iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 **Issue:** XII **Month of publication:** December 2025

DOI: https://doi.org/10.22214/ijraset.2025.76012

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue XII Dec 2025- Available at www.ijraset.com

Real-Time American Sign Language to Speech Conversion Using CNN and Computer Vision

Sunita M. Badadhe[1], Darsh Sonsale[2], Payal Jannawar[3], Rohan Somani[4], Akshita Verma[5], Sanika Tapar[6], Piyush Kanakdande[7]
Department of Engineering, Sciences and Humanities (DESH) Vishwakarma Institute of Technology, Pune, Maharashtra, India

Abstract: ***Sign language serves as a crucial communication bridge between individuals with hearing impairments and the general population; however, approximately 97% of non-signers are unable to understand sign-based communication, resulting in significant interaction barriers. This paper introduces a machine learning and computer vision–based American Sign Language (ASL) to Speech Conversion in real time. The system employs a pre-trained Convolutional Neural Network (CNN) model for gesture classification, while OpenCV and MediaPipe frameworks are utilized for hand detection, region-of-interest (ROI) extraction, and edge tracking to enhance recognition accuracy. Identified gestures are transformed into textual characters and combined to form meaningful sentences, which are then converted into speech using the pyttsx3 text-to-speech (TTS) engine. Experimental results demonstrate an accuracy range of 96% to 99% for static gesture recognition. The system performs effectively in offline settings, making it suitable for deployment in low-connectivity environments. This work contributes to improved accessibility and inclusion for ASL users by bridging the communication gap between signers and non-signers. Future enhancements may include dynamic gesture recognition, expanded datasets for improved generalization, and support for multiple sign languages to facilitate broader real-world applications.***

Keywords: American Sign Language (ASL), Sign Language to Speech Conversion, Machine Learning, Computer Vision, Convolutional Neural Network (CNN), Real-Time Gesture Recognition, Text-to-Speech (TTS).

I. INTRODUCTION

Sign language serves as a necessary mode of communication for the deaf community. For all the advancements in technology, there remains a significant communication barrier between sign language users and those who do not understand it. Existing solutions often fall short in providing accurate and real-time translation. Spoken language is composed of articulate sounds that correspond to specific words and grammatical structures to communicate meaningful messages. Sign language utilizes visual hand and body gestures to deliver meaningful messages. Around the globe today, there are approximately 138 to 300 different types of sign languages in use.in India, there are only about 250 certified sign language interpreters for a deaf population of around 7 million people. This creates a significant challenge for teaching sign language to the deaf and hard-of-hearing community due to the limited number of interpreters available. To address this issue, we are incorporating American Sign Language (ASL) into our project. In our world today, gesture recognition technology plays a vital role in individuals who are deaf or have hearing impairments. When speech is not an option, hand gestures become an effective means of communication. Gesture recognition allows computers to capture and understand these gestures, making it possible to use hand movement as commands. This paper aims to develop a sign language-to-voice translation that bridges this communication gap. By imposing advanced machine learning algorithms and real-time processing, our project looks to provide an accurate and systematic solution to translate sign language gestures into spoken words. Figure 1 illustrates the set of American Sign Language (ASL) Alphabets

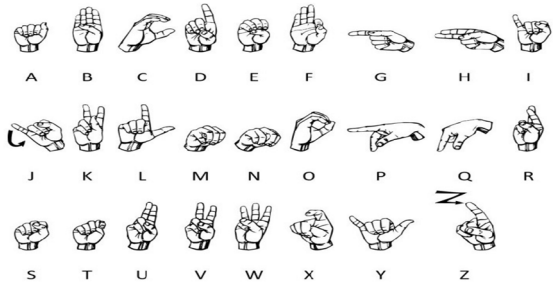


Figure 1. American Sign Language (ASL) Alphabets

II. LITERATURE REVIEW

For mute and deaf people, sign language is necessary to communicate. As, there is no common platform for communication, it becomes challenging for the people using sign language and people not using it to communicate. By transforming hand signals to spoken words, the Sign Language to Voice Translator (SL2V) helps to connect the communication gap, making their interactions more inclusive for everyone.

