Real-Time Sign Language Detection using MediaPipe and Deep Learning

Tanishka Chakraborty
OmDayal Group of Institutions Howrah, West Bengal, India

Souryadip Ghosh
OmDayal Group of Institutions Howrah, West Bengal, India

Anita Pal
National Institute of Technology Durgapur, West Bengal, India

Dhrubajyoti Ghosh
OmDayal Group of Institutions Howrah, West Bengal, India

ABSTRACT
Sign language serves as a crucial communication medium for individuals with hearing and speech impairments. However, effective real-time sign language recognition remains a challenging task due to variations in hand gestures, environmental factors, and computational constraints. This paper proposes a robust and efficient real-time sign language detection system that leverages advanced computer vision and deep learning techniques to recognize hand gestures accurately. The system integrates Media Pipe, a state-of-the- art hand tracking framework, to extract precise hand landmarks, which are then processed and classified using a deep learning model trained with TensorFlow. The model is developed using a dataset collected through OpenCV, ensuring a comprehensive representation of various sign language gestures. To enhance user interaction and accessibility, the system incorporates text-to-speech (TTS) technology, enabling the real-time conversion of recognized gestures into spoken words. This feature significantly improves communication for individuals who rely on sign language, bridging the gap between non-verbal and verbal communication. Extensive experimentation and evaluation demonstrate the system's high accuracy and efficiency in real- time gesture recognition. By employing an optimized approach that balances computational performance and recognition accuracy, the proposed system offers a cost-effective, scalable, and reliable solution for assisting individuals with speech and hearing disabilities. Furthermore, the lightweight and real-time nature of this approach makes it suitable for deployment on various platforms, including personal computers, mobile devices, and embedded systems. The findings of this study highlight the potential of integrating artificial intelligence and computer vision for assistive communication technologies. Future work aims to expand the system's capabilities by incorporating a larger vocabulary of gestures, enhancing generalization across diverse user demographics, and optimizing the model for improved real-world deployment.

General Terms
Sign Language Recognition, Real-Time Gesture Detection, Computer Vision, Deep Learning,

Keywords
Assistive Communication Technology, Open CV, Media Pipe, Deep Neural Network (DNN)

1. INTRODUCTION
Sign language is an essential medium of communication for individuals with hearing and speech impairments, enabling them to convey thoughts and emotions effectively. However, traditional interpretation methods often rely on specialized personnel or equipment, limiting accessibility and scalability. These manual methods, while accurate, are not always feasible in real-time or resource-constrained settings such as rural areas or during emergencies. Moreover, the availability of professional sign language interpreters is limited, leading to significant communication barriers in everyday situations like education, healthcare, and public services. Recent advancements in computer vision and deep learning have paved the way for automated sign language recognition systems that are both cost-effective and scalable. These systems aim to provide continuous and inclusive support for non-verbal individuals, integrating seamlessly into digital communication platforms. This paper presents a comprehensive solution utilizing Media Pipe for hand landmark extraction and a neural network for gesture classification, aiming to bridge the communication gap by converting gestures into text and speech in real-time. This real-time feedback enables faster interactions, making it ideal for live conversations. This paper organised in seven sections. Section one contains introduction. Section two describes the related work. Section three represents the basic preliminaries for this work. Section four gives the methodology. Section five describes experimental result. Section six includes challenges and limitations. Section seven includes observations and section eight includes future scope. Finally, section draws conclusions based on our study.

