International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 09 , September - 2026

Sign Language Gesture Recognition

[1]Sivasankari N, [2]Dhaya Varshini S K,
[1]Principal, Sri Aravindar Arts and Science College, Sedarapet, Vanur, India
[2]Department of Mathematics, College of Engineering, Anna University, chennai, India.

Abstract **- Sign language, a visual language, is one of the primary methods of communication for hearing-impaired individuals. This language includes hand gestures and facial expressions, and body language to convey the meaning of the context, and it is a language that has its grammar and syntax. They use sign language to share their thoughts and ideas with others. However, there is a difficulty in understanding sign language for those who are unfamiliar with it. One way of translating the sign language is by using language experts and translators, but this makes the process slow and dependent on someone. with all the rising technology, this paper aims to recognize sign language and detect it using technologies such as computer vision and deep learning. This is achieved by creating a real-time dataset using OpenCV and Mediapipe, where the mediapipe model detects the gesture and this gesture is converted into a numpy array, and this data is used to train an LSTM model, which is capable of classifying hand gestures. The system was able to produce a 92% accuracy on a varied lighting dataset. This solution can help hearing-impaired people solve the communication gap in their daily interactions.**

Keywords - sign language, recognition, computer vision, deep learning, Mediapipe, LSTM

I . INTRODUCTION

Sign language is a vital means of communication that is used by individuals with speech or hearing impairment. They use hand gestures, facial expressions, and body movements to express a specific word or gesture. Similar to spoken language, sign language also differs from country to country. There are many types, like American Sign Language (ASL), British Sign Language (BSL), and Indian Sign Language (ISL). In this paper, the model was built on the Indian sign language.

The main difficulties faced by the speech and hearing-impaired individuals are that many people, including educators and employees, are not familiar with sign language, which makes it difficult for the speech-impaired people to integrate fully into society, and there is very limited access to qualified sign language interpreters, which leads to a communication barrier in the required moment. Due to these reasons, there is a high possibility of speech and hearing impaired individuals to feel discriminated and exclusive from the society.

With upcoming Technologies like machine learning and deep Learning, a solution can be brought up to solve this communication barrier between the speech and hearing impaired people the society.

OpenCV (Open Source Computer Vision Library) [10] is an open-source library that functions as the eyes for the system, helping the computer see images and videos like humans and extract information from them. It plays a crucial role in image processing, where it can enhance images, increase brightness, convert RGB to BGR or grayscale, etc. It also allows recording video in real time by providing direct access to the webcam or an external camera. Therefore, OpenCV can be used to record videos of each hand gesture, which can be stored and used as training data for the model.

Mediapipe [11] is another open-source library that can be used for hand tracking, face detection, pose estimation, and holistic body analysis. MediaPipe offers pre-trained models that can detect and track various human landmarks using just a webcam feed. The Mediapipe hand tracking model can detect both left and right hands, recognize hand gestures, and track finger movements, identifying 21 key landmarks on each hand, such as fingertips, knuckles, and the wrist.

Deep learning is a sub-branch of Artificial Intelligence and machine learning, where it can learn independently without external computations, using multiple layers that enable it to understand and recognize patterns. Among various deep learning techniques, Long Short-Term Memory (LSTM) [12] is used to recognize sign language gestures. The LSTM model is a type of Recurrent Neural Network (RNN) designed to process time series and sequential data, capturing long-term dependencies.

The proposed system seeks to close the communication gap between people who use sign language and those who do not understand it. It does this by developing a real-time gesture-to-text translation system. The system uses computer vision techniques along with machine learning models to detect and classify sign language gestures. It then converts these gestures into text for smooth communication. Tools such as OpenCV, MediaPipe, and Long Short-Term Memory (LSTM) networks help ensure high accuracy and adaptability in different environments.

II. LITERATURE SURVEY

The primary goal of Sign Language Recognition (SLR) is to translate hand gestures, signs, along with poses and body language made by the user into a more understandable or spoken language. The development of SLR systems has advanced considerably over the years; it has shifted from using hardware, such as gloves, to employing deep learning models for translating sign language. The history of SLR systems dates back to 1977, when the automation of translation started with the creation of finger-spelling robotic hands called RALPH

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 09 , September - 2026

(Robotic Alphabet), which could perform finger spelling of American Sign Language.