In this, various methods have been researched, such as computer vision-based and machine learning-based methods, primarily Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN), utilized to increase the accuracy of translations. Since much has been accomplished, challenges like the need for real-time processing and less data availability still make communication more challenging than anticipated.

This is a critical review of the most recent work on sign language translation, with special attention to gesture recognition techniques, machine learning developments, and challenges with real-time execution. Considering a literature review, we aim to shed light on important findings, point out limitations, and propose potential avenues for future research.

Due to progress in deep learning and neural network technology, sign language recognition is in growing demand [1]. Detection of American Sign Language fingerspelling without the need for specific hardware is one of the very important methods. Different neural network architectures such as LeNet-5 and MobileNetV2 are employed with the motive of higher accuracy. For a wider and practical application, these models are included in web and mobile platforms. collected methods, like horizontal poling, also provide improved results, with some achieving 100% recognition rates. Additionally, efforts are made to convert gestures into text or speech, though some need more time to process and recognize dynamic gestures. A gap remains in the recognition of body movement and complex signs. This work is centered on static sign recognition using CNNs, with the aim of laying the foundation for future enhancement in dynamic gesture recognition.

This study discusses a variety of deep learning techniques that can be combined to improve systems for sign language recognition [2]. For example, current systems, like MATLAB and Haar Cascade Classifiers-based systems, were susceptible to lighting changes and were not user-friendly. Most systems lack the ability to capture significant features of sign language, like pauses and facial expressions, resulting in reduced speed and accuracy. DNNs are more flexible and accurate. They enhance performance in gesture recognition and have immense potential in other applications, like gesture-controlled devices. Future studies will be directed toward translating identified words into text, thus making it easier to communicate. These developments have the capability to eradicate barriers to communication in all aspects of life. The area of sign language recognition is transforming from mere image processing to advanced deep learning techniques. DNNs allow for the construction of systems capable of better encoding the meaning and affective dimension of signs, rendering them more accessible and interactive.

S. Pandey et al. in "Hand Speak: Sign Language Recognition System" presented a method of sign language gesture-to-text conversion in real-time [3]. The research paper focused on enhancing accessibility for hearing-impaired individuals using Convolutional Neural Networks (CNN) along with TensorFlow and OpenCV. The article highlighted the need for real-time processing of American Sign Language (ASL) to avoid spatial variation issues and manage complex hand movements. The proposed model attained a superior accuracy level of 97.2% in ASL sentence identification. Current research is now focused on recognizing multiple sign languages while simultaneously enhancing real-time performance.

S.R.Kodandaram et al. used CNNs to develop a deep learning approach for sign language interpretation [4]. Their objective was to detect static hand gestures and translate them into text or voice. To enhance precision, they employed a horizontal voting ensemble, comparing multiple architectures, including LeNet-5, MobileNetV2, and a custom-built CNN. The model achieved an impressive accuracy of 99.8% on ASL datasets before being integrated into a web app for real-time use. Building on this success, ongoing research is now working towards recognizing dynamic gestures to make the system even more responsive.

P. P. Likhitkar et al. explored deep learning techniques to improve sign language recognition [5]. Their objective was to create a system that would be capable of identifying sign language gestures and converting them into either speech or text. By integrating CNN and LSTM structures, they created a deep learning approach for gesture recognition by detecting spatial and temporal features of hand movement. The resultant model was highly effective on detecting static and dynamic gestures after being trained on benchmark datasets. Future work involved making the model scalable such that it could be used in real-world environments and for multiple sign languages.

In their research paper, A. Chauhan et al. proposed SignSpeak, a sign language inter converter using CNN for two-way voice-to-sign and sign-to-voice translation [6]. For the precise location of both static and dynamic hand gestures, their method employed a deep learning framework including CNNs.

For stable performance under varied gestures, the model was trained on various sign language datasets. For deaf access improvement, the system also had a voice synthesis module with real-time auditory feedback. The research demonstrated that the model can be implemented in real-time systems and has quite impressive gesture detection performance. Future improvements will be conducted to improve gesture segmentation procedures and improve the multi-sign language ability of the model.