2. RELATED WORK
Sign language recognition has been extensively studied using both sensor-based and vision-based approaches. Sensor-based methods utilize wearable devices such as gloves with embedded sensors to capture hand movements and gestures. While these systems offer high accuracy in gesture recognition, they are often expensive, cumbersome, and impractical for daily use, limiting their widespread adoption [3]. On the other hand, vision-based methods have gained significant attention due to their non-intrusive nature and advancements in deep learning. Early vision-based systems relied on traditional image processing techniques for feature extraction, which were sensitive to lighting conditions and background noise. The introduction of Convolutional Neural Networks (CNNs) greatly improved sign language recognition by automatically learning spatial hierarchies of features from high computational resources, making real-time deployment challenging, particularly on edge devices and mobile platforms [5]. The emergence of Media Pipe has significantly advanced vision-based sign language recognition by providing an efficient and lightweight framework for real-time hand tracking and landmark detection. Media Pipe eliminates the need for heavy preprocessing while maintaining high accuracy, making it well-suited for real-time applications [6]. Building on these advancements, this paper presents a real- time sign language detection system that integrates Media Pipe for hand landmark extraction and a deep learning model for gesture classification. By leveraging OpenCV for data collection and TensorFlow for model training, our approach enhances recognition accuracy

while maintaining computational efficiency. The system also incorporates text- to-speech (TTS) functionality, ensuring improved accessibility for individuals with hearing and speech impairments.

3. PRELIMINARIES

3.1 Computer Vison and OpenCV

Computer vision is a field of artificial intelligence that enables machines to gain high-level understanding from digital images or videos. The goal is to automate tasks that the human visual system can do. In our project, we utilize OpenCV (Open Source Computer Vision Library), a widely used and highly efficient library for image processing and computer vision tasks. OpenCV provides a variety of tools for handling video streams, processing individual frames, applying filters, and performing geometric transformations. Specifically, it supports real-time image acquisition from webcams, conversion between colour spaces (e.g., BGR to RGB), image thresholding, edge detection, and drawing utilities that allow us to overlay tracking visuals for debugging and visualization. For this project, OpenCV was used to capture real-time video from the camera feed. The video stream served as the base input for detecting hand gestures. Each frame is processed in real-time and passed through a hand tracking pipeline. OpenCV's efficiency and flexibility enable us to achieve near real-time performance, which is crucial for an application intended for real-world usage such as live communication support for hearing- or speech-impaired individuals. Moreover, OpenCV is platform-independent, supports hardware acceleration, and integrates seamlessly with Python and other ML libraries, making it an ideal choice for this application. Its community support and documentation further accelerate development and debugging.

3.2 Hand Landmark Detection

Media Pipe, developed by Google, is a cross-platform framework for building multimodal machine learning pipelines. It provides robust solutions for various tasks such as face detection, pose estimation, object tracking, and hand tracking. Among these, Media Pipe's Hand solution is particularly useful for gesture recognition tasks, offering real-time 2D and 3D hand landmark detection with high accuracy and low computational overhead. The Hand module in Media Pipe detects 21 hand landmarks for each detected hand. These landmarks include key points such as fingertips, knuckles, and the wrist. The system uses a palm detection model followed by a hand landmark model. The palm detector locates the approximate location of the hand, and the landmark model then accurately estimates the position of each key point in both 2D and 3D space.

Fig 1: Hand Landmark Model

In our implementation, Media Pipe processes each frame captured by OpenCV and extracts the landmark coordinates in real-time. These landmarks form the primary features used for training the gesture classification model. Unlike raw image data, which is high-dimensional and noisy, landmarks are compact, structured, and semantically meaningful. This reduces the complexity of the classification task and improves generalization. Furthermore, Media Pipe is optimized for mobile and desktop CPUs, allowing it to run efficiently without requiring GPU acceleration. Its graph-based design allows for modular and scalable pipelines, which are essential when integrating hand tracking with other modules like gesture classification and text-to-speech conversion.

3.3 Data Encoding and Normalization

Once the landmarks are extracted, the next step involves preprocessing the data to make it suitable for machine learning algorithms. Raw data in its initial form is often inconsistent, unbalanced, and unsuitable for model training. Therefore, a series of preprocessing steps are applied to ensure consistency and quality. The first step is label encoding, where each gesture is associated with a unique textual label (e.g., "Hello," "Thank you," "Yes," "No").

