HUMAN SIGN LANGUAGE RECOGNITION USING DEEP LEARNING

Major project report submitted
in partial fulfillment of the requirement for award of the degree of

Bachelor of Technology
in
Computer Science & Engineering

By

JAWAHAR V	(18UECS0332)	**(11361)**
SANTHAKUMAR D	(18UECS0771)	**(11343)**
RAKESH REDDY Y.V	(18UECS0946)	**(11631)**

Under the guidance of
Mrs.S.THYLASHRI,M.E.,
ASSISTANT PROFESSOR

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
SCHOOL OF COMPUTING

VEL TECH RANGARAJAN DR. SAGUNTHALA R&D INSTITUTE OF SCIENCE & TECHNOLOGY

(Deemed to be University Estd u/s 3 of UGC Act, 1956)
Accredited by NAAC with A Grade
CHENNAI 600 062, TAMILNADU, INDIA

June,2022

CERTIFICATE

It is certified that the work contained in the project report titled ”HUMAN SIGN LANGUAGE RECOGNITION USING DEEP LEARNING” by ”JAWAHAR V (18UECS0332), SANTHAKUMAR D (18UECS0771), RAKESH REDDY Y V (18UECS0946)” has been carried out under my supervision and that this work has not been submitted elsewhere for a degree.

Signature of Supervisor
Mrs.S.Thylashri
Assistant professor
Computer Science &Engineering
School of Computing
Vel Tech Rangarajan Dr.Sagunthala R&D
Institute of Science &Technology
June,2022

Signature of Head of the Department
Dr. V. Srinivasa Rao
Professor & Head
Computer Science & Engineering
School of Computing
Vel Tech Rangarajan Dr.Sagunthala R&D
Institute of Science & Technology
June,2022

DECLARATION

We declare that this written submission represents my 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.

(JAWAHAR V)

Date: / /

(SANTHAKUMAR D)

Date: / /

(RAKESH REDDY Y V)

Date: / /

APPROVAL SHEET

This project report entitled (HUMAN SIGN LANGUAGE RECOGNITION USING DEEP LEARNING)) by (JAWAHAR V (18UECS0332), (SANTHAKUMAR D (18UECS0771), (RAKESH REDDY Y.V (18UECS0946) is approved for the degree of B.Tech in Computer Science & Engineering.

Examiners **Supervisor**

Mrs S.Thylashri,M.E.,

Date: / /

Place:

ACKNOWLEDGEMENT

We express our deepest gratitude to our respected **Founder Chancellor and President Col. Prof. Dr. R. RANGARAJAN B.E. (EEE), B.E. (MECH), M.S (AUTO),D.Sc., Foundress President Dr. R. SAGUNTHALA RANGARAJAN M.B.B.S.** Chairperson Managing Trustee and Vice President.

We are very much grateful to our beloved **Vice Chancellor Prof. S. SALIVAHANAN,** for providing us with an environment to complete our project successfully.

We record indebtedness to our **Dean & Head, Department of Computer Science & Engineering Dr.V.SRINIVASA RAO, M.Tech., Ph.D.,** for immense care and encouragement towards us throughout the course of this project.

We also take this opportunity to express a deep sense of gratitude to our Internal Supervisor **Mrs.S.THYLASHRI,M.E.,** for his/her cordial support, valuable information and guidance, he/she helped us in completing this project through various stages.

A special thanks to our **Project Coordinators Mr. V. ASHOK KUMAR, M.Tech., Ms. C. SHYAMALA KUMARI, M.E., Ms.S.FLORENCE, M.Tech.,** for their valuable guidance and support throughout the course of the project.

We thank our department faculty, supporting staff and friends for their help and guidance to complete this project.

JAWAHAR V **(18UECS0332)**
SANTHAKUMAR D **(18UECS0771)**
RAKESH REDDY Y.V **(18UECS0946)**

ABSTRACT

Disability of deaf and dumb is one of the major significant problem in the human world. The development of technologies for recognition of deaf and dumb impaired peoples is done by several methodologies. So, We are developing this project using deep learning by recognizing the symbolic hand gestures of deaf and dumb impaired peoples. Sign language is learned by deaf and dumb impaired people, and usually it is not known to normal people, so it becomes a challenge for communication between a normal person and deaf and dumb impaired person. The main focus of our project is to make predictions of hand gestures from deaf and dumb peoples. Our project recognizes different types of hand gestures as input and make predictions with accuracy level based on pre-trained dataset models by using Convolutional Neural Network (CNN) Algorithm.

Keywords: Hand gestures, Recognition, Convolutional neural network

LIST OF FIGURES

LIST OF ACRONYMS AND ABBREVIATIONS

CNN	Convolutional Neural Network
ROI	Region of Interest
LDA	Linear Discriminant Analysis
WLASL	Word-Level American sign language
SLR	Sign Language Recognition

TABLE OF CONTENTS

Chapter 1

INTRODUCTION

1.1 Introduction

Communication of deaf and dumb impaired peoples is one of the major cause of gap between normal human beings and those impaired peoples. Lack of communication mostly involves with them and it leads less interactions with others. The development of technologies for bridging gap between these impaired peoples are done by several methodologies. Here we are using deep learning method which is based on artificial neural network using in our project which contain multiple processing layers for extracting more features from the given data. It is a machine learning technique that teaches machines or computers to learn that are naturally done by humans. So, we are trying to make model for better interaction with normal human beings and deaf and dumb impaired peoples. The scope of this project is to make predictions with hand gestures and its recognition. The focus will be placed on designing a model of sign language recognition with trained datasets. Training a dataset is used by OpenCV and Region of Interest. We will train dataset using deep learning algorithms and make predictions by comparing with pre-trained datasets and with level of accuracy.

1.2 Aim of the project

Our Aim of the project is to improve the communication of deaf and dumb people. In this manner, a system which can make predictions of hand gestures of deaf and dumb people. Here we are making a model for predicting the results of different hand gestures and we will process images and we will train datasets using deep learning algorithm. Communication is always having a great impact in every field and how it is considered the meaning of the thoughts and expressions that attract the researchers to bridge this gap for every living being.

1.3 Project Domain

Deep Learning is a subnet of machine learning, which is essential a neural network with three or more layers. Each layer builds up from the previous layer. A CNN is a deep learning algorithm which can take in an input image to various aspects in the image and be able to differentiate one from the other. Deep learning tools have been gaining familiarity as a machine learning because they classify richer and deeper representation of features.

1.4 Scope of the Project

The scope of the project is to make the predictions of symbolic hand gestures from pre-trained models and predicts the sign gestures as output based on the level of accuracy.

Chapter 2

LITERATURE REVIEW

Mahesh Kumar N B et al.(2018) [1] Proposed a techniques for Conversion of Sign Language into text, Sign Language Recognition is one of the most growing field and developing field in research area. In this paper it is used recognizing sign language of hand gestures using MATLAB.The Linear Discriminant Analysis (LDA) algorithm was used for gesture recognition and recognized gesture is converted into text and voice format and it helps to dimensionality reduction.

Bowen Shi1 , Aurora Martinez Del Rio2 , Jonathan Keane2 , Diane Brentari2 Greg Shakhnarovich1 , Karen Livescu1 et al.(2019) [2] Proposed a techniques for Fingerspelling recognition in the wild with iterative visual attention. Sign language recognition is a challenging gesture sequence recognition problem. It is an end-to-end model based on an iterative attention mechanism, without explicit hand detection or segmentation. It focuses on increasingly high-resolution regions of interest and outperforms prior work by a large margin.

N. Alnaim and M. Abbod et al.(2019) [3] Proposed a techniques for Hand Gesture Detection Using Neural Networks Algorithms. Human gesture is a form of body language usually used as a mean of communication. Here Artificial Neural Network and Convolutional Neural Network used as processing methods. It includes computer vision operations including hand segmentation, detection and tracking.

Mehreen Hurroo and Mohammad Elham Walizad et al.(2020) [4] proposed a techniques for Sign Language Recognition System using Convolutional Neural Network and Computer Vision. Sign Language is way of communicating their feelings and opinions to the world. In this Convolutional Neural Network is used for training and to classify the images. Predicts sign gestures based on the level of accuracy.

Dongxu Li , Cristian Rodriguez Opazo, Xin Yu, Hongdong Li et al.(2020) [5] proposed a techniques for Word-level Deep Sign Language Recognition from Video: A New Large-scale Dataset and Methods Comparison. Vision-based sign language recognition aims at helping deaf people to communicate with others. It used a new large-scale Word-Level American Sign Language (WLASL) video dataset.

Songyao Jiang, Bin Sun, Lichen Wang, Yue Bai, Kunpeng Li and Yun Fu et al.(2021) [6] proposed a techniques for Sign Language Recognition via Slketon-Aware Multi-Model Ensemble. Sign language is commonly used by deaf people to communicate. It uses a concept of the Skeleton Aware Multi-model sign Language Recognition Framework with Global Ensemble Model (SAM-SLR-v2). All local and global motion information is extracted and fused to make final predictions.

Necati Cihan Camgoz1 , Simon Oscar Koller2 , Hermann Ney2 , Richard Bowden1 et al.(2021) [7] proposed a techniques for Neural Sign Language Translation. Sign Language Recognition (SLR) has been an active research field for the last two decades. The objective is to generate spoken language translations from sign language videos, taking into account the different word orders and grammar. It is jointly learn to align, recognize and translate sign videos to spoken text.

Chapter 3

PROJECT DESCRIPTION

3.1 Existing System

In Existing system they are several methods used for implementing the hand gesture recognition. One of the method is converting the hand or finger gestures into text format, based on the given input and pre-trained dataset the output will be shown in text format. Another method is to aware of skeleton and bone joints and make predictions as results based on the given input. These are some of the existing methods available under sign and gesture recognition.

3.2 Proposed System

In our project we are training the datasets of different hand gestures and it will be used as datasets in our project and datasets will be trained using deep learning algorithm such as Convolutional neural network (CNN) and the trained model will be processed by algorithm is used to make segregate datasets and predictions will be compared with pre-trained hand gesture datasets and will get expected output.

3.3 Feasibility Study

3.3.1 Economic Feasibility

It is the check of the system in line with its work ability, impact on the organization, ability to satisfy user wants and effective use of resources this project will be very helpful to regions to control their economic level to maintain in required way. This project will be economically feasible since there is no use of any external resources and it can be implemented with minimum cost with best results.

3.3.2 Technical Feasibility

It is done by very user interface level. So, the user can interact very easily. Technically also it is not that that much harder to use In some cases, this method analysis step additionally includes a restricted quantity of analysis and style. This project is technically feasible since has a good user interface and is very user friendly and hence the end users can use this project with ease.

3.3.3 Social Feasibility

Among the foremost vital data contained in feasibility study is that the value profit analysis. That is, associate degree assessment of economic justification for computer based system. Analytic think in value for development and weight with in the system. This project will be socially feasible since the it helps government to take the decisions.

3.4 System Specification

3.4.1 Hardware Specification

Web cam
Laptop, Desktop

3.4.2 Software Specification

Operating system : Windows 7,8,10
Programming : Python
Library : OpenCv, Keras

3.4.3 Standards and Policies

Standards of the project ensure that quality is maintained throughout the project and to effectively manage all the documentation created throughout the project. The project is maintained and done according to the standards of the system . Policies of the project are managed effectively within the scope. quality, Resources (time and risk limitations). Appropriate governance and supervision are established During the life of a project, communication, quality, and risk management plans are developed

and executed. Appropriate authorization and acceptance shall be established during a project's lifetime.

