

**Project Report**

Project Title

**Sign Language Translator for Speech-impaired**

Submitted by:

**Noor-ul-Emaan (2020-CE-33)**

**Hareem Arshad (2020-CE-34)**

**Hira Sohail (2020-CE-41)**

**Palwasha Ali (2020-CE-48)**

Submitted to:

**Sir Raja Muzammil**

Course:

**CMPE-341L Artificial Intelligence**

Semester: **Fall 2022**

Date:

**December 19, 2022**

**Department of Computer Engineering**

**University of Engineering and Technology, Lahore**

**Table of Content**

**Abstract………………………………………………………………………….. 3**

**Introduction……………………………………………………………………… 3**

**Flowchart………………………………………………………………………… 4**

**Libraries Used………………………………………………………………….... 5**

**Datasets……………….………………………………………………………….. 5**

**ML Algorithms Used……………………………………………………………. 6**

**Model training…………………………………………………………………… 7**

**Features…………………………………………………………………………... 7**

**GUI……………………………………………………………………………… 12**

**Application Of project…………………………………………………………. 13**

**Difficulties Faced……………………………………………………………….. 13**

**Limitations……..……………………………………………………………….. 13**

**Conclusion………………………………………………………………………. 13**

**References………………………………………………………………………. 14**

**Abstract:**

There has always been a major problem of communication between special people and normal people. Sign Language Converter is a program developed to detect the hand signs and then convert them into short English sentences as well as it can convert the text into sound which also helps the user to understand the text in a much better way. The main purpose or the background of this program is to help deaf and dumb people. It can help them to understand the text written on the screen just by making different hand gestures. By using different hand conditions a whole language for those people can be developed and serve humanity.