Later, the concept of wearable sensor-based devices emerged, such as gloves embedded with motion sensors like accelerometers and gyroscopes, known as CyberGlove. These gloves could capture finger movements. However, they were expensive to produce, required regular calibration, and were uncomfortable to wear.

Eventually, these gloves were replaced by cameras once computer vision technology advanced, enabling vision-based SLR systems. Recognition models from this period mainly used Hidden Markov Models (HMMs) and Support Vector Machines (SVMs) to classify dynamic or static gestures. However, these systems often faced challenges due to background noise, lighting variations, and limited vocabulary size.

In contrast, recent advancements use computer vision, machine learning, and deep learning . This allows systems to work with just cameras. They provide practical, wearable-free solutions for recognizing hand gestures in real-time.

Real-Time Sign Language Fingerspelling Recognition with Convolutional Neural Network [1] by Abiodun Oguntimilehin and Kolade Balogun (2023). Here, the Convolutional Neural Network (CNN) was separately trained on two sets of datasets- binary and Red Blue Green RGB, each has 25,900 images of Nigerian Sign Language. A pre-trained deep neural module was utilized to identify hand gestures in the video stream that addressed the problem of intricate backgrounds, also demonstrated great detection in poorly lit areas. Accuracies of (98.95%, 76%) and (98.87%, 98.85%) were achieved respectively on training and the validation set.

Static Sign Language Recognition Using Deep Learning [4] by Lean Karlo S. Tolentino, Ronnie O. Serfa Juan, August C. Thio-ac and Maria Abigail B. Pamahoy (2019). A skin colour-based sign language recognition system was implemented in using CNN. Performance of this system is dependent on a background with uniformity and proper lighting. Averages of 90.04%, 93.44% and 97.52% were obtained for alphabetic identification, numeric identification and static word identification, respectively, which provided an average of 93.67% detection rate.

Sign Language Recognition for the Deaf and Dumb [5] R Rumana and Reddygari sandhya Rani (2021). This paper specializes in methodology that entails creating a dataset with OpenCV through image capture and processing, using Gaussian blur to extract features, and CNNs to classify gestures. Preprocessing involves capturing images in grayscale and resizing them for model training. Problems posed by dataset constraints, filter choice, and accuracy of the model are tackled through iteration refinement and parameter tuning to enhance recognition performance and attained accuracy approximately 92.0%.

Sign Language Recognition [7] by Satwik Ram Kodandaram, Pavan Kumar and Sunil G L(2021) employed Deep Learning Computer Vision to identify the hand gesture by constructing Deep Neural Network architectures (Convolution Neural Network Architectures) in which the model will learn to identify the hand gestures images throughout an epoch. For Convolution Neural Networks (CNN) complex architectures such as LeNET-5 and MobileNetV2 and attained 97% to 98.6% of accuracy.

Audio-Visual Speech and Gesture Recognition using Mobile Device Sensors [2] by Dmitry Ryumin ,Denis Ivanko and ElenaRyumina(2023) The paper presents two deep neural network-based model designs: one for Audio-visual speech recognition (AVSR) and the other for gesture recognition. The principal novelty concerning audio-visual speech recognition is in optimizing strategies for both acoustic and visual features as well as the presented end-to-end model that takes into account three modality fusion strategies: prediction-level, feature-level, and model-level and obtained AVSR accuracy for the LRW dataset being equal to 98.76% and gesture recognition rate for the AVSR dataset being equal to 98.56%.

Sign language recognition system [9] by Sudarshan Sadashiv Bandal, Shital Satish Jadhav, and Santoshi Keshav Govalkar(2024) It employs methods such as hand detection, segmentation, and feature extraction, in addition to machine learning algorithms (CNN and SVM). The pre-processed data is maintained in a database, supporting model training to provide correct recognition of sign language words and sentences.

Dual path background erasing convolutional neural network [3] by Junming Zhang, Xiaolong Bu, Yushuai Wang, Hao Dong, Yu Zhang and Haitao Wu (2024), this research presented a lightweight, Dual Path Background Erasing Deep Convolutional Neural Network (DPCNN) model for recognizing sign language. One is to learn the general features, while the other learns the background features, the total accuracy and Macro F1 scores of the suggested method are 99.52% and 0.997, respectively.

