Internship Project
On
Computer Vision Solution for Hearing-Impaired

Document Name	High Level Design(HLD) *Computer Vision Solution for Hearing-Impaired*
Version No.	1.0
Release Date	3/24/2022
Document ID	S/ CV/1.0/HLD

HLD History

Ver. No	Release Date	Created By / Modified By	Reviewed By and Date	Approved By and Date	Remarks and Changes Made
1.0	3/24/2022	Niranjan Shrestha, Prajin Bajracharya			

Table of Contents

Abstract

Sign language is a way of communication by people suffering from hearing loss. Around 360 million people globally suffer from disabling hearing loss out of which 328 million are adults and 32 million children. Thus, with increasing number of people with deafness, there is also a rise in demand for translators. Minimizing the communication gap between hearing impaired and normal people becomes a necessity to ensure effective communication among all. To develop a communication approach for healthy people to understand sign language used by hearing impaired people with ease, in this paper we propose a method of converting the cue symbols (ASL Gestures) to speech. We are in the verge of developing an application named "Computer Vision Solutions for Hearing Impaired" which is a project intended for helping every person to learn and understand sign language to make conversation easier and understandable for people. Although there are lots of application that can convert text to speech but only few places to learn sign language. Sign language conversion is a technique used for converting the hand gestures into their respective voice or text message. Different software can be used for this type of conversion like MATLAB, LABVIEW, C programming etc. But we used Python 3.0 as our core language for programming. We are using tensor flow as a machine learning library to process the sign language images as well as their meaning and for the detection of the sign language as input and automated speech as output.. To sum up, this application is helpful for each and every person willing to have conversation with the disabled person, this project is solely for mute person to converse with normal person using the modern technology to convert their sign language to automated speech.

Keywords: Sign language, tensor flow, disability, Python 3.0

1. Introduction

This document will be used for documenting High-level designs of project.

1.1 Purpose of the Document

The purpose of this plan is to

- Identify different design approaches
- Identify core modules/sub-systems of the system and sub-system boundary.
- Identify the best suitable technology for various sub-systems.
- Identify areas that need R&D
- Identify third party components required in the system.
- Identify components, state, life cycle and communication mechanisms between different sub-systems and also identify the external interface
- Identify various usage scenarios

1.2 Objective of HLD

1. To provide an overview of the entire system.
2. To provide a module-wise breakup of the entire system.
3. To provide introduction and high level working of every module involved.

1.3 Scope of HLD

The HLD documentation presents the structure of the system, such as the application architecture (layers), application flow (Navigation), and technology architecture. The HLD uses non-technical to mildly-technical terms which should be understandable to the administrators of the system.

1.4 Definitions

Abbreviatons	Definitions
UI	User Interface
ML	Machine Learning
ROI	Region of Interest
SL	Sign Language
ASL	American Sign Language
CNN	Convolutional Neural Network
API	Application Program Interface

2. System Overview

2.1 Product Prospective

The Computer Vision Solutions for Hearing-Impaired project is based on the classification based Machine Learning algorithms and also the application of Neural Network.

2.2 Problem Statement

Many people around the globe communicate with the help of sign language. In order to communicate with each other, people who cannot speak or hear generally use sign language. So, they usually need a person to act as interpreter in order to convey their messages to normal people.

All the disabled people may not get opportunity to learn sign language or even if they get to learn they may not be able to memorize it. Sometimes it creates difficulties in communication and learning it may also consume time. Communication between normal and handicapped person often leads to miscommunication. Only the person who have learnt about the sign language can freely converse with the handicapped person. The number of normal person who actually know anything about sign languages and how to use them only amounts to the minority of the population throughout the country.

Our project plays a vital role in solving the problem of communication. People who can hear but cannot speak can take great advantage from it. It makes the communication much easier since we can actually hear what a person who cannot speak is trying to say. People who are having a hard time understanding the signs can take a big sigh of relief.