Gokulakrishnan K. and others envisioned an OpenCV- and deep-learning-based sign language translation system in their article "Sign Language to Voice Translator Using TensorFlow and TTS Algorithm" [7]. The emerging system enables translating sign gestures to verbal English by leveraging a Raspberry Pi in addition to OpenCV, TensorFlow, and a text-to-speech algorithm. The study enhances the accuracy of recognition without the use of sensors based on CNN-based classification and image preprocessing. The proposed method is an efficient and economical sign-to-speech translation method.

Om Kumar C.U. et al. in the paper "Real-Time Detection and Conversion of Gestures to Text and Speech to Sign System" has presented deep learning techniques for enabling two-way sign language interpretation [8]. The paper uses an SSD MobileNet V2 model for real-time gesture recognition and an RNN-LSTM model with Connectionist Temporal Classification (CTC) training for speech-to-sign translation. The technology enables smooth translation between text, speech, and American Sign Language (ASL) to enable the communication of speech and hearing disabled people. The research improves existing approaches by combining neural network-based classification and data processing, which improves gesture recognition and translation accuracy in general.

III. PROPOSED METHODOLOGY

The Sign Language to Voice Converter (SL2V) system is designed to convert sign language gestures into speech and text through computer vision and machine learning technology. The approach involves various major phases such as gesture acquisition, feature extraction, machine learning-based classification, and speech synthesis. The following sections give an overview of each implementation stage.

A. System Architecture

The SL2V system has an architecture that is composed of:

1. Capture hand detection using OpenCV and MediaPipe.
2. Enhance image quality, and isolate gestures. and prepare data for recognition using the Trained CNN model.
3. Text generation from recognized signs.
4. Translate text into voice using a Text-to-Speech pyttsx3 engine.

The accurate workflow of the developed Sign Language to Speech Conversion (SL2V) system, including image preprocessing, CNN classification, text generation, and speech synthesis, is illustrated in Figure 2.

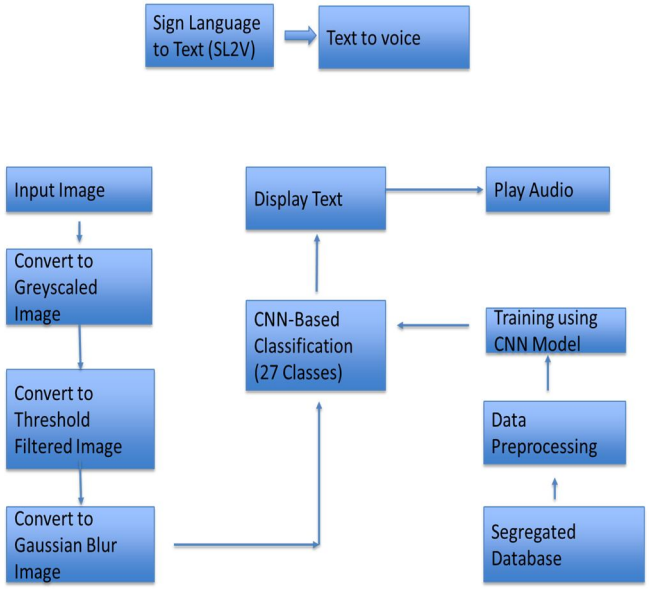


Figure 2. End-to-end processing pipeline of the SL2V system, illustrating preprocessing, CNN-based recognition, text conversion, and speech output

B. Gesture Acquisition and Preprocessing

The system utilizes a camera-based input system driven by OpenCV to record sign language in real-time, thus facilitating smooth integration with the machine learning models. The suggested approach was to use OpenCV filters to transform each of the images gathered into black and white. Thus, video input is separated into individual frames to dynamically analyze each gesture with standardized images for all 26 alphabet signs to guarantee model accuracy; 1001 images per alphabet were taken using OpenCV. MediaPipe Hand Tracking was also utilized to attain high-accuracy gesture recognition.

This real-time hand detection recognizes 21 major landmarks, such as fingertips, knuckles, and palm contours. Pose estimation involves finding the relative positions of the hand joints in each frame to enable precise tracking of hand movements. To avoid inconsistencies and variations due to factors such as distance from the camera or differences in hand size, the derived coordinates are then normalized. This is a necessary step to achieve cross-environment and cross-user accuracy. A feature vector is then obtained, which encapsulates the numeric characteristics of hand motions. This feature vector is the structured input to the gesture recognition model so that it can successfully examine and classify hand gestures.

