfitting using the ImageDataGenerator module. The augmentation strategies applied include:

- Random rotations within ±30 degrees,
- Horizontal and vertical shifts of up to 20% of the image dimensions,
- Shear transformations up to 20%,
- Zoom operations up to 20%,
- Horizontal and vertical flipping,
- Brightness adjustments range from 0.8 to 1.2.

All images were uniformly resized to 64×64 pixels to standardize the input shape for the model. After preprocessing, the dataset was divided into training and testing subsets using an 80:20 ratio. This resulted in 1,186 images allocated for training and 317 images reserved for testing.

C. Baseline Model Descriptions

To evaluate the performance of our presented model, EfficientSign-ISL**,** we benchmarked it against three state-of-the-art convolutional neural network architectures: ResNet50**,** MobileNetV2**,** and InceptionV3 [18]**.**

- **ResNet50**
  The deep CNN architecture known as ResNet50 (Residual Network with 50 layers) uses residual learning to solve the vanishing gradient issue [19]. ResNet learns residual functions concerning the layer inputs through identity shortcut connections rather than directly learning the underlying mapping. Each residual block is formulated as follows:

$$F(x) = H(x) - x \Longrightarrow H(x) = F(x) + z$$

  Where:
  - H(x) is the original desired mapping,
  - x is the input to the residual block,
  - F(x) is the residual function.

  This formulation allows very deep networks (such as 50 layers) to train effectively without degradation in accuracy.

  - Input Size: 224×224×3224
  - Total Parameters**:** ~25.6 million
  - Strengths**:** Excellent feature learning for complex visual patterns and deeper semantic representation.

- **MobileNetV2**
  MobileNetV2 is developed for resource-constrained environments like mobile and edge devices. It introduces two key innovations:
  - Inverted Residual Blocks**,** which expanse then compress channels around a depthwise convolution,
  - Linear Bottlenecks**,** which reduce representational bottlenecks while preserving information flow.

  Each block follows the sequence:

  Input→1×1 Convolution (expansion)→Depthwise 3×3 Convolution→1×1 Linear Convolution (p rojection)→Residual Connection

  - Input Size**:** 224×224×3224
  - Total Parameters**:** ~3.4 million

- **InceptionV3**
- The sophisticated CNN InceptionV3 uses Inception modules, which are mixtures of parallel convolutions with various filter sizes (e.g., 1×1, 3×3, 5×5), to extract features across multiple scales. To enhance learning effectiveness and generalization, it additionally makes use of label smoothing, auxiliary classifiers, and factorized convolutions.

$$\text{Inception Module Output} = [\text{Conv}_{1\times1}, \text{Conv}_{3\times3}, \text{Conv}_{5\times5}, \text{MaxPooling}_{3\times3}]$$

  - Input Size: 299×299×3
  - Total Parameters: ~23.8 million

D. Proposed EfficientSign-ISL Model

To efficiently and accurately interpret Indian Sign Language (ISL) movements, we suggest a novel model for deep learning named EfficientSign-ISL, which is based on the EfficientNetB0 architecture.

- **Backbone Architecture:** The foundation of EfficientNetB0 is a baseline architecture that was found using neural architecture search and enhanced by compound scaling. Using a compound coefficient ϕ, EfficientNet scales network dimensions consistently instead of wildly expanding the network size.

  - Depth (d) — number of layers
  - Width (w) — number of channels
  - Resolution (r) — input image size

  The scaling method can be mathematically formulated as:

$$\begin{gathered} \text{d}= \alpha^{\Phi} \\ w = \beta^{\Phi} \\ r = \gamma^{\Phi} \\ subject\ to\ \alpha.\beta^2.\gamma^2 \approx 2 \\ \alpha\beta\gamma\succ0 \end{gathered}$$

  With a superior balance between model size and performance thanks to this compound scaling technique, EfficientNetB0 is both computationally economical and very accurate, particularly for jobs requiring little data and processing power.

- **EfficientSign-ISL Architecture:** EfficientNetB0 is employed as the feature extractor in the EfficientSign-ISL model, utilizing its pre-trained weights on ImageNet for transfer learning. We use our dataset of 10 static sign classes to fine-tune the final classification layers exclusively for ISL gesture identification.