**Introduction:**

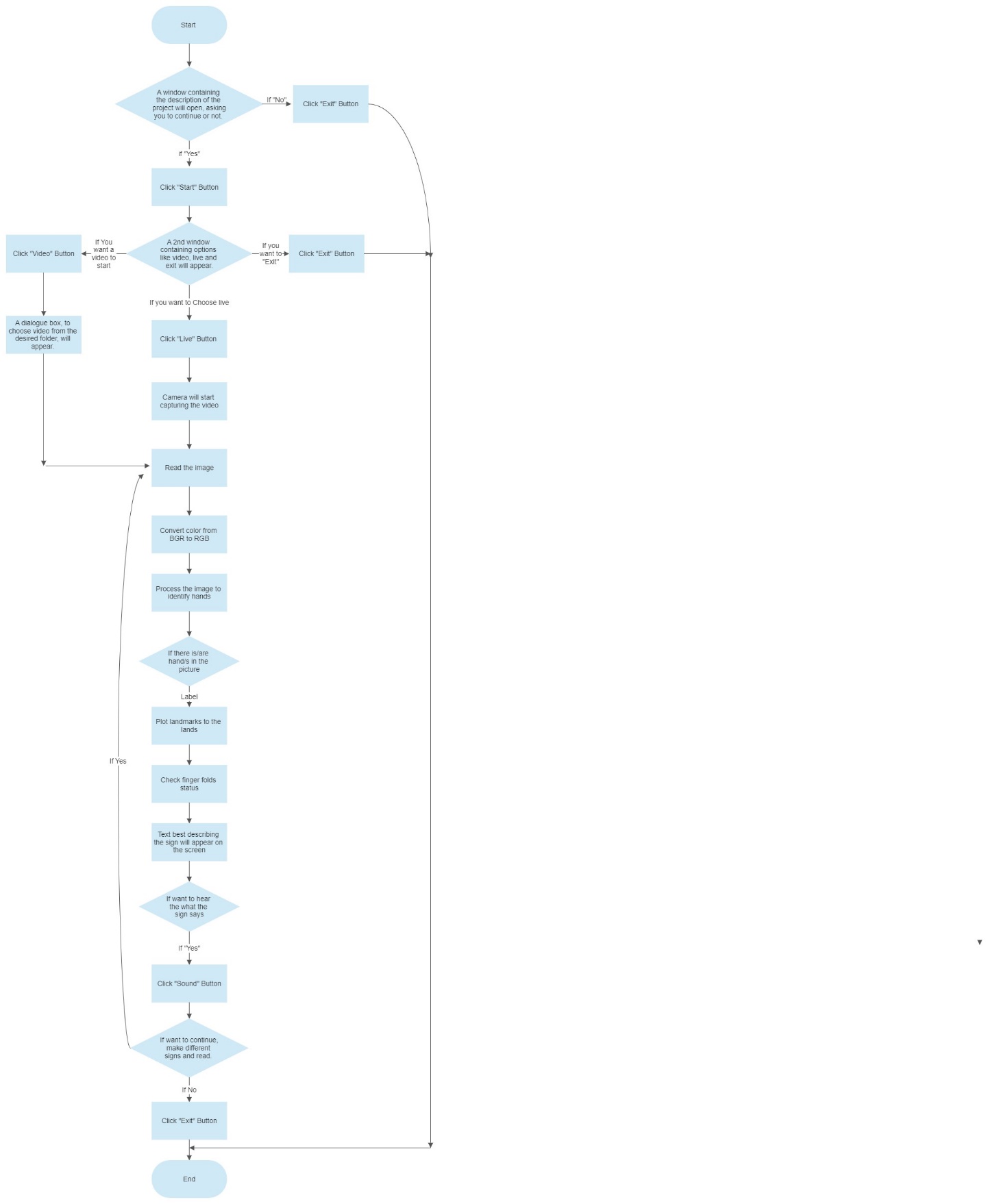
Sign language translation (SLT) is an important application to bridge the communication gap between deaf and hearing people. Normal people will find it difficult to communicate with hearing-impaired people. To break the barrier of communication between normal and hearing-impaired people, the Speech to Sign Language translator is used. This translator makes the interaction simpler and faster for normal people to convey their ideas to hearing impaired people. The translator converts speech or text to Sign Language using Natural Language Processing algorithms. Thus, the system is used to overcome the hurdles faced by normal people to share their thoughts with hearing impaired people and it will be an ear for the hearing-impaired.

Gesture is a symbol of physical behavior or emotional expression. It includes body gesture and hand gesture. Gesture recognition determines the user intent through the recognition of the gesture or movement of the body or body parts. Hand gesture recognition has great value in many applications such as sign language recognition, augmented reality (virtual reality), sign language interpreters for the disabled, and robot control.

This is a NLP and computer vision project

In this project, we will use an efficient and effective method for hand gesture recognition. The hand region is detected through the background subtraction method. Then, the palm and fingers are split so as to recognize the fingers. After the fingers are recognized, the hand gesture can be classified through a simple rule classifier.

