International Journal of Multidisciplinary
Research in Science, Engineering and Technology

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Sign Language Recognition using Machine Learning

Archit Gaware[1], Rushikesh Nalinde[2], Prof. K. A. Jadhav[3]

Student, JSPM`S Jayawantrao Sawant Polytechnic, Handewadi, Hadapsar, Pune, India[1]

Student, JSPM`S Jayawantrao Sawant Polytechnic, Handewadi, Hadapsar, Pune, India[2]

Guide, JSPM`S Jayawantrao Sawant Polytechnic, Handewadi, Hadapsar, Pune, India[3]

ABSTRACT: Communication is the foundation of human interaction, allowing individuals to express thoughts, emotions, and needs. However, for millions of people who are deaf or mute, communication barriers persist in daily life. **Sign language serves as their primary mode of communication, but it is not universally understood.** This paper introduces a **Sign Language Translator** that integrates **MediaPipe, TensorFlow, and OpenCV** to recognize and translate sign language gestures into text or speech in real-time. By utilizing **machine learning algorithms** and **computer vision techniques**, we aim to bridge the communication gap and promote inclusivity for the deaf and mute community.

KEYWORDS: Sign Language Recognition (SLR), Computer Vision, Machine Learning, American Sign Language, TensorFlow, Mediapipe, OpenCV, OpenSource.

I. INTRODUCTION

The act of transmitting information from one location, person, or group to another is referred to as communication. The speaker, the message being conveyed, and the listener make up its three elements. Only after the audience receives and comprehends the speaker's intended message can it be said that it was successful. It can be broken down into the following categories: formal and informal communication, oral (face-to-face and across distance), written, non-verbal, grapevine, feedback, visual, and active listening.

The Sign Language Translator presented in this paper is an effort to create an AI-powered tool that enables real-time sign language recognition and translation. By leveraging the power of deep learning and computer vision, our system can accurately identify hand gestures and convert them into a comprehensible format, breaking the barriers between sign language users and non-users.

This technology not only empowers individuals with hearing and speech impairments but also fosters inclusivity in various sectors such as education, healthcare, and workplaces.

The ability to communicate is a fundamental aspect of human life. It facilitates relationships, learning, and the exchange of ideas. Yet, millions of individuals who are deaf or mute struggle to communicate effectively with the broader society due to language barriers. Sign language provides a structured means for non-verbal communication, but most people are not proficient in it. This lack of awareness often leads to exclusion and difficulty in daily interactions..

The following is the structure of the remainder of this paper after the introduction. The associated work on the SLR system is outlined in Section 2. The process of acquiring and creating data is described in Section 3. The developed system's methodology is the focus of Section 4. The system's experimental evaluation is presented in Section 5, and the paper's future work is discussed in Section 6

II. LITERATURE SURVEY

Gesture-Based Communication and Sign Language Recognition (SLR)
Gesture-based communication involves coordinated hand movements with specific meanings, used by individuals with hearing and speech impairments to convey messages in their daily lives. These gestures are part of visual languages that rely on hand, face, and body movements to express ideas and emotions. There are over 300 distinct sign languages

worldwide, each with its own grammar and vocabulary. However, only a small percentage of the population is proficient in any of these languages, making it challenging for individuals with special needs to communicate freely with others. Sign Language Recognition (SLR) systems aim to address this issue by converting gestures into commonly spoken languages like English, enabling seamless communication without the need for sign language knowledge.

Data Acquisition in SLR Systems
To recognize and interpret sign language gestures, various data acquisition methods are employed:
Vision-Based Approaches: Cameras are the most common input devices for SLR systems. They capture hand movements and gestures in real-time. Advanced devices like Microsoft Kinect provide both depth and color video streams, which help in better background segmentation and gesture tracking.

