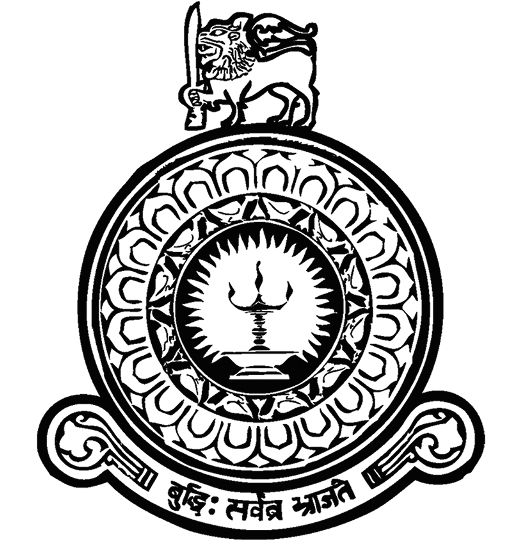
GESTURE DETECTOR USING DEEP LEARNING



A dissertation presented in partial fulfilment of the

requirements for the degree of

B.Sc. (Hons)

IT and Management

in

Individual Research

At IT Unit-2, Faculty of Science, University of Colombo

Sri Lanka.

Methmal Udan Abeywickrama

2022

**Declaration**

I hereby declare this the work reported in this research report is exclusively carried out by me under the supervision of Dr.Uditha Prabhath Liyanage. It describes the results of my independent research except where due reference has been made in the text. No part of this research report has been submitted earlier or concurrently for the same or any other degree.

Candidate:

R.M.K.G.D.Suranga

Reg. No: 2016S15719

Index No: s13555

Date ……………………….... Signature: …………………………

**Certification of Approval**

We hereby declare that this research report is from the student’s own and own effort also all other sources of information, which were used have been acknowledged.

This report has been submitted with our approval.

Supervisor:

Dr. U.P. Liyanage Signature

Senior Lecturer,

Department of Statistics & Computer Science,

University of Kelaniya.

Date ……………………….... Signature: …………………………

Coordinator:

Dr. S.K.P. Eranga

IT Major Theme Coordinator,

Department of Mathematics,

The University of Colombo.

Date ……………………….... Signature: ………………………

**Abstract**

 Gesture detection is an uncommon type of communication that often goes unstudied. Although those gesture languages can be different from person to person, should be able to understand the receiver. If not, it becomes useless. This is very important for individuals who have impaired hearing and inability to communicate with people who don't know the universal sign language. Aiding the cause, **Deep learning**, and **computer vision** can be used too to make an impact on this cause. So that this project is going to implement some software that detects the hand gesture and produces some output as a sentence. Here used CNN (Convolution Neural Network) For recognition. Train the model as a dataset I created my own generated dataset. For adding newly customized marks I used sift algorithm. Generating voice output, I export the generated text in a text file.

**Acknowledgments**

I would like to extend my sincere thanks to all individuals who helped and guided me in making my effort on this project more meaningful. First and foremost, I would like to offer my heartfelt gratitude to my project supervisor Dr. Uditha Prabhath Liyanage for helping, stimulating suggestions, supervision, and encouragement to continue my project successfully. Also, I would like to express my gratitude towards my parents for their kind co-operation and encouragement which help me in the completion of this project. Not only that, I would like to express my special gratitude and thanks to industry person Mr. Thilak Fernando who is managing director of Datadisca, for giving me such attention and time. My thanks and appreciations also go to my colleague in developing the project and people who have willingly helped me out with their abilities.

**Table of Contents**

Contents

[**1.** **CHAPTER 1** 7](#_Toc98093763)

[**1.1.** **Introduction** 7](#_Toc98093764)

[**1.1.1.** **Introduction of the real-world problem that is solved by the software** 7](#_Toc98093765)

[**1.1.2.** **Significance** 8](#_Toc98093766)

[**1.1.3.** **Customer requirements** 8](#_Toc98093767)

[**1.1.4.** **Scope and objectives of software solution** 8](#_Toc98093768)

[**1.1.5.** **Outlook of the thesis** 9](#_Toc98093769)

[2. CHAPTER 2 10](#_Toc98093770)

[3. CHAPTER 3 12](#_Toc98093771)

[4. CHAPTER 4 15](#_Toc98093772)

[5. CHAPTER 5 17](#_Toc98093773)

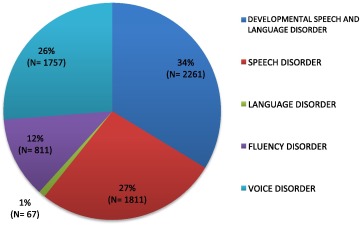
[6. CHAPTER 6 18](#_Toc98093774)

# **CHAPTER 1**

## **Introduction**

This chapter aims to provide a firm background on the research topic. This chapter discusses the Gesture Detection System.

### **Introduction of the real-world problem that is solved by the software**

The main problem people who have speaking ability is can’t express their feeling like normal people. Also, voice recognition and google search systems in smartphone can’t be able to use like normal people. So It is compulsory for such platforms for such kind of people. American Sign Language (ASL) is a complete language, But it complex. Because it signs made by not only moving the hands, but also combined with facial expressions.

Gesture detection is a unique type of communication that often goes unstudied. These gestures interpret opinions of each other without using voice in any specific language. Although those gestures' languages can be different from person to person, should be able to understand by the receiver. If not, it becomes useless. This is very important for individuals who have impaired hearing and inability to communicate with people who do not know the universal sign language.

Text

Description automatically generatedEspecially in university, I met many people with an inability to speak. When I try to communicate with them I couldn’t able to succeed most of the time. That’s how I got an idea to implement a project like this as a solution. But however not only we can use this with speaking inability, but we can also use this to communicate with foreigners

who don’t know any intersection language. So, we can install this system hotel reception, counters Likewise.

### **Significance**

This application includes two major parts. The first part detects gestures and displays the related letter. The second module will scan frames for a certain period get into some array. From it produces related words or sentences. It is simply strung. That goes some vice mechanism to produce it is voice output. Also, like special characters ‘/ ‘, they can build their custom gestures. Not only that user can include some word or small sentence using customer gestures.

**Customer requirements**

This system is designed for people who have a speaking ability. However, people who are deep and dump people have their universal language but all the people in our society aren’t capable of understanding that language. That is some kind of a barrier for people who have speaking disabilities to deal with common society.

Here I created a system using my dataset and we can include custom gestures for this system. So, the user can add customize a sign for his system.

Using user input of gesture’s sign system provides voice output from the system.

### **Scope and objectives of software solution**

* Implement software solution for people who are deaf and dumbing communicating with others as knowing sign language
* Language may be common to all, moreover, this can be extended to creating automatic editors, where the person can easily write by just their hand gestures.
* create a sign detector, which detects numbers from 1 to 10, including the alphabets that can very easily be an extended multitude of other signs and hand gestures.
* After detecting signs at moment, we want to read the sentence which was shown using sign language.