- The model architecture can be described in the following stages:

- Input Layer: Input images resized to 224×224×3, matching EfficientNetB0's expected input size.
- Feature Extraction: Pre-trained EfficientNetB0 is used up to the final global average pooling layer to extract high-level features:

$$F=\text{EfficientNetB0}(X)$$

Where, $X \in R^{224\times224\times3}$ is the input image, and $F \in R^{1280}$ is the output feature vector.

- Classification Head: A custom classification head is added on top of EfficientNetB0, which includes:
  Dropout Layer: Prevents overfitting

$$F_{drop}=\text{Dropout}(F,p=0.5)$$

Dense Layer with Softmax Activation: Predicts the probability distribution over the 10 BdSL classes

$$\hat{y}=\text{Softmax}(W\cdot F_{drop}+b)$$

where W and b are the weights and bias of the fully connected layer, and $\hat{y} \in R^{10}$ is the predicted class probability vector.

- Loss Function and Optimization: For multi-class classification, we use the categorical cross-entropy loss**,** defined as:

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

Where:

- C=10 (number of gesture classes),
- y_i is the true label (one-hot encoded),
- $\hat{y}_i$ is the predicted probability for class i.

The model is optimized using the Adam optimizer with an initial learning rate of 1×10^{-4}, incorporating techniques like early stopping and learning rate reduction on plateau to ensure convergence. The proposed model compilation processes and training process parameters are shown in Table 1.

Table 1: Parameter of the proposed model

Parameter	Value/Setting
Base Model	EfficientNetB0
Optimizer	Stochastic Gradient Descent (SGD)
Learning Rate	0.01 (1e-2)
Learning Rate Decay	Exponential Decay
Decay Steps	10,000
Decay Rate	0.9

Loss Function	Categorical Crossentropy
Evaluation Metrics	Accuracy, Precision, Recall, F1-Score
Batch Size	32 (default, modify if different)
Number of Epochs	300
Input Image Size	224 × 224 × 3
Number of Classes	10 (ISL gestures)
Activation Function	Softmax (final layer)
Data Augmentation	Rotation, Zoom, Flip, Brightness Adjustment

IV. RESULTS AND DISCUSSIONS

A. Experimental Setup

To validate the performance of the proposed EfficientSign-ISL model for Indian Sign Language recognition, a series of structured experiments were conducted using a manually curated and augmented image dataset of ISL gesture classes. To ensure that every class was fairly and consistently represented in both subsets, the data was split into two sections: 80% for training and 20% for validation and testing. Python 3.9 was used to develop the model, while TensorFlow 2.x, Scikit-learn, and the Keras API were used as deep learning frameworks to build, train, and refine the model.

- Processor: Intel Core i7
- Memory: 32 GB RAM
- Storage: 512 GB SSD
- Graphics Processing Unit: NVIDIA RTX 3080 (10GB VRAM)

This configuration enabled efficient model training, particularly beneficial during compound scaling of the EfficientNetB0 backbone.

B. Evaluation Metrics

For our work, we employed several evaluation metrics such as accuracy, precision, recall and f-score.

Accuracy: Accuracy: $\frac{Tp+TN}{Tp+TN+FP+FN}$

Precision: Precision: $\frac{TP}{TP+FP}$

Recall: Recall: $\frac{TP}{TP+FN}$

F1-Score: F1-score: 2 $\frac{Precision.Recall}{Precision + Recall}$

B. Performance of the Models

Model performance and the suggested EfficientSign-ISL were assessed based on individual contributions and personal gathering of data. A comprehensive comparison of the model's efficacy is provided in Table 2, which displays the assessment results based on accuracy, precision, recall, and F1-score.

Table 2: Performance of the models

Model	Precision (%)	Recall (%)	F1-Score (%)	Accuracy (%)
EfficientSign-ISL (Proposed)	97.54	97.10	96.96	99.38
ResNet50	88.63	87.91	88.27	89.79
MobileNetV2	84.33	83.96	84.64	85.61
InceptionV3	82.72	82.49	83.24	83.10

Our suggested EfficientSign-ISL model's performance evaluation shows that it outperforms the selected baseline architectures, ResNet50, MobileNetV2, and InceptionV3, on all essential measures, such as accuracy, precision, recall, and F1-score. EfficientSign-ISL achieved an impressive 99.38% accuracy, which was much higher than ResNet50 (89.79%), MobileNetV2 (85.61%), and InceptionV3 (83.10%). This consistent outperformance shows how stable and suitable the model identifies Indian Sign Language. When examining individual metrics, EfficientSign-ISL maintains balanced and high precision (97.54%) and recall (97.10%), effectively minimizing false positives and negatives. In contrast, while ResNet50 shows reasonably strong performance with precision and recall near 88%, the drop in MobileNetV2 and InceptionV3 suggests that these models may be less capable of capturing the nuanced features inherent in our dataset.

