Real-Time Communication System Powered by AI for Specially Abled

Team ID: PNT2022TMID23065

Velammal College of Engineering and Technology

Submitted BY

Shrimeenatchi K (913119104097)

Sneha S (913119104098)

Sonal L R (913119104099)

Srinithi M (913119104104)

1.INTRODUCTION

1.1 PROJECT OVERVIEW

One of the important problems that our society faces is that people with disabilities are finding it hard to cope up with the fast growing technology. The access to communication technologies has become essential for the handicapped people. Generally deaf and dumb people use sign language for communication but they find difficulty in communicating with others who don’t understand sign language. Sign language is an expressive and natural way for communication between normal and dumb people (information majorly conveyed through the hand gesture). So, we need a translator to understand what they speak and communicate with us. The sign language translation system translates the normal sign language to speech and hence makes the communication between normal person and dumb people easier. So, the whole idea is to build a communication system that enables communications between speech-hearing impaired and a normal person.

1.2 PURPOSE

The Project's purpose is to create a system that translates sign language into a human understandable language so that ordinary people may understand it. In our society, we have people with disabilities. The technology is developing day by day but no significant developments are undertaken for the betterment of these people. Communications between deaf-mute and a normal person has always been a challenging task. It is very difficult for mute people to convey their message to normal people. Since normal people are not trained on hand sign language. In emergency times conveying their message is very difficult. The human hand has remained a popular choice to convey information in situations where other forms like speech cannot be used. Voice Conversion System with Hand Gesture Recognition and translation will be very useful to have a proper conversation between a normal person and an impaired person in any language. The project aims to develop a system that converts the sign language into a human hearing voice in the desired language to convey a message to normal people, as well as convert speech into understandable sign language for the deaf and dumb. We are making use of a convolution neural network to create a model that is trained on different hand gestures. An app is built which uses this model. This app enables deaf and dumb people to convey their information using signs which get converted to human-understandable language and speech is given as output.

2.LITRATURE SURVEY

A literature review is **a comprehensive summary of previous research on a topic**. The literature review surveys scholarly articles, books, and other sources relevant to a area of research. The review shouldenumerate, describe, summarize, objectively evaluate and clarify this previous research.

In our project, We have taken the literature survey on IEEE papers. An intelligent communication device is developed to assist nonverbal, motor-disabled persons in the generation of writtenand spoken messages. The device is centered on knowledge base of the grammatical rules and message elements. A belief reasoning scheme based on both the information from external sources and the embedded knowledge issued to optimize the process of message search.

2.1 EXISTING PROBLEM

Some of the existing solutions for solving this problem are:

Communications between deaf-mute and a normal person has always been a challenging task. It is very difficult for mute people to convey their message to normal people.

Technology

One of the easiest ways to communicate is through technology such as a phone or laptop. A deaf person can type out what they want to say and a person who is blind or has low vision can use a screen reader to read the text out loud.

A blind person can alsousevoice recognition softwareto convert what they are saying in to text so that a person who is Deaf can then read it.

Interpreter

If a sign language interpreter is available, this facilitates easycommunication if the person who is deaf is fluent in sign language. The deaf person and person who is blind can communicate with each other via the interpreter. The deaf person can use sign language and the interpreter can speak what has been said to the personwho is blind and then translate anything spoken by the blindperson into sign language for the deaf person.

However, this is often not the most effective form of communication, as it is very dependent on the individual circumstances of both people and their environment (for example, some places may have too much background noise).

2.2 REFERENCES

[1] W. Jingqiu and Z. Ting, “An ARM-based embedded gesture recognition system using a data glove,” presented at the 26th Chinese Control Decision Conf., Changsha, China, May/Jun. 2014.

[2] A. Z. Shukor, M. F. Miskon, M. H. Jamaluddin, F. A. Ibrahim, M. F. Asyraf, and M. B. Bahar, “A new data glove approach for Malaysian sign language detection,” Procedia Comput. Sci., vol. 76, no. 1, pp. 60–67, Dec. 2015.

[3] N. Sriram and M. Nithiyanandham, “A hand gesture recognition based communication system for silent speakers,” presented at the Int. Conf. Hum. Comput. Interact., Chennai, India, Aug. 2013.

[4] S. V. Matiwade and M. R. Dixit, “Electronic support system for deaf and dumb to interpret sign language of communication,” Int. J. Innov. Res. Sci., Eng. Technol., vol. 5, no. 5, pp. 8683–8689, May 2016.

[5] S. Goyal, I. Sharma, and S. Sharma, “Sign language recognition system for deaf and dumb people,” Int. J. Eng. Res. Technol., vol. 2, no. 4, pp. 382–387, Apr. 2013.

[6] S. P. More and A. Sattar, “Hand gesture recognition system using image processing,” presented at the Int. Conf. Elect., Electron., Opt. Techn., Chennai, India, Mar. 2016.

[7] K. Murakami and H. Taguchi, “Gesture recognition using recurrent neural networks,” in Proc. SIGCHI Conf. Hum. Factors Comput. Syst., New York, NY, USA, 1991, pp. 237–242.

[8] P. R. V. Chowdary, M. N. Babu, T. V. Subbareddy, B. M. Reddy, and V. Elamaran, “Image processing algorithms for gesture recognition using MATLAB,” presented at the Int. Conf. Adv. Commun. Control Comput. Technol., Ramanathapuram, India, Jan. 2015.

[9] T. Khan and A. H. Pathan, “Hand gesture recognition based on digital image processing using MATLAB,” Int. J. Sci. Eng. Res., vol. 6, no. 9, pp. 338–346, Sep. 2015.

[10] J. Siby, H. Kader, and J. Jose, “Hand gesture recognition,” Int. J. Innov. Technol. Res., vol. 3, no. 2, pp. 1946–1949, Mar. 2015.

[11] L. Lamberti and F. Camastra, “Real-time hand gesture recognition using a color glove,”

presented at the 5 ITM Web of Conferences 32, 02003 (2020) https://doi.org/10.1051/itmconf/20203202003 ICACC-2020 Int. Conf. Image Anal. Process., Ravenna, Italy, Sep. 2011.

[12] Y. Iwai, K. Watanabe, Y. Yagi, and M. Yachida, “Gesture recognition using colored gloves,” in Proc. 13th Int. Conf. Pattern Recognit., Vienna, Austria, Aug. 1996, pp. 662–666.

[13] S. P. Dawane and H. G. A. Sayyed, “Hand gesture recognition for deaf and dumb people using GSM module,” Int. J. Sci. Res., vol. 6, no. 5, pp. 2226–2230, May 2017.

[14] C. Preetham, G. Ramakrishnan, S. Kumar, and A. Tamse, “Hand talkimplementation of a gesture recognizing glove,” presented at the Texas Instrum. India Edu. Conf., Bengaluru, India, Apr. 2013.

2.3 PROBLEM STATEMENT DEFINITION

Need for real-time communication system for Specially abled:

There are handicapped people in our society. Although technology is constantly evolving, little is being done to improve the lives of these people. It has always been difficult to communicate with someone who is deaf-mute. It is quite challenging for silent persons to communicate with non-mute people. because hand sign language is not taught to the general public. It might be quite challenging for them to communicate at times of crisis. In circumstances where other modes of communication, like speech, are not possible, the human hand has remained a common alternative for information transmission.

