

Mini Project Report on

SIGN LANGUAGE RECOGNITION

Submitted by

NAME	CLASS	ROLL NO.
Jay Karolia	TE-9	22
Kashyap Kurani	TE-9	25
Chintan Rajgor	TE-9	34
Ananya Tripathi	TE-9	56

Under the guidance of

Prof. Karuna Borhade,

Co-guide: Prof. Vanadana Soni



DEPARTMENT OF COMPUTER ENGINEERING
SHAH AND ANCHOR KUTCHHI ENGINEERING COLLEGE
CHEMBUR, MUMBAI – 400088.

2021 - 2022



Mahavir Education Trust's
SHAH AND ANCHOR KUTCHHIENGINEERING
COLLEGE

Mahavir Education Trust Chowk, W.T Patil Marg, D P Rd, next to Duke's Company, Chembur, Mumbai, Maharashtra 400088

Affiliated to University of Mumbai, Approved by D.T.E. & A.I.C.T.E.

Awarded accreditation of Computer & Information Technology Engineering by NBA

(for 3 years w.e.f. 1st July, 2019)

Certificate

This is to certify that the report of the mini project entitled

SIGN LANGUAGE RECOGNITION

is a bonafide work of

NAME	CLASS	ROLL NO.
Jay Karolia	TE-9	22
Kashyap Kurani	TE-9	25
Chintan Rajgor	TE-9	34
Ananya Tripathi	TE-9	56

submitted to the

UNIVERSITY OF MUMBAI

During semester VI

in

COMPUTER ENGINEERING DEPARTMENT

Guide

(Prof. Uday Bhawe)

I/c Head of Department

Approval for Mini Project Report for T. E. Semester VI

This mini project report entitled “SIGN LANGUAGE RECOGNITION” by Chintan Rajgor, Annanya Tripathi, Kashyap Kurani and Jay Karolia is approved for the partial fulfillment of the requirement for the completion of Semester V.

Name and Sign of Internal Examiner _____


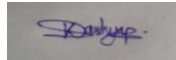
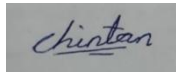
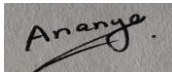
Name and Sign of External Examiner _____

Date:

Place: SAKEC

Declaration

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Name of Student	Class	Roll No.	Signature
1. Jay Karolia	TE9	22	
2. Kashyap Kurani	TE9	25	
3. Chintan Rajgor	TE9	34	
4. Ananya Tripathi	TE9	56	

Date:

Attendance Certificate

To,
The Principal
Shah and Anchor Kutchhi Engineering College,
Chembur, Mumbai-88

Date:

Subject: Confirmation of Attendance

Respected Sir,

This is to certify that Third year (TE) students
Chintan Rajgor, Annanya Tripathi, Kashyap Kurani and Jay Karolia

have duly attended the sessions on the day allotted to them during the period from
16/06/2021 to 06/10/2021 for performing the Mini Project titled “Sign Language Recognition”.

They were punctual and regular in their attendance. Following is the detailed record of the student's attendance.

Attendance Record:

Date	Chintan Rajgor	Annanya Tripathi	Kashyap Kurani	Jay Karolia
	Present/Absent	Present/Absent	Present/Absent	Present/Absent
16/06/2021	Present	Present	Present	Present
23/06/2021	Present	Present	Present	Present
07/07/2021	Present	Present	Present	Present
14/07/2021	Present	Present	Present	Present
28/07/2021	Present	Present	Present	Present
04/08/2021	Present	Present	Present	Present
11/08/2021	Present	Present	Present	Present
18/08/2021	Present	Present	Present	Present
08/09/2021	Present	Present	Present	Present

Signature and Name of Internal Guide

Abstract

Those who suffer from being deaf and have impaired hearing should not be sheltered from communicating with the rest of their peers. Just because a portion of the population cannot hear or speak, does not mean that their form of communication should not advance along with the rest of the world. Sign language is how the hearing impaired express their feelings, contribute to a conversation, learn, and overall live their lives as normal as possible. Many high schools, colleges, and universities are introducing sign language classes into their course catalogues, and rightfully so. Being that it is 2021 and there are widespread pushes toward equality and inclusion, sign language should be included in this worldly push. Sign language should be offered as either an elective, or language course. Although when learning and conducting sign language there is no speech involved, it is still a language of its own, and should be viewed as so. It is a form of communication, and is just as important as learning Spanish or French or any other language. Having sign language courses readily available to students will further enhance this push toward achieving complete inclusiveness.



Mahavir Education Trust's
Shah and Anchor Kutchhi Engineering College,
Chembur, Mumbai 400 088
UG Program in Computer Engineering

Organization of SEM.V T.E. MiniProject Report

1.	Front Page
2.	Certificate (from college)
3.	Project Report Approval
4.	Declaration
5.	Attendance Record (from college)
6.	Abstract
7.	Table of Contents
8.	List of Figures
9.	List of Tables
Chapter 1	Introduction
Chapter 2	Literature Survey: (Historical survey indicating the chronological developments in the relevant field.)
Chapter 3	Problem Statement:
Chapter 4	Project Design: <ul style="list-style-type: none">● System Block Diagram● Flow Chart● Algorithm
Chapter 5	Implementation Details (Module wise system implementation) <ul style="list-style-type: none">a. Module & Descriptionb. Snapshot
Chapter 6	Results & Analysis
Chapter 7	Conclusion and Future Scope
	Appendix (optional)
	References
	Acknowledgements

