International Journal of Recent Advances in Multidisciplinary Topics
Volume 4, Issue 5, May 2023
https://www.ijramt.com | ISSN (Online): 2582-7839

Mithra-Dhwani – Conversion of Sign to Speech

B. T. Abhilash[1], Amaresh Angadi[2*], N. Bhavyashree[3], C. Manoj[4], G. S. Jayadeva[5]
[1,2,3,4]*Student, Department of Electronics and Communication Engineering, BMS Institute of Technology and Management, Bengaluru, India*
[5]*Professor, Department of Electronics and Communication Engineering, BMS Institute of Technology and Management, Bengaluru, India*

***Abstract*: Sign language is a visual language used by deaf and mute people to communicate. It is a rich and complex language with its own grammar, syntax and vocabulary. Sign language is essential for hearing impaired people as it is their primary means of communication. It allows them to communicate with the world around them, express their thoughts and feelings and participate in social activities. Sign language is also important to the hearing community as it provides a bridge between the deaf and the normal people. By learning sign language, hearing people can communicate with their deaf peers, breaking down communication barriers and promoting inclusivity. Sign language is a vital and valuable language that deserves recognition and support. Here we are using the open-source MediaPipe library provided by Google.**

***Keywords*: hand gesture recognition, MediaPipe, Raspberry Pi, gesture to voice, Indian sign language.**

1. Introduction

Communication is main medium for human beings to connect, communicate and understand one another's ideas and opinions. Speech is often considered the best way to communicate ideas. There are more than 70 million people who are mute or hard of hearing. These people can communicate via postures, body movements, eyes, eyebrows, and hand gestures. People with hearing and/or speech impairments use sign language as their natural mode of communication [3].

Signs are gestures made with one or both hands accompanied by facial expressions with specific meanings. Although deaf, hard-of-hearing, and mute people can easily communicate with each other, integration in educational, social and work environments is a significant barrier for the differently abled. There is a communication barrier between an unimpaired person who is not aware of the sign language system and an impaired person who wishes to communicate. According to the survey conducted by google it is saying that more than 70 million people are deaf or hard of hearing. These people can communicate with facial expression, body movements, eyes, eyebrows and hand movements. Deaf or dumb impairments use sign language as a means of communication. Gestures are handed gestures with facial expressions that have specific meaning. Although deaf, hearing impaired and speech impaired people can communicate easily, integration of education, social and work environment are important implications for people with different abilities. There is a communication barrier between non-disabled people who do not understand the instructions and people with disabilities who want to communicate. But it is one of the most effective way to communicate for the impaired ones and it is one of the growing technique. Therefore, we use gesture recognition using MediaPipe and Raspberry Pi [2].

Many countries have their own languages. There is no universal language. Each country follows its own language and accent are different compared to others. Below ones are some of the different sign languages. There is American Sign Language (ASL), the UK has English (BSL), India has Indian Sign Language (ISL), China's own language, Chinese Sign Language (CSL), and Thailand has Thai Sign Language (TSL). We are using Indian Sign Language (ISL) [1].

2. Objectives

- The people who are specially challenged such as the deaf and dumb communicate with others using sign language and those people need a translator to help others understand what they are trying to tell.
- But not everyone has that capacity to employ a translator and thus they may not communicate with others easily.
- To implement MediaPipe based sign language to speech conversion.
- To interface it with raspberry pi to convert signs to speech.
- When the sign is shown, the webcam captures the sign and detection of sign is done.
- After detection, the output is heard through the speaker.
- A feature is added for the purpose of security, where by showing help sign the guardian of the specially abled gets the message through telegram.

3. Literature Review

A paper published [1] provides a comprehensive review of sign language recognition. In this paper MediaPipe library provided by google is used. It is a well-trained model to achieve high performance. Gestures of a hand can be determined using MediaPipe library using different technologies. Less computing power and the adaptability to smart devices makes the model robust and cost-effective. Training and testing with various sign language datasets show this framework can be adapted effectively for any regional sign language dataset and maximum accuracy can be obtained.

Another paper [2] covers about a real-time sign language

*Corresponding author: amareshangadi7179@gmail.com

recognition using MediaPipe. This paper gives knowledge about how MediaPipe library works. In this paper they have collected dataset consisting of 900 samples for 10 signs, for menu selection.

In paper [3], Raspberry Pi application for real-time motion gesture recognition using webcam input in Python is implemented. Future implementations or features which can be added is given in this paper.

