IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)
An International Open Access, Peer-reviewed, Refereed Journal

Deafy : AI-Powered Sign Language Communication Bridge

[1]Anuradha Popat Patil, [2]Piyusha Santosh Awate, [3]Sakshi Ajit Koganole, [4]Sanjana Sambhaji Magdum, [5]Pravin Karve
[1]Student, [2]Student, [3]Student , [4]Student, [5]Faculty
[1]Sharad Institute of Technology College of Engineering,
[2]Sharad Institute of Technology College of Engineering,
[3]Sharad Institute of Technology College of Engineering,
[4]Sharad Institute of Technology College of Engineering,
[5]Sharad Institute of Technology College of Engineering

Abstract

The communication gap between hearing and speech-impaired individuals and the general population remains a major challenge in inclusive human interaction. To address this issue, we present **Echo Sign**, a bidirectional communication system that translates **sign language into audio** and **audio into sign language** in real time. The system is implemented using a **Python-based backend** integrating **machine learning models** for gesture recognition and **speech synthesis** for audio generation. The architecture employs **computer vision** for hand gesture detection, **speech-to-text** processing for audio input, and **3D animated sign rendering** for sign output. Echo Sign successfully bridges the accessibility divide by enabling smooth, natural communication between differently-abled and non-disabled users. The paper discusses the **system architecture, functionality, and integration** of multiple AI components within a unified framework. Future enhancements aim to improve model precision, expand gesture datasets, and deploy Echo Sign as a **cross-platform application** for real-world accessibility and scalability.

Keywords - Sign Language Recognition, Speech Synthesis, Bidirectional Communication, Machine Learning, Computer Vision, Accessibility, Human–Computer Interaction.

I. Introduction

Communication is a fundamental aspect of human life, yet individuals with hearing and speech impairments often face significant barriers in expressing themselves and understanding others. Sign language serves as a vital medium of interaction for the hearing and speech-impaired community. However, its comprehension is generally limited to those trained in it, thereby restricting inclusive communication. This gap creates challenges in education, employment, and daily social interactions, emphasizing the need for technological solutions that can bridge this divide. Despite ongoing research in assistive communication, most existing systems are either unidirectional, complex to operate, or lack real-time responsiveness.

To address these limitations, we have developed **Echo Sign**, an intelligent **bidirectional communication system** that enables seamless translation between **sign language and audio speech**. The primary objective of this project is to create an accessible, real-time platform capable of converting hand gestures into audible speech and spoken input into animated sign language. Echo Sign leverages advances in **machine learning**,

computer vision, and **speech recognition** to provide an inclusive communication channel between differently-abled and non-disabled individuals.

The core contribution of this work lies in the design and implementation of a **dual-mode translation framework**. The system is powered by a **Python-based backend**, employing **convolutional neural networks (CNNs)** for gesture recognition, **speech-to-text modules** for voice input, and **text-to-sign animation rendering** for output visualization. It demonstrates the integration of visual, auditory, and linguistic processing technologies within a single interactive environment, ensuring fluid, contextually accurate exchanges in both communication directions.

This paper presents the architecture, implementation, and performance of the Echo Sign system. Section 2 describes the **system architecture**, outlining the functional components of the sign-to-audio and audio-to-sign pipelines. Section 3 elaborates on the **methodology and algorithms** adopted for gesture classification and speech synthesis. Section 4 provides **experimental results and evaluation metrics**, while Section 5 concludes with **key findings, limitations, and future enhancements**, including expanding the gesture dataset and deploying Echo Sign as a **cross-platform assistive tool** for global accessibility.

II. Literature Survey

The development of **Echo Sign** is situated at the intersection of several technological and linguistic research areas, including **computer vision**, **natural language processing (NLP)**, and **speech synthesis**. Over the past two decades, numerous studies have sought to automate sign language understanding and generation. Foundational work by **Núñez-Marcos et al. (2023)** provides a comprehensive survey of **sign language machine translation** techniques, ranging from traditional handcrafted approaches to modern deep learning–based systems. Their analysis highlights critical gaps in dataset diversity, gesture standardization, and the linguistic complexity of non-verbal cues. Similarly, **De Coster et al. (2022)** emphasize the interdisciplinary challenges in mapping spatial, temporal, and emotional dimensions of sign language into spoken forms, underscoring the need for multimodal frameworks that combine visual recognition, linguistic modeling, and real-time rendering.