LIST OF FIGURES:

- 1. CNN MODEL**
- 2. FLOWCHART**
- 3. MODEL OF ACCURACY**
- 4. MODEL OF LOSS**

LIST OF TABLES:

- 1. LITERATURE REVIEW**

Chapter 1 : Introduction

The desire of students to learn sign language increases every year. It can be assumed that this is because more students are expressing interest in being able to communicate with others who are disabled. Not only is sign language a lifelong skill that students should not take for granted, it is also, its own dialect. The importance of sign language is gaining momentum, finally, and it is evident more people are seeing the need for it in today's society. No one should be left in the dark, no matter what disabilities they may have. Also, it is an everlasting skill that can without a doubt make people more well-rounded. Learning the skill of sign language can also show the deaf community that they are not being forgotten, and they have the same access to communication with the rest of the world as anyone else, and that their voices should never be muted or disregarded. Sign languages are a visual representation of thoughts through hand gestures, facial expressions, and body movements. Sign Languages also have several variants, such as American Sign Language (ASL), Argentinean Sign Language (LSA), British Sign Language (BSL) and ISL.

Recognition of sign language can be done in two ways, either glove based recognition or vision based recognition. In glove based technique a network of sensors is used to capture the movements of the fingers. Facial expressions cannot be recognized in this method and also, wearing a glove is always uncomfortable for the users. This method cannot be implemented massively since data gloves are very much expensive. So, the proposed system uses the non- invasive vision based recognition method. The vision-based recognition can be achieved in two ways. They are Static recognition or Dynamic recognition. In static recognition system, the input may be an image of hand pose. It provides an only 2D representation of the gesture, and this can be used to recognize only alphabets and numbers. For recognition of continuous sign language, the dynamic gesture recognition system is used. Here the real-time videos are given as inputs to the system, a sequence of hand movements form the gesture of the word/sentence. Information Technology with its modern methodologies such as artificial intelligence and cloud computing has an impressive role in enhancing intercommunication among people with vocal disabilities and normal people.

Chapter 2 : Literature Survey

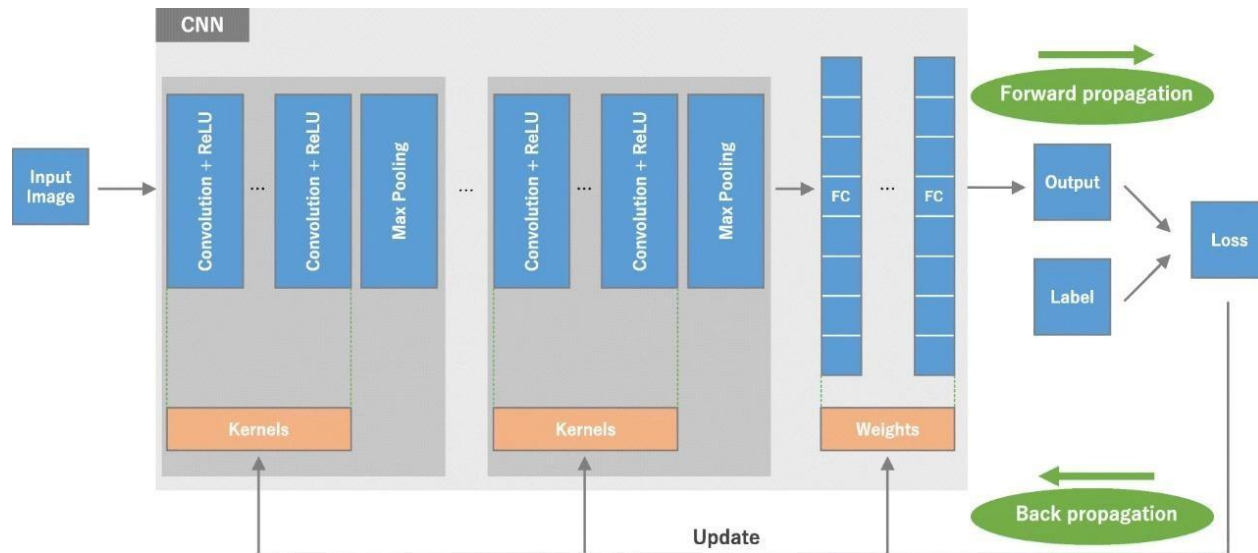
SR.NO	TITLE	Publication	YEAR	DESCRIPTION	Drawbacks
1	Real Time Recognition of Indian Sign Language	IEEE	2019	It uses dynamic noninvasion vision method to get realtime video as input and then use contours to extract face and both hands and given gesture id and it is compared with previously recorded gestures.	They have use Fuzzy c-means algorithm instead of commonly used CNN algorithm which is easy to implement.
2	Hand Gesture Recognition for Indian Sign Language using Skin Color Detection and Correlation- Coefficient algorithm with Neuro-Fuzzy Approach	IEEE	2019	In this paper authors have to choose to use correlation-coefficient algorithm with neuro-fuzzy approach rather than CNN model.	This system do not take real time video as an input it take images as an input so the words that can be predicted are also limited.
3	SIGN LANGUAGE INTERPRETER HAND USING OPTICAL-FLOW	IEEE	2016	This method uses optical flow technique formed by Lucas-Kanade Algorithm. This algorithm is basically used to find derivation of light intensity.	It has Low accuracy compared to most of the other existing systems.
4	Sign Language Detection from Hand Gesture Images using Deep Multi- layered Convolution Neural Network	IEEE	2019	This method uses multilayered CNN to recognize both static as well as dynamic gestures.	There is pre-processing of image which takes the hand-signs from a constant image and places it in a noisy background for the input, which improves the accuracy of detection but is slow compared to the other existing systems.