In paper [4], a real-time vision-based system is proposed to monitor objects (hand fingers). It is based on the Raspberry Pi with camera module and programmed with Python programming language supported by OpenCV library. The real-time vision-based system is implemented efficiently using python, OpenCV library, Raspberry Pi, camera and Linux based LCD.

The advantages and disadvantages of MediaPipe library can be summarized as follows:

- It gives high percentage of accuracy compared to other algorithms (example: Yolo).
- Open-source hand gesture tracking pipeline that is media pipe is a platform independent works on variety of platforms like Android, iOS, Web and desktop PC's.
- Although sign language modeling using image processing techniques has evolved over the past few years, the methods are complex and require high computational power.
- The time required to train the model is also high to training and testing.

4. Methodology

Fig. 1 depicts the different stages of our project that is data collection, pre-processing, training and testing, hardware implementation and sign recognition. Firstly, it is implemented in software using programming language python, algorithm media pipe and SVM. Then connection is made to the hardware. Components required are Raspberry Pi, speaker, pi camera, webcam and SD card.

Stage 1: Data Collection

In the project we are using Indian Sign Language (ISL). First step is to get an idea on what are the signs used in ISL. From [5] we selected some of the signs and started off with the data collection. We wrote python code for collection of data. We collected around 100 images for each sign stored them in a folder named datasets. Signs can be of one hand or both the hands, hence we have created different folders for single hand and both the hands.

Stage 2: Pre-processing

Python code or pre-processing is written, where we take the images collected in the previous stage as input and detect the hand present in the image. Fig 3 gives output for the code written. Then we crop the image, in such a way that only hand is visible. Then we apply media pipe algorithm on the cropped image and get the landmarks for the hands and store the landmarks as csv files. CSV is a text file format where comma-separated values store the whole data accordingly. Therefore, CSV data can be easily opened in various text editors like Notepad, excel and can be analyzed for fetching and mining the needed details.[1]

In this project, we get 63 columns of data for 1 hand and 189 columns of data for both hands as shown in Fig 4. In csv file we take into account the data in 3D plane hence we get so many columns of data.