**Flowchart:**

****

**Libraries Used:**

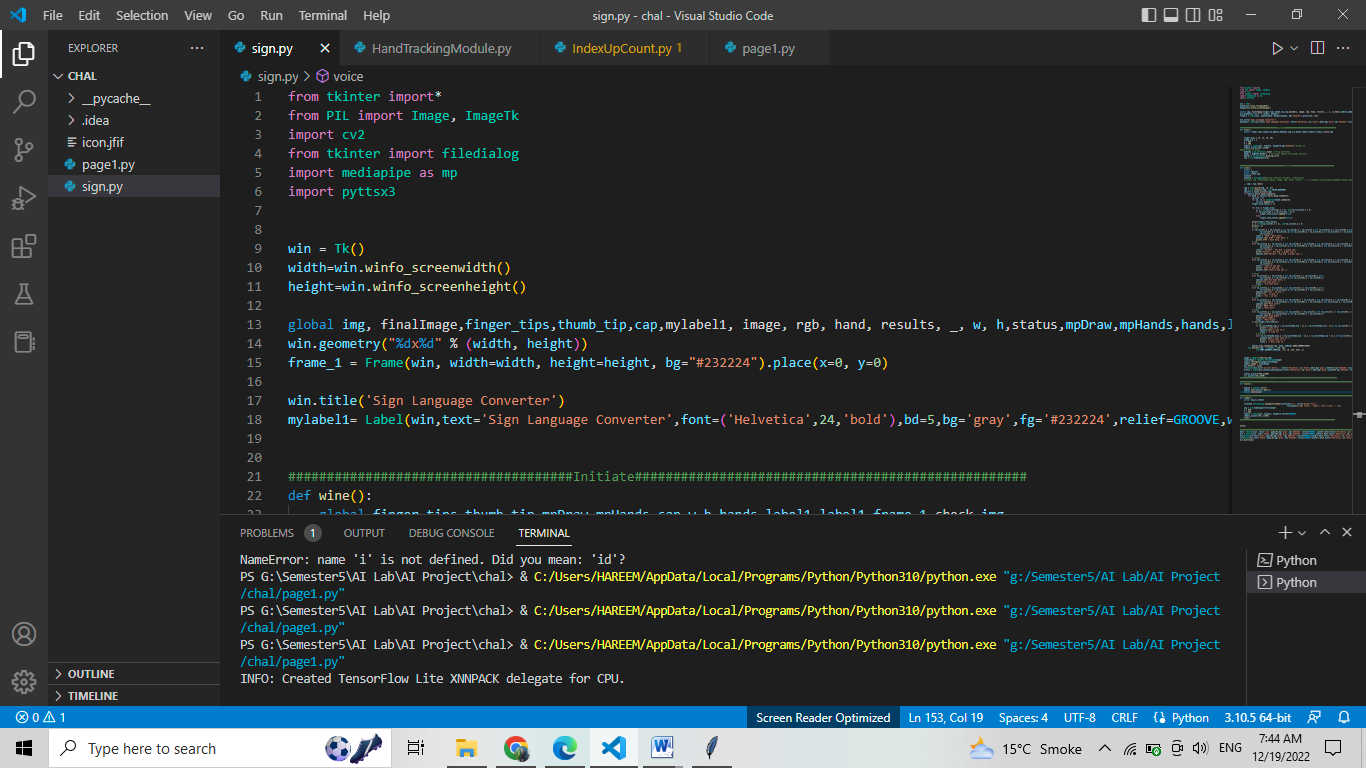
****

Figure 1

**1-OpenCV** (Open Source Computer Vision Library) is a library of programming functions mainly aimed at real-time computer vision. To explain computer vision it is simply a method from which a computer can gain knowledge from digital videos and photos. This program is mainly based on Gesture Recognition.

**2-Mediapipe** is a Framework for building machine learning pipelines for processing time-series data like video, audio, etc. This cross-platform Framework works in Desktop/Server, Android, iOS, and embedded devices like Raspberry Pi and Jetson Nano.

It generally consists of frameworks and solutions and in this program, we used the Mp Hands solution

**3-Tkinter** is the standard GUI library for Python. Python when combined with Tkinter provides a fast and easy way to create GUI applications. Tkinter provides a powerful object-oriented interface to the Tk GUI toolkit.

**Dataset:**

We will take videos to understand the position of hand and fingers to train our model as a reference from a website. Its link is given in reference heading

