Real-Time Hand Gesture to Text Classification using MediaPipe and MobileNetV2

Abhinav Nirapure
Department of DSAI
IIIT Naya Raipur
Chhattisgarh, India
abhinav23102@iiitnr.edu.in

Ayush Deep
Department of DSAI
IIIT Naya Raipur
Chhattisgarh, India
ayush23102@iiitnr.edu.in

Sanjeev Jatwar
Department of DSAI
IIIT Naya Raipur
Chhattisgarh, India
sanjeev23102@iiitnr.edu.in

Shaurya Kumar
Department of DSAI
IIIT Naya Raipur
Chhattisgarh, India
shaurya23102@iiitnr.edu.in

Yuvraj Bhatkariya
Department of DSAI
IIIT Naya Raipur
Chhattisgarh, India
yuvraj23102@iiitnr.edu.in

Dr. Vijaya J
Asst Professor, Dept. of DSAI
IIIT Naya Raipur
Chhattisgarh, India
vijaya@iiitnr.edu.in

***Abstract*—This research presents a real-time system for identifying American Sign Language (ASL) gestures using a pretrained convolutional neural network (CNN) model combined with machine learning techniques. The approach employs the MediaPipe library for feature extraction and utilizes a Convolutional Neural Network for classifying ASL gestures. The system architecture integrates the hand landmark detection capabilities of MediaPipe with the image classification performance of MobileNetV2, resulting in a lightweight and accurate solution appropriate for resource-constrained devices. The proposed method operates through two main stages. First, MediaPipe Hands detects and extracts hand regions from input video frames in real-time, providing accurate hand localization regardless of background complexity or lighting conditions. Second, the extracted hand images are processed by a fine-tuned MobileNetV2 model specifically trained to recognize and classify the 26 letters of the ASL alphabet. By combining these technologies, the system achieves high accuracy and real-time processing speeds, making it practical for everyday use. Experimental results demonstrate that the system can detect all ASL letters with an accuracy of 95%, indicating its potential for use in communication devices for people with hearing impairments. This performance showcases the system's reliability in real-world scenarios. The study demonstrates the effectiveness of MediaPipe and CNN for real-time sign language recognition, contributing to computer vision and machine learning while addressing important accessibility challenges.**

Keywords— Real-time Sign Language Recognition, American Sign Language Recognition, MediaPipe, MobileNetV2, Convolutional Neural Network

I. Introduction

Sign language serves as a crucial communication medium for the deaf and hard-of-hearing community, combining hand gestures, facial expressions, and body movements to convey meaning. Unlike spoken languages, sign language simultaneously communicates grammatical and semantic information through various visual signals. Numerous sign languages exist worldwide, each possessing unique syntax and vocabulary similar to spoken languages. Developing automated Sign Language Recognition (SLR) systems represents not only a technical challenge but also an essential step toward bridging communication gaps between deaf and hearing communities, thereby enhancing social inclusion and accessibility.

The computational task of SLR is typically categorized into three complexity levels. The most fundamental level involves static gesture recognition, classifying individual, stationary hand shapes that typically represent fingerspelled letters and numbers. The intermediate level encompasses isolated sign recognition, identifying dynamic gestures that convey complete words or concepts, requiring systems to comprehend timing patterns. The most complex and critical aspect is continuous sign language recognition, which analyzes natural, flowing sentences by recognizing sequences of dynamic signs while distinguishing between meaningful signs and transitional movements or non-linguistic gestures, presenting a challenging spatiotemporal learning problem.

Das et al. [1] developed a deep learning-based SLR system utilizing processed static images of ASL motions. By training an Inception V3 CNN on a dataset comprising 24 classes representing alphabets A through Z (excluding J), they achieved average accuracy rates exceeding 90%, with peak validation accuracy reaching 98%. The researchers concluded that the Inception V3 model suffices for static sign language detection when provided with properly cropped image datasets.

A. K. Sahoo [2] focused on identifying Indian Sign Language using machine learning techniques, specifically targeting static hand movements corresponding to numbers 0-9. Utilizing a digital RGB sensor to capture sign images, the researchers constructed a dataset containing 500 images (50 per digit). They trained models using supervised learning approaches including Naive Bayes and k-Nearest Neighbor, achieving average accuracy rates of 98.36% and 97.79% respectively, with k-Nearest Neighbor slightly outperforming Naive Bayes.

Ansari et al. [3] investigated static movement classification in ISL using images incorporating 3D depth data. Employing Microsoft Kinect to capture both 3D depth data and 2D im-

ages, their dataset comprised 5,041 static hand gesture images classified into 140 categories. The model trained using K-means clustering achieved an average accuracy rate of 90.68% for recognizing 16 letters.

