ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 9, Issue 7, July 2022

EquiSign Dynamic Sign Language Translator

[1] V. Valarmathi, [2] S. Sowmiya, [3] M.Viswanathan*

[1] Guide, Sri Sairam Engineering College, Tambaram, Chennai, Tamil Nadu, India.
[2] [3] Sri Sairam Engineering College, Tambaram, Chennai, Tamil Nadu, India.
Corresponding Author Email: viswanathanmukundan@gmail.com

Abstract*— This paper aims to address the problem of limited knowledge and outreach of the concept and structure of sign languages in order to facilitate better inclusion and integration of hearing-impaired people among their peers and general society as a whole. For the proposed system, the speech input is captured in real-time, converted into text using a machine-learning model based on natural language processing, and translate the text into the specified flavor of the sign language. The proposed system currently makes use of Python as it's core tech-stack.

Keywords*—sign language, hearing-impaired, machine learning, natural language processing, sign language, Python.

I. INTRODUCTION

Diversity & Inclusion is one of the trending and important needs of the current world, which aims to make the world a better and more supportive place for differently-abled people. One such group that comes under this category is that of hearing-impaired people, who may not be able to communicate in the conventional method. To support them, an ingenious system of sign-languages was developed, which consisted of visual gestures made using the hands and fingers so that these people could remain on an equal footing with their peers. But, every good system comes with its own set of problems. And the problem with this system is it's limited outreach and presumed difficulty in learning, both by the impaired people as well as the non-impaired ones. They might find it confusing to keep track of the gestures, especially the new learners. What would help in this situation is the availability of a guide or other similar device, which could act as an automatic, ever-present translator. This would enable smoother communication among the people and also act as a facilitator and promoter to D&I. With some extra work, the different 'flavours' of sign language, such as Indian Sign Language, American Sign Language, British Sign Language, and various other regional ones could also be accommodated in order to make the world an easier, better place for those who require it. With automation and dynamic nature incorporated into this language field, it could also help in the development of further systems that provide services like sub-titling the spoken transcripts in movies/videos, etc. so that, specialized cases of people like those who were born with impaired hearing and couldn't get a chance to learn the conventional language can use these auto-generated subtitles to keep track of the video contents. Everyone deserves to be heard and understood, and the language used in this process should be treated only as a medium and must never become a barrier or a hindrance.

II. LITERATURE SURVEY

- In the study of [1], Elmahgiubi et al. have developed a smart glove which is capable of capturing the real-time gestures of the hand when worn by the user and interprets the gestures as readable text. According to them it was able to achieve a recognition accuracy of 96% on 20 out of a total of 26 letters, by capturing the data in a set of commonly available sensors.
- In [2], Harini et al. Have detailed the process of implementing computer vision algorithms in the process of converting sign language to human-readable text.They have made use of image processing and segmentation to get the relevant gesture and made use of noise-filtering techniques in order to achieve better accuracy.
- In [3], Grover et al. Have investigated and made a varied, taxonomic classification of different, existing sign language translation systems.
- In [4], YJ Ku et al. propose the idea of sign language translation using a smart-phone application that records the gestures of the user's hands and translates them by making use of a classification model which was developed from two other pre-trained models used as reference for their study. The application identifies the hand position by first getting the skeleton of the person who is using it, and individually identifies the fingers in order to capture the motions.
- In [5], Siming He makes use of deep learning and Recurrent Neural Networks (RNN) in order to tackle this problem of translation from visual to aural language. Since deep learning is proven to be more powerful than ordinary machine learning and can give better results, this paper was a key point in the literature survey to build the project.
- In [6], Badhe and Kulkarni have made an inclusion of some simple, common phrases apart from the individual alphabets and dictionary words for

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 9, Issue 7, July 2022

translating from text to ISL. They have managed to achieve an accuracy of around 97.5% on a self-created dataset consisting of 130000 videos.

- In [7] Sharma, Panda, and Verma have explored how Convolutional Neural Networks can be implemented in the translation process, with an aim to facilitate the real-time conversations through fast and abrupt translation.
- In [8], Sonawane et al. focus on the specific domain of Indian Sign Language. They made use of depth sensing and motion capture abilities of the Microsoft Xbox Kinect in order to capture real-time motion data and gestures and packaged them into an Android application by converting the captured gestures into animations using Unity3D software. Since this is based on ISL, it was the primary point of focus from which the current idea branched out.
- In [9], Kanvinde et al. Have more or less followed standard procedures, but have made use of Unity3D software in order to display the sign-language output in the form of a 3-dimensional animation instead of just a sequence of images or GIF's. They have also designed a smart-phone application capable of bi-directional translation between the different language forms.