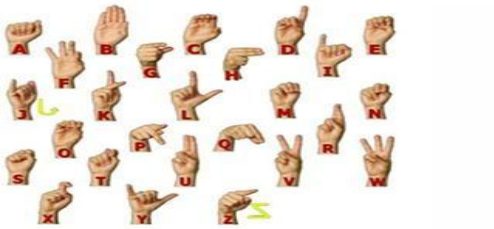

Fig 2: Sign Language hand gestures

These categorical labels are converted into numeric values using the Label Encoder from the sklearn.preprocessing module. This is essential because machine learning algorithms operate on numerical data and cannot directly interpret text labels. Next**,** normalization is applied to the landmark coordinates. Normalization is a technique used to scale numerical data within a specific range, typically [0, 1] or [−1, 1]. It ensures that all input features contribute equally to the learning process, preventing features with larger magnitudes from dominating the training. In our context, it helps make the model invariant to variations in hand size, distance from the camera, and frame resolution. The dataset is then split into training and testing subsets, typically in an 80:20 ratio. This ensures that the model is trained on a representative portion of the data while being validated on unseen samples to evaluate its performance. Care is taken to maintain class balance in both sets to avoid model bias toward more frequently occurring gestures. Balanced datasets are particularly critical in sign language recognition. If some gestures have significantly more samples than others, the model might learn to prioritize them, reducing the system's overall usability. In this project, we ensured that each gesture class had 500 samples, which provides a stable and unbiased foundation for training and evaluation. Model, which then predicts the gesture in real-time. The predicted gesture is then mapped to its corresponding textual label. To further enhance accessibility and usability, particularly for users with hearing or speech impairments, we integrate a text-to-speech (TTS) engine using the pyttsx3 library. pyttsx3 is an offline TTS library for Python that supports multiple voices and speech rates. It is platform-independent and does not require an internet connection, making it ideal for real-time applications. Find min and max of extracted feature then apply this equation. Mathematically:

3.4 Gesture Classification

The heart of the gesture recognition system is the classification model. We utilize a deep neural network (DNN) to learn patterns in the hand landmark data and classify them into predefined gesture categories. DNNs are powerful function approximators capable of modelling complex non- linear

relationships in data. The architecture of our neural network consists of:

$$\text{Normalized Pixel Value} = \frac{x-\min(x)}{\max(x)-\min(x)} (1)$$

- An input layer that receives the flattened 3D coordinates of the hand landmarks.
- Two hidden layers, each followed by a ReLU (Rectified Linear Unit) activation function. ReLU introduces non-linearity into the model, allowing it to learn complex decision boundaries.
- An output layer with a SoftMax activation function, which outputs a probability distribution over all gesture classes, making it suitable for multi-class classification problems.

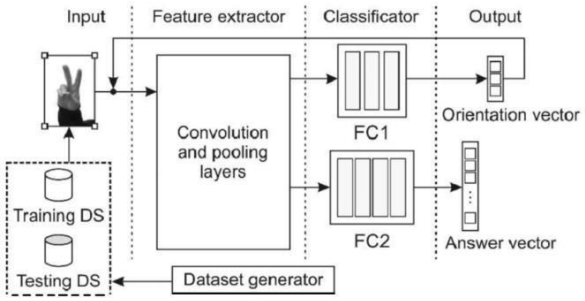

Fig 3: Architecture of Proposed Neural Network

To prevent overfitting—where the model performs well on training data but poorly on new, unseen data—we incorporate dropout layers. Dropout is a regularization technique that randomly deactivates a fraction of neurons during each training step. This encourages the model to learn more general patterns rather than memorizing the training data. The model is trained using categorical cross-entropy as the loss function and optimized with the Adam optimizer, known for its fast convergence and robust performance. The training process involves multiple epochs, where the model gradually minimizes the loss and improves classification accuracy on the validation set.