Chapter 4

METHODOLOGY

4.1 General Architecture

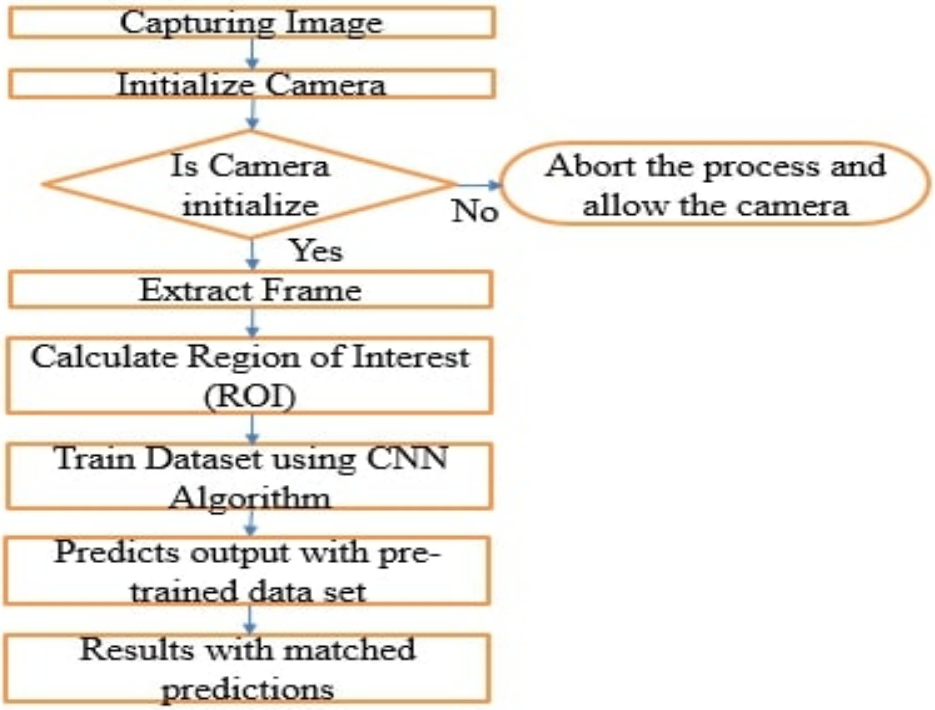


Figure 4.1: **Architecture Diagram**

In the above architecture the flow of image using our system. It describes about the image is processed using Convolutional neural network algorithm and predicts the hand gestures.

4.2 Design Phase

4.2.1 Data Flow Diagram

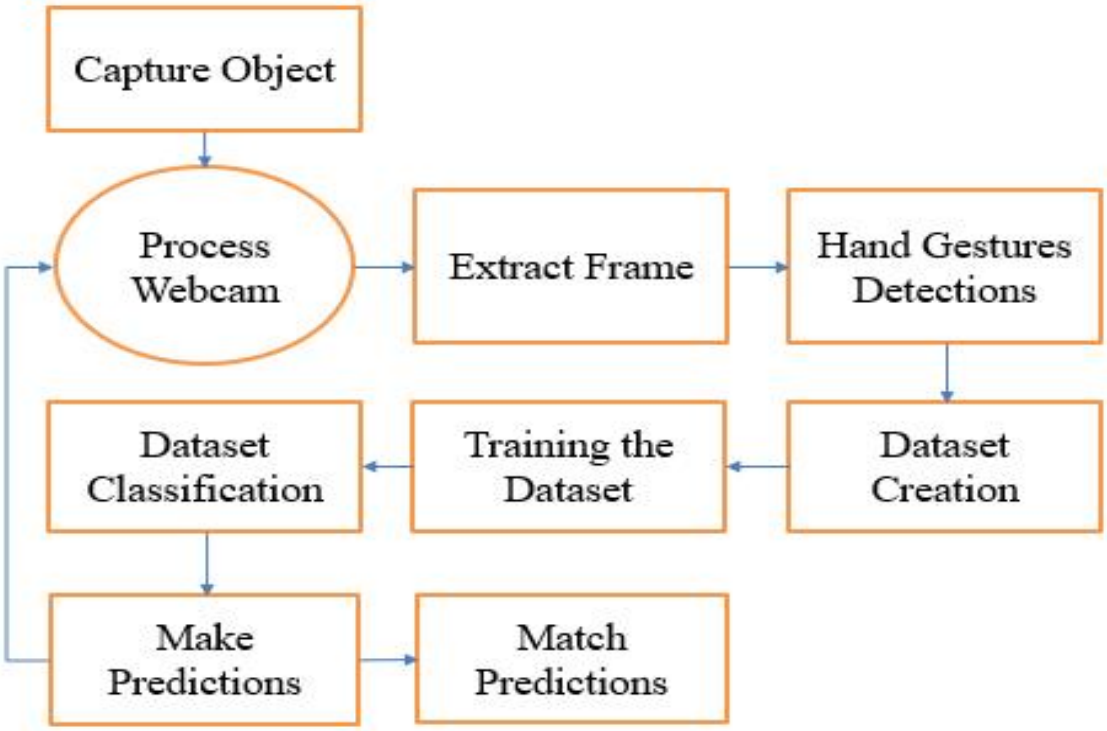


Figure 4.2: **Data flow diagram**

In the above data flow diagram first object will captured by webcam and then frame will be identified. Then dataset will be created and it will be trained and match predictions.

4.2.2 Use Case Diagram

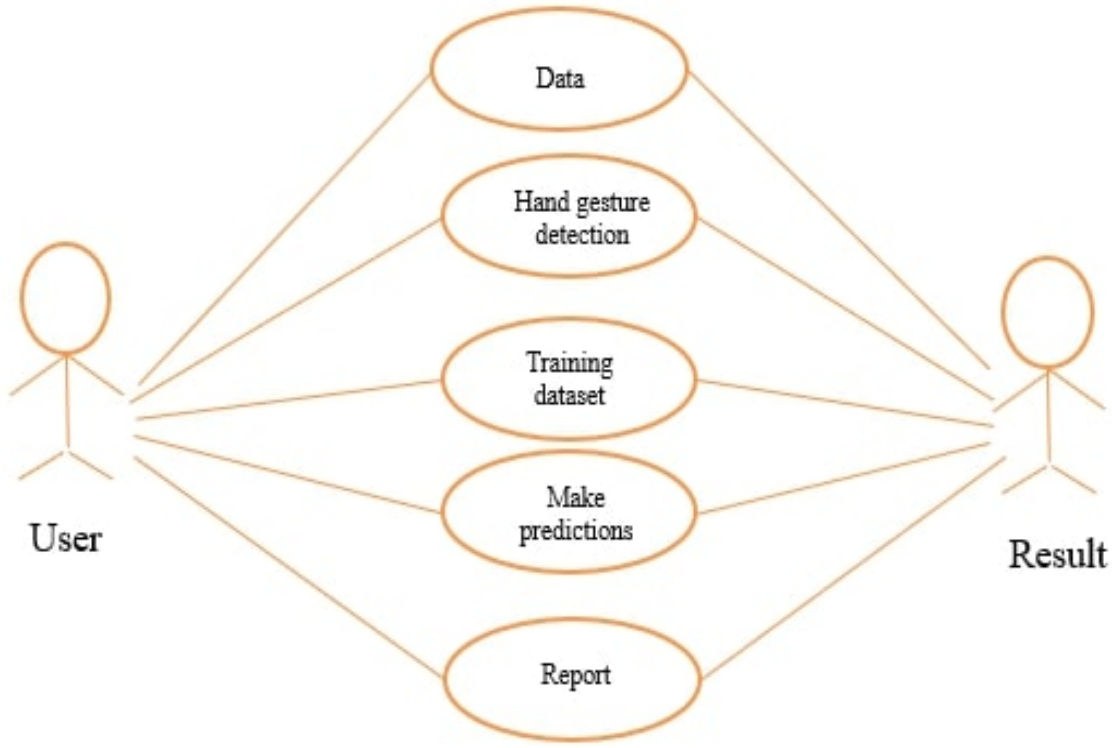


Figure 4.3: **Use Case diagram**

In the above Use case diagram the process between user and way of result is shown. Here, Data collections made and hand gestures will be detected and it will be trained, make predictions and reports the result.

4.2.3 Class Diagram

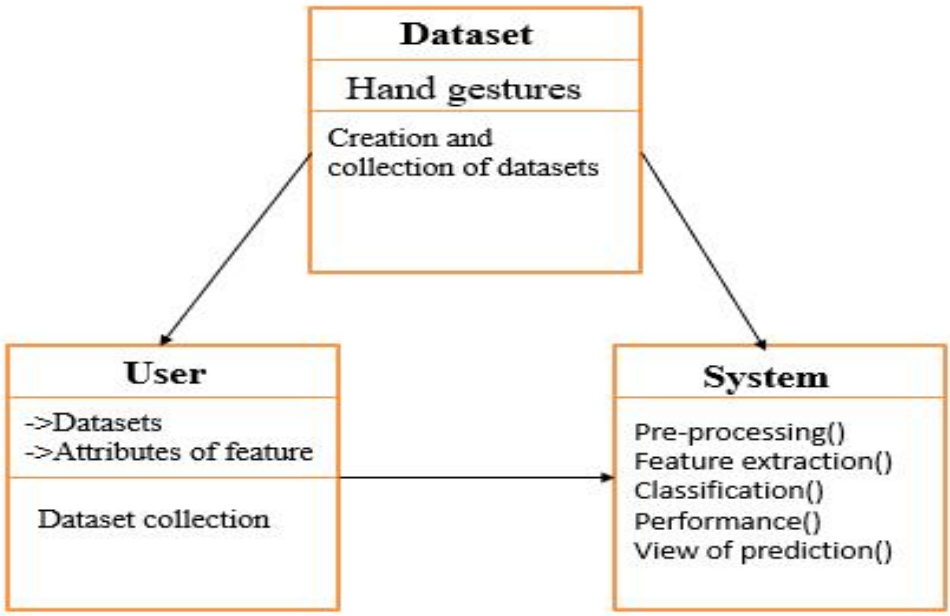


Figure 4.4: **Class diagram**

In the above Class diagram the flow goes from dataset to user and system and from user to system. Pre-processing, feature extraction are acts as attributes of system.

4.2.4 Sequence Diagram

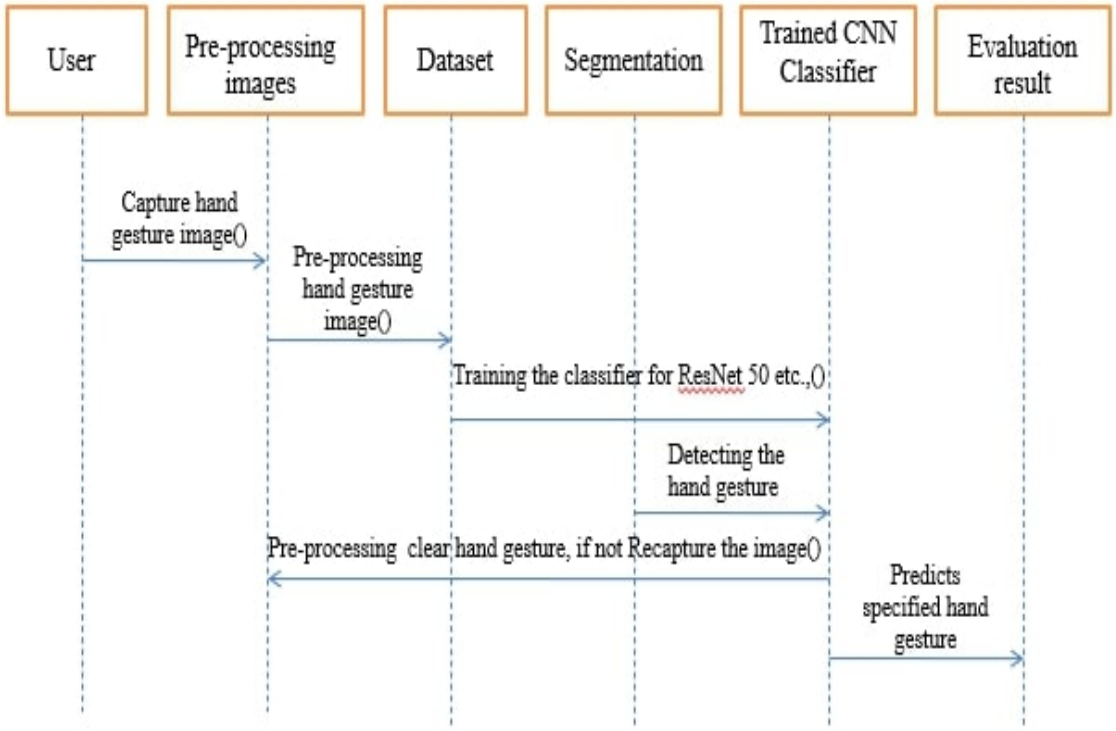


Figure 4.5: **Sequence diagram**

In the above diagram we are first creating a dataset and then it will be pre-processed and processed through the algorithm and it will be predicted based on the given datasets.

