SIGN LANGUAGE TO TEXT CONVERSION

FOR DUMB AND DEAF

A PROJECT REPORT

Submitted by

SIDDHARTHA SHARMA (20BAI10044)

SHIVAM SHARMA (20BAI10058)

AVRODEEP SAHA (20BAI10041)

MANSI SAHU (20BAI10158)

in partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

in

PROGRAM OF STUDY

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING

VIT BHOPAL UNIVERSITY

KOTHRIKALAN, SEHORE

MADHYA PRADESH - 466114

DEC 2021

VIT BHOPAL UNIVERSITY, KOTHRIKALAN, SEHORE

MADHYA PRADESH – 466114

BONAFIDE CERTIFICATE

Certified that this project report titled "**ISHARA - VOCALIZING THE SILENCE**" is the bonafide work of "**(SIDDHARTHA SHARMA: 20BAI10044), (SHIVAM SHARMA: 20BAI10058), (AVRODEEP SAHA: 20BAI10041), (MANSI SAHU: 20BAI10158)**" who carried out the project work under my supervision. Certified further that to the best of my knowledge the work reported at this time does not form part of any other project/research work based on which a degree or award was conferred on an earlier occasion on this or any other candidate.

PROGRAM CHAIR	**PROJECT GUIDE**
Dr. S Sountharrajan, Department Head.	Dr. Nilamadhab Mishra, Faculty
School of Computer Science and Engineering.	School of Computer Science and Engineering
VIT BHOPAL UNIVERSITY.	VIT BHOPAL UNIVERSITY

The Project Exhibition I Examination is held on 18/12/21.

ACKNOWLEDGEMENT

First and foremost, I would like to thank the Lord Almighty for His presence and immense blessings throughout the project work.

I wish to express my heartfelt gratitude to **Dr. S Sountharrajan**, Head of the Department, School of Computer Science and Engineering for much of his valuable support encouragement in carrying out this work.

I would like to thank my internal guide **Mr. Nilamadhab Mishra**, for continually guiding and actively participating in my project, giving valuable suggestions to complete the project work.

I would like to thank all the technical and teaching staff of the School of Computer Science and Engineering, who extended directly or indirectly all support.

Last, but not least, I am deeply indebted to my parents who have been the greatest support while I worked day and night for the project to make it a success.

TABLE OF CONTENTS

LIST OF ABBREVIATIONS

LIST OF FIGURES AND GRAPHS

LIST OF TABLES

ABSTRACT

- Purpose

The purpose of our project is to help deaf and mute people to communicate with others by the means of our project as their sign language will change into English letters on the screen and hence, they will be able to get their message across which was earlier not possible since normal people are not aware of sign language.

- Methodology

Our code makes use of distance between various finger points in order to understand which letter is actually used and this helps us to create the word using autocorrect library. A detailed explanation will be in chapter 4.

- Findings

We found many projects on the internet with the same target but they were restricted to a limited number of gestures like hello, bye or well done. However, using our project, you can make any word you like. The second thing which was common with the projects available on the net was that they used Machine learning, but we have not used machine learning rather we have used artificial intelligence in the form of open CV.

CHAPTER-1:

PROJECT DESCRIPTION AND OUTLINE

1.1 Introduction

Communication is the process of exchange of thoughts and messages in various ways such as speech, signals, behavior and visuals. The easiest and prominent way for communication of these people is by performing certain actions by hands to express different gestures, to express their ideas with other people. This nonverbal communication of deaf and mute people is called sign language.

1.2 Motivation for the work

The domain of Sign Language that is vastly on the opposite end of original text creates a language barrier for interaction between normal and special people i.e., deaf & mute people. They majorly depend on vision-based communication for interaction. There is a requirement & demand for a common interface that, converts the sign language to text which enables the gestures to be easily understood by other people. Immense research has been done for a vision-based interface system so these people can enjoy communication without really having a knowledge of each other's language. The aim is to develop a user-friendly human-computer interface (HCI) where the computer understands the human sign language. There are various sign languages all over the world, namely American Sign Language (ASL), French Sign Language, British Sign Language (BSL), Indian Sign language, Japanese Sign Language and work has been done on other languages all around the world.

1.3 About Introduction to the project including techniques

Our project aims at bringing equality to our society. In the contemporary time although we say that we are equal but the truth is that we cannot even communicate with deaf and mute people because we don't know sign language. However now we can end this discrimination as our application can convert hand gesture to words.

Our project is based on the distances between the various landmarks the landmarks encompasses the 4 point of all the fingers and thumb. The distance calculated between these landmarks during the gesture helps us to identify the letter. We have used open CV and created a hand module which was imported into the main project in order to mark the landmarks.

1.5 Problem Statement

The easiest and prominent way for communication of deaf & mute people is by performing certain actions by hand to express different gestures, to express their ideas with other people i.e., sign language. However, due to the lack of awareness of sign language, these people have hard times communicating. They still fave various psychological, educational, employment, and social problems. With this project, we aim to give a right to speak to those who cannot speak.

1.6 Objective of the work

The main objective is to translate sign language to text. The framework provides a helping-hand for speech-impaired to communicate with the rest of the world using sign language. This leads to the elimination of the middle person who generally acts as a medium of translation. This would contain a user-friendly environment for the user by providing text output for a sign gesture input.

1.7 Organization of the project

Our project makes use of a list of site packages which include mediapipe, OpenCV, NumPy, Math, Autocorrect, Translate and Time. Theses site packages help in providing the required backup we require for the main application. The next file is Hand tracking module of helps in detecting the hand and it's movement we import this into the main application. All these are kept in main folder. The main folder has Lib inside which there is a folder named site-packages and finally we have all the site packages there like NumPy OpenCV etc.