Chapter 3 : Problem Statement

Its already hard to communicate with Specially Abled (deaf and mute) people and its gotten more difficult after lockdown due to covid-19 . So we decided to design a real time software system that will be able to recognize hand-gestures using deep learning techniques. This project aims to predict the 'alphanumeric' gesture of the system as well as custom hand signs for users.

Chapter 4 : PROJECT DESIGN

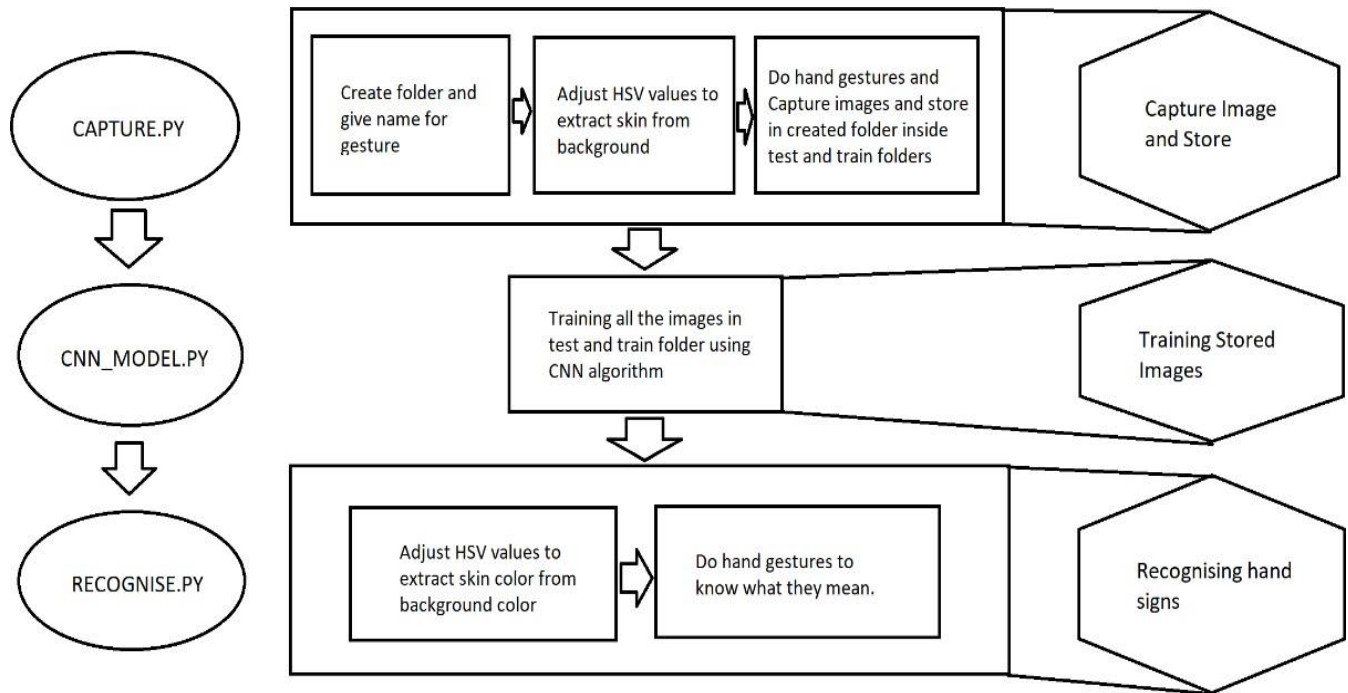
1. SYSTEM BLOCK DIAGRAM:



2. ALGORITHM:

- I. Storing hand-signs:
 - a. Switch on the camera
 - b. Set the HSV values to detect hand skin from background.
 - c. Capture the image which gets stored in 2 folders namely test and train.
- II. Training CNN (Convolution Neural Network) model:
 - a. CNN is a type of deep learning model for processing data that has a grid pattern, such as images, and designed to automatically and adaptively learn spatial hierarchies of features, from low- to high- level patterns.
 - b. Using this cnn model we train the software to recognize all the images from test and train folder.
- III. Recognition of hand-signs:
 - a. Switch on the camera
 - b. Set the HSV values to detect hand skin from background.
 - c. Recognize the hand-signs and display the output by comparing with the images stored in database.