Sign Language Recognition System using TensorFlow Object Detection API [8] by Sharvani Srivastava and Amisha Gangwar (2021) .The technique followed in this article utilizes Python and OpenCV for data capture, utilizing webcam images for gesture capture. It employs TensorFlow's Object Detection API with SSD MobileNet v2 model, which is fine-tuned from a specific dataset of sign language alphabets. The model is set for 26 classes of the alphabets. This process of the Indian Sign Language Recognition system was 93-96% accurate.

Sign language recognition employing the fusion of image and hand landmarks using multi headed convolutional neural network [6] by Refat Khan Pathan and Munmun Biswas (2023). This paper deals with sign language recognition in more complicated background with multi-colored lighting, background and user instead of plain controlled setting.ASL Finger Spelling dataset employed in this research illustrates this change by having images taken in

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 09 , September - 2026

complicated backgrounds. Preprocessing methods such as the conversion to grayscale, image normalization, and landmark detection of the hand are needed to enhance model performance. Multi-headed CNN models, which can process both raw images and hand landmarks in parallel, have been found to improve the recognition accuracy. It has achieved a validation accuracy of 98.98%.

III. IMPLEMENTATION

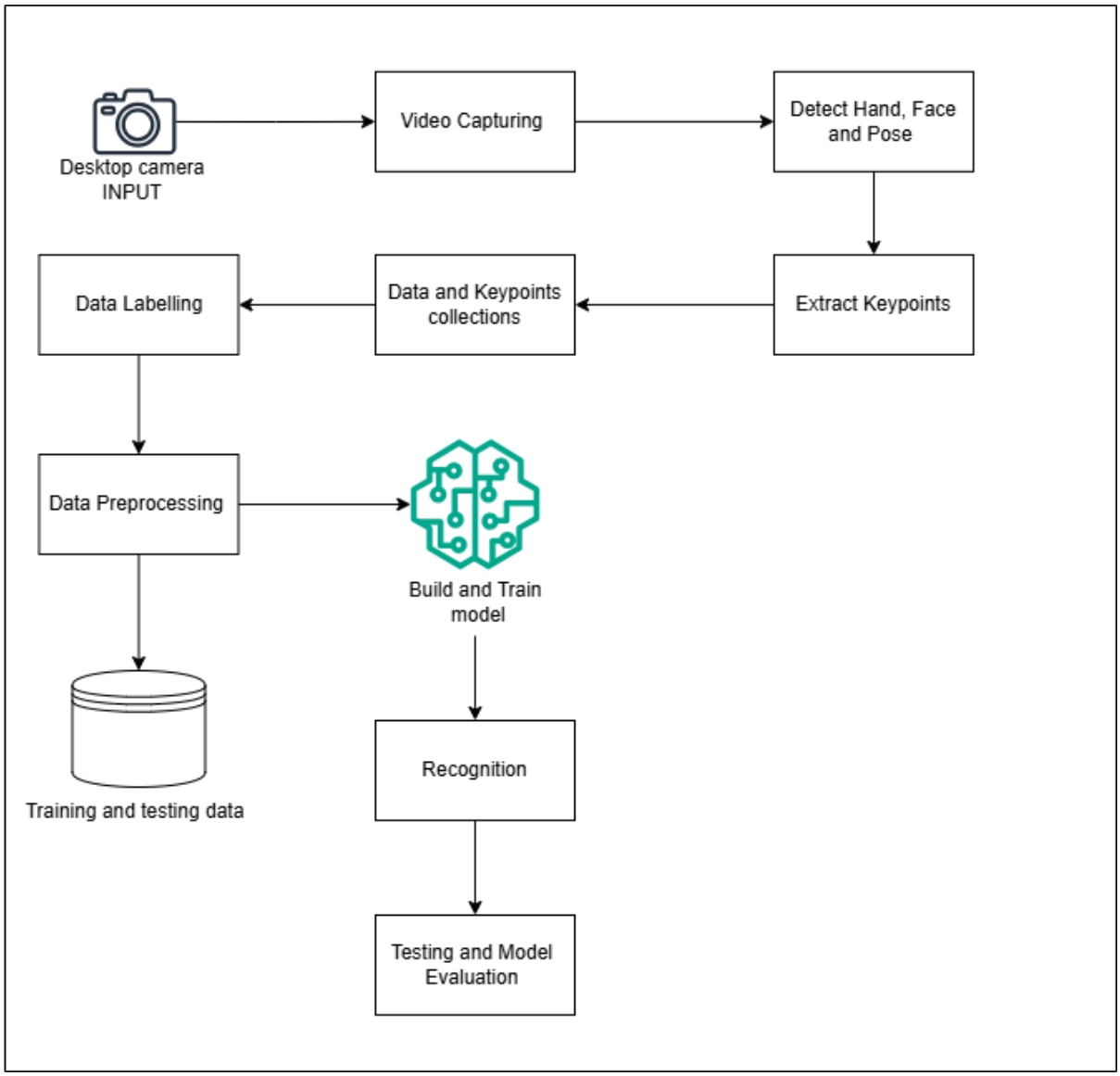

Figure 3.1 Architecture

This method doesn't use any gloves or any type of hardware; this method is a purely vision-based system that captures the hand gesture by using the system's webcam.

A. Dataset Creation