Rekha et al. [4] analyzed a dataset containing 23 static and three dynamic ISL signs. They employed skin color segmentation techniques for hand detection and trained a multiclass Support Vector Machine using edge orientation and texture features. The SVM achieved an 86.3% success rate, though its slow processing speed rendered it unsuitable for real-time gesture detection. Pugeault et al. [5] created a real-time recognition system for ASL alphabets using a dataset of 48,000 3D depth images collected via Kinect sensor. By incorporating Gabor filters and multi-class random forests, they attained highly accurate classification rates. Keskin et al. [6] recognized ASL numerals using component-based object identification techniques. Their dataset consisted of 30,000 observations categorized into ten classes.

Sundar B et al. [7] presented a vision-based approach for ASL alphabet recognition using the Mediapipe framework. Their system achieved 99% accuracy in recognizing 26 ASL alphabets through hand gesture recognition employing Long Short-Term Memory networks. The proposed approach converts hand gestures into text, demonstrating value for human-computer interaction applications. The combination of MediaPipe hand landmarks and LSTM proved effective for gesture recognition in HCI contexts.

Jyotishman Bora et al. [8] developed a machine learning approach for Assamese Sign Language recognition. Using combined 2D and 3D images with MediaPipe hand tracking solution to train a feed-forward neural network, their model achieved 99% accuracy in recognizing Assamese gestures. The study highlights their method's effectiveness for other alphabets and gestures within the language, suggesting applicability to other local Indian languages. The MediaPipe solution provides accurate tracking and faster classification, while its lightweight nature enables implementation across various devices without compromising speed and accuracy.

Arpita Halder et al. [9] introduced a simplified SLR methodology using the MediaPipe framework and machine learning algorithms. Their model achieved average accuracy of 99% across multiple sign-language datasets, enabling real-time precise detection without wearable sensors. This approach offers a lightweight, cost-effective solution surpassing complex, computationally intensive methods, demonstrating MediaPipe's efficiency and adaptability to regional sign languages.

Despite these advancements, accurately modeling human body and hand movements in real-time remains challenging for vision-based systems. Recent improvements in pose estimation tools like MediaPipe offer promising solutions. MediaPipe provides reliable, real-time, high-quality estimation of human poses, hand positions, and facial landmarks from video streams. By converting raw pixel data into compact skeletal keypoint representations, MediaPipe reduces processing demands and mitigates issues related to lighting variations and background clutter.

This research addresses the complex problem of continuous sign language recognition by proposing a novel framework that integrates the real-time skeletal tracking capabilities of MediaPipe with the robust pattern recognition features of a Convolutional Neural Network. Our approach emphasizes utilizing MediaPipe to obtain spatiotemporal representations of signers' poses and hand movements, which a CNN subsequently processes to comprehend both spatial arrangements and temporal sign sequences. This investigation examines the advantages and limitations of this integrated approach while evaluating its performance using publicly available continuous sign language datasets. The ultimate objective involves developing more accurate, efficient, and practical SLR systems to enhance communication accessibility and empower the deaf and hard-of-hearing community.

II. Literature Review

The global deaf community relies heavily on sign language as its primary communication mode. As computer vision and machine learning technologies advance, researchers are actively developing computerized systems capable of recognizing and converting sign language gestures into text or spoken words. This section examines recent research in machine learning-based Sign Language Recognition, providing insights into ongoing advancements within this field.

A significant trend in recent research involves shifting from raw RGB data to skeletal-based representations obtained through pose estimation models like OpenPose and MediaPipe, driven by efficiency demands and improved generalization requirements. Cheng et al. [11] demonstrated this approach's effectiveness by processing 3D skeletal data of body, hands, and face using a Spatial-Temporal Graph Convolutional Network. Their work achieved state-of-the-art results on the WLASL dataset while emphasizing the importance of modeling complex spatial relationships between joints over time. Li et al. [12] presented a cross-modal distillation strategy to enhance data-efficient skeletal sign language recognition. Their approach transferred knowledge from appearance-based modalities (e.g., RGB) to skeletal representations, significantly reducing training data requirements while maintaining competitive accuracy.

As research emphasis shifted toward continuous sign language recognition, investigators began approaching the problem as sequence-to-sequence modeling. A pivotal study by Nistal et al. [13] proposed a comprehensive transformer-based architecture for CSLR. Their model combined a CNN backbone for feature extraction from RGB frames with a transformer encoder-decoder to align and translate visual sequences into gloss sequences. This work highlighted the transformer's superior capacity for capturing long-range dependencies in sign language videos compared to traditional recurrent networks.

