

Speech to Indian Sign Language Translator

Mr. Deepak Bhagat¹, Mr. Hitesh Bharambe², Mr. Jashpal Joshi³, Prof. Shahe Gul⁴

Students, Department of Computer Engineering^{1,2,3}

Guide, Department of Computer Engineering⁴

Theem College of Engineering, Boisar (E), Mumbai, Maharashtra, India

Abstract: *Communication plays a critical role for people and is regarded as a skill in life. Having this important aspect of life and surroundings in mind, we present our project article, which focuses primarily on supporting patients with pain or silent speech. Our research work leads to improved contact with the deaf and the mute. Each sign language uses sign patterns visually conveyed to express the true meaning. The combination of hand gestures and/or motions of arm and body is called Sign Language and the Dictionary. It is the combination of hands and facial expressions. Our program project is able to understand signals in sign language. These symbols may be used to interact with hearing aids. Our article suggests a program that allows common people to interact effectively with others that are hard to understand. In this case, we are implementing the Indian Sign Language (ISL) method by using a microphone and a camera. Translation of the voice into Indian sign language system by the ISL translation system is possible. The ISL translation framework uses a microphone to get pictures (from ordinary people) or continuous video clips, which the application interprets. Deaf people always miss out the fun that a normal person does, may it be communication, playing computer games, attending seminars or video conferences, etc. Communication is the most important difficulty they face with normal people and also every normal person does not know the sign language. The aim of our project is to develop a communication system for the deaf people. It converts the audio message into the sign language. This system takes audio as input, converts this audio recording message into text and displays the relevant Indian Sign Language images or GIFs which are predefined. By using this system, the communication between normal and deaf people gets easier.*

Keywords: Indian Sign Language Translator.

I. INTRODUCTION

1.1 Overview

It is said that Sign language is the mother language of deaf people. This includes the combination of hand movements, arms or body and facial expressions. There are 135 types of sign languages all over the world. Some of them are American Sign Language (ASL), Indian Sign Language (ISL), British Sign Language (BSL), Australian Sign Language (Auslan) and many more. We are using Indian Sign Language in this project. This system allows the deaf community to enjoy all sort of things that normal people do from daily interaction to accessing the information Sign language is communication language used by the deaf peoples using face, hands or eyes while using vocal tract. Sign language recognizer tool is used for recognizing sign language of deaf and dumb people. Gesture recognition is an important topic due to the fact that segmenting a foreground object from a cluttered background is a challenging problem. There is a difference when human looks at an image and a computer looking at an image. For Humans it is easier to find out what is in an image but not for a computer. It is because of this, computer vision problems remain a challenge. Sign language is a language that consists of signs made with hands and other movements, facial expressions and postures of body, which is primarily used by people who are deaf or hard hearing peoples that they can easily express their thoughts or can easily communicate with other people. Sign language is very important a far the deaf people are concerned for their emotional, social and linguistic growth. First language for the deaf people is sign language which get proceeded bilingually with the education of national sign language as well as national written or spoken language. There are different communities of deaf people all around the world therefore the sign language for these communities will be different. The different sign languages used by different communities are America uses American Sign Language, Britain sign language is used by Britain, similarly, India uses Indian sign language etc. For expressing thoughts and communicating with each other.

According to 2011 census of India, there are 63 million people which sums up to 6.3% of the total population, who are suffering from hearing problems. Out of these people, 76-89% of the Indian hearing challenged people have no knowledge of language either signed, spoken or written. The reason behind this low literacy rate is either the lack of sign language interpreters, unavailability of Indian Sign Language tool or lack of researches on Indian sign language.

Sign language is a natural way of communication for challenged people with speaking and hearing disabilities. There have been various mediums available to translate or to recognize sign language and convert them to text, but text to sign language conversion systems have been rarely developed, this is due to the scarcity of any sign language corpus. This is done by eliminating stopwords from the reordered sentence. Stemming is applied to convert the words to their root form as Indian sign language does not support for inflections of the word. All words of the sentence are then checked against the words in the dictionary containing videos representing each of the words. If the words are not found in the dictionary, its corresponding synonym is used to replace it. The proposed system is innovative as the existing systems are limited to direct conversion of words into Indian sign languages whereas our system is capable to do the translation.

1.2 Problem Statement:

The main purpose of project is to take user input and convert it to sign language. Using Natural Language Processing (NLP) are implemented to classify the text/speech into small part. Then searching words/latter from database. At the end display the appropriate sign or gestures to the user. In this, problem we have considered that are;

1. Speech recognition & converting into text.
2. The whole statement converting into sign language.
3. No words are found in database/dataset.

Sign language is a language that uses manual communication methods such as facial expressions, hand gestures and bodily movements to convey information. This project makes use of videos for specific words combined to translate the text language into sign language. Speech impaired people use hand signs and gestures to communicate. Normal people face difficulty in understanding their language. Hence there is a need of a system which recognizes the different signs, gestures, and conveys the information to the deaf people from normal people. It bridges the gap between physically challenged people and normal people. Our approach provides the result in minimum time span with maximum precision and accuracy in comparison to other existing approaches.

1.3 Motivation

Sign language is a natural way of communication for challenged people with speaking and hearing disabilities. There have been various mediums available to translate or to recognize sign language and convert them to text, but text to sign language conversion systems have been rarely developed, this is due to the scarcity of any sign language corpus. This will provide information access and services to deaf people in Indian sign language.

1.4 Aim and Objectives

1.4.1 Aim

This project intends to:

1. To creating a translation system that consists of a parsing module which parses the input English sentence to phrase structure grammar representation on which Indian sign language grammar rules are applied.
2. To convert these sentences into Indian sign language grammar in real domain.
3. To develop a communication system for the deaf people.

1.4.2 Objective

The main aspect of this project is to help deaf and dumb people to communicate easily in society with people who don't know sign language. This web application convert the text in to sigh language whereas it is open source and freely available which in turn will benefit the dead community. To increase opportunity for advancement and success in education, employment, personal relationship, and public access venues.

1.5 Scheme

Speech is taken as an input by a normal person using a microphone of computer. With the help of a trained voice database, voice to text conversion takes place i.e., voice is converted into text-by-text recognition module. Meanings and symbols are found by comparing the database and converted text and then the sign symbols are displayed with text to hard hearing person. A movement made using part of body, especially using hands, arms, face, head to express meaningful information or emotions is known as gesture. Gesture recognition is valuable in applications that involve human machine interaction. The sign language translation system converts speech to sign. Speech recognizer is used for decoding the spoken voice into word sequence.

1.6 Speech to Sign Fundamentals

Speech is taken as an input by a normal person using a microphone of computer. Then process the input through NLP and display the proper sign language for that input. There are different types of solutions and approaches. In this we are taking most ideal way or approach.