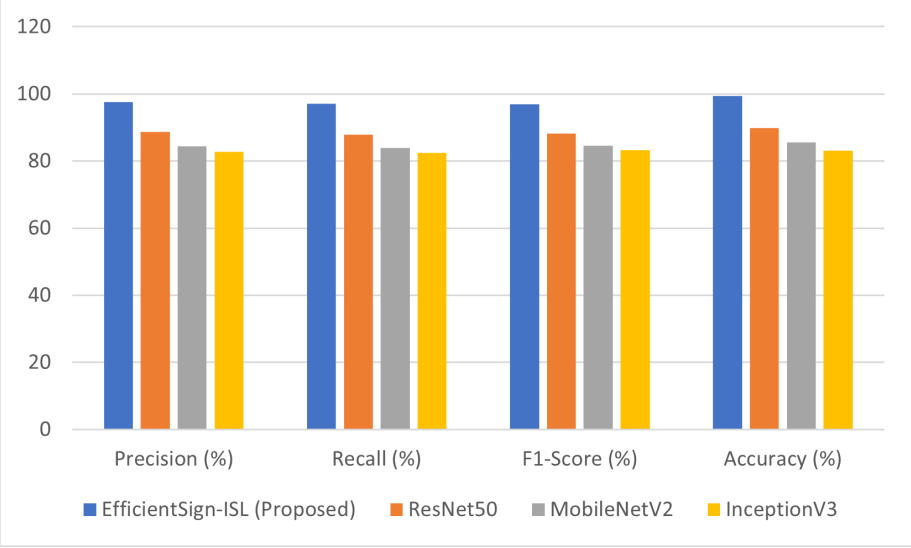

Fig. 4: Performance analysis of the models

Underfitting may be the cause of MobileNetV2 and InceptionV3's noticeably lower results. Due to architecture constraints or insufficient tweaking for this specific use, some models—especially InceptionV3—may not have sufficiently captured the complexity of the dataset during training. If a model is too simplistic or poorly trained to identify the underlying patterns, it is said to be underfitted, which results in lower accuracy and inferior generalization. While ResNet50 outperforms the other

baselines, it performs worse than EfficientSign-ISL, which might indicate a little underfitting problem or a mismatch between the network's design and the dataset's characteristics. There were no overfitting indicators in any baseline models, such as extremely high training accuracy but subpar testing outcomes, which might be a symptom of overlearning the training set.

EfficientSign-ISL's superior performance reflects its design, effectively balancing model complexity and generalization capability. The high and consistent performance across metrics suggests that the model avoids underfitting and overfitting, likely benefiting from appropriate regularization, architecture optimization, and better feature extraction tailored to Indian Sign Language. To sum up, whereas the baseline models offer fair standards, their comparatively poorer performance suggests that they cannot adjust to the dataset's features. According to the findings, EfficientSign-ISL is a strong candidate for practical use in identification systems for Bangla Sign Language as it generalizes and catches relevant features.

C. Training and Validation Loss Curve Analysis

The EfficientSign-ISL model's training and validation loss evolution over 300 epochs is depicted in Figure 5.

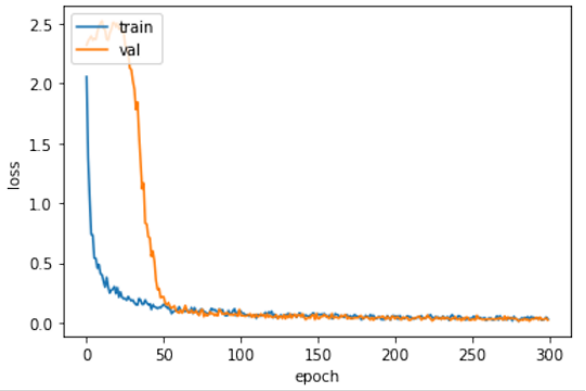

Fig. 5: Training and validation loss curve analysis

Both losses initially have a discernible disparity and begin at a higher value. However, the validation loss rapidly decreases and approaches the training loss when the model acquires significant features, especially after the first 30 epochs. According to convergence and stability, the model might generalize to unidentified data without being overfitting. The model has established that after epoch 50, the loss remains consistently inadequate, indicating a permanent optimization plateau.