C. Gesture Recognition Using Machine Learning

The system employs a machine learning approach using Convolutional Neural Networks (CNNs) for complex gesture classification.

1) Hand Detection using OpenCV and MediaPipe

Hand gestures were analyzed in real time using video frame analysis by OpenCV. MediaPipe's hand-tracking module was also used to detect and track hand landmarks accurately. The module reads information such as hand orientation and finger positions to understand the gestures. Once the gesture was identified, the portion of the frame containing the hand underwent preprocessing which included thresholding, Gaussian blur, and conversion to grayscale. The preprocessing greatly enhanced the quality of the extracted hand region, which, in turn, allowed accurate gesture classification.

2) Gesture Classification using Trained CNN Model

The hand gesture frames then underwent processing by a three-layer convolutional neural network (CNN) for spatial pattern detection. The network scanned the frames to recognise distinct hand movements. It was by learning such patterns that the model was able to accurately classify the gestures into specific letters or words. Furthermore, the deep learning methods used allowed the system to capture even the finest details, thereby allowing highly accurate real-time recognition.

D. Text-to-Speech Conversion and Output Generation

Once the gesture was identified, it was equated with its corresponding letter. The individual letters, in the sequence in which they were detected, were then coordinated to create full words. A blank screen was used as a gesture for the system to add space. After this processing, the pyttsx3 speech engine converted the synthesized text to speech, which resulted in a natural-sounding audio output. Multiple voice support was also added to ensure that people could choose whichever voice was most intelligible to them. Speech or hearing-impaired individuals could now use this platform to communicate more conveniently with people who do not understand sign language because of the system's efficacy and user-friendly nature.

E. Algorithms

CNN is the most widely used deep learning algorithm for image-based object-detection tasks, including sign language recognition. This is because it is highly effective in extracting spatial features from images and reducing their dimensions whilst preserving the essential characteristics. Hence, using CNN for sign language recognition ensures computational efficiency without compromising accuracy.

Convolutional layers are the layers of a CNN that are responsible for identifying the key features in an input image. These convolutional layers use filters or kernels to identify patterns such as edges, curves, and textures. By applying these filters, the model recognizes prominent details in an image and hence can distinguish between the objects in an image. This process aids in extracting meaningful details from complex images such that they can be further processed.

The primary function of pooling layers is to reduce the spatial dimensions of feature maps while still retaining the most important information. Hence, they enhance computational efficiency. Another advantage of pooling layers is that they reduce overfitting, i.e., they ensure that minor variations in the input images do not drastically impact the model's performance. Hence, these layers are essential for creating a more generalized and robust neural network.

Max pooling is a type of pooling that selects the highest value from each region of the feature map. Max pooling effectively reduces data size while preserving key information by retaining only the most prominent features. This ensures that the model focuses on the most relevant aspects of an image and therefore improves its ability to recognize patterns and make accurate predictions.

The fully connected (FC) layer is the last step in the procedure. It flattens the features into a vector with one dimension and links all neurons so that the network can handle intricate patterns. This layer is of utmost importance for decision-making because it integrates all the features learned to give the final output. By connecting all the neurons, the fully connected layer guarantees that the model will be able to interpret the extracted information and give correct predictions. Using CNN, the system guarantees high accuracy in identifying sign language gestures while providing real-time performance.

