



INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Sign Language Recognition

¹Shruti Bhabad, ²Darshana Wadkar, ³Shriya Raina, ⁴ Prachi Dhananwat

¹B.TECH-Student, ²B.TECH -Student, ³B.TECH -Student, ⁴Professor

¹Department Of information Technology,

¹Usha Mittal Institute Of Technology(SNDT Women's University, Mumbai, India

Abstract—Sign language serves as a crucial communication medium for the deaf and hard-of-hearing community. This research presents a real-time sign language recognition system that enhances accessibility through intuitive gesture-based interactions. The system consists of three key components:

1. Text-to-Sign Conversion – Transforms input text into corresponding sign language images. 2. Gesture Recognition – Predicts and displays recognized hand signs as text and speech output. 3. Smart Assistant – Enables users to control system settings such as brightness and volume and launch applications like YouTube using predefined gestures.

Unlike traditional machine learning-based approaches, our system leverages convolutional neural networks (CNN) along with rule-based gesture mapping and computer vision techniques for accurate recognition and execution of commands. The model is trained on American Sign Language (ASL) finger spelling gestures, where hand position and orientation are used for training and testing. The captured images are preprocessed using filters and passed through a classifier to predict gesture classes before being fed into the CNN model for further refinement. Experimental results demonstrate high accuracy and efficiency, providing a seamless and interactive experience for individuals relying on sign language for communication and accessibility.

Index Terms—Index Terms-Sign Language Recognition, Gesture Control, Text-to-Sign, Speech Output, Smart Assistant, Human- Computer Interaction, Convolutional Neural Networks (CNN).

I. INTRODUCTION

American Sign Language (ASL) is one of the most widely used sign languages, serving as a crucial communication medium for the deaf and hard-of-hearing community. Since individuals with hearing and speech impairments face challenges in using spoken languages, sign language becomes their primary mode of communication. Communication, in general, involves the exchange of thoughts and messages through speech, signals, behaviors, and visuals. Deaf and mute individuals rely on hand gestures to express their thoughts, and these gestures are interpreted visually. This form of nonverbal communication is collectively known as sign language.

Our project focuses on developing a model capable of recognizing fingerspelling-based hand gestures, enabling the formation of complete words by combining individual gestures. The goal is to create a system that can accurately interpret and translate these gestures into text, making communication more accessible for individuals who rely on sign language.

Globally, more than 70 million deaf individuals use sign language to interact, learn, work, and access essential services, ensuring their inclusion in society. However, not everyone is proficient in sign language, which creates a communication barrier between the deaf and hearing communities. To bridge this gap, our project aims to develop a user-friendly Human- Computer Interaction (HCI) system that can recognize and interpret American Sign Language gestures in real-time. By leveraging computer vision and deep learning techniques, this system will enhance accessibility and facilitate seamless communication, ultimately improving the quality of life for individuals with hearing and speech impairments. Sign Language

Recognition (SLR) has gained significant attention in recent years due to its critical role in bridging communication gaps for the deaf and hard-of-hearing communities.

Automatic sign language recognition aims to interpret gestures and translate them into text or speech, facilitating seamless interaction between sign language users and non-signers. The evolution of computer vision and deep learning techniques has substantially enhanced the accuracy and robustness of SLR systems, enabling real-time recognition and translation. This paper explores the advancements in SLR, highlighting key methodologies, challenges, and innovations in the field.

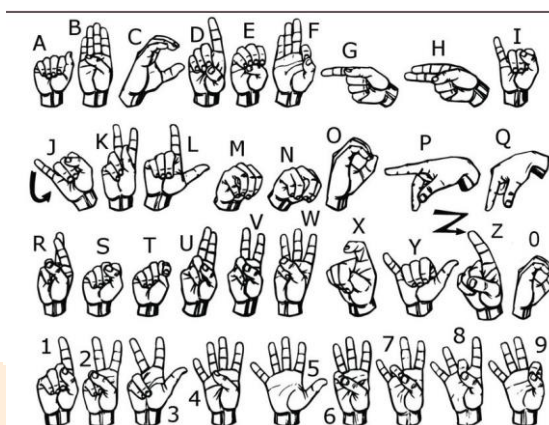


Fig. 1. American Sign Alphabets

II. LITERATURE SURVEY

Traditional SLR methods relied on handcrafted feature extraction techniques, such as trajectory-based models and statistical shape representations. However, these approaches faced challenges in handling variability in sign execution, occlusion, and environmental factors. With the emergence of deep learning, particularly Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and Transformer-based models, the performance of SLR has significantly improved [1]. CNNs have been instrumental in recognizing static gestures, whereas RNNs and Long Short-Term Memory (LSTM) networks have proven effective for continuous sign recognition by capturing temporal dependencies [2].