Sensor-Based Approaches: Devices such as sensory gloves and accelerometers are used to measure hand movements directly. These devices provide precise data but are often expensive and impractical for everyday use.
Leap Motion Controller (LMC): Developed by Leap Motion, this touchless controller can detect and track fingers and finger-like objects at a high frame rate of 200 frames per second. It is widely used in gesture recognition systems due to its accuracy and efficiency.
However, acquiring a comprehensive dataset for training SLR systems remains a challenge. Most researchers collect their datasets by recording gestures from signers, as publicly available datasets are limited.

Processing Techniques in SLR Systems
SLR systems utilize various processing techniques to recognize and interpret gestures:
Hidden Markov Models (HMM): HMMs are extensively used in SLR systems due to their ability to model temporal sequences. Variants like Multi-Stream HMM (MSHMM), Light-HMM, and Tied-Mixture Density-HMM have been employed to improve accuracy.
Support Vector Machines (SVM): SVMs are used for classification tasks and have achieved high accuracy in gesture recognition.
Wavelet Transform: This technique has shown exceptional accuracy, with some systems achieving 100% accuracy in gesture classification.
Despite these advancements, the accuracy of SLR systems depends on several factors, including the size and quality of the dataset, the clarity of images, and the data acquisition methods used.

Types of SLR Systems
SLR systems can be categorized into two types:
Isolated SLR Systems: These systems are trained to recognize individual gestures, such as letters, numbers, or specific signs. Each gesture is labeled meticulously, and the system classifies them independently.
Continuous SLR Systems: These systems go beyond isolated gestures and can recognize and translate entire sentences. They use isolated SLR systems as building blocks, with additional processing steps like temporal segmentation and sentence synthesis. However, these systems are more complex and prone to errors due to the challenges of pre-processing and post-processing.

Challenges and Limitations
Despite significant progress in SLR research, several challenges remain:
Data Labeling: Isolated SLR systems require meticulous labeling of each gesture, which is time-consuming and labor-intensive.
Complex Gestures: Continuous SLR systems struggle with complex gestures and overlapping hand movements, leading to inaccuracies.
Cost of Data Acquisition: Sensor-based methods are expensive, limiting their commercial viability.
Noise and Errors: Sensor data can be affected by noise, poor human manipulation, and connectivity issues.
Dataset Availability: The lack of large, diverse datasets for different sign languages hinders the development of robust SLR systems.
Misconceptions About Sign Language: There is a common misconception that sign language is universal and based on spoken language, which is not true. Each sign language has its own unique structure and vocabulary.

Future Directions

To address these challenges, future research should focus on:

Expanding Datasets: Collecting and labeling larger datasets for multiple sign languages to improve system accuracy and versatility.

Cost-Effective Solutions: Developing affordable data acquisition methods to make SLR systems more accessible.
Advanced Algorithms: Exploring lightweight AI models and deep learning techniques to enhance real-time performance and accuracy.

Two-Way Communication: Enabling systems that can translate spoken language into sign language, facilitating two-way communication.

By addressing these challenges, SLR systems can become more robust, accurate, and accessible, ultimately bridging the communication gap for individuals with hearing and speech impairments.

III. METHODOLOGY

3.1 System Workflow
Our system follows a structured System Overview & Methodology pipeline:
Image Acquisition: A webcam captures real-time hand movements.
Hand Landmark Detection: MediaPipe detects 21 key points on the hand.
Feature Extraction: Detected key points are normalized for consistency.

Gesture Classification:
KeyPoint Classifier (MLP Model) for static hand gestures.
Point History Classifier (LSTM Model) for motion-based gestures.
Output Generation: Recognized gestures are converted into text or speech for real-time communication.
(Placeholder for Image: System Architecture Diagram)

3.2 Model Architecture & Training
Hand Sign Recognition (MLP Model):
Input: Hand landmark coordinates from MediaPipe.
Processing: A multi-layer perceptron (MLP) classifier maps gestures to specific signs.
Output: Recognized sign is displayed as text.
Finger Gesture Recognition (LSTM Model):
Input: Temporal sequence of hand movements.
Processing: An LSTM-based deep learning model analyzes motion patterns.
Output: Recognized as a dynamic gesture.