4.2.5 Collaboration diagram

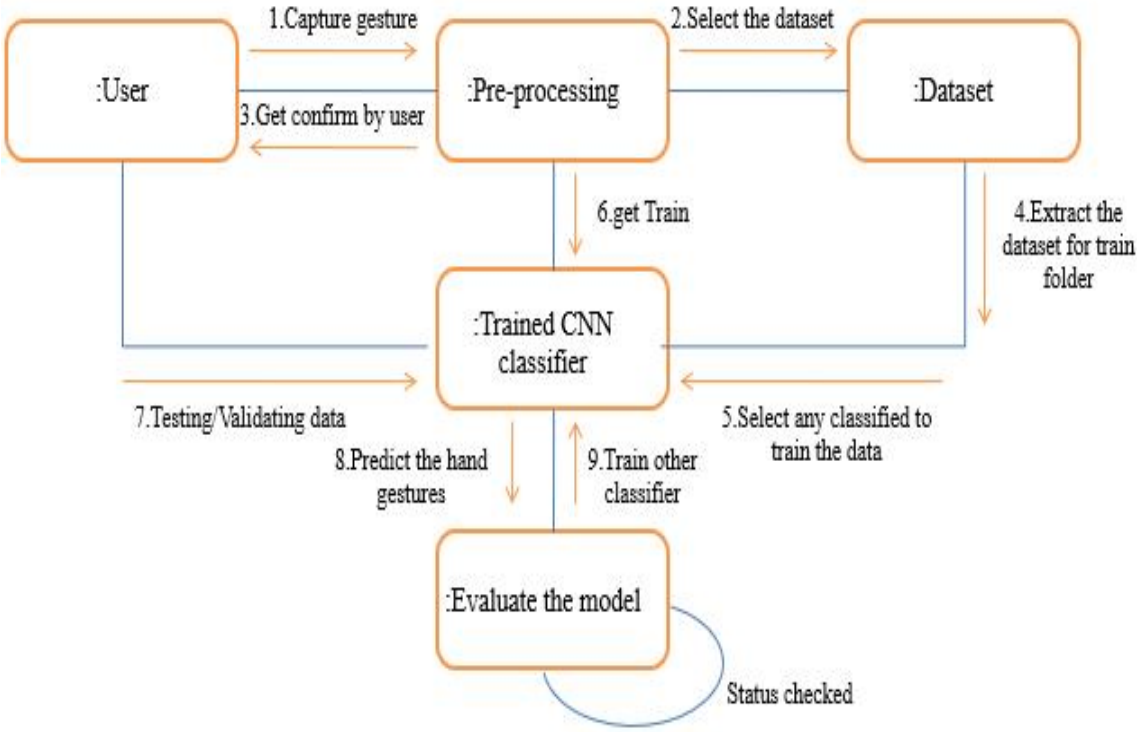


Figure 4.6: **Collaboration diagram**

In the above Collaboration diagram gesture images will be pre-processed and trained using CNN and it evaluates the model.

4.2.6 Activity Diagram

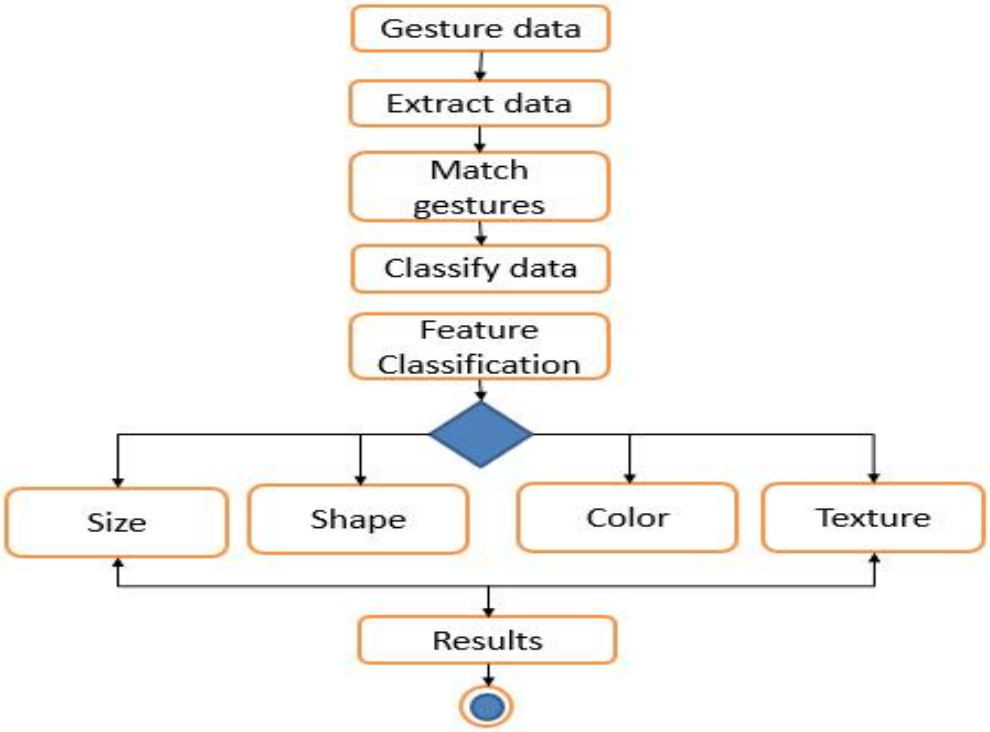


Figure 4.7: **Activity diagram**

In the above activity diagram, the features extracting from the hand gestures images are size, shape, color and texture are shown as flow. Here the gesture data will be extracts the feature of images and gives results with matched data.

4.3 Algorithm & Pseudo Code

4.3.1 Algorithm

Step1: Importing the hand gestures dataset.

Step2: Applying pre-processing technique.

Step3: Splitting the dataset into train and test.

Step4: Using required algorithm for training.

Step5: Train the data using pre-trained models

Step6: Predicting the gestures

4.3.2 Pseudo Code

Get the catagories
while true
frame = cap.read()
frame = cv2.flip(frame, 1)
Get inputs for x1, y1, x2, y2
set roi = frame[y1:y2, x1:x2]
set roi = cv2.resize(roi, (64,64))
set roi = cv2.cvtcolor
result = predict(test image reshape)
predict = sorted (test image)
prediction = 'zero': result[0][0],
'one': result[0][1],
'two': result[0][2],
'three': result[0][3],
'four': result[0][4],
'five': result[0][5]

4.4 Module Description

In this we are using an deep learning algorithm known as Convolutional neural network (CNN).It has an input layer, output layer and numerous hidden layers which allowing to know about complicated objects and patterns. It will handle pooling processes by sub dividing the input before functioning. Pooling is used to shrink spatial size for reducing the parameters and computations in the given network. This algorithm results in excellent classification and recognition results and handles audio, text, pattern.

4.4.1 Collecting the datasets

In this module we are doing the Collection of dataset and creation of dataset for our project by using the required code.

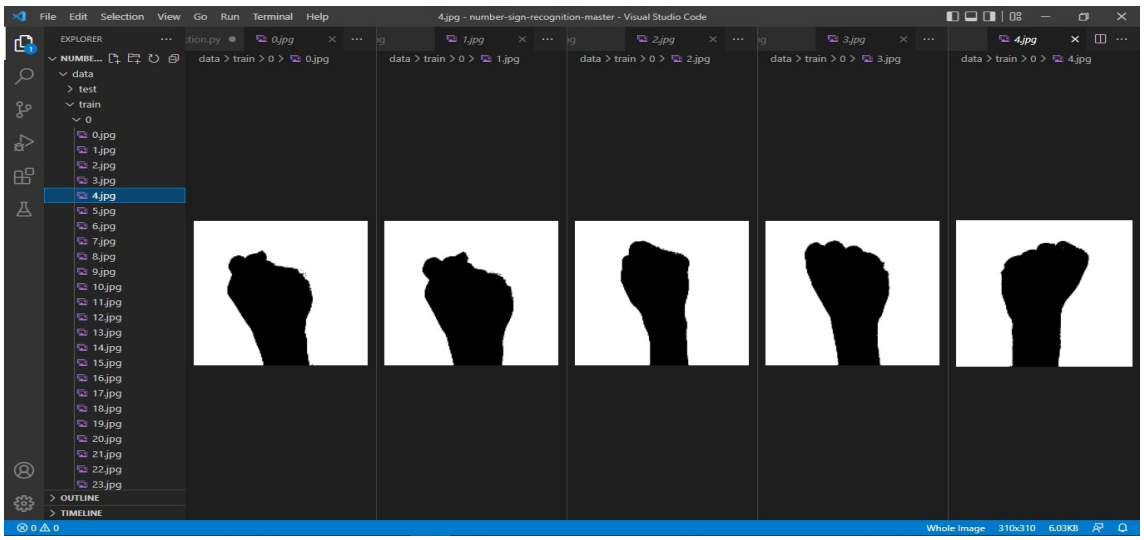


Figure 4.8: **Collection of Dataset**

4.4.2 Extracting frame

In this module Region of interest (ROI) will be extracted with frame for capturing the different hand gestures with in the ROI.

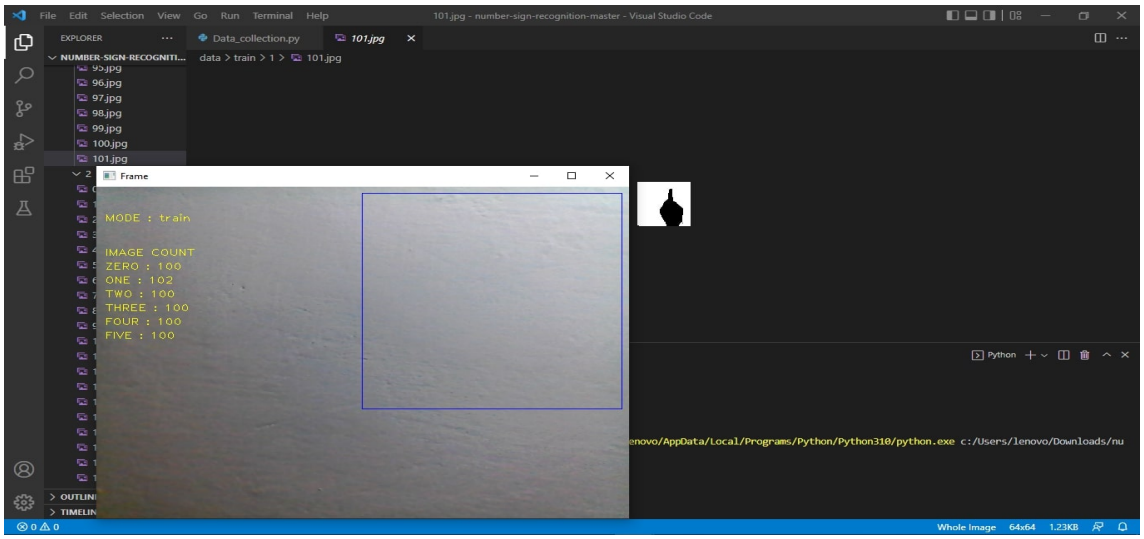


Figure 4.9: **Region of interest**

4.4.3 Taining and prediction of data

Finally, In this module the hand gestures will be trained and predicted based on the different features extracted from images. Gestures will be recognized by using webcam as input and predicts the range as output.

