A Mid-Term Progress Report

On

"A NOVEL APPROACH TO CONVERT SIGN LANGUAGE TO TEXT IN A CLIENT- SERVER ENVIRONMENT"

Submitted in partial fulfillment of the requirements for the award of the degree of

Bachelor of Technology

In

"Computer Science and Engineering"

SUBMITTED BY: G-7

GUNEET KOHLI(1805172)

HARNEET KAUR(1805182)

JASHANPREET KAUR(1805188)

Project Guide: Dr Sumeet Kaur Sehra

GURU NANAK DEV ENGINEERING COLLEGE, LUDHIANA 141006

INDEX

CHAPTER 1
INTRODUCTION

Sign Language is the most natural and expressive way for the hearing impaired people. Sign languages are used by deaf people for communication. In sign languages, humans use hand gestures, body, facial expressions and movements to convey meaning. Humans can easily learn and understand sign languages, but automatic sign language recognition for machines is a challenging task. Thus, to put an end to this, automatic sign language converters are needed today.

It is very difficult for the deaf people to communicate with the hearing person and there are not many options available to help them. And all of these alternatives have some major flaws. Interpreters are not usually available and are expensive. Pen and paper is also not a good idea, it is uncomfortable, messy and even time consuming, both for the deaf and the hearing person. With the evolution of IoT, everything around is getting automated.

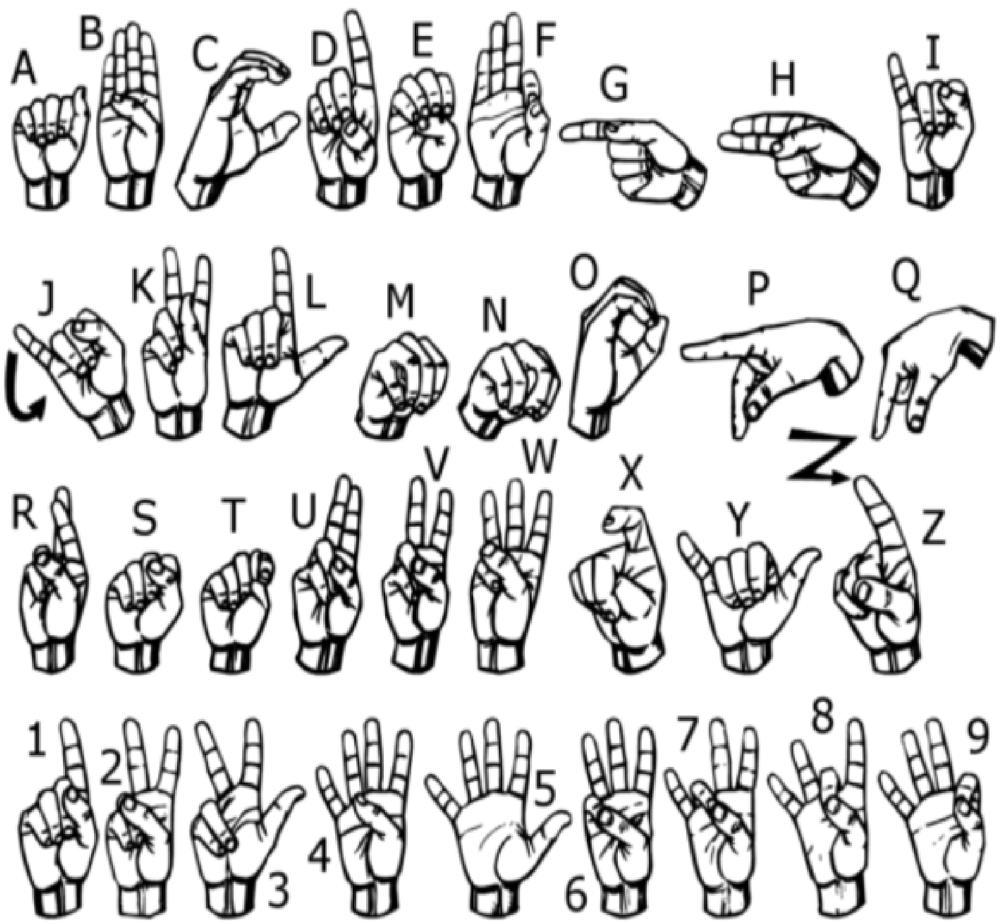


Fig 1.0: American Sign Language

The demand for Machine Learning and its applications is very high. The accuracy and efficiency of any algorithm and the model developed must be very high to make it useful.The knowledge of Machine Learning thus becomes very important. In the era of Machine Learning where everything is getting automated, the need for an Interpreter to translate Sign language to text is just a waste of resources. The classification of objects, object detection and image processing plays a very vital role. Thus, the main aim is to bridge the gap between the normal and the deaf and mute individuals by providing an automatic translation system.

Today, there are almost 2 million people classified as Deaf and Dumb. They have great difficulty in communicating with each other and with other individuals as the only means of communication is sign language. They need to learn this sign language. It is extremely difficult for a person who is unaware of this sign language to understand and decode their actions. It is impossible to identify anything without it's prior knowledge.

Even for computers, they need to have information in their memory to identify and provide data related to any object. The major technologies used are IMAGE PROCESSING and CNN (Convolutional Neural Networks). We design this software to bridge the communication gap between normal and deaf-mute individuals by providing an automatic translation system.

It is a simple CNN project which helps its users to convert the sign language directly into text using different types of gestures which can be converted into text on the basis of the standard American Sign Language. The project is trained on Keras. It will be using word identification and working mathematical operations. The major technologies used are IMAGE PROCESSING and CNN (Convolutional Neural Network).

The signs are captured by using a webcam. These signs are processed for feature extraction using some colour model. The extracted features are compared by using a pattern matching algorithm. In order to calculate the sign recognition, the features are compared with a testing database. Finally, the recognized gesture is converted into text. This system provides an opportunity for a deaf-dumb people to communicate with non-signing people without the need of an interpreter.

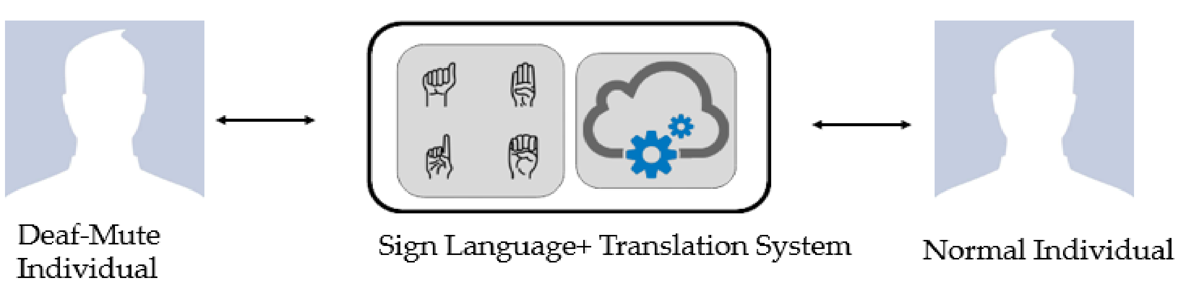


Fig 1.2: Proposed System

A number of images of some gestures are taken and processed to make the dataset.The CNN model is then trained using Keras on these captured and modified images.The signs to be translated are then fed to the software which matches it with the existing images and classifies it.