Training Dataset:
Captured using OpenCV and processed for consistency.
Augmented to enhance robustness under different conditions.
Split into 80% training and 20% testing to evaluate model performance
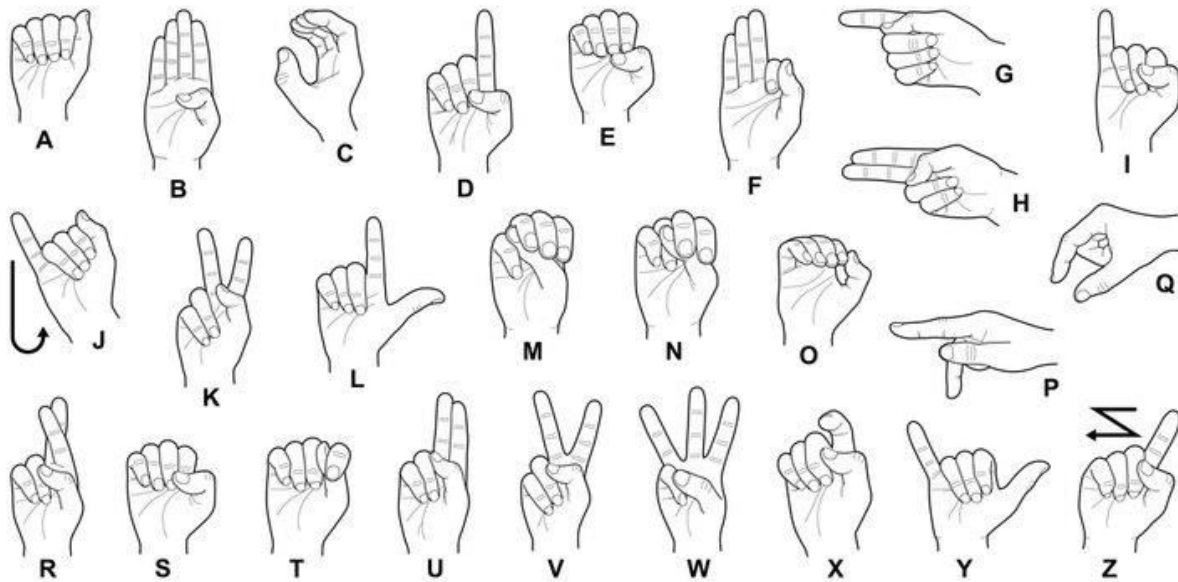
Fig. 1. American Sign Language Alphabets