### **Outlook of the thesis**

This dissertation is divided into six chapters, each of which has a brief introduction as follows

Chapter 1: Introduction

* This chapter contains the introduction of the real-world problem that is solved by the software solution, significance, requirements, scope, and objectives of software solution, and the outlook of the thesis

Chapter 2: Literature Review

* Presents a clear analysis of the existing situation and the requirement analysis of the system. It further analyses the available system options and defines the most feasible option to proceed with.

Chapter 3: Software specification and design

* Is the depiction of the furtherance of the project after the requirement analysis and specification. This will provide a better understanding of the system behaviour and interactions with the diagrams used. The mind map will lead to another to explain the functionality entities and their relationships. By the end of the chapter, database design will show the tables which will be used in the system and Graphical User Interfaces will further elaborate its performance.

Chapter 4: Software Feature Description

* This chapter contains a complete description of the software solution, justification of the features, possible features, and feature highlights

Chapter 5: Discussion and Demonstration

* Manual document of the software solution with screenshots. then explain the uniqueness and innovativeness of the software solution

Chapter 6: Conclusion

* In this chapter discusses the developed software solution and encountered problems. Then discuss the possible extensions of the software solution

# CHAPTER 2

### Literature Review

Literature review of the problem shows that there have been several approaches to address the issue of gesture recognition in video using several different methods. There are some exciting systems and their description.

### Existing solution and their description

* TLDR It is presented a dual-cam first-vision translation system using convolutional neural networks. A prototype was developed to recognize 24 gestures. The vision system is composed of a head-mounted camera and a chest-mounted camera and the machine learning model is composed of two convolutional neural networks, one for each camera.
* M. M. Sole and M. Tsoeu used Extreme Learning Machine (ELM) to learn to classify static hand gestures on the letters of the Auslan dictionary in AFRICON, 2011.
* T. Kim, W. Wang, H. Tang, and K. Livescu create a system “Signer-independent fingerspelling recognition with deep neural network adaptation,” which is demonstrated a way to utilize a deep neural network (DNN) to classify frames in image sequences of fingerspelled letters. Their work was signer-independent meaning capable of recognizing hand poses of any user. They used Histogram of Gradients (HoG) image features [\*] as input to the deep neural networks.
* The work of N. Pugeault and R. Bowden created the system “Spelling it out: Real-time asl fingerspelling recognition,”, which is presented the use of depth images from a Microsoft Kinect device. They used multiclass random forest classification and tested their method by varying the input from image only, depth only, and combined image with depth. They achieved their best result when depth is combined with the image. Their system is also fast enough for real-time classification

**Comparison of the advantages and disadvantages**

**Advantages of the existing system**

* Used two CNN network
* Used ASL system

**Disadvantages of the existing system**

* Used common open-source source dataset
* Cannot be customized their language
* Doesn’t provide voice output of the system
* Can’t produce voice output

**Literature for the analytical part of the solution**

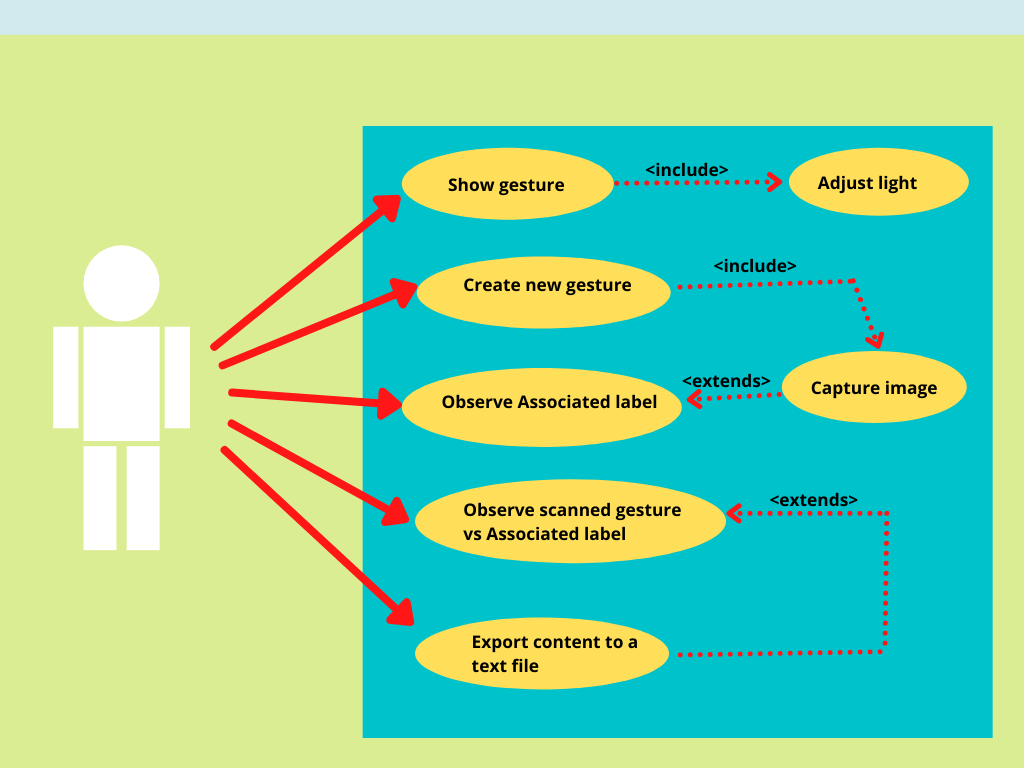
# CHAPTER 3

**Software specification and design**

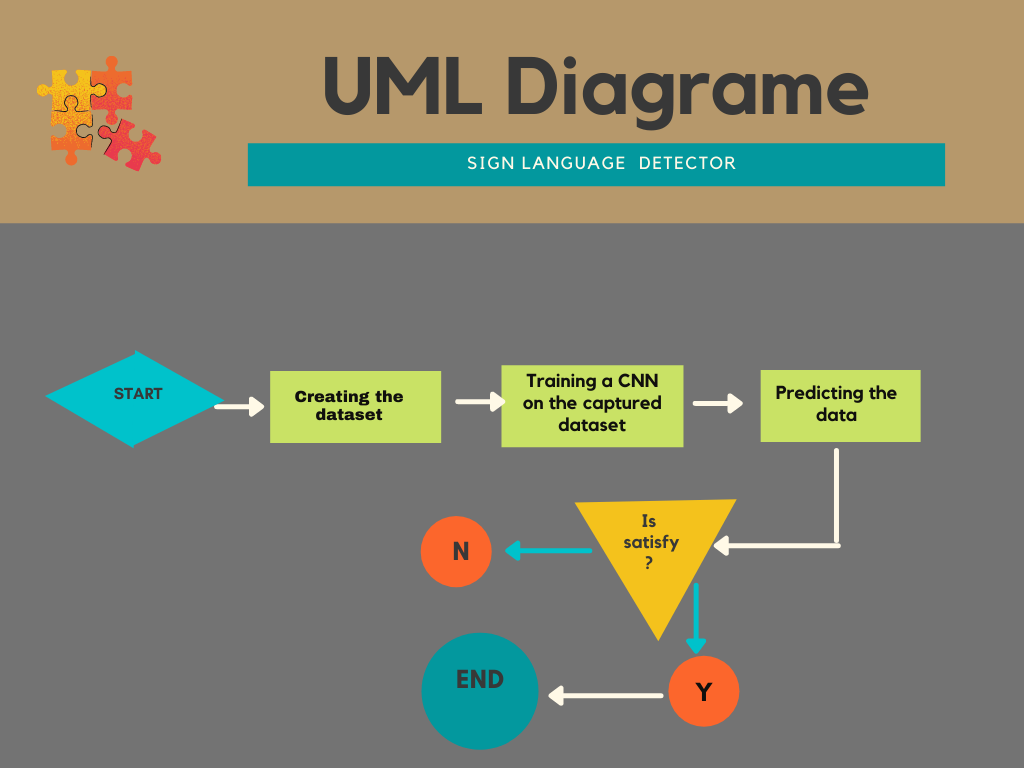
This chapter has presented approaches for designing the proposed system. This chapter includes the design for the research project. The tasks and functionalities performed by the proposed system are Introduced and analyzed through the use case analysis. Also, important use cases are furthermore described by using use case descriptions. The way of performing the functionalities is explained by using a use case diagram and mind map. Also, there are some prototype interfaces are shown related to the system interface.