1.6.1 Speech Recognition

- Set Device ID to the selected microphone: In this step, we specify the device ID of the microphone that we wish to use in order to avoid ambiguity in case there are multiple microphones. This also helps debug, in the sense that, while running the program, we will know whether the specified microphone is being recognized. During the program, we specify a parameter device_id. The program will say that device_id could not be found if the microphone is not recognized.
- Allow Adjusting for Ambient Noise: Since the surrounding noise varies, we must allow the program a second or too to adjust the energy threshold of recording so it is adjusted according to the external noise level.
- Speech to text translation: This is done with the help of Google Speech Recognition. This requires an active internet connection to work. However, there are certain offline Recognition systems such as Pocket Sphinx, but have a very rigorous installation process that requires several dependencies. Google Speech Recognition is one of the easiest to use.³

1.6.2 Conversion of Speech to Text

NLTK is a leading platform for building Python programs to work with human language data. It provides easy-to-use interfaces to over 50 corpora and lexical resources such as WordNet, along with a suite of text processing libraries for classification, tokenization, stemming, tagging, parsing, and semantic reasoning, wrappers for industrial-strength NLP libraries, and an active discussion forum. Thanks to a hands-on guide introducing programming fundamentals alongside topics in computational linguistics, plus comprehensive API documentation, NLTK is suitable for linguists, engineers, students, educators, researchers, and industry users alike. NLTK is available for Windows, Mac OS X, and Linux. Best of all, NLTK is a free, open source, community-driven project.

1.6.3 Finding ISL in Datasets

By using Punkt sentence tokenizer we can splitting the words into latter. This tokenizer divides a text into a list of sentences by using an unsupervised algorithm to build a model for abbreviation words, collocations, and words that start sentences. It must be trained on a large collection of plaintext in the target language before it can be used. However, Punkt is designed to learn parameters (a list of abbreviations, etc.) unsupervised from a corpus similar to the target domain. The pre-packaged models may therefore be unsuitable: use `PunktSentenceTokenizer(text)` to learn parameters from the given text. After all according to latter sign language is displayed which are predefined in dataset.

II. LITERATURE REVIEW

2.1 General Review

As per Amit Kumar Shinde on his study of sign language to text and vice versa in Marathi Sign language recognition is one of the most important research and it is the most natural and common way of communication for the people with hearing problems. A hand gesture recognition system can help deaf persons to communicate with normal people in the absence of

an interpreter. The system works both in offline mode and through web camera. Neha Poddar, Shrushti Rao, Shruti Sawant, Vrushali Somavanshi, Prof. Sumita Chandak in their paper discussed about the prevalence of deafness in India is fairly significant as it is the second most common cause of disability. A portable interpreting device which convert higher mathematics sign language into corresponding text and voice can be very useful for the deaf people and solve many difficulties.

The glove based deaf-mute communication interpreter introduced by Anbarasi Rajamohan, Hemavathy R., Dhanalakshmi is a great research. The glove comprises of five flex sensors, tactile sensors and accelerometer. The controller matches the gesture with pre-stored outputs. The evaluation of interpreter was carried out for ten letters _A,,_B,,_C,,_D,,_F,,_I,,_L,,_O,,_M,,_N,,_T,,_S,,_W,,.

As per the Neha V. Tavari A. V. Deorankar Dr. P. N. Chatur in his report discuss that many physically impaired people rely on sign language translators to express their thoughts and to be in touch with rest of the world. The project introduces the image of the hand which is captured using a web camera. The image acquired is processed and features are extracted. Features are used as input to a classification algorithm for recognition. The recognized gesture is used to generate speech or text. In this system, flex sensor gives unstable analog output and also it requires many circuits and is thus very expensive.

Purushottam Kar et al. [20] in their 2007 paper developed INGIT, a system for translating Hindi strings to Indian Sign Language. It was developed specifically for Railway Inquiry domain. FCG was used to implement the grammar for Hindi. The developed module converts the user input into a thin semantic structure. Unnecessary words are removed by feeding this input to ellipsis resolution. The ISL generator module then generated a suitable ISL-tag structure depending on the type of sentence. A graphical simulation was then generated by a HamNoSys converter. The system was successful for about 60% cases in generating the semantic structures.

Ali et al. [21] developed a domain-specific system in which the input fed had to be English text. The text was converted into ISL text which was further translated into ISL symbols. The architecture of the system had the following components: 1) A text translation input module. 2) Tokenizer to break down the sentence into separate words. 3) A ISL symbols repository which was specific to railway inquiries. If a word had no corresponding sign assigned to it in the repository, then it's synonyms sign was used. 4) All the words were mapped with their corresponding symbols by a purposefully built translator. It also filtered the words to be translated by eliminating the words which were offensive or abusive or did not have any sign stored. 5) An accumulator which accumulated the words in the sequence entered.

Vij et al. [22] developed a 2-phase system of Sign Language Generation. The first phase dealt with preprocessing Hindi 2 Sentences and converting it into ISL grammar. The phase used a combination of Dependency Parser and WordNet for this purpose. Dependency graphs in the Dependency Parser represented words and their relationships between head words and words which modify those heads. In the second phase, HamNoSys was used for converting this grammar into different corresponding Sign Language symbols. The generated symbols are converted into XML tags form using SIGML. The XML tags form is then readable by a 3D rendering software.

MS Anand et al. [3] developed a two-way ISL translation system. In the speech- to-sign module, the input speech was first put through the noise removal submodule. The output was then used as an input for the speech recognizer for decoding the spoken speech into a textual word sequence. A natural language converted the word sequence into a sequence of signs by a rule-based technique. Finally, a sign animation module with text annotation was used for displaying the signs.

In the system generated by Dasgupta et al. [17]. English text was taken as input and it was then converted into the corresponding ISL structure that adhered to the rules of the grammar. Their system comprised of the following key modules: a) Analysing text coupled with parsing Syntax b) Representation using LFG f-structure c) Transferring grammar rules and finally d) Generating proper ISL sentences . Minipar Parser was used to parse the input sentence and the parse tree was used to construct a dependency structure. An f-structure is generated which encoded the grammatical relation of the input sentence. When we say Grammatical relation of the input sentence, we essentially refer to the subject, object and tense of the sentence. We represented this This information as a set of attribute-value pairs. Each attribute corresponded to the actual grammatical symbol name. On applying proper grammar transfer rules, the English f-structure generated was converted to Indian Sign Language f-structure. It needs to be mentioned that evaluating this system is extremely difficult due to the unavailability of a proper, official ISL written orthography.

2.2 Basic Concept

First, we use the webrtc SpeechRecognition to capture audio as input. We then use the Chrome/Google Speech API to transform the audio to text. Currently, we use NLP (natural language processing) to break down the material into smaller, more easily comprehensible chunks. We have a reliance parser that analyses the sentence's grammatical structure and builds up the word connections. Finally, we converted audio/text into Sign language and user will get videos/clips as sign language for given input.

2.3 Google Speech API

A Speech-to-Text API synchronous recognition request is the simplest method for performing recognition on speech audio data. Speech-to-Text can process up to 1 minute of speech audio data sent in a synchronous request. After Speech-to-Text processes and recognizes all of the audio, it returns a response.

A synchronous request is blocking, meaning that Speech-to-Text must return a response before processing the next request. Speech-to-Text typically processes audio faster than real-time, processing 30 seconds of audio in 15 seconds on average. In cases of poor audio quality, your recognition request can take significantly longer.