Our Plan:

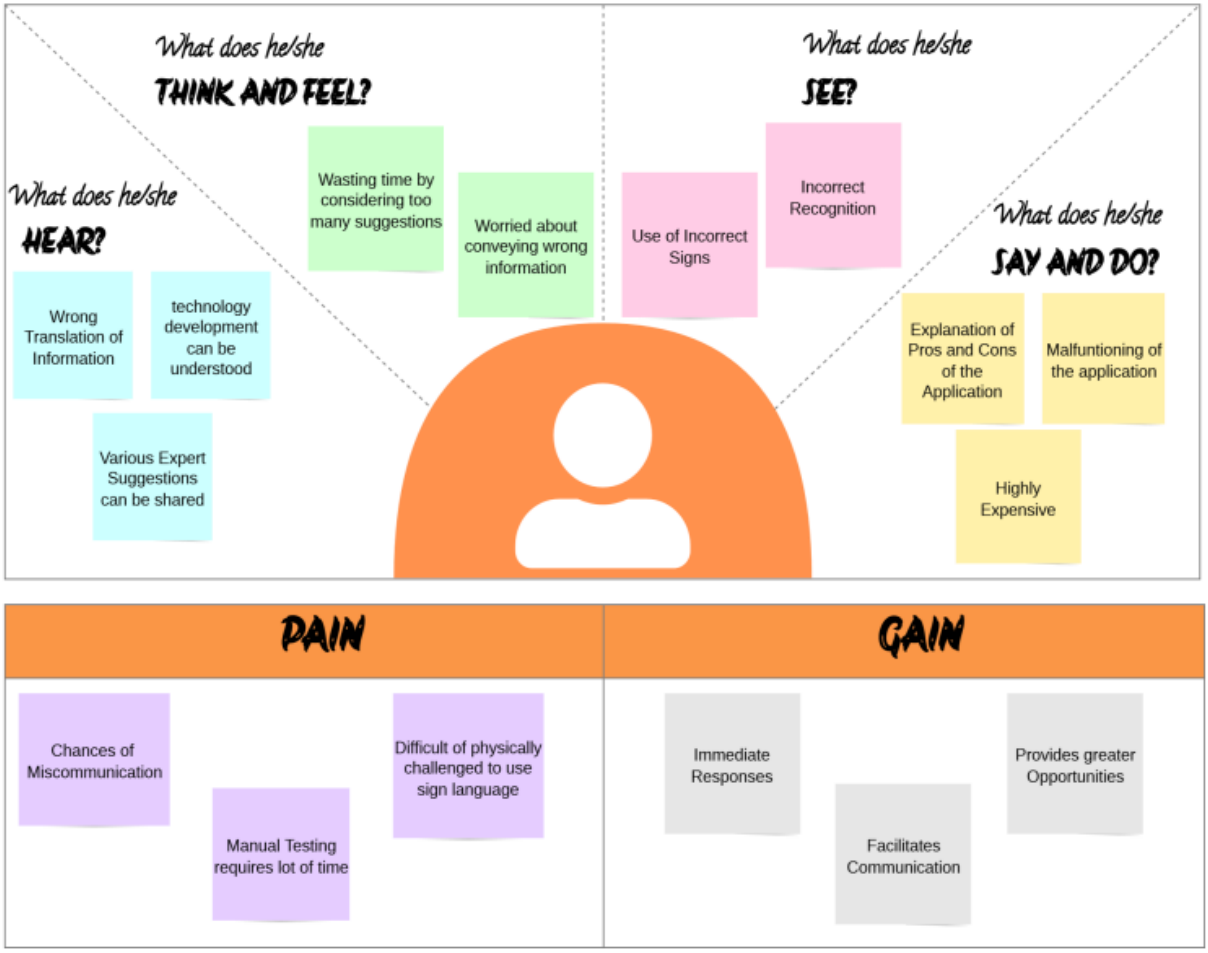
The project intends to create a system that can translate speech into understandable sign language for the deaf and dumb as well as translate sign language into a human hearing voice in the desired language to communicate a message to normal people.

Abstract:

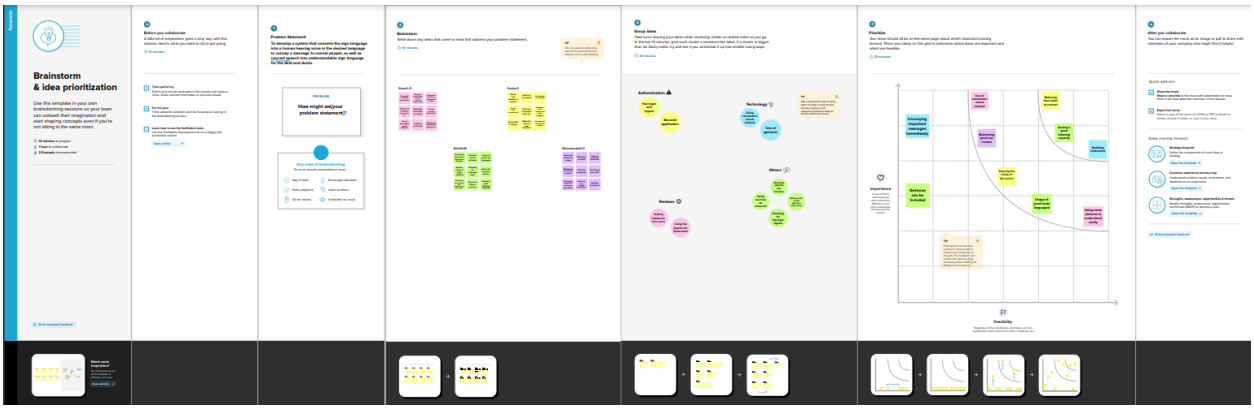
A convolution neural network is being used to build a model that is trained on various hand motions. On the basis of this model, an app is created. With the help of this app, persons who are deaf or dumb can communicate using signs that are translated into speech and human-understandable words. To have a proper communication between a normal person and a handicapped person in any language, a voice conversion system with hand gesture recognition and translation will be very helpful.

3.IDEATION AND PROPOSED SOLLUTION

3.1 EMPATHY MAP CANVAS



3.2 IDEATION AND BRAINSTORMING



4. REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENTS

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Form Registration through Gmail
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	System Requirements	Desktop with high resolution camera
FR-4	Authorization	The two levels of authorization are standard access level and advanced access level
FR-5	External Interface	Ethernet, Wi-Fi, USB to provide internet facility to access the resources with real time communication
FR-6	Reporting	Any issues found in the application, then automatically it will be notified to the developer

4.2 NON-FUNCTIONAL REQUIREMENTS

Following are the non-functional requirements of the proposed solution.

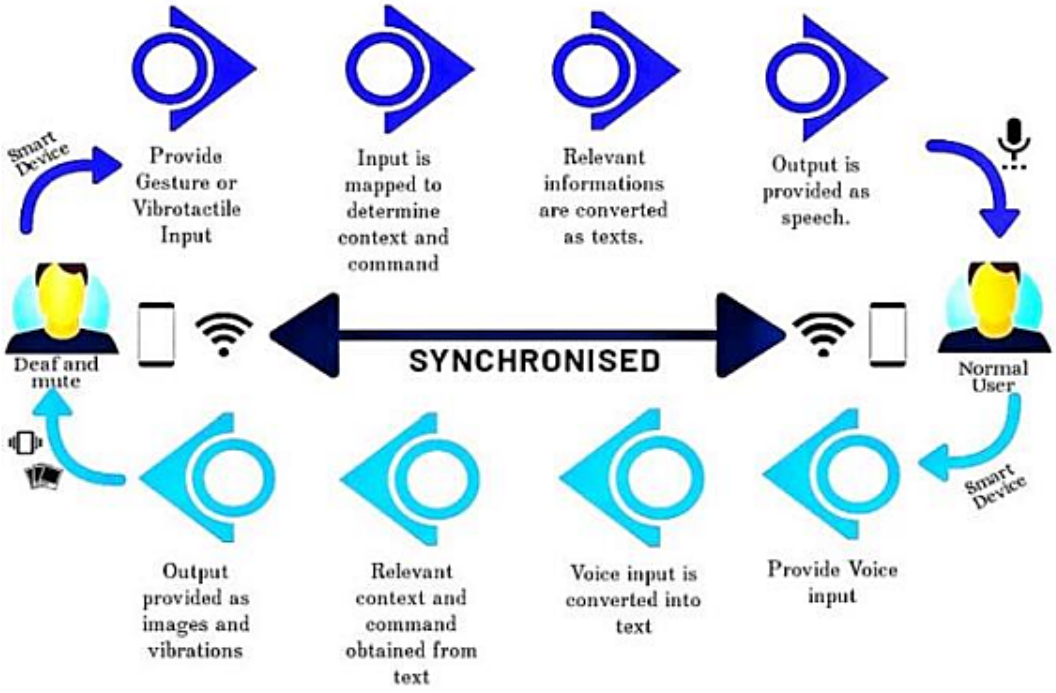
FR No.	Non-Functional Requirement	Description
NFR-1	Usability	To communicate a message to the normal people and to translate speech into sign language that the deaf and the dumb may understand.
NFR-2	Security	Only users have access to material that has been translated from signs into voice
NFR-3	Reliability	Provides information about possible problems with desktop apps on controlled devices.
NFR-4	Performance	For real-time communication, the process of translating signals into voice needs to happen more quickly
NFR-5	Availability	Provides as much spontaneous recuperation as feasible.
NFR-6	Scalability	AI technology helps disabled people to open up new opportunities for accessibility inclusion in society and independent living.