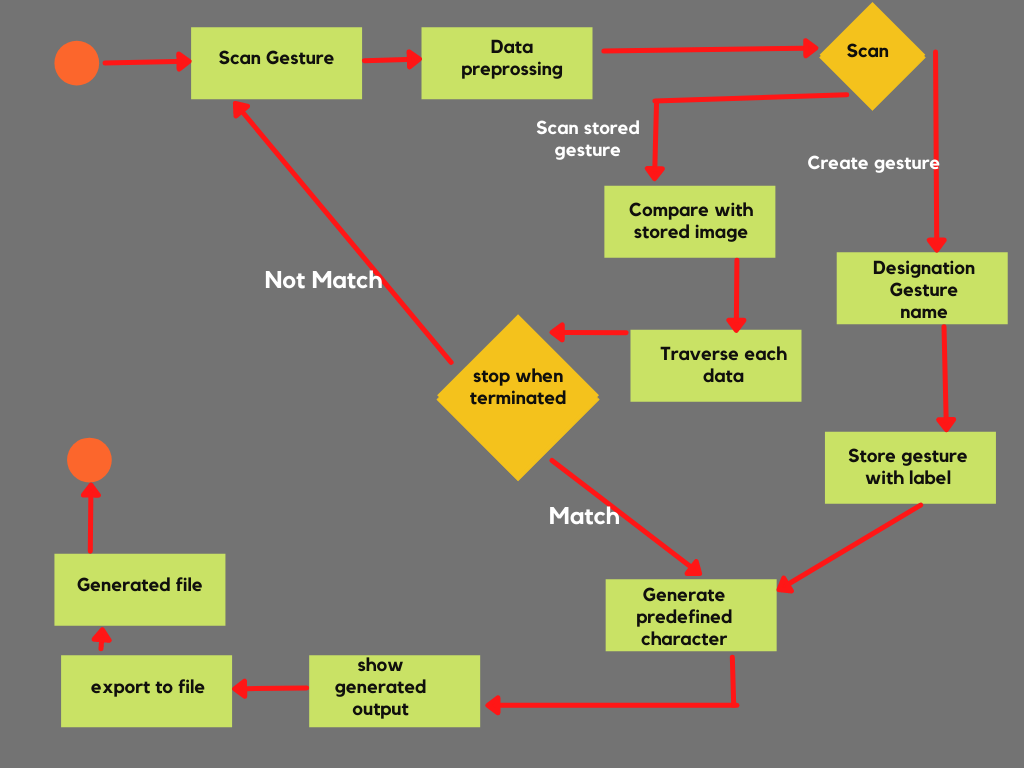
**Use case diagram for overall system**

The functionalities of the proposed system are shown in a uses case diagram to provide a brief and clear explanation of the overall project

**Mind map with relational justification**

**Mind map with relational justification**



**Research activity diagram**

# CHAPTER 4

**Software Feature Description**

* User-friendly interface- In this project, I created a user interface using the PYQT5 python package. Using designer. Here In the user-interface welcome interface and then there are four buttons, such that ‘Create gesture’, ‘Scan Gesture’, ‘Scan Sentences’, ’Export to file’. Also, there are color balancing panels and related boxes for displays and input-output facilities.
* Real-time character recognition- Here using CNN trained model file to connect to the dash.py file in the project. Also, here project captures continuous images using the same name.
* Customized character – If there is not any gesture related to some character, the User can add some gesture to it. For that, I used shift algorithm. Using it user can input the character name and gesture as his preference.
* Provide voice output of gestures – Using the TTS (text to speech) mechanism I added the voice output feature. As an export text file which is made from ‘Scan Sentences’ and used as input here.

**Justification of the feature within the software solution**

After the run, the system user can view the home view of the system

Then he He can select the three main selection

There are ‘Create Gesture’, ‘Scan Gesture’, ‘Scan Sentences’, and ‘Export to file’.

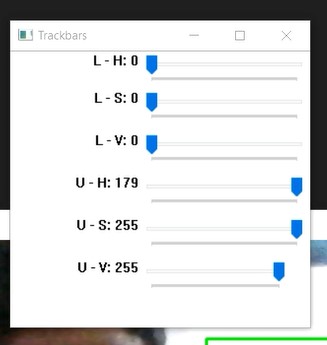
If he selects ‘Create Gesture’ he can add customize gesture value and character to the project.

In ‘Scan Gesture’ users can scan the gestures and get the output of it.