2.3 Proposed Solution

The solution here is a Classification based Machine Learning model. It can be implemented by using most popular Classification algorithm using Deep Neural Network such as CNN(Convolutional Neural Network). Here, First we are performing Dataset Preparation and Preprocessing Step in which the data is captured, made into proper dataset and feed through the respective CNN model and thus model predicts the desired outputs.

2.4 Technical Requirements

In this Project the requirements to check to predict the appropriate hand signs for a particular information according to the prepared dataset. For that, in this project we have used different technologies. Here is some requirements for this project.

- Model should be exposed through API or User Interface, so that anyone can test model.
- Model should be able to use simple web cam for predicting hand signs for interpreting information.

2.5 Data Requirements

Data Requirement is most necessary part of the project. As data must be preprocessed using various methods of OpenCV which are listed below:

- Transformation
- Smoothing
- Edge Detection
- Segmentation
- Thresholding
- Contours Detection

For training and testing the model, we used the dataset that are prepared by ourselves in that it includes:

- 26-class fingerspelling recognition
- 2-extra function recognition
- 6-word class fingerspelling recognition

2.6 Tools Used

- Visual Studio Code is used as IDE.
- For various operation for real time video such as transformation, segmentation, thresholding etc. OpenCV is used.
- For dataset preparation pickle module is used.
- For preparation of Machine learning model Tensorflow and keras is used.
- Front end is development is done using HTML/CSS , Bootstrap

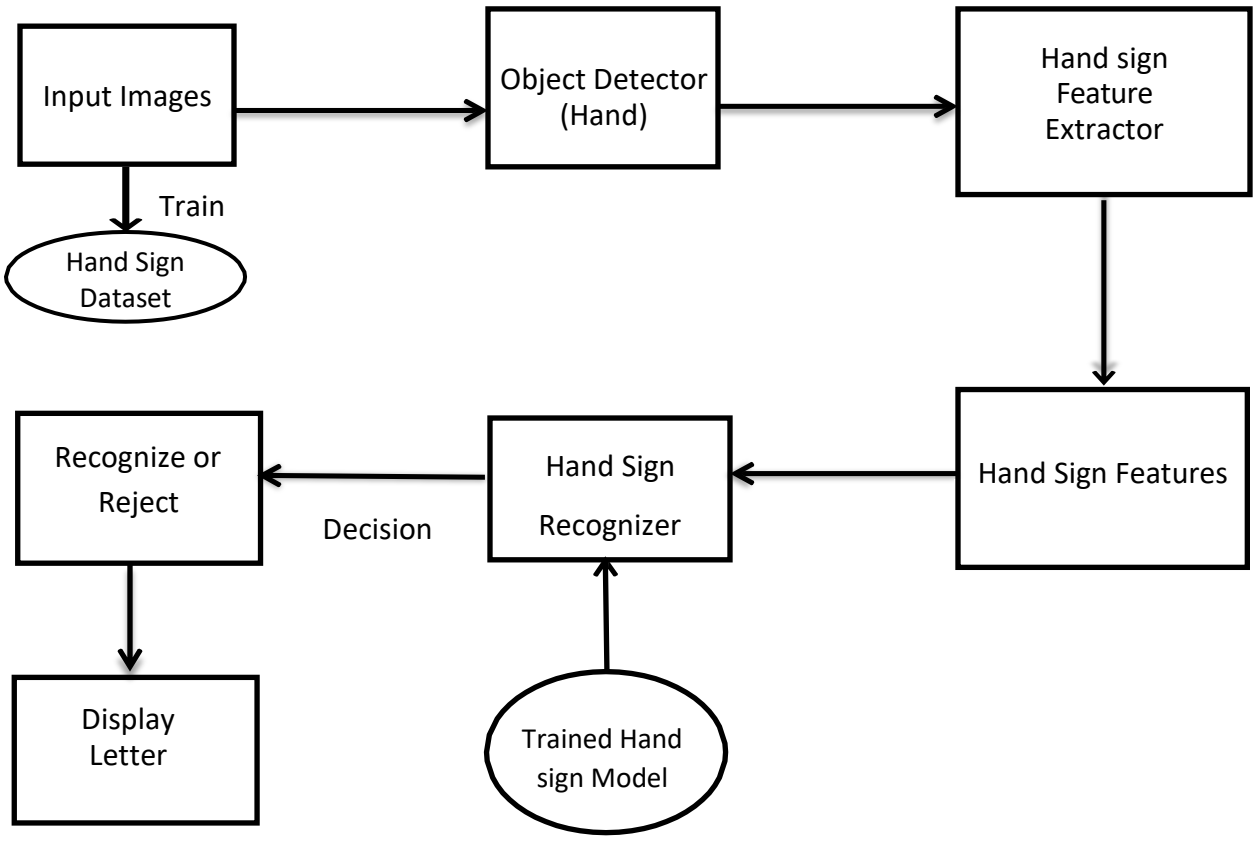
- Flask is used for backend development and for API development.
- Github is used as version control System