1.1 STUDY OF EXISTING SYSTEM AND ITS SHORTCOMINGS: Sign Language Recognition is not a new problem in the field of Computer Vision. In the 90's researchers used the Hidden Markov Approach for Sign Language Recognition because of their capability to capture temporal information.

Researchers have also used extra hardware sensors as feature extractors and use classifiers on the extracted features. Above mentioned work dealt with static data and was not real-time. With the increasing popularity of deep neural networks and their capability to provide high performance, many researchers started using these new deep-learning techniques for sign language recognition which gave them better performance but still, they were dealing with static data. Recently, the world has achieved real-time sign recognition performance with complex convolutional neural networks and they have worked on dynamic data. Deep Neural Network methods achieve high accuracy; however, they need a lot of training data.
Previous literature has few shortcomings that we address with our work:

- Introducing a new deep-learning architecture which achieves comparable results with limited training data
- Automate sign-language recognition for isolated signs

1.2 OBJECTIVES

- To create gesture samples using OpenCV.
- To classify the gestures and label them with one of the various categories already defined while training the dataset
- To be able to identify sign language characters individually and as words when concatenated

CHAPTER 2
SYSTEM REQUIREMENTS

2.1 SOFTWARE

Table 2.1 : Software Requirements

Software	Minimum Requirements
Python	Version 3.0 or Higher
Operating System	Single Core 1.0 GHzor Higher
Tensorflow	Version 2.2 or Higher
Keras	Version 2.3.0 or Higher
OpenCV	Version 3.4 or Higher
Graphics Card	64Mb or Higher
Disk Space	650 MB or Higher
RAM	800MB or Higher
GNS3	Version 2.2.0 or Higher

2.2 HARDWARE

Table 2.2 : Hardware Requirements

Hardware	Minimum Requirements
Any Computing Device	2 GHz minimum, multi-core processor
Disk Space	At least 1 GB
Memory (RAM)	At least 2GB, preferably higher
Camera	2 MegaPixel or Higher

CHAPTER 3
SOFTWARE REQUIREMENTS ANALYSIS

3.1 PROBLEM FORMULATION

Given a query video say Q we want to find its corresponding sign S.

Our final goal is to learn a function A: Q →S that maps any query video Q belonging to a domain X to its corresponding sign S. Query video Q is a sequence of frames = ($q1$, $q2$, … , qt). This becomes a many-to-one problem because we want to map multiple inputs in a sequence to a single output.

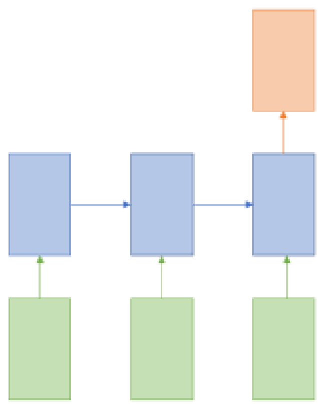


Fig 3.0 : Many-to-One Problem

So, we can rewrite learning function A: Q → S to A: (q_1, q_2, … , q_t) →S where we simply substitute Q to (q_1, q_2, … , q_t).

We want a first model, that we call Model-1, to learn a function E: Q → F that maps input sequence (q_1, q_2, … , q_t) to feature sequence (f_1, f_2, … , f_t) by minimizing the Mean Squared Error loss function: $M.\ S.\ E\ Loss = (x- y)^2$

where x is the image pixels which is fed as input to the Model-1 and y is the pixels predicted by the Model-1.

3.2 Modules and their functionalities

3.2.1 Purpose

The aim of this document is to provide a detailed description of the translator of Sign language to text. It will cover the applications and features of the system, the interfaces of the system, what the system is expected to do, the constraints that the project will work under and how it behaves in response to external stimuli. This is intended for both the developers and the users of this system.

3.2.2 Scope

This system is primarily intended for making an Interpreter. This will have applications in Business who want to employ deaf and mute employees can use it to convey employee messages to the end consumer. It will be used majorly by the deaf and mute to communicate. The applications can further be extended to security purposes, by developing a sign language of your own. And even observing and analysing any suspicious actions.

3.2.3 Glossary

• **Feature:** Features are individual measurable properties or characteristics of a phenomenon being observed. These require classification.

• **Label:** Labels are the final output. We can also consider the output classes to be the labels.

• **Model:** A machine learning model is a mathematical portrayal of a real-life problem. There are various algorithms that perform different tasks with different levels of accuracy.

• **Regression:** Regression is a statistical method that is used to predict real and continuous valued functions.

• **Classification:** In classification, we will need to categorize data into a finite number of predefined classes.

• **CNN :** It is a Machine Learning unit algorithm, for supervised learning, which is used in classificaton of large amount of data

• **Image Processing :** The various modifications done on a raw image to make it suitable for the training model.

• **Training-set :** This is the data set over which CNN model is trained. The predictions are completely dependent on the training-data set.

Overview of Document

The next section, the Overall Description section, of this document gives an overview of the functionality of the project. It describes the informal requirements and is used to establish a context for the technical requirements specification in the next section. The third section, Requirements Specification section, of this document is written primarily for the developers and describes in technical terms the details of the functionality of the product. Both sections of the document describe the same software product in its entirety, but are intended for different audiences and thus use different language.

3.3 Overall Description

3.3.1 SYSTEM ARCHITECTURE

The main objective of the software is to classify the gestures and label them with one of the various categories already defined while training the dataset. We have then tested the same for some data with the help of deep learning. The system has been trained on previously captured gestures which have been labeled with the text associated with them. Multiple copies of each gesture image have been created to extract the features efficiently, then the CNN model is trained. The new gesture which has to be translated, has to be signaled in front of the camera connected to the system, which will be recorded and matched and classified using CNN algorithm.

3.4 Requirement Specifications

3.4.1 FUNCTIONAL REQUIREMENTS

The Prime objective of the software is to translate the sign language into text. Initially, one character was translated at a time and later, the software was trained and developed to translate even words. Words can be formed by concatenating various characters as well and the formed word will be displayed on the output window. The captured images need to be pre-processed. The system modified the images captured and trained the model to classify the signals in one of these defined labels.

3.4.2 NON-FUNCTIONAL REQUIREMENTS

The sole purpose of the software is to facilitate communication for the disabled. The previous available devices were slow and inefficient. Thus, the software is built to translate the signs accurately and at a relatively faster rate. The software is designed efficiently such that it can be modified easily making it easy to maintain.