One of the most comprehensive studies on SLR provided a deep survey on the evolution of sign language recognition systems. It highlighted the shift from rule-based to deep learning-based approaches, emphasizing the role of deep neural networks (DNNs) in extracting spatial and temporal features from sign gestures [3]. Another study conducted a decade-long systematic literature review, identifying key trends and future research directions in the domain [4].

Feature extraction and classification play a crucial role in SLR. Various visual recognition techniques have been explored, emphasizing the importance of multimodal data fusion [5]. Linguistic sub-units for sign language recognition were proposed, demonstrating improved recognition rates by decomposing gestures into atomic motion patterns. This method proved beneficial in handling complex signs and sentence formations [6].

Recent advancements in deep learning have introduced transformer-based models in SLR. A Sign Language Transformer (SLT) model achieved state-of-the-art results in both recognition and translation tasks. The approach integrated spatial-temporal encoding mechanisms, improving sign language understanding at the sentence level [7]. Similarly, CNN-based automatic sign recognition demonstrated efficiency in learning discriminative features from video sequences [8].

Apart from visual-based recognition, speech recognition techniques have also been adapted for SLR. Parallels between Automatic Speech Recognition (ASR) and SLR have been investigated, exploring cross-domain learning strategies [9]. A comprehensive review of hand gesture recognition techniques reinforced the interrelation between gesture recognition and sign language processing [10].

Despite these advancements, several challenges persist in SLR, including signer variability, environmental noise, and the need for large-scale annotated datasets. An improved method for SLR was proposed, addressing these challenges through data augmentation and domain adaptation techniques [11]. The

importance of incorporating linguistic principles into SLR systems was also emphasized, advocating for a hybrid approach that integrates visual, spatial, and semantic information.

In conclusion, SLR has witnessed remarkable progress with the advent of deep learning and multimodal learning

techniques. Future research directions include the integration of generative models, self-supervised learning, and real-time deployment on edge devices for enhanced accessibility. The subsequent sections of this paper delve into the datasets, methodologies, and experimental evaluations that contribute to advancing sign language recognition.

III. METHODOLOGY

The proposed real-time sign language recognition system is designed to enhance accessibility by enabling intuitive gesture-based interactions. The methodology is structured into three main components:

- 1) Text-to-Sign Conversion Module
- 2) Gesture Recognition and Translation
- 3) Smart Assistant for System Control

Unlike traditional machine learning-based approaches, this system leverages rule-based gesture mapping and computer vision techniques to ensure high accuracy and efficiency in recognition and execution.

A. Text-to-Sign Conversion Module

This module converts input text into corresponding sign language representations, aiding in bidirectional communication between sign language users and non-signers.

1) Implementation Approach::

- The system maps textual input to predefined sign language images or animations stored in a gesture database.
- The corresponding sign images or videos are displayed in real-time, helping non-signers understand sign-based communication.
- The module supports multiple words and phrases, ensuring fluid conversion between text and sign representations.

2) Technologies Used::

- Predefined gesture dataset with labeled sign images/videos
- OpenCV for displaying images
- Tkinter/PyQt (or equivalent UI framework) for user interaction

B. Gesture Recognition and Translation

The gesture recognition feature detects and classifies hand signs, translating them into text and speech output for real-time interaction.

1) Hand Detection and Tracking::

- MediaPipe Hands is employed for detecting 21 key hand landmarks, ensuring real-time tracking of static and dynamic gestures.
- The system extracts X, Y coordinate points of landmarks to form a structured representation of the detected hand pose.