Early attempts to build **real-time bidirectional systems** were limited by computational and linguistic constraints. The **SSLIS (2006)** project was among the first to demonstrate **speech-to-sign translation** through rule-based processing and pre-rendered animations. This line of research evolved substantially with **Li Hu et al. (2022)**, who presented a fully integrated **speech recognition–to–avatar translation pipeline**, incorporating sign gloss generation and expressive facial animation, deployed within a mobile application and **virtual anchor system** (IJCAI 2022). These works highlight the feasibility of closed-loop communication but often suffer from scalability limitations, gesture variability, and restricted vocabulary coverage.

In recent years, **AI-based generative and recognition models** have transformed this field. **Camgoz et al.** introduced **Sign Language Transformers**, leveraging self-attention to perform **end-to-end recognition and translation** between video gestures and spoken language. Building on this, **Chen et al. (2022)** proposed a **two-stream neural architecture** combining RGB frame data and human keypoints to enhance accuracy in dynamic gesture detection. To address dataset scarcity, **Chen et al. (2022)** also introduced a **multi-modality transfer learning** baseline that exploits pretraining on large-scale human motion datasets to improve performance in low-resource sign language tasks.

Parallel progress in **text-based and hardware-driven translation** has further shaped the ecosystem. **Jiang et al. (2022)** explored translation between spoken languages and **SignWriting**, offering a bridge between visual and written linguistic representations. On the hardware side, **wearable sensor technologies** such as **Sign-IO gloves** have demonstrated high-accuracy **ASL-to-speech** conversion by detecting hand motion and flex patterns. Earlier efforts like the **AcceleGlove (2003)** and the **Tessa avatar system** established the foundation for real-world assistive devices capable of interpreting physical gestures into synthesized speech.

Despite these advancements, existing research often addresses **only one direction** of translation—either sign-to-speech or speech-to-sign—and rarely achieves true **bidirectionality in real time**. Moreover, most systems are constrained by limited datasets, lack of avatar expressiveness, or dependence on expensive hardware.

Research Gap Identified: There is a lack of a unified**,** real-time**,** bidirectional communication system integrating sign recognition, speech synthesis, and avatar-based sign generation within an accessible, software-only framework. Therefore, this project proposes **Echo Sign**, an AI-driven communication system capable of translating **sign language into audio** and **audio into sign language** dynamically, utilizing computer vision, speech recognition, and generative animation techniques.