CHAPTER 4
SOFTWARE DESIGN

4.1 Zero level DFD

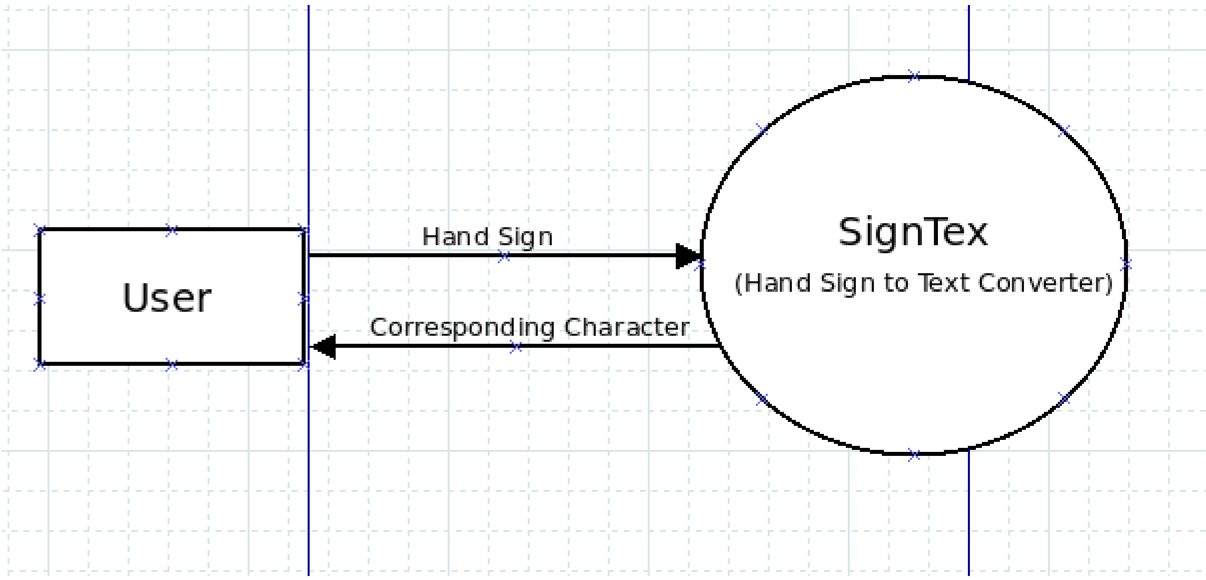


Figure 4.1: Zero Level DFD

4.2 First level DFD

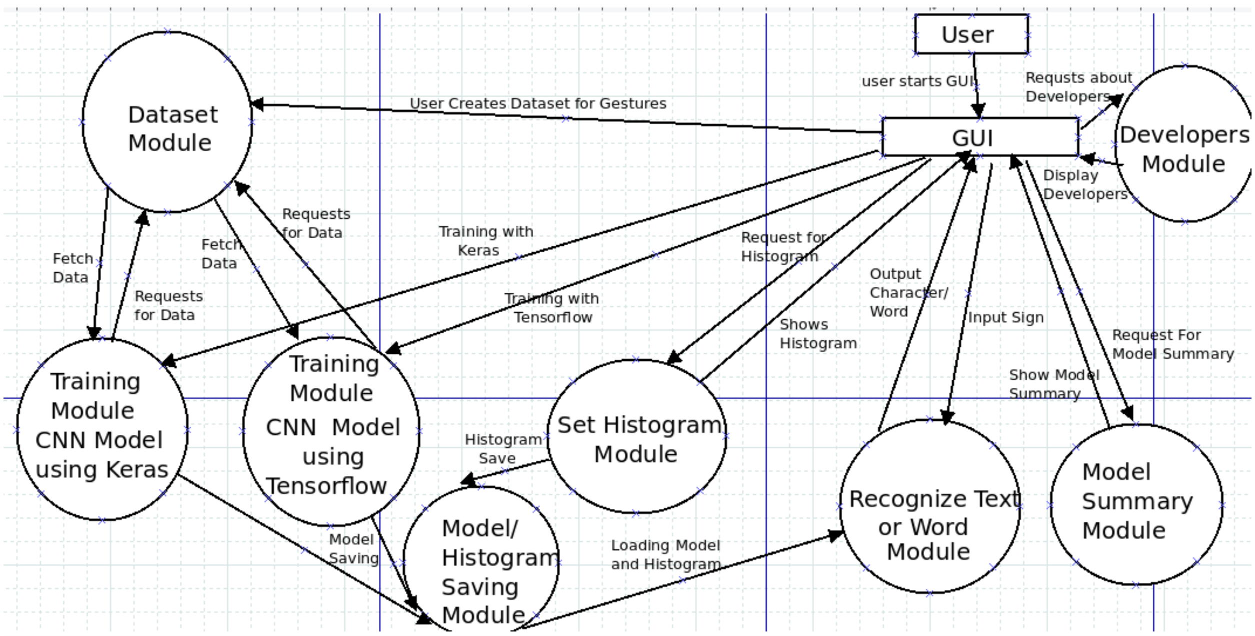


Figure 4.2: First Level DFD

4.3 Overview of Approach

We have created an User Interface which allows the User to add new gestures with some specific meaning, and the option to train the model with the added features. The UI also directs the user to the ”hand recognition” window, which will further be used to do gestures. Then finally two options are provided for translating sign language to Text, one allowing one character at a time, and second to concatenate the characters to form a complete word.

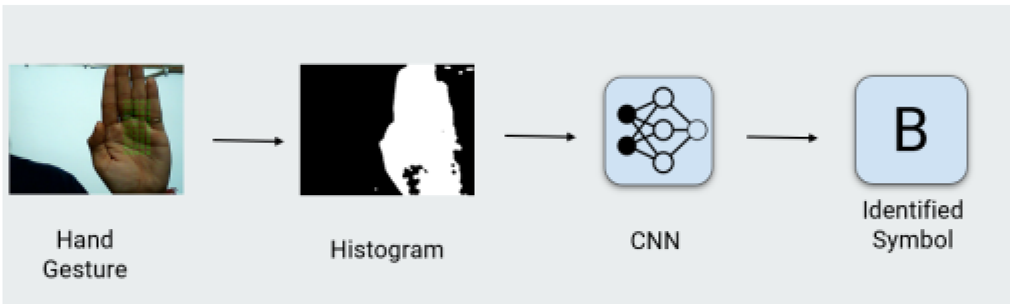


Figure 4.3: Flowchart of the solution

There are a total of 44 gestures for which the model is trained, including 26 English alphabets, 10 numeric digits and 8 other commonly used symbols. 1200 images of each gesture have been captured and then each image is flipped, making a total of 2400 images for each gesture. The images have then been resized to 50*50 pixels and converted to grayscale.

A histogram has been created which identifies the skin of the hand of the speaker and separates it from the background. The Neural Network is trained using Keras on each gesture. Whenever the translator is to be used, the person uses the sign language to be recorded in the camera of the device with the software, which then fetches it into the Neural Network and the sign is classified.

4.4 Pre-processing the Data

4.4.1 HISTOGRAM CREATION

The OpenCV library has been used to generate a histogram that will separate the hand gestures from the background. For this purpose 50 squares in the form of 5*10 are displayed and the hand must cover all the squares. Then the image is captured and a histogram is plotted of the area covering the squares.

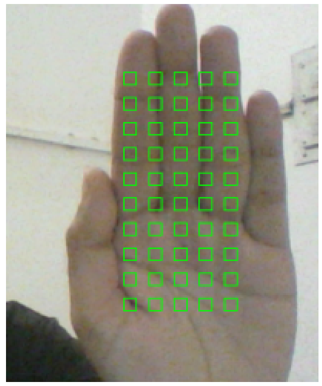


Figure 4.4: Hand detection

4.4.2 IMAGE PROCESSING

Figure 4.5: Histogram of the hand

The images captured of the gestures using the intensities obtained from the histogram are then processed. The images are resized to 50*50 pixels and then converted to gray scale.Each image is then flipped along the vertical axis

4.4.3 CONVOLUTION

The image obtained is then convoluted with the feature detector to form the feature map. This is the most important part in feature detection.The image is convoluted with a number of features and hence a number of feature maps are present.Larger the number of features, better it is to classify the image.

$$(f * g)(t) = \int_{-\infty}^{\infty} f(\tau)g(t-\tau)d\tau$$

The main aim of convolution is to detect features.