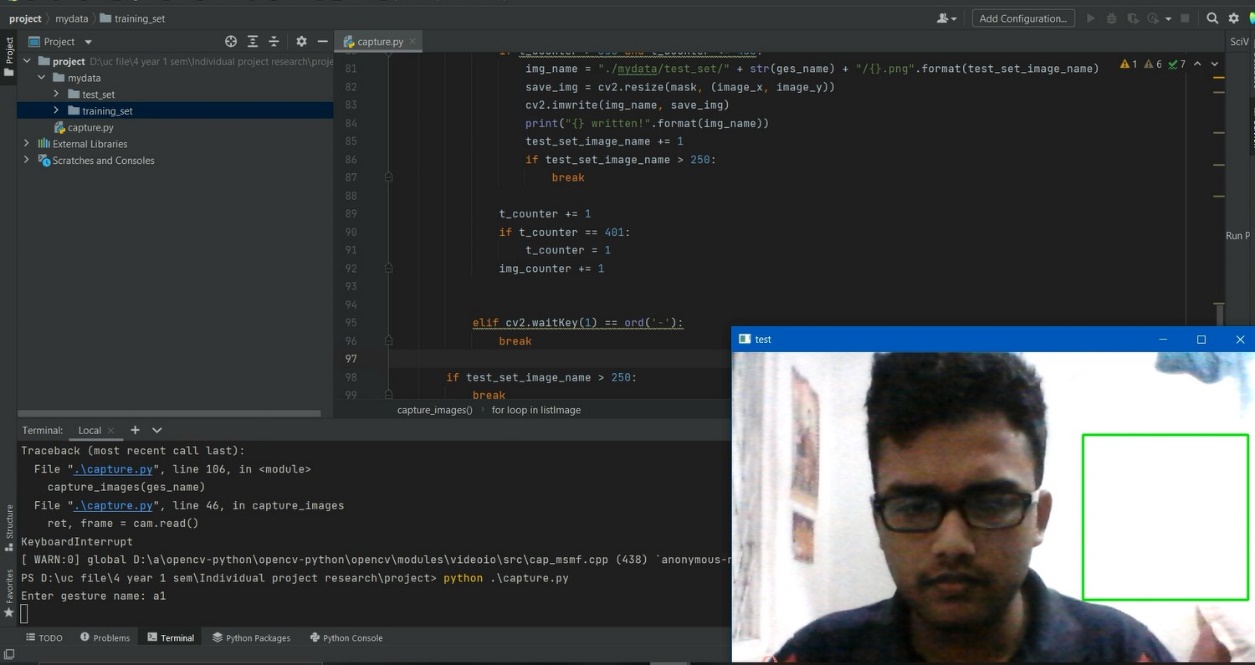
But if users select ‘Scan Sentences’, there is a mechanism to create sentences using gestures as input. In there are add gesture into an array and create a text file using it

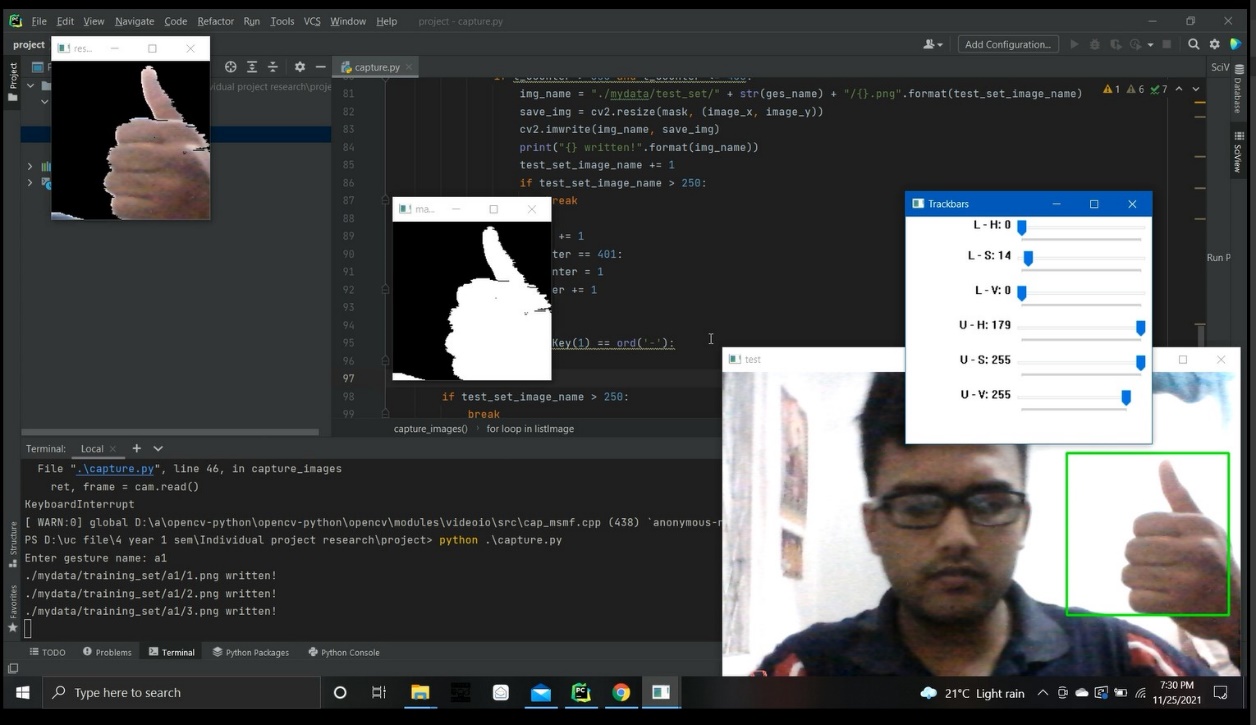
Finally, ‘Export to file’ interface was developed to import the saved text file in the previous step and produce voice output of it. Using the TTS system can produce voice output.

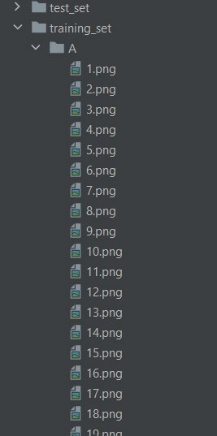
**Feature highlights**

As a first step data related to the research area was searched by using the internet. Before building the system, I refer to CNN networks and how it works. So get enough knowledge about it a move to learn about how to make my own gesture data set using python. Because in this project I planned to develop my own data set for this project. Using that I created my own data set for this project.

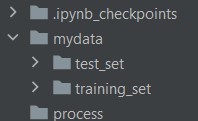
I created capture.py file and do code file. In there After running the file four windows will be pop out and got all them and adjust on my preference. After adjusting color bar according, the environment condition, I enter the gesture name and pressed enter. Then showing constant gesture will be captures by web camera and after pressing ‘c’ button it will automatically save as png files. After getting 350 training data it moved to get 50 test data. After it finished again store the next 350 training images. This mixture gaining method will help to keep randomness of the dataset. After taking total number of 1750 training images and total number of 250 testing images one character data set will be created. Like this way I created whole alphabet



