Journal of Science, Computing and Engineering Research (JSCER)
Volume-8, Issue-4, April 2025.
DOI: *https://doi.org/10.46379/jscer.2025.080403*

ENGLISH AUDIO TO SIGN LANGUAGE CONVERTER

[1]Neeraj Jain, [2] Ravi Kumar, [3]Rohan Jonwal, [4]Vishal,[5] Punit Choudhary,[6] Neeraj Saini

[1]Associate Professor, Department of CSE, Modern Institute of Technology and Research Centre, Alwar, Rajasthan, India

[2,3,4,5,6] UG Student, Department of AI&DS, Modern Institute of Technology and Research Centre, Alwar, Rajasthan, India

Article Information

Received : 22 April 2025

Revised : 24 April 2025

Accepted : 25 April 2025

Published : 28 April 2025

Corresponding Author:

Neeraj Saini

Abstract— This Communication is an essential element of human life, but for individuals with hearing impairments, everyday interactions can present significant challenges due to the absence of accessible, real-time translation tools. Although technologies such as speech-to-text converters and closed captioning systems have improved accessibility to some extent, they often fall short when it comes to catering to users who primarily rely on sign language for communication. This paper presents the design and development of the English-Audio-Speech-to-Sign-Language-Converter, an intelligent system that enables real-time translation of spoken or typed input into animated Indian Sign Language (ISL) gestures. The system aims to bridge the communication divide by functioning as a digital interpreter capable of translating both audio and text inputs into visual sign animations that are clear, accurate, and contextually meaningful. The project incorporates a multi-layered methodology involving the integration of Google Speech-to-Text API for accurate speech recognition, Natural Language Processing (NLP) techniques for textual analysis, and a locally stored animation database for rendering sign gestures. It features a user-friendly interface that supports both speech and text input, providing greater accessibility and usability across different scenarios. The modular architecture allows the system to process input in real time, analyze sentence structure, extract relevant keywords, and map them to corresponding animations, which are then rendered seamlessly through the frontend. This system not only demonstrates the practical application of AI and NLP in the field of accessibility technology but also lays the groundwork for future enhancements such as multilingual support, mobile platform integration, and dynamic sign gesture generation. Through this work, the project aims to promote digital inclusivity and empower hearing-impaired individuals by offering a tool that is both innovative and impactful. The proposed system, though still in development for broader application, presents a promising step forward in making technology more inclusive and accessible for all segments of society.

Citation: Neeraj Jain, Ravi Kumar, Rohan Jonwal, Vishal, Punit Choudhary, Neeraj Saini, "ENGLISH AUDIO TO SIGN LANGUAGE CONVERTER", Journal of Science, Computing and Engineering Research, 8(04), April 2025.

I. INTRODUCTION

Comparator is a circuit that output is binary information depending upon the comparison of two input voltages here the comparison in

Communication is a fundamental aspect of human interaction, essential for expressing ideas, emotions, and information. However, for individuals with hearing impairments, communication often presents significant challenges, especially in environments where sign language interpreters are not readily available. This limitation creates barriers in education, healthcare, employment, and day-to-day interactions, hindering social inclusion and equal participation. With advancements in artificial intelligence (AI) and natural language processing (NLP), there is a growing opportunity to develop intelligent systems that can mitigate such barriers and promote digital accessibility.

The English-Audio-Speech-to-Sign-Language-Converter is a technologically innovative solution aimed at bridging the communication gap between hearing and non-hearing individuals. The system captures speech or text input and translates it into corresponding sign language animations, enabling users with hearing impairments to understand spoken content visually. Unlike traditional speech-to-text systems, which only provide written transcriptions, this converter emphasizes visual understanding by leveraging animated sign gestures, thereby catering to users who rely on sign language as their primary mode of communication.

This project integrates several cutting-edge technologies to ensure a real-time and accurate translation system. By using Google Speech-to-Text API, the system transcribes speech into text with high precision. The transcribed text is then processed through a language understanding module that interprets the context, extracts key phrases, and maps them to a sign animation database. These animations, stored locally in the system, are rendered and displayed to the user via an intuitive user interface.

The system's dual input functionality—supporting both voice and text—broadens its usability, making it accessible in diverse environments such as classrooms, hospitals, government offices, and customer service platforms. In essence, this project is not merely a technological implementation but a step toward societal inclusivity. It reflects the potential of AI and NLP to create impactful assistive technologies that empower individuals and foster equitable access to communication.

II. PROBLEM STATEMENT

Despite the widespread use of sign language among individuals with hearing and speech impairments, communication with the broader community remains a challenge due to the limited number of people proficient in sign language. This communication gap often leads to social isolation, misunderstanding, and lack of access to essential services for differently-abled individuals.