4.5 Steps to execute/run/implement the project

4.5.1 Collection of datasets

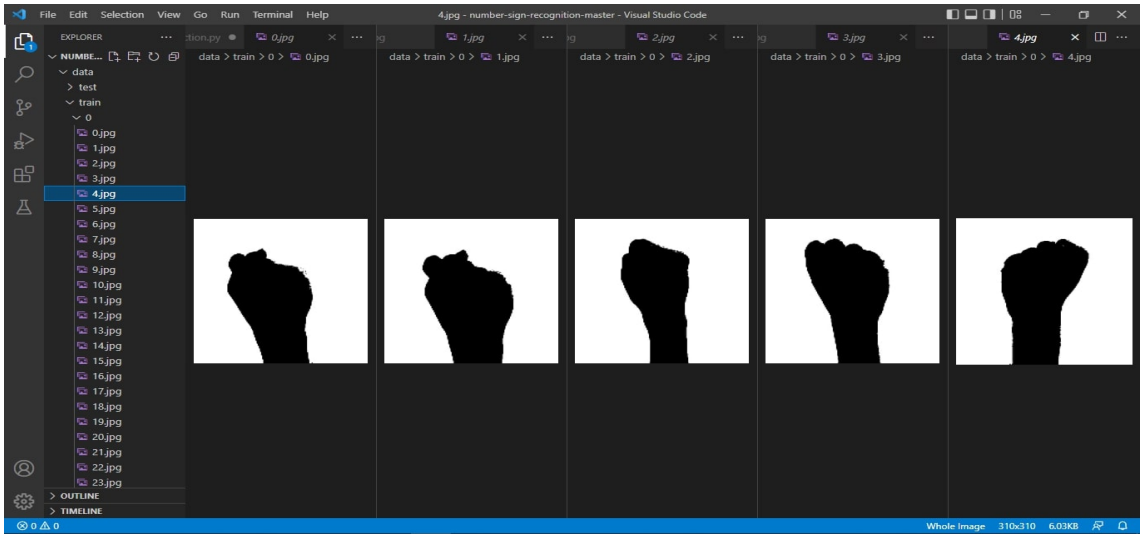


Figure 4.10: **Dataset for zero gesture**

Here we have collected a data for hand recognition symbol of number zero. We have taken 100 data samples as a dataset as shown in the above image. It will be used as data further in our project.

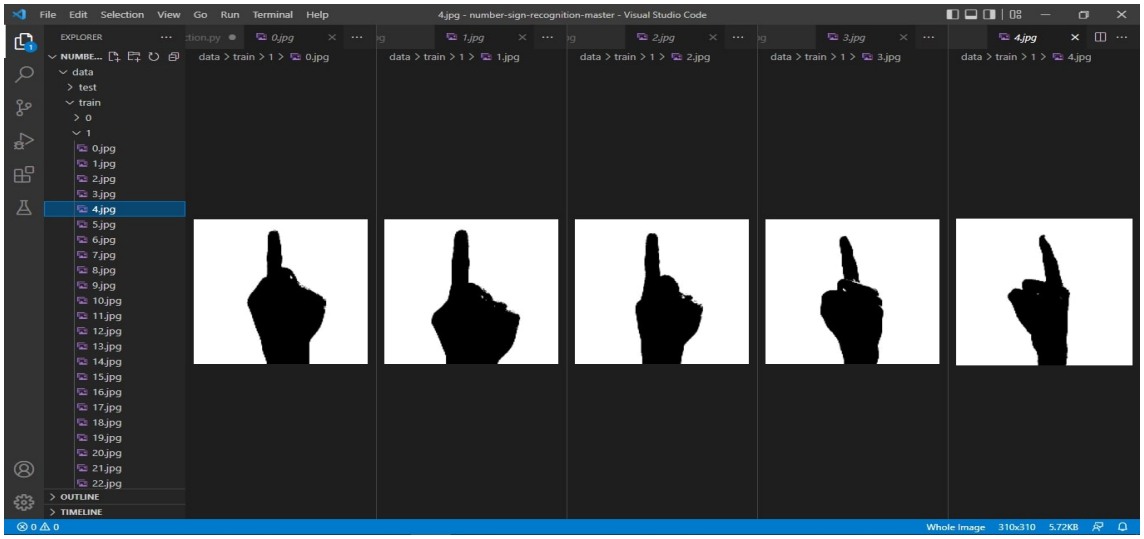


Figure 4.11: **Dataset for one gesture**

Here we have collected a data for hand recognition symbol of number one. We have taken 100 data samples as a dataset as shown in the above image. It will be used as data further in our project.

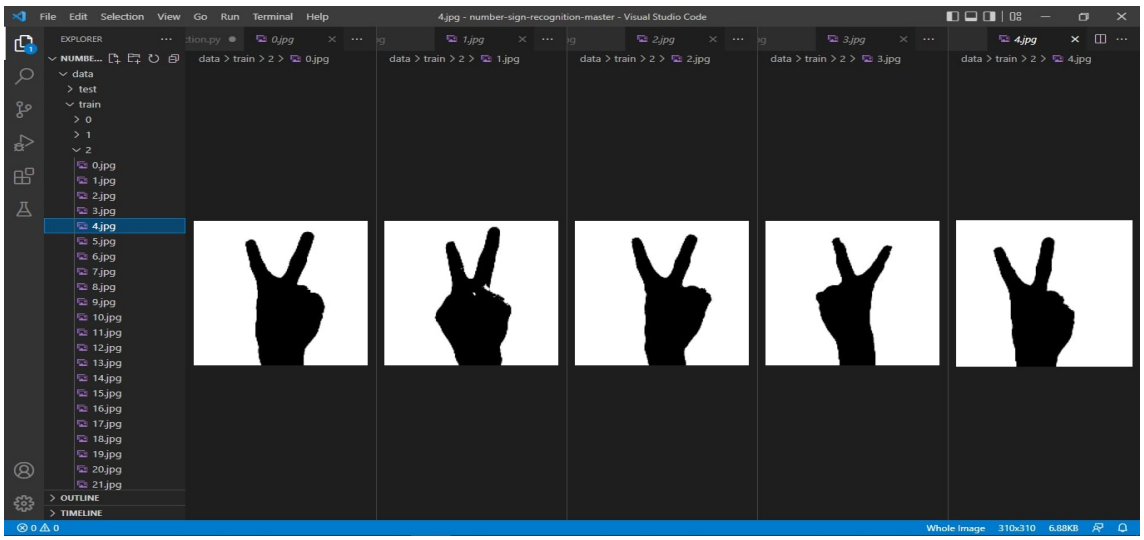


Figure 4.12: **Dataset for two gestrue**

Here we have collected a data for hand recognition symbol of number two. We have taken 100 data samples as a dataset as shown in the above image. It will be used as data further in our project.

4.5.2 Predicting the data

In this step we are recognizing the hand gestures by using webcam and result will be displayed in screen