2.7 Constraints

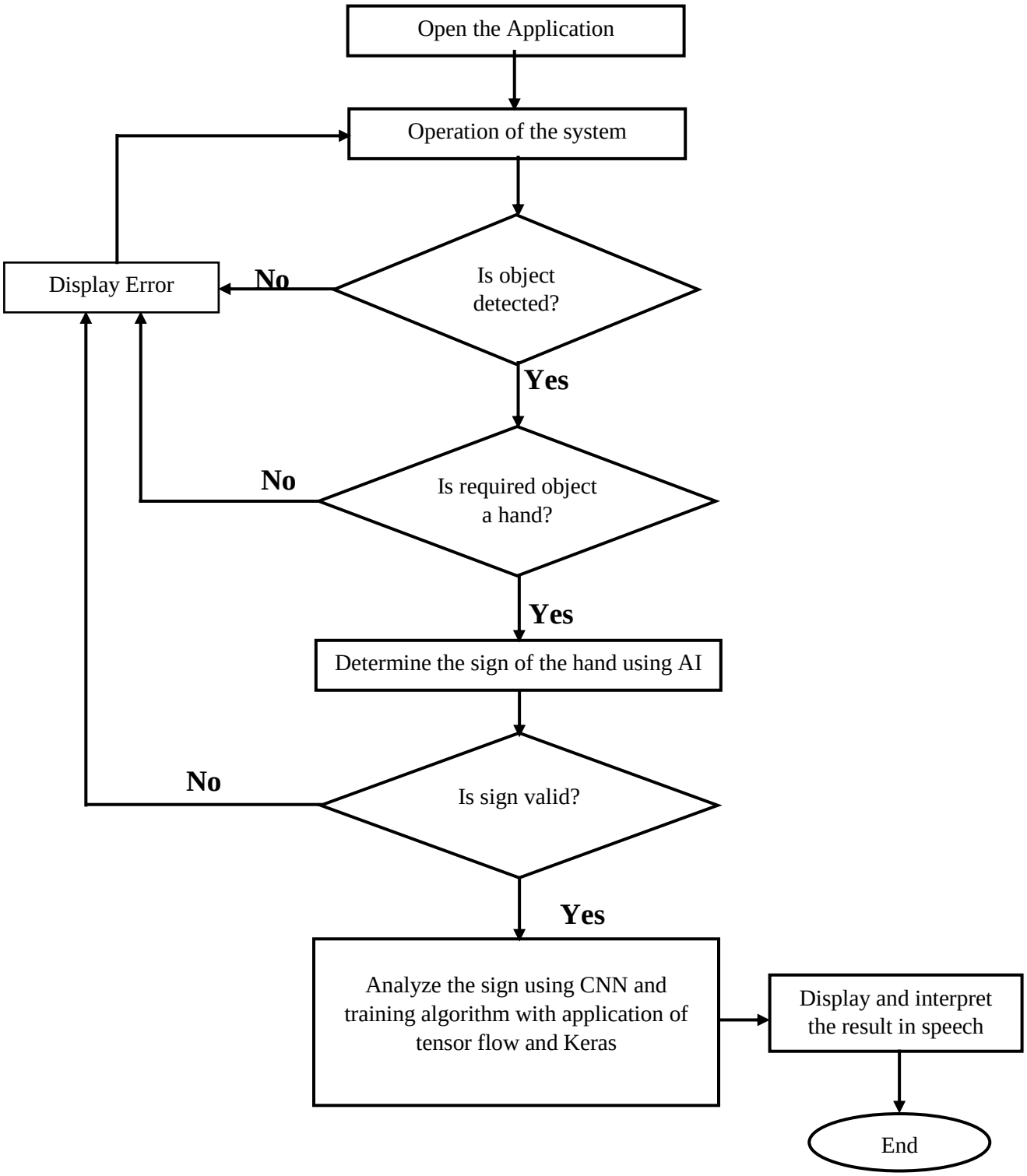
The Computer Vision Solution for hearing- Impaired must be user friendly, errors free and users should not know the deep or inner working of the project.

3. Design Flow

3.1 Process Flow



3.2 Overall Prediction Process



3.3 Event Log

In this Project we are logging every process so that user will know what process is running internally.

Step-By-Step Description:

- In this Project we defined logging for every function, class.
- By logging we can monitor every insertion, every flow of data in database.
- By logging we can monitor every step which may create problem or every step which is important in file system.
- We have designed logging in such a way that system would not hang even after so many logging's so that we can easily debug issues which may arise during process flow.
- With use of logging we can check the status of model is loaded or not

3.4 Error Handling/ Exception Handling

We have designed this project in such a way that, at any step if error occur then our application still runs instead of predicting accurate result it outputs irregular pattern or information so for there is less chance of error happening in our project and thus can be seen in the system flow diagram above.

4. Performance

Computer Vision Solution for Hearing Impaired is used to predict the different kinds of hand signs in well predicted manner such that there is possible accurate prediction.

That's why before building this model we followed complete process of Machine Learning. Here are summary of complete process:

a) First, we created the data of different hand signs using the webcam and the simple python program to capture the images of about 2000 each with the help of OpenCV.
b) Similarly different processes of data preprocessing were applied in those data as well.
c) Then after , dataset were created in the form of pickle file as (train_images, train_label, test_images, test_label, val_images and val_label) for different phases such as

training,testing and validation where the number of dataset for each categories is determined through splitting of dataset in numerical way.

d) After training the dataset in CNN algorithm, which is also regarded as best algorithm for training images as it contains so many neural network layers, we got the highest accuracy of 98%.

e) After that we applied manual search which is a type of algorithm of hyper parameter tuning, and thus found out that appropriate parameter which made bias-variance tradeoff phenomenon true.

f) On ahead, we saved the model in h5 format (default format of model) using Keras for model deployment.

g) The model was thus deployed in Web application using flask as backend and HTML, CSS and bootstrap as frontend.

4.1 Re-usability

We have done programming of this project in such a way that it would be reusable in near future thus, making anyone can add and contribute without facing any problems.

4.2 Application Compatibility

The different module of this project is using Python as an interface between them. Each modules have its own job to perform and it is the job of the Python to ensure the proper transfer of information.