III. ARCHITECTURE

The basic idea behind the working architecture is to record the input from the user, and translate it into the other category. Eg. If the user input is audio, it is to be converted into visual and vice-versa.

In order to convert the audio to visual, it must first be converted into the intermediate form of “text” in some suitable language (English, in this case). This can be achieved via some speech-to-text conversion modules available in Python itself, as part of the open source contribution. Once the text is obtained, it has to be processed using some natural language processing techniques before moving on to the conversion. Standard processes such as tokenization, stopword removal, and n- gram analysis are carried out on the piece of text before passing it to the model for analysis. The model then predicts the output and displays it.

For the reverse process, a bit more complexity is involved, since it works with visual input. The correct parts of the video have to be isolated (the user's hands which form the gestures) from the rest of the background. Once isolated, all the gestures are captured in real-time and sent to a pre-trained Convolutional Neural Network which consists of multiple layers in order to analyze the captured images and predict them in order to get the text/audio output.

For recording the gesture input, a camera of any type is needed which is capable of transmitting the recorded motions to the model or the processor that is running the application. In order to view the output for both the directional translations, a display screen, or an audio device is needed.

IV. MODULE DESCRIPTION

The system operates on a web-page application that can either be stand-alone, or be integrated with other websites as an add-on, if it is compatible. The modules used in the software are:

1. Audio Listener module: This module is used in converting audio to sign-form. It is responsible for listening to whatever speech input the people have to offer, and sending it for further processing to the STT module.
2. STT Module: Speech-to-text module. This module takes in the audio input, and converts it into whatever language the user has chosen, as accurately as possible. After converting to text, some standard language processing methods such as tokenization, stopword removal, and n-gram identification are carried out on the obtained text. It then passes the processed text input with all the necessary annotations to the Sign-Generator module.
3. Sign-Generator: This module is responsible for taking in the text input from the STT Module, and intelligently converting it into the specified sign-language flavor. It has to determine whether the input text can be converted into sign as a whole, or whether some intermediate words, which do not have any standard sign notations in the gesture collection, have to be individually spelled out.
4. Gesture Capture module: This module is used in the conversion of sign-language to text. It captures the user's gestures in real-time, and performs necessary preprocessing techniques in order to remove the background objects from the video frames, focus only on the hand and finger gestures of the user, and translate them into the specified language as accurately as possible, by using a pre-trained model which at the most basic level, has to recognize at least all the letters and digits that are present in the alphabet.
5. Display module: Common display module for both conversion directions which displays the output in the appropriate format. In the case of sign output, the module displays it either as a 3-D animation, or as a sequence of pre-collected GIF's or short videos. In the case of text output, it displays the text in a specific area, with options given to the user for changing the display language.
6. Data module: Responsible for storing and maintaining the relevant data to the application and supplying it on demand. It stores all the gestures, in a variety of different poses and angles, captured on both hands, in order for maximum output accuracy of the model.

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 9, Issue 7, July 2022

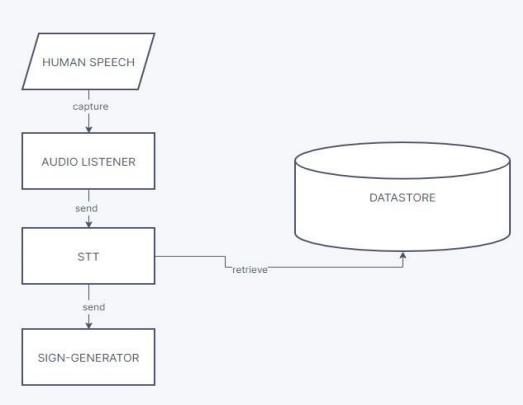

Fig 4.1: High-Level Module Architecture

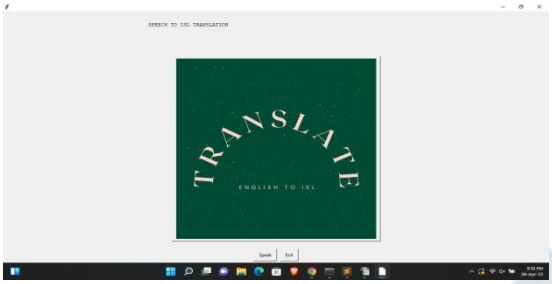

Fig 4.2: User-Page. User Presses the Button to Speak

Fig 4.3: The Spoken Word is Translated into ISL Signs Displayed as a Sequence of Gif's