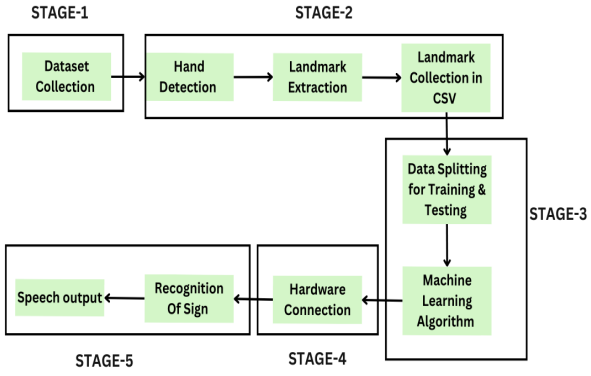

Fig. 1. Block diagram

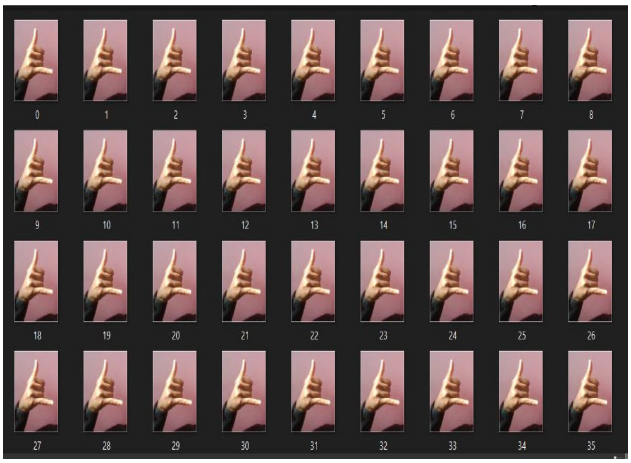

Fig. 2. Dataset collected for call

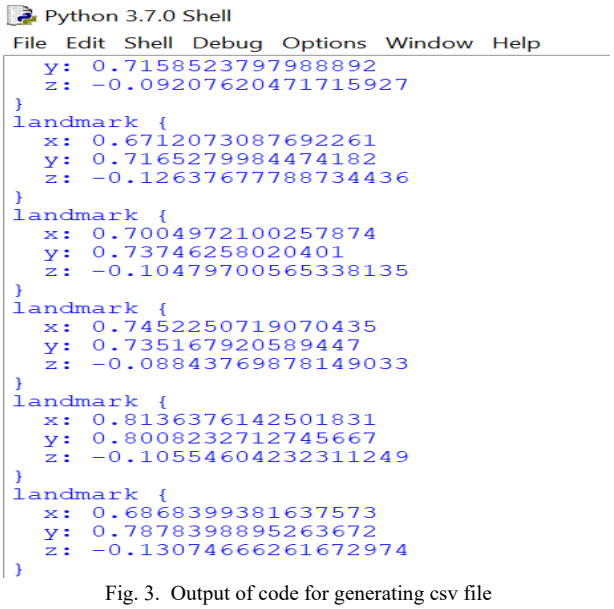

Fig. 3. Output of code for generating csv file

	AS	AT	AU	AV	AW	AX	AY	AZ	BA	BB
1	-0.20653	0.47649	0.6133	-0.17254	0.45821	0.7081	-0.1161	0.62746	0.61577	-0.10815
2	-0.20239	0.47443	0.61494	-0.16736	0.45871	0.70925	-0.11117	0.62689	0.61712	-0.10699
3	-0.20149	0.47508	0.62064	-0.16862	0.45665	0.71362	-0.11274	0.62127	0.61838	-0.1047
4	-0.20054	0.47224	0.60106	-0.16644	0.45653	0.6957	-0.11156	0.62057	0.59992	-0.10572
5	-0.19617	0.47231	0.6013	-0.16016	0.45942	0.69659	-0.10405	0.62067	0.60252	-0.10369
6	-0.19659	0.47188	0.60125	-0.1624	0.4572	0.69532	-0.10769	0.62002	0.60084	-0.10419
7	-0.19813	0.4702	0.60411	-0.16203	0.45691	0.6998	-0.10539	0.61925	0.59961	-0.10138
8	-0.19825	0.47228	0.60462	-0.16247	0.45675	0.69988	-0.10631	0.61816	0.60529	-0.10215
9	-0.19825	0.47228	0.60462	-0.16247	0.45675	0.69988	-0.10631	0.61816	0.60529	-0.10215
10	-0.19738	0.47036	0.60395	-0.16336	0.45544	0.69855	-0.10807	0.61708	0.60189	-0.10305
11	-0.20226	0.46943	0.60776	-0.1674	0.4529	0.70276	-0.11117	0.61632	0.60328	-0.10327
12	-0.19874	0.4732	0.60893	-0.16422	0.45575	0.70356	-0.10803	0.61164	0.60503	-0.1003
13	-0.20384	0.46747	0.60584	-0.16984	0.45227	0.70002	-0.11414	0.61875	0.60822	-0.10881
14	-0.19841	0.47317	0.62121	-0.16657	0.45632	0.71587	-0.11202	0.62135	0.61854	-0.10079
15	-0.20245	0.46805	0.61095	-0.16651	0.45423	0.7056	-0.10948	0.61666	0.60684	-0.10534
16	-0.19609	0.46769	0.60797	-0.16383	0.45051	0.70278	-0.10968	0.61882	0.60819	-0.10231
17	-0.19719	0.46905	0.6116	-0.16412	0.45222	0.70536	-0.10885	0.61547	0.60463	-0.09998
18	-0.1995	0.47008	0.60906	-0.16598	0.45574	0.70376	-0.11047	0.61632	0.60781	-0.10323

Fig. 4. csv file generated

```
[ ] y_pred = svm.predict(x_test)
    y_pred

    array(['house', 'house', 'strong', 'house', 'strong', 'play', 'play',
           'owl', 'house', 'strong', 'strong', 'house', 'play', 'owl',
           'house', 'owl', 'owl', 'strong', 'play', 'play', 'play', 'strong',
           'house', 'owl', 'owl', 'house', 'house', 'house', 'house', 'play',
           'house', 'house', 'house', 'owl', 'house', 'owl', 'play', 'play',
           'house', 'house', 'strong', 'play', 'play', 'strong', 'strong',
           'owl', 'owl', 'play', 'play', 'owl', 'owl', 'strong', 'strong',
           'house', 'house', 'play', 'owl', 'strong', 'house', 'house',
           'strong', 'strong', 'house', 'play', 'owl', 'owl', 'owl', 'owl',
           'play', 'house'], dtype=object)

[ ] cf_matrix = confusion_matrix(y_test, y_pred)
    f1 = f1_score(y_test, y_pred, average='micro')
    recall = recall_score(y_test, y_pred, average='micro')
    precision = precision_score(y_test, y_pred, average='micro')
    f1, recall, precision

    (1.0, 1.0, 1.0)
```