5.PROJECT DESIGN

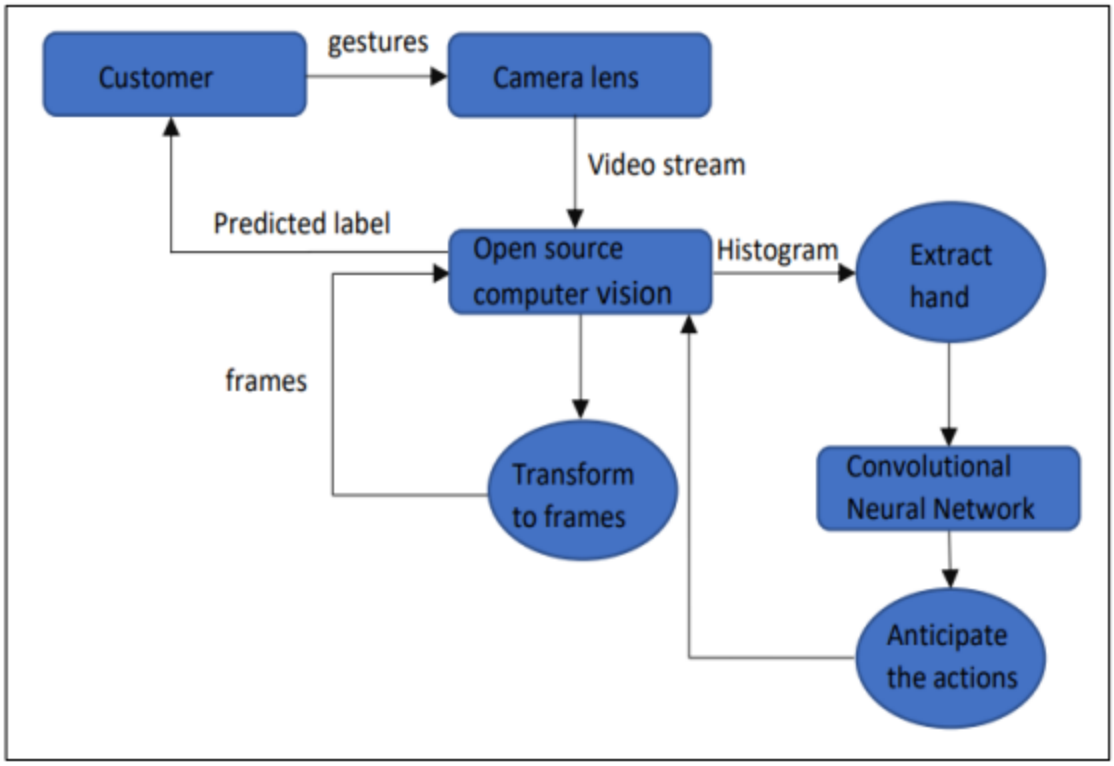
5.1 DATA FLOW DIAGRAMS

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.

Example: (Simplified)



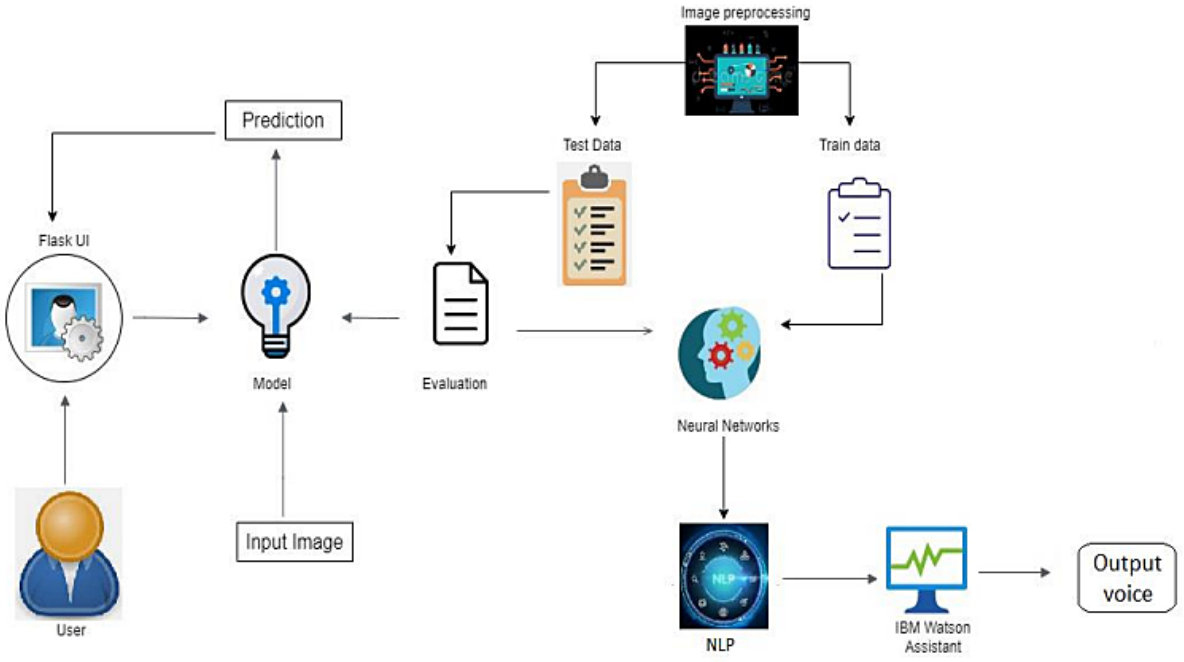
Example: DFD Level 0 (Industry Standard)



5.2 SOLUTION & TECHNICAL ARCHITECTURE

SOLUTION ARCHITECTURE

The proposed solution will help convert the sign language into a human hearing voice in the desired language to convey a message to normal people, as well as convert speech into understandable sign language for the deaf and dumb.



The architecture of the propose system consists of various blocks:

IMAGE DETECTION:

This is the step that comes right after camera capture. Image detection refers to detecting the image that is obtained and, in this case, it is found out if the obtained image is that of a hand or not. A binary classifier is to be trained beforehand to check the same. A binary classifier has the task of classifying sets into two groups, depending on the criteria the sets meet. It checks for one or more qualities that a particular set should possess. It is according to that factor that a binary classifier decided to which group the set should be sent to.

IMAGE RECOGNITION:

Image recognition is the most crucial procedure of this project. The acquired image is converted to its vector form. The model used is SVM, support vector machine. Support Vector Machine is used to analyze data and classify them. Support Vector Machine comes under supervised machine learning. SVM represents the examples as points in space that are mapped so that they are separated according to the category they come under.

EVALUATION:

To evaluate the system, both a quantitative and a qualitative evaluation should be performed. One of the major difficulties in performing the evaluation, is that it takes more time for the disabled students to do their course work. Asking them to spend time on the evaluation of a prototype adds a load to their already busy schedule. Nevertheless, as discussed later, a more complete evaluation with several disabled and non-disablity post-secondary students is planned for.