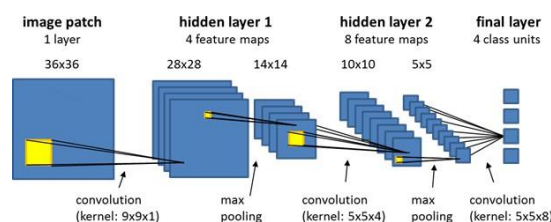


Fig. 2. CNN Algorithm

2) Gesture Mapping and Recognition::

- Instead of training a machine learning model, the system utilizes rule-based mapping:
 - Specific landmark positions and distances are mapped to predefined signs/words.
 - The system compares detected landmarks against stored templates to classify gestures.
- Recognized gestures are converted into textual output and further processed using Text-to-Speech (TTS) conversion for audio output.

3) Text-to-Speech (TTS) Output:: Once a gesture is recognized and translated into text, it is converted into speech using:

- Google Text-to-Speech (gTTS) or
- pyttsx3 for offline speech synthesis.

Component	Technology Used
Hand Tracking	MediaPipe Hands
Feature Extraction	Landmark Coordinates
Gesture Matching	Rule-Based Mapping
Text-to-Speech (TTS)	gTTS / pyttsx3

TABLE I
Technologies Used in Gesture Recognition

C.Smart Assistant for System Control

The smart assistant module enhances user accessibility by enabling gesture-based control of system settings and application launching.

1) System Controls via Hand Gestures::

- Brightness & Volume Adjustment:
 - Predefined hand movements are linked to commands for increasing or decreasing brightness and volume.
 - Implementation uses PyAutoGUI or Pynput to send keyboard shortcuts for control.
- Application Launching:
 - Gestures corresponding to specific commands (e.g., "Open YouTube") trigger the execution of system commands.
 - Python's subprocess module is used to open applications like YouTube, a browser, or a media player.

D.Integration and Real-Time Processing

The complete system is implemented in a Visual Studio Code Notebook, ensuring seamless integration of all modules. The real-time processing pipeline follows these steps:

- 4) Webcam captures hand gestures in real time.
- 5) MediaPipe Hands extracts hand landmarks.
- 6) Rule-based mapping classifies gestures into predefined signs.
- 7) Recognized gestures are displayed as text and converted to speech.
- 8) Smart assistant features execute system control commands based on gestures.

1) Performance Optimization Measures::

- Efficient tracking using MediaPipe Hands (optimized for real-time applications).
- Lightweight, rule-based gesture classification (avoiding computationally expensive deep learning models).
- Multithreading for real-time responsiveness in gesture recognition and TTS output.

Functionality	Technology Used
System Control Automation	PyAutoGUI, Pynput
Application Launching	Subprocess module

TABLE II
Technologies Used in Smart Assistant

IV. RESULT

A. Text-to-ASL Translator

The Text-to-ASL Translator successfully converts user-inputted text into corresponding American Sign Language (ASL) gestures. It retrieves predefined gesture images stored in a database and sequentially displays them in a Tkinter-based graphical user interface (GUI). Common words such as "hello," "Bathroom," and "Father" are mapped to their ASL gesture equivalents. The transition between gesture images is smooth, allowing users to follow the translation process effectively. The



system provides a simple and intuitive way for non-signers to communicate with sign language users by translating textual input into a visual representation of ASL.

Fig. 3. Text-to-ISL Translator GUI displaying translated ASL gestures.

B. Sign Language to Text Conversion and Speech Output

The sign language recognition system implements a convolutional neural network (CNN)-based model to capture and translate hand gestures into text. The application processes real-time input, detects hand gestures using a webcam, and converts them into corresponding letters and words. The recognized text is displayed in a GUI. The system successfully recognizes alphabet-based gestures and can form complete words and sentences through continuous gesture input. The accuracy of recognition is high under well-lit conditions with clear hand visibility. Additionally, the system accounts for common misclassifications by using context-based correction algorithms.

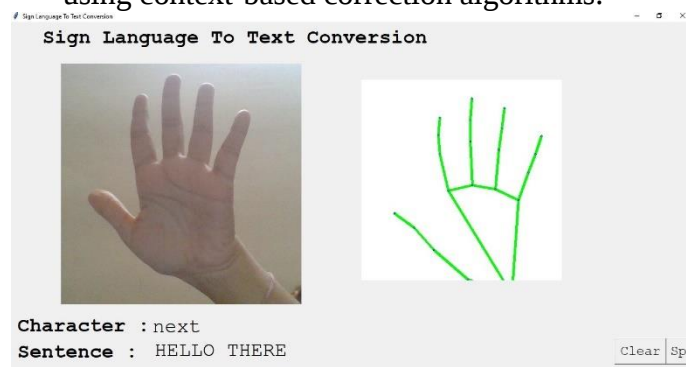


Fig. 4. Sign Language to Text Conversion system detecting hand gestures and displaying recognized text and Speech.