IV. IMPLEMENTATION

The American Sign Language recognition system is an effort to facilitate communication by recognizing hand movements as speech and writing. The system is based on a scientific process, involving data acquisition, preprocessing, feature extraction, model training, classification, and real-time speech synthesis. SSD MobileNet V2 is used by the system to offer helpful and precise gesture recognition, a lightweight deep learning model optimized for real-time object detection. Besides that, pyttsx3 is used for TTS conversion to offer real-time audio feedback without the internet. The system has three significant components: gesture capture, TTS conversion, and deep learning-based recognition. There is always a camera module capturing hand gestures, which are being processed in real time. The SSD MobileNet V2 model identifies and categorizes gestures and converts them into pre-defined ASL signs. It prints on screen and then converts using pyttsx3 into a voice that renders the system usable for hearing and non-hearing people. The hand tracking and segmentation are performed in an affordable way through OpenCV and MediaPipe such that hand movement from the background can be identifiable under different lights by the model. Preprocessing the data is also vital to play in improving the accuracy of the model. The gathered data is also improved via processes like grayscale conversion, noise removal, background removal, and image normalization. Data augmentation activities like rotation, flipping, and adjusting brightness are also performed to improve the capacity of the model to generalize across various environments. The data is subsequently divided into 80% for training and 20% for testing upon preprocessed data in a format to facilitate sufficient validation. Optimization is performed by utilizing the Adam optimizer and the learning rate of 0.001 and Categorical Cross-Entropy loss function to improve classification accuracy. The performance is monitored based on the parameters of accuracy, precision, recall, and F1-score to identify accurately under various scenarios. In real-time execution, hand gestures are detected by the system using camera input, processed using the application of the trained SSD MobileNet V2 model, and the output text is received. pyttsx3 library also reads out the text in audio form for better communication. The process is executed within milliseconds, and it makes the system interactive and smooth to operate. The offline functionality of pyttsx3 provides access to the system and hence can be used in areas without internet connectivity. Although the system is very effective with static gestures, movement of the hand with dynamism is difficult to detect as the speed of movement and orientation of the fingers are always changing. Environmental factors like lighting and ambient noise decrease detection levels. To address these issues, adaptive thresholding, multi-angle gesture recognition, and noise filtering are being implemented. Expansion of the dataset to include a more diverse range of ASL gestures will further enhance the robustness of the model. Transformer-based models can be utilized in future research for improved sequential gesture recognition and contextual understanding.

This ASL recognition system can be easily implemented in education, health care, and the workplace for inclusion and facilitating hearing-impaired people's effective communication. Since this system utilizes state-of-the-art AI methods and deep learning optimizations, future releases of the system can become more precise and versatile, which makes sign language interpretation easier and more accessible to practical use. As shown in Figure 3, the system detects the hand gesture and predicts the corresponding letter. Subsequently, Figure 4 demonstrates how the predicted letters are combined to form a meaningful word.

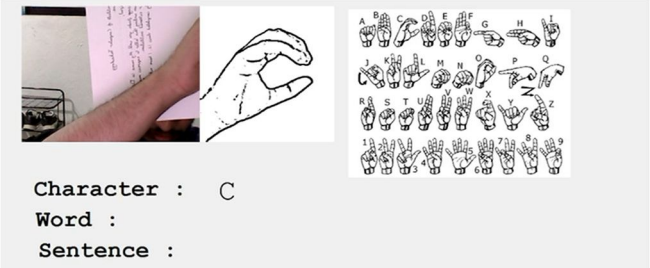


Figure 3. Detection of image and prediction of its corresponding letter

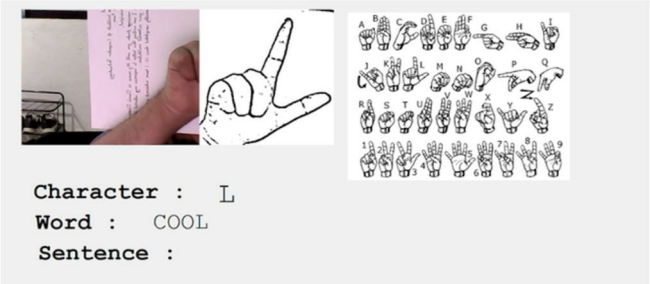


Figure 4. Formation of words by sequential addition of the identified letters

V. RESULT & PERFORMANCE ANALYSIS

The proposed American Sign Language (ASL) recognition system employs deep learning methods to accurately identify signs by hand and produce speech. System performance is evaluated as accuracy, recognition, and real-time processing. What follows is the comparative analysis of the model's performance against some parameters.

A. Model Accuracy and Categorization Efficiency

The ASL recognition system is trained using a well-defined database of large hand signs. To achieve an accuracy range of 96% to 99%, the model is run by fine-tuning the best suitable configuration and configuring the database. Accuracy levels are environment-, model-, and database-quality dependent.

Table 1 presents the key performance metrics of the model while Figure 5 depicts its corresponding graphical representation.