1.8 Summary

So, we can summarize that Sign Language is the most natural and expressive way for hearing impaired people. People, who are not deaf, never try to learn sign language for interacting with deaf people. This leads to the isolation of deaf people. But if the computer can be programmed in such a way that it can translate sign language to text format, the difference between normal people and the deaf community can be minimized. Indian sign language (ISL) uses both hands to represent each alphabet and gesture. ISL alphabets are derived from British Sign Language (BSL) and French Sign Language (FSL). Around 63 million people in India suffer from either complete or partial deafness, and of these, at least 50 lakh are children. So, we realize that there is a dire need to innovate this sign language to text converter.

CHAPTER-2:

RELATED WORK INVESTIGATION

2.1 Introduction

A World Health Organization report says **around 63 million people** in India suffer from either complete or partial deafness, and of these, at least 50 lakh are children.

One barrier standing before the country's deaf population is the threadbare infrastructure of educational facilities such as specialized schools. Almost 90% of schools lack modern facilities and instrumentation. There is a huge student to school ratio. Only 388 such schools exist in our country which is really low.

Approximately 1 out of 20 people in India are deaf, which results in a huge communication barrier for deaf and mute people. The use of sign language which is one of the most popular ways to remove the communication barrier has its own set of ordeals.

2.2 Core area of the project

Deaf and Dumb people rely on sign language interpreters for communications. A Sign Language Recognition system is designed and implemented to recognize 26 gestures by hand gesture recognition system for text generation. The signs are captured by using web cam. It is based on the distances between the various landmarks. The landmarks encompasses the 4 point of all the fingers and thumb. The distance calculated between these landmarks during the gesture helps us to identify the letter. Finally, recognized gesture is converted into text. This system provides an opportunity for a deaf-dumb people to communicate with non-signing people without the need of an interpreter.

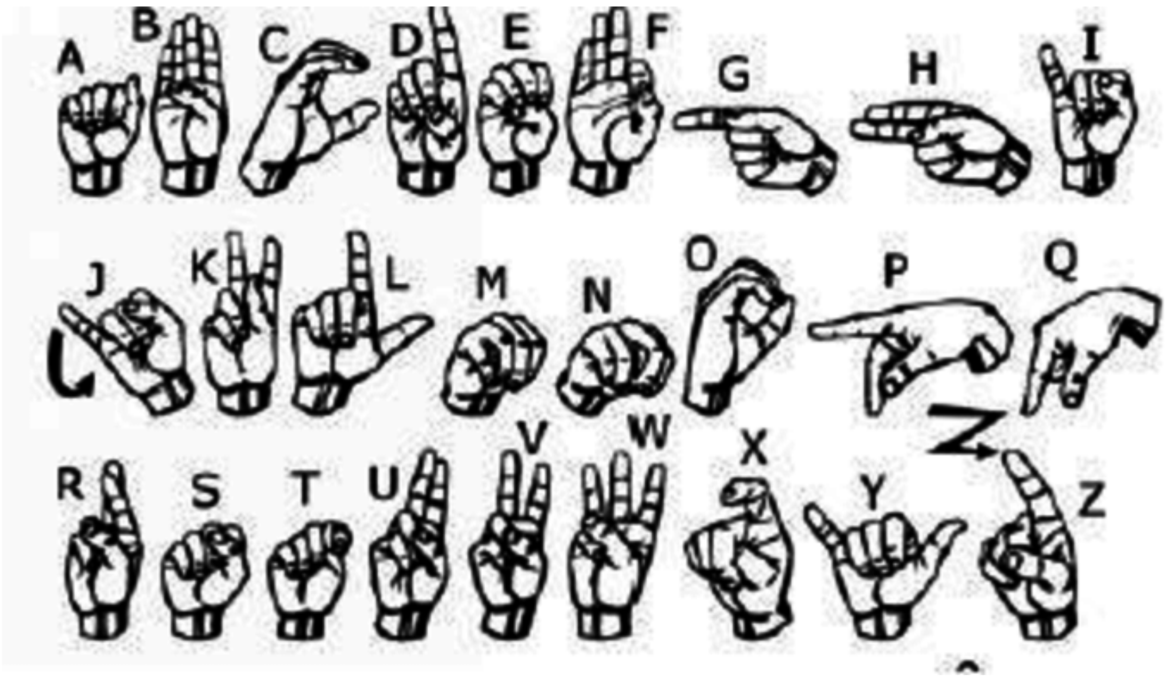


Figure 1: Hand Gestures

2.3 Existing Approaches/Methods

A various hand gestures were recognized with different methods by different researchers in which were implemented in different fields. The recognition of various hand gestures were done by various approaches: Work done so far in this field has been much more focused on underlying architecture for most of the systems are based on:

2.3.1 Approaches/Methods -1

Most of the projects and programs found commonly used CNN for the development of their translation programs. CNN or convolutional neural networks proves to be a useful method for this purpose. Where a large amount of datasets are fed into a learning algorithm. Tests are conducted to actually predict the sign languages.

2.3.2 Approaches/Methods -2

Human translation: This is one of the most basic and oldest method yet known to facilitate communication between the two communities. It requires a person who is well versed with sign language and also aware of the language to be spoken.

2.3.3 Approaches/Methods -3

Direct translation: Words from source language are directly transformed into target language words. Output may not be a desired one.

2.4 Pros and cons of stated Approaches

A major portion of the projects of Gesture translations, focuses on machine learning, using CNN or many other methods. On further research it was noticed that it doesn't only takes a larger space to store and run but also has a relatively complex GUI and usages. Also, most of them could not be accommodated in low end devices.

Human translation essentially switches the table in terms of pros and cons. A higher standard of accuracy comes at the price of longer turnaround times and higher costs. What you have to decide is whether that initial investment outweighs the potential cost of mistakes. Translators rarely work for free. Also, they are not always easily available.

2.5 Issues/observations from investigation

This project was mainly done with consideration of ease of use and affordability which was primarily our goal so that it could be used by a large number of audiences. Hence, a special care has been taken care so that the application doesn't require too much memory, or time to run and display requirements.