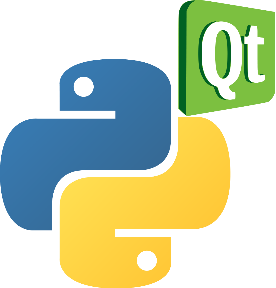


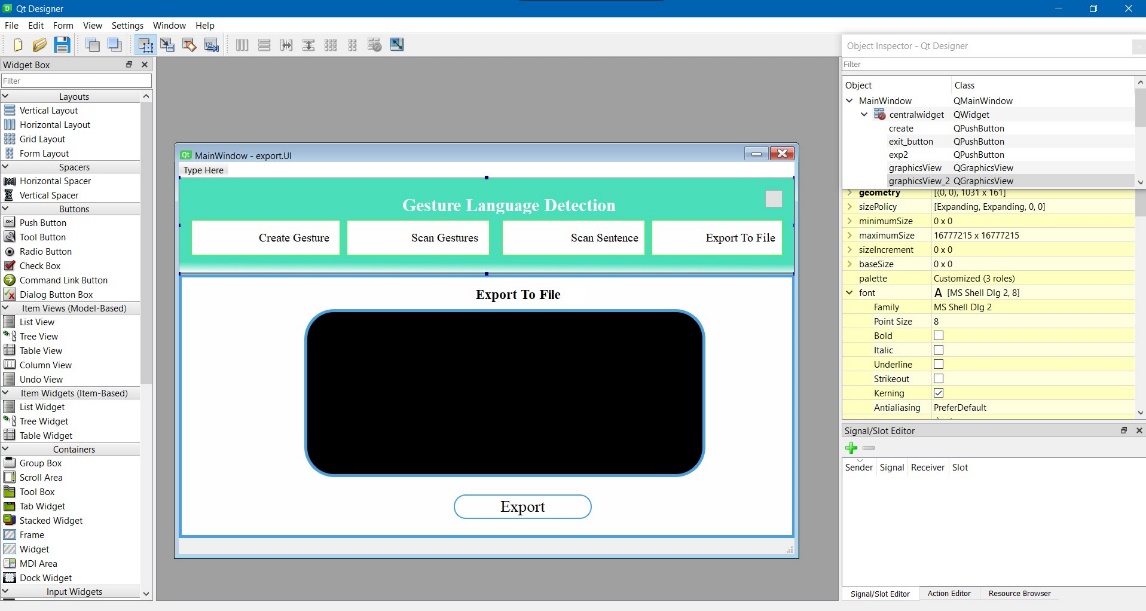




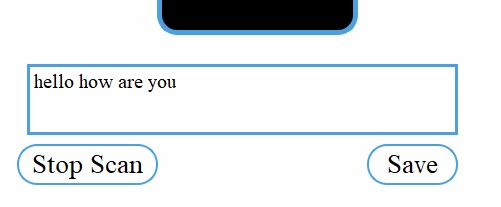


The next step was designing a sketch of my project. This sketch was based on the functions of the project. By using that plan of the project, I drew the entity-relationship Diagram. The development process was planned and divided into two parts. There are backend development and front-end development.

****The technology selected for front-end development was the Python PYQT5 using the designer. From it, I created interfaces that can be used to relevant categories in the features selections.

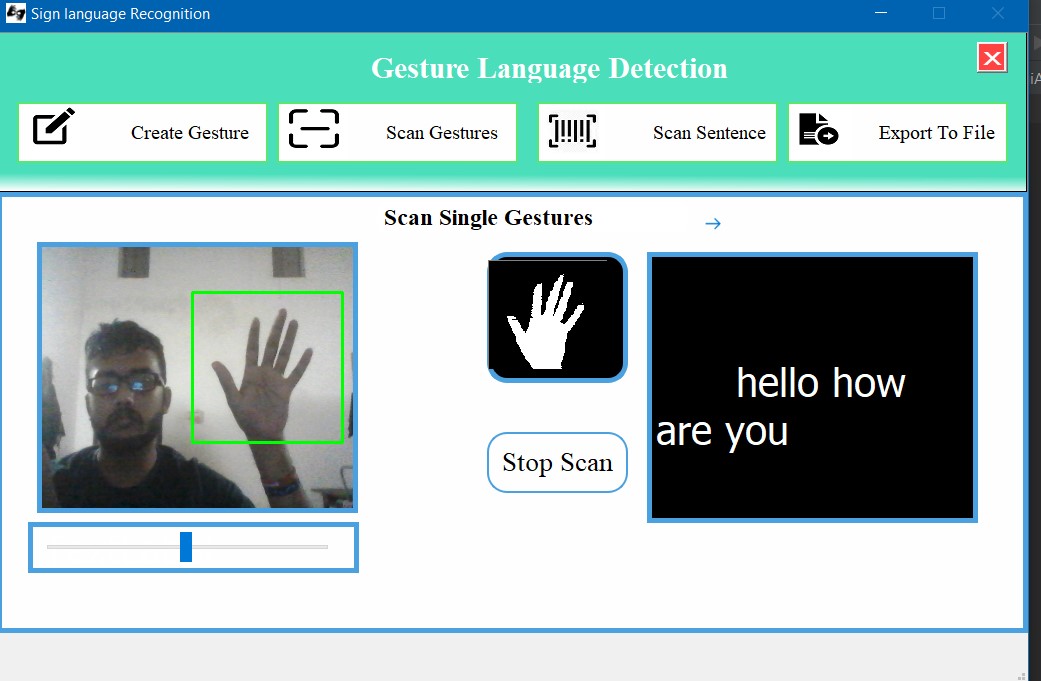
****

This Interface is very user-friendly. In Scan Sentences option user can make a text file using gesture input. After that he/she can export it. To do that user must go to the ‘Export To File Option’. Then system will be produced relevant output. For that I used TTS mechanism.

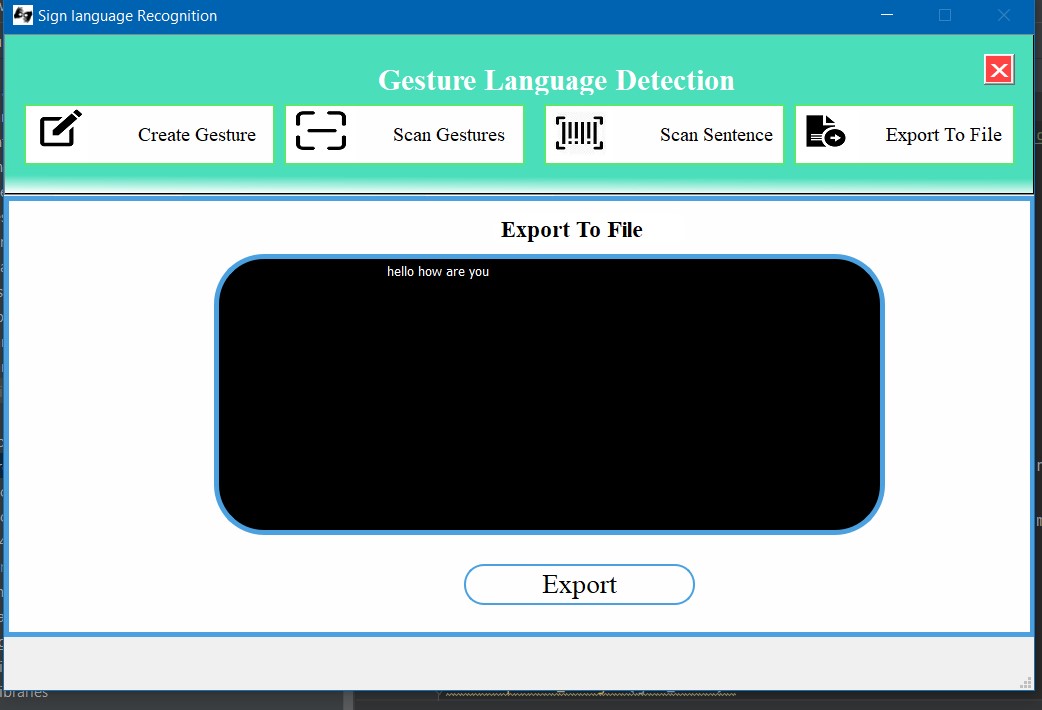
****

Also, there is a mechanism to create new gesture for newly added gesture using shift algorithm. In there user can be abled to create gestures not only single character and symbol but also sentences.