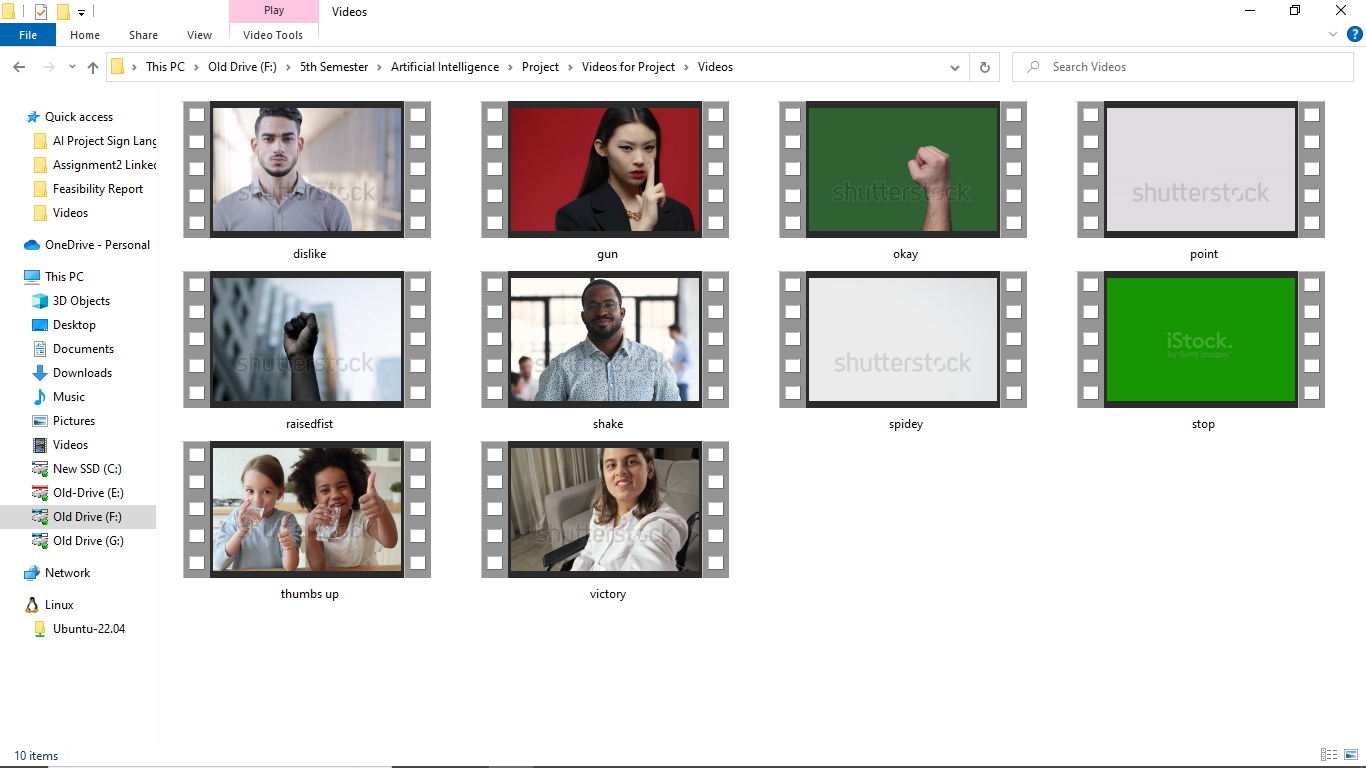


Figure 2

Suppose a person has 2 fingers mold and thumb is upward it would be a sign of gun. We will train the model with algorithm and programming by using the conditions and svm algorithm. X and y position of landmark points will be used.

**SVM Algorithm:**

SVM is a machine learning algorithm belonging to the supervised learning group. It is used in classification or regression problems. It is a binary classification algorithm. The SVM takes the input and classifies them into two different classes. SVM training algorithm builds a model to classify them into those two categories.

The idea of SVM is to find a hyperlane to separate data points. This hyperplane will divide the space into different domains and each domain will contain a type of data. For example, we have a dataset of blue and red points placed on the same plane.

We will define the conditions of x y points. We will at least train the model with 30 signs i-e 30 conditions.