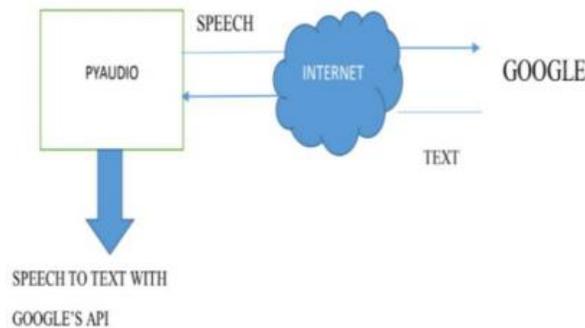


Fig. 2.1 Conversion of speech to text through Google speech API

Noise removal is the process of removing the unwanted noise or any absurd noise from the input data which is in takes in terms of speech. Different types of noise removal techniques are Filtering technique, spectral restoration and many more. Modulation detection and synchrony detection are the two noise removal techniques. Since the speech from the user or the normal person is taken using a microphone of computer or a cellular phone clarity of sound may not be guaranteed therefore it is sent to the noise removal.

2.4 Unsupervised Algorithm for Text Document

Text classification is a problem where we have fixed set of classes/categories and any given text is assigned to one of these categories. In contrast, Text clustering is the task of grouping a set of unlabeled texts in such a way that texts in the same group (called a cluster) are more similar to each other than to those in other clusters.

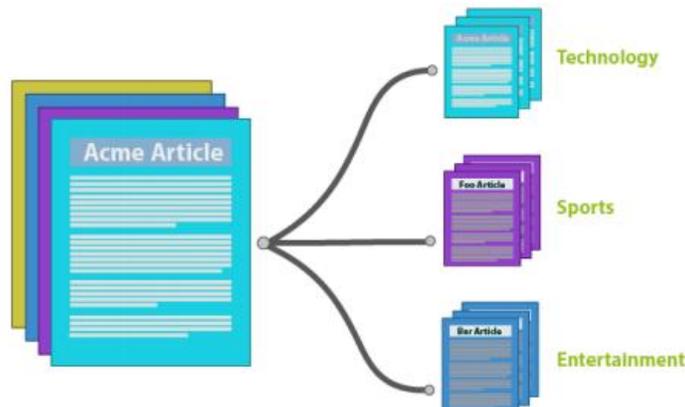


Fig. 2.2 Unsupervised algorithm for text mining

In information retrieval or text mining, the term frequency-inverse document frequency also called TF-IDF, is a well-known method to evaluate how important is a word in a document. TF-IDF are also a very interesting way to convert the textual representation of information into a Vector Space Model (VSM).

Google has already been using TF*IDF (or TF-IDF, TFIDF, TF.IDF, Artist formerly known as Prince) as a ranking factor for your content for a long time, as the search engine seems to focus more on term frequency rather than on counting keywords. TF*IDF is an information retrieval technique that weighs a term's frequency (TF) and its inverse document frequency (IDF). Each word or term has its respective TF and IDF score.

$$w_{i,j} = tf_{i,j} \times \log \left(\frac{N}{df_i} \right)$$

Fig. 2.3 TF-IDF weights formula

The product of the TF and IDF scores of a term is called the TF*IDF weight of that term. The TF*IDF algorithm is used to weigh a keyword in any content and assign the importance to that keyword based on the number of times it appears in the document. More importantly, it checks how relevant the keyword is throughout the web, which is referred to as corpus.

2.5 Natural Language Processing (NLP)

Natural language processing (NLP) refers to the branch of computer science—and more specifically, the branch of artificial intelligence or AI—concerned with giving computers the ability to understand text and spoken words in much the same way human beings can. NLP combines computational linguistics—rule-based modelling of human language with statistical, machine learning, and deep learning models. Together, these technologies enable computers to process human language in the form of text or voice data and to ‘understand’ its full meaning, complete with the speaker or writer’s intent and sentiment. NLP drives computer programs that translate text from one language to another, respond to spoken commands, and summarize large volumes of text rapidly—even in real time. There’s a good chance you’ve interacted with NLP in the form of voice-operated GPS systems, digital assistants, speech-to-text dictation software, customer service chatbots, and other consumer conveniences. But NLP also plays a growing role in enterprise solutions that help streamline business operations, increase employee productivity, and simplify mission-critical business processes.

A ConvNet is able to successfully capture the Spatial and Temporal dependencies in an image through the application of relevant filters. The architecture performs a better fitting to the image dataset due to the reduction in the number of parameters involved and reusability of weights. In other words, the network can be trained to understand the sophistication of the image better.

2.5.1 NLP Task

Several NLP tasks break down human text and voice data in ways that help the computer make sense of what it's ingesting. Some of these tasks include the following:

Speech recognition, also called speech-to-text, is the task of reliably converting voice data into text data. Speech recognition is required for any application that follows voice commands or answers spoken questions. What makes speech recognition especially challenging is the way people talk—quickly, slurring words together, with varying emphasis and intonation, in different accents, and often using incorrect grammar.



Fig. 2.4 Speech tagging

Part of speech tagging, also called grammatical tagging, is the process of determining the part of speech of a particular word or piece of text based on its use and context. Part of speech identifies ‘make’ as a verb in ‘I can make a paper plane,’ and as a noun in ‘What make of car do you own?’ Sentiment analysis attempts to extract subjective qualities—attitudes, emotions, sarcasm, confusion, suspicion—from text.

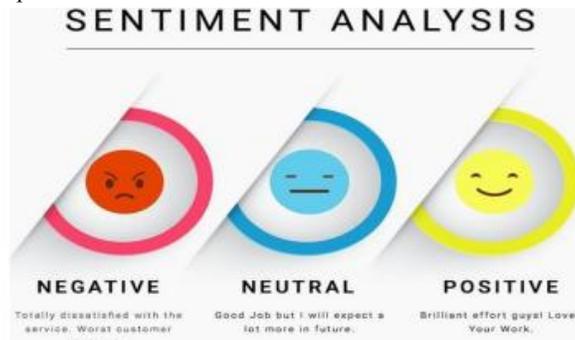


Fig. 2.5 Sentiment analysis

Natural language generation is sometimes described as the opposite of speech recognition or speech-to-text; it's the task of putting structured information into human language.

2.5.2 Natural Language Toolkit (NLTK)

The Python programming language provides a wide range of tools and libraries for attacking specific NLP tasks. Many of these are found in the Natural Language Toolkit, or NLTK, an open source collection of libraries, programs, and education resources for building NLP programs. The NLTK includes libraries for many of the NLP tasks listed above, plus libraries for subtasks, such as sentence parsing, word segmentation, stemming and lemmatization (methods of trimming words down to their roots), and tokenization (for breaking phrases, sentences, paragraphs and passages into tokens that help the computer better understand the text). It also includes libraries for implementing capabilities such as semantic reasoning, the ability to reach logical conclusions based on facts extracted from text.

2.5.3 Statistical NLP, Machine Learning, and Deep Learning

The earliest NLP applications were hand-coded, rules-based systems that could perform certain NLP tasks but couldn't easily scale to accommodate a seemingly endless stream of exceptions or the increasing volumes of text and voice data. Enter statistical NLP, which combines computer algorithms with machine learning.