In the figure you can see, has been entered sentences for single gesture. Output of it is will be like following as figures.



After importing that output user can produce relevant voice output through the system



# CHAPTER 5

**Discussion and Demonstration**

This chapter explains how to use software with the documentation and screenshots use to explain. Also, the process of the software.

Installation

This section presents hardware and software requirements for this system.

* Intel Core i5 gen processor or later.
* 512 MB disk space.
* 512 MB RAM.
* Camera which is inbuild or external.

Software Requirements

* Microsoft Windows 10
* Python Interpreter 3.6
* TensorFlow framework

Hardware Requirements

This project requires python version 3.6 or later to synchronize with TensorFlow. WinGuiAuto.py. Using Capture.py file, create my own dataset and cnn\_model.py file. With the help of a deep neural network trained my model and Build the model with the name "trained.h5" using cnn.py.

Install the required packages

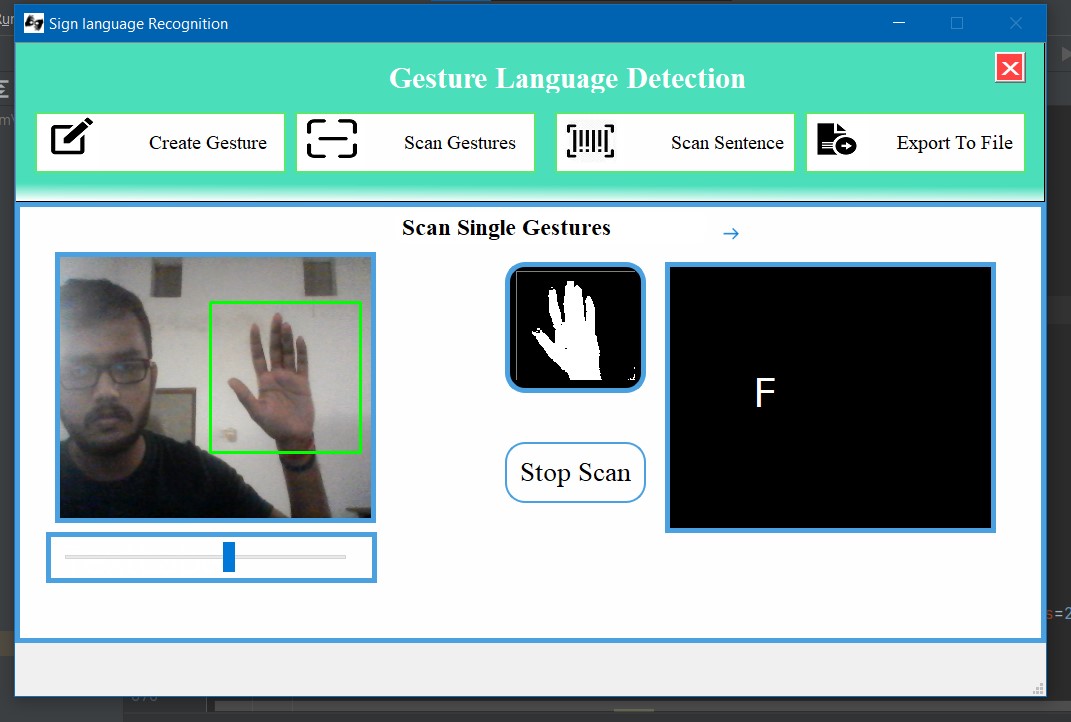
Start using by running the dash.py file.

* Packages that required to install
* python 3.6 or later
* keyboard
* pyqt5, Tkinter
* winGuiAuto
* pypiwin32
* pyttsx3
* Keras
* scipy
* opencv

First execute the system. Then you will receive the Opening interface. Opening view system view as follows. Always try to build some system which has interface easy for use.

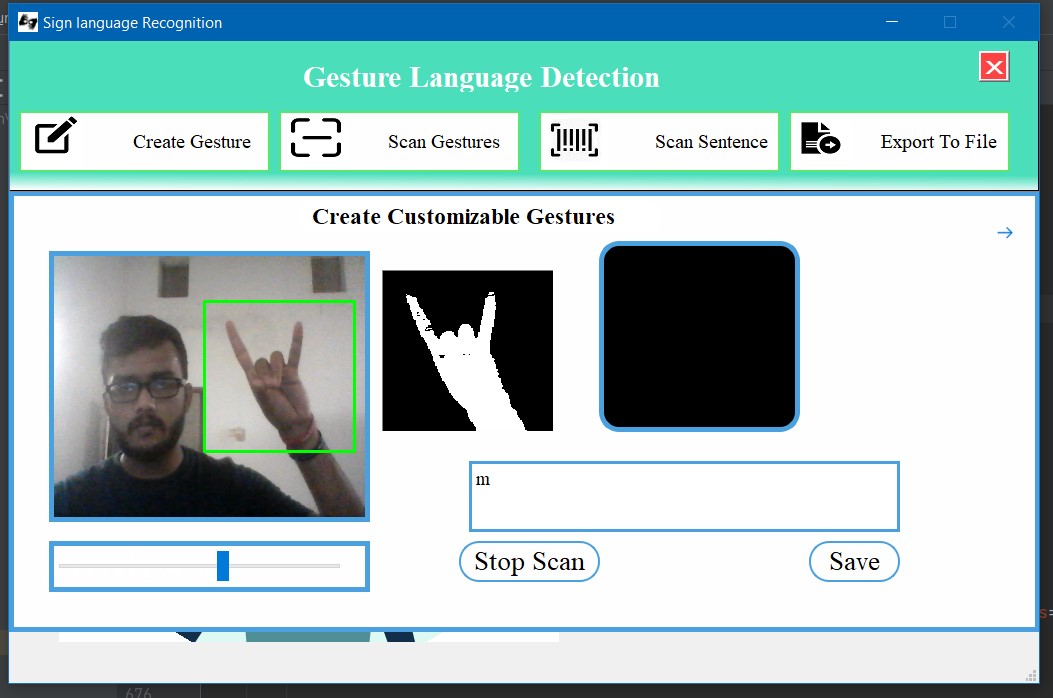


Then you have four options. There are Create Gesture, Scan Gestures, Scan Sentences and Export to file. If you Choose Scan sentences you will end up with following interface.