Fig. 5. Training output got for two hand datasets

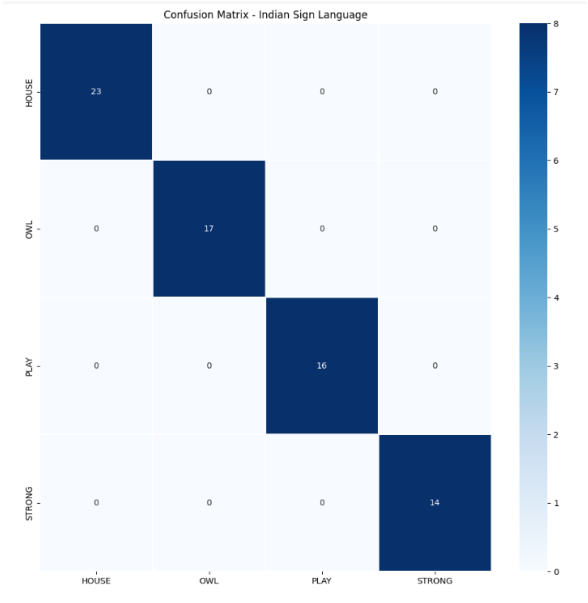

Fig. 6. Confusion matrix generated for two hand datasets

Stage 3: Training and Testing

Python code for training is written where we take csv file generated in previous stage as input and divide them into training and testing data. Using the SVM algorithm we train and test the model shown in Fig. 5. Fig. 5 shows heatmap (Confusion matrix where we get to know false predictions and true predictions) generated. In the end we generate a pickle file. Pickles allow for flexibility when deserializing objects. We can easily save different variables into a Pickle file and load them back in a different Python session, recovering our data exactly the way it was without having to edit our code [2].

Stage 4: Hardware connection

After training, we write a python code for recognition of signs which is loaded to the Raspberry pi. The input is the pickle file which is generated. Once the code is loaded, we make required connection to the raspberry pi, pi camera and speaker.

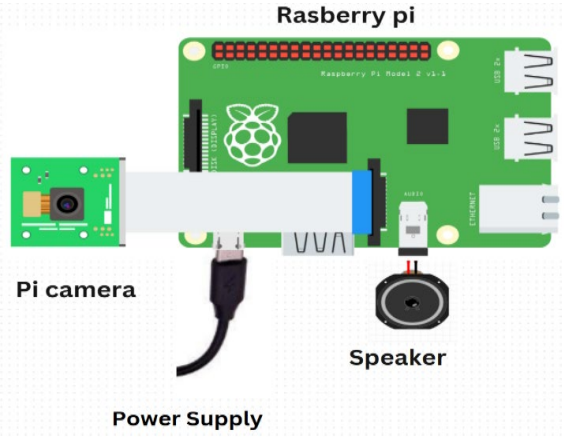

Fig. 7. Hardware connection

Stage 5: Sign Recognition

After all the connections are made, we can show the signs through the pi camera and the output is heard through the speaker. We have one emergency symbol present, when the symbol is shown, the message is sent to the guardian of the impaired person along with the image. The message is gone through telegram.

5. Proposed Model Requirement

1) Software requirement

- Operating system: Windows 10 and later version
- Coding Language: python
- Tools: Python IDLE, Google collab
- Algorithm: MediaPipe, SVM

2) Hardware Requirement

- Raspberry PI
- Raspberry PI Camera or Web Camera
- Speaker
- Battery & SD card

6. Result and Discussion

We trained the model and started off with the recognition of signs. Some of the results are shown in the below figures.

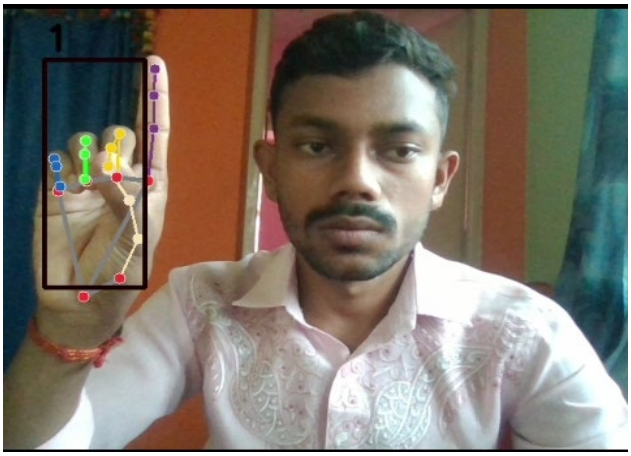

Fig. 8. Input for sign 1

In Fig. 8, Input for sign 1 is given and the output is detected and output is got through speaker.