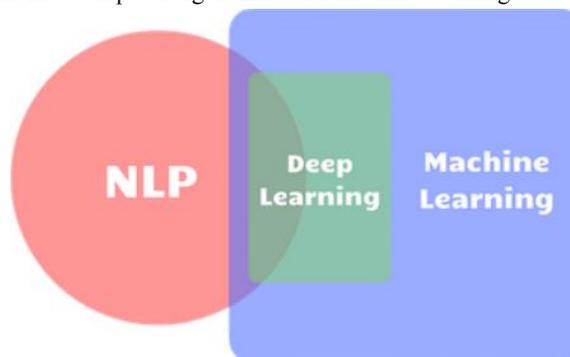


Fig. 2.6 Deep learning

Deep learning models to automatically extract, classify, and label elements of text and voice data and then assign a statistical likelihood to each possible meaning of those elements. Today, deep learning models and learning techniques based on convolutional neural networks (CNNs) and recurrent neural networks (RNNs) enable NLP systems that 'learn' as they work and extract ever more accurate meaning from huge volumes of raw, unstructured, and unlabelled text and voice data sets.

III. SYSTEM ANALYSIS

3.1 Overview

This project has a scope, It converts the audio message into the sign language. The system takes audio as input, converts this audio recording message into text and displays the relevant Indian Sign Language clips or GIFs which are predefined. By using this system, the communication between normal and deaf people gets easier. With the technology developing day by day need for better system that can give best results have a great scope and this project as well has marked a huge scope for speech to sign language. The target audience for this system is not limited to hearing impaired individuals as communication for hearing impaired people in common places like railway stations, bus stands, banks, hospitals etc. is very difficult because a vocal person may not understand sign language and thus won't be able to convey any message to hearing impaired person. Thus, it is targeted to all those who wish to learn this translation, to facilitate better communication

3.2 Existing System

Although sign language is used across the world to bridge the gap of communication for hearing or speech impaired which depend mostly on sign language for day to day communication, there are not efficient models that convert text to Indian sign language. There is a lack of proper and effective audio visual support for oral communication. While significant progress has already been made in computer recognition of sign languages of other countries but a very limited work has been done in ISL computerization. Work done so far in this field has been much more focused on American sign language (ASL) or British sign language, but for Indian sign language, systems that have been developed are very few.

3.3 Proposed system

Few works have been done to generate a system that is based on the above concepts listed in the existing approaches section and cater to Indian sign language. Thus we propose to develop one for Indian sign language based on transfer based translation. The success of this translation system will depend on the conversion of English text to Indian sign language bearing its lexical and syntactic knowledge. Our objective is to help people suffering from the problem of hearing. There have been many projects done on the sign languages that convert sign language as input to text or audio as output. But audio to sign language conversion systems have been rarely developed. It is useful to both normal and deaf people. In this project we introduce new technology that is audio to sign language translator using python. In this it takes audio as input, display the text on screen and finally it gives sign code/language of given input. All the words in the sentence are then checked against the words in the dataset containing videos and GIFs representing the words. If the words are not found, it splits the words into an individual letter and show the corresponding videos/clips which are predefined in the system. In this section we will discuss about our project. Our system consists of four main steps: input audio or text, tokenizing the input, searching the words/letter form dataset and display videos/clips.

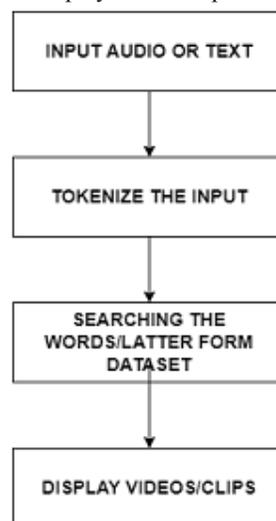


Fig.3.1 Proposed System

3.4 Hardware and Software Requirements

The system will require the basic facilities needed to develop a system and implement it. Major requirements are as follows.

3.4.1 Hardware requirements

1. Disk Space of around 50 Gigabyte.
2. Intel Atom processor or Intel Core i3 processor.
3. A GPU (1GB) is recommended for training and for inference speed, but is not mandatory.
4. RAM (4GB)
5. A microphone
6. A keyboard

3.4.2 Software requirements

1. Both Windows and Linux are supported.
2. Python \geq 3.6
3. Chrome or other browsers
4. Internet connectivity

3.5 Advantages and Disadvantages

3.5.1 Advantages

1. It can be applied to higher level applications: This system will extract an input audio features. This output can be applied to higher level applications.
2. Takes lesser time: It applies clustering by extracting audio features and it tends to consume lesser time than to compare separately with every other sign language on the dataset.
3. More accurate results: As it uses speech features instead of metadata the sign language comparison is more accurate and thus we can achieve higher precision of results.
4. Easy user interface: The system is implemented as a simple and easy-to-use user interface kept in mind. Any normal user can easily retrieve their features without any hindrance.

3.5.2 Disadvantages

1. Limited training dataset: The project is currently implemented on a finite dataset stored in a folder/personal system. Although it can be expanded there are certain storage constraints to which the project is limited.
2. Size and format constraints: The project can be applied only to .mp4 files as feature extraction is easier for such files. Moreover larger video clips that exceed the limit are hard to analyze as they require larger space to store and process.

IV. SYSTEM DESIGN DETAILS

4.1 DFD with Detailed Explanation

A data flow diagram (DFD) is a graphical representation of the flow of data through an information system, modelling its process aspects. A DFD is often used as a preliminary step to create an overview of the system without going into great detail, which can later be elaborated. DFDs can also be used for visualization of data processing.

A DFD shows what kind of information will be input to and output from the system, how the data will advance through the system, and where the data will be stored. It does not show information about process timing or whether processes will operate in sequence or in parallel, unlike a traditional structured flowchart which focuses on control flow.

The logic data flow diagram can be drawn using four simple notations i.e., to represent a process, and data store. We have used the symbols as per the Gain and Sarson notation. Square boxes represents the external entity, curved boxes show the process, rectangular open boxes denotes the data store and arrow marks represents the flow of data.



The dataflow diagram has various levels. The level 0 DFD also called the context level, represents the entire software as a single element. Additional process and information flow parts is represented in the next level, i.e., Level 1 DFD. Any process, which is complex in Level 1, will be further represented into sub function in the next level i.e., Level 2 and so on.

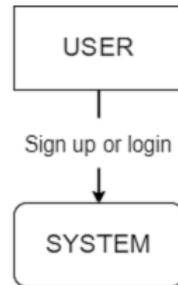


Fig.4.1 Level 0 DFD

4.2 UML diagrams

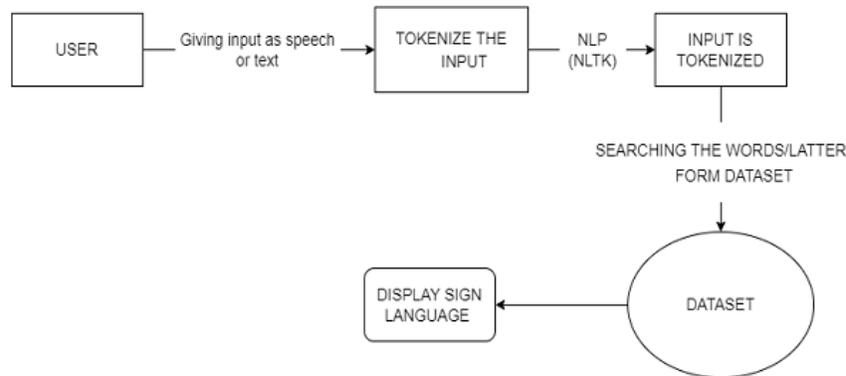


Fig.4.2 Level 1 DFD

The Unified Modelling Language (UML) is a general-purpose, developmental, modelling language in the field of software engineering that is intended to provide a standard way to visualize the design of a system. UML offers a way to visualize a system's architectural blueprints in a diagram including elements such as:

- any activities (jobs);
- individual components of the system;
- and how they can interact with other software components;
- how the system will run;
- how entities interact with others (components and interfaces);
- External user interface.