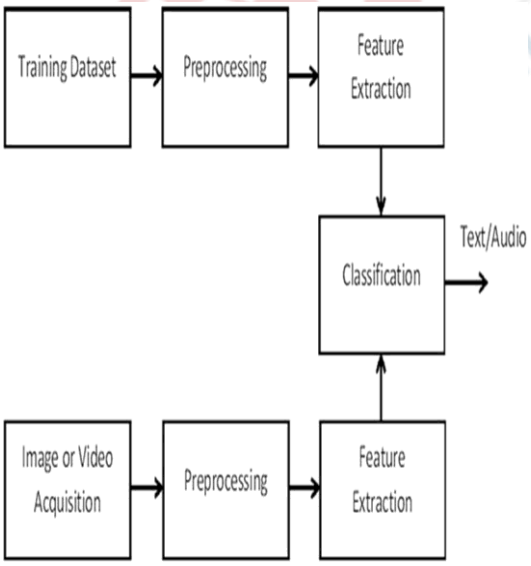

Fig 4.4: Process Flow of the Application

V. TRANSLATING TEXT TO SIGN

One direction of the translation process is to translate spoken words into sign. To do this, the spoken words first have to be captured through the audio input and converted into text in a common language (preferably English). The next step in this process is to remove the "stop-words". Stop words are those words which occur very commonly but do not add any weight or meaning to the text. Examples of stopwords are "a", "the", "is", "are", etc. These words are dropped entirely from the text in order to simplify and shorten the amount of data to be processed in the backend.

After removing stopwords, the remaining words are then passed to the model, which essentially checks the input words against the data store to see if there are any matching signs or sequences. If there is any matching data turned up, it is displayed as video output to the user as a GIF (Graphics Interchange Format).

If no matching sequence can be found in the datastore, the words are split into their individual letters and the corresponding sign for each letter is displayed in a sequence, essentially spelling out the word, letter-by- letter.

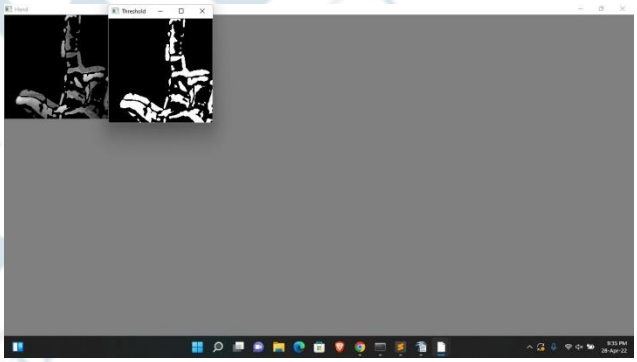

Fig 5.1: Screengrab Depicting How the Sign Shown By the User Is Segmented and Extracted Before Being Sent To the Model for Prediction

VI. TRANSLATING SIGN TO TEXT

This is the reverse process of translation of text to sign, and is more complex since it involves image processing, contour detection, and classification. The hand signs of the communicating person are to be captured as real-time video input, and the correct part of the hand needs to be detected and brought into focus, by removing all other unnecessary background entities. This can be achieved through a combination of edge detection filters, haar- cascade object detection algorithms, and other related computer vision techniques. This process has to be done for each and every captured frame in the video.

Once the relevant part of the image is detected and isolated, the model has to detect the correct hand motion and compare it against the pre-configured images it has been trained on. On getting the correct, matching image pattern, it has to get the text or word or sentence associated with that

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 9, Issue 7, July 2022

pattern, and output it real-time to the user.

VII. TRAINING THE MODEL

For training a model to recognize the correct signs for individual letters/words/phrases,a dataset is needed. This dataset should contain at-least all 26 letters and 10 digits in the English alphabet, formed in the correct sign poses as instructed in the sign-language. Apart from the letters, some common words have their own signs in each sign language. The video format for each of those simple words could also be gathered and saved in the datastore. Further data could be added to contain the pre-defined motions for simple gestures such as "Hello", "Help", "How are you?", etc.

The collected data has to be labeled properly in order to avoid errors and bugs while the model predicts the text. Once the data is properly labeled, it can be passed to the model to train it against the variety of poses, angles, illuminations, and even dominant hands i.e. some people may be right-handed while others may be left-handed. If every person signs only with their dominant hand, it has to be provided as prior training to the model in order to ensure the correct output.