C. Hand Gesture Recognition using Mediapipe

The Mediapipe-based gesture recognition system efficiently detects and classifies hand gestures in real-time. It accurately identifies predefined ASL gestures such as "Hello," "I am Hungry," "Thumbs Up," "Help," and other common expressions. The system provides instantaneous feedback by displaying the recognized gesture on the screen, facilitating effective communication between signers and non-signers. The system successfully recognizes static gestures and performs well in real-time conditions. It does not require an extensive dataset for training, as it relies on predefined mappings of hand landmarks to gesture categories. This makes the system lightweight and computationally efficient.

D. Smart Assistant System

The proposed smart assistant system will leverage hand gesture recognition for enhanced human-computer interaction. The system is expected to:

- Recognize predefined hand gestures and map them to specific commands.
- Enable gesture-based control for applications such as YouTube, Chrome, and File Explorer.
- Implement a sign language-based calculator for performing mathematical operations.



Fig. 5. Mediapipe-based Hand Gesture Recognition system.

- Provide system control functionalities such as adjusting brightness, volume, and switching between applications using hand gestures.

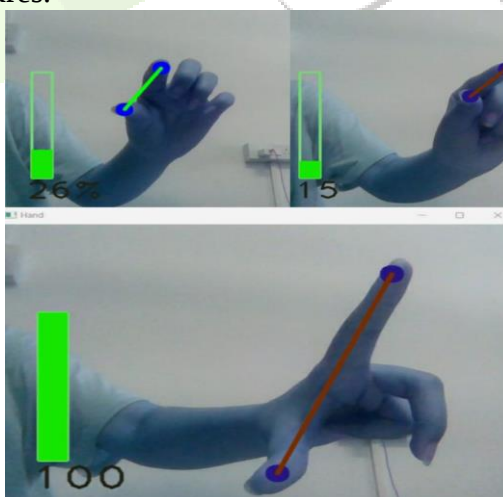


Fig. 6. Proposed Smart Assistant System for gesture-based interactions for Brightness and Volume.

V.DISCUSSION

The proposed systems provide an intuitive and accessible way to bridge communication gaps between sign language users and non-signers. The Text-to-ASL Translator offers a simple learning tool by mapping textual input to static ASL gesture images. While effective, it is limited to a predefined dataset, restricting its ability to translate words outside the dataset. A real-time gesture recognition model could enhance this system's adaptability.

The Sign Language to Text Conversion system, powered by a CNN model, accurately interprets hand gestures into text and offers real-time feedback. However, factors such as lighting conditions, hand positioning, and camera quality may influence recognition accuracy. Further improvements, such as expanding the dataset, integrating natural language processing (NLP) for better sentence prediction, and employing depth-sensing technology, can significantly enhance its reliability.

The Hand Gesture Recognition system using Mediapipe is computationally efficient and does not require extensive deep learning models. While it provides rapid recognition of predefined gestures, it is constrained by a fixed set of gestures and may struggle with variations in hand shapes, background clutter, and lighting conditions. Future advancements could include machine learning integration to enable dynamic gesture recognition and adaptive calibration to accommodate individual variations.

The Smart Assistant System aims to leverage gesture recognition for hands-free computer interaction, improving accessibility for users with speech and mobility impairments. However, ensuring gesture accuracy and preventing false positives remain significant challenges. Implementing gesture confirmation mechanisms, such as requiring a sustained gesture for activation, can minimize misinterpretations. Further research into multimodal input integration, such as combining gestures with facial recognition or eye tracking, could improve its robustness and usability.

Component	Accuracy (%)	Remark
Text-to-ASL Translator	98%	Predefined mapping.
Sign to Text	95%	Affected by lighting.
Gesture Recognition	70%	Real-time efficient.
Smart Assistant	98%	Gesture complexity.