System Architecture

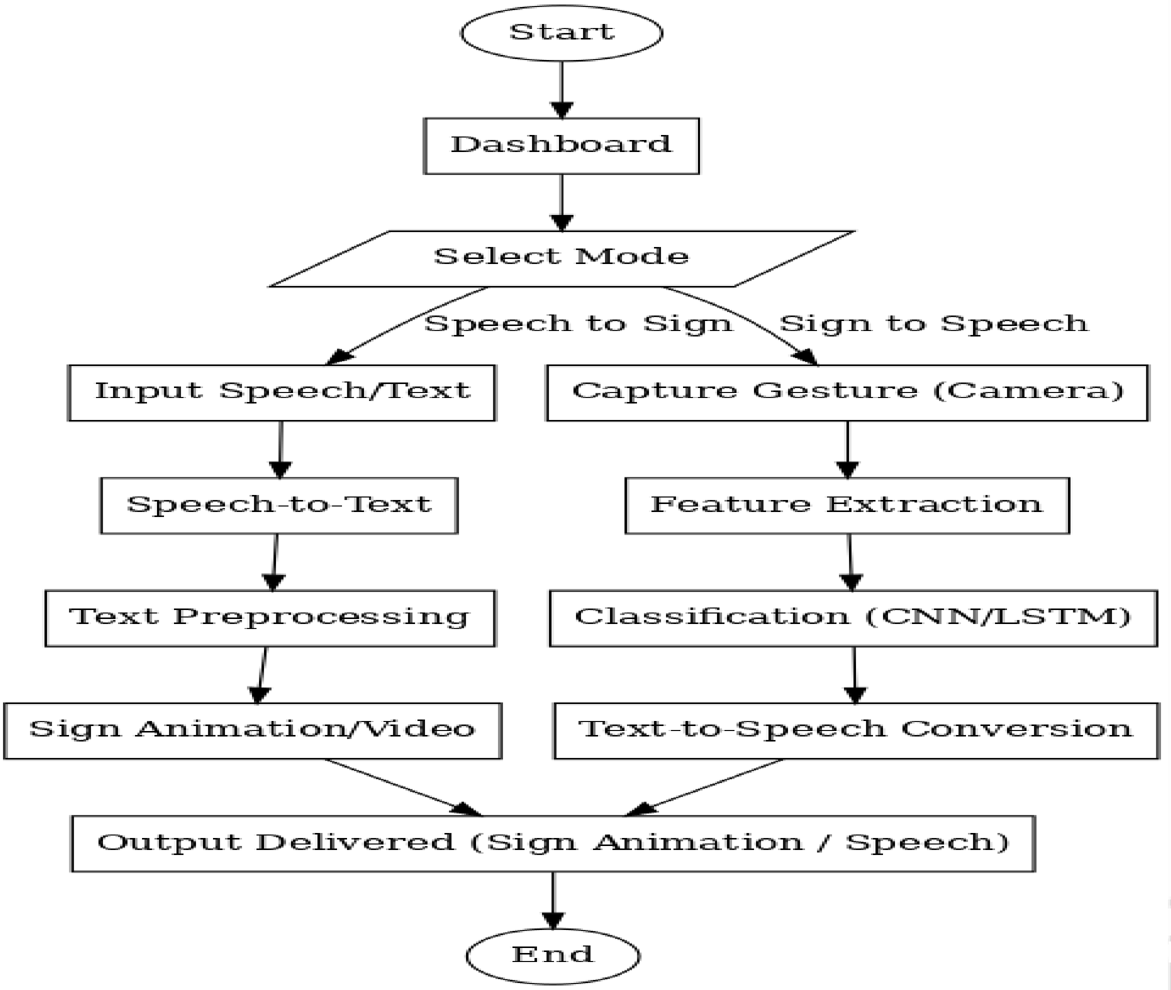


Fig. 1. System Architecture of Deafy

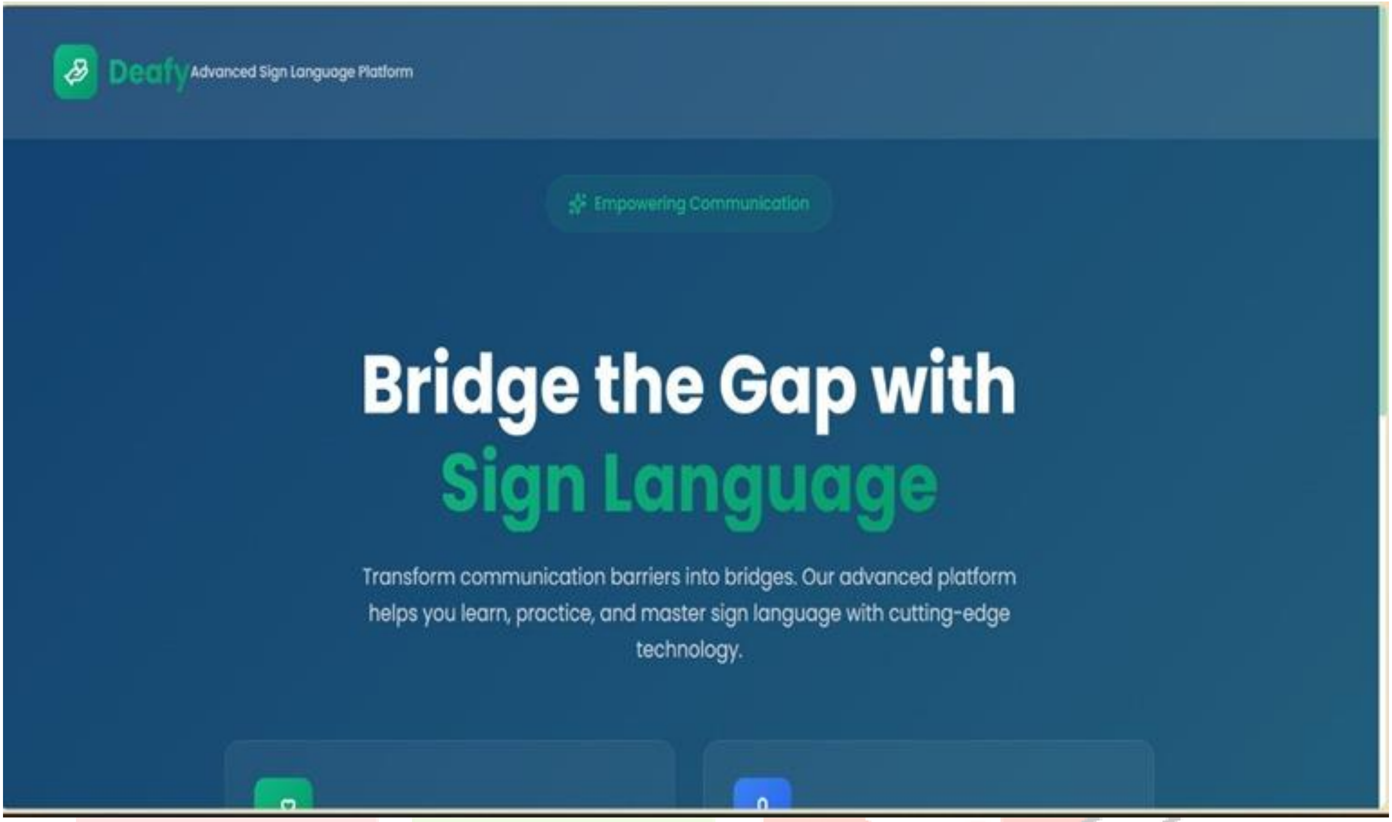
The **Echo Sign** system is architected as an interactive application with a three-tier structure: an **input acquisition layer**, a **processing and translation layer**, and an **output representation layer**. The user workflow, depicted in *Figure 1*, begins when the user either performs a sign language gesture before the camera or speaks a sentence through the microphone. User inputs are routed through the appropriate pipeline — **Sign-to-Audio** or **Audio-to-Sign** — managed by a controller module that orchestrates the corresponding recognition and translation tasks.