**ML Algorithms Used:**

**MP HANDS solution:**

Media Pipe Hands is a high-fidelity hand and finger tracking solution. It employs machine learning (ML) to infer 21 3D landmarks of a hand from just a single frame. Whereas current state-of-the-art approaches rely primarily on powerful desktop environments for inference, our method achieves real-time performance on a mobile phone, and even scales to multiple hands. We hope that providing this hand perception functionality to the wider research and development community will result in an emergence of creative use cases, stimulating new applications and new research avenues.

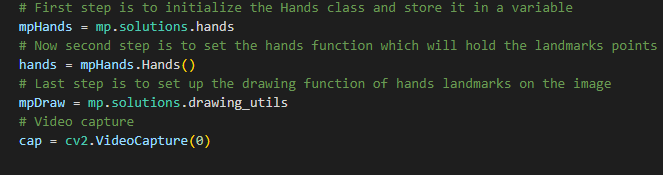


Figure 3

**Model training:**

We will develop the program by putting it into different parts. In the first part, we will use OpenCV and media pipe to detect the movement of the hand and display it on the screen using the webcam of the computer.

We will made functions for the better understanding of code and this function will help to convert the picture into the respective RGB color and will implement all the conditions for hand gestures which the program will show on the output window.

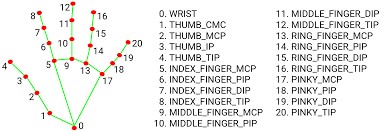


Figure 4

It will show 21 different landmarks point plotted on hands and with their help, track of the movement of the hand will be quickly recorded. On any finger, there are two possibilities either a point is greater or smaller in the respective x and y planes. If we take one example of the index finger as you can see **lm\_list[8].y < lm\_list[6].y**in the snippet it means that the point 6 is greater than the point 8 which means the finger is open state and there is no fold in it but if we change the condition to**lm\_list[8].y >lm\_list[6].y**which means now point number 8 is greater if the finger is closed or in the folded state. Like this, we can write the conditions for all of the fingers and implement all the conditions. Here lm\_list represent landmark list.

**Features:**

# Live Feature:

This is a live feature where webcam will open to detect the person hand.

In this Feature, we made functions for a better understanding of code. This function will help to convert the picture into the respective RGB color and will implement all the conditions for hand gestures which the program will show on the output window.

This condition will be called when all the five fingers of the hand will be opened which means an open palm position the system will detect the condition as you can see in the hand landmark picture shown above the fourth and second point represents the thumb, eight and six the index finger, twelve and ten the middle finger, sixteen and fourteen the ring finger, twenty and eighteen shows the pinky finger, zero shows the wrist position whereas 5 and 17 show the base points of index and pinky finger respectively.

In this program, we developed around eight different conditions, but it can be increased to the requirement of the programmer.