3. FLOWCHART:



Chapter 6: SYSTEM REQUIREMENTS

HARDWARE REQUIREMENTS:

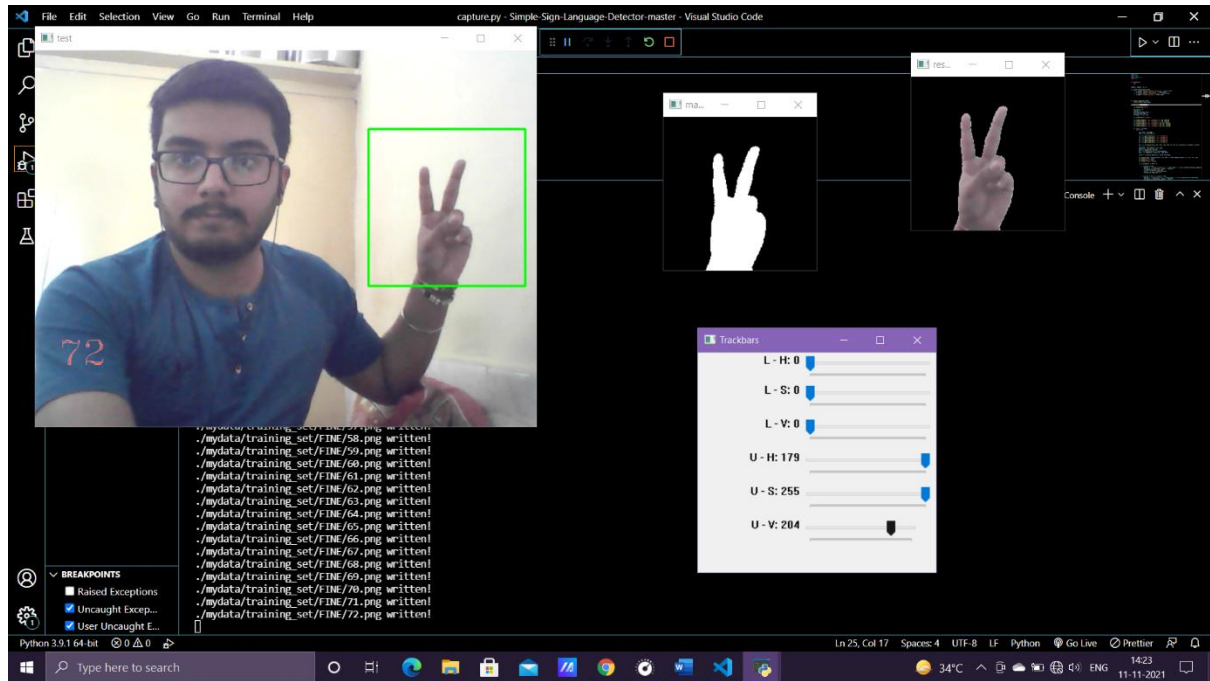
- a. Hard Disk minimum of 40 GB.
- b. RAM minimum of 2 GB.
- c. Dual Core and up ,15” Monitor.
- d. Integrated webcam or external webcam (15 -20fps)

SOFTWARE REQUIREMENTS:

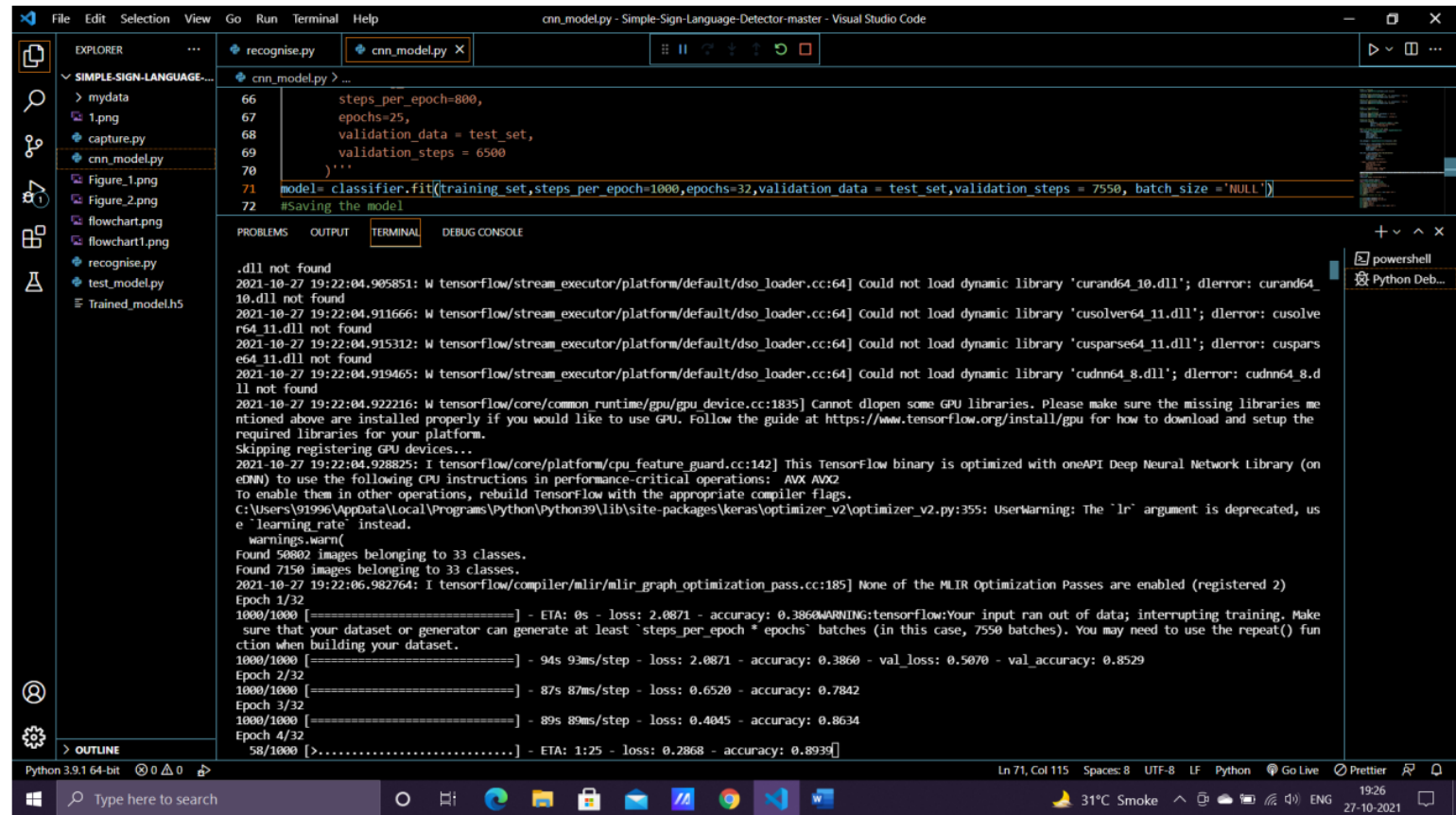
- a. Microsoft Visual Studio 2010
- b. Python 3
- c. Tensorflow
- d. Keras
- e. OpenCV 3