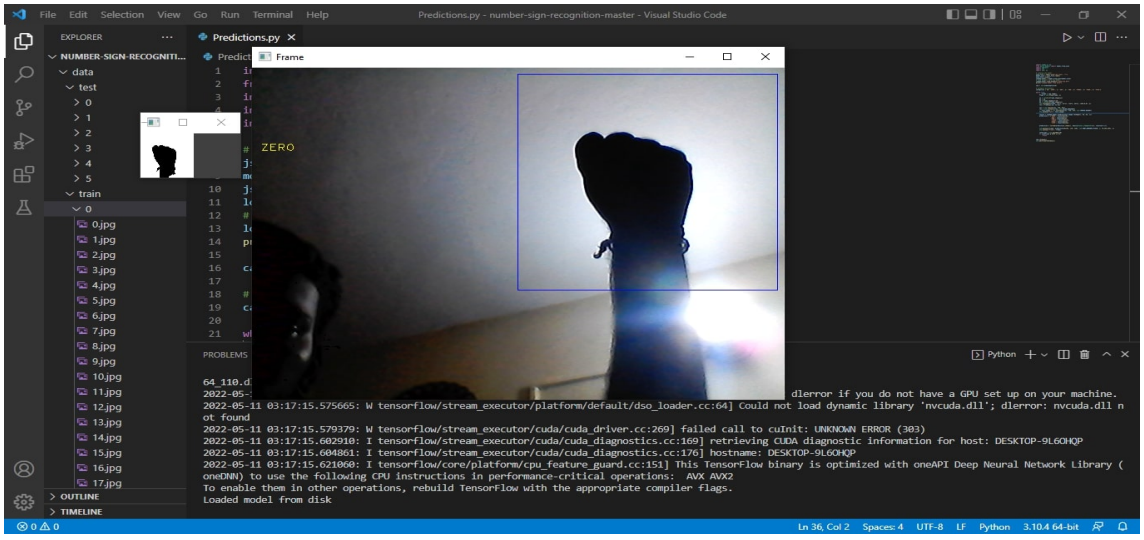


Figure 4.13: **Zero gesture detection**

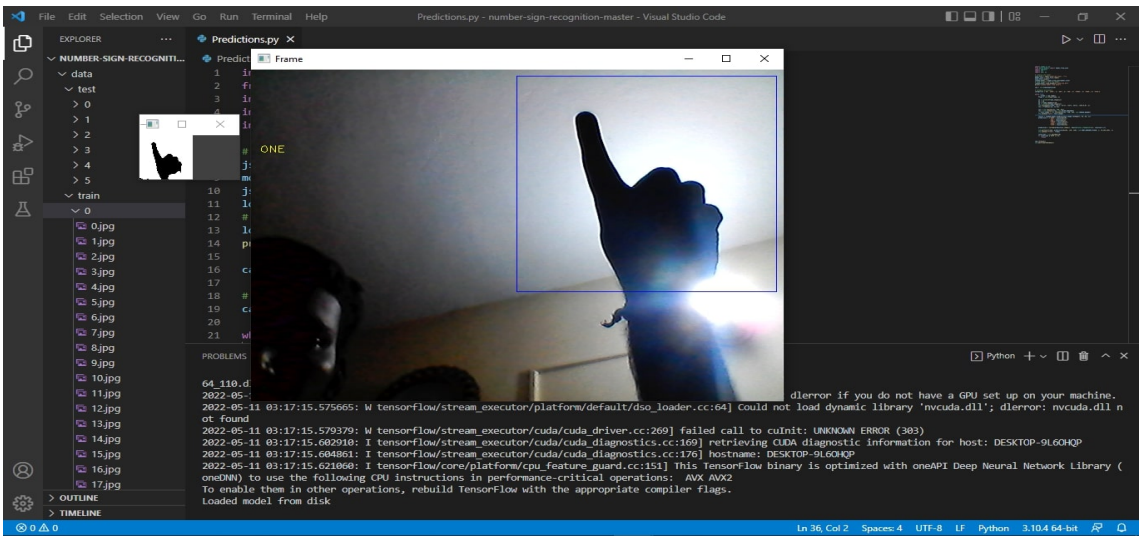


Figure 4.14: **One gesture detection**

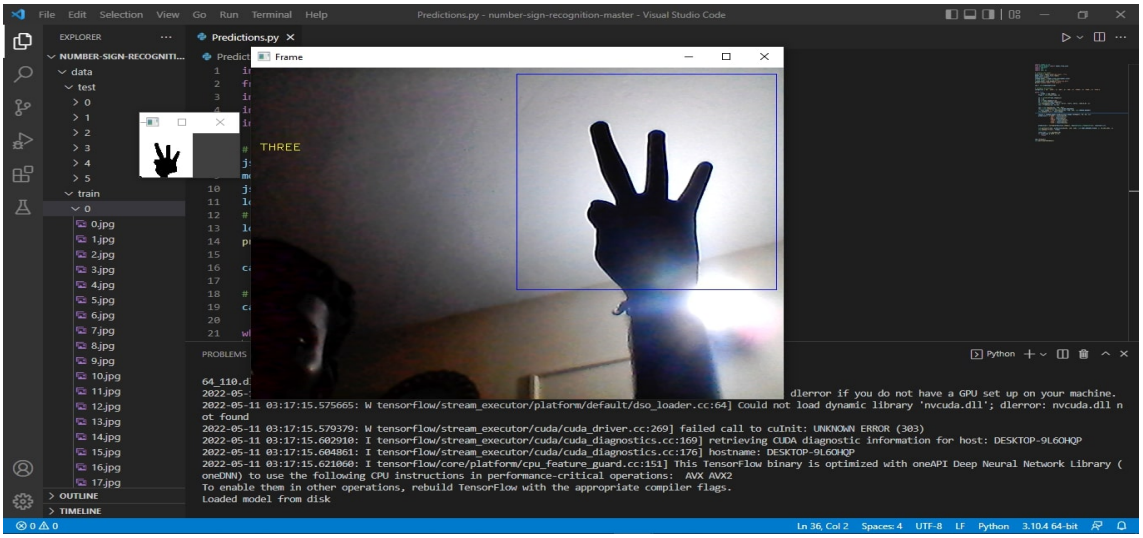


Figure 4.15: **Three gesture detection**

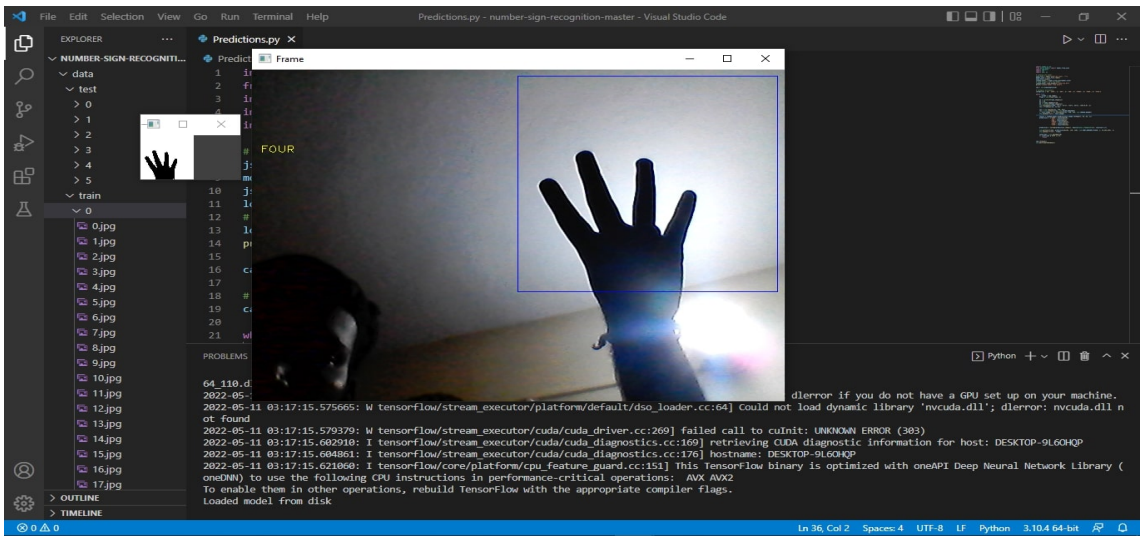


Figure 4.16: **Four gesture dectection**

Here we are giving hand gestures as input in the webcam and it will be recognized based on datasets and it will make predictions and results output in frame.

4.5.3 Implementation of training and prediction

```
# Importing the Keras libraries and packages
from keras.models import Sequential
from keras.layers import Convolution2D
from keras.layers import MaxPooling2D
from keras.layers import Flatten
from keras.layers import Dense

# Initializing the CNN
classifier = Sequential()

# First convolution layer and pooling
classifier.add(Convolution2D(32, (3, 3), input_shape=(64, 64, 1), activation='relu'))
classifier.add(MaxPooling2D(pool_size=(2, 2)))
# Second convolution layer and pooling
classifier.add(Convolution2D(32, (3, 3), activation='relu'))
classifier.add(MaxPooling2D(pool_size=(2, 2)))


classifier.add(Flatten())

classifier.add(Dense(units=128, activation='relu'))
classifier.add(Dense(units=6, activation='softmax'))


classifier.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])
from keras.preprocessing.image import ImageDataGenerator

train_datagen = ImageDataGenerator(
        rescale=1./255,
        shear_range=0.2,
        zoom_range=0.2,
        horizontal_flip=True)

test_datagen = ImageDataGenerator(rescale=1./255)
```

Figure 4.17: **Code for training**

```
Training.py > ...
34  test_datagen = ImageDataGenerator(rescale=1./255)
35
36  training_set = train_datagen.flow_from_directory('data/train',
37                                                   target_size=(64, 64),
38                                                   batch_size=5,
39                                                   color_mode='grayscale',
40                                                   class_mode='categorical')
41
42  test_set = test_datagen.flow_from_directory('data/test',
43                                              target_size=(64, 64),
44                                              batch_size=5,
45                                              color_mode='grayscale',
46                                              class_mode='categorical')
47  classifier.fit_generator(
48          training_set,
49          steps_per_epoch=600,
50          epochs=10,
51          validation_data=test_set,
52          validation_steps=30)
53
54
55
56  model_json = classifier.to_json()
57  with open("model-bw.json", "w") as json_file:
58      json_file.write(model_json)
59  classifier.save_weights('model-bw.h5')
60
61
```

Figure 4.18: **Code for training**

Chapter 5

IMPLEMENTATION AND TESTING

5.1 Input and Output

5.1.1 Input Design

Hand gestures are showing by user will be taken as input for predicting the similar hand gesture data.

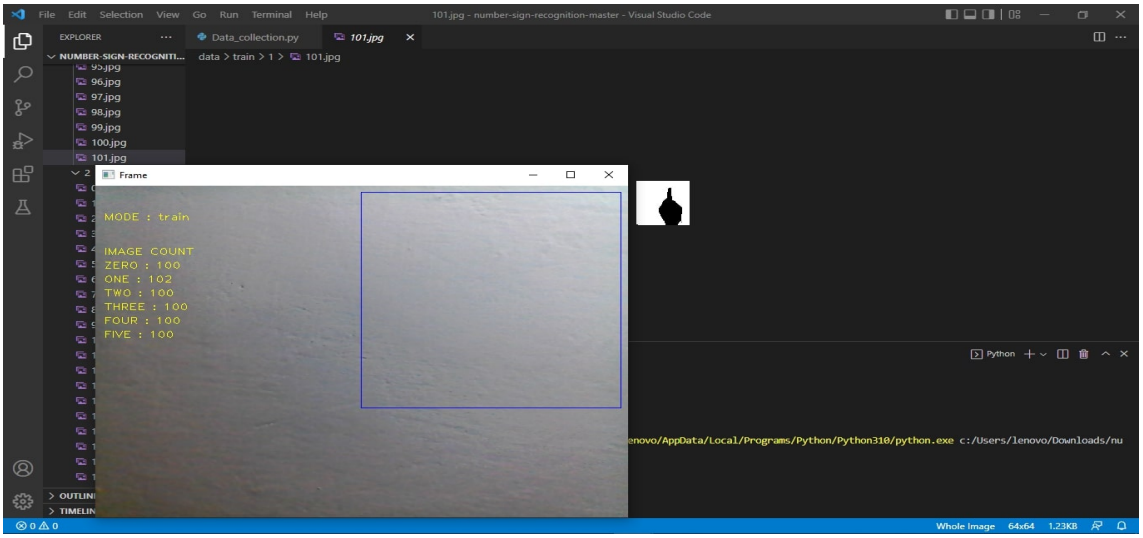


Figure 5.1: **Input**

5.1.2 Output Design

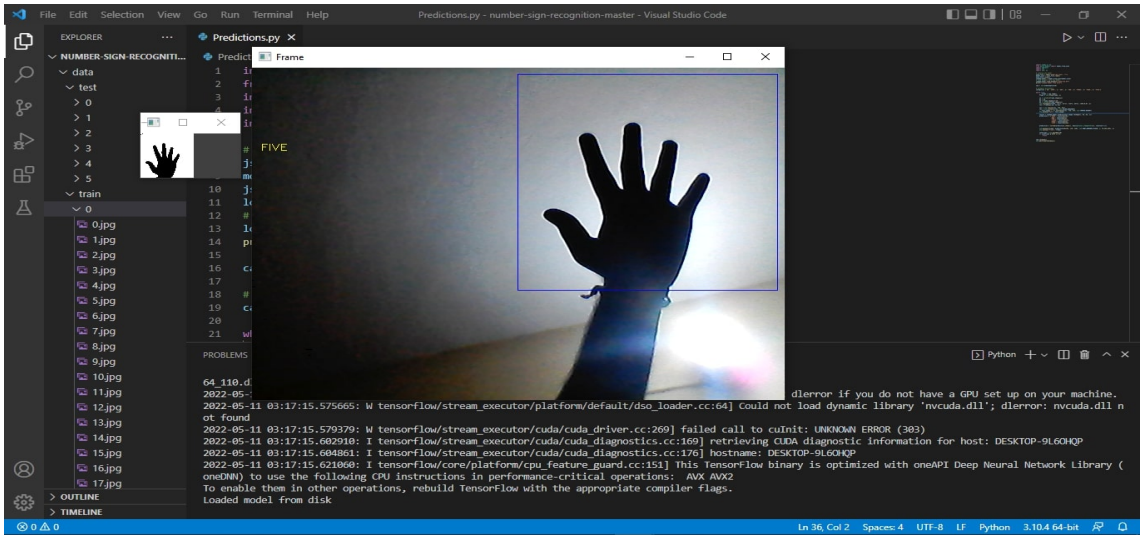


Figure 5.2: **Output**

5.2 Testing

Testing is process of executing programs with the intention of finding out the errors. During this process, the project is executed with set of the test called test set and the output of the object detector is evaluated to determine whether the project is performing as expected or not. It makes a logical assumption that if all the parts of the module are correct, then goal will be successfully achieved. Testing includes after the completion of the coding phase. The project is tested from the very beginning and also at each and every step by entering different type of data. In the testing phase, some errors are found, which did not come across to knowledge of programmer while coding phase.

5.3 Types of Testing

5.3.1 Unit testing

In unit testing, each and every block of code is tested independently such that the cases are passed. Each and every block of code is tested for all boundary conditions which are possible for the project. The blocks are not combined and blocks which require calls to other blocks are combined together into a single unit and tested with and without providing inputs according to the requirement of each block. Here we

are tested the region of interest of frame for giving input as hand gestures and prediction of the hand gestures.

Test result

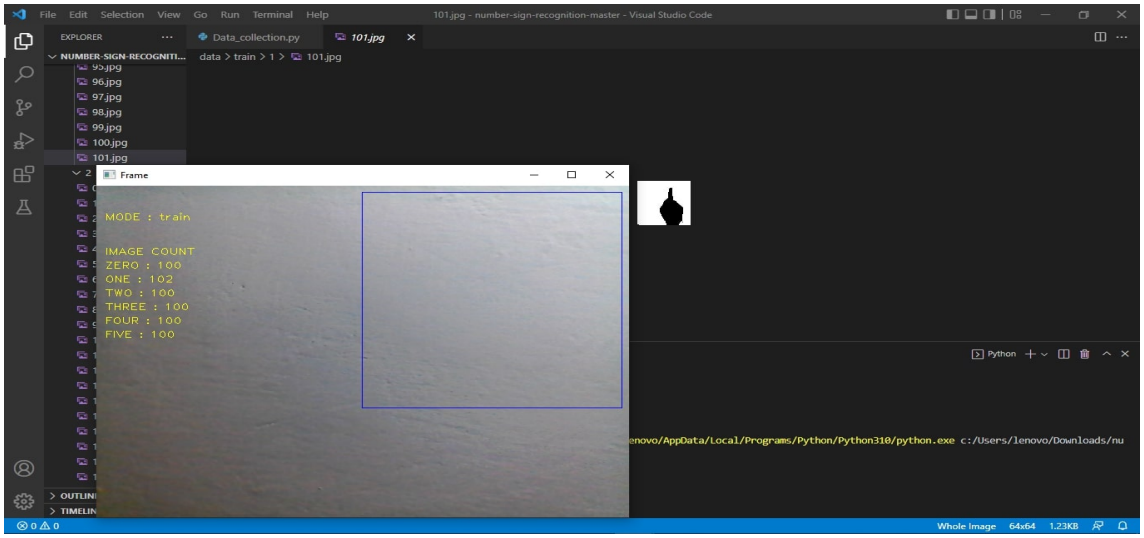


Figure 5.3: **unit test**

5.3.2 Integration testing

Integration Testing checks the data flow from one module to another module. Integration testing is conducted to evaluate the compliance of a system or component with specified functional requirements.Here prediction flow from one gesture to another gesture is tested and its matches the test results.

Test result

```
classifier.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])
```

Figure 5.4: **Integration test**

5.3.3 System testing

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation and user manuals. Here we are tested specific function handled for detecting features from hand gesture images.

Test Result

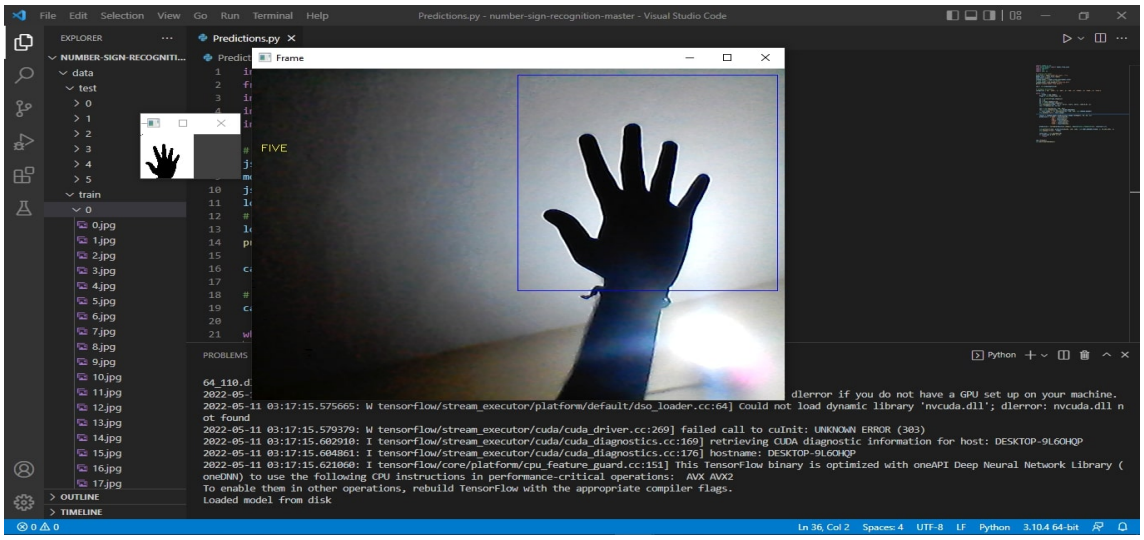


Figure 5.5: **System test**

Chapter 6

RESULTS AND DISCUSSIONS

6.1 Efficiency of the Proposed System

In our proposed system we developed a model for disability of deaf and dumb people. Here we are trying to predict hand gestures by using deep learning algorithm. Results will be predicted based on the features of different hand gesture images. It does not require much cost. It is cost efficiency.

6.2 Comparison of Existing and Proposed System

In the existing system there are several methodologies implemented in the way of recognizing skeleton and joints, visual based recognitions. In our proposed system we are implementing hand gestures detections by using some image features.

6.3 Sample Code

```
import cv2
import numpy as np
import os

if not os.path.exists("data"):
    os.makedirs("data")
    os.makedirs("data/train")
    os.makedirs("data/test")
    os.makedirs("data/train/0")
    os.makedirs("data/train/1")
    os.makedirs("data/train/2")
    os.makedirs("data/train/3")
    os.makedirs("data/train/4")
    os.makedirs("data/train/5")
    os.makedirs("data/test/0")
    os.makedirs("data/test/1")
    os.makedirs("data/test/2")
    os.makedirs("data/test/3")
    os.makedirs("data/test/4")
```

```
    os.makedirs("data/test/5")


mode = 'train'
directory = 'data/'+mode+'/'

cap = cv2.VideoCapture(0)

while True:
    _, frame = cap.read()
    frame = cv2.flip(frame, 1)

    count = {'zero': len(os.listdir(directory+"/0")),
             'one': len(os.listdir(directory+"/1")),
             'two': len(os.listdir(directory+"/2")),
             'three': len(os.listdir(directory+"/3")),
             'four': len(os.listdir(directory+"/4")),
             'five': len(os.listdir(directory+"/5"))}

    cv2.putText(frame, "MODE : "+mode, (10, 50), cv2.FONT_HERSHEY_PLAIN, 1, (0,255,255), 1)
    cv2.putText(frame, "IMAGE COUNT", (10, 100), cv2.FONT_HERSHEY_PLAIN, 1, (0,255,255), 1)
    cv2.putText(frame, "ZERO : "+str(count['zero']), (10, 120), cv2.FONT_HERSHEY_PLAIN, 1,
        (0,255,255), 1)
    cv2.putText(frame, "ONE : "+str(count['one']), (10, 140), cv2.FONT_HERSHEY_PLAIN, 1, (0,255,255)
        , 1)
    cv2.putText(frame, "TWO : "+str(count['two']), (10, 160), cv2.FONT_HERSHEY_PLAIN, 1, (0,255,255)
        , 1)
    cv2.putText(frame, "THREE : "+str(count['three']), (10, 180), cv2.FONT_HERSHEY_PLAIN, 1,
        (0,255,255), 1)
    cv2.putText(frame, "FOUR : "+str(count['four']), (10, 200), cv2.FONT_HERSHEY_PLAIN, 1,
        (0,255,255), 1)
    cv2.putText(frame, "FIVE : "+str(count['five']), (10, 220), cv2.FONT_HERSHEY_PLAIN, 1,
        (0,255,255), 1)

    x1 = int(0.5*frame.shape[1])
    y1 = 10
    x2 = frame.shape[1]-10
    y2 = int(0.5*frame.shape[1])
    cv2.rectangle(frame, (x1-1, y1-1), (x2+1, y2+1), (255,0,0) ,1)
    roi = frame[y1:y2, x1:x2]
    roi = cv2.resize(roi, (64, 64))

    cv2.imshow("Frame", frame)

    roi = cv2.cvtColor(roi, cv2.COLOR_BGR2GRAY)
    _, roi = cv2.threshold(roi, 120, 255, cv2.THRESH_BINARY)
    cv2.imshow("ROI", roi)
```

Output

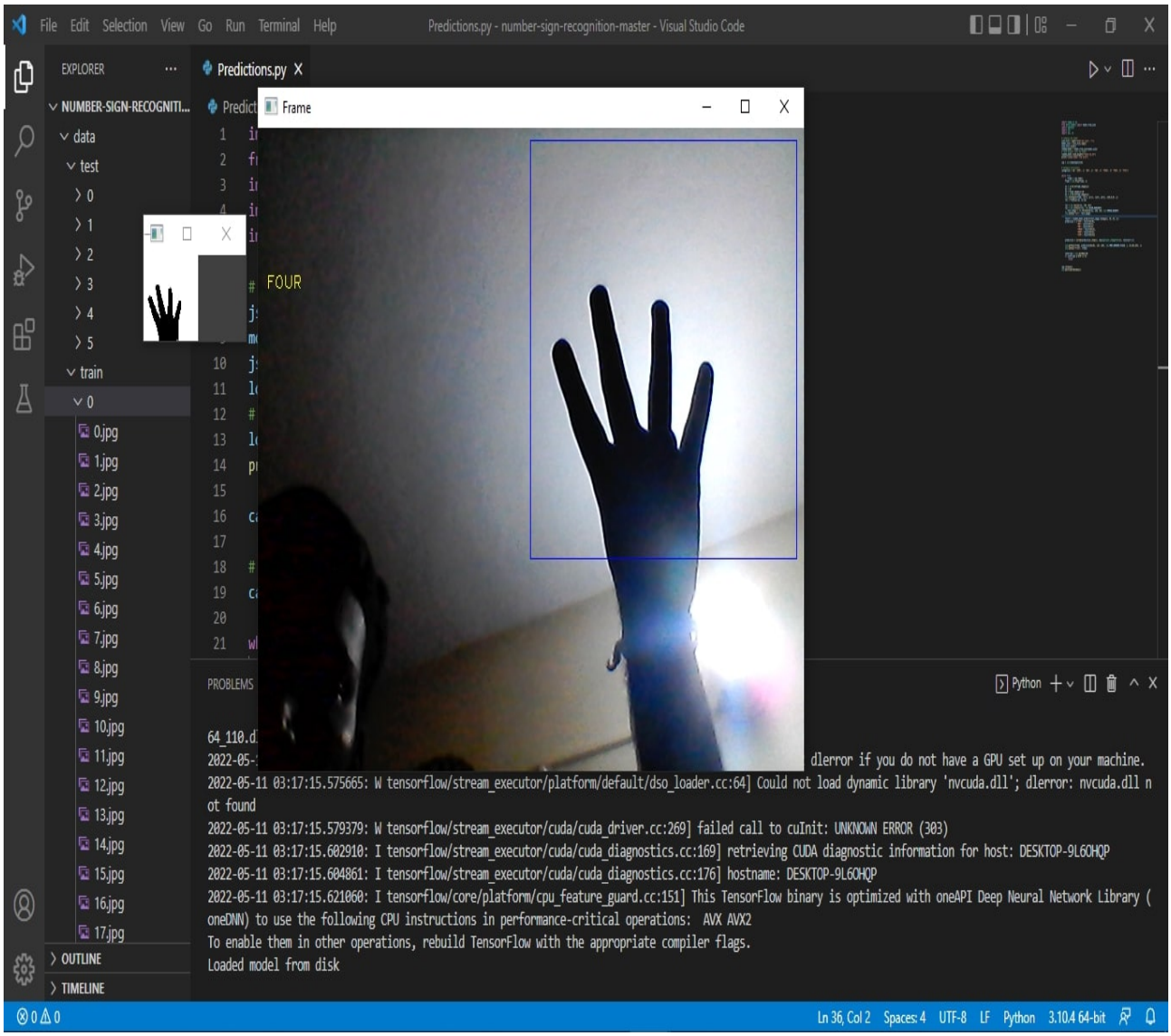


Figure 6.1: **Detection of hand gesture four**

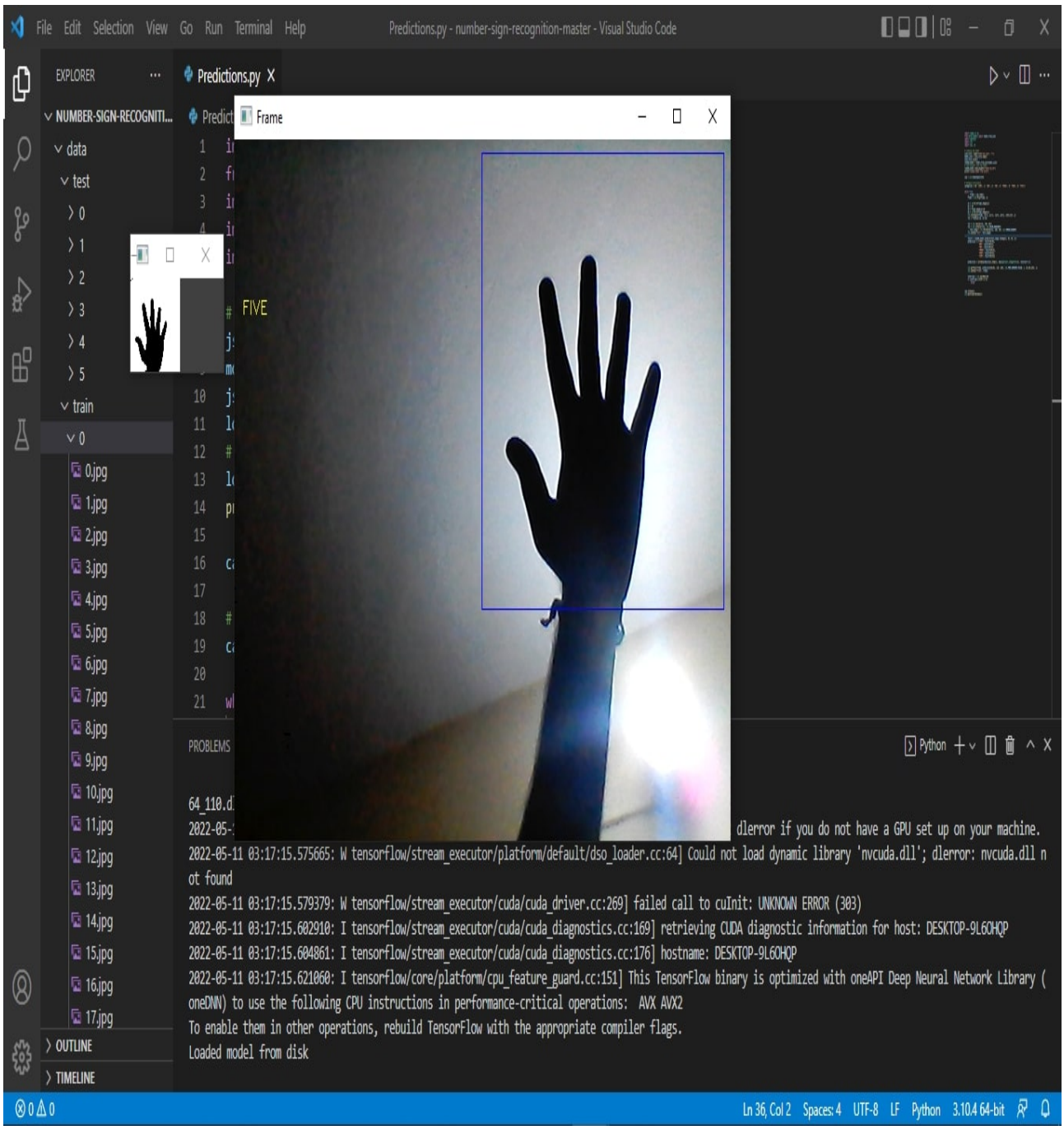


Figure 6.2: **Detection of hand gesture five**

Chapter 7

CONCLUSION AND FUTURE ENHANCEMENTS

7.1 Conclusion

We have collected and creating a hand gestures dataset as images to train the dataset using required algorithm. We will implement the process further by analysing the algorithm and by using the dataset. We will make predictions by using the pre-trained dataset. In upcoming we train the different prediction dataset for sign language recognition.

7.2 Future Enhancements

In our project we are dectecting the particular handgestures and training the dataset and make predictions. In future process we are trying to implement different hand gestures and other gestures. We will implement our project more efficiently with more improvments for future purposes.

INDUSTRY DETAILS

7.3 Industry name

Wipro HR Services India Private limited.

7.3.1 Duration of Internship (From Date - To Date)

Full time

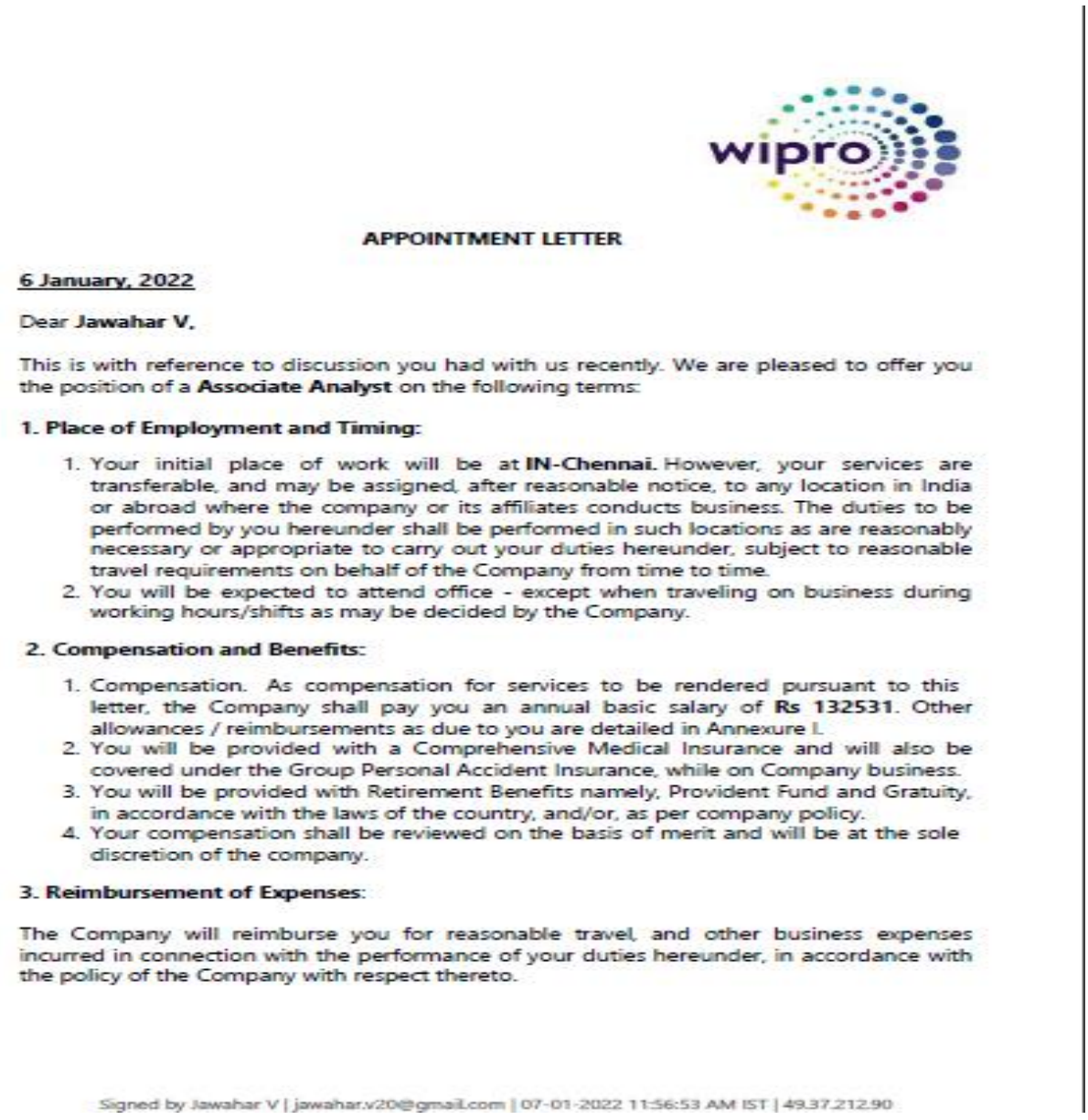
7.3.2 Duration of Internship in months

From 10th January

7.3.3 Industry Address

2683+4CH, olympia technology park, Altius block, Guindy, Chennai, Tamil Nadu 600032

7.4 Internship offer letter



APPOINTMENT LETTER

6 January, 2022

Dear **Jawahar V,**

This is with reference to discussion you had with us recently. We are pleased to offer you the position of a **Associate Analyst** on the following terms:

1. Place of Employment and Timing:

1. Your initial place of work will be at **IN-Chennai.** However, your services are transferable, and may be assigned, after reasonable notice, to any location in India or abroad where the company or its affiliates conducts business. The duties to be performed by you hereunder shall be performed in such locations as are reasonably necessary or appropriate to carry out your duties hereunder, subject to reasonable travel requirements on behalf of the Company from time to time.
2. You will be expected to attend office - except when traveling on business during working hours/shifts as may be decided by the Company.

2. Compensation and Benefits:

1. Compensation. As compensation for services to be rendered pursuant to this letter, the Company shall pay you an annual basic salary of **Rs 132531**. Other allowances / reimbursements as due to you are detailed in Annexure I.
2. You will be provided with a Comprehensive Medical Insurance and will also be covered under the Group Personal Accident Insurance, while on Company business.
3. You will be provided with Retirement Benefits namely, Provident Fund and Gratuity, in accordance with the laws of the country, and/or, as per company policy.
4. Your compensation shall be reviewed on the basis of merit and will be at the sole discretion of the company.

3. Reimbursement of Expenses:

The Company will reimburse you for reasonable travel, and other business expenses incurred in connection with the performance of your duties hereunder, in accordance with the policy of the Company with respect thereto.

Signed by Jawahar V | jawahar.v20@gmail.com | 07-01-2022 11:56:53 AM IST | 49.37.212.90

APPOINTMENT LETTER

7 January, 2022

Dear **D SANTHAKUMAR,**

This is with reference to discussion you had with us recently. We are pleased to offer you the position of a **Associate Analyst** on the following terms:

1. Place of Employment and Timing:

1. Your initial place of work will be at **IN-Chennai.** However, your services are transferable, and may be assigned, after reasonable notice, to any location in India or abroad where the company or its affiliates conducts business. The duties to be performed by you hereunder shall be performed in such locations as are reasonably necessary or appropriate to carry out your duties hereunder, subject to reasonable travel requirements on behalf of the Company from time to time.
2. You will be expected to attend office - except when traveling on business during working hours/shifts as may be decided by the Company.

2. Compensation and Benefits:

1. Compensation. As compensation for services to be rendered pursuant to this letter, the Company shall pay you an annual basic salary of **Rs 132531**. Other allowances / reimbursements as due to you are detailed in Annexure I.
2. You will be provided with a Comprehensive Medical Insurance and will also be covered under the Group Personal Accident Insurance, while on Company business.
3. You will be provided with Retirement Benefits namely, Provident Fund and Gratuity, in accordance with the laws of the country, and/or, as per company policy.
4. Your compensation shall be reviewed on the basis of merit and will be at the sole discretion of the company.

3. Reimbursement of Expenses:

The Company will reimburse you for reasonable travel, and other business expenses incurred in connection with the performance of your duties hereunder, in accordance with the policy of the Company with respect thereto.

Signed by D SANTHAKUMAR | santhajohn2000@gmail.com | 08-01-2022 06:50:38 AM IST | 157.51.152.50

APPOINTMENT LETTER

8 March, 2022

Dear **Yasam Reddy,**

This is with reference to discussion you had with us recently. We are pleased to offer you the position of a **Associate Analyst** on the following terms:

1. Place of Employment and Timing:

1. Your initial place of work will be at **IN-TN-Chennai.** However, your services are transferable, and may be assigned, after reasonable notice, to any location in India or abroad where the company or its affiliates conducts business. The duties to be performed by you hereunder shall be performed in such locations as are reasonably necessary or appropriate to carry out your duties hereunder, subject to reasonable travel requirements on behalf of the Company from time to time.
2. You will be expected to attend office - except when traveling on business during working hours/shifts as may be decided by the Company.

2. Compensation and Benefits:

1. Compensation. As compensation for services to be rendered pursuant to this letter, the Company shall pay you an annual basic salary of **Rs 132531**. Other allowances / reimbursements as due to you are detailed in Annexure I.
2. You will be provided with a Comprehensive Medical Insurance and will also be covered under the Group Personal Accident Insurance, while on Company business.
3. You will be provided with Retirement Benefits namely, Provident Fund and Gratuity, in accordance with the laws of the country, and/or, as per company policy.
4. Your compensation shall be reviewed on the basis of merit and will be at the sole discretion of the company.