Although originally intended for object-oriented design documentation, UML has been extended to a larger set of design documentation (as listed above) and has been found useful in many contexts.

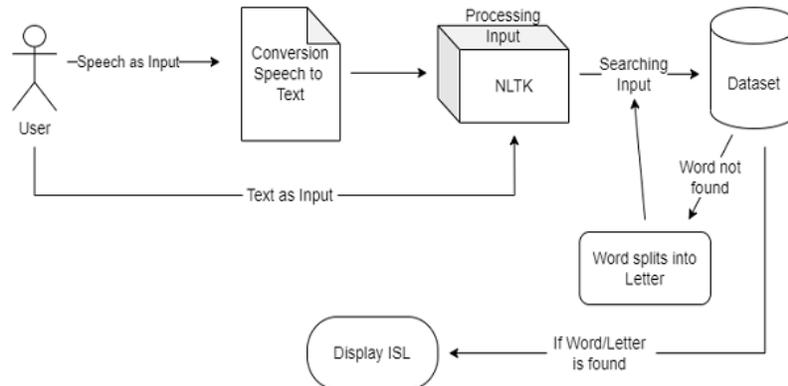


Fig. 4.3 UML diagram for system

4.2.1 Flow Diagram:

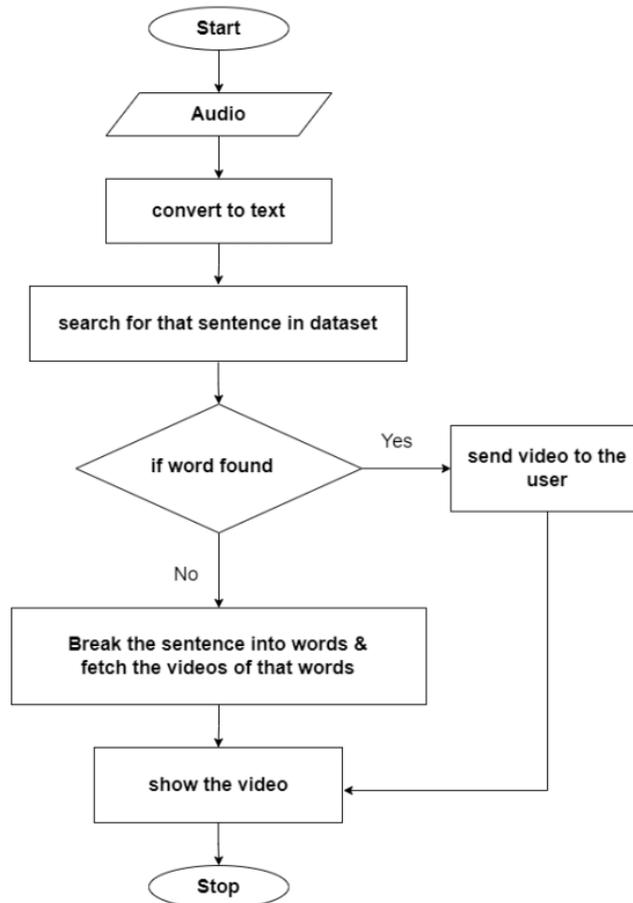


Fig.4.4 Flow diagram of the proposed system

4.3 Proposed Algorithm

The system is then broken down into 4 sequential components:

1. Signup or login
2. Take Input from user
3. Process the input through NLTK

4. Display the relevant sign language

4.3.1 Algorithm

1. Open Web Application.
2. Signup or login.
3. Input the text or click on microphone to speak.
4. Click on submit.
5. Input is process by system.
6. Start button for display of animation.
7. Shows the Required result.
8. Close.

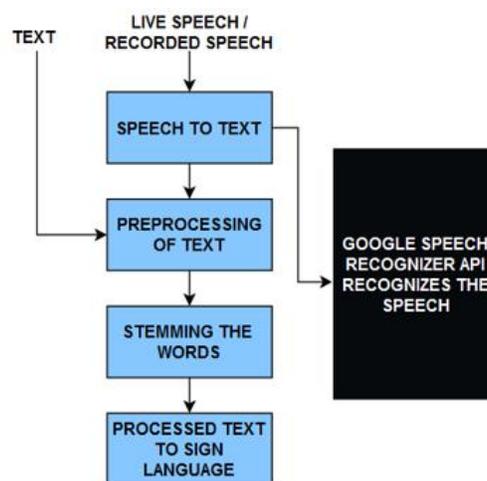


Fig.4.5 Block diagram of proposed system

V. IMPLEMENTATION

5.1 Overview

World's normal people has been in a difficult situation in the society because of their inability to communicate vocally with hard hearing people in connection with that the indifference of others to learn their language, the sign language. With the arrival of multimedia, animation and other computer technologies, it is now becoming possible to bridge the communication gap between the normal and hearing-impaired person. Sign language is a visual/gestural language that serves as the primary means of communication for hard hearing individuals, just as spoken languages are used among the hearing. Hard hearing individuals encounter the difficulty that most hearing individuals communicate with spoken language.

1. First, we use the webkitSpeechRecognition to capture audio as input.
2. We then use the Chrome/Google Speech API to transform the audio to text.
3. Currently, we use NLP (natural language processing) to break down the material into smaller, more easily comprehensible chunks.
4. We have a reliance parser that analyses the sentence's grammatical structure and builds up the word connections.
5. Finally, we converted audio into Sign language and user will get videos/clips as sign language for given input

5.1.1 Forms of Input

Our project is intended to get inputs in multiple formats. The inputs can be of forms:

- Text input
- Live speech input.

5.1.2 Speech Recognition

The live speech is received as input from the microphone of our system. This is done using the Python package PyAudio. PyAudio is a Python package that is used to record audio on a variety of platforms. The audio thus received is converted into text using Google Speech Recognizer API. It is an API that helps to convert audio to text by incorporating neural network models. In the input format of giving the audio file, the received audio is translated into text by using this Google Speech Recognizer. For lengthier audio files, the audio is divided into smaller chunks based on the occurrence of silence. The chunks are then passed into the Google Speech Recognizer to efficiently convert into text.

5.1.3 Pre-processing of text

The filler words which are used to fill the gap in the sentence are apparently lesser- meaning words. They provide less context to the sentence. There are around 30+ filler words in the English Language which hardly makes sense in the sentence. So, the system removes the filler words from the sentence and makes it more meaningful. By removing these words, the system will save time.

5.1.4 Porter Stemming Algorithm

Porter Stemming algorithm provides a basic approach to conflation that may work well in practice. Natural Language Processing (NLP) helps the computer to understand the human natural language. Porter Stemming is one of the Natural Language Processing techniques. It is the famous stemming algorithm proposed in 1980. Porter Stemmer algorithm is known for its speed and ease. It is mainly used for data mining and to retrieve information. It produces better results than any other stemming algorithms. It has less error rate.