Chapter 5 : IMPLEMENTATION DETAILS

1, We have Successfully captured and stored images with capture.py



2. We have Successfully Trained Images with CNN model



```
File Edit Selection View Go Run Terminal Help
cnn_model.py - Simple Sign Language Detector-master - Visual Studio Code

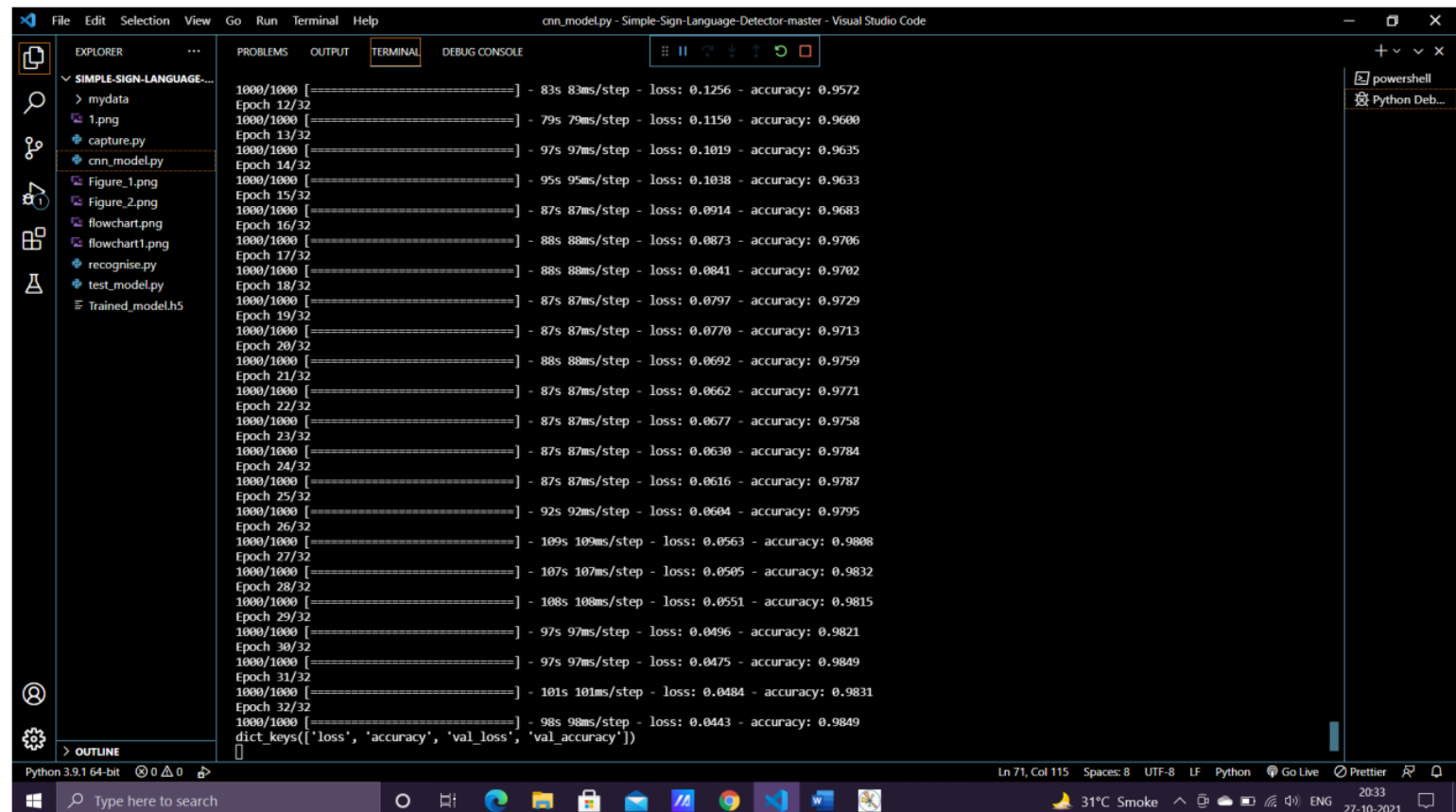
EXPLORER
SIMPLE-SIGN-LANGUAGE-...
  mydata
  1.png
  capture.py
  cnn_model.py
  Figure_1.png
  Figure_2.png
  flowchart.png
  flowchart1.png
  recognise.py
  test_model.py
  Trained_model.h5

cnn_model.py > ...
66     steps_per_epoch=800,
67     epochs=25,
68     validation_data = test_set,
69     validation_steps = 6500
70 )'''
71 model=classifier.fit(training_set,steps_per_epoch=1000,epochs=32,validation_data = test_set,validation_steps = 7550, batch size = "NULL")
72 #Saving the model

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

.dll not found
2021-10-27 19:22:04.905851: W tensorflow/stream_executor/platform/default/dso_loader.cc:64] Could not load dynamic library 'curand64_10.dll'; dlerror: curand64_10.dll not found
2021-10-27 19:22:04.911666: W tensorflow/stream_executor/platform/default/dso_loader.cc:64] Could not load dynamic library 'cusolver64_11.dll'; dlerror: cusolver64_11.dll not found
2021-10-27 19:22:04.915132: W tensorflow/stream_executor/platform/default/dso_loader.cc:64] Could not load dynamic library 'cusparse64_11.dll'; dlerror: cusparse64_11.dll not found
2021-10-27 19:22:04.919465: W tensorflow/stream_executor/platform/default/dso_loader.cc:64] Could not load dynamic library 'cudnn64_8.dll'; dlerror: cudnn64_8.dll not found
2021-10-27 19:22:04.922216: W tensorflow/core/common_runtime/gpu/gpu_device.cc:1835] Cannot dlopen some GPU libraries. Please make sure the missing libraries mentioned above are installed properly if you would like to use GPU. Follow the guide at https://www.tensorflow.org/install/gpu for how to download and setup the required libraries for your platform.
Skipping registering GPU devices...
2021-10-27 19:22:04.928825: I tensorflow/core/platform/cpu_feature_guard.cc:142] This TensorFlow binary is optimized with oneAPI Deep Neural Network Library (onnx) to use the following CPU instructions in performance-critical operations: AVX AVX2
To enable them in other operations, rebuild TensorFlow with the appropriate compiler flags.
C:\Users\S91996\AppData\Local\Programs\Python\Python39\lib\site-packages\keras\optimizer_v2\optimizer_v2.py:355: UserWarning: The `lr` argument is deprecated, use `learning_rate` instead.
warnings.warn(
Found 50802 images belonging to 33 classes.
Found 7150 images belonging to 33 classes.
2021-10-27 19:22:06.982764: I tensorflow/compiler/mlir/mlir_graph_optimization_pass.cc:185] None of the MLIR Optimization Passes are enabled (registered 2)
Epoch 1/32
1000/1000 [=====] - ETA: 0s - loss: 2.0871 - accuracy: 0.3860WARNING:tensorflow:Your input ran out of data; interrupting training. Make
sure that your dataset or generator can generate at least `steps_per_epoch * epochs` batches (in this case, 7550 batches). You may need to use the repeat() fun
ction when building your dataset.
1000/1000 [=====] - 94s 93ms/step - loss: 2.0871 - accuracy: 0.3860 - val_loss: 0.5070 - val_accuracy: 0.8529
Epoch 2/32
1000/1000 [=====] - 87s 87ms/step - loss: 0.6520 - accuracy: 0.7842
Epoch 3/32
1000/1000 [=====] - 89s 89ms/step - loss: 0.4045 - accuracy: 0.8634
Epoch 4/32
58/1000 [>.....] - ETA: 1:25 - loss: 0.2868 - accuracy: 0.8939

Python 3.9.1 64-bit 0 0 0
Ln 71, Col 115 Spaces: 8 UTF-8 LF Python Go Live Prettier 19:26 27-10-2021
```