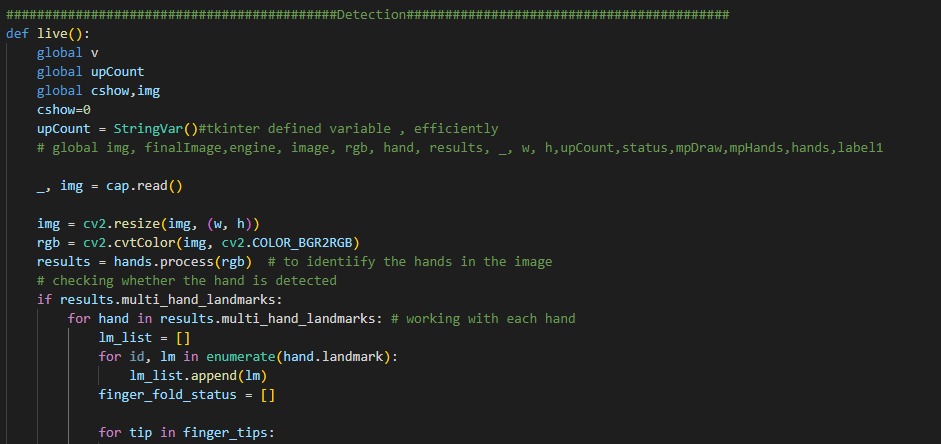


Figure 5

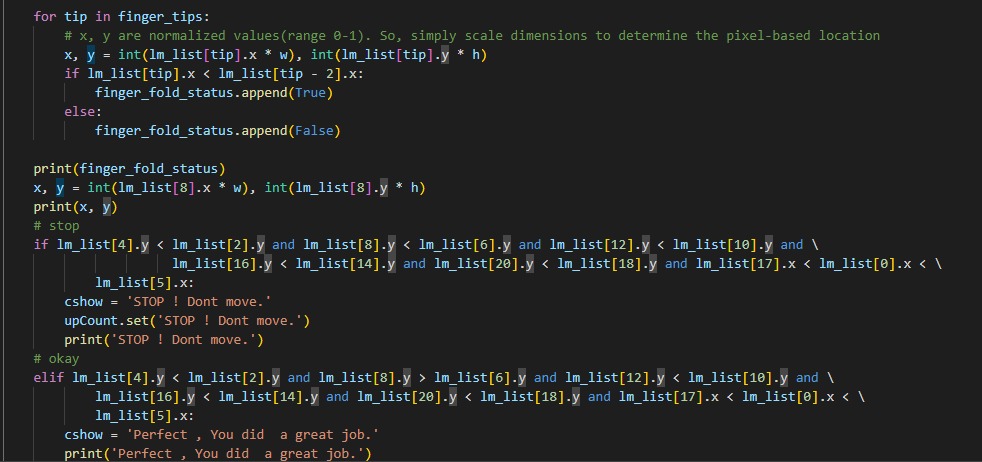


Figure 6

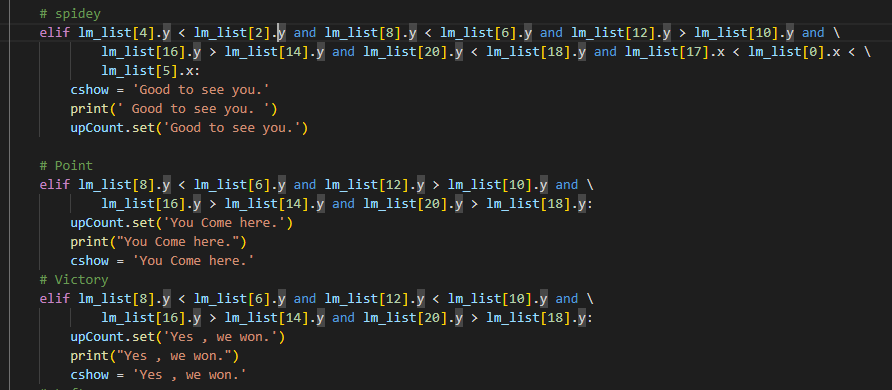


Figure 7

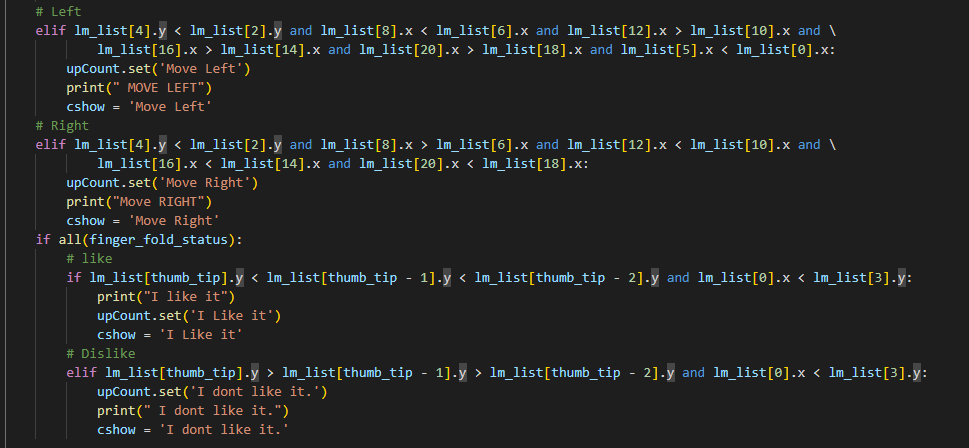


Figure 8



Figure 9

# Video Feature:

In this program, we will add a feature for user convenience that they can switch between live cameras or run any video from the computer hence all the logic of the program will be implemented on that video. This part of the code will ask the user to select any video and after it, the **LIVE** function i.e. to display landmarks and detect the state of hand in the video will be called.