While various technologies exist for speech recognition and translation, few systems focus on converting spoken language into Indian Sign Language (ISL), which is essential for inclusive communication within the Indian context. Moreover, many existing solutions require expensive hardware or are not user-friendly.

There is a pressing need for an accessible, real-time, and cost-effective tool that can convert spoken audio into ISL gestures to assist communication between differently-abled individuals and non-sign language users.

III. PROPOSED METHOD

The methodology adopted in this project is designed to ensure a smooth and effective translation of spoken or written language into sign language animations. The system architecture is modular and consists of five primary components: speech input processing, text input handling, natural language processing, sign gesture mapping, and animation rendering. The goal is to ensure high accuracy and responsiveness at each stage of the workflow while maintaining a user-friendly interface.

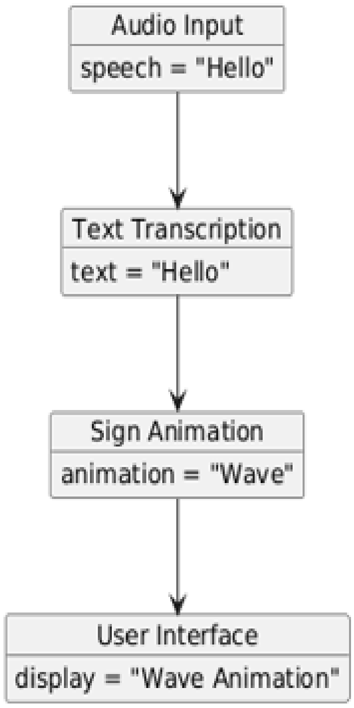
The first component is the Speech Recognition Module, which uses the Google Speech-to-Text API to transcribe spoken input into plain text. This API is chosen for its high accuracy, support for multiple languages, and ease of integration. For users who prefer to type, the system also allows direct text input, which bypasses the speech recognition stage and moves straight into processing. This dual-input approach ensures flexibility and usability across different contexts and user preferences.

Once the system has textual input, it is passed to the Text Processing and NLP Module. This module is responsible for analysing the input text, identifying key phrases, eliminating stop words, and understanding grammatical structure. Using basic natural language processing techniques, the module breaks down the sentence into interpretable segments. These segments are then matched against a predefined sign language database using keyword mapping logic.

The Sign Gesture Mapping Module plays a crucial role in ensuring accurate translation. It contains a library of pre-recorded or animated gestures stored locally. Each keyword or phrase in the processed text is mapped to its corresponding gesture in Indian Sign Language (ISL). If a word or phrase lacks a direct sign equivalent, the system either decomposes it into sign able parts or flags it for future inclusion in the database.

Finally, the Sign Language Rendering Module displays the mapped animations in a sequence, maintaining the context and sentence flow. This visual output is presented through a web-based user interface built with standard front-end technologies. The animations are optimized for clarity and comprehension, allowing users to easily interpret the intended message. The entire process is streamlined to function in real-time, ensuring minimal delay between input and output, which is crucial for dynamic conversations.

IV. OBJECT DIAGRAM:



V. IMPLEMENTATION

The system is implemented as a web-based application combining both frontend and backend technologies to deliver real-time audio-to-sign language conversion. The frontend is developed using HTML, CSS, and JavaScript, with the Web Speech API integrated to handle live audio input and convert spoken words into text within the browser. This eliminates the need for any external software installations and ensures the system runs seamlessly across different platforms.

The converted text is then sent to the backend, which is built using Python and Django. In the backend, the Natural Language Toolkit (NLTK) is used for text processing tasks such as tokenization, stop-word removal, and keyword extraction. This step ensures that only meaningful and relevant words are passed on for gesture mapping.

Each keyword is matched against a predefined set of sign language gestures, which are stored as animation files created using Blender 3D. These animations are organized in a structured folder system and accessed dynamically based on the identified keywords. Once the appropriate animations are selected, they are rendered on the frontend, where users can view the sign language interpretation of the spoken input.

The system is designed with simplicity, accessibility, and real-time performance in mind. It supports common vocabulary and is optimized for quick execution, making it suitable for use in classrooms, public service areas, and personal communication. The modular architecture also allows for future expansion, such as support for additional languages, voice profiles, or more complex sentence handling.

VI. TECHNOLOGY STACK:

1. Programming Language:

- Python: Used for backend development, text processing, and integration of speech recognition and NLP functionalities due to its vast library support and simplicity.

2. Backend Framework:

- Django (Python Framework): Employed to build the web-based backend system for routing, data management, and logic implementation. Django's built-in security features and scalability made it ideal for the project.