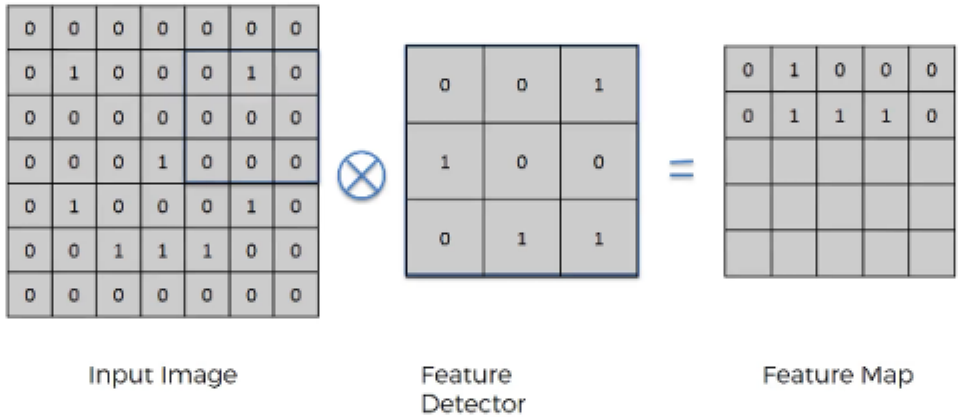


Figure 4.6: Convolution of a matrix

4.4.4 ReLU Layer

The Convoluted image is then passed through the Activation function. The activation function used is the RECTIFIER FUNCTION.This is used to increase the non-linearity in the image.

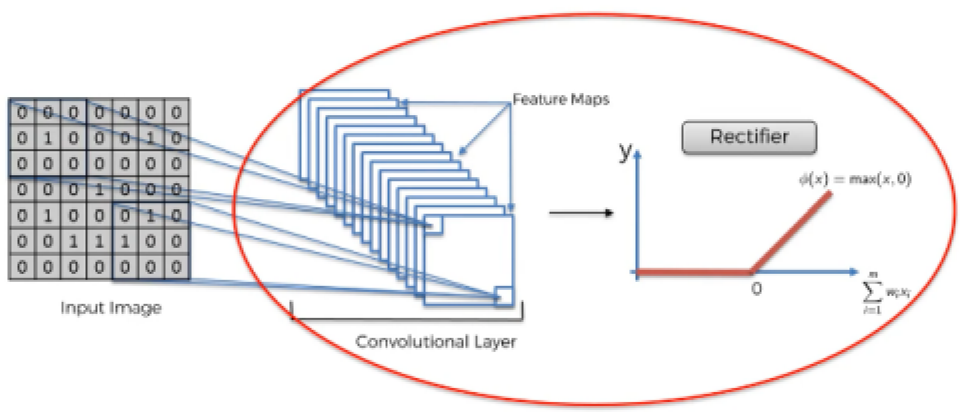


Figure 4.7: Activation function: Rectifier function

4.4.5 MAX POOLING

Now the feature map contains the convoluted result. There are a large number of such maps, hence enormous data. Further, one feature may differ in size, orientation in different images. Both these issues are resolved using Max Pooling. A small grid is selected and then the maximum value is preserved, reducing the size of data as well.

Figure 4.8: Max Pooling

4.4.6 FLATTENING

The pooled image features need to be flattened so that they can be used as input in the next Artificial Neural Network. These form the input layer of the Artificial Neural Network.The max pooling output is transformed into a column.

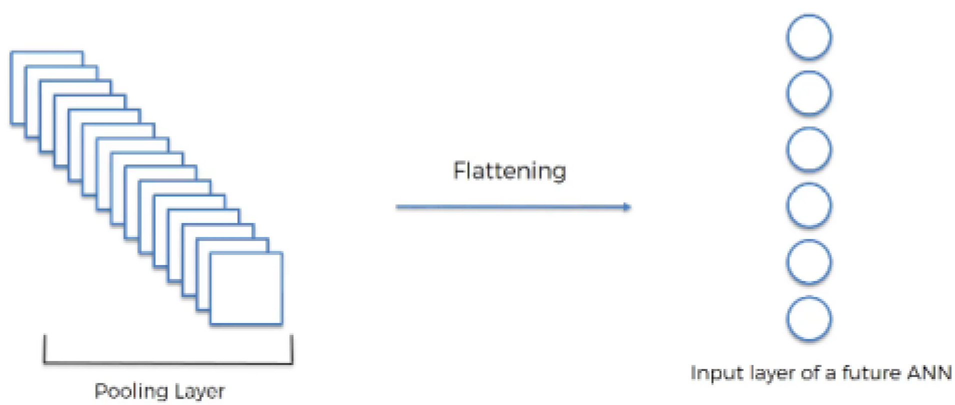


Figure 4.9: Flattening

4.5 Building the Model

The CNN is a type of Artificial Neural Network that is used in Image recognition and processing. It detects specific features in the images and then classifies them accordingly based on the presence of those features.In our model we have 5 layers in total including 3 Convolutional layers and 2 fully connected Dense layers.

The first CNN layer consists of 16 neurons, the second layer consists of 32 neurons and the third layer consists of 64 neurons. The first dense layer consists of 128 neurons and uses the Rectifier function as the Activation function. The second dense layer uses Softmax function as the Activation function.

4.5.1 ACTIVATION FUNCTIONS

Activation function s the function that is applied to the sum of weighted inputs to the neuron. This is where calculations happen. In our neural network, we have used two different activation functions.

- **Relu Activation Function:** ReLU stands for Rectified Linear Unit. It's cheap to compute as there is no complicated math[1]. The model can therefore take less time to train or run. This is used to increase the non-linearity of the images.It is especially useful when dealing with small values as in our case.The formula for ReLu function is given by:

  $$y=max(0,x)$$

- **Softmax Activation function**:Softmax is an activation function. It is frequently used in classifications. Softmax output is large if the score is large. Its output is small if the score is small. The proportion is not uniform. Softmax is exponential and enlarges differences. The formula for Softmax function is given by:

  $$Softmax(x_i) = \frac{e^{x_i}}{\sum_j e^{x_j}}$$

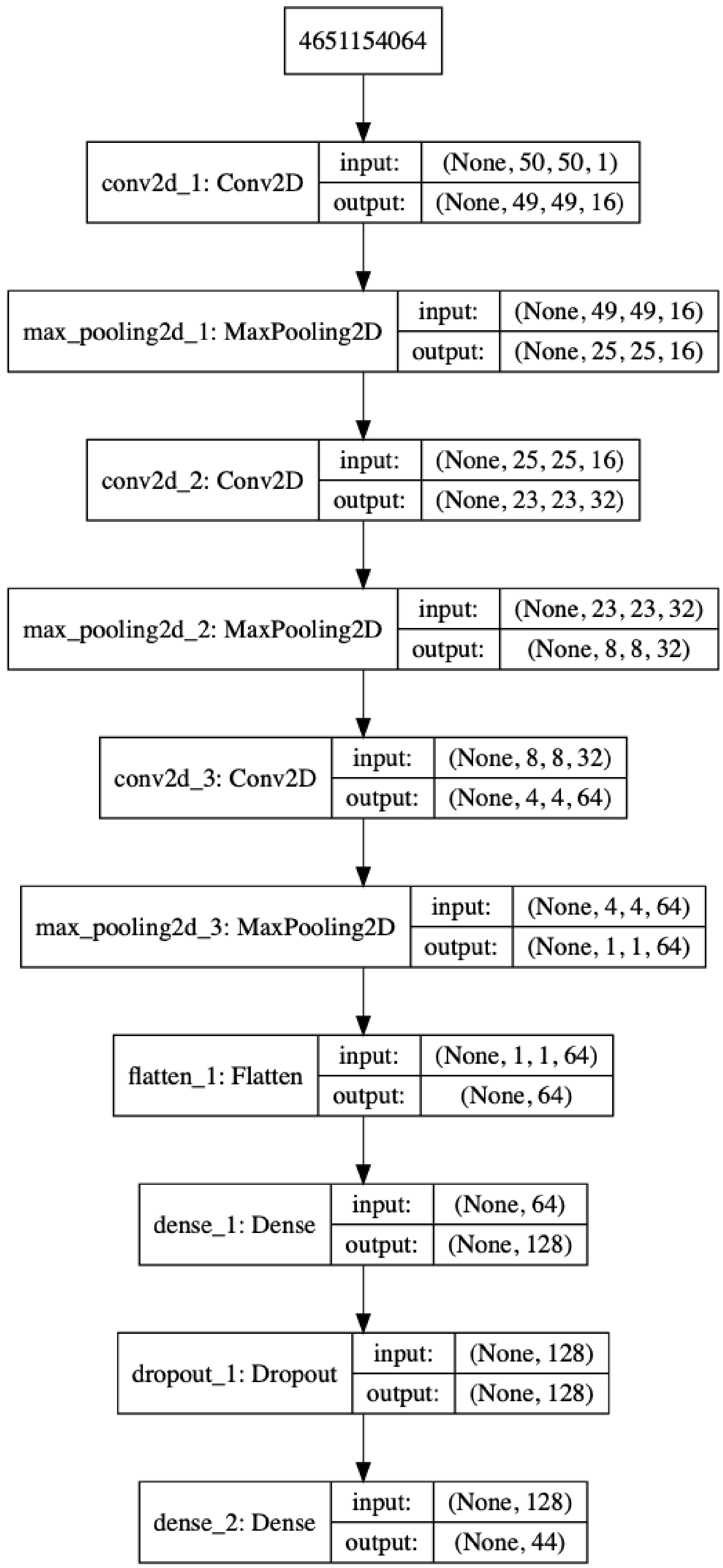


Figure 4.10: Model consisting of 5 layers

CHAPTER 5

Coding /CoreModule

5.1 Training our model

```
import numpy as np
import pickle
import cv2, os
from glob import glob
from keras import optimizers
from keras.models import Sequential
from keras.layers import Dense
from keras.layers import Dropout
from keras.layers import Flatten
from keras.layers.convolutional import Conv2D
from keras.layers.convolutional import MaxPooling2D
from keras.utils import np_utils
from keras.callbacks import ModelCheckpoint
from keras import backend as K
from keras.callbacks import TensorBoard
#K.set_image_dim_ordering('tf')
K.set_image_data_format("channels_last")

os.environ['TF_CPP_MIN_LOG_LEVEL'] = '3'

def get_image_size():
    img = cv2.imread('gestures/1/100.jpg', 0)
    return img.shape

def get_num_of_classes():
    return len(glob('gestures/*'))

image_x, image_y = get_image_size()

def cnn_model():
    num_of_classes = get_num_of_classes()
    model = Sequential()
    model.add(Conv2D(16, (2,2), input_shape=(image_x, image_y, 1),
activation='relu'))
```

```
    model.add(MaxPooling2D(pool_size=(2, 2), strides=(2, 2),
padding='same'))
    model.add(Conv2D(32, (3,3), activation='relu'))
    model.add(MaxPooling2D(pool_size=(3, 3), strides=(3, 3),
padding='same'))
    model.add(Conv2D(64, (5,5), activation='relu'))
    model.add(MaxPooling2D(pool_size=(5, 5), strides=(5, 5),
padding='same'))
    model.add(Flatten())
    model.add(Dense(128, activation='relu'))
    model.add(Dropout(0.2))
    model.add(Dense(num_of_classes, activation='softmax'))
    sgd = optimizers.SGD(lr=1e-2)
    model.compile(loss='categorical_crossentropy', optimizer=sgd,
metrics=['accuracy'])
    filepath="cnn_model_keras2.h5"
    checkpoint1 = ModelCheckpoint(filepath, monitor='val_acc', verbose=1,
save_best_only=True, mode='max')
    callbacks_list = [checkpoint1]
    from keras.utils import plot_model
    plot_model(model, to_file='model.png', show_shapes=True)
    return model, callbacks_list

def train():
    with open("train_images", "rb") as f:
        train_images = np.array(pickle.load(f))
    with open("train_labels", "rb") as f:
        train_labels = np.array(pickle.load(f), dtype=np.int32)

    with open("val_images", "rb") as f:
        _images = np.array(pickle.load(f))
    with open("val_labels", "rb") as f:
        _labels = np.array(pickle.load(f), dtype=np.int32)

    train_images = np.reshape(train_images, (train_images.shape[0],
image_x, image_y, 1))
    val_images = np.reshape(val_images, (val_images.shape[0], image_x,
image_y, 1))
    train_labels = np_utils.to_categorical(train_labels)
    val_labels = np_utils.to_categorical(val_labels)
```

```
    print(val_labels.shape)

    model, callbacks_list = cnn_model()
    model=cnn_model()
    model.summary()
    losses=TensorBoard(log_dir='./logs',batch_size=500)
    model.fit(train_images, train_labels, validation_data=(val_images, 
val_labels), epochs=20, batch_size=500,callbacks=callbacks_list)
    scores = model.evaluate(val_images, val_labels, verbose=0)
    print("CNN Error: %.2f%%" % (100-scores[1]*100))
    #model.save('cnn_model_keras2.h5')

train()
K.clear_session();
```