To avoid such problems, we tried to make our software easy to use, accommodate as well as affordable. Instead of running huge calculations behind, training huge datasets, storing the results, we targeted to find the uniqueness of each gesture and match with the input gestures. This approach has not only made the program to run fast but we could also achieve the above-mentioned targets.

2.6 Summary

The target audience for our project 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. A special care has been taken care so that the application doesn’t require too much memory, or time to run and display requirements. It has also been kept in mind during the designing that the application doesn’t require too many add on or high-tech devices to run. so that it could be used by a large number of audiences.

CHAPTER-3:

REQUIREMENT ARTIFACTS

3.1 Introduction

In this module we will be covering on how usable is our product taking a look towards the software and hardware requirements, it’s performance, security etc. We will be able to get a better idea of the outlook of our product.

3.2 Hardware and Software requirements

- **Python (3.6.6)**

Python is a programming language that lets you work quickly and integrate systems more effectively.

- **NumPy (version 1.15.3)**

NumPy brings the computational power of languages like C and Fortran to Python, a language much easier to learn and use. With this power comes simplicity: a solution in NumPy is often clear and elegant

- **Cv2 (version 81.3.4.3)**

OpenCV (Open Source Computer Vision Library) is an open source computer vision and machine learning software library. OpenCV was built to provide a common infrastructure for computer vision applications and to accelerate the use of machine perception in the commercial products. Being a BSD-licensed product, OpenCV makes it easy for businesses to utilize and modify the code.

- **Time**

Python has a module named time to handle time-related tasks.

- **Math**

Python has a built-in module that you can use for mathematical tasks. The math module has a set of methods and constants.

- **Mediapipe**

MediaPipe offers ready-to-use yet customizable Python solutions as a prebuilt Python package. MediaPipe Python package is available on PyPI for Linux, macOS and Windows.

- **Autocorrect**
- **Translate**

This is a simple, yet powerful command line translator with google translate behind it. You can also use it as a Python module in your code.

3.3 Specific Project requirements

Our program requires a webcam with a lens of greater than 8 mp and memory space >150 mb.

3.3.1 Performance and security

Our project is highly efficient and unique. Most of the projects available on the internet have used Machine learning to solve this problem and they are not dynamic in the sense that you cannot make all the words using those programs but our project is unique because we have used open cv and in our program any and every word is possible. Since our program does not use any model requiring lots of time our output is fast and useful.

3.3.2 Look and feel requirements

Our program is currently being run in a window which comes up when we run the program. The window has a separate line for all the words and the box where the people are required to make the hand gesture is clearly specified. The look and feel is simple, professional and user friendly.

3.4 Summary

Our program is based on simple but effective code. Hence it does not require high specifications and therefore is more useful than other programs trying to achieve the same. Our dynamic approach has made it possible to make it useful for the people who are blind and deaf.

CHAPTER-4:

DESIGN METHODOLOGY AND ITS NOVELTY

4.1 Methodology and goal

The design methodology of this project was mainly done with consideration of ease of use and affordability which was primarily our goal so that it could be used by a large number of audiences. Hence, a special care has been taken care so that the application doesn't require too much memory, or time to run and display requirements. It has also been kept in mind during the designing that the application doesn't require too many add on or high-tech devices to run. We also targeted to keep program GUI relatively simple and easy to understand.

Since main objective of our project was to facilitate easy communication between the deaf and dumb population with others without the need of learning to understand the sign language. Even though many researchers and project work are made or is being made on gesture translations, a major portion of the projects focuses on machine learning, using CNN or many other methods. On further research it was noticed that it doesn't only takes a larger space to store and run but also has a relatively complex GUI and usages. Also, most of them could not be accommodated in low end devices.

To avoid such problems, we tried to make our software easy to use, accommodate as well as affordable. Instead of running huge calculations behind, training huge datasets, storing the results, we targeted to find the uniqueness of each gesture and match with the input gestures. This approach has not only made the program to run fast but we could also achieve the above-mentioned targets.

4.2 Functional modules design and analysis

The Initial step was the detection of hands. Media pipe proves to be an excellent tool for tracing hands, objects, face and even body. In order to just restrict our model to gestures of a single hand, a hand-tracking-module was created and was set up for detecting only one hand even if both hands could be seen. It was further observed that media pipe had a few fluctuations while detecting hands, it was primarily due to the quality of the camera, which was causing to identify wrong gestures. We tried to adjust the detection concentration to a reasonable amount so that it could also be used in low end devices.

In order to record the gestures, OpenCV was the best and optimized module we could find and was thus used. The main functional designing procedure in this project was carried out with the simulations with the help of media-pipe module, which is basically the crux of our project. It was observed that each of the defined gestures in sign language were unique for each letter, word, number and so on. They were not only unique in the looks but also the distances between each of the fingers were unique too. Hence, we decided to focus primarily on this feature. Unique distances for each of the gesture were thus calculated using math and NumPy module and a database was setup. Initially gesture of alphabets were entered after which numbers and then words. Two major problems were encountered, the program behaved abnormally when lighting condition was not proper and gesture distances were either too far or too close. Potential lighting problems could be encountered using a lighting optimization and distance related issues with the help of the value of disparity.

