Grenze International Journal of Engineering and Technology, June Issue

Unlocking Communication: ASL Fingerspelling Recognition using Deep Learning Algorithms

Asmita Manna[1], Vaibhavi Pawar[2], Swayam Patil[3], Sairaj Pawar[4] and Suhani Patil[5]
[1-5]Pimpri Chinchwad College of Engineering, Department of Computer Engineering, Pune, India
Email: asmita.manna@pccoepune.org, {vaibhavi.pawar21, swayam.patil211, sairaj.pawar21, suhani.patil21}@pccoepune.org

***Abstract*— Fingerspelling in American Sign Language (ASL) is an important means of communication for the Deaf and hard-of-hearing community, but limitations in awareness of ASL as a form of communication and other technological factors lead to communication limitations. The project seeks to improve ASL fingerspelling identification through deep learning techniques for easier, more accessible communication. Evaluation is conducted on five advanced convolutional neural network architectures-VGG16, ResNet50, MobileNetV2, DenseNet121, and InceptionV3 with a very large dataset consisting of more than three million hand gesture samples from a diverse background. DenseNet121 among the other networks achieved the highest validation accuracy of 87.86% and provided a good generalization and reliability in real-life applications. While MobileNetV2 performs well in training, it struggles with overfitting, and InceptionV3 maintains a good balance between precision and recall. Overall, these findings support and highlight the possibility for AI-facilitated ASL recognition to break communication barriers and help with assistive technology, providing inclusion all over the world with individuals reliant on sign language in communication.**

***Index Terms*—R-CNN, RNN, TensorFlow, ASL, MediaPipe, VGG16, ResNet50, MobileNetV2, DenseNet121, and InceptionV3.**

I. INTRODUCTION

Fingerspelling is significant for people who experience communication disabilities, in particular, who are deaf or hard of hearing. Fingerspelling is an important alternative or additional method of communication for this population when communicating with others to express words and ideas [1][2]. It can enhance their ability to communicate thoughts and have meaningful conversations, and therefore contribute to a better, more inclusive, and accessible communication environment [3]. Furthermore, fingerspelling is an excellent educational tool for individuals with communication disabilities learning sign languages, as fingerspelling provides a way for these individuals to learn how to spell and understand words, which can also further their overall language development and communication with both deaf and hearing individuals. Therefore, fingerspelling is an important way to support individuals with communication disabilities by giving them another option for effective expression and communication. Figure 1. depicts American Sign Language (ASL) hand shapes for the English alphabet and numbers. People with communication impaired disabilities are limited because they may not be aware of or understand fingerspelling [4].

Grenze ID: 01.GIJET.11.2.592_1

Limited communication abilities can limit their engagement and development in social, educational and professional situations, where communication is necessary. Outside of a professional perspective, struggle to convey messages using fingerspelling can evoke a sense of isolation, where others may be less likely to engage in conversations, or the individual may experience exclusion due to perceived communication challenges. This paper considers how deep learning models can address the communication barriers present in the ASL fingerspelling.

The work answers key research questions: How well can various deep learning models classify ASL fingerspelling? Which model balances accuracy and efficiency for portable devices like smartphones? To address these challenges and explore effective solutions, the study focuses on evaluating deep learning models for ASL fingerspelling recognition. By leveraging models such as VGG16, ResNet50, MobileNetV2, DenseNet121, and InceptionV3, we achieved validation accuracies of 86.00%, 68.99%, 79.05%, 87.86%, and 86.57%, respectively. DenseNet121 was identified as the most well-rounded model, offering strong generalization with high validation accuracy. The paper is organized as follows: Section II presents a literature review, Section III presents the proposed architecture, Section IV details the methodology, and Section V concludes with results and future work.

Figure 1. American Sign Language (ASL) hand gestures for alphabet letters and numbers

II. LITERATURE REVIEW

To obtain a comprehensive understanding of advancements in ASL, a comparative analysis of existing works is conducted. This analysis highlights the strengths and limitations of prior studies, providing a foundation for the development of a novel methodology. The reviewed papers span publications from 2015 to 2024 [1] [15]. Table I. presents a detailed comparison of various approaches to sign language recognition.

TABLE I. COMPARISON OF DIFFERENT APPROACHES FOR SIGN LANGUAGE RECOGNITION

Paper ID	Dataset Used	Algorithms Used	Strengths	Weaknesses
Shi et. al., 2018 [1]	Dataset includes fingerspelling segments extracted from ASL videos available on YouTube	Attention-based recurrent encoder-decoders, CTC-based approaches.	1)CTC-Based Recognizer 2)Accuracy - 42%	1)Incomplete 2)Ambiguous 3)Unspecified 4)Unclear 5)Inadequate
Ameen et. al., 2017 [2]	The dataset comprises over	CNN	1)Innovative Approach 2)Empirical Evaluation	1) The dataset consists of 31 hand signs, covering all

	60,000 images, with each user contributing more than 500 images for each specific sign.		3)Performance Improvement	fingerspelled letters and numbers, except for J and Z, which require motion-based analysis for accurate classification.
Rahman et. al., 2019 [3]	Four datasets: Massey University Gesture, Turkish sign language digits, ASL fingerspelling from the University of Surrey, and ASL Alphabet with 87,000 samples.	SLRNet-8, a CNN	1) The proposed CNN model enhances ASL recognition accuracy by 9% compared to several well-established methods.	1)High computational complexity. 2)Resource-intensive.
Shivashankara et. al., 2018 [4]	ASL static gestures and video gestures	1)HSV and YCbCr Colour Model	1)Quantitative Results 2)Comparative Analysis 3)Categorised Recognition Rates	1)Limited Generalizability 2)Incomplete Future Directions 3)Omitted Segmentation and Background Removal
Hao Dong et. al., 2024 [5]	ASL fingerspelling dataset with 24 alphabetic categories and complex background.), capturing more than 500 RGB images for each category	Two-path convolutional network	1)Accuracy of 99.52% on the test set. 2)Proposed method can be applied to small terminals, such as mobile phones.	1) Tested the proposed method on only one dataset,. 2)Does not provide a comprehensive evaluation of the proposed method.
Khartheesvar et. al., 2023 [6]	INCLUDE,INCLUDE-50	The feature extraction process utilizes the MediaPipe Holistic pipeline, followed by a Long Short-Term Memory (LSTM) network for sequence modeling.	1)High accuracy (94.8% - INCLUDE-50, 87.4% - INCLUDE). 2)Effective MediaPipe and LSTM integration.	1)Evaluation focused on isolated word recognition, not broader communication contexts.
Mohsin et. al., 2024 [7]	ASL hand gestures collected from MNIST Database	1)ResNet50 2)MobileNetV2 3)InceptionV3 4)VGG16 5)CNN	1)Comprehensive Analysis 2)Use of Established Dataset 3)Transfer Learning Application	1)Lack of Real-world Testing 2)Dependency on Existing Models 3)Limited Model Comparison
Cui et. al., 2018 [8]	12,000 gesture images	CaffeNet architecture	1)Fine-tuning 2)High Accuracy 3)Performance Specific	1)Incomplete Discussion of Results 2)Limited Generalizability 3)Limited Scope
Xiao et. al., 2022 [9]	Sign Language Digits Dataset and Sign Language MNIST Dataset.	SLR-CapsNet, a Capsule Network architecture, is employed for sign language digits and alphabet recognition tasks.	1)Capsule Network's higher generalisation. 2)Robust routing iterations.	1)Limited exploration of broader sign language translation challenges.
Lee et. al., 2021 [10]	The model is trained with 2600 samples	Long Short-Term Memory Recurrent Neural Network (LSTM RNN)	1)ASL Alphabet Recognition 2)Real-time Leap Motion 3)Learning Program	1)Controller variations 2)Right-hand bias 3)Limited left-hand recognition 4)Positioning fragility

Kakizaki et. al., 2024 [11]	1)JSL Dataset 2) LSA64 Dataset	1)MediaPipe 2)Random Forest(RF) 3)SupportVector Machine(SVM)	1)Proposes an effective method for dynamic JSL recognition 2)The proposed approach outperforms existing methods by achieving higher accuracy.	1)Does not discuss the limitations and challenges of the proposed method 2) The proposed method lacks a comparison with state-of-the-art approaches for dynamic sign language recognition.
Pooya Fayyazsanavi et. al., 2024 [12]	ChicagoWild, ChicagoWild+	Encoder-Decoder Transformer Model	1)Transformer-based architecture. 2)Two-stage inference. 3)Novel word length prediction.	1)Limited nuance capture. 2)Data sensitivity.
Alsharif et. al., 2023 [13]	87,000 images of the ASL alphabet hand gestures	1)AlexNet 2)ConvNeXt 3)EfficientNet 4)ResNet-50 5)VisionTransformer	1)Clear Presentation of Results 2)Highest Accuracy - 99.988% 3)Acknowledgment of Potential Limitations	1)Absence of External Validation 2)Potential Bias in Dataset Resizing
Kumwilaisak et. al., 2022 [14]	"ChicagoFSWild" dataset	AlexNet and LSTM	1)Proposed 50% Accuracy 2)Naturally Taken Fingerspelling Videos	1)Iterative 2)Recognition 3)Performance
Ali et. al., 2022 [15]	8,000 images, divided into 40 classes representing numbers, letters, and words, with 200 images in each class	YOLO-based model	1)Accuracy: 98.01% 2)Loss: 1.3 3)Recall: 0.96 4)F1 Score: 0.96 5)Frames per Second (fps): 28.9	1)Dataset limitations 2)Vocabulary gaps 3)Image precision

III. PROPOSED ARCHITECTURE

The proposed architecture outlines the process of recognizing ASL fingerspelling. It starts with preprocessing input images through resizing, normalization, and augmentation, followed by feature extraction using transfer learning models like VGG16, ResNet50, MobileNetV2, DenseNet121, and InceptionV3. The extracted features are used for model training with custom layers and loss functions. Finally, predictions are evaluated based on accuracy, precision, recall, and F1-score. Figure 2. represents Workflow of ASL Fingerspelling Recognition System.

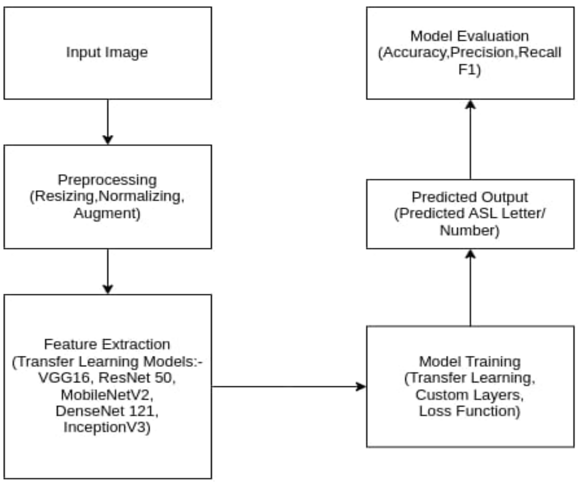

Figure 2. Workflow of ASL Fingerspelling Recognition System

IV. METHODOLOGY OVERVIEW

The paper investigates the performance of various Convolutional Neural Network (CNN) models for the task of classifying fingerspelling images in American Sign Language (ASL). The methodology involves the following keys

A. About Dataset

Our dataset is the dataset provided by Google in a Kaggle Competition. The dataset consists of over three million fingerspelled characters collected from more than 100 Deaf signers captured via the selfie camera of a smartphone with a variety of backgrounds and lighting conditions. The ASL Fingerspelling Recognition Corpus (version 1.0) contains hand and facial landmarks extracted using MediaPipe version 0.9.0.1. [6] from videos featuring fingerspelled phrases, addresses, phone numbers, and URLs by Deaf signers.teps:

B. Data Preprocessing

The dataset consists of images representing ASL fingerspelling for different letters. These images are preprocessed, including resizing to a fixed dimension (224x224), normalization, and data augmentation (rescaling, shear, zoom, horizontal flipping) to enhance the robustness of the models [7] [8].