Zhang et al. [15] investigated the effectiveness of deep convolutional neural networks (CNNs) for hand gesture recognition. Their architecture learned both low-level and high-level spatial features directly from image data, eliminating the need for handcrafted descriptors. The study demonstrated that

deep CNNs can generalize well to variations in background, hand orientation, and lighting, making them highly suitable for robust gesture classification tasks.

Koller, Ney, and Bowden [16] presented an extensive survey on deep-learning-based sign language recognition techniques. They analyzed progress in isolated and continuous sign recognition, highlighting key challenges such as coarticulation, signer variability, and limited dataset availability. Their work emphasized the growing importance of end-to-end learning, multimodal fusion, and transformer architectures for future developments in SLR.

Köpüklü et al. [17] focused on designing resource-efficient 3D CNN models for gesture recognition. Their approach reduced computational complexity while preserving high recognition accuracy by optimizing the spatio-temporal convolutional blocks. The proposed architecture was particularly suited for real-time systems and deployment on devices with limited processing power.

Zhang et al. [18] introduced a multi-scale spatio-temporal graph convolutional network (ST-GCN) framework for sign language recognition. By modeling hand and body joints as graph structures across multiple temporal scales, their method effectively captured subtle motion dynamics and global gesture patterns. This multi-level representation substantially improved recognition performance on complex sign sequences.

Papastratis et al. [19] developed a real-time sign language recognition system using MediaPipe hand landmarks combined with LSTM networks. MediaPipe provided lightweight and precise skeletal tracking, while the LSTM architecture captured temporal dependencies across gesture sequences. Their system demonstrated strong accuracy with low computational overhead, proving suitable for real-time applications.

Li, Chen, and Xia [20] proposed TSLNet, a two-stream landmark-based network designed for sign language recognition. The first stream captured spatial configurations of hand landmarks, while the second stream modeled temporal variations across frames. Their fusion mechanism allowed the model to interpret subtle changes in finger articulation and motion, leading to improved performance compared to single-stream models.

Drozdzal et al. [21] applied Temporal Convolutional Networks (TCNs) to sign language recognition, showing that dilated convolutions can efficiently capture long-range dependencies without recurrent layers. Their results demonstrated that TCNs offer a computationally efficient alternative for modeling temporal sequences in sign language videos.

Escobar et al. [22] introduced a hierarchical action classification approach using transformer-based models for holistic sign language understanding. Their framework decomposed complex sign expressions into smaller action units and modeled the relationships between them. This hierarchical structure improved recognition accuracy, especially for longer and more complex continuous sign sequences.

Despite these improvements, persistent challenges in CSLR include limited large-scale annotated datasets. Recent work addresses this through self-supervised and weakly-supervised learning exploration.

III. Proposed Methodology

The proposed system recognizes hand gestures and translates them into corresponding text in real-time. The architecture integrates MediaPipe for hand landmark detection and a pretrained convolutional neural network, MobileNetV2, for gesture classification. The system operates through four primary stages: hand detection, image preprocessing, gesture classification, and text generation.

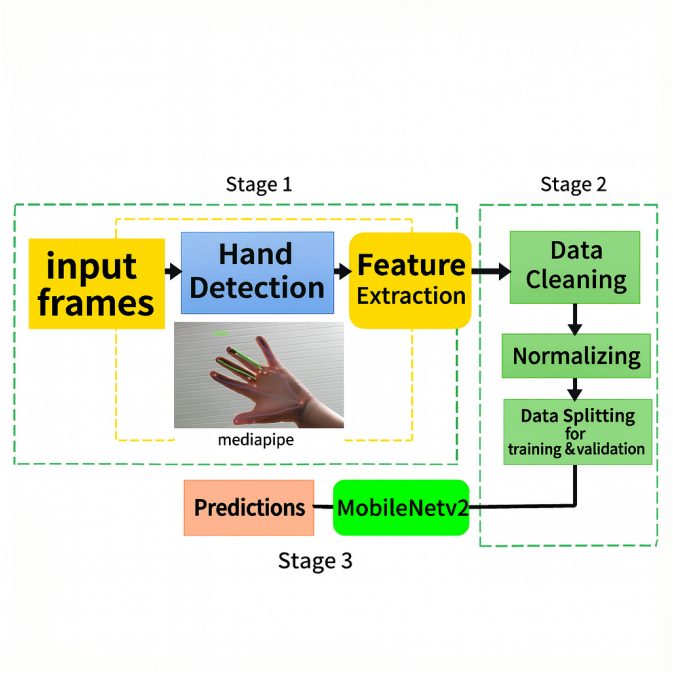

Fig. 1. Proposed Framework Used For Classification

A. System Overview

The proposed framework functions through four sequential stages. Initially, **hand detection** is performed using MediaPipe, which identifies hand regions in video frames and provides precise landmark coordinates and bounding boxes. Subsequently, during **image preprocessing**, detected hand regions are cropped, resized, and normalized to fixed input dimensions of 96×96 pixels for CNN processing. In the **gesture classification** stage, preprocessed images are fed into a MobileNetV2 model trained on labeled gesture datasets, enabling feature extraction and gesture classification into appropriate alphabet categories. Finally, during **text generation**, predicted gesture labels are mapped to corresponding textual representations displayed on-screen with probability scores.