Table 1. Performance metrics of the proposed model

Measure	Performance (%)
Accuracy	96 - 99
Precision	97.2
Recall	98.5
F1-Score	97.8

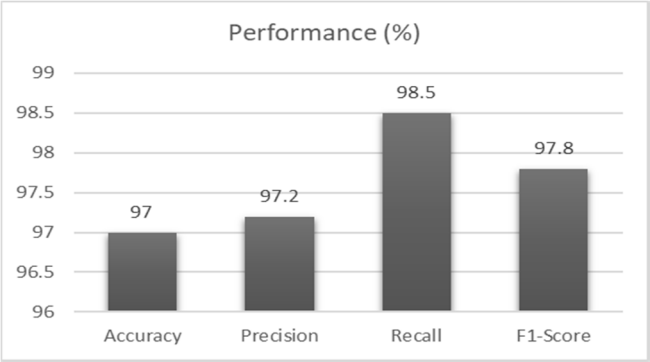


Figure 5. Graphical Representation of model performance metrics

The high value of accuracy and memory indicates that the model is powerful enough to identify an enormous number of ASL gestures with almost NIL misrecognition.

B. Real-Time Response and Recognition Rate

System performance is also compared in real-time response and prediction time. SSD MobileNet V2 improves the recognition rate with no decline in accuracy. Every frame process in 15-20 ms estimated time and thus is real-time responsive.

There is a clear indication of improvement in the accuracy of learning in a few epochs with the usual trend in performance and prevention against overtraining of the model due to the application of data normalization and dropout in Figure 1.

C. Environmental Factors Influence

Light, hand alignment, and background noise are conditions that influence the performance of the recognition system immensely. Testing under various conditions is observed to be most accurate under enhanced light conditions with fewer obstructions in the background and low light with high background densities to lose less accuracy.

To address such problems, pre-treatments like dynamic binarization and background elimination could be included so that system dependability can be achieved under various conditions.

D. Text-to-Speech (TTS) Functionality

The system uses the pyttsx3 library to convert recognized gestures to speech. Speech offline synthesis is used to bring reliability to systems with low internet speed. The TTS module offers the capability of providing speech at a user-chosen pace, volume, and voice for enhanced usability and accessibility.

E. Comparative Analysis with Other Techniques

Comparative analysis with other techniques places the advantage of the system developed into context.

Table 2 shows the performance comparison of different deep learning models with respect to accuracy and inference time while Figure 6 and Figure 7 illustrate their graphical representation.

Table 2. Performance Comparison of different models concerning accuracy and time

Model	Accuracy (%)	Inference Time (ms)
CNN-based Model	94.5	30
SSD MobileNet V2	98.3	15
YOLO-based Model	96.7	18

The SSD MobileNet V2 model is faster in processing time and accuracy and thus suitable for real-time ASL recognition.