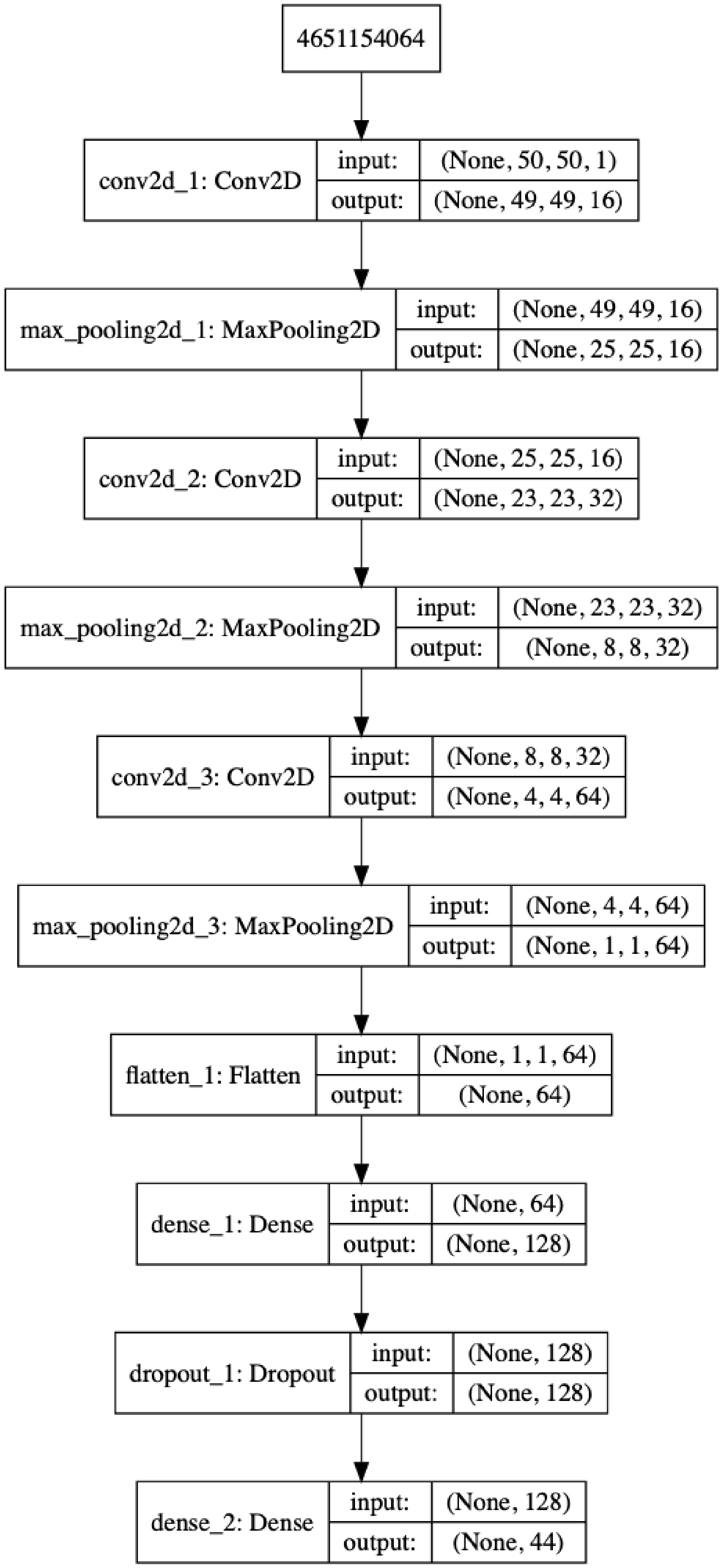


Figure 5.1: Model

5.2 Getting model reports

```
from keras.models import load_model
from sklearn.metrics import classification_report, confusion_matrix
import pickle
import numpy as np
import time
import matplotlib.pyplot as plt

def plot_confusion_matrix(cm,target_names,title='Confusion
matrix',cmap=None,normalize=True):

    import itertools

    accuracy = np.trace(cm) / float(np.sum(cm))
    misclass = 1 - accuracy

    if cmap is None:
        cmap = plt.get_cmap('Blues')

    plt.figure(figsize=(20, 20))
    plt.imshow(cm, interpolation='nearest', cmap=cmap)
    plt.title(title)
    plt.colorbar()

    if target_names is not None:
        tick_marks = np.arange(len(target_names))
        plt.xticks(tick_marks, target_names, rotation=45)
        plt.yticks(tick_marks, target_names)

    if normalize:
        cm = cm.astype('float') / cm.sum(axis=1)[:, np.newaxis]


    thresh = cm.max() / 1.5 if normalize else cm.max() / 2
    for i, j in itertools.product(range(cm.shape[0]), range(cm.shape[1])):
        if normalize:
            plt.text(j, i, "{:0.4f}".format(cm[i, j]),
                     horizontalalignment="center",
                     color="white" if cm[i, j] > thresh else "black")
```

```
        else:
            plt.text(j, i, "{:,}".format(cm[i, j]),
                     horizontalalignment="center",
                     color="white" if cm[i, j] > thresh else "black")


    plt.tight_layout()
    plt.ylabel('True label')
    plt.xlabel('Predicted label\naccuracy={:0.4f};
misclass={:0.4f}'.format(accuracy, misclass))
    plt.savefig('confusion_matrix.png')


image_x, image_y = 50, 50
with open("test_images", "rb") as f:
    test_images = np.array(pickle.load(f))
with open("test_labels", "rb") as f:
    test_labels = np.array(pickle.load(f), dtype=np.int32)
test_images = np.reshape(test_images, (test_images.shape[0], image_x,
image_y, 1))


model = load_model('cnn_model_keras2 (1).h5')
pred_labels = []

start_time = time.time()
pred_probabs = model.predict(test_images)
end_time = time.time()
pred_time = end_time-start_time
avg_pred_time = pred_time/test_images.shape[0]
print("Time taken to predict %d test images is %ds"
%(test_images.shape[0], pred_time))
print('Average prediction time: %fs' % (avg_pred_time))

for pred_probab in pred_probabs:
    pred_labels.append(list(pred_probab).index(max(pred_probab)))

cm = confusion_matrix(test_labels, np.array(pred_labels))
classification_report = classification_report(test_labels,
np.array(pred_labels))
```

```python
print('\n\nClassification Report')
print('---------------------------')
print(classification_report)
plot_confusion_matrix(cm, range(44), normalize=False)
```

```
Classification Report
---------------------------
              precision    recall  f1-score   support

           0       0.99      1.00      1.00       198
           1       1.00      1.00      1.00       190
           2       1.00      1.00      1.00       200
           3       1.00      1.00      1.00       203
           4       0.99      1.00      1.00       177
           5       1.00      1.00      1.00       188
           6       1.00      1.00      1.00       204
           7       1.00      1.00      1.00       205
           8       1.00      1.00      1.00       209
           9       1.00      1.00      1.00       202
          10       1.00      1.00      1.00       212
          11       1.00      1.00      1.00       209
          12       1.00      0.99      0.99       194
          13       1.00      1.00      1.00       228
          14       1.00      1.00      1.00       208
          15       1.00      1.00      1.00       187
          16       1.00      1.00      1.00       184
          17       1.00      1.00      1.00       193
          18       1.00      1.00      1.00       189
          19       0.99      0.99      0.99       199
          20       1.00      1.00      1.00       201
          21       1.00      1.00      1.00       212
          22       1.00      1.00      1.00       199
          23       1.00      1.00      1.00       199
          24       1.00      1.00      1.00       212
          25       1.00      1.00      1.00       200
          26       1.00      1.00      1.00       196
          27       1.00      1.00      1.00       187
          28       1.00      1.00      1.00       205
          29       1.00      1.00      1.00       205
          30       1.00      1.00      1.00       232
          31       1.00      1.00      1.00       216
          32       1.00      1.00      1.00       187
          33       1.00      1.00      1.00       196
          34       1.00      1.00      1.00       199
          35       1.00      1.00      1.00       182
          36       1.00      1.00      1.00       206
          37       1.00      1.00      1.00       195
          38       1.00      1.00      1.00       200
          39       1.00      1.00      1.00       203
          40       1.00      1.00      1.00       209
          41       1.00      1.00      1.00       211
          42       1.00      1.00      1.00       170
          43       1.00      1.00      1.00       199

    accuracy                           1.00      8800
   macro avg       1.00      1.00      1.00      8800
weighted avg       1.00      1.00      1.00      8800
```

Figure 5.2 : Model Reports

5.3 HTML

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">

    <title>Bootstrap 4, from LayoutIt!</title>

    <meta name="description" content="Source code generated using
layoutit.com">
    <meta name="author" content="LayoutIt!">

    <link href="css/bootstrap.min.css" rel="stylesheet">
    <link href="css/style.css" rel="stylesheet">