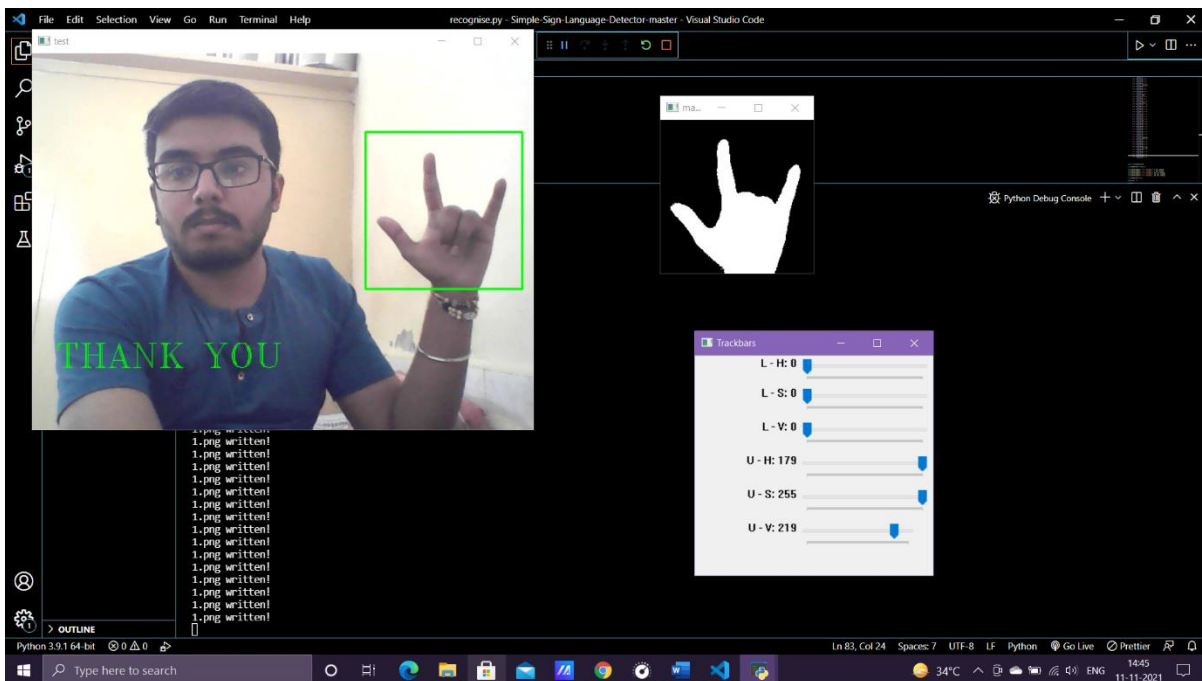
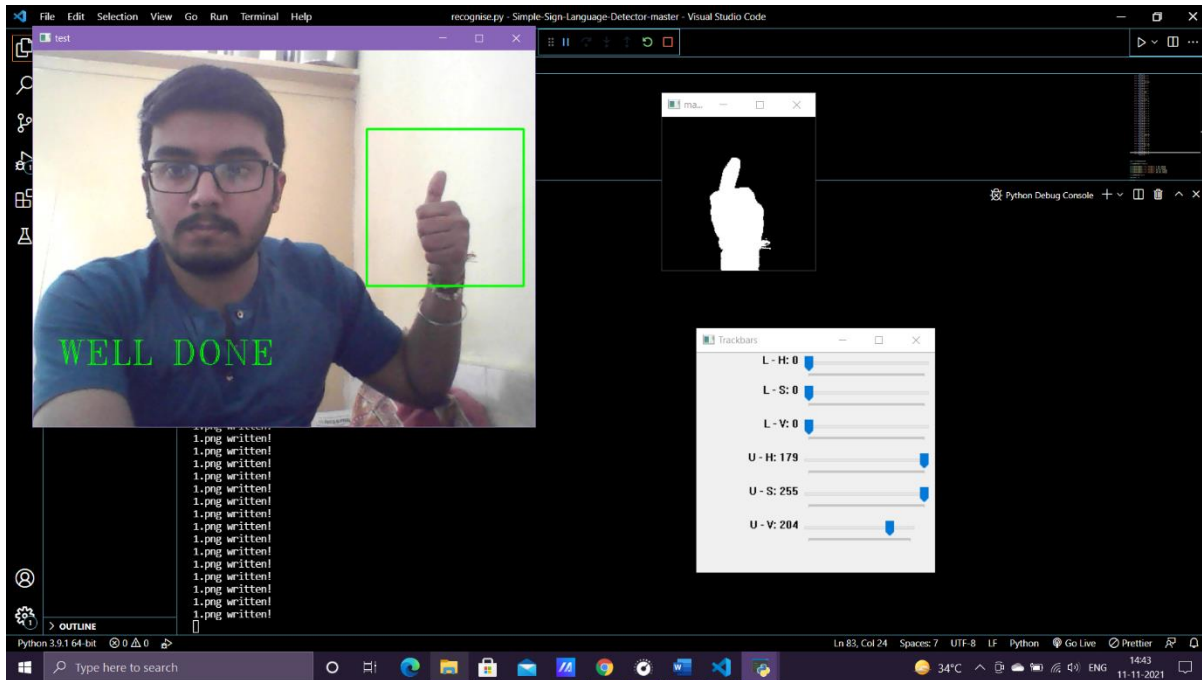
```
File Edit Selection View Go Run Terminal Help
cnn_model.py - Simple Sign Language Detector-master - Visual Studio Code

EXPLORER
SIMPLE-SIGN-LANGUAGE-...
  mydata
  1.png
  capture.py
  cnn_model.py
  Figure_1.png
  Figure_2.png
  flowchart.png
  flowchart1.png
  recognise.py
  test_model.py
  Trained_model.h5

cnn_model.py > ...
1000/1000 [=====] - 83s 83ms/step - loss: 0.1256 - accuracy: 0.9572
Epoch 12/32
1000/1000 [=====] - 79s 79ms/step - loss: 0.1150 - accuracy: 0.9600
Epoch 13/32
1000/1000 [=====] - 97s 97ms/step - loss: 0.1019 - accuracy: 0.9635
Epoch 14/32
1000/1000 [=====] - 95s 95ms/step - loss: 0.1038 - accuracy: 0.9633
Epoch 15/32
1000/1000 [=====] - 87s 87ms/step - loss: 0.0914 - accuracy: 0.9683
Epoch 16/32
1000/1000 [=====] - 88s 88ms/step - loss: 0.0873 - accuracy: 0.9706
Epoch 17/32
1000/1000 [=====] - 88s 88ms/step - loss: 0.0841 - accuracy: 0.9702
Epoch 18/32
1000/1000 [=====] - 87s 87ms/step - loss: 0.0797 - accuracy: 0.9729
Epoch 19/32
1000/1000 [=====] - 87s 87ms/step - loss: 0.0770 - accuracy: 0.9713
Epoch 20/32
1000/1000 [=====] - 88s 88ms/step - loss: 0.0692 - accuracy: 0.9759