The system removes the morphological and inflexional endings of the English words. The system uses Porter stemming Algorithm to remove the commonly used suffixes and prefixes of the words and find the root word or original word. For example, the Porter stemming algorithm reduces the words “agrees”, “agreeable”, “agreement” to the root word “agree”. Because of this stemming, we can reduce the time taken for searching the sign language for the given word.

5.1.5 Text to Sign Language

The system iterates through every word in the processed text sentence which is received from the previous step and searches the corresponding sign language video sequences in the local system. If the word is found, the system shows the output as a video sequence. If the word is not found in the local system, then it splits the word into letters, according to letter the sign video clips are play.

5.2 Technologies used

5.2.1 HTML (Hyper Text Markup Language)

The Hypertext Markup Language or HTML is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript. Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

HTML elements are the building blocks of HTML pages. With HTML constructs, images and other objects such as interactive forms may be embedded into the rendered page. HTML provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. HTML elements are delineated by tags, written using angle brackets. Tags such as `` and `<input />` directly introduce content into the page. Other tags such as `<p>` surround and provide information about document text and may include other tags as sub-elements. Browsers do not display the HTML tags but use them to interpret the content of the page.

5.2.2 CSS (Cascading Style Sheets)

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language such as HTML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript. CSS is designed to enable the separation of presentation and content, including layout, colors, and fonts. This separation can improve content accessibility; provide more flexibility and control in the specification of presentation characteristics;

enable multiple web pages to share formatting by specifying the relevant CSS in a separate .css file, which reduces complexity and repetition in the structural content; and enable the .css file to be cached to improve the page load speed between the pages that share the file and its formatting.

Separation of formatting and content also makes it feasible to present the same markup page in different styles for different rendering methods, such as on-screen, in print, by voice (via speech-based browser or screen reader), and on Braille-based tactile devices. CSS also has rules for alternate formatting if the content is accessed on a mobile device.

5.2.3 Python

Python is a high-level, general-purpose programming language. Its design philosophy emphasizes code readability with the use of significant indentation. Its language constructs and object-oriented approach aim to help programmers write clear, logical code for small- and large-scale projects. Python is dynamically-typed and garbage-collected. It supports multiple programming paradigms, including structured (particularly procedural), object-oriented and functional programming. It is often described as a "batteries included" language due to its comprehensive standard library. Guido van Rossum began working on Python in the late 1980s as a successor to the ABC programming language and first released it in 1991 as Python 0.9.0. Python 2.0 was released in 2000 and introduced new features such as list comprehensions, cycle-detecting garbage collection, reference counting, and Unicode support. Python 3.0, released in 2008, was a major revision that is not completely backward-compatible with earlier versions. Python 2 was discontinued with version 2.7.18 in 2020. Python consistently ranks as one of the most popular programming languages.

5.2.4 Django framework

Django's primary goal is to ease the creation of complex, database-driven websites. The framework emphasizes reusability and "pluggability" of components, less code, low coupling, rapid development, and the principle of don't repeat yourself. Python is used throughout, even for settings, files, and data models. Django also provides an optional administrative create, read, update and delete interface that is generated dynamically through introspection and configured via admin models. Some well-known sites that use Django include Instagram, Mozilla, Disqus, Bitbucket, Nextdoor and Clubhouse.

5.3 NLTK Library

NLTK has been called "a wonderful tool for teaching, and working in, computational linguistics using Python," and "an amazing library to play with natural language."

5.3.1 word tokenizes

We use the method `word_tokenize()` to split a sentence into words. The output of word tokenization can be converted to Data Frame for better text understanding in machine learning applications. It can also be provided as input for further text cleaning steps such as punctuation removal, numeric character removal or stemming. Machine learning models need numeric data to be trained and make a prediction. Word tokenization becomes a crucial part of the text (string) to numeric data conversion. Please read about Bag of Words or CountVectorizer. Please refer to below word tokenize NLTK example to understand the theory better.

5.3.2 Elimination of Stop Words

Since ISL deals with words associated with some meaning, unwanted words are removed these include various parts of speech such as TO, POS(possessive ending), MD(Modals), FW(Foreign word), CC(coordinating conjunction), some DT(determiners like a, an, the), JJR, JJS(adjectives, comparative and superlative), NNS, NNPS(nouns plural, proper plural), RP(particles), SYM(symbols), Interjections, non-root verbs.

5.3.3 Lemmatization and Synonym replacement

Indian sign language uses root words in their sentences. So we convert them to root form using Porter Stemmer rules. Along with this each word is checked in bilingual dictionary, if word does not exist, it is tagged to its synonym containing the same part of speech.

5.3.4 WordNet

WordNET is a lexical database of words in more than 200 languages in which we have adjectives, adverbs, nouns, and verbs grouped differently into a set of cognitive synonyms, where each word in the database is expressing its distinct concept. The cognitive synonyms which are called synsets are presented in the database with lexical and semantic relations. WordNET is publicly available for download and also we can test its network of related words and concepts using this link. Below are a few test images when accessed this through the browser. "Wordnet"

5.3.5 Punkt

Punkt is designed to learn parameters (a list of abbreviations, etc.) unsupervised from a corpus similar to the target domain. The pre-packaged models may therefore be unsuitable: use ``PunktSentenceTokenizer(text)`` to learn parameters from the given text.

VI. TESTING AND RESULT

6.1 Testing

Testing is the process of executing a program with the intent of finding an error. Testing is a crucial element of software quality assurance and presents ultimate review of specification, design and coding. System testing is an important phase. Testing represents an interesting anomaly for the software. Thus, a series of testing are performed for the proposed system before the system is ready for user accepting testing. A good test case is one that has a high probability of finding an undiscovered error. A successful test is one that uncovers an as undiscovered error.

6.1.1 Testing Objectives

1. Testing is a process of executing a program with the intent of finding an error
2. A good test case is one that has a probability of finding an as yet undiscovered error
3. A successful test is one that uncovers an undiscovered error.

6.1.2 Testing principles

- All tests should be traceable to end user requirements
- Tests should be planned long before testing begins
- Testing should begin on a small scale and progress towards testing in large
- Exhaustive testing is not possible
- To be most effective testing should be conducted by an independent third party.

The primary objective for test case design is to derive a set of tests that has the highest livelihood for uncovering defects in the software. To accomplish the objective two different categories of test case design techniques are used. They are

- White box testing
- Black box testing

A. White Box Testing

White box testing focuses on the program control structure. Test cases are derived to ensure that all statements in the program have been executed at least once | conditions have been executed

B. Black Box Testing

Black box testing is designed to validate functional requirements without regard to the internal workings of a program. Black box testing mainly focuses on the information domain of the software, deriving test cases by partitioning input and output in a manner that provides through test coverage. Incorrect and missing functions, interface errors, errors in data structures, error in functional logic are the errors falling in this category.