  </head>
  <body>

    <div class="container-fluid">
    <div class="row">
        <div class="col-md-12">
            <nav class="navbar navbar-expand-lg navbar-light bg-light
static-top">

                <button class="navbar-toggler" type="button"
data-toggle="collapse" data-target="#bs-example-navbar-collapse-1">
                    <span class="navbar-toggler-icon"></span>
                </button> <a class="navbar-brand"
href="#">GNDEC,Ludhiana</a>
                <div class="collapse navbar-collapse"
id="bs-example-navbar-collapse-1">
                    <ul class="navbar-nav">
                        <li class="nav-item active">
                             <a class="nav-link" href="#"> Sign Language
to Text Generation<span class="sr-only">(current)</span></a>
                        </li>
                        <li class="nav-item dropdown">
```

```
                              <a class="nav-link dropdown-toggle"
href="http://example.com" id="navbarDropdownMenuLink"
data-toggle="dropdown">Services</a>
                             <div class="dropdown-menu"
aria-labelledby="navbarDropdownMenuLink">
                                  <a class="dropdown-item" href="#">Create
Gesture</a> <a class="dropdown-item" href="#">Display Gesture</a> <a
class="dropdown-item" href="#">Set Hand Histogram</a> <a
class="dropdown-item" href="#">Recognize Gesture as a character</a> <a
class="dropdown-item" href="#">Recognize Gesture as a word</a> <a
class="dropdown-item" href="#">Model Summary </a> <a class="dropdown-item"
href="#"> Developers</a>
                                 <div class="dropdown-divider">
                                 </div> <a class="dropdown-item"
href="#">Our Code</a>
                             </div>
                         </li>
                     </ul>
                     <form class="form-inline">
                         <input class="form-control mr-sm-2" type="text">
                         <button class="btn btn-primary my-2 my-sm-0"
type="submit">
                             Search
                         </button>
                     </form>
                     <ul class="navbar-nav ml-md-auto">
                         <li class="nav-item active">
                              <a class="nav-link" href="#">Home <span
class="sr-only">(current)</span></a>
                         </li>
                         <li class="nav-item dropdown">
                              <a class="nav-link dropdown-toggle"
href="http://example.com" id="navbarDropdownMenuLink"
data-toggle="dropdown">Our Services</a>
                             <div class="dropdown-menu dropdown-menu-right"
aria-labelledby="navbarDropdownMenuLink">
                                  <a class="dropdown-item" href="#">Display
Message</a> <a class="dropdown-item" href="#">Play Voice Message</a> <a
class="dropdown-item" href="#">Text to Sign Language</a>
                                 <div class="dropdown-divider">
```

```
                                </div> <a class="dropdown-item"
href="#">Link to Our Repository</a>
                            </div>
                        </li>
                    </ul>
                </div>
            </nav>
        </div>
    </div>
    <div class="row">
        <div class="col-md-12">
            <div class="carousel slide" id="carousel-342093">
                <ol class="carousel-indicators">
                    <li data-slide-to="0" data-target="#carousel-342093"
class="active">
                    </li>
                    <li data-slide-to="1" data-target="#carousel-342093">
                    </li>
                    <li data-slide-to="2" data-target="#carousel-342093">
                    </li>
                </ol>
                <div class="carousel-inner">
                    <div class="carousel-item active">
                        <img class="d-block w-100" alt="Carousel Bootstrap
First"
src="https://uploads-ssl.webflow.com/5ffd73946b77c6ca0c73e187/60118a3fa324
346611d01fe6_hero.jpg">
                        <div class="carousel-caption">
                            <h4 style="color:black;">
                                Real Life Use Case
                            </h4>
                            <p style="color:black;">
                                UI asking for sign language image input
from user and translating it to text
                            </p>
                        </div>
                    </div>
                    <div class="carousel-item">
                        <img class="d-block w-100" alt="Carousel Bootstrap
Second"
```

```
src="https://www.researchgate.net/publication/328396430/figure/fig1/AS:683619848830976@1539999081795/The-26-letters-and-10-digits-of-American-Sign-Language-ASL.jpg">
                        <div class="carousel-caption">
                            <p
style="background-color:white;color:black;">
                                AMERICAN SIGN LANGUAGE
                            </p>
                        </div>
                    </div>
                    <div class="carousel-item">
                        <img class="d-block w-100" alt="Carousel Bootstrap
Third"
src="https://assets.skyfilabs.com/images/blog/sign-language-translator.webp">
                        <div class="carousel-caption">
                            <h4
style="background-color:white;color:black;">
                                Some samples of sign language
                            </h4>

                        </div>
                    </div>
                </div> <a class="carousel-control-prev"
href="#carousel-342093" data-slide="prev"><span
class="carousel-control-prev-icon"></span> <span
class="sr-only">Previous</span></a> <a class="carousel-control-next"
href="#carousel-342093" data-slide="next"><span
class="carousel-control-next-icon"></span> <span
class="sr-only">Next</span></a>
            </div>
        </div>
    </div>
    <div class="row">
        <div class="col-md-8">
            <h3>
                Introduction
            </h3>
            <p>
```

```
                This is a simple CNN project which helps its users to
convert the sign language directly into text.
                We have used different types of gestures which can be
converted into text on the basis of the standard American Sign Language.
                The project is trained on Keras. It has word
identification and working mathematical operations.
                The major technologies used are <strong> IMAGE PROCESSING
and CNN (Convolutional Neural Networks) </strong>.
            </p>
             <a id="modal-417538" href="#modal-container-417538"
role="button" class="btn" data-toggle="modal">Accuracy of the gestures
used by the model.</a>

            <div class="modal fade" id="modal-container-417538"
role="dialog" aria-labelledby="myModalLabel" aria-hidden="true">
                <div class="modal-dialog" role="document">
                    <div class="modal-content">
                        <div class="modal-header">
                            <h5 class="modal-title" id="myModalLabel">
                                Modal title
                            </h5>
                            <button type="button" class="close"
data-dismiss="modal">
                                <span aria-hidden="true">×</span>
                            </button>
                        </div>
                        <div class="modal-body">
                            ...
                        </div>
                        <div class="modal-footer">

                            <button type="button" class="btn btn-primary">
                                Save changes
                            </button>
                            <button type="button" class="btn
btn-secondary" data-dismiss="modal">
                                Close
                            </button>
                        </div>
                    </div>
```

```html
                </div>

            </div>

            <blockquote class="blockquote text-right">
                <p class="mb-0">
                     "A blind man knows he cannot see and is glad to be
led, though it be by a dog; but he that is blind in his understanding,
which is the worst blindness of all, believes he sees as the best and
scorns a guide."
                </p>
                <footer class="blockquote-footer">
                    Samuel Butler
                </footer>
            </blockquote>
        </div>
        <div class="col-md-4">
            <div class="tabbable" id="tabs-420433">
                <ul class="nav nav-tabs">
                    <li class="nav-item">
                        <a class="nav-link active" href="#tab1"
data-toggle="tab">Developer Details</a>
                    </li>
                    <li class="nav-item">
                        <a class="nav-link" href="#tab2"
data-toggle="tab">Contact Us</a>
                    </li>
                </ul>
                <div class="tab-content">
                    <div class="tab-pane active" id="panel-185654">
                        <p>
                            <br> Developers: </br>
                            <br> <strong> Guneet Kohli </strong>
guneet1815017@gndec.ac.in </br>
                            <br> <strong> Harneet Kaur </strong>
harneet1815026@gndec.ac.in </br>
                            <br> <strong> Jashanpreet Kaur </strong>
jashanpreet1815034@gndec.ac.in </br>
```