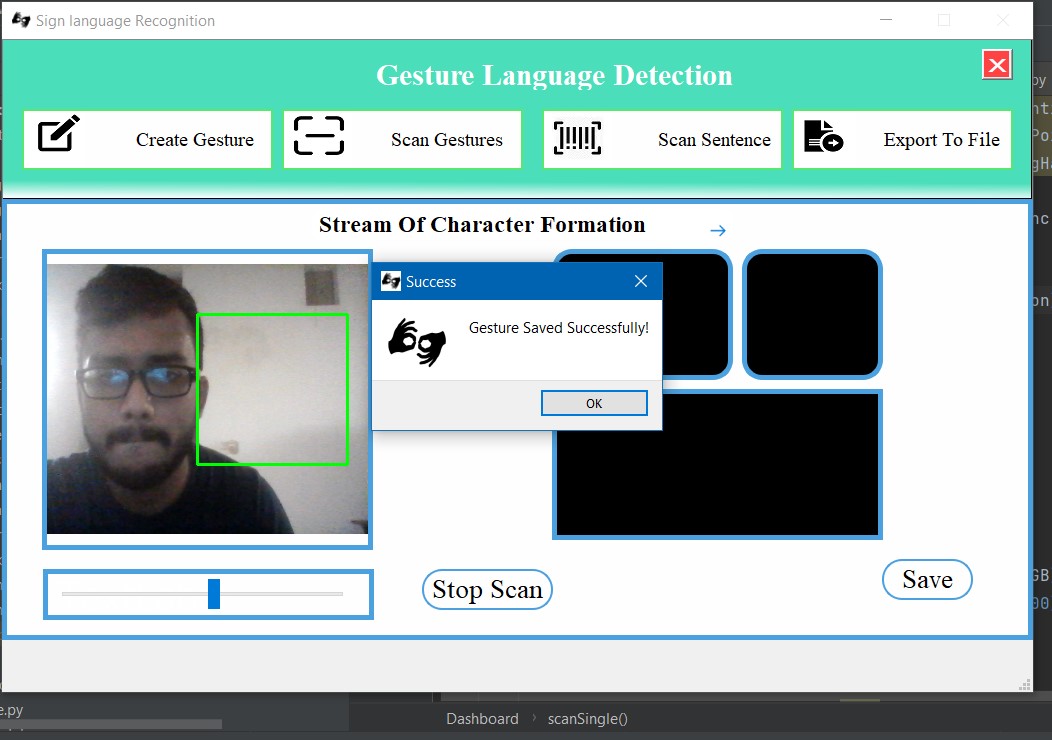


But you have to control lighting system according your environment. It is depended on your light around you. After suitable adjusting, you have to show your gesture in the green box and it will automatically get input and show the gesture after removing the RGB colors. Also, relevant character output you can see in next frame. If you press the stop button process will be paused and you have to select what to do next.

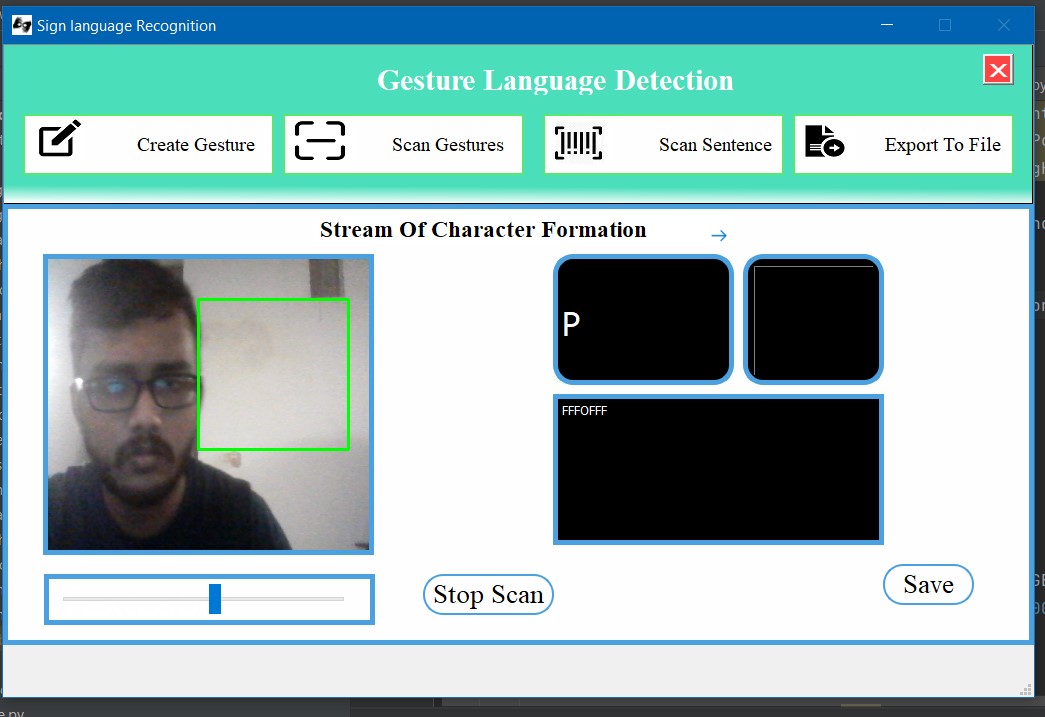
If you select Create Gesture button as next move then you ended up with following Interface.



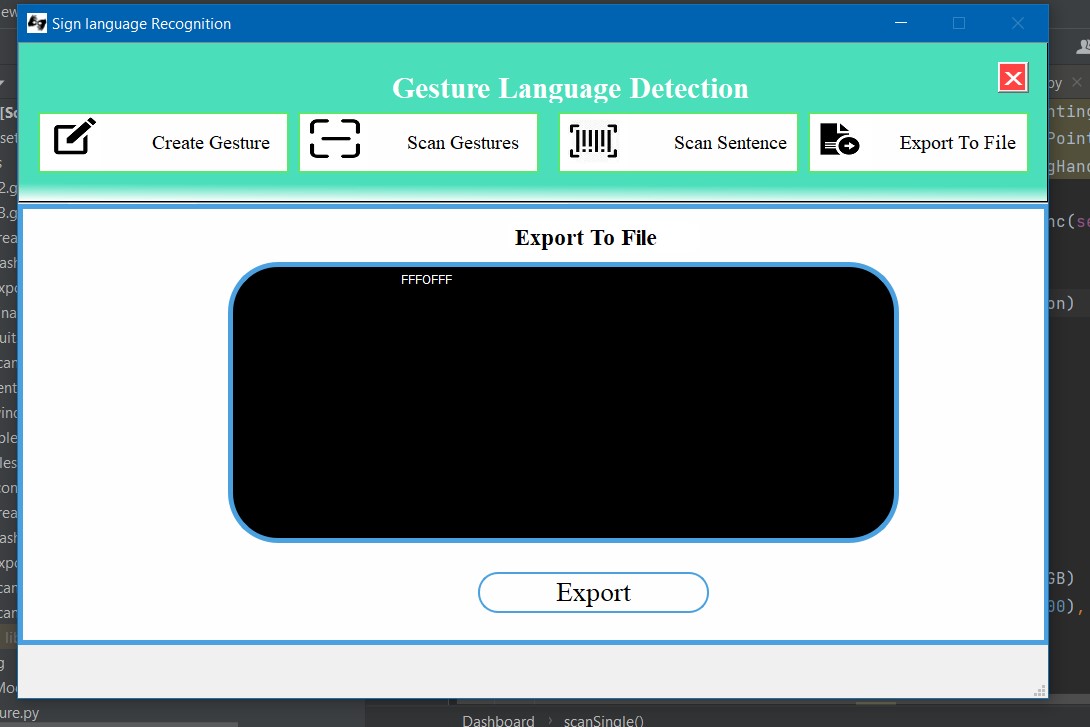
Here In ‘Custom Customizable Gestures you can observe three interfaces, one input panel and parameter bar for lightning control. Also, there are ‘stop scan’ and ‘Save’ buttons too. After adjusting lightning control panel, you can easily get the input of your gesture. After displaying into the frame of your gesture you have to enter character that represent that gesture on your preference. It may be single character, sentences or symbol. After that you can select save button or “Shift +s” button with together. After that you will get the popup massage that show “Gesture Saved Successfully”. It will add to the system. User can check it through the ‘Scan Gestures’ option. After go to the that section you have to show your newly added gesture to your system. Then it will show relevant output.