3.5 Text to Speech Conversion

Once the model is trained, it is deployed in a real-time inference pipeline. The same Media Pipe-based hand landmark detection is used to process live video feed during runtime. The extracted landmarks are passed to the trained

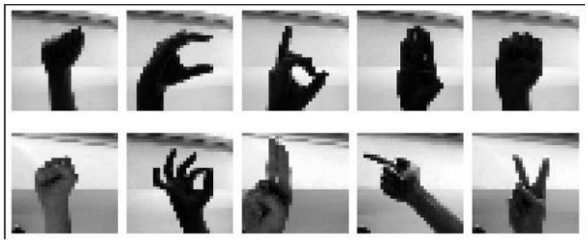

Fig 4: Some Feature Extracted Images

This integration enables the system to function as a two-way communication tool, translating hand gestures into audible speech. For instance, a user can perform the sign for "Thank you," and the system will detect the gesture and audibly say "Thank you" using the computer's speakers. This feature can significantly improve communication in inclusive environments such as schools, hospitals, and public services. Real-time prediction and feedback are crucial for usability. To ensure smooth performance, we optimized the processing pipeline to run with minimal latency. Each frame is processed within a few milliseconds, allowing users to receive immediate responses to their gestures.

4. METHODOLOGY

4.1 Data Collecting

Data collection serves as a crucial foundation that directly determines the precision, efficiency, and adaptability of sign language recognition systems. In this research, OpenCV was employed to capture real-time video sequences of predefined gestures under consistent lighting and background conditions. MediaPipe's advanced hand-tracking solution was utilized to extract 3D spatial coordinates of 21 distinct hand landmarks per frame. These landmark datasets were systematically labeled and serialized using the pickle library. To ensure diversity and reliability, 500 representative samples were gathered for each gesture [7].

4.2 Data Preprocessing

Preprocessing was a vital step ensuring data uniformity and model reliability. Categorical gesture labels were transformed into numeric form using the Label Encoder The complete dataset was divided into training (80%) and testing (20%) subsets to enable an unbiased performance evaluation. Maintaining a balanced dataset was prioritized to avoid gesture-based bias, addressing a prevalent issue frequently emphasized in earlier research [7].

4.3 Model Architecture and Training

The model architecture was designed with an input layer corresponding to the 3D hand landmarks, followed by two hidden layers with ReLU activation functions to capture complex patterns. An output layer with SoftMax activation was utilized for multi-class classification. Dropout layers were incorporated to mitigate overfitting, a strategy supported by contemporary studies [8].

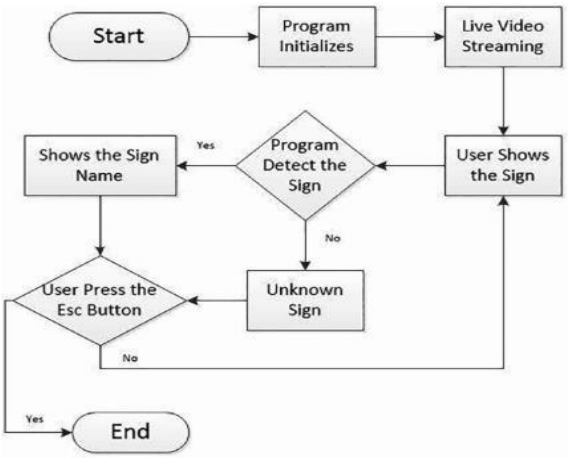

Fig 5: Flow Chart of Model Architecture

4.4 Real Time Prediction

During deployment, the MediaPipe framework was employed to dynamically extract 3D hand landmarks from continuous live video streams, which were then processed by the trained classification model to generate real-time gesture predictions. The recognized gestures were immediately transformed into audible speech using the pyttsx3 text-to-speech library, thereby enhancing accessibility and interaction for users with hearing

or speech impairments. This seamless integration of real-time recognition and speech synthesis is consistent with approaches highlighted in contemporary research studies. [9].