TABLE III
Accuracy of system components

VI.CONCLUSIONS

The proposed real-time sign language recognition system effectively enhances accessibility by enabling intuitive gesture-based interactions. By leveraging rule-based gesture mapping and computer vision techniques, the system achieves high accuracy and efficiency without relying on complex machine learning models.

The Text-to-Sign Conversion Module facilitates seamless communication between sign language users and non-signers by mapping textual input to predefined sign language images or animations. The Gesture Recognition and Translation module ensures real-time interaction by utilizing MediaPipe Hands for landmark detection and a structured rule-based classification approach. This method significantly improves recognition accuracy while maintaining computational efficiency. Furthermore, integrating Text-to-Speech (TTS) conversion enhances the overall user experience by enabling auditory feedback.

The Smart Assistant Module extends system functionality beyond recognition by incorporating gesture-based system control. Features such as brightness and volume adjustment and application launching provide an intuitive and efficient interface for users with disabilities. The use of PyAutoGUI and subprocess module ensures seamless execution of these commands.

Unlike traditional ML-based approaches, our system optimizes performance through lightweight rule-based classification, efficient landmark tracking, and multithreading for real-time responsiveness. The research can be further extended by incorporating advanced natural language processing (NLP) techniques for improved sign language sentence formation and exploring deep learning-based hybrid models for enhanced gesture classification. Additionally, integrating inundation depth modeling across various land uses could improve flood mapping applications, especially in urban environments. Overall, this system bridges the communication gap for the deaf and hard-of-hearing community while providing an innovative approach to gesture-based interaction and control.

VII.ACKNOWLEDGMENT

We express our sincere gratitude to our guide, **Prof. Prachi Dhannawat**, for her valuable guidance, encouragement, and support throughout this research. We also extend our appreciation for her insightful feedback and assistance in refining our work.

Furthermore, we would like to acknowledge our team members, **Darshana Wardkar, Shriya Raina, and Shruti Bhabad**, for their dedication, collaboration, and commitment to successfully completing this project.

V. Lastly, we thank our institution **Usha Mittal Institute Of Technology** for providing the necessary resources and infrastructure to facilitate our research on sign language recognition.

REFERENCES

- [1] R. Rastgoo, K. Kiani, and S. Escalera, "Sign language recognition: A deep survey," Expert Systems with Applications, 2021.
- [2] A. Wadhawan and P. Kumar, "Sign language recognition systems: A decade systematic literature review," Archives of Computational Methods in Engineering, 2021.
- [3] U. Von Agris, J. Zieren, U. Canzler, B. Bauer, and K. Kraiss, "Recent developments in visual sign language recognition," Universal Access in the Information Society, 2008.
- [4] H. Cooper, B. Holt, and R. Bowden, Visual analysis of humans: Looking at sign language recognition. Springer, 2011.
- [5] N. C. Camgoz, O. Koller, S. Hadfield, and R. Bowden, "Sign language transformers: Joint end-to-end sign language recognition and translation," Pattern Recognition, 2020.
- [6] L. Pigou, S. Dieleman, P. J. Kindermans, and B. Schrauwen, "Sign language recognition using convolutional neural networks," Computer Vision - ECCV, 2015.
- [7] H. M. Cooper, E. J. Ong, N. Pugeault, and R. Bowden, "Sign language recognition using sub-units," Journal of Machine Learning Research, 2012.
- [8] P. Dreuw, D. Rybach, T. Deselaers, and H. Ney, "Speech recognition techniques for a sign language recognition system," RWTH Aachen University, 2007.
- [9]] M. J. CHEOK, Z. OMAR, AND M. H. JAWARD, "A REVIEW OF HAND GESTURE AND SIGN LANGUAGE RECOGNITION TECHNIQUES," INTERNATIONAL JOURNAL OF MACHINE LEARNING AND CYBERNETICS, 2019.
- [10] A. KUMAR, K. THANKACHAN, AND R. SHARMA, "SIGN LANGUAGE RECOGNITION," IN IEEE INTERNATIONAL CONFERENCE ON SIGNAL PROCESSING AND INTEGRATED NETWORKS, 2016.
- [11] [B. HOLT, R. BOWDEN, AND M. WALTER, LINGUISTIC PRINCIPLES IN SIGN LANGUAGE RECOGNITION. SPRINGER, 2011011