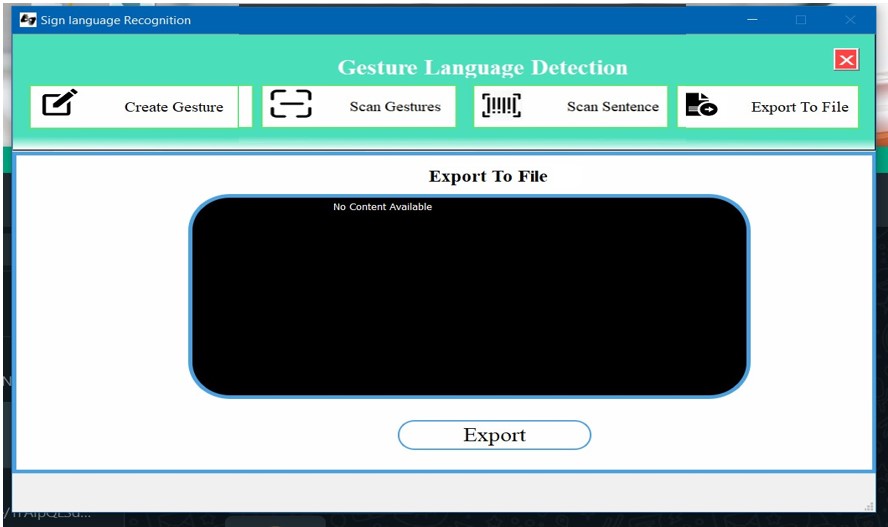
As next move, if user choose ‘Scan Sentence’ it will go to the window like as following. In their user can observe mainly four frames.



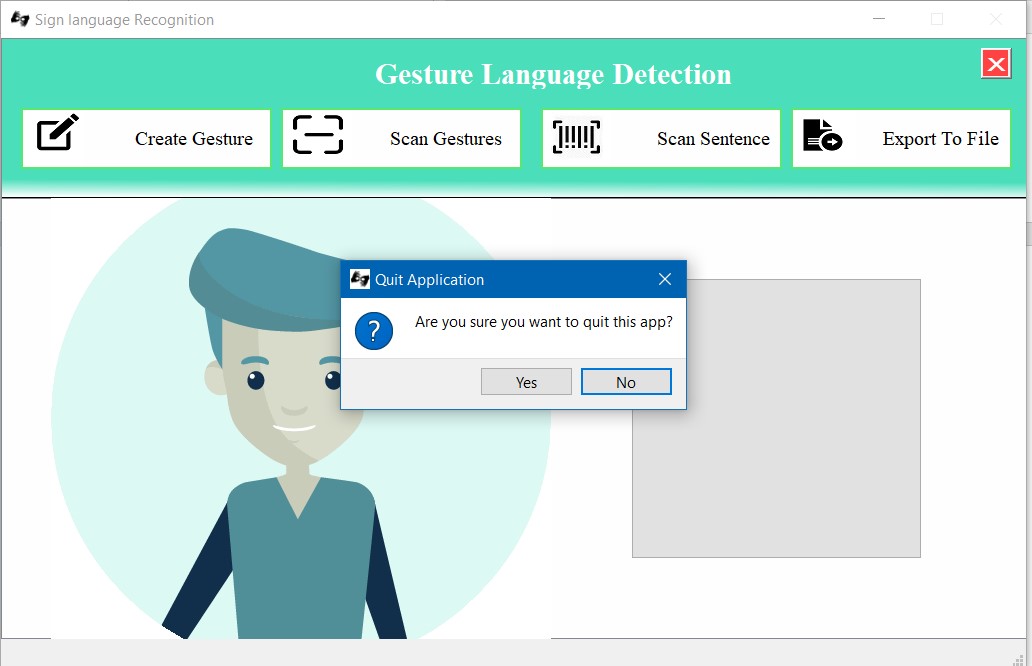
After adjusting the lightning color balancing according to the environment, you can get the user have to show relevant gesture according to the output and when the time relevant output showing user should press ‘s’. Then that gesture will add into the sentence area as a array element. After several trial user can get make the relevant output and should have to press ‘Save’ button. Then that sentence will be saved as text document. If user press the ‘Stop Scan’ button this scanning process will be terminated. Then as final step user can choose ‘Export to File’. It will export the text file which is saved in previous step. Its interface can be observed as follow.



After pressing the ‘Export’ button It export the saved text file and do the ‘Text to Speech’ process. For that it uses the pytx3 library. So, it can can used in offline too. It give voice output of the relevant text file. But it is given voice output for one time onetime only. Second time if user press the ‘Export’, Then It show and give voice output as ‘No Content available’.



If user want to exit from the whole program, must click the cross mark on the top right corner. Then following massage popup and choose ‘yes’ as answer.



# CHAPTER 6

**CONCLUSION**

**Conclusion Summery**

(Please rewrite)

From this project, have tried to give some solutions to some of the difficulties faced by persons who have speaking disabilities. The reason for that is they can’t give their feeling more freely. So that the other side of the receiver can’t able get the exact massage. So, this application is for people who want to talk with people who don’t know the universal sign language. From this person can quickly adapt gestures to produce their output. Also, if their no other gesture for a specific character users can customize it by themselves. They can quickly show relevant characters through the screen. I used Keras API, TensorFlow for implementation. Front-end designed by using pyQt5.

An export to file is designed to provide Text-To-Speech assistance. So whatever the sentence was formed letters, will be able to listen to it.

**Outlook of possible extensions**

It can be integrated with different search engines. Also can use the texting applications such as Google, WhatsApp audio massages, and Cortana. So that even illiterate people could be able to chat with other people, search for things on the web just with the help of gestures. This project is working on the image but it can be developed to detect the motion of video.

**Appendices**

A screenshot of a computer

Description automatically generated with medium confidence