Fig. 9. Input for sign “Hello”

In Fig. 9, Input for sign hello is given and the output is got through the speaker.

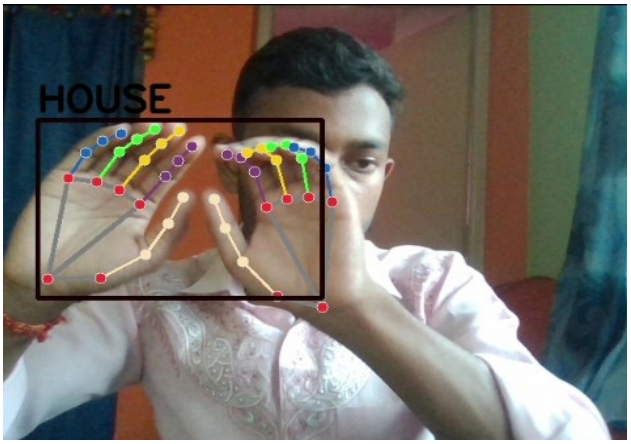

Fig. 10. Output for sign “House”

In Fig 10 input for House is given and the output is got through the speaker.

Fig. 11. Hardware interface

Fig. 11 shows the hardware interface of the project.

Output:

Fig. 12. Output got after assembling hardware

In Fig. 12, we have got the output for sign one after interfacing with Raspberry Pi.

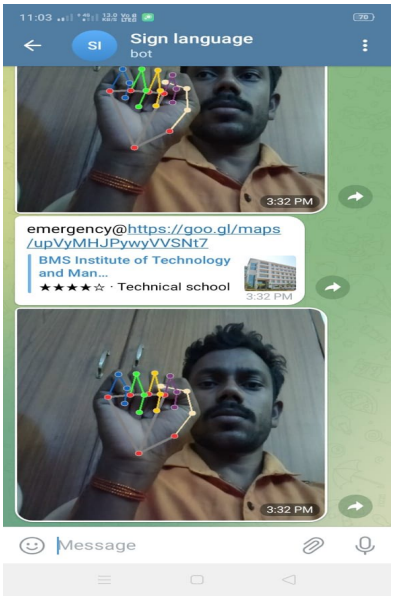

Fig. 13. Message received through telegram

In Fig. 13, for sign Help the message along with the picture is sent to the guardian's phone through telegram.

7. Future Scope

- Application that is both mobile and web-based can be developed.
- Image processing should be enhanced so that the system can communicate in both directions, i.e., transform conventional language to sign language and vice versa.
- Recognize motion-related indicators.

8. Conclusion

This paper gives one of the best methods for hand gesture recognition using media pipe. This will be responsible to create meaning in the lives of disabled people. With an average accuracy of 99% on most sign language dataset using Media Pipe and machine learning, our proposed methodology shows that MediaPipe can be effectively used as a tool to accurately detect complex hand gestures. With less computing power and adaptability to smart devices, the model is robust and cost-effective. Training and testing with different sign language datasets show that this framework can be efficiently adapted for any regional sign language dataset and maximum accuracy can be achieved. Faster real-time detection demonstrates the effectiveness of the model better than the current state of the art. In the future, the work can be extended to introduce word detection in sign language from videos using Media pipe and various algorithms.

References

[1] K. M. Kavana and N. R. Suma, "Recognization of Hand Gestures Using MediaPipe Hands," in *International Research Journal of Modernization in Engineering Technology and Science*, vol. 4, no. 6, pp. 4149-4156, June 2022.

[2] Indriani, M. Harris, and A. S, Agoes, "Applying Hand Gesture Recognition for User Guide Application Using MediaPipe," in *Proceedings of the 2nd International Seminar of Science and Applied Technology (ISSAT 2021)*, pp. 101-108, 2021.

[3] S. Pawar, A. Bamgude, S. Kamthe, A. Patil, and R. Barapte, "Gesture Language Translator Using Raspberry Pi," in *International Journal for Research in Applied Science and Engineering Technology*, vol. 10, no. 5, pp. 566-570, May 2022.

[4] V. Saraswathi, S. M. Ahmad, G. G. Krishna, and A. Khan, "Raspberry Pi for Hand Gesture Recognition," in *Juni Khyat*, vol. 12, no. 1, pp. 814-818, 2022.

[5] https://indiansignlanguage.org/