Epoch 21/32
1000/1000 [=====] - 87s 87ms/step - loss: 0.0662 - accuracy: 0.9771
Epoch 22/32
1000/1000 [=====] - 87s 87ms/step - loss: 0.0677 - accuracy: 0.9758
Epoch 23/32
1000/1000 [=====] - 87s 87ms/step - loss: 0.0630 - accuracy: 0.9784
Epoch 24/32
1000/1000 [=====] - 87s 87ms/step - loss: 0.0616 - accuracy: 0.9787
Epoch 25/32
1000/1000 [=====] - 92s 92ms/step - loss: 0.0604 - accuracy: 0.9795
Epoch 26/32
1000/1000 [=====] - 109s 109ms/step - loss: 0.0563 - accuracy: 0.9808
Epoch 27/32
1000/1000 [=====] - 107s 107ms/step - loss: 0.0505 - accuracy: 0.9832
Epoch 28/32
1000/1000 [=====] - 108s 108ms/step - loss: 0.0551 - accuracy: 0.9815
Epoch 29/32
1000/1000 [=====] - 97s 97ms/step - loss: 0.0496 - accuracy: 0.9821
Epoch 30/32
1000/1000 [=====] - 97s 97ms/step - loss: 0.0475 - accuracy: 0.9849
Epoch 31/32
1000/1000 [=====] - 101s 101ms/step - loss: 0.0484 - accuracy: 0.9831
Epoch 32/32
1000/1000 [=====] - 98s 98ms/step - loss: 0.0443 - accuracy: 0.9849
dict_keys(['loss', 'accuracy', 'val_loss', 'val_accuracy'])

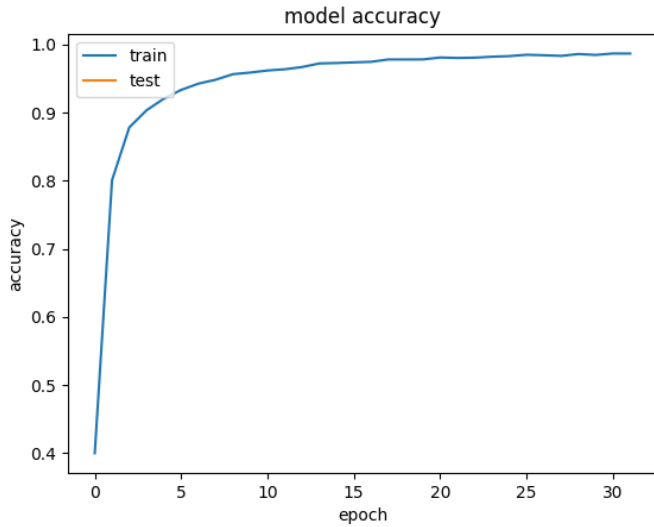
Python 3.9.1 64-bit 0 0 0
Ln 71, Col 115 Spaces: 8 UTF-8 LF Python Go Live Prettier 20:33 27-10-2021
```