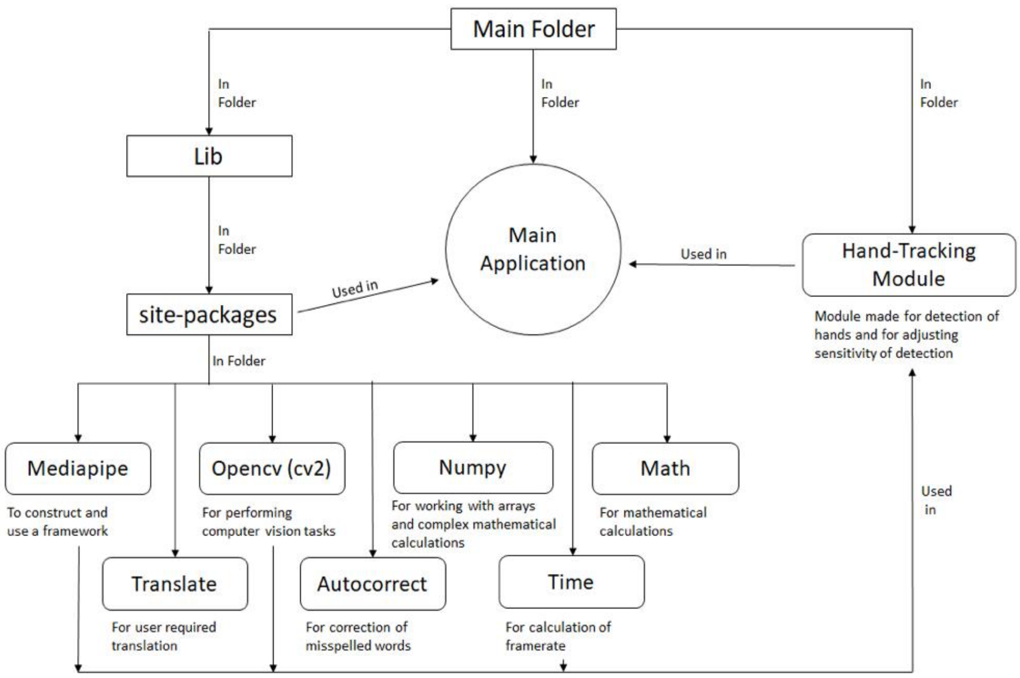


Figure 2: Module Design

4.3 Software Architectural designs

Initially when the program captures the gesture of the candidate. A one hand tracing of the gesture is done by the media pipe with the help of the hand-tracking-module integrated in the code. The program the works on the simple principal of match and display algorithm. The program keeps on capturing frames of the structure of the media pipe from the of the video of the candidate's gesture been captured. It simultaneously adjusts light and depth of the gesture. The program keeps on matching distances defined in the database stored with the distances being currently fed as the candidate keeps on showing his gestures. Once a gesture is matched with that of the database within a range, the program displays an output. We further tried to display a proper output if words are misspelled or misinterpreted by using dictionary module. The candidate showing gesture has to hold the last letter of a word, or hold a gesture to permanent the word after which he can keep on adding words for the gesture in the same process. We have also tried to give a translation option to the user so that he can read the text in any preferable language.

It was also seen that many developers tried to use cloud-based services in order to store the huge databases, not only the size becomes dynamic but also costly, but the approach taken by us has a fixed size which can either be operated on cloud with a minimal space or in the device being used without the need of connecting to the server as well. In all the project concept had the potential to satisfy the needs of a large group of people and can be extended for devices from low to high end thus facilitating a large number of users with a minimal cost.

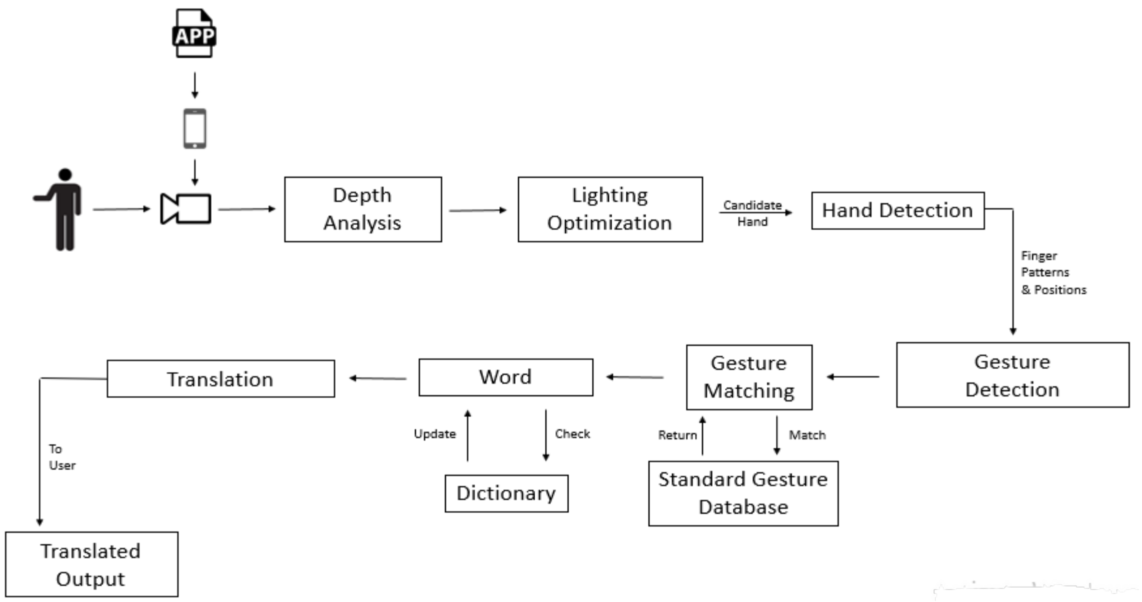


Figure 3: Software Design

4.4 Subsystem services

For making the entire program various subsystem functions were used, of which the most important was the hand tracking module. This module had to be exclusively made to make media-pipe detect hands only. Also, in the prototype model we decided to start with the one hand gestures, so we particularly dedicated the module to detect only one of the hands while being captured. The detection concentration had to also be adjusted accordingly to cope with different conditions and systems. Few changes to the default values of the media-pipe module were given out as an option to adjust in the places where and whenever needed. After making changes the program was converted into a generalized module and then it was imported in the main application’s program.

4.5 User Interface designs

The user interface design of the program was targeted to keep as simple as possible. As soon as the program executes, the device camera launches and searches for any detectable hands that is being recorded. As soon as a hand is detected, the user observes a colored structure tracing the hands with red dots at each joint.