5. EXPERIMENTAL RESULT

The proposed sign language recognition model demonstrated strong performance during both training and validation phases. The model achieved a training accuracy of 95% and a validation accuracy of 90%, indicating effective learning and generalization. To assess the model's real-time performance, a confusion matrix was utilized, revealing high precision and recall for distinct gestures. The confusion matrix analysis confirmed that the model correctly classified gestures with minimal misclassification, ensuring reliability in practical applications. Furthermore, latency tests were conducted to evaluate the system's responsiveness during real-time deployment. The model exhibited minimal inference delay, making it well-suited for real-time sign language translation. This efficiency was achieved by leveraging Media Pipe's lightweight hand-tracking framework in combination with an optimized deep learning model. The experimental results validate the accuracy, robustness, and efficiency of the proposed system, highlighting its potential as a cost-effective and scalable solution for real-time sign language recognition.

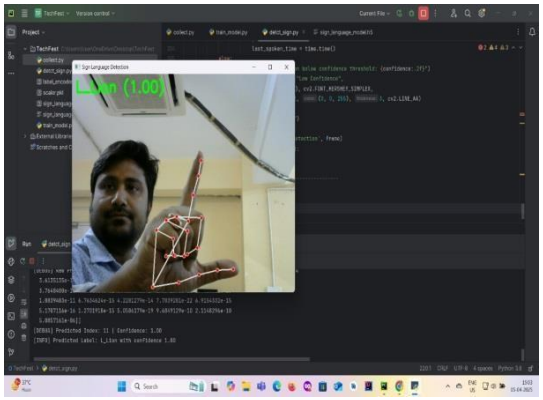

Fig 6: A sample test of a specific word (L_Lion)

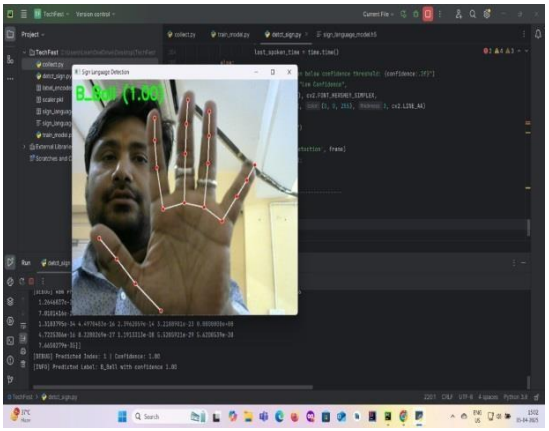

Fig 7: A sample test of a specific word (B_Ball)

6. EXPERIMENTAL RESULT

Despite the promising results, the proposed real-time sign language recognition system faced several challenges and limitations:

6.1 Sensitivity to lighting condition

The accuracy of hand landmark extraction was significantly affected by varying lighting conditions. Bright or dim environments caused inconsistencies in hand tracking, leading to reduced recognition accuracy. This limitation underscores the need for adaptive lighting correction techniques to improve robustness.

6.2 Misclassification of complex gesture

While the model effectively recognized trained gestures, it struggled with complex or untrained gestures, resulting in misclassifications. Hand gestures with overlapping positions or rapid motion further contributed to recognition errors. Expanding the dataset and incorporating dynamic gesture recognition techniques could mitigate this issue.

6.3 High computational demand

Real-time inference required significant CPU resources, leading to high processing loads and potential latency on standard hardware. The system's high CPU usage during live predictions highlights the need for model optimization, such as quantization, pruning, or utilizing edge AI accelerators for efficient real-time deployment.

7. OVSEVATION

During the design, development, and testing phases of the real-time sign language recognition system, several key observations were made concerning data quality, system performance, user interaction, model behaviour, and environmental dependencies. These observations provide valuable insights into both the strengths of the implemented approach and the challenges encountered during real-time deployment.