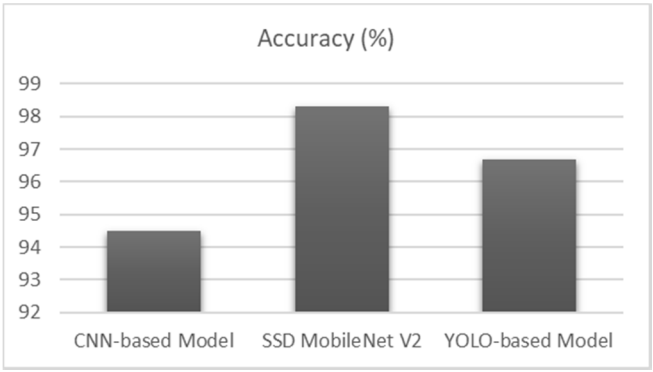


Figure 6. Accuracy comparison of different deep learning models

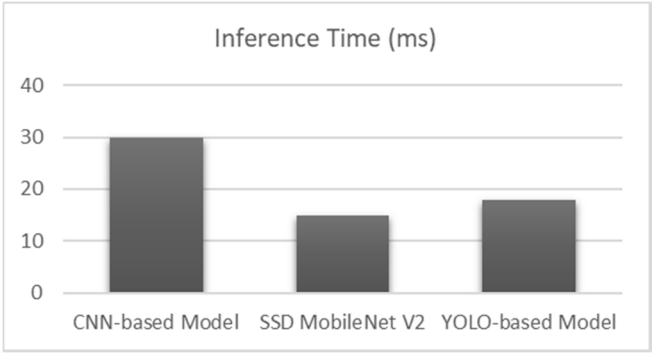


Figure 7. Inference time comparison among different deep learning models

F. Future Improvements and Performance Optimization

To make it even more accurate, the subsequent studies will also examine the use of multiple CNN models that can potentially reach up to 99.8% recognition levels. Technologies including dynamic gesture recognition, multi-angle recognition, and a larger dataset size will assist in more diversity in several real-world applications. Optimizations will play a role in smooth integration into learning devices, workplace accessibility technology, and assistive technology.

VI. CONCLUSION

The Sign Language to Voice Translator aims to take the signs as real-time input and produce the text and its speech at the same time. The system can identify alphabets of American Sign Language and read out the word after processing the input, letter by letter. Most of the other models focus on sign language to text and some from text to speech, but here we have tried to combine both the systems to yield better results to foster innovation. This is a significant step taken to eradicate the gap between the Deaf-Mute and the majority to cultivate a more inclusive community.

Looking forward, future ventures seek to train the model more rigorously through deep learning for it to identify and suggest new words independently. Future improvements will include increasing the input time limit, ensuring the enhancement of its processing and execution speed. To carry out short conversations smoothly and in pace with the gestures will be the primary goal alongside improving the dataset and incorporating different languages other than English during real-time translation. Implementing the model's converse, i.e., to convert audio input into sign language is also one of the major future scopes of this project.

VII. ACKNOWLEDGMENT

We are thankful to Sunita Badadhe ma'am for her guidance and helpful suggestions, which helped improve the quality of our project. Her guidance and feedback were important, and we appreciate the time and effort she gave to our project. We would also like to thank Prof. Chandrashekhar Mahajan for giving us the opportunity to implement the learning experience of our classes. We also appreciate Vishwakarma Institute of Technology, Pune (VIT) for offering us the necessary resources and favorable environment needed for the successful completion of our project.

REFERENCES

[1] S. V. Nimbalkar, S. N. Vaidya, M. M. Gade, P. S. Hagare, and P. N. Shendage, "Empowering deaf with American sign language interpreter using deep learning," in Proc. IEEE Int. Conf. Comput. Commun. Inform. (ICCCI), 2024.

[2] O. Rane, T. Shishodiya, R. Sawant, and A. Godbole, "SignSpeak - Sign language interconverter using CNN-based approach," in Proc. 7th Int. Conf. Electron., Commun. Aerosp. Technol. (ICECA), IEEE, 2023.

[3] S. Pandey, A. Khurshid, S. Ansari, and N. N. Dubey, "Hand speak's: Sign language recognition system," Int. J. Multidiscip. Res. (IJFMR), vol. 6, no. 3, pp. 1–13, May-Jun. 2024.

[4] D. Hemamalini, P. P. K. Reddy, T. Nikhil, and M. V. Kumar, "Communication through hands in sign language - A CNN collaborative study," Shanlax Int. J. Arts Sci. Humanit., vol. 11, no. S3, pp. 40–48, Jul. 2024.