Once the labeling is done, the model has to be constructed out of multiple "layers". Since it's an image recognition model, a CNN (Convolutional Neural Network) is used which makes use of multiple individual layers to break down an image into numeric matrices/tensors and uses them as the actual data to train itself. The CNN is run through multiple epochs (training cycles) and the final accuracy of the model is calculated at the end of the last epoch. Based on the value of accuracy, it can be decided whether the current model is to be used or it has to be retrained with additional data or different configurations.

VIII. FUTURE WORK

- Currently the application is made to support only a single language (Indian sign language). As future work it could be plausible to expand the scope in order to support the other types of sign languages such as American, British, Arabic, etc. In order to do this, extensive data collection and analysis is needed for getting all aspects of the language correct, since each language has its own nuances, styles, and syntax.
- Going in the same line, it can also be configured so that the output text content is translatable into different languages.
- Also, since this is the age of portable devices and technology, it would be a possible use-case to convert this project into a portable application that can run on all popularly supported platforms and enable users to conveniently access it at any time.
- It can also be developed as an API (Application Programming Interface) with all the correct end-points so that any website or other application can integrate it into their architecture for easy access by users of their sites.
- With the constant and continuous evolution of machine learning algorithms, the algorithm used to train the models can be continually updated in order to achieve constantly improving accuracy.
- Going with the focus on cloud technologies and off-premises applications, the API idea can be expanded in order to support a centralized deployment of the application with the subscribers accessing it from the central server, the only requirement being a stable internet connection.
- The application could be further expanded in order to accommodate “Formal” and “Informal” settings in order to tailor the output phrases/signs to the current context.

IX. CONCLUSION

To conclude with, the main aim of this project is to bring an element of inclusiveness to all those who are entitled to it, and facilitate the others in playing their part to bring forth this equality. This can be considered to be one of the key aspects in promoting diversity and inclusiveness, which is considered to be a key point of focus in promoting equality amongst all people, regardless of their physical capabilities, or other distinguishing factors. It also plays an important part in making people realize that everyone is entitled to be heard, that language is merely a medium of communication, and to bridge the gap between different cultures and even physical abilities. The bottom line is that, everyone deserves voice to speak with, and abilities to understand others.

REFERENCES

[1] M. Elmahgiubi, M. Ennajar, N. Drawil and M. S. Elbuni, "Sign language translator and gesture recognition," 2015 Global Summit on Computer & Information Technology (GSCIT), 2015, pp. 1-6, doi: 10.1109/GSCIT.2015.7353332.

[2] R. Harini, R. Janani, S. Keerthana, S. Madhubala and S. Venkatasubramanian, "Sign Language Translation," 2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS), 2020, pp. 883-886, doi: 10.1109/ICACCS48705.2020.9074370.

[3] Y. Grover, R. Aggarwal, D. Sharma and P. K. Gupta, "Sign Language Translation Systems for Hearing/Speech Impaired People: A Review," 2021 International Conference on Innovative Practices in Technology and Management (ICIPTM), 2021, pp. 10-14, doi: 10.1109/ICIPTM52218.2021.9388330.

[4] Y. -J. Ku, M. -J. Chen and C. -T. King, "A Virtual Sign Language Translator on Smartphones," 2019 Seventh International Symposium on Computing and Networking Workshops (CANDARW), 2019, pp. 445-449, doi: 10.1109/CANDARW.2019.00084.

[5] S. He, "Research of a Sign Language Translation System Based on Deep Learning," 2019 International Conference on Artificial Intelligence and Advanced Manufacturing (AIAM), 2019, pp. 392-396, doi: 10.1109/AIAM48774.2019.00083.

[6] P. C. Badhe and V. Kulkarni, "Indian sign language translator using gesture recognition algorithm," 2015 IEEE International Conference on Computer Graphics, Vision and Information Security (CGVIS), 2015, pp. 195-200, doi: 10.1109/CGVIS.2015.7449921.

[7] A. Sharma, S. Panda and S. Verma, "Sign Language to Speech Translation," 2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT), 2020, pp. 1-8, doi: 10.1109/ICCCNT49239.2020.9225422.

[8] P. Sonawane, K. Shah, P. Patel, S. Shah and J. Shah, "Speech To Indian Sign Language (ISL) Translation System," 2021 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS), 2021, pp. 92-96, doi: 10.1109/ICCCIS51004.2021.9397097.

[9] A. Kanvinde, A. Revadekar, M. Tamse, D. R. Kalbande and N. Bakereywala, "Bidirectional Sign Language Translation," 2021 International Conference on Communication information and Computing Technology (ICCICT), 2021, pp. 1-5, doi: 10.1109/ICCICT50803.2021.9510146.