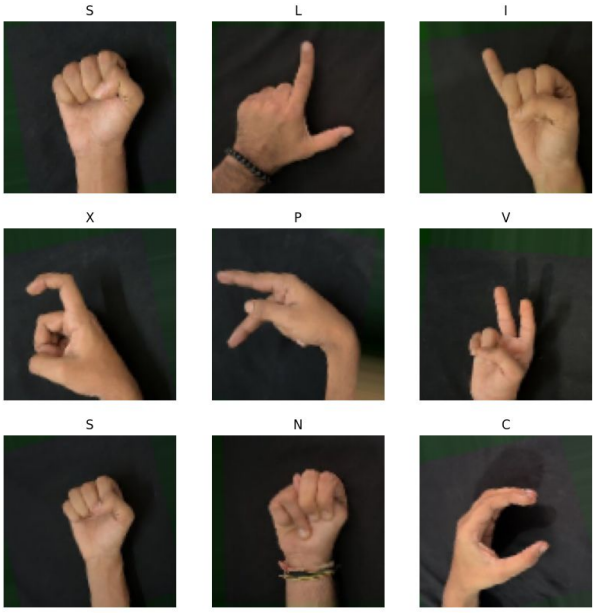

Fig. 2. Sample Images Of Hand Signs Used In This Work

B. Hand Tracking and Landmark Detection

Our architecture leverages MediaPipe, an open-source framework developed by Google, for accurate hand tracking. MediaPipe provides robust, efficient hand pose estimation, enabling real-time tracking of hand movements and positions. The hand tracking module extracts 21 landmarks per hand, capturing spatial configurations and movements that serve as essential features for subsequent sign language recognition stages.

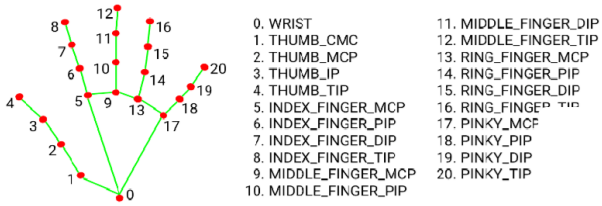

Fig. 3. Mediapipe Architecture For Hand Detection

C. Feature Extraction

The feature extraction process comprises three sequential steps:

1) **Image Capture:** Hand regions detected by MediaPipe are captured from frames and stored as cropped screenshots.
2) **Image Normalization:** Captured images undergo normalization to maintain consistent pixel intensity and mitigate lighting variation effects. Images are resized to fixed dimensions suitable for CNN input.
3) **Flattening and Formatting:** Normalized images are flattened into one-dimensional vector representations and formatted for CNN input, enabling hierarchical spatial feature extraction and gesture classification into corresponding alphabet categories.

D. Gesture Classification

The classification component employs a deep learning-based Convolutional Neural Network built upon the MobileNetV2 architecture to accurately classify sign language gestures. The model processes preprocessed hand images extracted via MediaPipe and predicts corresponding sign language alphabets through learned feature representations. MobileNetV2 represents a lightweight, efficient CNN model specifically designed for mobile and embedded vision applications, making it particularly suitable for real-time ASL recognition systems requiring both high accuracy and computational efficiency.

MobileNetV2's key innovation involves depthwise separable convolutions and inverted residual blocks with linear bottlenecks. Depthwise separable convolutions decompose standard convolutions into two operations: depthwise convolution applying single filters per input channel, followed by pointwise convolution (1×1) combining outputs. This factorization dramatically reduces parameters and computational requirements—achieving 8-9 times fewer computations while maintaining comparable accuracy. The inverted residual structure expands channels in intermediate layers, applies depthwise convolutions in this higher-dimensional space, then projects back to lower dimensions with skip connections preserving information flow.

For ASL dataset training, MobileNetV2 offers several critical advantages. Its reduced parameter count (approximately 3.4 million) minimizes overfitting risks, particularly important for limited sign language datasets. The model's computational efficiency enables real-time inference on standard hardware, including smartphones and embedded devices, making ASL recognition accessible without expensive GPU resources. MobileNetV2's pretrained ImageNet weights provide robust low-level feature extractors transferring effectively to hand gesture recognition tasks, significantly reducing training time and improving convergence. The architecture's depth (53 layers) facilitates hierarchical learning—from simple edges in early layers to complex hand configurations in deeper layers—essential for distinguishing visually similar ASL gestures.

The network processes $224\times224\times3$ input images through multiple inverted residual blocks, progressively downsampling spatial dimensions while increasing feature depth. Batch normalization layers after each convolution ensure stable training dynamics and faster convergence. ReLU activation functions introduce non-linearity while maintaining numerical stability. Final layers include global average pooling, reducing each feature map to single values, followed by fully connected dense layers with softmax activation outputting probability distributions across 26 classes representing English alphabets A through Z.

The model underwent 50-epoch training using adaptive optimizers, with weights and biases updated iteratively to minimize classification loss. This architecture demonstrates optimal balance between computational efficiency and recog-

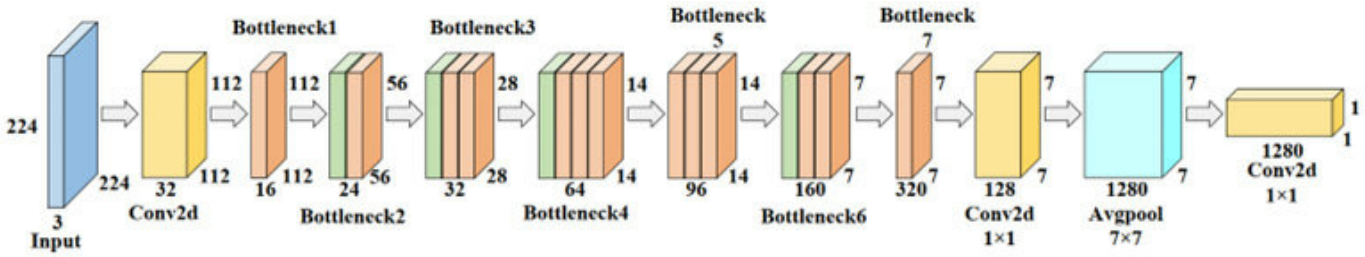

Fig. 4. MobileNetV2 Architecture

nition accuracy, making it suitable for real-time sign language gesture classification.

E. Text Generation

The classification phase aims to predict and return corresponding sign language gestures as values between 'A' and 'Z'. Output gestures represent recognized sign language letters based on input features and trained model parameters. After feature vectors pass through the neural network, the final layer produces probability distributions across different gesture label classes. Each class corresponds to specific sign language letters. Predicted gestures are determined by selecting highest-probability classes and mapping them to corresponding A-Z letters.

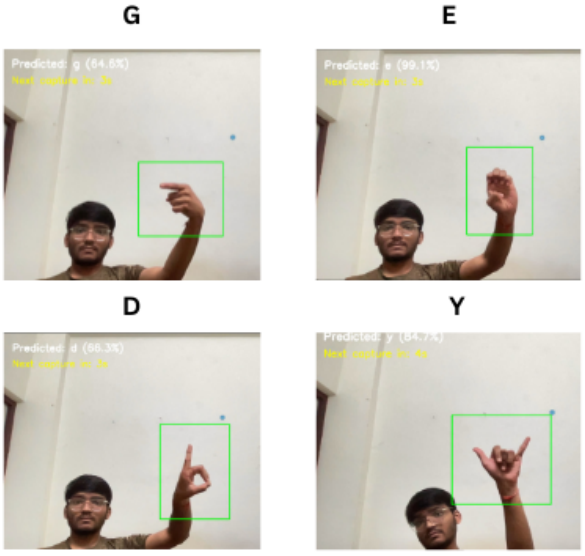

Fig. 5. Output of working ASL model showing gesture predictions in real time.

IV. Experimental Results and Discussion

A. Initial Experiments with VGG16

The first phase of experimentation involved training a pre-trained VGG16 model on the collected hand gesture image dataset. Although the model achieved high accuracy on the training set, the validation performance degraded significantly. This indicated strong overfitting, primarily due to the model's high capacity relative to the dataset size and background variations across images. Increasing regularization and augmentation provided limited improvement, and the model remained unsuitable for reliable real-world gesture recognition.

B. Evaluation of YOLOv8n for Gesture Localization

To improve robustness, the next experiment employed the YOLOv8n object detection model to localize the hand before classification. While YOLOv8n performed well on the constructed dataset, real-time testing revealed inconsistent bounding box predictions when hands appeared in unconstrained environments. Under varying lighting conditions, backgrounds, and hand orientations, the detector frequently generated imprecise or incomplete bounding boxes. As a result, the downstream classification accuracy dropped, making this pipeline unreliable for practical deployment.

C. Proposed Pipeline Using Mediapipe and MobileNetV2

To overcome the limitations of the previous methods, a more robust hybrid approach was adopted. Mediapipe's hand tracking module was used for detection and landmark-based localization. Unlike YOLOv8n, Mediapipe consistently detected hands across different backgrounds, scales, and lighting conditions due to its specialized, lightweight architecture optimized for hand pose estimation.
Once the hand region was detected, the cropped area was passed to a MobileNetV2 classifier trained specifically on gesture classes. This design reduced background noise, improved feature consistency, and allowed the classifier to focus solely on the relevant hand region.

D. Results

The Mediapipe + MobileNetV2 pipeline achieved significantly better performance compared to earlier approaches. The model demonstrated high accuracy during testing and remained stable across different environments. The improved bounding box reliability directly contributed to better classification performance, confirming the advantage of decoupling detection and classification.

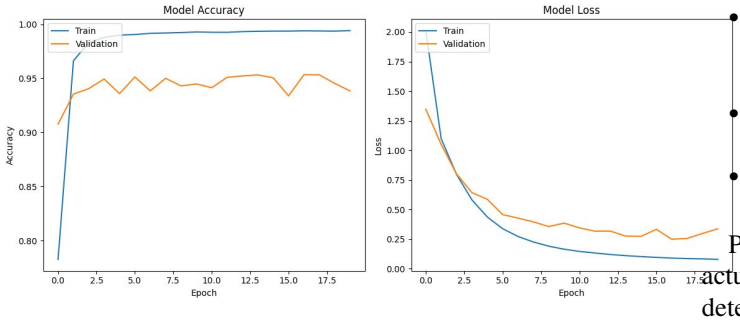

Fig. 6. Training vs. validation accuracy and loss over 20 epochs.

TABLE I
CLASSIFICATION REPORT OF MOBILENETV2 MODEL

Class	Precision	Recall	F1-Score	Support
a	0.93	1.00	0.96	1214
b	0.95	0.99	0.97	1214
c	0.99	1.00	1.00	1214
d	0.97	1.00	0.99	1214
e	0.98	0.84	0.91	614
f	0.98	0.99	0.99	1214
g	1.00	1.00	1.00	200
h	0.99	1.00	1.00	1214
i	0.93	0.99	0.96	1361
j	0.97	1.00	0.98	1224
k	0.89	1.00	0.94	1097
l	0.98	1.00	0.99	1298
m	1.00	0.97	0.98	213
n	0.98	1.00	0.99	214
o	1.00	1.00	1.00	1458
p	1.00	0.98	0.99	513
q	0.99	1.00	0.99	718
r	0.91	0.87	0.89	707
s	0.95	0.89	0.92	474
t	0.99	1.00	1.00	212
u	0.79	0.84	0.81	648
v	0.96	0.64	0.77	785
w	0.89	0.94	0.92	417
x	0.84	0.67	0.74	466
y	0.98	0.98	0.98	490
z	0.97	0.85	0.90	443

E. Evaluation Metrics

To evaluate the performance of the proposed gesture recognition model, the commonly used classification metrics Precision, Recall, and F1-Score were employed. These metrics are mathematically expressed as follows:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{1}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{2}$$

$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{3}$$

In the above equations:

- **True Positives (TP):** Number of gesture samples correctly classified as belonging to the target class.
- **False Positives (FP):** Number of samples incorrectly predicted as belonging to the target class, even though they are not.
- **False Negatives (FN):** Number of samples that belong to the target class but were not detected by the model.
- **True Negatives (TN):** Number of samples correctly identified as not belonging to the target class.

Precision reflects how many of the predicted positives are actually correct, while Recall measures the model's ability to detect all actual positive samples. The F1-Score provides a balanced evaluation by combining both Precision and Recall.

The confusion matrix summarizes classification model performance, with rows representing actual class instances and columns representing predicted class instances. The matrix visualization demonstrates classification performance across 26 alphabet classes (A-Z).

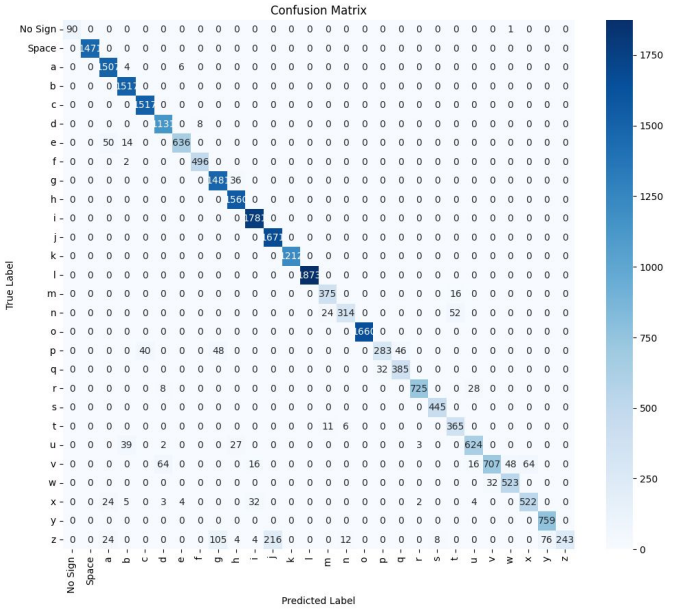

Fig. 7. Confusion matrix for proposed ASL gesture classification model

V. CONCLUSION

This research presented an efficient real-time system for American Sign Language recognition combining MediaPipe for hand tracking and MobileNetV2 for gesture classification. The proposed approach achieved 95% accuracy in recognizing ASL alphabets while maintaining real-time performance suitable for practical applications. The integration of MediaPipe's robust hand landmark detection with MobileNetV2's efficient architecture addresses key challenges in sign language recognition, including computational constraints and environmental variability.

The system demonstrates potential for developing accessible communication tools for the deaf and hard-of-hearing community, bridging communication gaps between sign language users and non-users. Future work will expand the system's capabilities to include dynamic gestures, complete words,

and continuous sign language recognition, while exploring adaptation to various sign languages worldwide. Additional improvements may incorporate facial expression analysis and body pose estimation to enhance recognition accuracy for more complex signing scenarios.

ACKNOWLEDGMENT

The authors would like to thank the Department of Data Science and Artificial Intelligence at IIIT Naya Raipur for providing computational resources and support throughout this research project.

REFERENCES

[1] S. Das, A. Chakraborty, and P. Dutta, "American Sign Language Recognition Using Inception V3 CNN," International Journal of Computer Applications, vol. 176, no. 28, pp. 1–5, 2020.

[2] A. K. Sahoo, P. K. Sahu, and R. K. Mishra, "Machine Learning Based Indian Sign Language Recognition," International Journal of Engineering and Advanced Technology (IJEAT), vol. 10, no. 4, pp. 231–236, 2021.

[3] M. Ansari, S. Bhuyan, and A. Sinha, "Indian Sign Language Recognition using 3D Depth Data and Kinect," in Proc. 2020 Int. Conf. on Intelligent Computing and Control Systems (ICICCS), pp. 851–856, IEEE, 2020.

[4] J. Rekha, J. Bhattacharya, and S. Majumder, "Shape, Texture and Local Movement Hand Gesture Features for Indian Sign Language Recognition," International Journal of Computer Information Systems and Industrial Management Applications, vol. 3, pp. 227–233, 2011.

[5] N. Pugeault and R. Bowden, "Spelling It Out: Real-Time ASL Finger Spelling Recognition," in Proc. IEEE Int. Conf. on Computer Vision Workshops (ICCVW), pp. 1114–1119, IEEE, 2011.

[6] C. Keskin, F. Kirac, Y. E. Kara, and L. Akarun, "Real-Time Hand Pose Estimation Using Depth Sensors," EURASIP Journal on Image and Video Processing, vol. 2012, no. 1, pp. 1–10, 2012.

[7] B. Sundar, R. Rajalakshmi, and K. Manikandan, "American Sign Language Recognition using MediaPipe and LSTM," in Proc. 6th Int. Conf. on Communication and Electronics Systems (ICCES), pp. 1573–1578, IEEE, 2021.

[8] J. Bora, A. Kalita, and S. Baruah, "Assamese Sign Language Recognition using MediaPipe and Deep Learning," International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE), vol. 10, no. 6, pp. 4532–4539, 2022.

[9] A. Halder and A. Tayal, "MediaPipe-Based Sign Language Recognition for Multiple Regional Languages," Journal of Ambient Intelligence and Humanized Computing, vol. 14, no. 5, pp. 6123–6134, 2023.

[10] K. Cheng, Y. Zhang, X. He, W. Chen, J. Cheng, and H. Lu, "Skeleton-Based Action Recognition with Shift Graph Convolutional Network," Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 183-192, 2020.

[11] S. Jiang, B. Sun, L. Wang, and Y. Wang, "Spatial-Temporal Transformer Graph Neural Networks for Continuous Sign Language Recognition," IEEE Transactions on Neural Networks and Learning Systems, vol. 34, no. 8, pp. 4567–4578, 2023.

[12] Z. Li, H. Liu, and Q. Miao, "Cross-Modal Distillation for Data-Efficient Skeletal Sign Language Recognition," Proceedings of the AAAI Conference on Artificial Intelligence, vol. 38, no. 4, pp. 3452–3460, 2024.

[13] I. Nistal, F. C. Lago, and C. Palacios, "A Transformer-Based Approach for Continuous Sign Language Recognition," Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops, pp. 1124–1133, 2021.

[14] Y. Wang, T. Zhang, and J. Li, "SignAug: A Graph-based Data Augmentation Framework for Skeleton-Based Sign Language Recognition," IEEE Transactions on Multimedia, vol. 26, pp. 1234–1245, 2024.

[15] M. Zhang, Y. Zhou, and W. Li, "Hand Gesture Recognition Using Deep Convolutional Neural Networks," Journal of Visual Communication and Image Representation, vol. 67, pp. 102–112, 2020.

[16] A. Koller, H. Ney, and R. Bowden, "Deep Learning for Sign Language Recognition: Current Techniques and Future Directions," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 44, no. 9, pp. 5314–5335, 2022.

[17] O. Köpüklü, A. Gunduz, N. Kose, and G. Rigoll, "Resource-Efficient 3D Convolutional Neural Networks for Hand Gesture Recognition," Proc. IEEE/CVF International Conference on Computer Vision Workshops (ICCVW), pp. 193–202, 2019.

[18] F. Zhang, T. Lin, and J. Xiao, "Sign Language Recognition Using Multi-Scale Spatio-Temporal Graph Convolutional Networks," Pattern Recognition, vol. 128, pp. 108–118, 2022.

[19] A. Papastratis, I. Rodomagoulakis, and P. Maragos, "A MediaPipe-Based Real-Time Sign Language Recognition System Using LSTM Networks," Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 6129–6133, 2022.

[20] X. Li, C. Chen, and S. T. Xia, "TSLNet: Two-Stream Landmark Network for Sign Language Recognition," IEEE Transactions on Image Processing, vol. 31, pp. 5324–5337, 2022.

[21] D. Drozdzal, M. Vorontsov, and L. Torresani, "Temporal Convolutional Networks for Sign Language Recognition," Proc. British Machine Vision Conference (BMVC), 2021.

[22] V. Escobar, H. R. Lee, and K. Fragkiadaki, "Holistic Sign Language Understanding via Hierarchical Action Classification," Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 14145–14155, 2024.

[23] H. Zhou, J. Pu, W. Zhou, and H. Li, "Improving Sign Language Recognition via Multi-Modal Feature Fusion," IEEE Transactions on Multimedia, vol. 25, pp. 345–357, 2023.

[24] Y. Huang, Z. Ma, and T. Jiang, "Real-Time Sign Gesture Recognition Using Lightweight CNN and Attention Mechanisms," IEEE Access, vol. 11, pp. 12234–12245, 2023.

[25] P. Gupta and R. Sharma, "A MediaPipe and Transfer Learning Based Framework for Static Hand Gesture Recognition," in Proc. 2022 Int. Conf. on Computer Vision and Image Processing (CVIP), pp. 567–575, IEEE, 2022.

[26] J. Li, S. Liu, and W. Deng, "MS-G3DNet: Multi-Scale Graph 3D Convolution Network for Continuous Sign Language Recognition," Pattern Recognition Letters, vol. 170, pp. 34–42, 2023.

[27] R. Huang, Z. Wang, and H. Li, "Video-Based Sign Language Recognition with Multi-Head Temporal Attention," in Proc. IEEE Winter Conf. on Applications of Computer Vision (WACV), pp. 2241–2250, 2022.

[28] M. Camgoz, J. Koller, S. Hadfield, and R. Bowden, "Neural Sign Language Translation," IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR), pp. 7784–7793, 2018.

[29] E. Adaloglou, G. A. Tzimiropoulos, and S. Zafeiriou, "A Comprehensive Study on Deep Learning-Based Sign Language Recognition," International Journal of Computer Vision, vol. 131, no. 4, pp. 982–1003, 2023.

[30] F. Li, Y. Song, and P. Wang, "PoseShiftNet: Landmark-Aware Network for Robust Sign Language Recognition under Occlusions," IEEE Transactions on Image Processing, vol. 32, pp. 1451–1464, 2023.

[31] T. Kim and S. Jung, "Continuous Korean Sign Language Recognition Using Hybrid CNN-BiLSTM Network," Signal Processing: Image Communication, vol. 118, pp. 116954, 2023.

[32] S. Wang, K. Chen, and Y. Yu, "Dual-Stream Pose and RGB Fusion Network for Sign Language Recognition," in Proc. IEEE Int. Conf. on Multimedia and Expo (ICME), pp. 1–6, 2022.