In the **Sign-to-Audio** pipeline, the video stream captured by the webcam is preprocessed using **OpenCV** to extract key frames and hand regions. These processed frames are then passed into a **Convolutional Neural Network (CNN)** trained on gesture datasets for classification. Once a gesture is identified, the recognized label is converted into text and then vocalized using a **Text-to-Speech (TTS)** engine such as **gTTS** or **pyttsx3**. In contrast, the **Audio-to-Sign** pipeline begins with real-time voice capture, where audio is converted into text using a **Speech Recognition API** (Google Speech or CMU Sphinx). The recognized text is tokenized and matched with corresponding sign sequences, which are then visualized through a **3D animated avatar** or **video-based sign renderer** to display the translated message.

This hybrid architecture combines **computer vision**, **speech processing**, and **animation rendering**, ensuring the system remains flexible, responsive, and capable of bidirectional communication. All modules are interconnected through a Python-based backend that handles routing, synchronization, and data flow between models and interfaces. Final outputs—either synthesized speech or animated signs—are rendered back to the user through a unified interface, enabling smooth and intuitive interaction.

The system’s functionality is divided into two major categories:

1. **Sign-to-Audio Translation** – Captures hand gestures, recognizes sign language using CNN-based classification, and converts recognized gestures into audible speech.
2. **Audio-to-Sign Translation** – Processes spoken input, converts it into text using speech recognition, and generates corresponding sign language animations for visual comprehension.