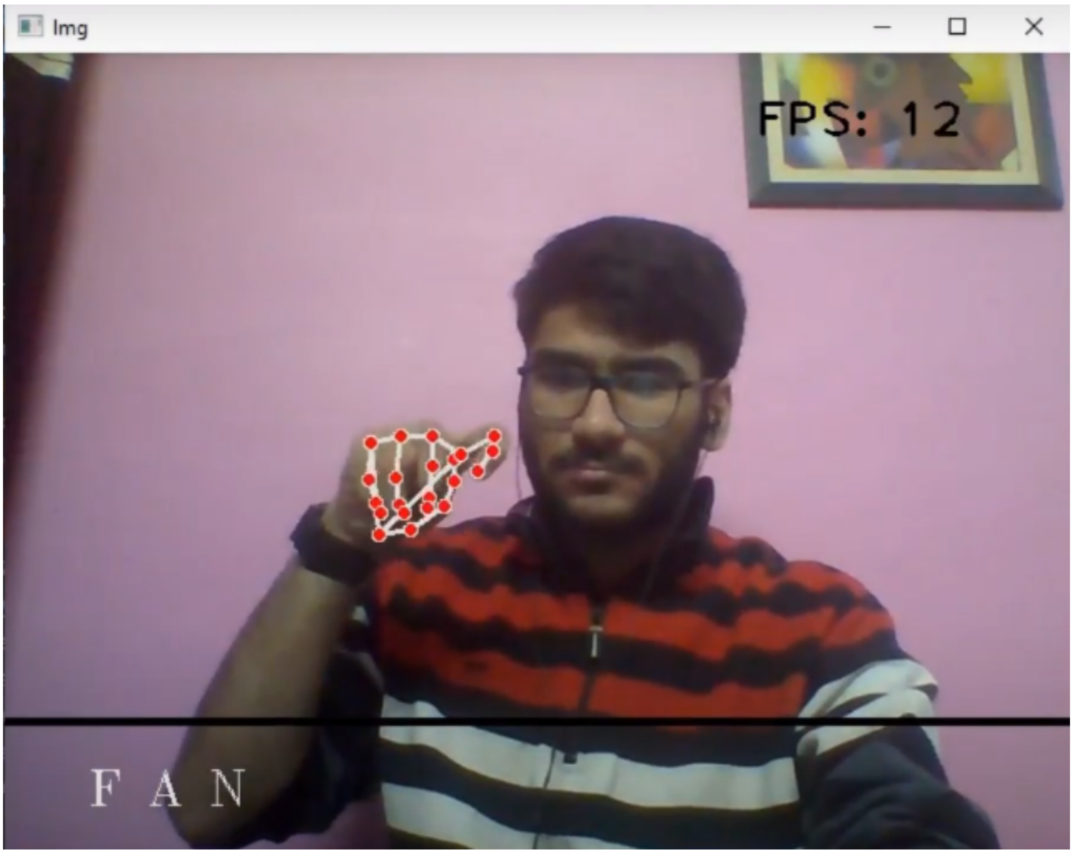


Figure 4: User Interface Design

The user screen is divided horizontally into two parts, of which the lower most portion shows the gestured text. Once the word is confirmed by the candidate the user sees a formed word or a sentence in the lowermost portion of the screen. Additional features of text to voice output and translation option can be found at the rightmost portion of the window, which activates when selected.

4.6 Summary

The software works primarily on the principal of match and dispatch principle. Well-defined calculations are fed for each gesture which when matched outputs the text. As the program does not involve any sort of datasets and learning algorithms, it gives an additional advantage of being fast and also requires a static and a defined memory space.

CHAPTER-5:

TECHNICAL IMPLEMENTATION & ANALYSIS

5.1 Outline

Through the software developed, it was aimed so that it can be used by more and more people. A special care has been taken to ensure it can run on wide variety of devices, from low to high end mobiles and computers to many real time devices, with a minimal as well as a static memory requirement.

The program requires no such additional software's and hardware's, only a minimal amount of space is required to install in the device. Camera is the only compulsory requirement in the device with which the software records the gestures. The program can run offline and doesn't require any sort of connectivity from servers.

5.2 Technical coding and code solutions

Importing libraries

```
import cv2
import time
import numpy as np
import Hand_Tracking_module as htm
import math
```

Figure 5

Setting up camera and using hand detection module to detect hands

```
wCam, hCam = 640, 480
cap = cv2.VideoCapture(0)
cap.set(3, wCam)
cap.set(4, hCam)
pTime = 0
detector = htm.handDetector(detectionCon=0.7)
```

Figure 6

Setting up global variables

```
count = 0
CN=25
word = []
pword = " "
str = " "
```

Figure 7

Defining a function to append gesture words

```
def call(let):
    global pword
    if (pword != let):
        word.append(let)
        pword = let
```

Figure 8

Initiating for loop and running cv2 as well as hand detection to identify our hands gestures

```
while True:
    success, img = cap.read()
    img = detector.findHands(img)
    lmList = detector.findPosition(img, draw=False)

    cv2.line(img, (0, 400), (640, 400), (0, 0, 0), 3)
```

Figure 9

Setting up variables to store the pattern positions

```
    if len(lmList) != 0:

        Tx, Ty = lmList[4][1], lmList[4][2]
        T2x, T2y = lmList[3][1], lmList[3][2]
        Ix, Iy = lmList[8][1], lmList[8][2]
        Sx, Sy = lmList[20][1], lmList[20][2]
        M4x, M4y = lmList[9][1], lmList[9][2]
        M3x, M3y = lmList[10][1], lmList[10][2]
        Mx, My = lmList[12][1], lmList[12][2]
        Px, Py = lmList[0][1], lmList[0][2]
        I3x, I3y = lmList[6][1], lmList[6][2]
        Rx, Ry = lmList[16][1], lmList[16][2]
```