IMAGE PREPROCESSING:

Image processing is used to convert an image into digital form and perform certain operations on it to obtain an improved image or extract useful information from it. Preprocessing refers to all the transformations on the raw data before it is fed to the machine learning or deep learning algorithm. For instance, training a convolutional neural network on raw images will probably lead to bad classification performances.

NLP (NATURAL LANGUAGE PROCESSING):

Natural language processing refers to the branch of computer science and more specifically, the branch of artificial intelligence or AI, concerned with giving computers the ability to understand text and spoken words in much the same way human beings can. Natural language processing helps computers communicate with humans in their own language and scales other language-related tasks.

TECHNICAL ARCHITECTURE

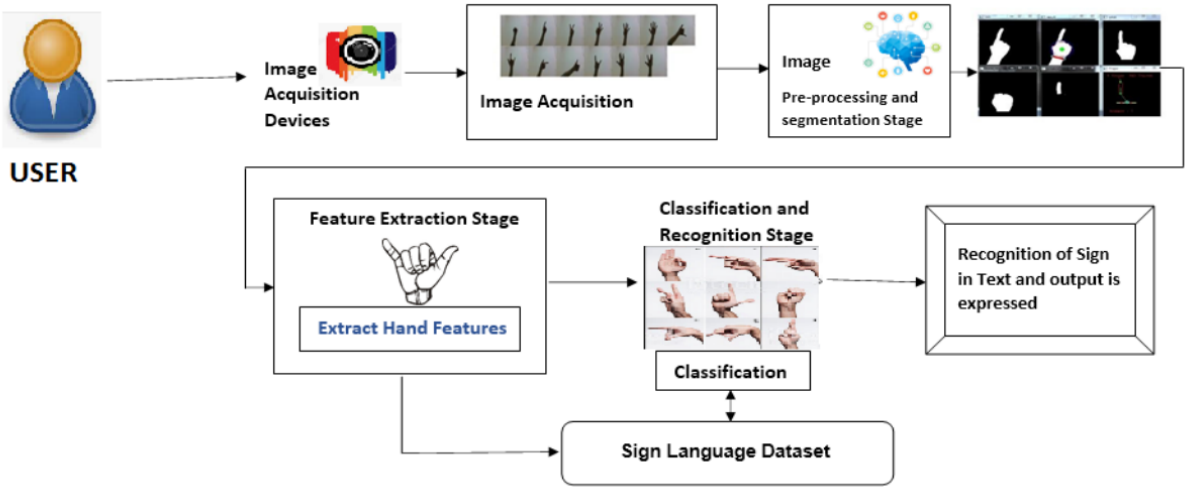


Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	The customer must log in using the appropriate website or phone number. The user interface will then be used in interaction.	javascript, CSS,HTML
2.	Application Logic-1	To develop the project, many sorts of libraries and frameworks are needed.	Java / Python
3.	Application Logic-2	Aids in translating verbal expression from human gestures and actions.	Machine learning
4.	Application Logic-3	Recognises the human gestures and then offers useful, realistic solutions	ANN,CNN
5.	Database	Data could consist of words or numbers.	MySQL, Rational database
6.	Cloud Database	Giving customers access to host databases without requiring them to purchase additional hard ware	Deep learning and neural networks
7.	File Storage	Fast, dependable, and adaptable file storage are all possible..	Local file system

8.	External API-1	Used to access cloud-based information.	Weather API
9.	External API-2	Used to get information so you may make data-driven decisions	Aadhar API
10.	Machine Learning Model	A variety of algorithms that are necessary for implementation interact with machine learning.	Image acquisation
11.	Infrastructure (Server / Cloud)	Implementing an application on a local system or setting up a local cloud server. Run the installer after installing the Windows version.	Local, Cloud Foundry, Kubernetes, etc.

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	The framework that is employed.	Tensor flow, Theano, RNN, PyTorch
2.	Security Implementations	Security measures that a firewall can implement.	Firewall and some security related softwares..
3.	Scalable Architecture	The design will be expandable (Micro services).	Data, models, speed and consistency..
4.	Availability	The application's accessibility (use of load balancers, distributed servers etc)	Image recognition, sign/gestures recognition, text recognition & real time captioning..
5.	Performance	Design considerations for an application's performance (the number of requests made per second, the utilisation of caching, etc.)	Using Convolutional neural network, maching learning for conversation and improve the sensivity of the performance..

5.3 USER STORIES

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming the password.	I can access my account / dashboard	High	Sprint-1
		USN-2	As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High	Sprint-1
		USN-3	As a user, I can register for the application through Facebook	I can register & access the dashboard with Facebook Login	Low	Sprint-2
		USN-4	As a user, I can register for the application through Gmail	I can register via some third party's link	Medium	Sprint-1
	Login	USN-5	As a user, I can log into the application by entering email & password	I can type manually and also can used saved login credentials	High	Sprint-1
	Dashboard	USN-6	As a customer, I can get all	I can access my	Medium	Sprint-2

			services and help through the dashboard	dashboard and change profile		
Customer (Web user)	Registration	USN-7	As a customer, I could able to login through registered phone number by using otp instead of Gmail	I could able to register & login via phone number to access my account	High	Sprint-2
Customer Care Executive	Service	USN-8	Can avail the service by calling customer care or reaching through E-mail.	Can avail the service by calling customer care or reaching through Email.	Medium	Sprint-1
Administrator	Sign up	USN-9	Customer have to sign-up to use these things and all	Have to enter valid credentials	High	Sprint-2
	Wishlist	USN-10	Customer's desired choices to avail these services.	As a customer can review and choose their services as he want/preferred.	Medium	Sprint-1
	Enrolment	USN-11	The customer can avail all services once he/she enrolled.	As a customer, it's quite enchanting	Medium	Sprint-2

PROJECT PLANNING & SCHEDULING

6.1 SPRINT PLANNING & ESTIMATION

Product Backlog, Sprint Schedule, and Estimation

Use the below templateto create productbacklog and sprint schedule

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Data Collection	USN-1	Collect Dataset	9	High	Sneha S Shrimeenatchi K
Sprint-1		USN-2	Image pre-processing	8	Medium	Srinithi M Sonal L R
Sprint-2	Model Building	USN-3	Import the required libraries, add the necessary layers and compilethe model	10	High	Srinithi M Sneha S
Sprint-2		USN-4	Training the image classification model using CNN	7	Medium	Shrimeenatchi K, Sonal L R
Sprint-3	Training and Testing	USN-5	Training the model andtesting the model's performance	9	High	Sneha S Sonal L R
Sprint-4	Implementation of theapplication	USN-6	Converting the input signlanguage images intoEnglish alphabets	8	Medium	Shrimeenatchi K, Srinithi M

6.2 SPRINT DELIVERY SCHEDULE

Project Tracker, Velocity & Burndown Chart:

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date(Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	10	6 Days	24 Oct 2022	29 Oct 2022	8	29 Oct 2022
Sprint-2	10	6 Days	31 Oct 2022	04 Nov 2022	5	04 Nov 2022
Sprint-3	10	6 Days	07 Nov 2022	11 Nov 2022	7	11 Nov 2022
Sprint-4	10	6 Days	14 Nov 2022	18 Nov 2022	5	18 Nov 2022

VELOCITY:

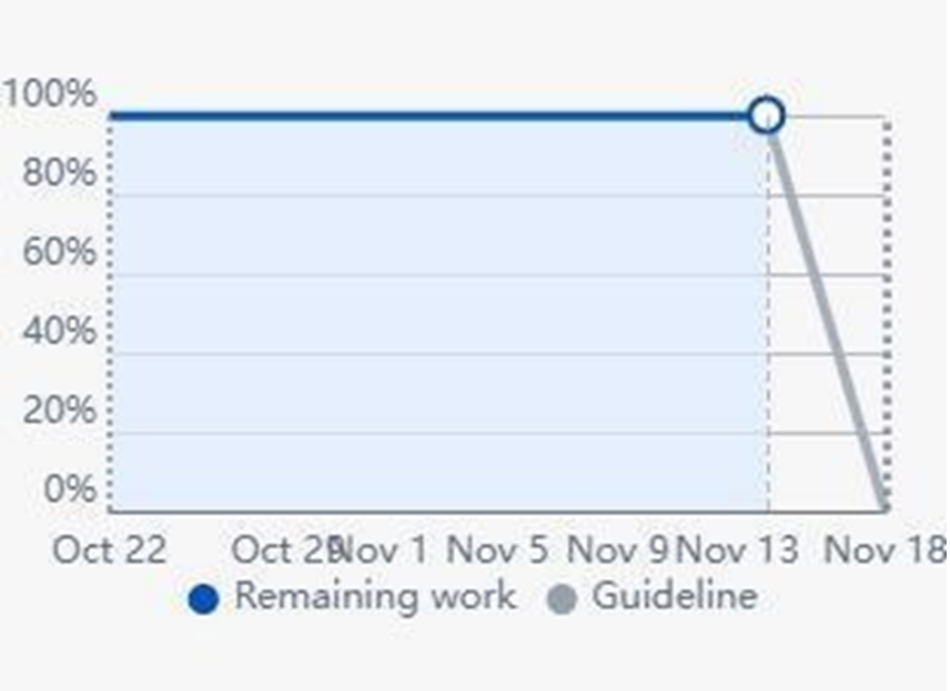
$$AV = \frac{sprint\ duration}{velocity}$$

AV = 6/10

= 0.6

Burndown chart:

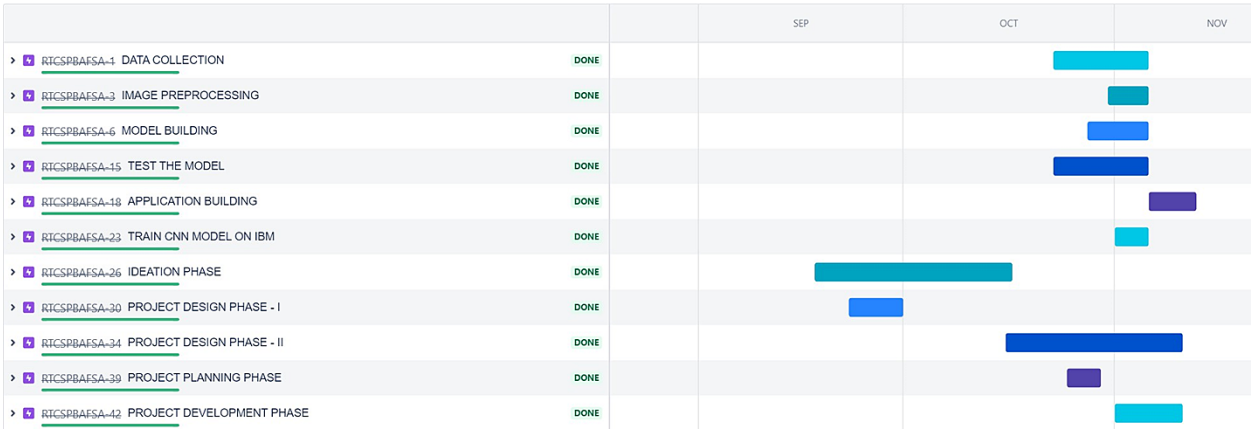
SPRINT BURNDOWNCHART:



6.3 REPORTS FROM JIRA

Weekly Progress Report

Monthly Progress Report



Quarterly ProgressReport

7.CODING & SOLUTIONING

In order to design website that coverts sign language into English alphabets, we need to develop the website. For developing the website, primarly we need a platform that is uesful for developing the code. Coding is nothing bu the applications developed by the developers in a certain computer language. Here we are using Python language for developing the website.

7.1 FEATURE 1

Image Preprocessing

In [1]:
```
from tensorflow.keras.preprocessing.image import ImageDataGenerator
```

In [2]:
```
train_datagen=ImageDataGenerator(rescale=1./255,horizontal_flip=True,vertical_flip=True,zoom_range=0.2)
```

In [3]:
```
test_datagen=ImageDataGenerator(rescale=1./255)
```

In [6]:
```
x_train = train_datagen.flow_from_directory(r"/content/drive/MyDrive/IBM project/training_set",target_size=(64,64),class_mode="categorical",batch_si
```

```
Found 15130 images belonging to 9 classes.
```

In [7]:
```
x_test = test_datagen.flow_from_directory(r"/content/drive/MyDrive/IBM project/test_set",target_size=(64,64),class_mode="categorical",batch_size=30)
```

```
Found 1035 images belonging to 9 classes.
```

Model Building

In [8]:
```
from keras.models import Sequential
from keras.layers import Dense
from keras.layers import Convolution2D
from keras.layers import MaxPooling2D
from keras.layers import Dropout
from keras.layers import Flatten
```

In [20]:
```
model.fit(x_train,epochs=10,validation_data=x_test,steps_per_epoch=len(x_train)//10,validation_steps=len(x_test))
```

```
Epoch 1/10
50/50 [==============================] - 407s 8s/step - loss: 0.1036 - accuracy: 0.9707 - val_loss: 0.1421 - val_accuracy: 0.9362
Epoch 2/10
50/50 [==============================] - 75s 1s/step - loss: 0.0937 - accuracy: 0.9767 - val_loss: 0.0786 - val_accuracy: 0.9768
Epoch 3/10
50/50 [==============================] - 74s 1s/step - loss: 0.0833 - accuracy: 0.9740 - val_loss: 0.0380 - val_accuracy: 0.9894
Epoch 4/10
50/50 [==============================] - 65s 1s/step - loss: 0.0493 - accuracy: 0.9853 - val_loss: 0.0354 - val_accuracy: 0.9913
Epoch 5/10
50/50 [==============================] - 56s 1s/step - loss: 0.0514 - accuracy: 0.9851 - val_loss: 0.0484 - val_accuracy: 0.9913
Epoch 6/10
50/50 [==============================] - 58s 1s/step - loss: 0.0661 - accuracy: 0.9813 - val_loss: 0.0597 - val_accuracy: 0.9894
Epoch 7/10
50/50 [==============================] - 50s 1s/step - loss: 0.0488 - accuracy: 0.9872 - val_loss: 0.0888 - val_accuracy: 0.9662
Epoch 8/10
50/50 [==============================] - 48s 958ms/step - loss: 0.0492 - accuracy: 0.9820 - val_loss: 0.0670 - val_accuracy: 0.9874
Epoch 9/10
50/50 [==============================] - 35s 709ms/step - loss: 0.0599 - accuracy: 0.9820 - val_loss: 0.0129 - val_accuracy: 0.9971
Epoch 10/10
50/50 [==============================] - 39s 770ms/step - loss: 0.0582 - accuracy: 0.9770 - val_loss: 0.1378 - val_accuracy: 0.9314
```

Out[20]:

In [21]:
```
model.save("aslpng.h5")
```

7.2 FEATURE

Testing the model

In [22]:
```
from keras.models import load_model
import numpy as np
import cv2
```

In [23]:
```
from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing import image
import numpy as np
```

In [34]:
```
model=load_model("aslpng.h5")
img = image.load_img(r"/content/drive/MyDrive/IBM project/test_set/D/10.png",target_size=(64,64))
img
```

Out[34]:

In [35]:
```
x = image.img_to_array(img)
x
```

Out[35]:
```
array([[[0., 0., 0.],
        [0., 0., 0.],
        [0., 0., 0.],
        ...,
        [0., 0., 0.],
        [0., 0., 0.],
        [0., 0., 0.]],

       [[0., 0., 0.],
        [0., 0., 0.],
        [0., 0., 0.],
        ...,
        [0., 0., 0.],
        [0., 0., 0.],
        [0., 0., 0.]],

       [[0., 0., 0.],
        [0., 0., 0.],
        [0., 0., 0.],
        ...,
        [0., 0., 0.],
        [0., 0., 0.],
        [0., 0., 0.]],

       ...,

       [[0., 0., 0.],
        [0., 0., 0.],
        [0., 0., 0.],
        ...,
        [0., 0., 0.],
        [0., 0., 0.],
        [0., 0., 0.]],

       [[0., 0., 0.],
        [0., 0., 0.],
        [0., 0., 0.],
        ...,
        [0., 0., 0.],
        [0., 0., 0.],
        [0., 0., 0.]],

       [[0., 0., 0.],
        [0., 0., 0.],
        [0., 0., 0.],
        ...,
        [0., 0., 0.],
        [0., 0., 0.],
        [0., 0., 0.]]], dtype=float32)
```

In [36]:
```
x.shape
```

Out[36]:
```
(64, 64, 3)
```

In [37]:
```
x = np.expand_dims(x,axis=0)
x.shape
```

Out[37]:
```
(1, 64, 64, 3)
```

In [38]:
```
pred = model.predict(x)
```

```
1/1 [==============================] - 0s 63ms/step
```

In [39]:
```
pred
```

Out[39]:
```
array([[0., 0., 0., 1., 0., 0., 0., 0., 0.]], dtype=float32)
```

In [45]:
```
class_name=["A","B","C","D","E","F","G","H","I"]
pred_id = pred.argmax(axis=1)[0]
pred_id
```

Out[45]:
```
3
```

8.TESTING

8.1 TEST CASES

				Date	12-Nov-22							
				Team ID	PNT2022TMID01158							
				Project Name	Project - Real time communication system powered by AI for specially abled							
				Maximum Marks	4 marks							
Test case ID	Feature Type	Component	Test Scenario	Pre-Requisite	Steps To Execute	Test Data	Expected Result	Actual Result	Status	Comments	TC for Automation(Y/N)	BUG ID	Executed By
Login Page_TC_001	Functional	Home Page	Verify user is able to see the homepage	Mozilla Firefox Browser	Enter URL in browser and click go	http://127.0.0.1:5001	Homepage should be displayed	Working as expected	Pass	Steps are clear to follow	NO	NA	SHALINI A ,NAGA NANDHINI ,NIVEDHA ,MONIKA PRABHA S
Login Page_TC_002	UI	Home Page	Verify the UI elements in homepage	Mozilla Firefox Browser	1.Enter URL and click go 2.Verify homepage with given UI elements: Reference,camera access display,Introduction to project	http://127.0.0.1:5001	Application should show below UI elements: a Reference b camera access display c Introduction to project	Working as expected	Pass	Steps are clear to follow	NO	NA	SHALINI A ,NAGA NANDHINI ,NIVEDHA ,MONIKA PRABHA S
Login Page_TC_003	UI	Home page	Verify whether reference page is working	Mozilla Firefox Browser	1.Enter URL(http://127.0.0.1:5000) and click go 2.Click on reference button	http://127.0.0.1:5002	User should navigate to reference page where all alphabet images displayed	working as expected	Pass	Steps are clear to follow	Yes	NA	SHALINI A ,NAGA NANDHINI ,NIVEDHA ,MONIKA PRABHA S
Login Page_TC_004	Functional	Home Page	Verify Camera access	Mozilla Firefox Browser,Web-Camera	1.Enter URL(http://127.0.0.1:5000) and click go 2.Click allow camera access	Allow camera access	Camera access is allowed and image is displayed	working as expected	Pass	Steps are clear to follow	Yes	NA	SHALINI A ,NAGA NANDHINI ,NIVEDHA ,MONIKA PRABHA S
Login Page_TC_004	Functional	Home Page	Gesture detection	Mozilla Firefox,CNN	1.Enter URL(http://127.0.0.1:5000) and click go 2.Click camera access 3.Image displayed 4.Detection of gesture occurs	Detection of gestures	Hand gestures needs to be detected and predicted	working as expected	Pass	Steps are clear to follow	Yes	NA	SHALINI A ,NAGA NANDHINI ,NIVEDHA ,MONIKA PRABHA S
Login Page_TC_005	Functional	Home page	Output prediction	CNN trained model	1.Enter URL(http://127.0.0.1:5000) and click go 2.Click camera access 3.Image displayed 4.Detection of gesture occurs 5.Output prediction	Predicted gestures	Hand gestures are detected and predicted ASL alphabets are displayed	working as expected	Pass	Predicted output is displayed	Yes	NA	SHALINI A ,NAGA NANDHINI ,NIVEDHA ,MONIKA PRABHA S

A test case is nothing but a seriesof step executedon a design, using a predefined set of input data, expected to produce a pre-defined set of outputs, in a given environment. It describes “how” to implementthose test cases.

8.2 USER ACCEPTANCE TESTING

User acceptance testing (UAT), also called application testing or end-user testing, is aphase of software development in which the software is tested in the real world by its intended audience.

1. **Purpose of Document**
   The purpose of this document is to briefly explain the test coverage and open issues of project-Real Time Communication System Powered By AI For Specially Abled at the time of the release to User Acceptance Testing (UAT).

2. **Defect Analysis**
   This report shows the number of resolved closed bugs at each severity level, and how they were resolved.

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	0	0	0	2	2
Duplicate	1	0	0	0	1
External	0	0	1	0	1
Fixed	0	1	1	0	2
Not Reproduced	0	1	0	0	1
Skipped	0	0	0	0	0
Won't Fix	0	1	0	0	1
Totals	1	3	2	2	8

3. Test Case

Analysis This report shows the number of test cases that have passed, failed, and untested.

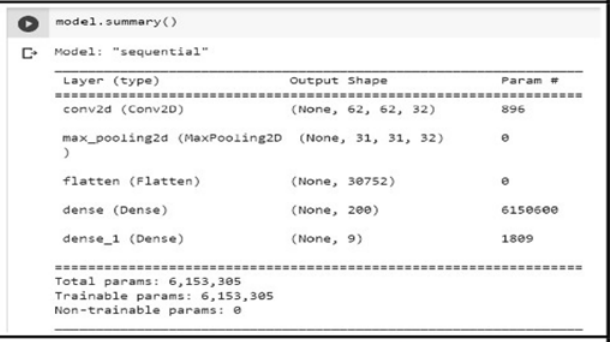
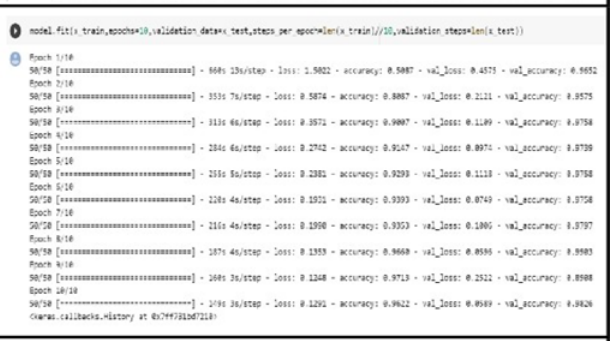
Section	Total Cases	Not Tested	Fail	Pass
View Home Page	7	0	1	6
Click Reference	15	0	3	12
Image displayed	12	0	0	12
Allow camera access	11	0	2	9
PrintEngine	8	0	0	8
ClientApplication	49	0	0	49
Security	4	0	0	4
OutsourceShipping	4	0	0	4
ExceptionReporting	11	0	0	11
FinalReportOutput	2	0	0	2
VersionControl	1	0	0	1

9.RESULTS

9.1 PEROMANCE METRICS

Model Performance Testing:

Project team shall fill the following information in model performance testing template.

S.No.	Parameter	Values	Screenshot
1.	Model Summary	Model - Sequential model Layers: Conv2D-(None,62,62,32) MaxPooling2D-(None,31,31,32) Flatten-(None,30752) Dense-(None,200) Dense_1 -(None,9)	
2.	Accuracy	Training Accuracy - 0.9622 Validation Accuracy -0.9826	
3	Confidence Score	Class Detected – N/A Confidence Score -N/A	N/A

The proposed procedure was implemented and tested with set of images. The set of 15750 images of Alphabets from “A” to “I” are used for training database and a set of 2250 images of Alphabets from “A” to “I” are used for testing database. Once the gestureis recognize the equivalent Alphabet is shown on the screen.