3, Now we have Identified Signs and Predicted the Meaning with Recognise.py successfully

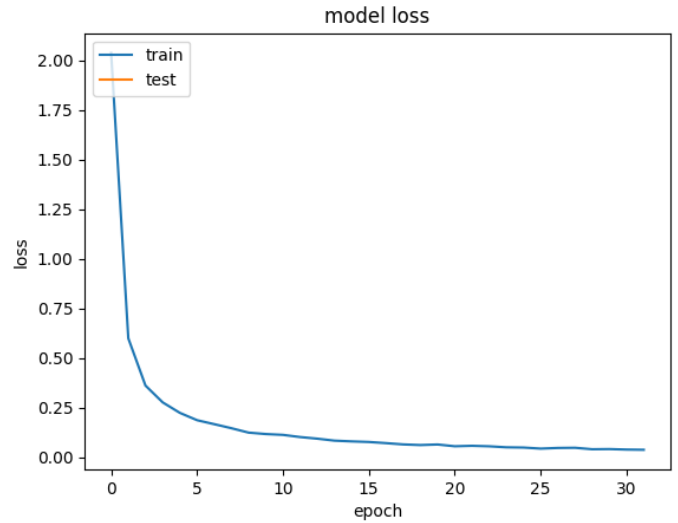


CHAPTER 6: RESULT AND ANALYSIS

MODEL OF ACCURACY:



MODEL OF LOSS:



ANALYSIS:

As we can see from the model of accuracy and loss, our model has an accuracy of 97%.

Chapter 7 : Conclusion and Future Scope

- **Conclusion:**

The core aim of this project is to establish a model which will help us in communicating with the people who are specially abled without having any difficulties. The model is very helpful in portraying an impressive role in enhancing casual communication among people with hearing disabilities and normal people.

- **Future Scope:**

In the future, we will be adding more words into the system and also make the system capable of dynamic type recognition.

References

- ❖ <https://ieeexplore.ieee.org/document/9036832>
- ❖ <https://ieeexplore.ieee.org/document/9362897>
- ❖ <https://ieeexplore.ieee.org/document/7873837>
- ❖ <https://ieeexplore.ieee.org/document/8945850>
- ❖ <https://ieeexplore.ieee.org/document/8862125>
- ❖ <https://ieeexplore.ieee.org/document/8622141>
- ❖ <http://talkinghands.co.in/>
- ❖ <https://insightsimaging.springeropen.com/articles/10.1007/s13244-018-0639-9#Sec3>