6.2 System Testing Plan

Sr. No.	Test Case Description	Expected Result	Actual Result	Test Case Criteria (P/F)
1.	Sign up successfully or not	Sign up successfully	Sign up successfully	P
2.	Login successfully or not	Login successfully	Login successfully	P
3.	Input text uploaded successfully or not	Input text uploaded successfully	Input text uploaded successfully	P
4.	Input speech uploaded successfully or not	Input speech uploaded successfully	Input speech uploaded successfully	P
5.	Microphone button working or not	Microphone button working properly	Microphone button working properly	P
6.	All buttons working properly or not.	All buttons working properly	All buttons working properly	P
7.	Inputs are successfully processed or not.	Inputs are successfully processed	Inputs are successfully processed	P
8.	Sign language is showing properly or not	Sign language is showing properly	Sign language is showing properly	P

6.3 Screenshots

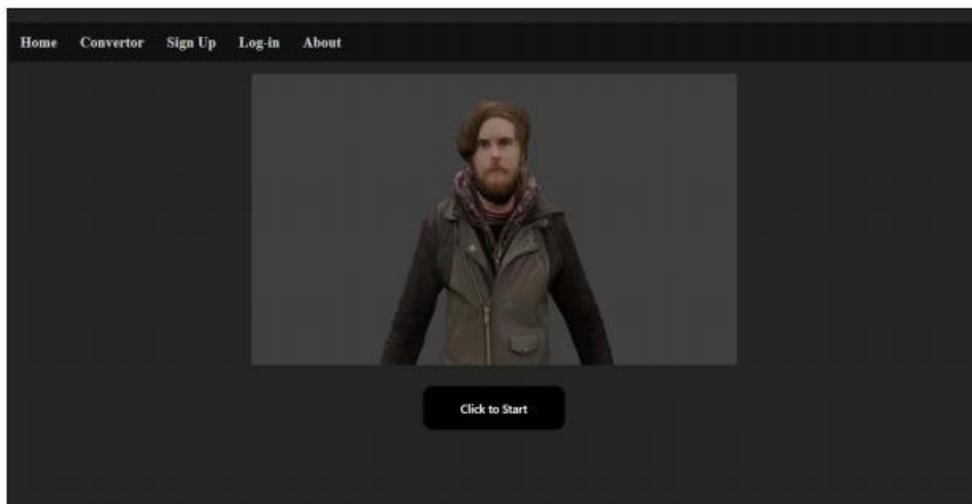


Fig.6.1 Screenshot of the Home

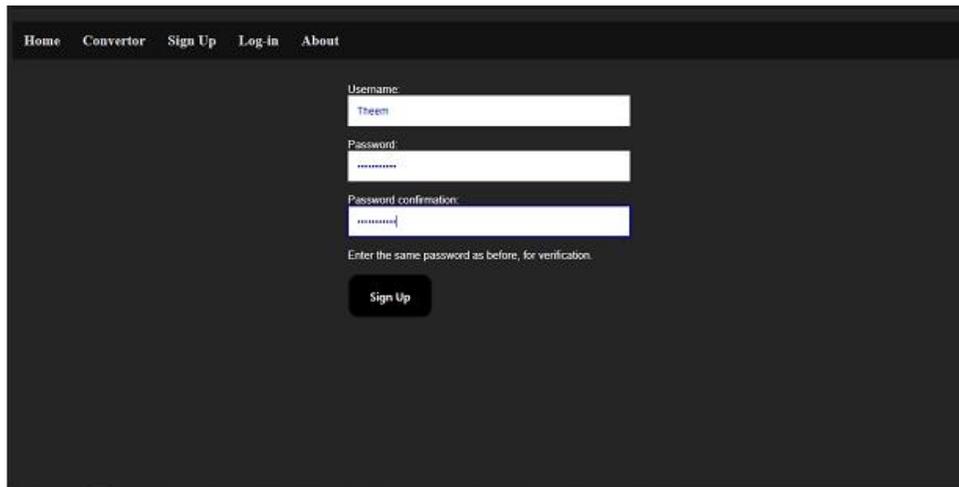


Fig.6.2 Screenshot of Sign up

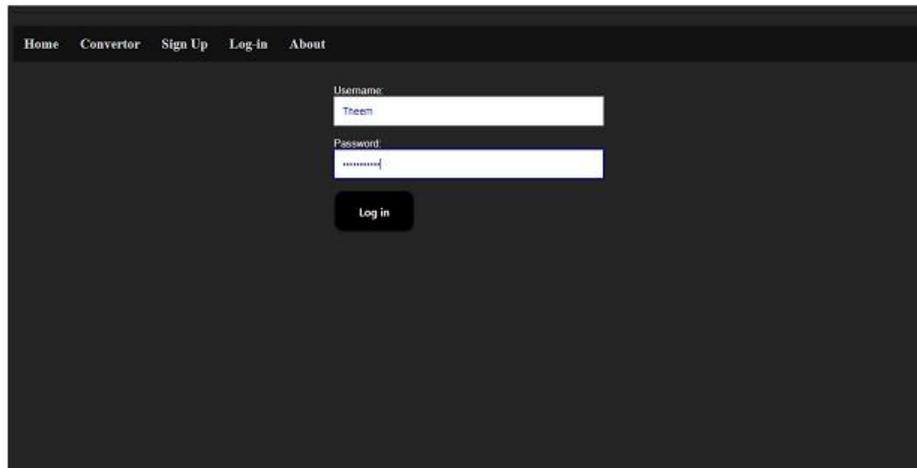


Fig.6.3 Screenshot of Login

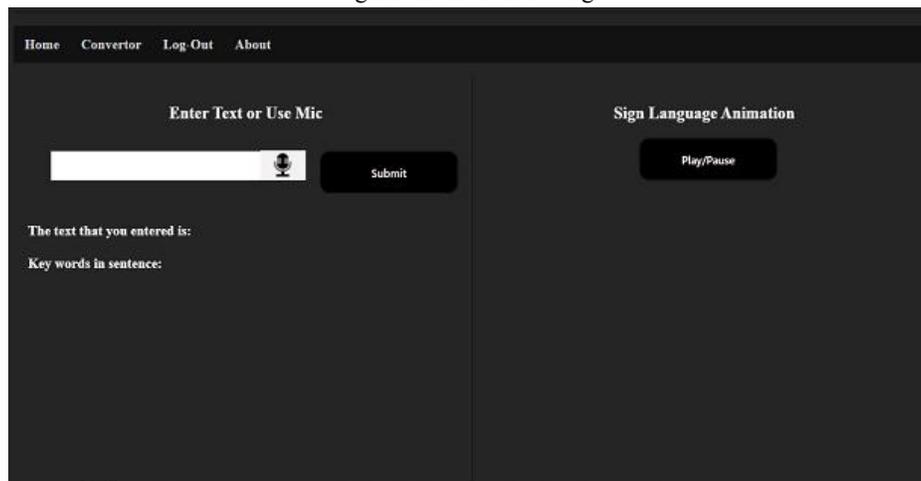


Fig.6.4 Screenshot of Converter

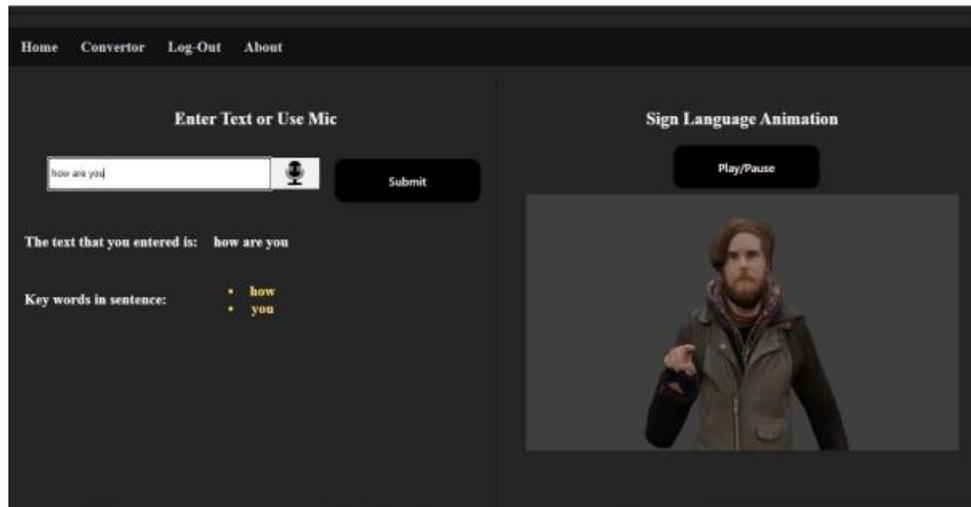


Fig.6.5 Signifies “how are you” as sign language

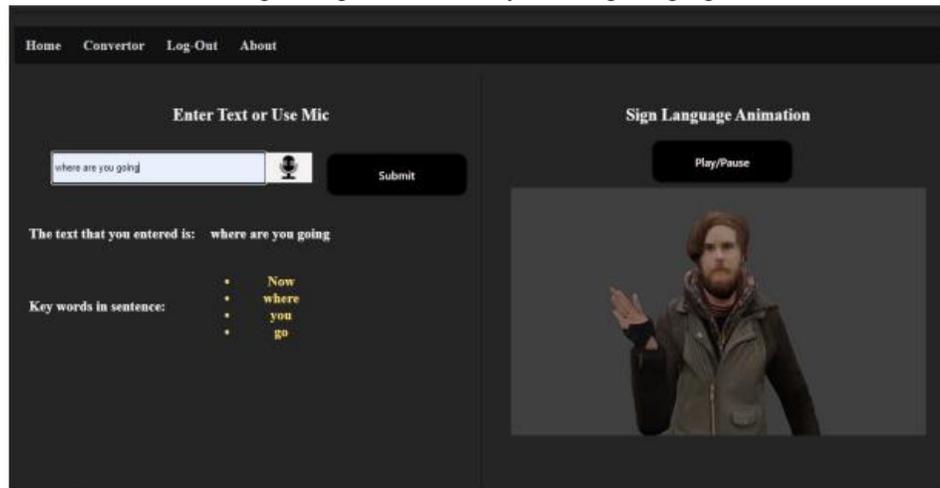


Fig.6.6 Signifies “where are you” as sign language

VII. CONCLUSION AND FUTURE SCOPE

7.1 Conclusion

A significant section of the Indian society suffers from hearing and speech impairment. This population uses Indian Sign Language as their primary mode of communication. Due to the difficulty in learning and understanding the meaning and context of written texts, sign language is preferred. Sign language involves the usage of hands, lip movements and expressions in order to communicate words, emotions and sounds. The proposed system provides an efficient method to aid communication between an individual with hearing and speech impairment. It is a field that has had little development over the years particularly in successful implementation in Python programming language. The system would improve access to information for the hearing-impaired population of the country like India. Moreover, the system can also act like an educational tool to learn ISL.

Here, we have attempted to create a model that will allow people with disabilities to express themselves distinctly, which will help them blend with the rest of the world without any difficulty. Our proposed model will successfully convert the given input audio into an animation. Many improvements along this route can be made as and when the ISL Dictionary grows. The words in the ISL are small, so many improvements can be made by adding new words to their dictionary to increase their breadth. In addition, text-and-speech integration can be done on a project to enable Monaural / Speech to

Indian Sign Language Translator better communication techniques that will allow users to convert Text into Indian Sign Language by hand-input text.

7.2 Future Scope

- In future, the proposed approach will be tested against unseen sentences. Furthermore, machine translation approach will be studied and implemented on parallel corpora of English and ISL sentences. The ISL corpus will be used for testing ISL sentences and the performance will be evaluated with evaluation parameters.