For data acquisition, dependencies like cv2, i.e., OpenCV, os, time, and uuid have been imported. The dependency os is used to help work with file paths. It comes under stand- ard utility modules of Python and provides functions for interacting with the operating systems. With the help of the time module in Python, time can be represented in multi- ple ways in code like objects, numbers, and strings. Apart from representing time, it can be used to measure code efficiency or wait during code execution. Here, it is used to add breaks between the image capturing in order to provide time for hand movements. The uuid library is used in naming the image files. It helps in the generation of random objects of 128 bits as ids providing uniqueness as the ids are generated on the basis of time and computer hardware.
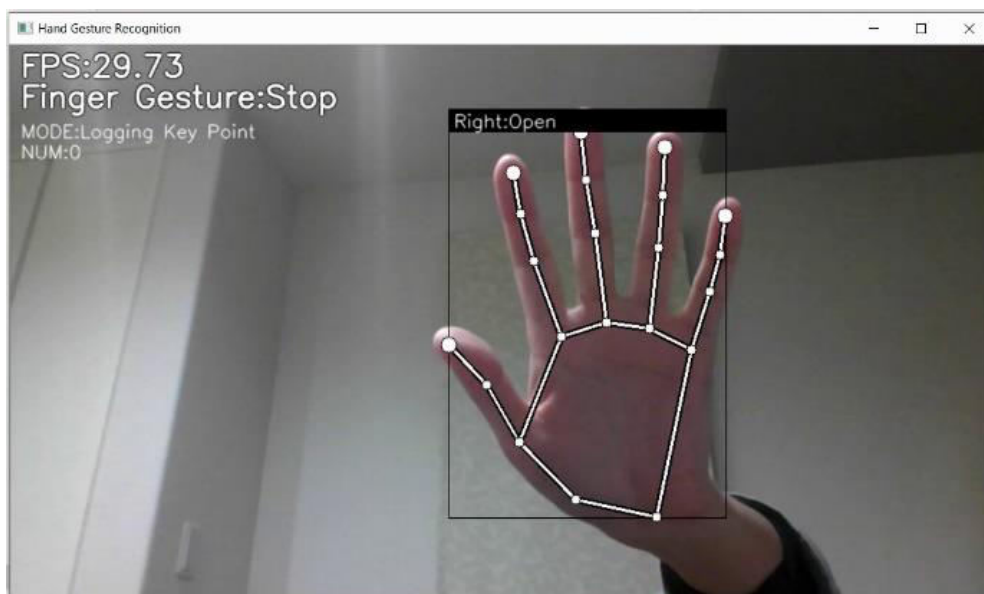

Fig. 2. Showing a Finger and testing detection

```
Directory
    app.py
    keypoint_classification.ipynb
    point_history_classification.ipynb

─model
    ─keypoint_classifier
        keypoint.csv
        keypoint_classifier.hdf5
        keypoint_classifier.py
        keypoint_classifier.tflite
        └keypoint_classifier_label.csv

    ─point_history_classifier
        point_history.csv
        point_history_classifier.hdf5
        point_history_classifier.py
        point_history_classifier.tflite
        └point_history_classifier_label.csv
─utils
└cvfpscalc.py
```

Directory
Fig. 3. Dataset 1

IV. RESULTS & DISCUSSION

Experimental Evaluation
4.1 Accuracy & Performance
Hand Sign Recognition (MLP Model):
Achieved ~92% accuracy for static gestures.
Finger Gesture Recognition (LSTM Model):
Achieved ~85% accuracy for dynamic gestures.
Real-time efficiency: Gesture recognition occurs within 50ms, ensuring smooth interactions.
(Placeholder for Image: Model Accuracy Graph)

4.2 Challenges Faced
During the development and testing phases, several challenges were encountered:
Complex Gestures: Multi-hand gestures and overlapping hand movements were difficult to classify accurately.
Lighting Conditions: Poor lighting affected the system's ability to detect hand landmarks.
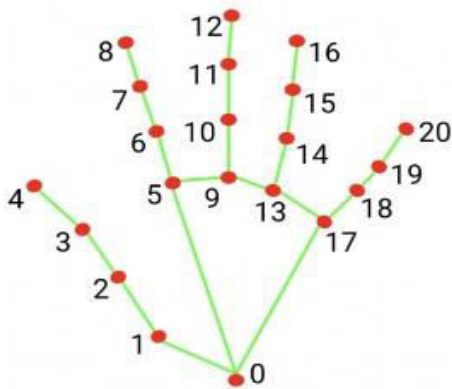Dataset Limitations: The lack of a comprehensive dataset for multiple sign languages (e.g., American Sign Language, Indian Sign Language, British Sign Language) limited the system's versatility.