C. Transfer Learning Models

The study leverages transfer learning by using pre-trained CNN models as feature extractors [9]. These models are pre-trained on the ImageNet dataset and are fine-tuned on the ASL dataset. The following CNN architectures are used:

- VGG16: Known for its deep architecture with 16 layers, primarily used for large-scale image classification tasks.
- ResNet50: Utilizes residual connections to facilitate the training of deep networks by mitigating the vanishing gradient issue.
- MobileNetV2: Designed for mobile and resource-constrained environments, emphasizing efficiency with depthwise separable convolutions.
- DenseNet121: Features dense connections between layers, improving gradient flow and reducing the risk of vanishing gradients.
- InceptionV3: Integrates convolution operations of different kernel sizes, enabling the model to extract features across multiple spatial scales.

D. Model Training and Fine-Tuning

Each model is fine-tuned by adding custom layers on top of the pre-trained base, including fully connected layers, batch normalization, and dropout layers. The models are trained using the ASL dataset with categorical cross-entropy as the loss function along with accuracy, precision, and recall as performance metrics.

V. RESULTS AND CONCLUSION

After analyzing these multiple models that were evaluated through the training process and evaluation process including VGG, ResNet50, MobileNetV2, DenseNet121, and InceptionV3, there are some notable performance measures. MobileNetV2 performed strong through the training phase, but showed a large drop off in validation performance, indicating it most likely was overfitting [10].

DenseNet121 showed the best balance in performance measures when comparing training and validation measures [6].

Besides DenseNet121, InceptionV3 performed comparably between training and validation measures and resulted in having the lowest validation loss, indicating that it exhibited good predictions based on the balance of precision and recall measures.

VGG and ResNet50 both performed well in the training process, but had validation measures that indicated they needed more tuning for improved generalization.

Figure 3. represents Training Accuracy and Loss of DenseNet121 Model and Figure 4. represents Training Accuracy and Loss of InceptionV3 Model.

In conclusion, DenseNet appears to be the best-structured model, with a high performance in terms of training with validations. DenseNet can be recommended for those applications that really need a robust generalization, while MobileNet and InceptionV3 can be sought after for tasks that emphasize training accuracy and the lowest validation loss, respectively. Table II. compares training and validation accuracy of different models.

A. DenseNet121 Model

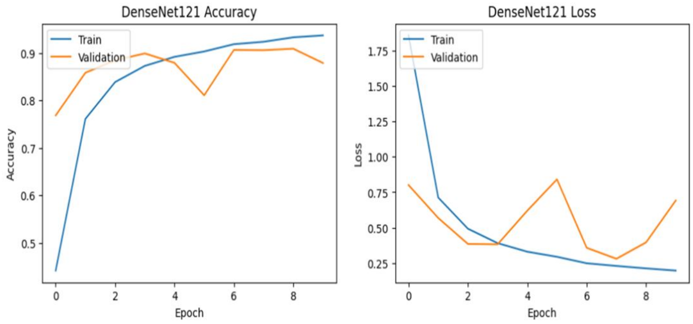

Figure 3. Training Accuracy and Loss of DenseNet121 Model

B. InceptionV3 Model

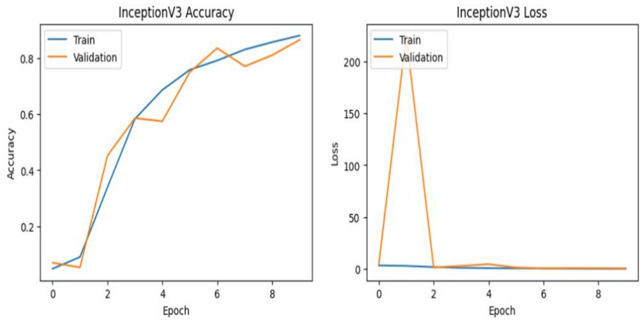

Figure 4. Training Accuracy and Loss of InceptionV3 Model

TABLE II. COMPARISON OF TRAINING AND VALIDATION ACCURACY OF DIFFERENT MODELS

Model	Training Accuracy	Validation Accuracy
VGG 16	90.25%	86.00%
ResNet 50	91.60%	68.99%
MobileNet V2	95.94%	79.05%
DenseNet 121	93.93%	87.86%
Inception V3	87.81%	86.57%

REFERENCES

[1] Shi, Bowen, Aurora Martinez Del Rio, Jonathan Keane, Jonathan Michaux, Diane Brentari, Greg Shakhnarovich, and Karen Livescu. "American sign language fingerspelling recognition in the wild." In 2018 IEEE Spoken Language Technology Workshop (SLT), pp. 145-152. IEEE, 2018.