By using Computer Vision OpenCV [15], the system webcam was accessed, and using the webcam, different hand gestures were captured as separate videos. There are four different gestures on which the model was trained, which are based on the sign language

- Hello
- Thanks
- Help
- Bye

For each action, 50 videos were captured, each lasting 2 – 3 seconds. Each video will be converted into 30 different frames, where each of these frames will be sent to the Mediapipe Holistic model [14], which will be able

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 09 , September - 2026

to detect the gesture and extract keypoints from the frame. The Holistic model captures

- Left Landmarks: 21 Keypoints x 3 values (x , y, z) = 63 values
- Right Landmarks: 21 keypoints x 3 values (x , y, z) = 63 values
- Pose Landmarks: 468 keypoints x 3 values (x , y, z) = 1404 values
- Face Landmark: 33 keypoints x 4 values (x , y, z, visibility)= 132 values

Total per frame: 1662 keypoints

Where the x, y, z, and visibility represent:

- x and y coordinates: Represent the horizontal and vertical positions of keypoints in the video frame.
- Z coordinate: Provides depth information, indicating the distance of keypoints from the camera.
- Visibility: indicates the confidence level of the landmark's visibility.

For each Getsure, 50 videos were captured, and for each frame, the keypoints are extracted and these keypoints are converted into a numpy array and stored in numpy files for each frame. The dataset's folder structure is

- Root Directory (Data): Contains all gesture data.

- Gesture Folders: Subdirectories for each gesture (e.g., "hello," "thanks"), housing gesture-specific data.
- Video Folders: Subfolders within gesture directories (e.g., 0, 1…,49).
- Frame Folders: each folder contains .npy files for individual frames of a video, which have the keypoint values