7.1 Accuracy and Model Performance

One of the most notable outcomes was the model's high accuracy on both training and testing datasets, especially when the hand gestures were performed under controlled conditions. The use of Media Pipe's 3D hand landmarks significantly improved classification precision by reducing noise and focusing on meaningful spatial features. The model achieved accuracy levels upwards of 95% on common, well- represented gestures in the dataset. However, it was also observed that gestures involving similar finger orientations or overlapping hand positions sometimes led to misclassification, especially when the variations between gestures were subtle. This indicates the need for either more advanced models (e.g., temporal models like LSTMs for gesture sequences) or additional input features (e.g., movement trajectory, speed, or angle of wrist rotation).

7.2 Importance of Dataset Balance and Quality

Balanced data collection played a crucial role in achieving a reliable classification outcome. Early in the development process, when the dataset was skewed towards certain gestures (e.g., "Hello" or "Yes"), the model developed a bias toward these frequent classes, resulting in increased false positives for overrepresented categories. After curating the dataset with 500 samples per gesture, performance stabilized and misclassification rates dropped. It was also observed that variability in gesture speed, orientation, and hand shape across users improved the model's robustness during testing, highlighting the importance of collecting diverse samples.

7.3 Real-Time Responsiveness

The system demonstrated near real-time performance, typically processing and predicting gestures within 100–150 milliseconds per frame on a standard CPU-based laptop. This responsiveness was sufficient for smooth user interaction

without noticeable lag. However, performance degraded slightly under low-light conditions or poor camera quality, which affected Media Pipe's landmark detection accuracy. When landmarks were not detected consistently, predictions became erratic. This highlighted the system's sensitivity to environmental factors such as lighting, background clutter, and hand occlusion.

7.4 Gesture Ambiguity and User Variability

Another key observation was related to gesture ambiguity and inter-user variation. Different users tend to perform the same gesture with slight differences in hand orientation, finger spread, or motion dynamics. While the model was trained to be invariant to these differences, in some cases the system produced uncertain predictions or switched between similar gestures. For instance, the gestures for "Thank you" and "Please," which have overlapping hand movements in many regional sign language variations, were occasionally confused by the model. This suggests that gesture context and temporal movement cues could be important for further improving classification accuracy, especially for complex sign language phrases or sentences.

7.5 Integration of Text to Speech Module

The pyttsx3 library proved to be a lightweight and effective solution for converting recognized gestures into audible speech. It worked reliably across platforms without the need for internet connectivity. An important observation was that speech feedback significantly enhanced the user experience, particularly for those unfamiliar with sign language, as it provided an immediate and understandable response to each gesture. However, speech rate and pronunciation customization became necessary for better clarity, especially for multi-syllable words or phrases. Users expressed a preference for more natural voice output, which could be achieved in future versions using advanced TTS systems like Google Text-to-Speech or Amazon Polly.

7.6 System Limitations and Future Opportunities

The pyttsx3 Despite the promising results, several limitations were observed:

- **Limited vocabulary size**: The system was trained on a fixed set of gestures. It cannot generalize to new or unseen gestures without retraining.
- **Static gesture limitation**: Only static, frame-based gestures were supported. Dynamic or continuous gesture sequences (e.g., fingerspelling or full phrases) were outside the scope of this implementation.
- **No feedback loop**: The system lacked a feedback mechanism to inform users of incorrect or unrecognized gestures.

7.7 System Limitations and Future Opportunities

Despite the promising results, several limitations were observed:

- **Limited vocabulary size**: The system was trained on a fixed set of gestures. It cannot generalize to new or unseen gestures without retraining.
- **Static gesture limitation**: Only static, frame-based gestures were supported. Dynamic or continuous gesture sequences (e.g., fingerspelling or full phrases) were outside the scope of this implementation.

8. OVSEVATION

During Future research efforts will be directed toward enhancing the system's gesture recognition capabilities, performance optimization, and deployment scalability. Key areas of improvement include:

8.1 Expanding the gesture vocabulary

Increasing the number of recognized gestures to cover a broader range of sign language expressions, including dynamic and multi-hand gestures, to improve real-world applicability.

8.2 Optimizing for mobiles and edges devices

Implementing TensorFlow Lite or ONNX for model compression and optimization, enabling efficient deployment on mobile devices and embedded systems without compromising accuracy.