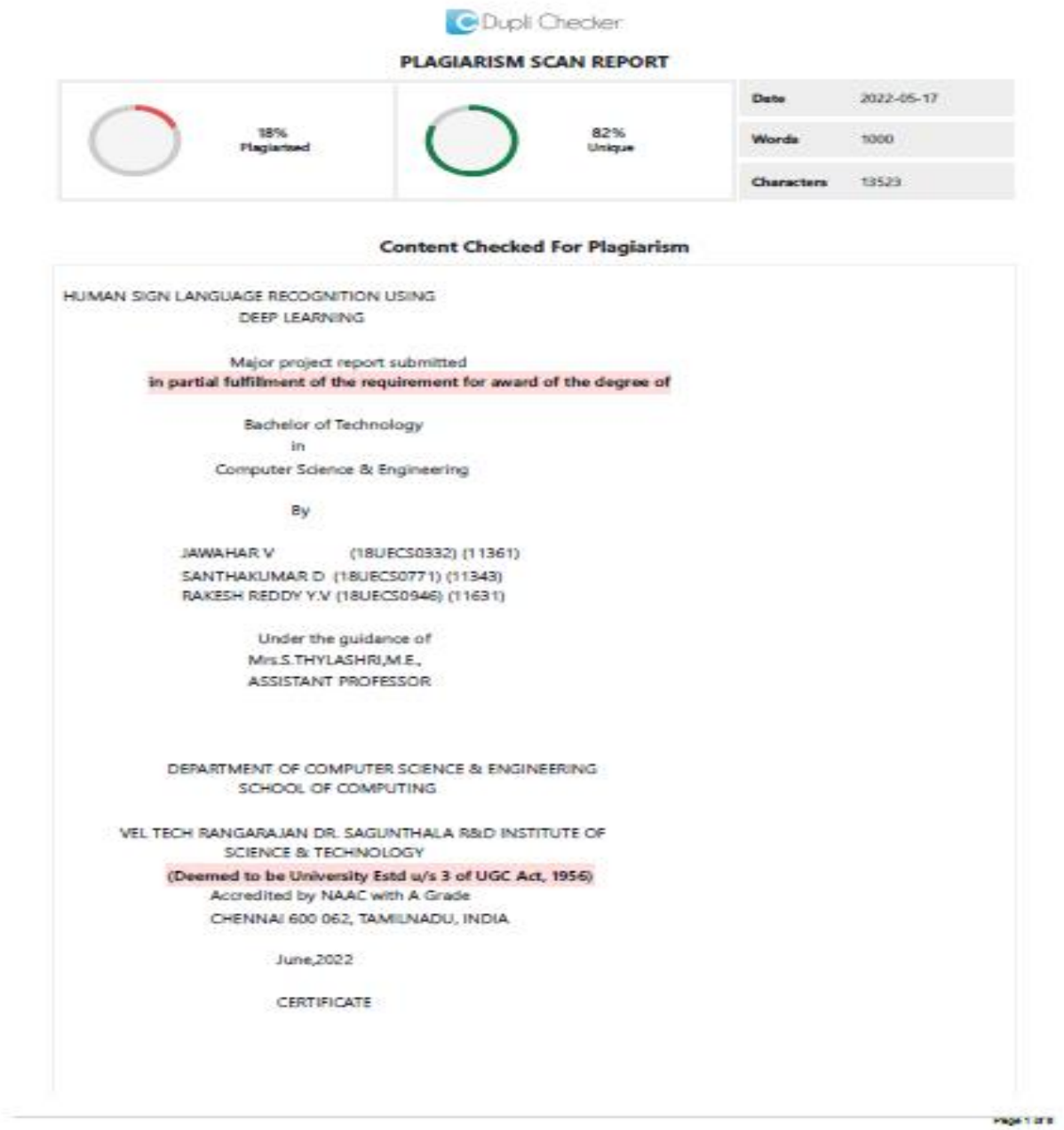
3. Reimbursement of Expenses:

The Company will reimburse you for reasonable travel, and other business expenses incurred in connection with the performance of your duties hereunder, in accordance with the policy of the Company with respect thereto.

Signed by Yasam Reddy | yvrakeshreddy9955@gmail.com | 08-09-2022 10:29:44 PM IST | 115.240.192.142

Chapter 8

PLAGIARISM REPORT



Dupli Checker

PLAGIARISM SCAN REPORT

18% Plagiarised	82% Unique
Date	2022-05-17
Words	1000
Characters	13523

Content Checked For Plagiarism

HUMAN SIGN LANGUAGE RECOGNITION USING
DEEP LEARNING

Major project report submitted
in partial fulfillment of the requirement for award of the degree of

Bachelor of Technology
in
Computer Science & Engineering

By

JAWAHAR V (18UECS0332) (11361)
SANTHAKUMAR D (18UECS0771) (11343)
RAKESH REDDY Y.V (18UECS0946) (11631)

Under the guidance of
Mrs.S.THYLASHRI,M.E.,
ASSISTANT PROFESSOR

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
SCHOOL OF COMPUTING

VEL TECH RANGARAJAN DR. SAGUNTHALA R&D INSTITUTE OF
SCIENCE & TECHNOLOGY
(Deemed to be University Estd u/s 3 of UGC Act, 1956)
Accredited by NAAC with A Grade
CHENNAI 600 062, TAMILNADU, INDIA

June,2022

CERTIFICATE

Figure 8.1: **Detection of hand gesture five**

Chapter 9

SOURCE CODE & POSTER PRESENTATION

9.1 Source Code

```
import numpy as np
from keras.models import model_from_json
import operator
import cv2
import sys, os

# Loading the model
json_file = open("model-bw.json", "r")
model_json = json_file.read()
json_file.close()
loaded_model = model_from_json(model_json)
loaded_model.load_weights("model-bw.h5")
print("Loaded model from disk")

cap = cv2.VideoCapture(0)

# Category dictionary
categories = {0: 'ZERO', 1: 'ONE', 2: 'TWO', 3: 'THREE', 4: 'FOUR', 5: 'FIVE'}

while True:
    _, frame = cap.read()
    frame = cv2.flip(frame, 1)

    x1 = int(0.5*frame.shape[1])
    y1 = 10
    x2 = frame.shape[1]-10
    y2 = int(0.5*frame.shape[1])
    cv2.rectangle(frame, (x1-1, y1-1), (x2+1, y2+1), (255,0,0) ,1)
    roi = frame[y1:y2, x1:x2]

    roi = cv2.resize(roi, (64, 64))
    roi = cv2.cvtColor(roi, cv2.COLOR_BGR2GRAY)
    _, test_image = cv2.threshold(roi, 120, 255, cv2.THRESH_BINARY)
    cv2.imshow("test", test_image)

```

```
    result = loaded_model.predict(test_image.reshape(1, 64, 64, 1))
    prediction = {'ZERO': result[0][0],
                  'ONE': result[0][1],
                  'TWO': result[0][2],
                  'THREE': result[0][3],
                  'FOUR': result[0][4],
                  'FIVE': result[0][5]}

    prediction = sorted(prediction.items(), key=operator.itemgetter(1), reverse=True)

    cv2.putText(frame, prediction[0][0], (10, 120), cv2.FONT_HERSHEY_PLAIN, 1, (0,255,255), 1)
    cv2.imshow("Frame", frame)

    interrupt = cv2.waitKey(10)
    if interrupt & 0xFF == 27:
        break


cap.release()
cv2.destroyAllWindows()



# Importing the Keras libraries and packages
from keras.models import Sequential
from keras.layers import Convolution2D
from keras.layers import MaxPooling2D
from keras.layers import Flatten
from keras.layers import Dense

classifier = Sequential()
classifier.add(Convolution2D(32, (3, 3), input_shape=(64, 64, 1), activation='relu'))
classifier.add(MaxPooling2D(pool_size=(2, 2)))
classifier.add(Convolution2D(32, (3, 3), activation='relu'))
classifier.add(MaxPooling2D(pool_size=(2, 2)))
classifier.add(Flatten())
classifier.add(Dense(units=128, activation='relu'))
classifier.add(Dense(units=6, activation='softmax'))
classifier.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])
from keras.preprocessing.image import ImageDataGenerator

train_datagen = ImageDataGenerator(
        rescale=1./255,
        shear_range=0.2,
        zoom_range=0.2,
        horizontal_flip=True)

test_datagen = ImageDataGenerator(rescale=1./255)

training_set = train_datagen.flow_from_directory('data/train',
```

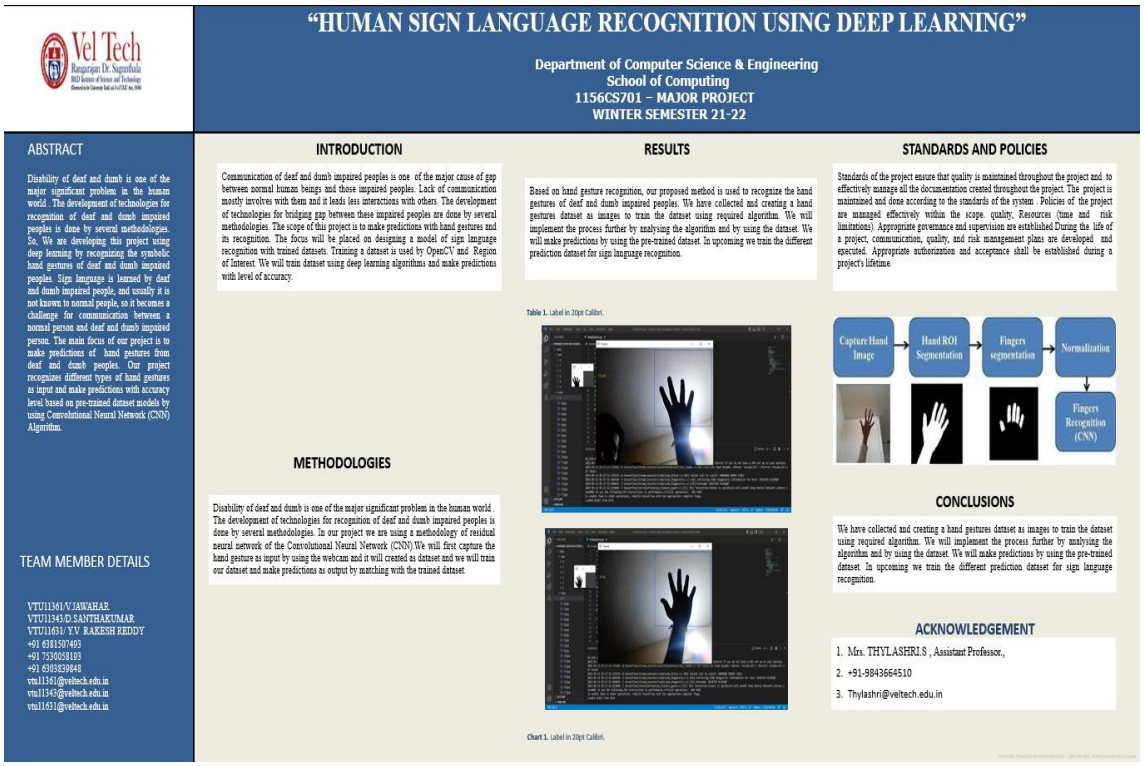
```
                                                 target_size=(64, 64),
                                                 batch_size=5,
                                                 color_mode='grayscale',
                                                 class_mode='categorical')

test_set = test_datagen.flow_from_directory('data/test',
                                            target_size=(64, 64),
                                            batch_size=5,
                                            color_mode='grayscale',
                                            class_mode='categorical')
classifier.fit_generator(
        training_set,
        steps_per_epoch=600,
        epochs=10,
        validation_data=test_set,
        validation_steps=30)



model_json = classifier.to_json()
with open("model-bw.json", "w") as json_file:
    json_file.write(model_json)
classifier.save_weights('model-bw.h5')
```

9.2 Poster Presentation



References

[1] Mahesh Kumar N B (2018). Conversion of Sign Language into text, International Journal of Applied Engineering research, 13(9), 7154-7161

[2] Bowen Shi1 , Aurora Martinez Del Rio2 , Jonathan Keane2 , Diane Brentari2 Greg Shakhnarovich1 , Karen Livescu1 (2019). Fingerspelling recognition in the wild with iterative visual attention, IEEE Xplore, Issue 28

[3] N. Alnaim and M. Abbod (2019). Hand Gesture Detection Using Neural Networks Algorithms, International Journal of Maching Learning and Computing 9(6)

[4] Mehreen Hurroo and Mohammad Elham Walizad (2020). Sign Language Recognition System using Convolutional Neural Network and Computer Vision, International Journal of Engineering Research, 9(12), 2278 -0181

[5] Dongxu Li , Cristian Rodriguez Opazo, Xin Yu, Hongdong Li (2020). Word-level Deep Sign Language Recognition from Video: A New Large-scale Dataset and Methods Comparison, IEEE Xplore

[6] Songyao Jiang, Bin Sun, Lichen Wang, Yue Bai, Kunpeng Li and Yun Fu (2021). Sign Language Recognition via Slketon-Aware Multi-Model Ensemble, IEEE Xplore

[7] Necati Cihan Camgoz1 , Simon Oscar Koller2 , Hermann Ney2 , Richard Bowden1 (2021). Neural Sign Language Translation, IEEE Xplore