[2] Ameen, Salem, and Sunil Vadera. "A convolutional neural network to classify American Sign Language fingerspelling from depth and colour images." Expert Systems 34, no. 3 (2017): e12197.

[3] Rahman, Md Moklesur, Md Shafiqul Islam, Md Hafizur Rahman, Roberto Sassi, Massimo W. Rivolta, and Md Aktaruzzaman. "A new benchmark on american sign language recognition using convolutional neural networks." In 2019 International Conference on Sustainable Technologies for Industry 4.0 (STI), pp. 1-6. IEEE, 2019.Harvard.

[4] Shivashankara, Ss, and S. Srinath. "American sign language recognition system: an optimal approach." International Journal of Image, Graphics and Signal Processing 10, no. 8 (2018): 18.

[5] Hao Dong, Yushuai Wang, Xiaolong Bu, Junming Zhang and Haitao Wu. "Sign language recognition based on dual-path background erasure convolutional neural network." (2024).

[6] Khartheesvar, G., Kumar, M., Yadav, A.K. et al. Automatic Indian sign language recognition using MediaPipe holistic and LSTM network. Multimed Tools Appl (2023). https://doi.org/10.1007/s11042-023-17361-y.

[7] Mohsin, Sanaa, Baraa Wasfi Salim, Ashty Kamal Mohamedsaeed, Banar Fareed Ibrahim, and Subhi RM Zeebaree. "American Sign Language Recognition Based on Transfer Learning Algorithms." International Journal of Intelligent Systems and Applications in Engineering 12, no. 5s (2024): 390-399.

[8] Cui, Qi, Zhili Zhou, Chengsheng Yuan, Xingming Sun, and QM Jonathan Wu. "Fast American Sign Language Image Recognition Using CNNs with Fine-tuning." Journal of Internet Technology 19, no. 7 (2018): 2207-2214.

[9] Xiao, Hongwang & Yang, Yun & Yu, Ke & Tian, Jiao & Cai, Xinyi & Muhammad, Usman & Chen, Jinjun. (2022). Sign language digits and alphabet recognition by capsule networks. Journal of Ambient Intelligence and Humanized Computing. 13. 10.1007/s12652-021-02974-8.
[10] Lee, Carman KM, Kam KH Ng, Chun-Hsien Chen, Henry CW Lau, S. Y. Chung, and Tiffany Tsoi. "American sign language recognition and training method with recurrent neural network." Expert Systems with Applications 167 (2021): 114403.
[11] Kakizaki, Manato, Abu Saleh Musa Miah, Koki Hirooka, and Jungpil Shin. "Dynamic Japanese Sign Language Recognition Throw Hand Pose Estimation Using Effective Feature Extraction and Classification Approach." Sensors 24, no. 3 (2024): 826.
[12] Pooya Fayyazsanavi, Negar Nejatishahidin, Jana Košecká; Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV) Workshops, 2024, pp. 1120-1130
[13] Alsharif, Bader, Ali Salem Altaher, Ahmed Altaher, Mohammad Ilyas, and Easa Alalwany. "Deep learning technology to recognize the American sign language alphabet." Sensors 23, no. 18 (2023): 7970.
[14] Kumwilaisak, Wuttipong, Peerawat Pannattee, Chatchawarn Hansakunbuntheung, and Nattanun Thatphithakkul. "American Sign Language Fingerspelling Recognition in the Wild with Iterative Language Model Construction." APSIPA Transactions on Signal and Information Processing 11, no. 1 (2022).
[15] Ali, AL-SHAHEEN, Mesut ÇEVİK, and Alzubair ALQARAGHULI. "American sign language recognition using the YOLOv4 method." International Journal of Multidisciplinary Studies and Innovative Technologies 6, no. 1 (2022): 61-65.