Figure 10

Setting up pattern relations

```
INTH = math.hypot(Ix - Tx, Iy - Ty)
M4TH = math.hypot(M4x - Tx, M4y - Ty)
M3TH = math.hypot(M3x - Tx, M3y - Ty)
MI = math.hypot(Mx - Ix, My - Iy)
IP = math.hypot(Ix - Px, Iy - Py)
I3TH = math.hypot(I3x - Tx, I3y - Ty)
RS = math.hypot(Rx-Sx, Ry-Sy)
ST = math.hypot(Sx-Tx, Sy-Ty)
T2M = math.hypot(T2x-Tx, T2y-Ty)
MT = math.hypot(Mx-Tx, My-Ty)
SI = math.hypot(Sx-Ix, Sy-Iy)
MR = math.hypot(Mx-Rx, My-Ry)
RI = math.hypot(Rx-Ix, Ry-Iy)
RT = math.hypot(Rx-Tx, Ry-Ty)
```

Figure 11

Defining letters and words for gesture matching

```
if (IP >= 75 and IP <= 110 and ST>160 and ST<200):
    count = count + 1
    if count > CN:
        let = "A"
        count = 0
        call(let)
```

Figure 12

To display fps

```
cTime = time.time()
fps = 1 / (cTime - pTime)
pTime = cTime

cv2.putText(img, f'FPS: {int(fps)}', (450, 50),
            cv2.FONT_HERSHEY_PLAIN, 2, (0, 0, 0), 2)
```

Figure 13

5.3 Working Layout of Forms

The application developed doesn’t involve any sort of forms. The user can simply execute the application to record gestures to get a translated output.

5.4 Prototype submission

A prototype of the model was used to experiment and create analysis of the performance. The prototype was delivered to a small group of people and tested for its accuracy, in different conditions, hand sizes backgrounds. Also, it was said to them to randomly show the gestures using the provided defined ISL datasets, and in turn we had to record the output displayed by the software. Success and thus accuracy was calculated when the shown gesture and the predicted word matched.

5.5 Test and validation

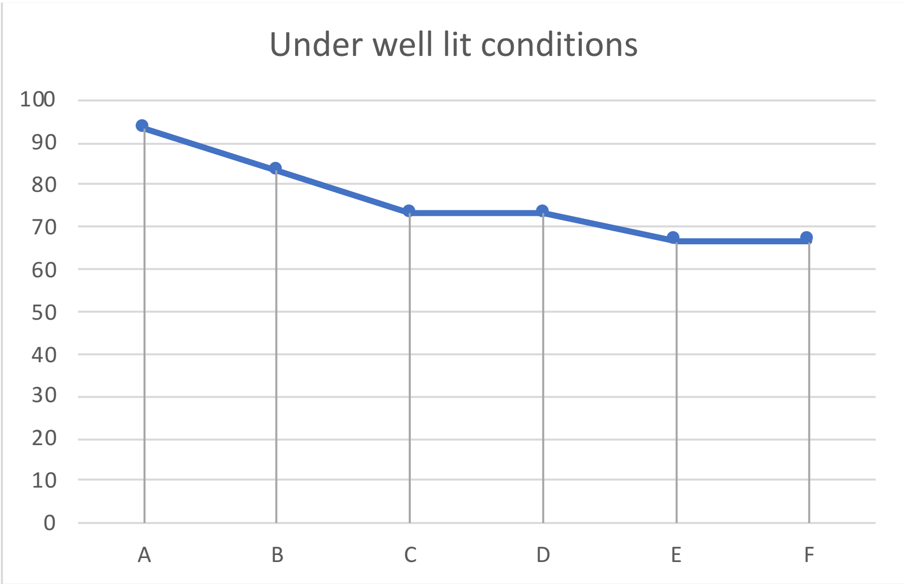
On analysis on various platforms, devices and situations various reports were created. Devices along with features were divided into 6 broad groups, where group A had the best features, while group F had the least features. The grouping table is as following. Almost 30 tests were conducted and successes and failures were noted and an accuracy percentage was calculated.

Groups	Features
A	13mp camera, depth sensor, lightning optimizer, 8gb ram, 144 htz refresh rate
B	13mp camera, depth sensor, 8gb ram, 144 htz refresh rate
C	13mp camera, depth sensor, 6gb ram, 120 htz refresh rate
D	10 mp camera, 6gb ram, 120 htz refresh rate
E	8mp camera, 4gb ram, 60 htz refresh rate
F	4mp camera, 2 gb ram, 30 htz refresh rate

Table 1

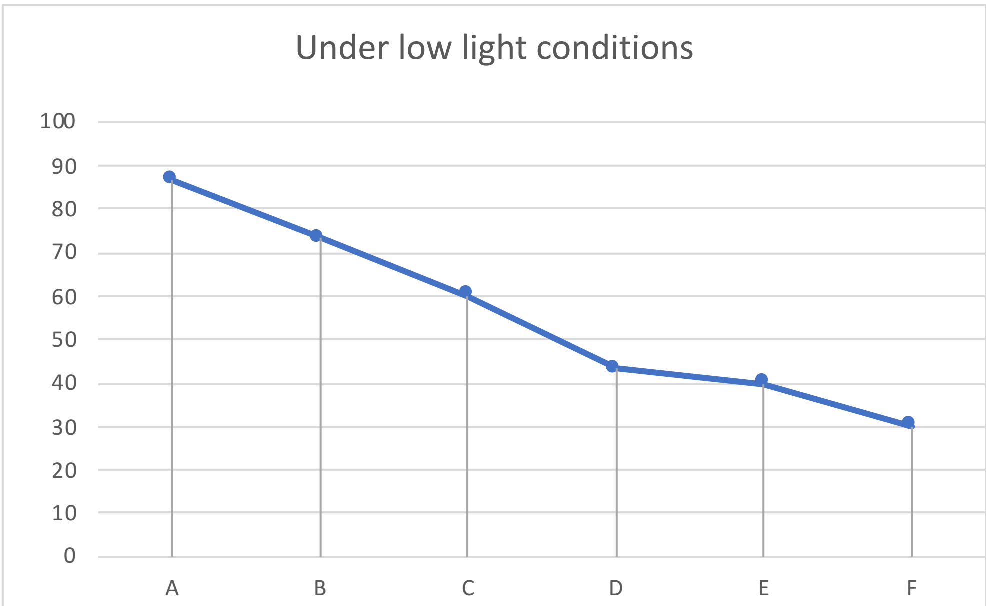
The following result analysis were obtained, where the y axis shows the accuracy percentage while the x axis shows the groups:

1. Accuracy of the software under well-lit condition



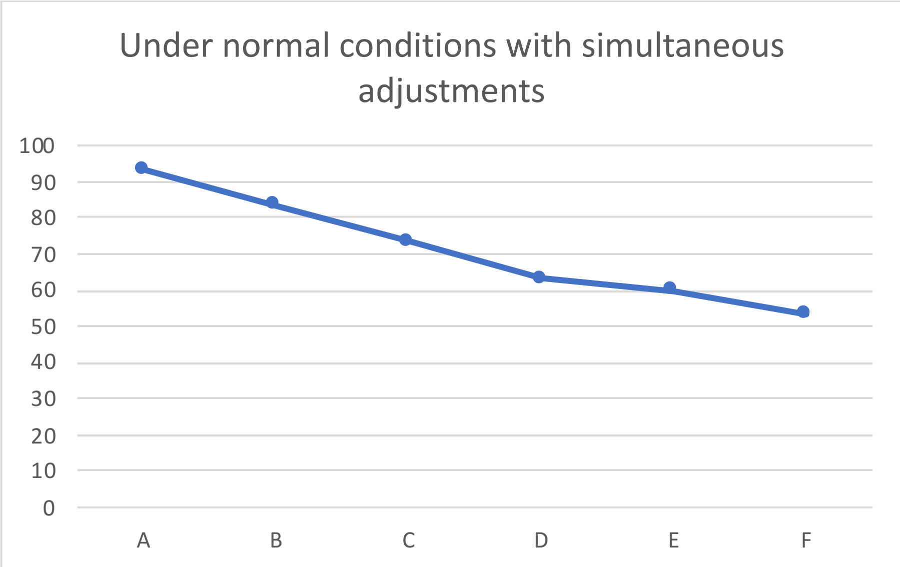
Graph 1

2. Accuracy of the software in low light condition



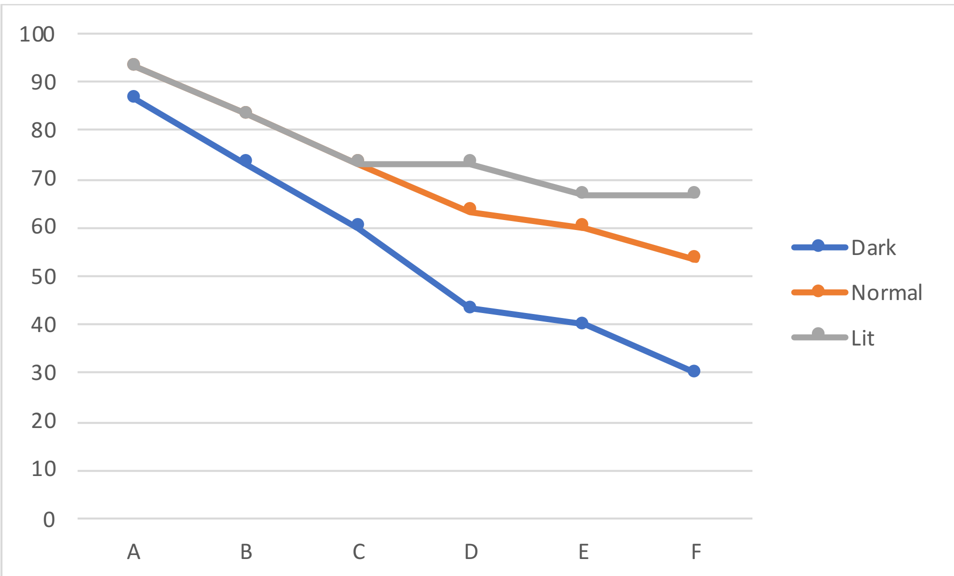
Graph 2

3 Accuracy of the software under normal conditions with simultaneous adjustments .



Graph 3

The combined results were as following:



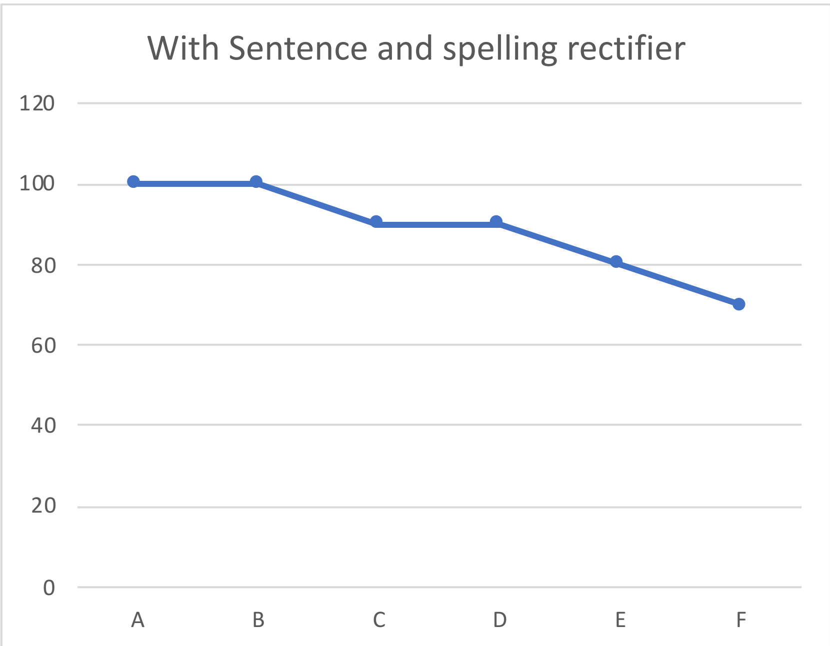
Graph 4

5.6 Performance Analysis(Graphs/Charts)

It was seen that the program works best when the camera quality of the device was more than 8-megapixel. Although there were a bit fluctuations and disturbances in normal conditions, it could provide an almost acceptable result. Adjusting the detection concentration as well as sensitivity also helped in providing accuracy. It was also reported that gesture detection showed a high level of accuracy in a well-lit condition with the hands clearly detectable.