Size of the dataset : (4 gesture) x (50 video) x (30 frame) = 6000 ,= .npy file

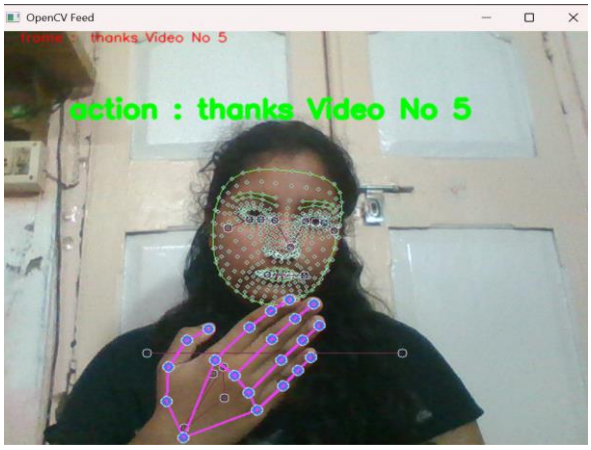

Figure 3.A (1) Thanks gesture Recognition

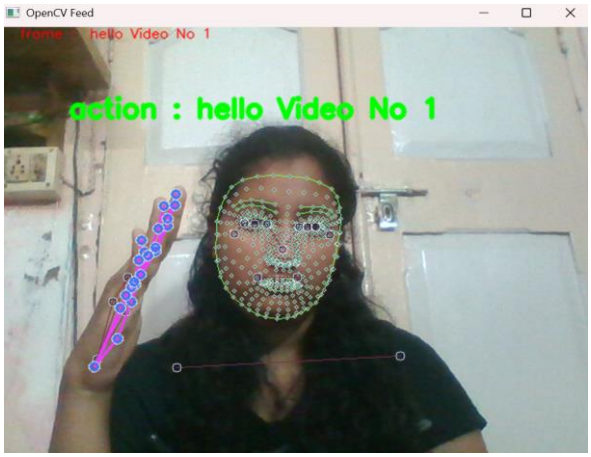

Figure 3.A (2) Hello gesture Recognition

Figure 3.A (3) Help gesture Recognition

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 09 , September - 2026

Figure 3.A (4) Bye gesture Recognition

A. LSTM Model

For sequences of videos or time series like human gestures, Recurrent Neural Networks (RNNs), specifically Long Short-Term Memory (LSTM) networks, are better suited. LSTMs can learn long-term dependencies and extract temporal patterns in sequences. The model employed in the current research utilizes stacked LSTM layers, dropout, and dense layers to create a strong model for real-time gesture recognition.

1. Long Short-Term Memory (LSTM) Layer

The LSTM layer [13] is created to capture long-term dependencies and maintain long-term patterns within sequential data like video frames. It performs particularly well when processing sequences of gestures that change as time passes. Every LSTM unit has a cell state and three gates:

- Forget gate: Determines what information to forget from the cell state.
- Input gate: Decides what new information to remember.
- Output gate: Directs the output depending on the state of the current cell.

This gated process permits the network to recall meaningful motion features in a sequence of frames while discarding meaningless noise. Activation Function used is tanh (Hyperbolic Tangent) applied for both cell state updates and calculation of the output. Output values in the range of [-1, 1].

2. Dense (Fully Connected) Layer

The dense layer is the classifier or the feature refiner. It transforms learned temporal features into a higher-level abstract space or directly into the scores of the gesture class.Each neuron in a dense layer takes input from all the neurons in the preceding layer. Calculates weighted sum followed by a non-linear activation ReLU (Rectified Linear Unit.

Softmax (in last output layer) used when outputting probabilities across multiple gesture classes. It converts the raw output scores (logits) from the previous Dense layer into probabilities for each class maps raw scores to probabilities whose sum is 1.

3. Dropout Layer

Dropout is a regularization method employed to avoid overfitting. It enhances the generalization ability of the model by randomly dropping out a proportion of neurons during training. At each forward pass at training, some percentage of neurons are randomly disabled (e.g., 30% using Dropout(0.3)). This compels the model not to become overly dependent on individual neurons, but instead to encourage redundancy and resilience.

The proposed model design integrates temporal feature extraction and classification with three LSTM layers, two Dense layers, and one Dropout layer. The LSTM layers are stacked to discover long-term dependencies and sequential motion patterns from gesture sequences. After the LSTM layer, the Dense layers flatten these features and project them to gesture class scores, and the last Dense layer with Softmax activation gives the probability distribution over gesture classes. Adding a Dropout layer improves the generalization of the model by keeping it away from overfitting while training. Overall, such a layered structure allows the model to handle real-time video streams, learn significant temporal patterns, and produce precise predictions for gesture recognition operations.

IV. RESULTS

Metric	Value(%)
Accuracy	95.83
Precision	96.67
Recall	95.83
F1-score	95.90

TABLE I. CLASSIFICATION REPORT

The performance of the proposed sign language recognition model was evaluated using several standard classification metrics to assess its effectiveness. Mainly, training and validation accuracy curves were analysed to observe how the model learns from training data and generalises to unseen data. Similarly, training and

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 09 , September - 2026

validation loss curves were examined to assess the convergence characteristics of the model.