D. Training and Validation Accuracy Curve Analysis

Figure 6 presents the accuracy trend during the same training period. An initial gap between training and validation accuracy is evident due to the model's early-stage learning. On the other hand, the validation accuracy rises sharply after epoch 40 and approaches the training accuracy, which stays nearly flawless beyond epoch 100. The stability and reliability of our proposed architecture in recognizing Indian sign language gestures across training and unseen validation samples are demonstrated by the close coupling of accuracy metrics.

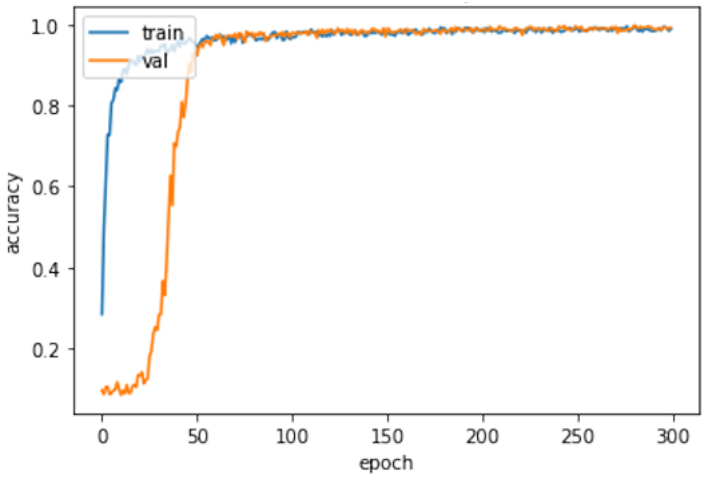

Fig. 6: Training and validation accuracy curve analysis

E. Confusion Matrix Analysis

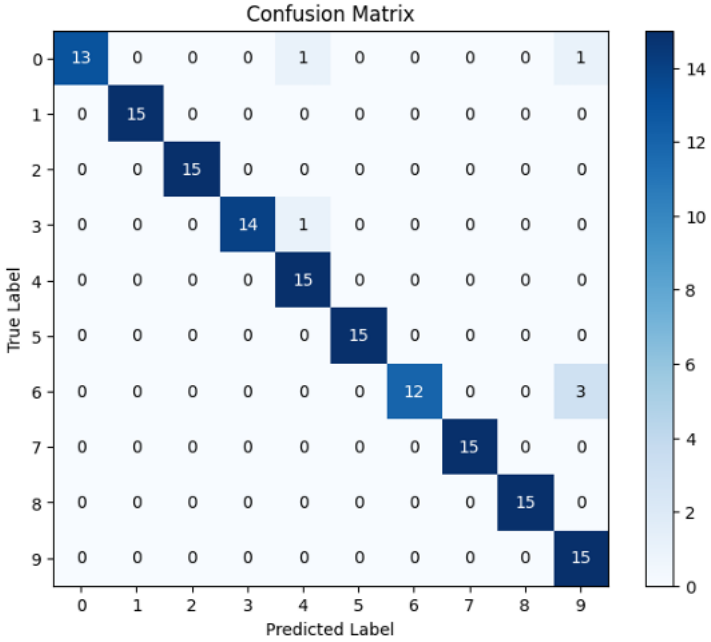

Fig. 7: Confusion matrix of the model

Finally, we evaluated our model using the confusion matrix. The confusion matrix details the model's performance in each class regarding multiclass classification, considering numerous classes. Understanding the advantages and disadvantages of a categorization model requires an interpretation of the confusion matrix. It guides the fine-tuning of the model for better performance by identifying its benefits and shortcomings. A confusion matrix is used in Figure 7 to provide a thorough perspective of the model's categorization behavior over ten gesture classes. The incredible precision of the model is demonstrated by diagonal dominance, where most predictions match their real labels. Sometimes, visually identical motions might be confused, as seen by minor misclassifications, particularly for classes 0, 3, and 6. The overall pattern nevertheless supports the classifier's ability to discriminate Indian sign language signals accurately.

F. Class-wise Performance Analysis

Table 3 summarizes the proposed EfficientSign-ISL model's class-wise performance metrics, highlighting its ability to maintain strong recognition performance across all individual Indian Sign Language classes. Most classes achieved high precision and recall, with F1 Scores generally falling above 0.90, indicating reliable and consistent predictions.