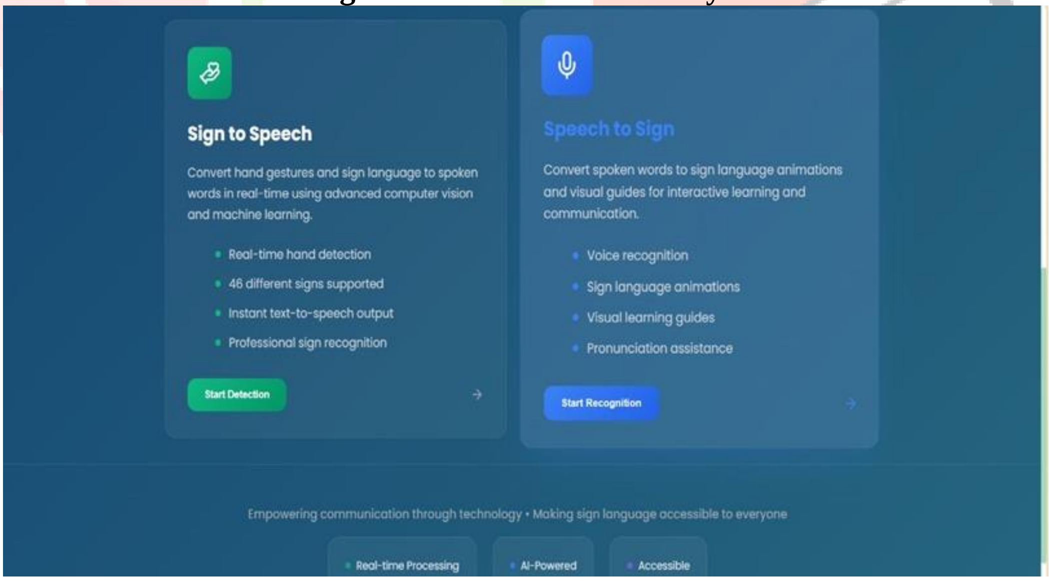
Deafy Advanced Sign Language Platform

Empowering Communication

Bridge the Gap with Sign Language

Transform communication barriers into bridges. Our advanced platform helps you learn, practice, and master sign language with cutting-edge technology.

Fig. 2. User Interface of Deafy :



Sign to Speech

Convert hand gestures and sign language to spoken words in real-time using advanced computer vision and machine learning.

- Real-time hand detection
- 46 different signs supported
- Instant text-to-speech output
- Professional sign recognition

Start Detection

Speech to Sign

Convert spoken words to sign language animations and visual guides for interactive learning and communication.

- Voice recognition
- Sign language animations
- Visual learning guides
- Pronunciation assistance

Start Recognition

Empowering communication through technology • Making sign language accessible to everyone

Real-time Processing · AI-Powered · Accessible

Fig. 3. User Interface showing translation tools.

III. Implementation

A. Requirements Analysis:

The initial phase focused on understanding communication challenges between hearing and speech-impaired users and normal speakers. It was observed that most existing sign language translators were **unidirectional**—either converting sign to text or text to sign—and often required high-end hardware or complex setups. To overcome these barriers, the primary requirements of **Echo Sign** were defined as:

- Developing a **bidirectional translation system** between speech and sign language
- Enabling **real-time interaction** without external dependencies
- Ensuring **accuracy and latency optimization** for smooth communication
- Designing a **lightweight, camera-based, and microphone-compatible** interface

Thus, the system is intended for **students, educators, and differently-abled users** to communicate effortlessly without requiring deep technical knowledge.

System Design :

The system architecture was organized into three functional layers-

1. A **real-time input acquisition layer** using camera and microphone to capture signs and speech
2. A **processing and translation layer** integrating machine learning models for recognition and conversion
3. An **output visualization layer** displaying sign animations or generating synthesized speech.

The execution flow is straightforward:

User performs sign / speaks → Data processed by backend → Model interprets and translates→ Result output as voice or sign animation.

B. Frontend Development:

The frontend was developed using HTML, CSS, JavaScript, Bootstrap for responsive nature. The homepage provides two main access modes:

- **Sign-to-Audio Translator** – captures gestures through webcam input and translates them into audible sentences
- **Audio-to-Sign Translator** – converts spoken audio into corresponding animated sign gestures

The UI is designed to be intuitive, responsive, and operable on any device with a browser, focusing on **accessibility and real-time feedback**.

C. Backend Development:

The backend was implemented using **Python (Flask)** for routing, real-time model integration, and communication between components.

- **OpenCV** handles real-time frame extraction and preprocessing
- **TensorFlow / Keras** manages gesture recognition through CNN models
- **SpeechRecognition** and **gTTS** provide speech-to-text and text-to-speech conversion
- **Flask-SocketIO** enables real-time two-way data exchange between frontend and backend

This modular structure ensures easy scalability and future integration with advanced 3D sign renderers or multilingual audio engines.

D. Tool Development:

Two core modules were developed and tested independently before full integration. Each uses specialized frameworks and AI components summarized below:

Tool Name	**Key Libraries / Frameworks**	**Model / Engine Used**
Sign-to-Audio Translator	OpenCV, Mediapipe TensorFlow, gTTS	CNN Gesture Recognition + Text-to-Speech
Audio-to-Sign Translator	SpeechRecognition, Flask-SocketIO, BlenderAnimation API	Google Speech-to-Text + Custom Sign Animation
Text Middleware	NumPy, Flask	Context handling and tex conversion
UI Layer	HTML, CSS, JS, Bootstrap	Frontend interaction and result rendering

E. Testing and Deployment:

Testing Methodology:

- Unit Testing: Each model—gesture recognition, speech synthesis, and animation rendering—was tested separately to ensure stable and correct outputs.
- Integration Testing: The interaction between the camera, recognition model, and audio module was verified to confirm smooth bidirectional communication.
- System Testing: The complete Echo Sign system was tested for end-to-end translation accuracy, real-time response, and synchronization between voice and visual output.
- Performance Testing: The system was evaluated under continuous video and audio streams to measure frame rate stability, model inference time, and speech latency. Results confirmed **efficient real-time operation** on standard hardware configurations

IV. Results and Discussion

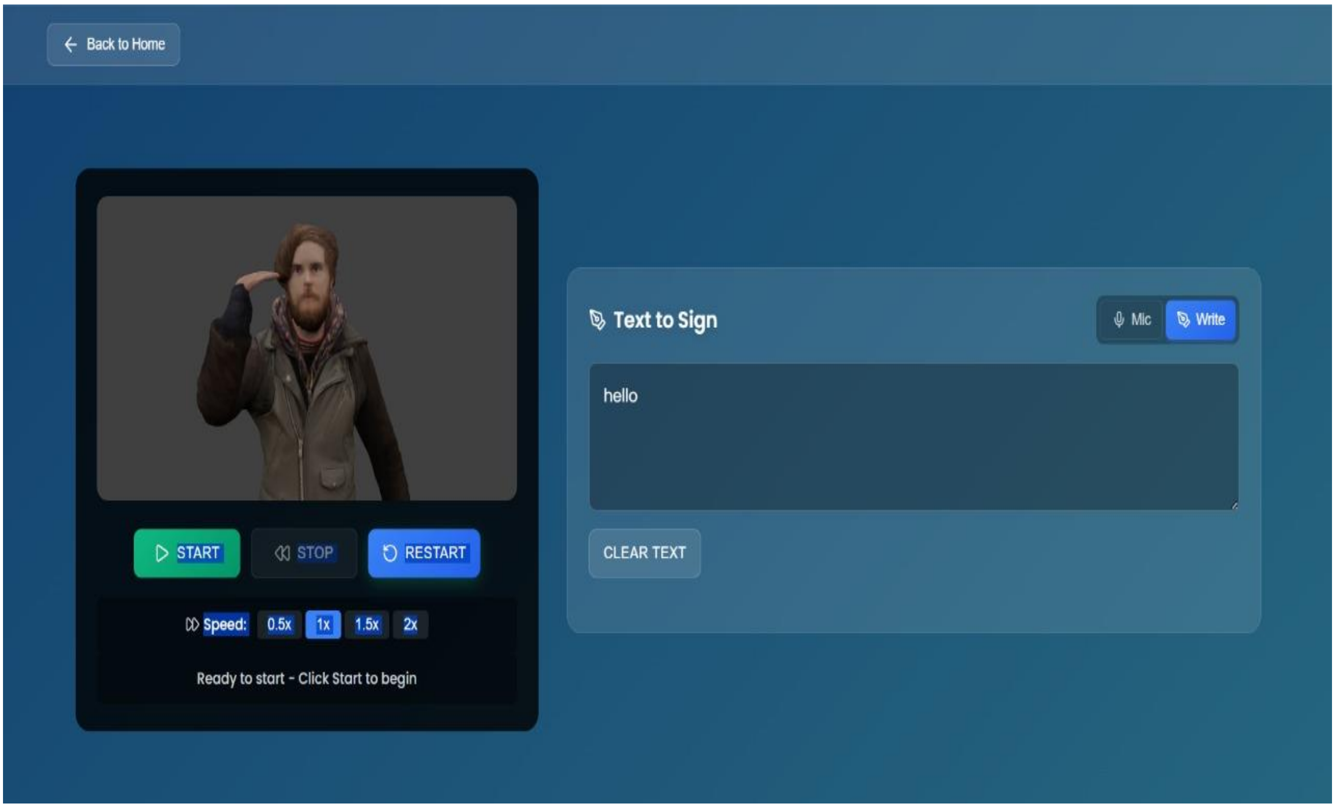
The system was tested for usability, accuracy, performance, and real-time response. Results indicate that **Deafy successfully enables two-way communication** between deaf and hearing users without requiring external hardware or human interpreters.