10.ADVANTAGES & DISADVANTAGES

Advantages:

1. Create a mobile application to bridge the communication gap between deaf and dumb persons and the general public.
2. Sign language standards exist, their datasetcan be added,and the usercanchoose which sign language to read.

Disadvantages:

1. Model only works from alphabets A to I.
2. Absence of gesture recognition, alphabets from J cannot be identified.
3. As the quantity/quality of images in the datasetis low, the accuracy is not great.

11.CONCLUSION

Through this project we are trying to build flexible system for the physically impaired people that will ease their life.An attempt to create a sign language to text conversion wireless lightweight system,through the use of information gestured by physically impaired person which can be effectively conveyed to a normal person.The system will try to convert the sign language of the physically impaired person into text that can interpreted by their own genre and also to the rest of the world.

12.FUTURE SCOPE

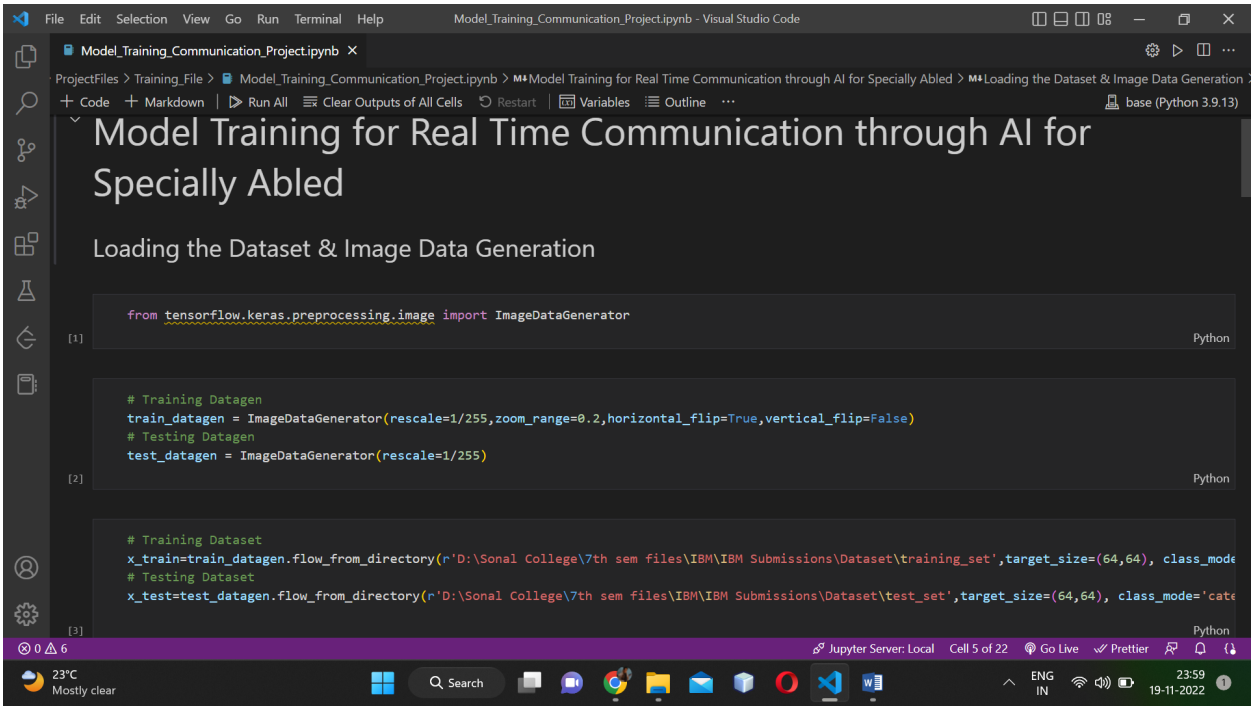
Having a technology that can translate hand sign languageto its corresponding alphabet is a game changerin the field of communication and AI for the speciallyabledpeople such as deaf and dumb. With introduction of gesture recognition, the web app can easilybe expanded to recognize letters beyond 'I', digits and other symbols plus gesture recognition can also allow controlling of software/hardware interfaces.

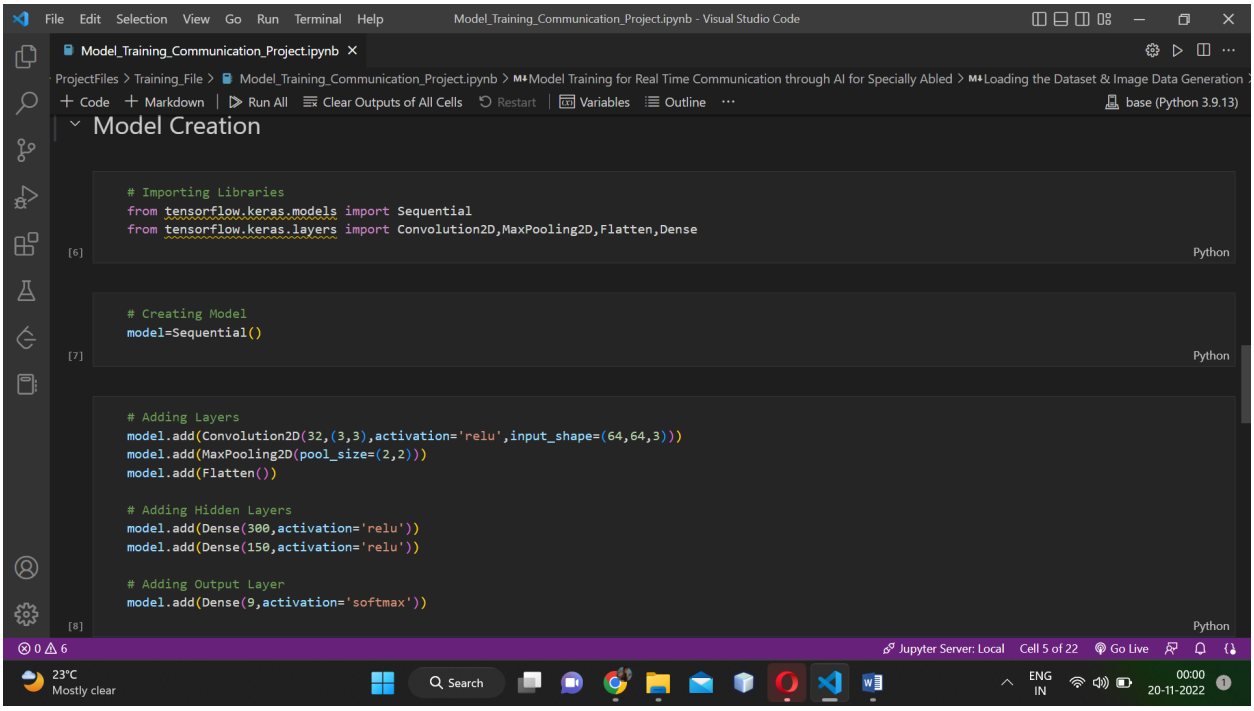
We can develop a model for ISL word and sentence level recognition. This will require a system that can detectchanges with respectto the temporal space. We can also develop a completeproduct that will help the speech and hearing-impaired people,andthereby reduce the communication gap.