Table 3: Class-wise performance analysis

Class	Precision	Recall	F1-Score
Class 0	0.96	0.89	0.92
Class 1	1.00	0.97	0.98
Class 2	0.95	1.00	0.97
Class 3	0.91	0.88	0.89
Class 4	0.92	0.95	0.93
Class 5	0.89	0.91	0.90
Class 6	0.90	0.84	0.87
Class 7	1.00	0.96	0.98
Class 8	0.94	0.92	0.93
Class 9	0.86	0.93	0.89

Notably, F1 scores of 0.98 or higher indicate outstanding performance in Classes 1, 2, and 7. Figure 8 shows that the model is quite specific in recognizing these motions, perhaps because of their unique forms and the large number of training instances. In contrast, the third, sixth, and ninth classes performed slightly imperfectly. Class 6 had an F1-score of 0.87 and a recall of 0.84, for instance. This suggests that although the model correctly recognized the majority of relevant cases, some samples could have been incorrectly categorized, most likely due to intra-class variability or visual resemblance with other classes.

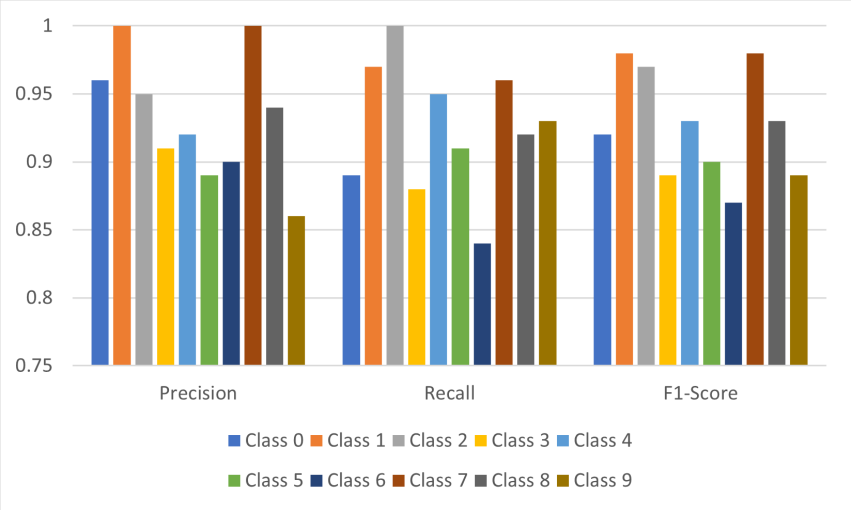

Fig. 8: Class-wise performance analysis of the proposed model

To achieve this level of resilience, data augmentation techniques including random shifting, zooming, and brightness correction were involved. Additionally, to improve the model's ability to generalize to several picture situations, such as angle, illumination, and orientation, these augmentations boosted dataset's

variety. As a result, the model managed even grayscale input images efficiently, which is particularly important for deployment on low-resource or grayscale camera devices.

V. CONCLUSION AND FUTURE WORK

The proposed *EfficientSign-ISL* framework focuses on recognizing static hand gestures in Indian Sign Language (ISL) by employing a transfer learning approach. Prioritizing both accuracy and low computational demand, the system uses EfficientNetB0, which utilizes a compound scaling method to ensure performance efficiency—making it suitable for mobile and embedded platforms. The model's effectiveness was validated through careful data handling, including structured training-validation splits and evaluations against well-known architectures like ResNet50, MobileNetV2, and InceptionV3. Although the current version targets static image classification and supports a limited set of ISL gestures, the results indicate strong potential for further development.

Future work should incorporate dynamic gesture sequences captured from video data to enable the recognition of complete phrases or sentences. This extension would benefit from integrating temporal models such as bidirectional LSTMs, GRUs, or Transformer-based networks for processing sequential information. Furthermore, a more inclusive and thorough dataset that considers hand orientations, lighting variations, geographical variations in gestures, and many signers is necessary. Collaboration with members of the deaf community and sign language experts will be essential to guarantee that the system preserves linguistic and cultural peculiarities while improving accessibility. Overall, *EfficientSign-ISL* marks a meaningful step toward inclusive communication technologies and creates opportunities for continued progress in sign language recognition tailored to ISL.