The model achieved an overall accuracy of 95%, approximately indicating high level of performance on the test dataset. The training and validation accuracy curves given in the figure 4.A illustrate a steady improvement in model performance throughout the training process. Both curves converged towards a high value of accuracy, reaching 95% approximately. This indicates successful learning of the temporal gesture pattern.

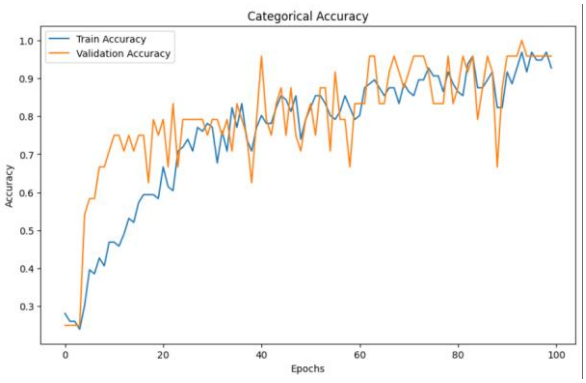

Figure 4.A (5) Training and validation accuracy curve

Similarly, the training and validation loss curve as shown in the figure 4.B exhibit a continuous reduction in loss value as training progessed, indicating stable optimization and effective parameter learning.The confusion matric further validates the robustness of proposed approach. The gesture classes “hello” and “help” achieved perfect classification performace, with all samples correctly recognized. The “bye” gesture was also successfully identified across all test instaces, while the “thanks” gesture achieved a high recognition rate.

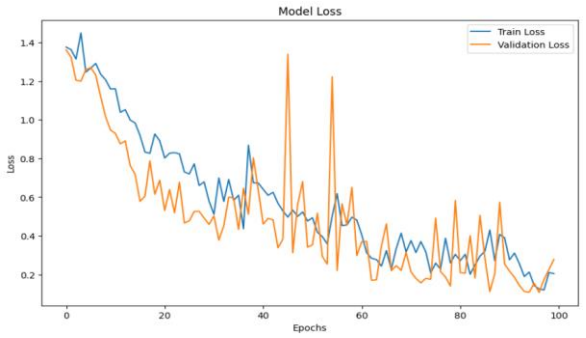

Figure 4.B (6) Training and validation loss curve

The confusion matric further validates the robustness of proposed approach. The gesture classes “hello” and “help” achieved perfect classification performace, with all samples correctly recognized. The “bye” gesture was also successfully identified across all test instaces, while the “thanks” gesture achieved a high recognition rate.

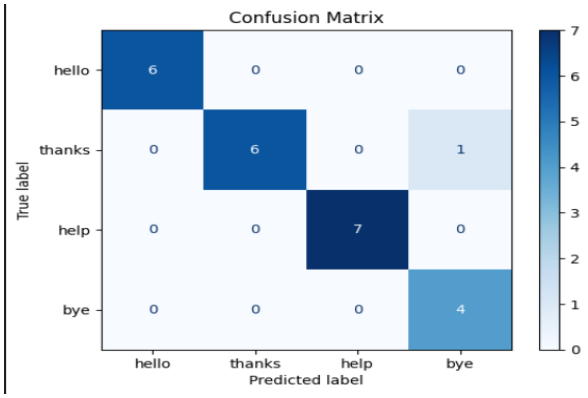

Figure 4.C (7) Confusion Matrix

V. DISCUSSION

The experimental results demonstrate that the proposed sign language recognition system is capable of accurately identifying dynamic hand gestures using computer vision and deep learning techniques. The High classification accuracy achieved by the LSTM model indicates that temporal information extracted from gesture sequences plays a significant role in distinguishing between different sign language gestures. By processing multiple frames as sequence rather than analyzing individual images independently, the model successfully captured the motion patterns associated with each gesture.

The strong performance obtained across the evaluation metrics confirms the effectiveness of combining Mediapipe provided a robust representation of hand and body movements through landmark coordinates, while the LSTM network effectively learned the temporal dependencies present in the gesture sequences.

The findings of this study contribute to ongoing efforts aimed at reducing communication barriers faced by individuals with hearing and speech impairments. By automatically translating sign language gesture into understandable outputs, the proposed system has the potential tto facilitate smoother interaction between sign language user and non sign language users.

Overall, the result validate the effectiveness of the proposed LSTM-based sign language recognition framework and demonstrate its potential for real-world implementation. The combination of computer vision, landmark-based feature extraction , and deep learning provides a reliable foundation for developing intelligent assistive technologies that support inclusive communication and improve the quality of life for hearing-impairment individuals.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 09 , September - 2026

IV. CONCLUSION

In this research, an SLR system was developed using a custom dataset, which was developed by capturing videos and converting them into frames by using OpenCV and extracting the keypoints from hands, face, and pose by using the Mediapipe Holistic model and storing them into .npy files for 4 different gestures: Hello, Thanks, Help, and Bye. This dataset was split into training and test data, where the test size was 5% and the training was 95%.
The training data was given as input to a Deep learning model LSTM The model architecture employed stacked LSTM layers, dropout regularization, and fully connected dense layers, with a final softmax activation to classify gestures effectively. Given the temporal and sequential nature of gesture data, LSTMs proved to be well-suited for capturing long-term dependencies and motion dynamics across video frames the model produced a accuracy of 95.83 % on training data and 97% on test data, demonstrating high effectiveness in recognizing complex human gestures.

V. REFERENCES

[1] Abiodun Oguntimilehin and Kolade Balogun "*Real-Time Sign Language Fingerspelling Recognition using Convolutional Neural Network"* (Vol. 21, No. 1, January 2024) https://doi.org/10.34028/iajit/21/1/14

[2] Lean Karlo S. Tolentino, Ronnie O. Serfa Juan, August C. Thio-ac and Maria Abigail B. Pamahoy *"Static Sign Language Recognition Using Deep Learning"* (Vol. 9, No. 6, December 2019) https://www.ijmlc.org/vol9/879-L0320.pdf

[3] Satwik Ram Kodandaram, Pavan Kumar and Sunil G L *"Sign Language Recognition*" (Vol.12 No.14 2021) , Article in Turkish Journal of Computer and Mathematics Education (TURCOMAT) · August 2021

[4] R Rumana and Reddygari sandhya Rani *"Sign Language Recognition for the Deaf and Dumb"* (Vol. 2 Issue 4, April - 2013) , Journal from International Journal of Engineering Research & Technology (IJERT) https://www.ijert.org/a-review-paper-on-sign-language-recognition-for-the-deaf-and-dumb

[5] Dmitry Ryumin ,Denis Ivanko and ElenaRyumina *"Audio-Visual Speech and Gesture Recognition by Sensors of Mobile Devices*" , 2023 https://doi.org/10.3390/s23042284

[6] Sudarshan Sadashiv Bandal, Shital Satish Jadhav, and Santoshi Keshav Govalkar "*Sign language recognition system"*, Volume 12, Issue 4 April 2024 , International Journal of Creative Research Thoughts (IJCRT), https://ijcrt.org/archiveold.php?vol=12&issue=4&page=98

[7] Junming Zhang , Xiaolong Bu , Yushuai Wang , Hao Dong , Yu Zhang and Haitao Wu "Sign language recognition based on dual-path background erasure convolutional neural network" (2024) , https://doi.org/10.1038/s41598-024-62008-z

[8] Sharvani Srivastava and Amisha *Gangwar "Sign Language Recognition System using TensorFlow Object Detection API"* (2021) https://doi.org/10.1007/978-3-030-96040-7_48

[9] Refat Khan Pathan and Munmun *Biswas "Sign language recognition using the fusion of image and hand landmarks through multi-headed convolutional neural network" ,2023, https://doi.org/10.1038/s41598-023-43852-x*

[10] Mediapipe https://ai.google.dev/edge/mediapipe/solutions/guide

[11] Computer visionhttps://keras.io/examples/vision/

[12] LSTM model https://www.tensorflow.org/api_docs/python/tf/keras/layers/LSTM

[13] LSTM model https://keras.io/api/layers/recurrent_layers/lstm/

[14] Mediaipe https://blog.roboflow.com/what-is-mediapipe/

[15] Computer vision and mediapipe https://viso.ai/computer-vision/mediapipe/