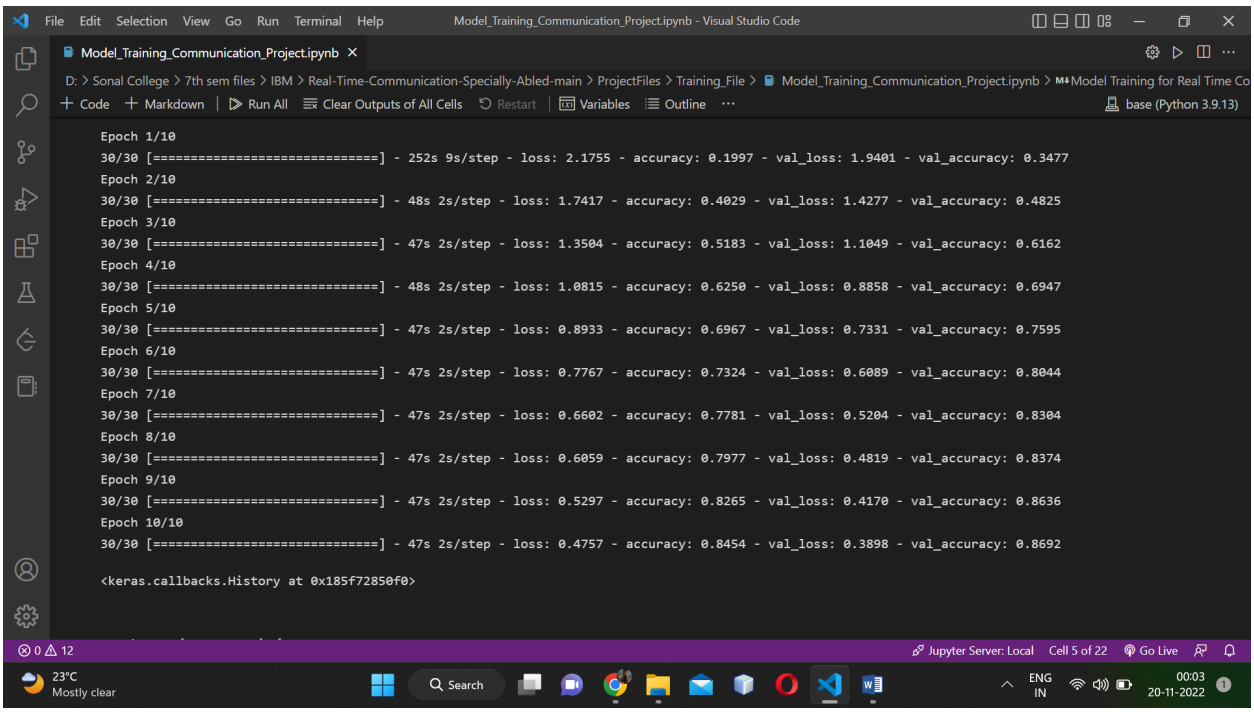
13.APPENDIX

SOURCE CODE

Model Training







Model Testing

File Edit Selection View Go Run Terminal Help Model_Testing.ipynb - Visual Studio Code

Model_Training_Communication_Project.ipynb | Model_Testing.ipynb

D: > Sonal College > 7th sem files > IBM > Real-Time-Communication-Specially-Abled-main > ProjectFiles > Training_File > Model_Testing.ipynb > Testing the model

+ Code + Markdown | Run All | Clear Outputs of All Cells | Outline ... base (Python 3.9.13)

Testing the model

```
[17] Python
import numpy as np
from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing import image
```

```
[18] Python
model=load_model('asl_model_84_54.h5')
img=image.load_img(r'D:\Sonal College\7th sem files\IBM\IBM Submissions\Dataset\test_set\D\2.png',target_size=(64,64))
```

```
[19] Python
img
```

```
x=image.img_to_array(img)
```

⊗ 0 ⚠ 14 Jupyter Server: Local Cell 1 of 19 Go Live

23°C Mostly clear | Search | ENG IN | 00:01 20-11-2022

File Edit Selection View Go Run Terminal Help Model_Testing.ipynb - Visual Studio Code

Model_Training_Communication_Project.ipynb | Model_Testing.ipynb

D: > Sonal College > 7th sem files > IBM > Real-Time-Communication-Specially-Abled-main > ProjectFiles > Training_File > Model_Testing.ipynb > Testing the model

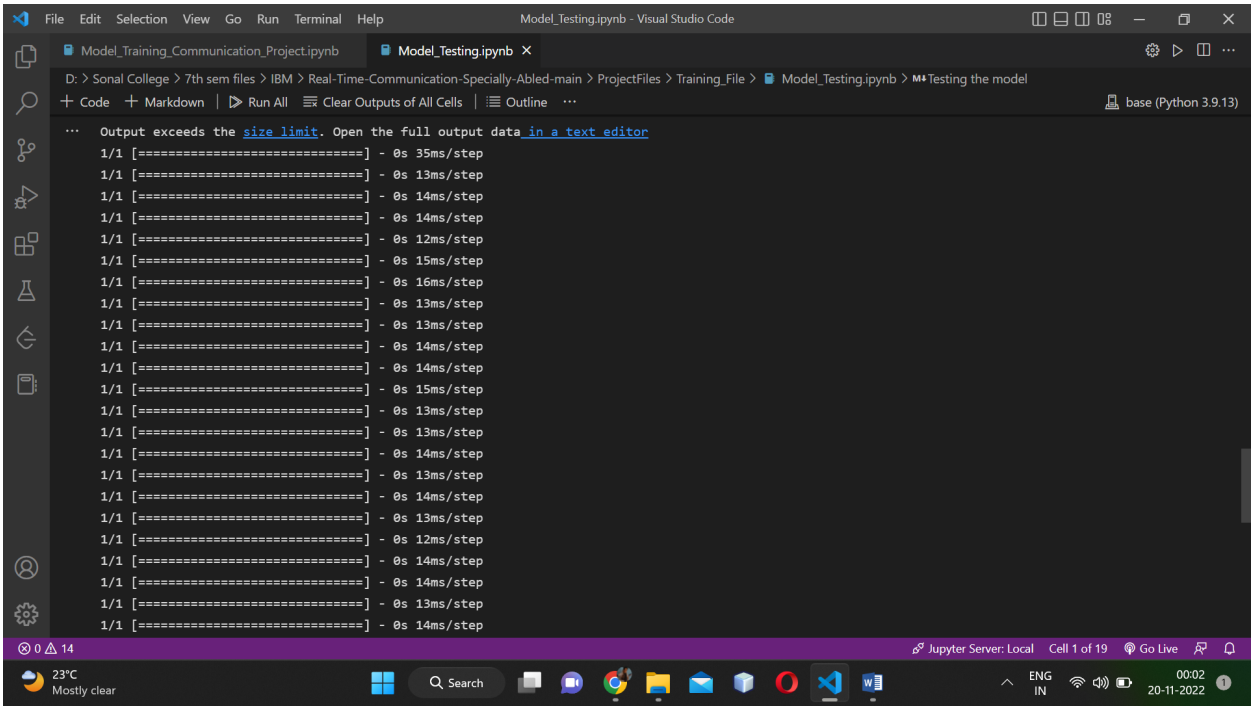
+ Code + Markdown | Run All | Clear Outputs of All Cells | Outline ... base (Python 3.9.13)

CNN Video Analysis

```
[31] Python
import cv2
import numpy as np
from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing import image
model=load_model('asl_model_84_54.h5')
video=cv2.VideoCapture(0)
index=['A','B','C','D','E','F','G','H','I']
while 1:
    succes,frame=video.read()
    cv2.imwrite('image.jpg',frame)
    img=image.load_img('image.jpg',target_size=(64,64))
    x=image.img_to_array(img)
    x=np.expand_dims(x,axis=0)
    pred=np.argmax(model.predict(x),axis=1)
    y=pred[0]
    copy = frame.copy()
    cv2.rectangle(copy, (320, 100), (620,400), (255,0,0), 5)
    cv2.putText(frame,'The Predicted Alphabet is: '+str(index[y]),(100,100),cv2.FONT_HERSHEY_SIMPLEX,1,(0,0,0),4)
    cv2.imshow('image',frame)
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break
video.release()
cv2.destroyAllWindows()
```

⊗ 0 ⚠ 14 Jupyter Server: Local Cell 1 of 19 Go Live

23°C Mostly clear | Search | ENG IN | 00:01 20-11-2022



GITHUB & PROJECT DEMO LINK

GitHub Link: https://github.com/IBM-EPBL/IBM-Project-20102-1659712615

Project Demo Link:
https://drive.google.com/file/d/1YV5yJSpGMxxzypj5_k9QI2MYsiLWFBui/view?usp=drive_web