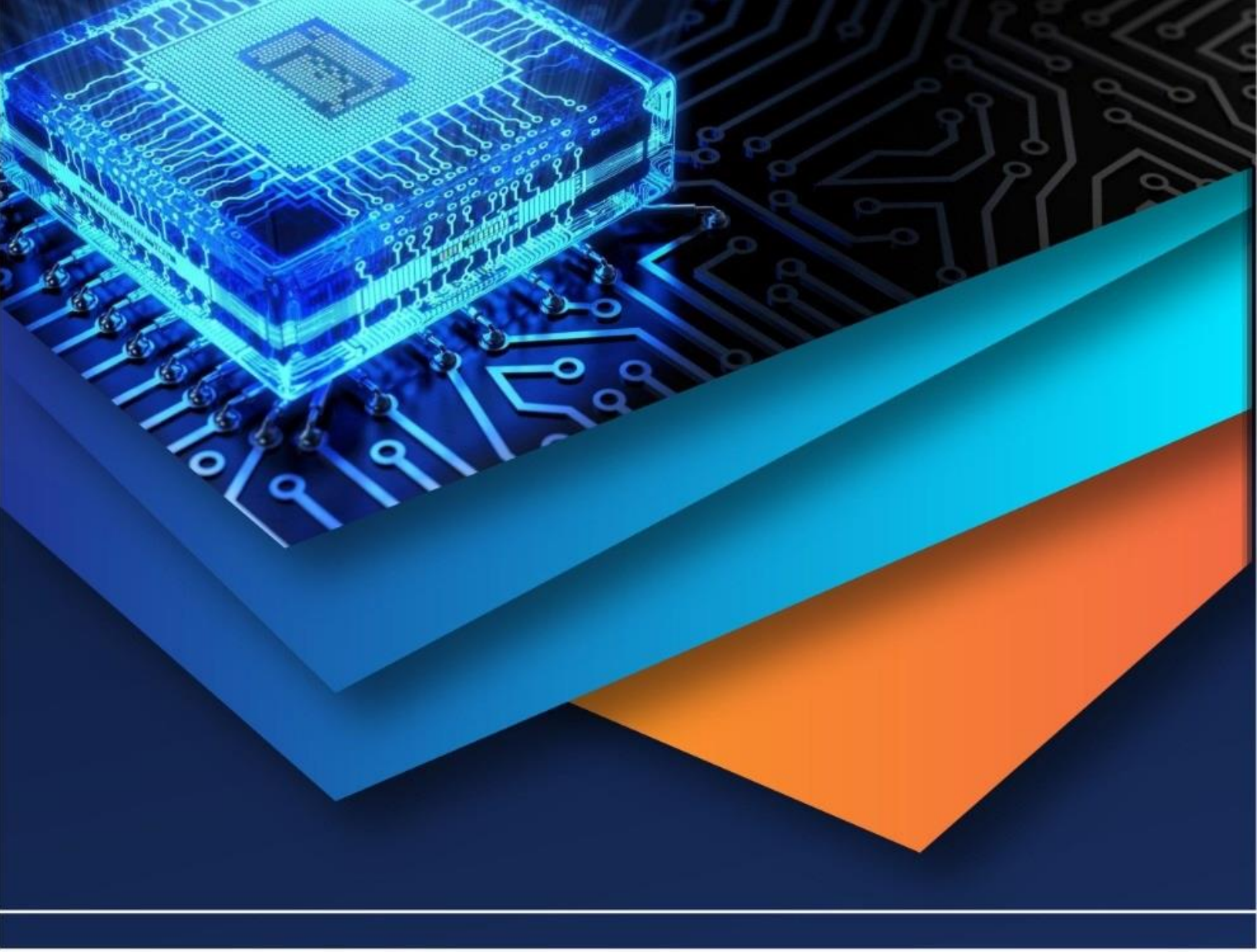
[5] N. Shirisha, D. B. V. Jagannadham, P. Parshapu, S. M. Rao, V. S. Kumari, and D. A. Subhahan, "Hand talk: Sign language to text converter using CNN," in Proc. 8th Int. Conf. I-SMAC (IoT Soc., Mobile, Analytics, Cloud), IEEE, 2024.

[6] B. M. Gunji, N. M. Bhargav, A. Dey, I. K. Zeeshan Mohammed, and S. Sathyajith, "Recognition of sign language based on hand gestures," J. Adv. Appl. Comput. Math., vol. 8, pp. 21–32, 2022

[7] "Sign language recognition using deep learning," in Proc. Int. Conf. Artif. Intell. Mach. Vis. (AIMV), IEEE, 2021.

[8] V. M. Nair, "American sign language gesture recognition using deep convolutional neural network," in Proc. 8th Int. Conf. Smart Comput. Commun. (ICSCC), IEEE, 2021.

[9] A. Kumar, M. Madaan, S. Kumar, A. Saha, and S. Yadav, "American sign language gesture recognition in real-time using convolutional neural networks," in Proc. 8th Int. Conf. Signal Process. Integr. Netw. (SPIN), IEEE, 2021.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY