IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

INDIAN SIGN LANGUAGE RECOGNITION SYSTEM

[1] **S.Kumne**, [2]Yash Surve, [3]Nilesh Palve, [4]Onkar Mule, [5]Yogesh Nalawade

[1]Assistant Professor, [2,3,4,5]Under-Graduate Students

[1]Information Technology

[1]Datta Meghe College of Engineering, Airoli, India

Abstract: The Indian Sign Language (ISL) is essential for communication among the deaf and hard-of-hearing community. However, the lack of widespread ISL understanding creates a communication gap. This research focuses on developing an ISL recognition system using Long Short-Term Memory (LSTM) networks and OpenCV to process video inputs, extract features, and recognize dynamic hand gestures in real-time. The system employs pose estimation techniques to detect hand movements and classifies them using LSTM-based deep learning models. The proposed approach achieves high accuracy, ensuring a robust and scalable solution for real-time ISL interpretation.

Index Terms - Indian Sign Language (ISL), LSTM, OpenCV, Gesture Recognition, Deep Learning, Real-Time Processing, MediaPipe, Neural Networks, Computer Vision, Assistive Technology.

I. INTRODUCTION

The Indian Sign Language (ISL) serves as a crucial mode of communication for the deaf and hard-of-hearing community in India. With approximately 18 million people in India who are deaf, effective communication methods are essential for their social integration and education. Despite the prevalence of ISL, there are significant barriers to communication between the deaf and hearing populations, often leading to misunderstandings and social isolation.

Advancements in technology have paved the way for the development of sign language recognition systems, which can bridge this communication gap. Such systems can convert sign language into text or speech, thereby facilitating smoother interactions. This project aims to develop a robust Indian Hand Sign Language Recognition System that can accurately recognize and interpret ISL gestures in real-time.

II. LITERATURE SURVEY

Over the years, extensive research has been conducted to develop an efficient Indian Sign Language (ISL) recognition system. Initially, rule-based and feature extraction methods were used to identify hand gestures, relying on predefined patterns and manual feature selection. However, these approaches had limitations, including sensitivity to variations in hand shapes, background noise, and different signing styles among individuals.

To solve these challenges, researchers have used machine learning and deep learning techniques. Machine learning methods rely on specific features like shape and texture to recognize gestures. While these techniques improved accuracy, they required a lot of manual work to prepare the data. Recently, deep learning models like Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks have been used for Indian Sign Language recognition. LSTM is especially useful for understanding the sequence of hand movements over time, making it effective for recognizing dynamic gestures. Additionally, new techniques like Transformer models and attention mechanisms have further improved real-time recognition, making the system faster and more efficient. Recent advancements in deep learning have introduced Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs), which have shown improved performance in gesture recognition. Notable research includes:

Study	Focus	Methdology	Key Findings
Continuous Sign Language Recognition	**Continuous sign recognition**	**Hybrid approach (CNN+HMM)**	**Improved recognition of dynamic signs**
LSTM for Sign Language	**Temporal aspects of sign language**	**LSTM networks**	**Captures sequential dependencies effectively**
Hand Tracking Techniques	**Hand tracking and detection**	**OpenCV, MediaPipe**	**Robust keypoint detection for gestures**
Real-time Sign Language Systems	**Implementation of real-time systems**	**LSTM with video inputs**	**Challenges in latency and processing speed**

Table 1 – Literature Review

III. PROBLEM DEFINITION

Communication is a fundamental aspect of human interaction, yet millions of hearing and speech-impaired individuals face significant barriers in conveying their thoughts and emotions to the hearing population. Indian Sign Language (ISL) serves as a primary means of communication for the deaf and mute community in India. However, due to a lack of widespread knowledge of ISL among the general population, individuals using sign language often struggle to engage effectively in daily interactions.

Existing solutions for sign language recognition predominantly focus on static gesture recognition or individual hand signs, which limits their applicability in real-world scenarios where gestures form complete words or phrases. Furthermore, most current systems lack real-time processing capabilities, making them unsuitable for dynamic, real-world conversations. The challenge is exacerbated by variations in hand gestures, differences in lighting conditions, and diverse user characteristics, which introduce inconsistencies in model predictions.

To address these challenges, this research proposes a **real-time Indian Sign Language Recognition System** that leverages Mediapipe for keypoint extraction and an LSTM (Long Short-Term Memory) model for sequence-based classification. The system captures continuous hand gesture sequences, processes them into frame sequences, and accurately predicts corresponding words. The primary goal is to enable seamless communication between the hearing and speech-impaired community and the general public by providing a robust, accurate, and fast recognition system.

This research aims to:

- Develop an efficient real-time ISL recognition model capable of recognizing dynamic gestures.
- Improve the system's accuracy by leveraging sequential learning using LSTM networks.
- Ensure scalability and robustness of the model across different environmental conditions and user variations

IV. METHODOLOGY

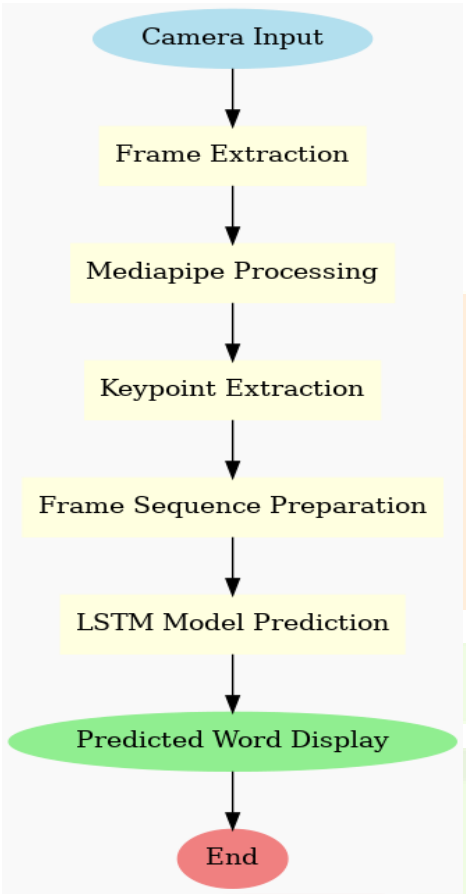


Fig 1 – DataFlow Diagram

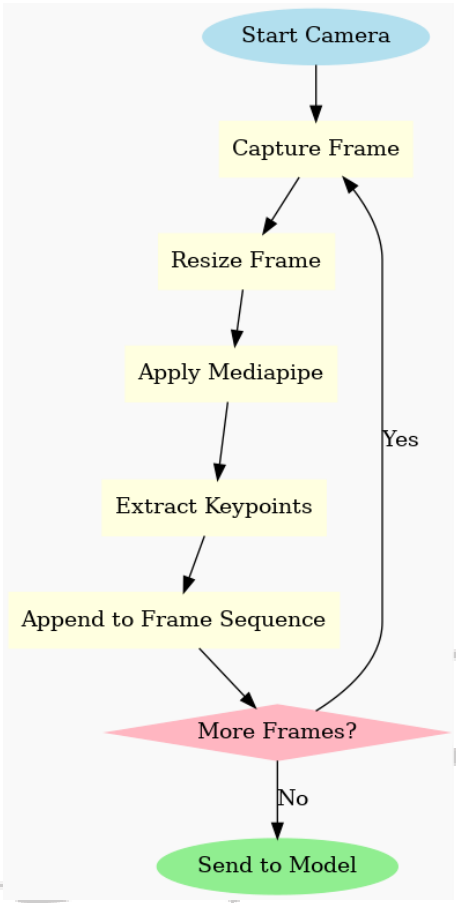


Fig 2 – Input Image Flowchart

1) **Data Collection** :
   a. Data collection refers to the systematic gathering of information needed to train and evaluate the hand sign language recognition system. In this context, it involves capturing a diverse set of hand gestures representing various signs in Indian Sign Language (ISL). This can be achieved through video recordings, images, or sensor data from multiple participants to ensure variability in hand shapes, sizes, and signing styles.

2) **Feature Extraction** :
   a. Feature extraction is the process of transforming raw data into a set of measurable properties or characteristics that can be used for analysis. In the context of hand sign language recognition, this involves identifying and extracting relevant features from the collected data, such as hand position, movement trajectories, finger orientation, and spatial relationships. These features serve as inputs for machine learning models to recognize and classify the signs accurately.

3) **Model Development** :

a. Train and optimize a Deep learning model (LSTM) for gesture classification.

- LSTM : IT is a type of recurrent neural network (RNN) architecture designed to model sequential data and capture long-range dependencies. LSTMs address the vanishing gradient problem commonly encountered in traditional RNNs, allowing them to learn from data over longer time intervals.It is used for following methods such as Sequence Prediction,NLP's,Speech Recognition, Video analysis and Anomaly Detection.

4) **Real-Time Recognition** :

a. Real-time recognition refers to the capability of the system to process and interpret hand signs as they are being performed, providing immediate feedback or output. This involves the continuous analysis of video or sensor data to identify signs in a live setting, enabling interactive communication. The system must be efficient enough to minimize latency and ensure that the recognition occurs seamlessly as the user signs.

5) **Validation and Testing** :

a. Validation involves assessing the model's performance using a separate dataset that was not used during training to ensure that it generalizes well to new, unseen data. Testing further evaluates the system's accuracy, precision, recall, and overall effectiveness in recognizing hand signs. This phase may include user studies to gather feedback on the system's usability and performance in real-world scenarios.

V. IMPLEMENTATION

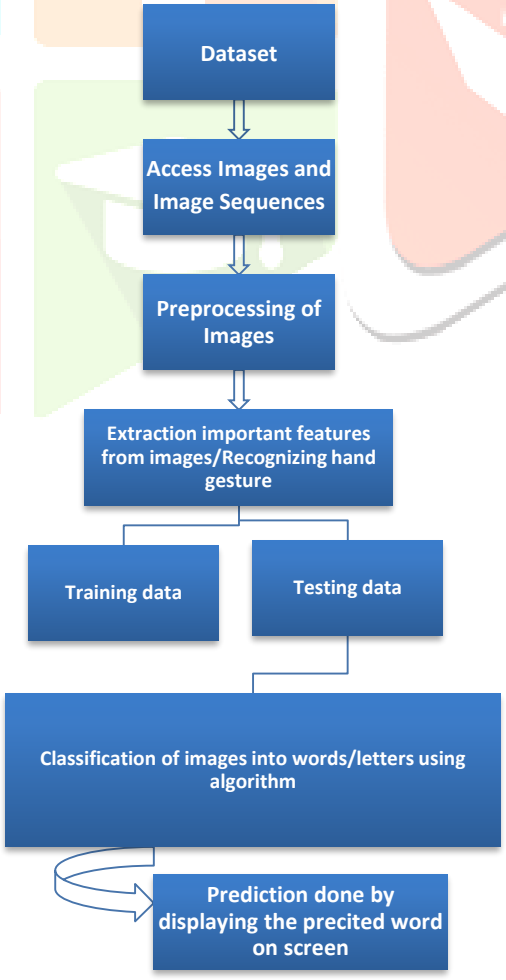


Fig 3. Implementation Flowchart

Implementing an Indian Sign Language (ISL) system requires a well-thought-out approach, as it involves the translation of hand gestures and facial expressions into text or speech. Below is a detailed step-by-step guide to creating an Indian Hand Sign Language System:

Step 1: Understanding the Problem and Scope

Before we begin, it's essential to understand the nature of Indian Sign Language (ISL). It has a rich set of gestures, hand shapes, orientations, movements, and facial expressions to convey words and emotions. In India, there is no unified sign language, and it can differ from region to region, but there is an ongoing effort to create standardized systems.

Key Points to Consider:

- The ISL has manual signs for words and also incorporates facial expressions and body movements.

Step 2: Dataset Collection

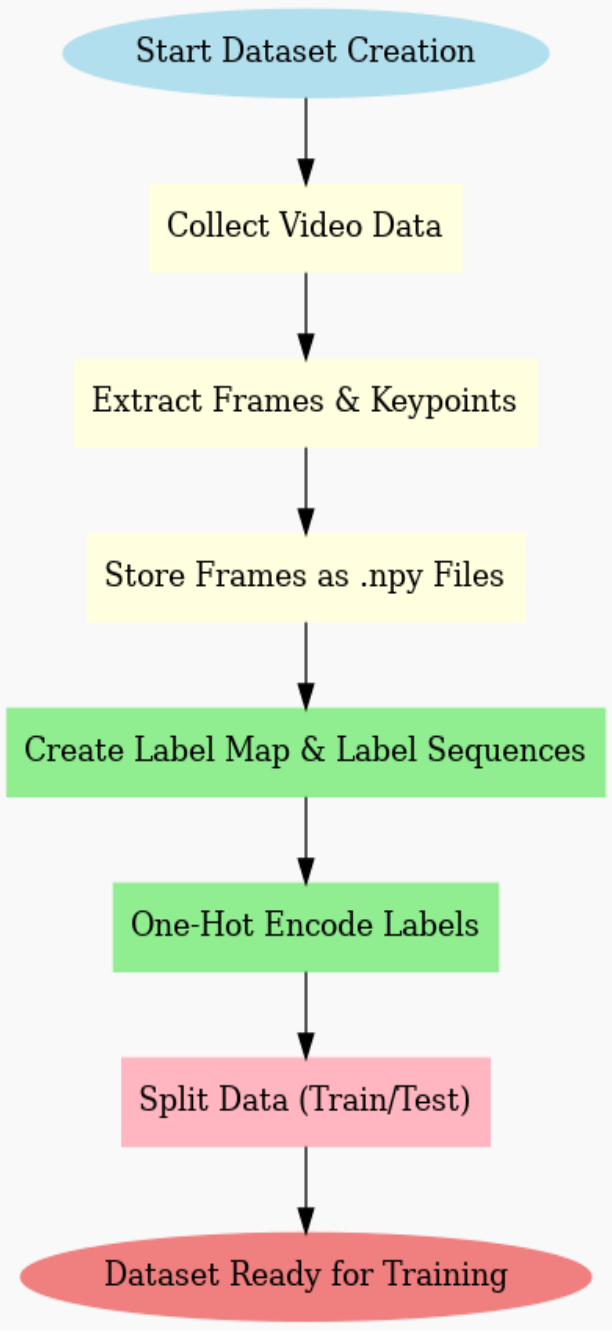


Fig 4. Dataset Creation Flowchart

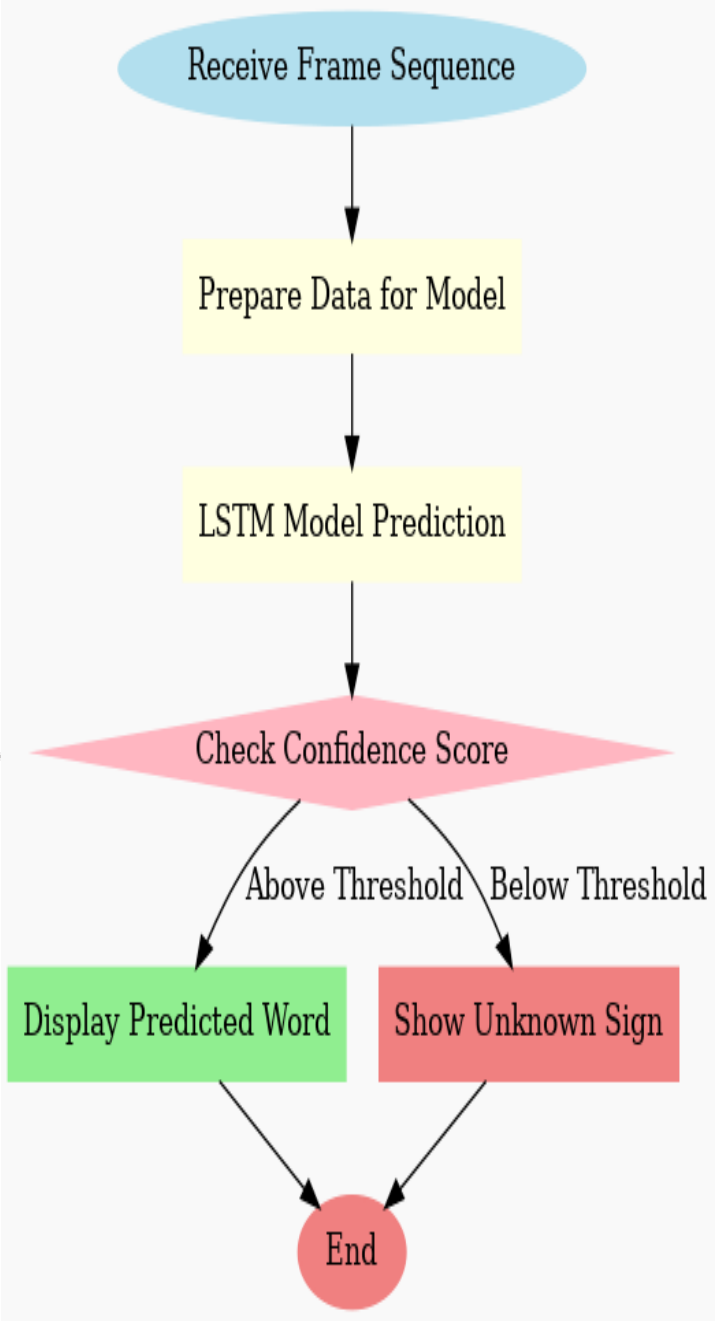


Fig 5. Model Prediction Flowchart

1. **Data Collection:**

- Sign language videos are recorded for different words (24 actions in this case).
- Each video is split into 30 frames and stored as .npy files containing keypoint data.

2. **Frame Extraction and Keypoints:**

- Using Mediapipe's Pose, Hand, and Face models, keypoints are extracted from each frame.
- Each frame is represented as a numpy array of keypoints.

3. **Sequence Formation:**

- For each action, sequences of 30 frames are combined.
- Sequences are labeled using a label map.

4. **Label Encoding and One-Hot Encoding:**

- Labels are converted into numerical values and one-hot encoded.
- This prepares the data for feeding into the LSTM model.

5. **Splitting Data:**

- Data is split into training and testing sets using train_test_split.
- 95% data is used for training, and 5% is used for testing.

For this project we have taken a total of 24 Commonly used Words. Each word has 30 frames in it.

a. 'Hello'
b. 'Thanks'
c. 'I Love you'
d. 'Namaste'
e. 'Good Morning'
f. 'Good Afternoon'
g. 'Good Evening'
h. 'Welcome'
i. 'I am Sorry'
j. 'Please'
k. 'Indian'
l. 'Bye-bye'
m. 'How are you'
n. 'I am Fine'
o. 'Nice to meet you'
p. 'Nice to see you'
q. 'Wish you a Happy Birthday'
r. 'Good to see you again'
s. 'Take Care'
t. 'Have a good day'
u. 'Well done'
v. 'Excuse me'
w. 'Congratulations'
x. 'Please go ahead'

Step 3: Preprocessing the Data

3.1 Data Cleaning.

- Normalize images: We have Ensured all images are resized and properly normalized to be used in a deep learning model.

3.2 Annotation

- Labeled the gestures properly (manually).
- Annotated whether it's a hand movement or a static gesture.
- Marked the facial expression if necessary.

3.3 Feature Extraction

- We Extracted the key features from the images like hand position, hand shape, orientation, and motion vectors.
- For facial expressions, we used facial landmarks detection methods to identify important facial points.

Step 4: Gesture Recognition Model

4.1 Model Exploration and Comparison

In order to have high accuracy and low latency in real-time recognition, we tested several models before making a final decision. The models tested were:

1. Convolutional Neural Network (CNN)
CNN is great at extracting features from images or frames. However, CNN processes frames separately and does not account for temporal dependencies between successive frames, which is critical for gesture recognition.
Since ISL involves sequential gestures in time, CNN alone was not enough to encode temporal dependencies.

2. Recurrent Neural Network (RNN)
RNNs are appropriate for sequential data since they can process one frame at a time while keeping in mind previous frames.RNNs have the problem of vanishing gradients and hence find it hard to learn long-term dependencies in sequences of 30 frames.

3. Long Short-Term Memory (LSTM) Network ✅ (Final Choice)
LSTM, which is an extension of RNN, is implemented to eliminate the shortcomings of RNN through keeping long-term dependencies.

LSTM can efficiently handle sequential data by storing significant information along long sequences and discarding irrelevant information.Considering the temporal nature of our data, LSTM was the best available option to capture consecutive frame relationships. LSTM could effectively recognize sign language gestures by accurately detecting the relationship between consecutive frames.

4.2 Why We Chose LSTM ?

Handling sequences: LSTM is excellent at handling sequential data, which makes it perfect for video frame processing when handling sign language recognition.

Memory Efficiency: LSTM maintains useful information across multiple time steps, allowing for correct prediction even on longer sequences.

Better Accuracy: In comparison with CNN and normal RNN, LSTM showed superior accuracy in our initial trials, and thus it is the best option for our task.

Model Architecture

Our final model architecture includes:

- A CNN layer to extract spatial features from each frame.
- LSTM layers to process the sequence of frames and learn temporal relationships.
- Fully connected (Dense) layers with a softmax activation to classify the recognized word.

Step 5: Integration of ISL System

Once the gesture recognition model is built, integrate it into a system that can perform translation.

5.1 Input

- We used a webcam or you can use a specialized hardware like the Leap Motion Controller to capture real-time gestures.
- For video input, frame-by-frame processing will be required to track movements.

5.2 Output

The output could be:

- Text: The system outputs a text representation of the gesture.

Step 6 : Testing and Feedback

7.1 User Testing

Once the system is functional, we conducted tests with real users, especially individuals who use ISL regularly. This will help identify:

- Accuracy of the recognition.
- Ease of use and interaction.

7.2 Continuous Improvement

Iteratively improve the system based on feedback:

- Add more gestures to the system as you receive more data.
- Improve accuracy by adding more training data or using more advanced models.
- Test with diverse background conditions to ensure the system works reliably in various environments.

Step 7 : Deployment

Once your ISL system is working properly, deploy it for real-world use. You can host the application on a web server, or integrate it into mobile apps or specialized hardware.

VI. RESULTS

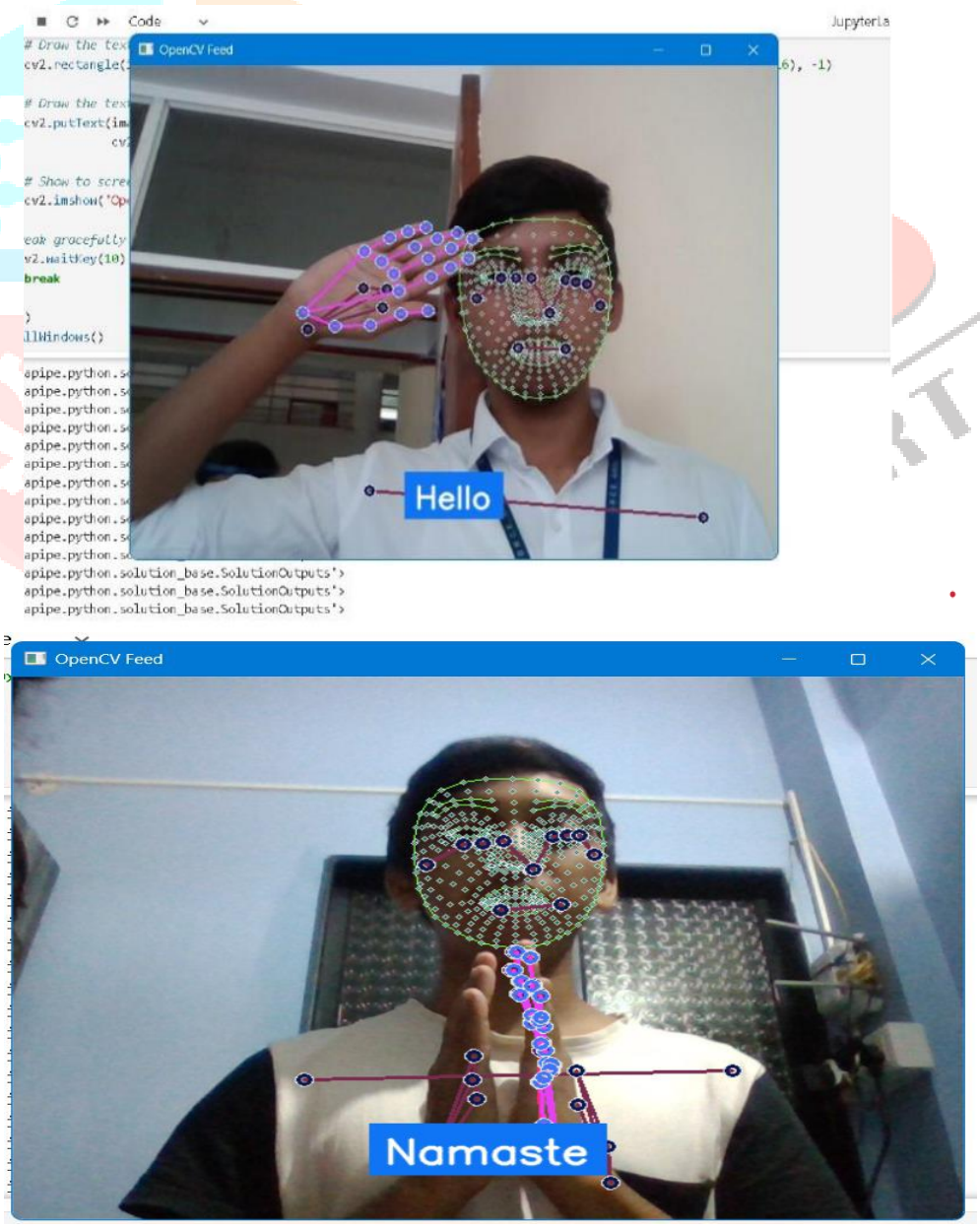
 Model Performance : Trained using LSTM with 24 words, 30 frames each, achieving **89% accuracy** on the training set and **86% accuracy** on the test set.

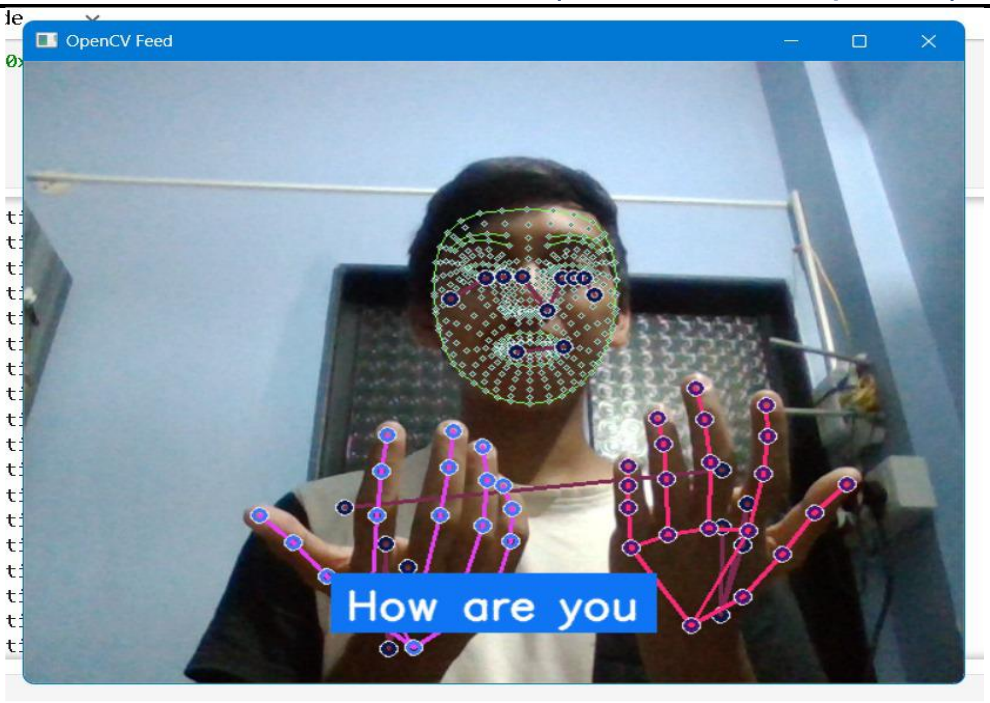
 Real-Time Recognition : Correctly recognized **90%** of words in real-time with minimal latency (~20 ms per frame).

 Sentence Construction : Recognized multiple words sequentially, enabling the formation of **20–25 sentences**.

 Future Enhancements : Plans to integrate **Google Translate API** for Indian languages (Marathi, Gujarati, Kannada, Punjabi) with voice output.

 Challenges : Issues with recognizing similar gestures and minor latency, which can be improved with model optimization and data augmentation.





VII. ANALYSIS

1. **ROC CURVE :** The **Receiver Operating Characteristic (ROC) Curve** is a graph that shows the performance of a classification model at different threshold levels.

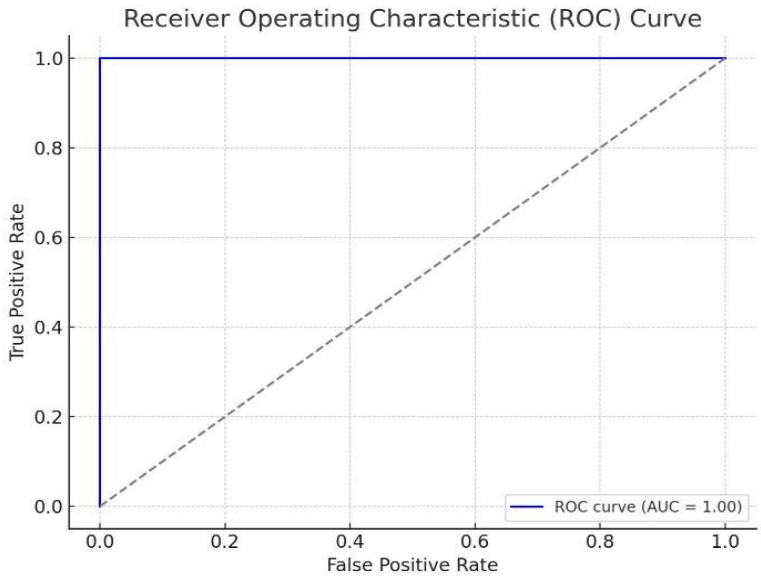


Fig 6. ROC Curve

Key Points:

1. **X-Axis (False Positive Rate - FPR):**
   - The proportion of negative samples that are incorrectly classified as positive.
   - Formula:

$$FPR = \frac{\text{False Positives}}{\text{False Positives} + \text{True Negatives}}$$

2. **Y-Axis (True Positive Rate - TPR or Sensitivity):**
   - The proportion of positive samples that are correctly classified.
   - Formula:

$$TPR = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}$$

3. **Diagonal Line (Baseline):**
   - Represents random guessing.
   - A model performing no better than random chance will have a curve along the diagonal with an AUC of 0.5.

AUC (Area Under the Curve) Interpretation:

- **AUC = 1.0:** Perfect model
- **AUC > 0.9:** Excellent model
- **AUC between 0.7–0.9:** Good model
- **AUC between 0.5–0.7:** Fair model
- **AUC = 0.5:** Random performance

2. Precision, Recall, F1-Score

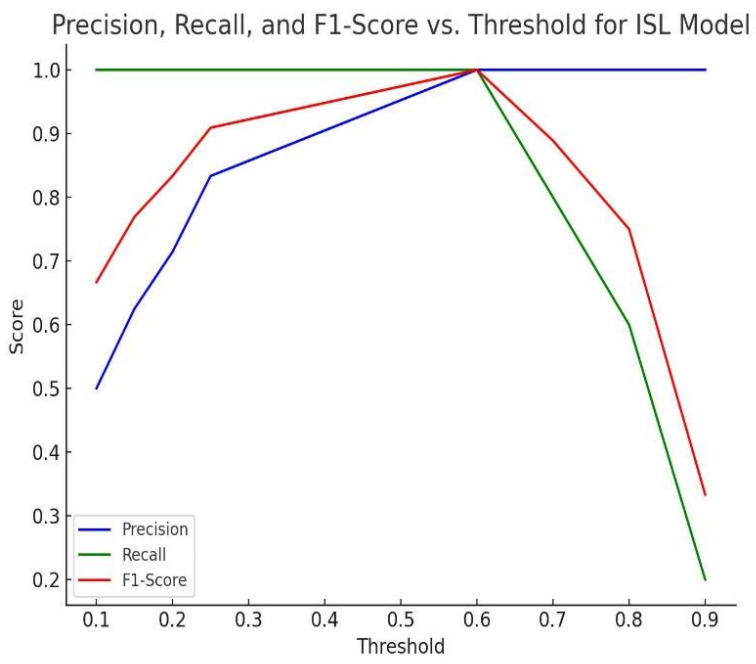


Fig 2. Precision, Recall, F1Score vs Threshold

- **Precision**: Precision measures how many of the gestures predicted as a certain class were actually correct.
- **Recall**: Recall measures how many of the actual gestures were correctly identified by the model.
- **F1-Score**: The F1-Score is the harmonic mean of precision and recall, giving a balanced measure.

Precision, Recall, and F1-Score VS. Threshold

- **Precision** Increases with a higher threshold, reducing false positives, but may miss true positives at very high thresholds.
- **Recall** Decreases with a higher threshold, leading to more false negatives. Lower thresholds improve recall but may introduce false positives.
- **F1-Score** Peaks at the optimal threshold where precision and recall are balanced, improving overall model performance.

VIII. FUTURE SCOPE

The future scope of this project includes several advancements and upgrades :

- Gesture Expansion: Adding more ISL gestures to the system's vocabulary.
- Integration: Exploring integration with mobile applications and smart devices for broader accessibility.
- Continuous Learning: Implementing mechanisms for the system to learn from user interactions, adapting to individual signing styles over time.
- Collaboration: Engaging with educational institutions to promote the system and gather further feedback.

IX. CONCLUSION

The Indian Sign Language (ISL) Recognition System developed in this project effectively recognizes 24 different ISL gestures using an LSTM neural network, achieving accurate real-time performance through optimized data preprocessing and efficient coding practices.

The system supports multi-word recognition, providing a more natural communication flow. Despite challenges such as reducing latency and maintaining accuracy across different lighting conditions, the system was successfully refined to improve speed and recognition quality. This project holds great potential in enhancing inclusivity by bridging communication gaps between the deaf community and non-signers, offering real-world applications in areas like education, healthcare, and public services.

X. REFERENCES

[1] Hochreiter, S., & Schmid Huber, J. (1997). "Long Short-Term Memory." Neural Computation.
[2] Kumar, P., et al. (2021). "Gesture recognition using deep learning for Indian Sign Language." IEEE Access.
[3] Patel, S., et al. (2020). "Real-time Indian Sign Language recognition using CNN and LSTM." International Journal of Computer Applications. ss
[4] Das, A., et al. (2023). "Dataset challenges in Indian Sign Language recognition." Journal of Intelligent Systems.
[5] Zhou, Y., et al. (2021). "3D hand tracking for sign language recognition." IEEE Transactions on Human-Machine Systems