- This could enable sign language users to access personal assistants, to use text- based systems, to search sign language video content and to use automated real-time translation when human interpreters are not available. With the help of AI, automated sign language translation systems could help break down communication barriers for deaf individuals.
- Various front-end options are available such as .net or android app, that can be used to make the system cross platform and increase the availability of the system.
- The system can be extended to incorporate the knowledge of facial expressions and body language too so that there is a complete understanding of the context and tone of the input speech.
- A mobile and web based version of the application will increase the reach to more people.
- Integrating hand gesture recognition system using computer vision for establishing 2-way communication system.
- We can develop a complete product that will help the speech and hearing impaired people, and thereby reduce the communication gap.

REFERENCES

- [1]. <https://www.kaggle.com/datasets/vaishnaviasonawane/indian-sign-language- dataset/code>
- [2]. M.Elmezain,A.Al-Hamadi,J.Appenrodt and B.Michaelis, A Hidden Markov Model- based Continuous Gesture Recognition System for Hand Motion Trajectory, 19th International Conference on IEEE, Pattern Recognition, 2008, ICPR 2008, pp. 1–4, (2008).
- [3]. P. Morguet and M. Lang M, Comparison of Approaches to Continuous Hand Gesture Recognition for a Visual Dialog System,IEEE International Conference on IEEE Acoustics, Speech, and Signal Processing, 1999, Proceedings, 1999, vol. 6, pp. 3549–3552, 15– 19March(1999).
- [4]. Rao, R R, Nagesh, A, Prasad, K. and Babu, K E (2007) Text-Dependent Speaker Recognition System for Indian Languages.International Journal of Computer Science and Network Security, Vol. 7, No.11
- [5]. T.Starner,“Visual Recognition of American Sign Language Using Hidden Markov Models,” Master’s thesis, MIT, Media Laboratory, Feb. 1995.
- [6]. Neha Poddar, Shrushti Rao, Shruti Sawant, Vrushali Somavanshi, Prof.Sumita Chandak "Study of Sign Language Translation using Gesture Recognition" International Journal of Advanced Research in Computer and Communication Engineering Vol. 4, Issue 2, February 2015.
- [7]. Deaf Mute Communication Interpreter Anbarasi Rajamohan, Hemavathy R., Dhanalakshmi M.(ISSN: 2277-1581) Volume 2 Issue 5, pp: 336-341 1 May 2013.
- [8]. Zouhour Tmar, Achraf Othman & Mohamed Jemni: A rule-based approach for building an artificial English-ASL corpus <http://ieeexplore.ieee.org/document/6578458/>
- [9]. Dictionary | Indian Sign Language. (n.d.). Retrieved July 15, 2016, from <http://indiansignlanguage.org/dictionary>
- [10]. P. Kar, M. Reddy, A. Mukherjee, A. M. Raina. 2017. INGIT: Limited Domain Formulaic Translation from Hindi Strings to Indian Sign Language. ICON.
- [11]. M. Vasishta, J. Woodward and S. DeSantis. 2011. An Introduction to Indian Sign Language. All India Federation of the Deaf (Third Edition)
- [12]. V. Lpez-Ludea, C. Gonzalez-Morcillo, J.C. Lpez, E. Ferreiro, J. Ferreiros, and R. San- Segundo. Methodology fordeveloping an advanced communications system for the deaf in a new domain. Knowledge-Based Systems, 56:240 – 252, 2014.