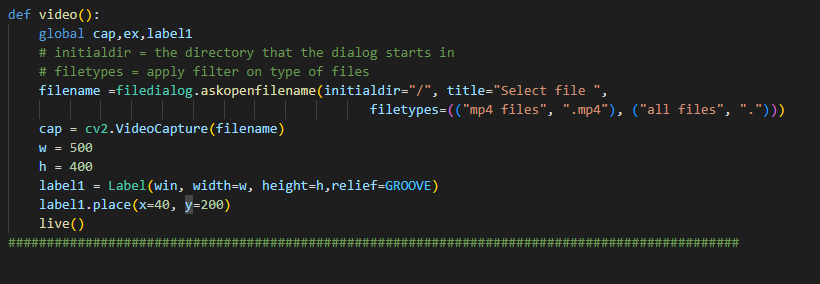


Figure 10

* **Sound Feature:**

This function is used to convert the given text generated by the signs we give them to sound. It uses pyttsx3 library to convert text to speech. It uses different speech engines based on your operating system.

The pyttsx3 module supports two voices first is female and the second is male which is provided by “sapi5” for windows.

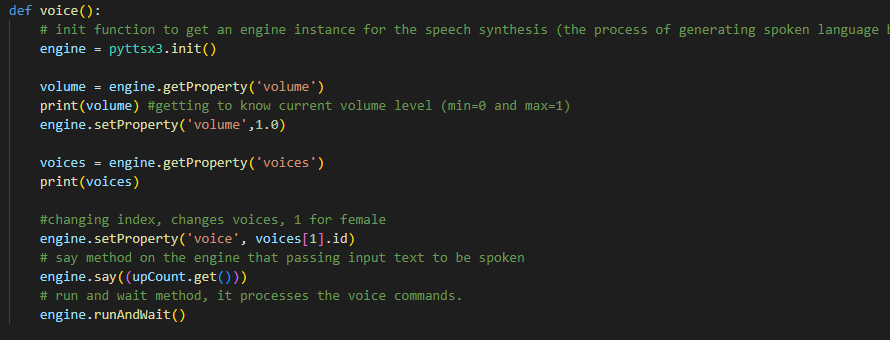


Figure 11

**GUI:**

To make this whole program user friendly and to give a new dimension to it. We will implement GUI in the program, including following features:

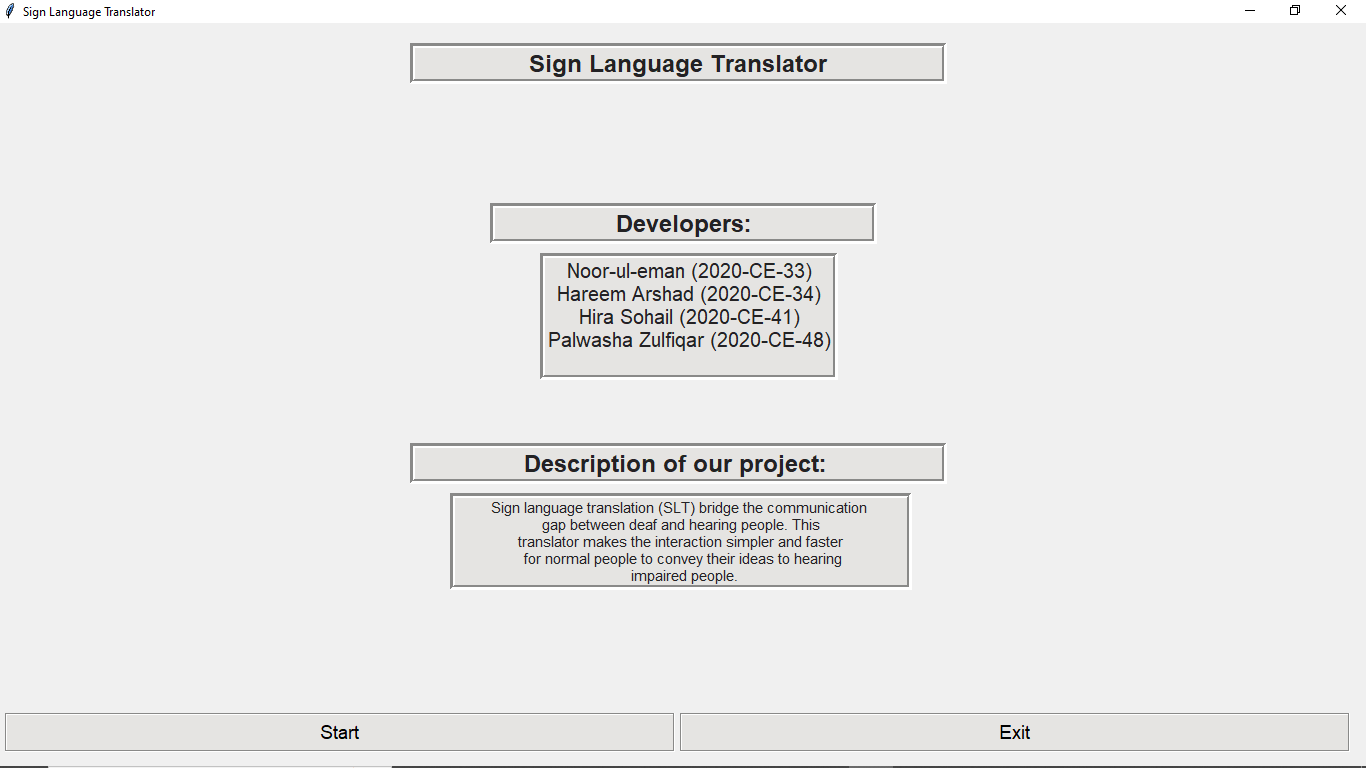
  
Figure 12



Figure 12

**Application Of project:**

* The translator will make interaction simpler and faster for normal people to convey ideas to deaf people.
* The main application of this project is to provide aid for the speech-impaired
* Due to the simplicity of the model, it can also be implemented in smartphones and is regarded as our future plan to do so.
* It will be helpful for blind people as they can’t read the text so they can hear the text converted from text.

**Difficulties Faced:**

* As time was less, we faced difficulty in training model and with datasets.
* As we access landmarks with coordinates so we faced difficulty in defining conditions.

**Limitation:**

The whole machine learning process has some limitations to it like poor quality of data, lack of training data, slow implementation, etc. So, we can say that the output of the program is not 100% in every case there come some lapses in different parts which at the current time can not be handled but in the future, we hope that things will become better.

**Conclusion:**

We have successfully developed a sign language detection project. This is an interesting machine learning python project to gain expertise. This can be further extended to detecting the English alphabet.

**References**

[1]. https://mediapipe.dev/index.html

[2]. https://google.github.io/mediapipe/solutions/hands

[3]. https://docs.opencv.org/4.x/index.html

[4]. https://www.youtube.com/watch?v=01sAkU\_NvOY

[5]. <https://www.youtube.com/c/MurtazasWorkshopRoboticsandA>

[6]. <https://www.scienceofpeople.com/hand-gestures/>