Devices which further had features of light optimization and depth analysis results observed were not only fast but also very accurate. For low end devices, it was sometimes seen the program took a bit time to start and run, but was smooth once started.

Low accuracy of detection was corrected using dictionary which resulted into almost correct outputs.



Graph 5

5.7 Summary

The program can run on any sort of smart devices, from mobile phones, to laptops and computers to various real time devices like cars, music systems and more. Although to give a minimum accurate result, camera quality of at least 6-megapixel is required on all to be used devices. Performance accuracy could be seen increasing as the features of the device in which program been run was increased.

Hence it can be concluded that the program or the software can be relatively extended to a large number of audiences with a minimal or no additional expenses. Even though high-end devices showed a better performance, but the program faces almost no issues when run-on low-end devices but required a bit of adjustments in detection and sensitivity. In all the software proves be a potential, effective as well as a hustle free method for bridging the gap between the two communities.

CHAPTER-6:

PROJECT OUTCOME AND APPLICABILITY

6.1 Outline

Our project aims at giving an equal status to deaf and mute people. We keep on saying that they are equal but still if we can't even converse with them because we don't know sign language. Here comes our project as it gives them a chance to use their sign language and our program will interpret it and give others the message in English. They do not need to learn a new language they can speak any word cause our program uses letters and nor do others need to learn sign language. This in totality explains why our project is made what it aims to do and where is it applicable.

6.2 key implementations outlines of the System

In order to understand our project, we need to know open CV. OpenCV (Open Source Computer Vision Library) is an open source computer vision and machine learning software library. OpenCV was built to provide a common infrastructure for computer vision applications and to accelerate the use of machine perception in the commercial products. Being a BSD-licensed product, OpenCV makes it easy for businesses to utilize and modify the code. For our project we used the hand detection feature of open CV where it detects all the points of the hand.

Figure 14: Hand Detection Features

Next we created a program which actually detected the hands and could distinguish it from the rest of the background. Once we had created this code. We used it in our main file by importing it. In this file we first gave names to all the specific points then we named the distance along the x-axis and the distance along the Y-axis when we have got these distances, we take the hypotenuse and got the actual distance between the points. We then made the user interface where the person will be able to show their hand and a specific place so that they can see the result. After this we took the signs by trial and error got the distances and completed all the alphabets.

6.3 Significant project outcomes

The outcome of the project is to give voice to the opinions of deaf and mute so that they can also help us towards the welfare of the society. Our project is acting as a bridge to gap the distance between the deaf mute people and others by allowing the latter to understand in their own language while the deaf mute people keep on using sign language.

6.4 Project applicability on Real-world applications

Our project can be used in day to day conversation provided you have your laptop with you and the place you are in has adequate lighting. The application will work perfectly for communicating in an office. It is a realistic application which is fast and efficient which is needed to be productive for the welfare.

6.5 Inference

Our project is efficient, fast and usable helping in giving the right to be equals in the society to the deaf and mute people.

CHAPTER-7:

CONCLUSIONS AND RECOMMENDATION

7.1 Outline

After months of reviewing our project, assessing it, modifying our goals in general, we have concluded that our project is enabling deaf, mute people to communicate. Our Team Ishara has collaboratively made a project to empower those who can’t speak by giving them the right to speak. We’ve effectively made a program that solves the underlying problem of sign language and its use by giving everyone a common platform to communicate which demolishes any discrimination and promotes an inclusive environment and society around us.

7.2 Limitation/Constraints of the System

Though, even after months of coding and regressive attempts to be error-free, we do have some limitations on our project. The Autocorrect function is not entirely perfect and there are still things we can develop more in that area. Apart from that, the camera requirements are still there, which means, there is a limitation of up to greater than an 8-megapixel webcam and a memory space of up to 150 megabytes.

7.3 Future Enhancements

As our project has progressed in this past time, we are very much engaged & excited for various future endeavors and enhancements regarding our project. We plan to focus on a lot of unique integrations and updates, such as a virtual assistant integrated into our system that can work efficiently and further aid ourws motive. Moreover, the overall system can be adapted to learn the facial expressions & body language of an individual to assess the tone of the input speech through the involvement of Machine Learning. To fully achieve our final goal of a more inclusive society, we also plan to develop a web-based application of our program to increase the reach of our project, through the WWW. Furthermore, we plan to initiate a two-way system for our project which makes the back and forth of sign language even easier.

- Auto correct

We are using autocorrect library to recommend the words. However, this library is not the best hence in future we will be thinking of using another library or eventually creating a library which will be suited for our project needs

- Configuration of backspace

Backspace is required if anyone makes a mistake and therefore it would really help the project to take it to the next level.

7.4 Inference

At last, we infer that there has been immense research to find various methodologies to bring the special people in streamline with the society more. As junior developers, we have the power to develop better technologies so as to bring a better way for these people to communicate

References

[1] Anuja V.Nair, Bindu.V, “A Review on Indian Sign Language Recognition”, International journal of computer applications, Vol. 73, pp: 22, (2013).

[2] Archana S. Ghotkar, Rucha Khatal, Sanjana Khupase, Surbhi Asati & Mithila Hadap, “Hand Gesture Recognition for Indian Sign Language”, IEEE International Conference on Computer Communication and Informatics

(lCCCI), pp: 1-4, (2012).

[3] Hu Peng, “Application Research on Face Detection Technology based on Open CV in Mobile Augmented Reality”, International Journal of Signal

Processing, Image Processing and Pattern Recognition, Vol. 8, No. 2 (2015).

[4] Meenakshi Panwar, “Hand Gesture Recognition based on Shape Parameters” International Conference on Computing, Communication and Application (ICCCA), pp: I-6, IEEE, (2012).