3. Speech Recognition:

- Google Speech-to-Text API: Integrated to convert real-time voice input into text using advanced cloud-based AI models. It supports multiple languages and offers high accuracy with low latency.

4. Natural Language Processing:

- NLTK (Natural Language Toolkit): Used to perform sentence tokenization, keyword extraction, stop-word removal, and grammatical analysis to process the textual input efficiently.

5. Database Management:

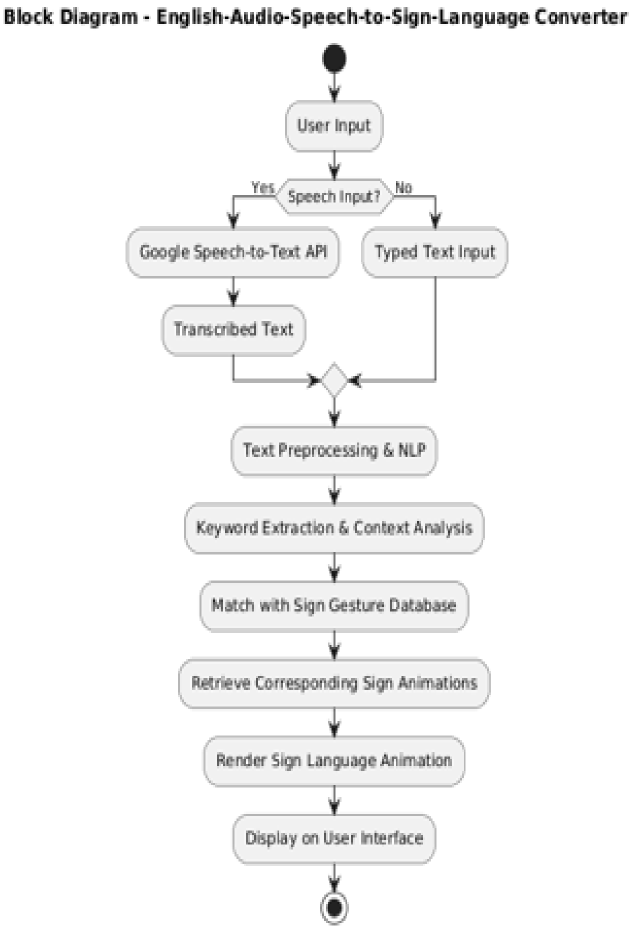
- SQLite: A lightweight relational database used to store mappings of text phrases to sign language animations and user input logs. It is embedded and easily portable, which suits the project’s requirements.

6. Animation Handling:

- Local Animation Library (MP4/GIF files): Pre-recorded or rendered Indian Sign Language gestures are stored and mapped to processed keywords, which are then displayed sequentially.

7. Frontend Technologies:

- HTML5, CSS3, and JavaScript: These technologies were used to develop a responsive and accessible user interface. JavaScript was particularly useful in handling dynamic animation rendering and improving user interactivity.

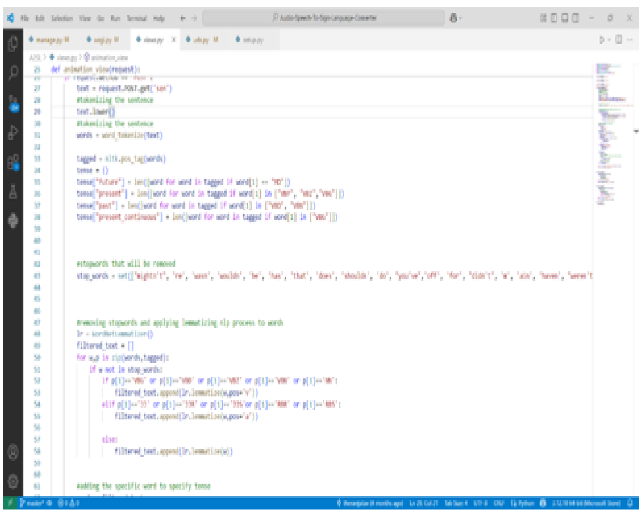


Sample Code:

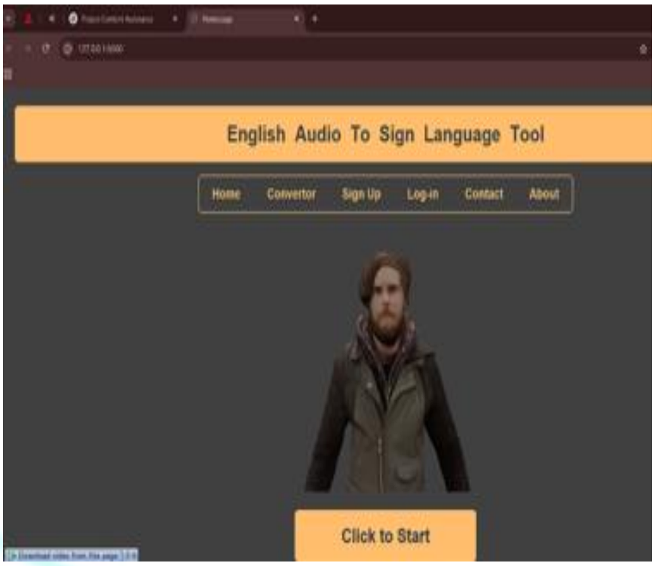
```
words = filtered_text
    temp=[]
    for w in words:
      if w=='I':
        temp.append('Me')
      else:
        temp.append(w)
    words = temp
    probable_tense = max(tense,key=tense.get)

    if probable_tense == "past" and tense["past"]>=1:
      temp = ["Before"]
      temp = temp + words
      words = temp
    elif probable_tense == "future" and tense["future"]>=1:
      if "Will" not in words:
          temp = ["Will"]
          temp = temp + words
          words = temp
      else:
        pass
    elif probable_tense == "present":
      if tense["present_continuous"]>=1:
        temp = ["Now"]
        temp = temp + words
        words = temp
```

VII. CODE SCREENSHOT:



VIII. RESULTS



IX. . CONCLUSION

The English-Audio-Speech-to-Sign-Language-Converter represents a significant step forward in leveraging artificial intelligence and natural language processing to address real-world accessibility challenges. By enabling real-time conversion of speech and text into sign language animations, the system empowers individuals with hearing impairments to participate more fully in conversations and public interactions. Its dual-input design, use of reliable APIs, and clean user interface make it a practical and scalable solution that can be adapted for use in diverse settings.

While the current implementation focuses on Indian Sign Language and basic sentence structures, the project lays a solid foundation for future developments, including support for multiple languages, advanced grammar handling, and dynamic animation generation. In a world that increasingly values digital inclusivity, systems like this not only bridge communication gaps but also serve as a testament to the potential of technology to foster equality and accessibility for all.

References

[1]. 1. Hemang Monga, Jatin Bhutani, Muskan Ahuja, Nitika Maid, “Speech to Indian Sign Language Translator”, Recent Trends in Intensive Computing, 2021.

[2]. 2. Rishin Tiwari, Saloni Birthare, Mayank Lovanshi, “Audio to Sign Language Convertor”, International Journal for Research in Applied Science and Engineering Technology, 2022.

[3]. 3. Alisha Kulkarni, Archith Vinod Kariyal, Dhanush V, Paras Nath Singh, “Speech to Indian Sign Language Translator”, Atlantis Press International B.V., 2021.

[4]. 4. Maria Papatsimouli, Konstantinos-Filippos Kollias, Lazaros Lazaridis, George Maraslidis, Herakles Michailidis, Panagiotis Sarigiannidis and George F. Fragulis “Real Time Sign Language Translation Systems”, University of Western Macedonia, 2022.

[5]. 5. Lalit Goyal, Vishal Goyal. “Development of Indian Sign Language Dictionary using Synthetic Animations”, Indian Journal of Science and Technology, Vol 9(32) August 2016.

[6]. 6. Kumud Tripati, Neha Baranwal, and GC Nandi, “Continuous Indian Sign Language and Gesture Recognition”, ELSIVIER, IMCIP 2015.

[7]. 7. Khalid El-Darymli, Othman O. Khalifa, Hassan Enemosah. “Speech to Sign Language Interpreter System (SSLIS)”, the IEEE International Conference of Computer and Communication Engineering (ICCCE ’06), in proceedings of Kuala Lumpur, Malaysia, 2006.

[8]. 8. Divya Deora and Nikesh Bajaj. “INDIAN SIGN LANGUAGE RECOGNITION”, 21st International Conference of Emerging Technology Trends in Computer Science,2012.

[9]. 9. Sugandhi, Parteek Kumar, and Sanmeet Kaur. “Sign Language Generation System Based on Indian Sign Language Grammar” ACM Transactions on Asian and Low-Resource Language Information Processing, Vol 19, Issue 4, July 2020.

[10]. 10. Kulkarni, A., e, “Speech to Indian Sign Language Translator,” 3rd International Conference on Integrated Intelligent Computing Communication & Security (ICIIC 2021), 2021.

[11]. 11. Sharma, P., “Translating Speech to Indian Sign Language Using Natural Language Processing,” Future Internet 2022, 14(9), 253, 2022.