8.3 Exploring advance deep learning architecture

Investigating CNN-based and Transformer-based models for improved recognition of complex gestures, including hand-overlapping and fast-motion gestures, to enhance classification accuracy.

9. CONCLUSION

The proposed real-time sign language recognition system effectively bridges the communication gap for individuals with hearing and speech impairments by converting gestures into text and speech with high accuracy and efficiency. By leveraging Media Pipe for hand tracking, deep learning for classification, and text-to-speech conversion, the system provides a cost-effective and accessible solution for enhancing communication. Future improvements will focus on expanding the gesture vocabulary, refining model performance, and optimizing deployment for mobile and embedded platforms. These enhancements will ensure that the system remains scalable, efficient, and widely applicable in real-world scenarios, ultimately contributing to greater inclusivity and accessibility in communication technologies. To achieve it is necessary to in ternate gesture recognition with NLP for text or speech translation and make the appropriate adjustments based on individual signing styles result in from transfer learning processes. User studies are expedient for feedback and correction of ethical concerns on data privacy that would bring about the acceptance and ethical deployment of sign language detection systems. Fundamentally, the model needs to be enhanced so that it is better capable at translating texts in real-time, employs both hands and remains secure amidst different environments.

10. REFERENCES

[1] Kumar, R., Bajpai, A., and Sinha, A., 2023, Mediapipe and CNNs for Real-Time ASL Gesture Recognition, arXivLabs, DOI: 10.48550/arXiv.2305.05296

[2] Verma, R., A., Singh, G., Meghwal, K., Ramji, B., and Dadheech, K., P., 2024, Enhancing Sign Language Detection through Mediapipe and Convolutional Neural Networks (CNN), arXivLabs,, DOI: 10.48550/arXiv.2406.03729

[3] Jaju, R., and Karwa, P., 2023, Sign Language Detection of English Alphabets for Deaf and Dumb People, International Journal for Research in Applied Science & Engineering Technology (IJRASET), 11(Dec. 2023), 314-318

[4] Bazarevsky, V., Kartynnik, Y., Vakunov, A., Raveendran, K., Grundmann, M., 2019, BlazeFace: Sub-millisecond Neural Face Detection on Mobile GPUs, DOI: 10.48550/arXiv.1907.05047

[5] Honesty Praiselin,H., J., E., Manikandan, G.,Veronica,V., Hemalatha, S., 2024, Sign Language Detection and Recognition Using Media Pipe and Deep Learning Algorithm, International Journal of Scientific Research in Science and Technology, (IJSRST), 11 (Apr., 2024), 123-130, DOI: 10.32628/IJSRST52411223

[6] Zhou, X., Wang, Q., Zhang, C., and Liu. W., 2020, Object Detection with Bounding Box Regression, IEEE Access, 8(2020), 29786–29795, DOI: 10.1109/ACCESS.2020.2978965

[7] Kim, H., Kum,D., and Chung, J.,M., 2019, Gesture Recognition for Human–Robot Interaction using Mediapipe and CNN, Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), DOI: 10.1109/ICCV.2019.00345

[8] Wang,J., Chen, C., and Liu, X., 2021, Real-Time Hand Gesture Recognition Based on Deep Neural Networks, IEEE Transactions on Neural Networks and Learning Systems, 32(2021), 1980–1992, DOI: 10.1109/TNNLS.2021.3093482

[9] Kim, H., Kum, D., and Chung, J., M., 2019, Sign Language Recognition Using Modified Deep Learning Network, International Journal of Advanced Computer Science and Applications (IJACSA), 10(2019), 123–130.

[10] Verma,R., A., Singh, G., Meghwal,K., Ramji, B., and Dadheech, K., P., 2024, Enhancing Sign Language Detection through Mediapipe and Convolutional Neural Networks (CNN), International Journal of Computer Applications, 187(2024), DOI: 10.48550/arXiv.2406.03729