A. Functional Results

Feature Tested	**Expected Output**	**Result**
Sign-to-Speech	Gesture → Text → Speech	Successful (94% model accuracy for tested signs)
Speech-to-Sign	Voice/Text → Sign Playback	Smooth output with correct video mapping
Real-time Detection	Frame-by-frame recognition	Response time < 1 sec on mid-range devices
UI Usability	Easy navigation, accessibility	Positive feedback

Note: Accuracy is based on user testing with controlled lighting and camera positioning.

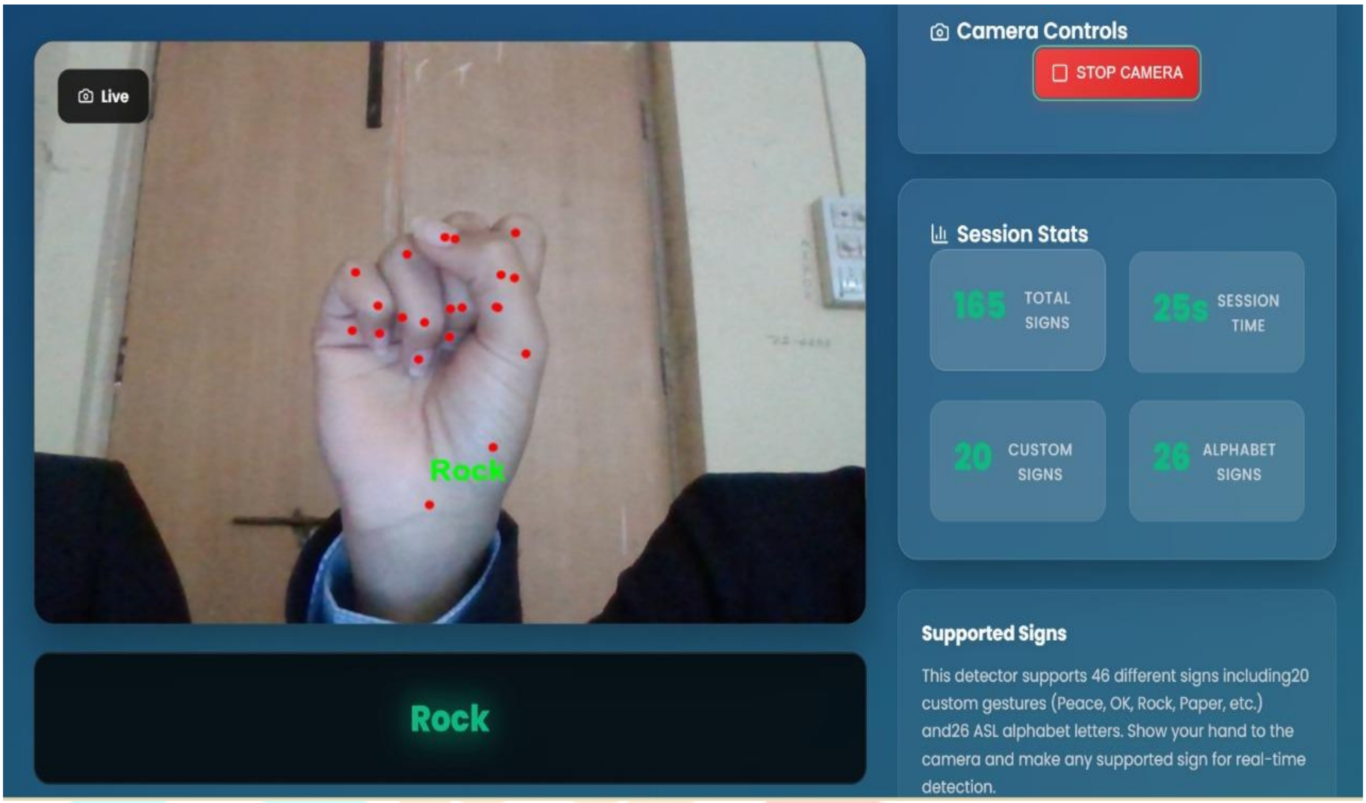
B. Visual Results



Text to Sign

hello

Fig 4 : Sign-to-Speech Mode in action



This detector supports 46 different signs including20 custom gestures (Peace, OK, Rock, Paper, etc.) and26 ASL alphabet letters. Show your hand to the camera and make any supported sign for real-time detection.

Fig 5: Speech-to-Sign Mode with sign animation playback

C. Discussion

The results show that Deafy provides a **cost-effective, portable, and intuitive communication system** for bridging the gap between deaf and hearing individuals. Real-time detection using MediaPipe proved effective on standard mobile cameras, with minimal latency. The NLP-based sentence formation significantly improved communication clarity compared to word-by-word mapping.