4.3 Resource Utilization

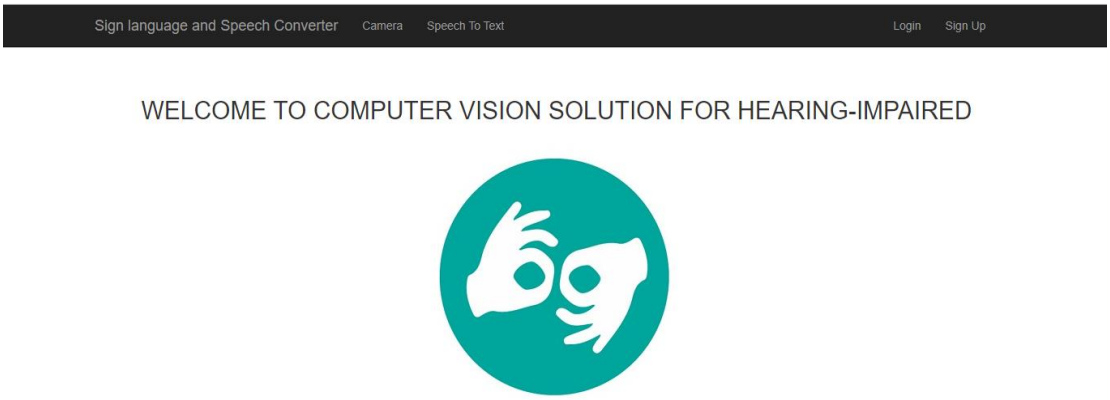
In this project, when any task is performed, it will likely that the task will use all the processing power available fin that particular system until its job finished. By keeping this in mind. In this project we have used the concept of multithreading.

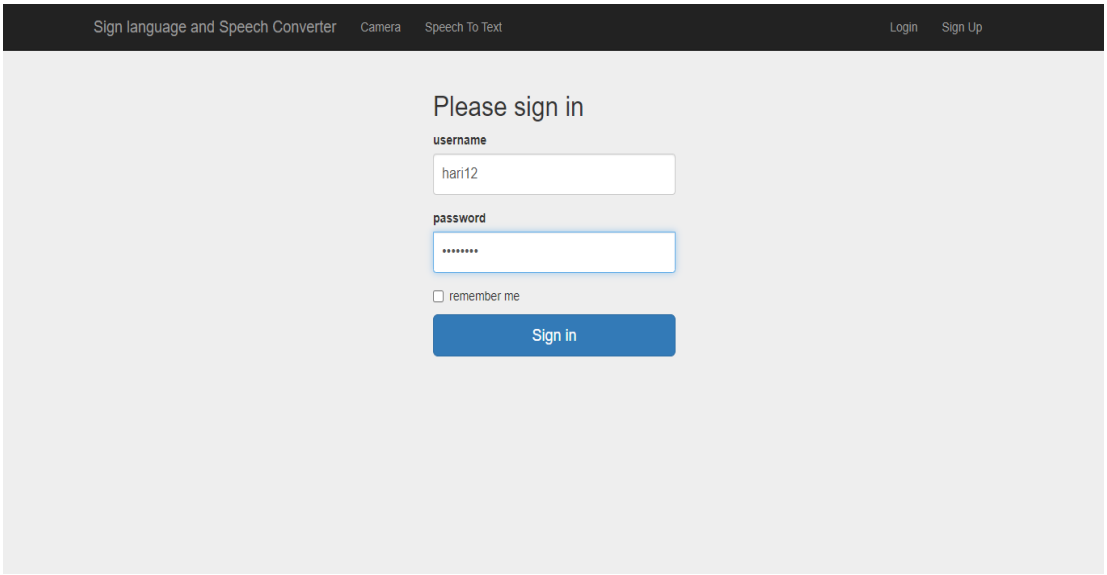
4.4 Deployment

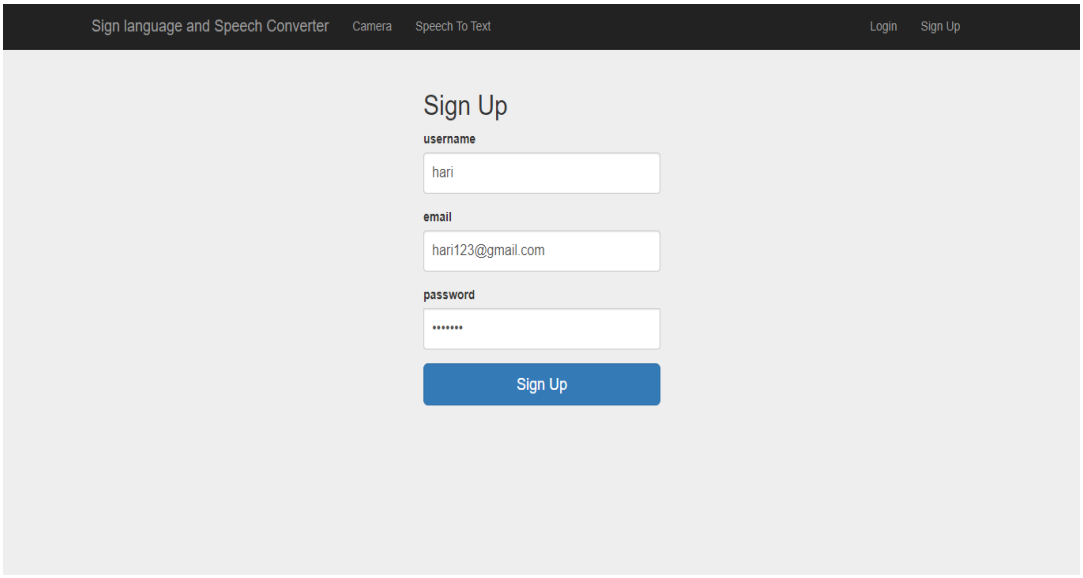
We wanted to deploy in docker or any other cloud deployment apps but our project requires lots of resources so it is little difficult to deploy in those so, due to these undeniable circumstances we deployed in local host as output with a proper justification of our system design.

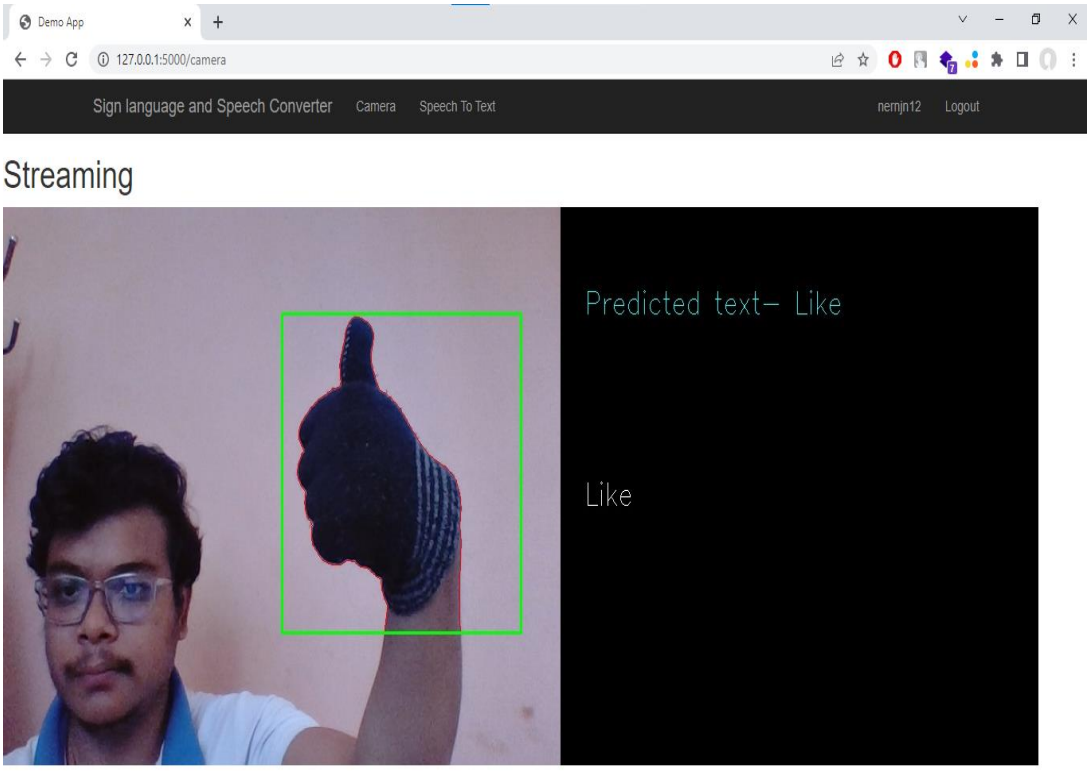
4.5 User Interface

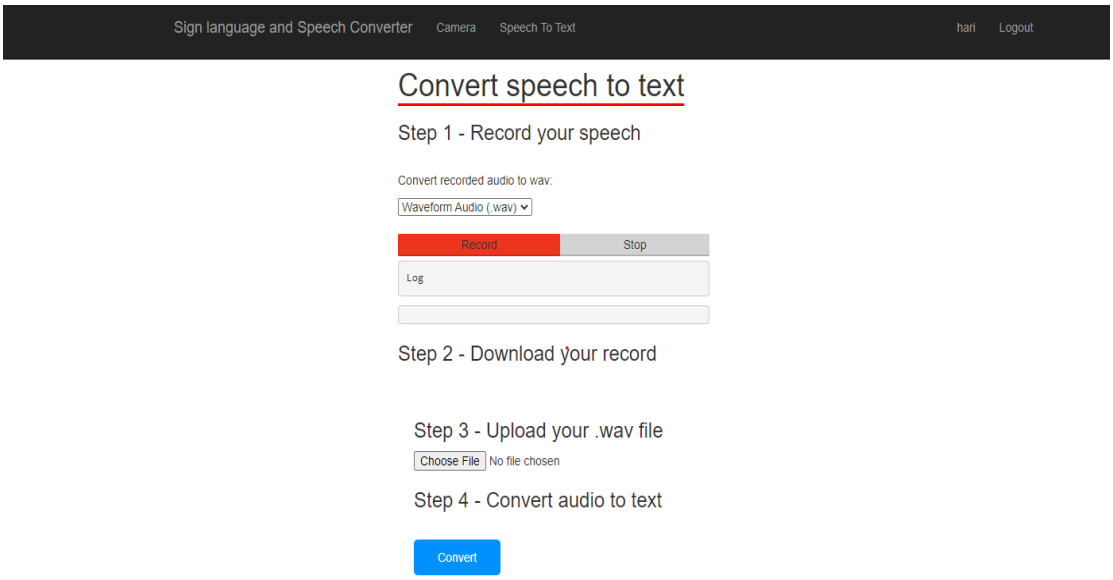
We have created a UI for user by using HTML, CSS and Bootstrap.






Sign language and Speech Converter
Camera
Speech To Text
Login
Sign Up
Sign Up
username
hari
email
hari123@gmail.com
password
Sign Up


Demo App
127.0.0.1:5000/camera
Sign language and Speech Converter
Camera
Speech To Text
nemjn12
Logout
Streaming
Predicted text- Like
Like



5. Conclusion and Future work

In this proposed approach, we have successfully designed and implemented a sign language interpretation system for American Sign Language with the help of wearable glove. This system allows translation hand signs through video feeds. The gestures performed by the user are converted into text and speech which can be easily understood by the normal people. In this project, we have implemented an automatic sign language gesture recognition system in real-time, using tools learnt in computer vision and machine learning. We learnt about how sometime basic approaches work better than complicated approaches. We also realized the time constraints and difficulties of creating and training datasets from scratch. Although our classification system works quite well as has been demonstrated through tables and images, there is still a lot of scope for possible future work. Due to long training time of our model, it became cumbersome to test various hyper parameters, architectures, image filtering etc. It became challenging to adjust the background and the light reflection during the extraction of features.

Possible extensions to this project would be extending the gesture recognition system to all alphabets of the ASL and non-alphabet gestures as well. Having used VS code and python as the platform for implementation, we feel that we can also improve upon the speed of our real-time system by proper memory management. The framework of this project can also be

extended to several other applications like mobile, controlling robot navigation using hand gestures and the like.

The system develop is effective to extract the features through the video stream and match those features in the dataset.