REFERENCES

1. Powell, D. S. W., & McCoy, R. G. (2025). Hearing care within diabetes care: Strategies for direct inclusion. *Clinical Diabetes*, *cd250002*.
2. Batts, S., Pham, N., Tearney, G., & Stankovic, K. M. (2025). The state of high-resolution imaging of the human inner ear: A look into the black box. *Advanced Science*, *e00556*.
3. Penobsm, D. M., & Gomwalk, N. V. (2025). Effects of sign language interpreters' services on academic performance of students with hearing impairment University of Jos. *BW Academic Journal*.
4. Dritsas, E., Trigka, M., Troussas, C., & Mylonas, P. (2025). Multi-modal interaction, interfaces, and communication: A survey. *Multimodal Technologies and Interaction, 9*(1), 6.
5. Saleed, F. M., Shabbir, M., Poudyal, S., & Amalanathan, G. M. (2025). A systematic review of Indian sign language detection systems: Evaluating the evolution, techniques, and challenges in ISL recognition technology. In *Tools for Promoting Independent Living Skills in Individuals with Disabilities* (pp. 297–326).
6. Rastogi, U., Mahapatra, R. P., & Kumar, S. (2025). Advancements in machine learning techniques for hand gesture-based sign language recognition: A comprehensive review. *Archives of Computational Methods in Engineering*, 1–38.
7. Khan, A., Jin, S., Lee, G.-H., Arzu, G. E., Nguyen, T. N., Dang, L. M., Choi, W., & Moon, H. (2025). Deep learning approaches for continuous sign language recognition: A comprehensive review. *IEEE Access*.
8. Navin, N., Farid, F. A., Rakin, R. Z., Tanzim, S. S., Rahman, M., Rahman, S., Uddin, J., & Karim, H. A. (2025). Bilingual sign language recognition: A YOLOv11-based model for Bangla and English alphabets. *Journal of Imaging, 11*(5), 134.
9. Rubaiyeat, H. A., Youssouf, N., Hasan, M. K., & Mahmud, H. (2025). BDSLW401: Transformer-based word-level Bangla sign language recognition using relative quantization encoding (RQE). *arXiv Preprint arXiv:2503.02360*.
10. Ahmed, S., Hossain, M., Amit, A. A., Tahsin, M., Mahmud, M., & Kaiser, M. S. (2025). Real-time Bangla sign language detection and recognition using YOLOv10. In *International Conference on Trends in Computational and Cognitive Engineering* (pp. 383–400). Springer.

11. Rafi, A. M., Nawal, N., Bayev, N. S. N., Nima, L., Shahnaz, C., & Fattah, S. A. (2019). Image-based Bengali sign language alphabet recognition for deaf and dumb community. In *2019 IEEE Global Humanitarian Technology Conference (GHTC)* (pp. 1–7). IEEE.
12. Ahammad, K., Shawon, J. A. B., Chakraborty, P., Islam, M. J., & Islam, S. (2021). Recognizing Bengali sign language gestures for digits in real time using convolutional neural network. *International Journal of Computer Science and Information Security (IJCSIS), 19*(1).
13. Patel, U., & Ambekar, A. G. (2017). Moment based sign language recognition for Indian languages. In *2017 International Conference on Computing, Communication, Control and Automation (ICCUBEA)* (pp. 1–6). IEEE.
14. Biyani, D., Doohan, N. V., Rode, M., & Jain, D. (2023). Real time sign language recognition using YOLOv5. In *2023 IEEE World Conference on Applied Intelligence and Computing (AIC)* (pp. 582–588). IEEE.
15. Singh, A., Singh, S., & Mittal, A. (2023). A novel deep learning approach for recognition of Hindi vowels of Indian sign language. In *International Conference on Systems, Control and Automation* (pp. 463–474). Springer.
16. Hamza, H. M., & Wali, A. (2023). Pakistan sign language recognition: Leveraging deep learning models with limited dataset. *Machine Vision and Applications, 34*(5), 71.
17. Mirza, M. S., Munaf, S. M., Azim, F., Ali, S., & Khan, S. J. (2022). Vision-based Pakistani sign language recognition using bag-of-words and support vector machines. *Scientific Reports, 12*(1), 21325.
18. Prova, N. N. I. (2024). Improved solar panel efficiency through dust detection using the InceptionV3 transfer learning model. In *2024 8th International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)* (pp. 260–268). IEEE.
19. Prova, N. N. I. (2024). Enhancing fish disease classification in Bangladeshi aquaculture through transfer learning, and LIME interpretability techniques. In *2024 4th International Conference on Sustainable Expert Systems (ICSES)* (pp. 1157–1163). IEEE.