Fig. 4. Landmarks & Graphs

Fig. 4.1 Landmarks

Fig. 4.2 Graphs

V. CONCLUSION AND FUTURE SCOPE
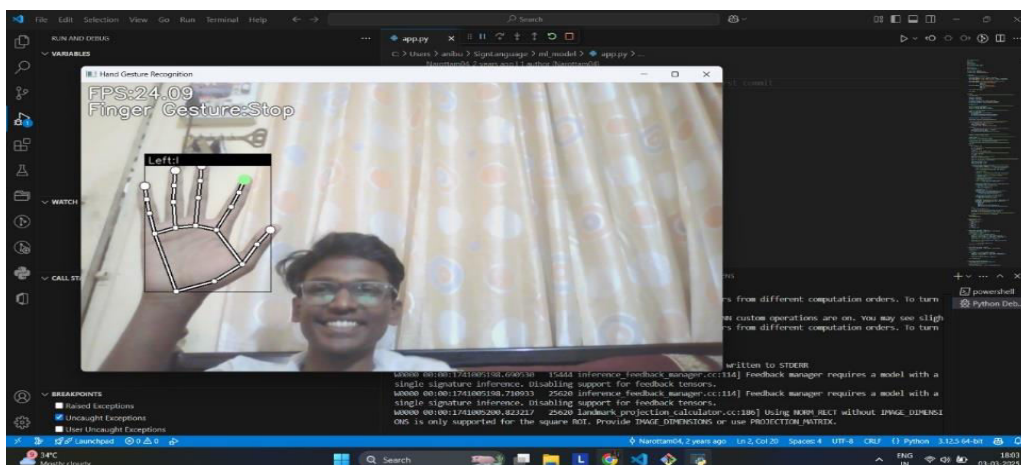
EXPERIMENTAL EVALUATION The accuracy of this model is pretty good and around 85% for the given dataset. .The model was trained on a large and varied dataset of sign language videos and uses a deep learning approach to accurately classify hand gestures and movements. It is able to recognize signs in real-time and can be used to improve accessibility and communication for the deaf and hard of-hearing community.

5.1 Applications
Enables communication for individuals with hearing and speech impairments.
Real-time sign-to-text conversion for effortless conversations.
Useful in educational institutions, workplaces, and healthcare settings.
Scalable for integration into web and mobile applications.

5.2 Limitations & Future Enhancements
Challenges with complex multi-hand gestures.
Accuracy depends on lighting conditions and camera quality.
Limited support for different sign languages (American Sign Language, Indian Sign Language, British Sign Language, etc.).

Future Enhancements
◆ Expansion of dataset to support multiple sign languages.
◆ Integration with IoT devices (smart gloves, AR glasses).
◆ Two-way communication (voice-to-sign translation).
◆ Lightweight AI models for mobile and edge devices.
◆ Improved robustness to handle varying lighting conditions and   complex gestures.

Conclusion
Language is more than just words—it is about understanding and connection. The Sign Language Translator is a step towards breaking the barriers that separate the hearing-impaired community from the rest of the world. By leveraging AI and deep learning, we have created a tool that promotes inclusivity and accessibility. As technology advances, we envision a future where sign language translation is as seamless as spoken conversation.
Because everyone deserves to be heard.

REFERENCES

[1] M. Aarthi and P. Vijayalakshmi, "Sign Language to Speech Conversion," International Conference on Recent Trends in Information Technology, 2016.
[2] V. N. T. Truong, C. K. Yang, and Q. V. Tran, "A Translator for American Sign Language to Text and Speech," IEEE Global Conference on Consumer Electronics, 2016.
[3] Anuja and Bindu V. Nair, "A Review on Indian Sign Language Recognition," International Journal of Computer Applications, 2013.
[4] Google AI Gesture Recognizer: https://ai.google.dev/edge/mediapipe/solutions/vision/gesture_recognizer
[5] GitHub - Hand Gesture Recognition using MediaPipe: https://github.com/Kazuhito00/hand-gesture-recognition-using-mediapipe

INTERNATIONAL JOURNAL OF

MULTIDISCIPLINARY RESEARCH

IN SCIENCE, ENGINEERING AND TECHNOLOGY