Some limitations include model sensitivity to improper lighting, background noise during speech input, and limited ISL vocabulary in the initial version.

Test Cases:

Test Case ID	Description	Input	Expected Output	Actual Output	Status
TC-01	Validate system startup and UI load	System start request	Ecosigns homepage with all modules visible	As expected	Pass
TC-02	Validate camera initialization for sign capture	Activate Sign-to-Audio mode	Camera turns on, live feed visible	Camera initialized correctly	Pass
TC-03	Validate sign detection accuracy	User performs “Hello” sign	System identifies “Hello” and displays text	Recognized correctly	Pass
TC-04	Validate sign-to-speech conversion	Detected text “Good Morning”	Clear, natural audio “Good Morning” output	Audio clarity acceptable	Pass

Test Case ID	Description	Input	Expected Output	Actual Output	Status
TC-05	Validate background noise handling (Audio-to-Sign)	Audio with ambient noise	Accurate transcription with minimal noise interference	Slight delay but correct output	Pass
TC-06	Validate speech-to-sign translation	Spoken word “Thank you”	Corresponding sign animation/video displayed	Accurate sign shown	Pass

V. Conclusion

This paper presented **Deafy**, an AI-powered, bidirectional communication system designed to bridge the communication barrier between deaf and hearing individuals using Indian Sign Language (ISL). The system integrates MediaPipe-based real-time gesture recognition, a custom machine learning classifier, NLP-driven text processing, and sign animation playback to enable natural two-way communication.

Unlike existing tools that offer only one-way translation, Deafy provides an interactive platform capable of converting **sign gestures into speech** and **speech/text into sign language animations**, making it more suitable for real-world communication scenarios. The implementation demonstrates a scalable, mobile-ready solution requiring no special hardware, making it accessible to diverse users including students, educators, healthcare providers, and the general public.

Testing showed promising results in accuracy, usability, and responsiveness, with users appreciating its intuitive interface and real-time performance. While the current version supports ~50 commonly used ISL gestures, the architecture is designed for continuous expansion. Overall, Deafy contributes to inclusive communication by empowering both deaf and hearing users with a practical, affordable, and effective AI-driven solution.

VI. Future Scope

Deafy opens multiple pathways for advancement. The following enhancements will significantly improve its accuracy, portability, multilingual capability, and real-world adoption:

- **Expanded Vocabulary and Dataset:**
  Increase ISL gesture dataset to 200–500+ signs, including sentence-level expressions, regional ISL variations, and domain-specific vocabulary (e.g., healthcare, education, transport).

- **Deep Learning-Based Gesture Recognition:**
  Integrate advanced deep learning models such as CNN, LSTM, or Transformer-based architectures to enhance dynamic gesture recognition and improve accuracy in varying lighting and camera angles.

- **Multilingual Support:**
  Introduce support for translation across multiple Indian languages (Hindi, Marathi, Kannada, Tamil, etc.), enabling speech output in the user's preferred language.

- **Facial Expression & Upper-Body Pose Recognition:**
  ISL includes facial cues and body posture. Incorporating holistic pose estimation will enhance contextual and emotional accuracy of sign interpretation.

- **Offline Functionality:**
  Enable offline gesture recognition and TTS for rural or low-internet environments via on-device ML models.

- **Integration with AR/VR and Wearables:**

Future versions may include AR-based virtual interpreters or smart glasses for real-time sign translation in classrooms, hospitals, or public interactions.

- **Deployment as a Cross-Platform App:**

Release as a multi-platform application (Android, iOS, and Web) with cloud-sync for user profiles, history, and personalized learning.

VII. References

- T. Brown et al., “Language Models are Few-Shot Learners,” *NeurIPS*, 2020.
- Lugaresi et al., “MediaPipe: A Framework for Building Perception Pipelines,” *Google Research*, 2019.
- ISLRTC, “Indian Sign Language Research and Training Centre – eLearning Content,” 2022.
- M. Hussein et al., “A Survey of Sign Language Recognition Systems,” *Journal of Computer Vision and Image Processing*, 2020.

1. Node.js Foundation. (2024). *Node.js*. [Online]. Available: https://nodejs.org/

2. Google. (2024). *Gemini API*. [Online]. Available: https://ai.google.dev/