```
                        </p>
                    </div>
                    <div class="tab-pane" id="tab2">
                        <p>
                        <a href="url"> https://github.com/guneet-coder</a>
                        </p>
                    </div>
                </div>
            </div>
        </div>
    </div>
    <div class="row">
        <div class="col-md-12">
            <div class="row">
                <div class="col-md-4">
                    <div class="card">
                        <img class="card-img-top" alt="Bootstrap Thumbnail
First"
src="https://miro.medium.com/max/2000/1*ZSJ_vYSZyeQTau5G68tN6g.png">
                        <div class="card-block">
                            <h5 class="card-title">
                                Confusion Matrix
                            </h5>
                            <p class="card-text">
                                It is used to describe the performance of
the classifying model. The higher th diagonal value, the better is the
model.
                            </p>
                            <p>
                                <a class="btn btn-primary" href="#">Read
more</a>
                            </p>
                        </div>
                    </div>
                </div>
                <div class="col-md-4">
                    <div class="card">
                        <img class="card-img-top" alt="Bootstrap Thumbnail
Second"
src="https://www.researchgate.net/publication/346129022/figure/fig3/AS:961
```

```
239823970323@1606188841513/Visualizing-accuracy-recall-aka-sensitivity-and
-precision-which-are-the-common.ppm">
                        <div class="card-block">
                            <h5 class="card-title">
                                Recall
                            </h5>
                            <p class="card-text">
                                Recall is the fraction of the total number
of relevant instances that were retrieved.
                                Recall=TP/(TP+FN)
                                Precision is defined as the fraction of the
relevant instances among the instances retrieved.
                                Precision=TP/(TP+FP)
                            </p>
                            <p>
                                <a class="btn btn-primary" href="#">Read
more</a>
                            </p>
                        </div>
                    </div>
                </div>
                <div class="col-md-4">
                    <div class="card">
                        <img class="card-img-top" alt="Bootstrap Thumbnail
Third" src="https://www.monperrus.net/martin/f1-measure.gif">
                        <div class="card-block">
                            <h5 class="card-title">
                                F1 Score
                            </h5>
                            <p class="card-text">
                            F1 is a standard evaluation metric from
information retrieval research.
                            It combines the precision and the recall. In
order to understand this combination, here is a visualization of the
landscape of the F1-score.
                            A perfect system (with precision and recall ~
100%) would be on the top right hand side of the figure.
                            </p>
                            <p>
```

```
                                <a class="btn btn-primary" href="#">Read
more</a>
                            </p>
                        </div>
                    </div>
                </div>
            </div>
            <div class="row">
                <div class="col-md-4">

                    <address>
                         <strong> Guru Nanak Dev Engineering
College</strong><br> Gill Park <br> Ludhiana, Punjab 141006 <br> <abbr
title="Phone">P:</abbr> (0160) 250-2700
                    </address>
                </div>
                <div class="col-md-4">
                    <form role="form">
                        <div class="form-group">

                            <label for="exampleInputEmail1">
                                Email address
                            </label>
                            <input type="email" class="form-control"
id="exampleInputEmail1">
                        </div>
                        <div class="form-group">

                            <label for="exampleInputPassword1">
                                Password
                            </label>
                            <input type="password" class="form-control"
id="exampleInputPassword1">
                        </div>
                        <div class="form-group">

                            <label for="exampleInputFile">
                                Input American Sign Language image here
                            </label>
```

```
                              <input type="file" class="form-control-file"
id="exampleInputFile">

                          </div>
                          <div class="checkbox">

                              <label>
                                  <input type="checkbox"> I agree to the
terms and conditions
                              </label>
                          </div>
                          <button type="submit" class="btn btn-primary">
                              Submit
                          </button>
                      </form>
                  </div>
                  <div class="col-md-4">
                      <div id="card-766728">
                          <div class="card">
                              <div class="card-header">
                                   <a class="card-link"
data-toggle="collapse" data-parent="#card-766728"
href="#card-element-843366">Organization Details</a>
                              </div>
                              <div id="card-element-843366" class="collapse
show">
                                  <div class="card-body">
                                      This a product of GNDEC, Ludhiana
                                  </div>
                              </div>
                          </div>
                          <div class="card">
                              <div class="card-header">
                                   <a class="collapsed card-link"
data-toggle="collapse" data-parent="#card-766728"
href="#card-element-511313">Forgot Password</a>
                              </div>
                              <div id="card-element-511313"
class="collapse">
                                  <div class="card-body">
```

```html
                                    <div class="form-group">
                                <label for="exampleInputEmail1">
                                Please enter your email address here
                                </label>
                            <input type="email" class="form-control"
id="exampleInputEmail1">
                            </div>

                                </div>
                            </div>
                        </div>
                    </div>
                </div>
            </div>
            <nav>
                <ol class="breadcrumb">
                    <li class="breadcrumb-item">
                        <a href="#">Home</a>
                    </li>
                    <li class="breadcrumb-item">
                        <a href="#">Library</a>
                    </li>
                    <li class="breadcrumb-item active">
                        Data
                    </li>
                </ol>
            </nav>
        </div>
    </div>
</div>

    <script src="js/jquery.min.js"></script>
    <script src="js/bootstrap.min.js"></script>
    <script src="js/scripts.js"></script>
  </body>
</html>
```

CHAPTER 6
Performance of the project Developed

6.1 ANALYSIS

The accuracy of the translator is increasing as the number of epochs are increased while training the model and when the number of images for each gesture are increased.

Epochs : It is the number of times the model is trained over the same data-set.

6.1.1 RECALL

It is the fraction of the total number of relevant instances that were retrieved.

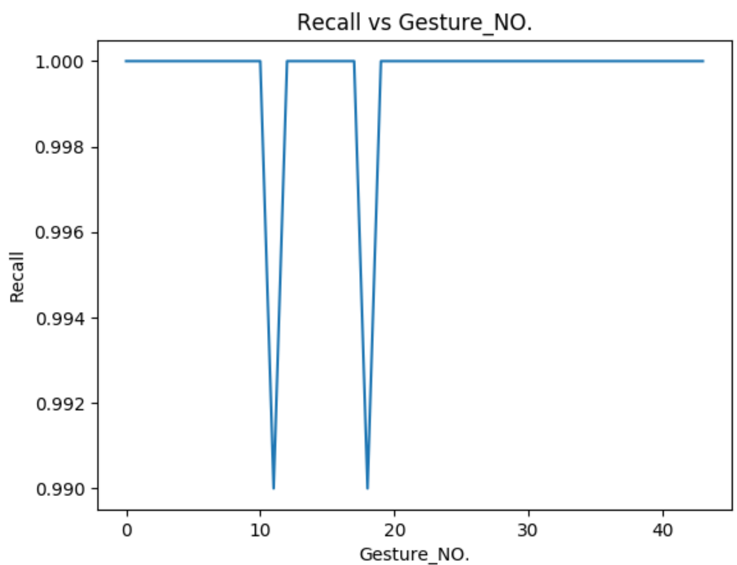


Figure 6.1:Recall vs Gesture Number

6.1.2 PRECISION

It is defined as the fraction of the relevant instances among the instances retrieved.

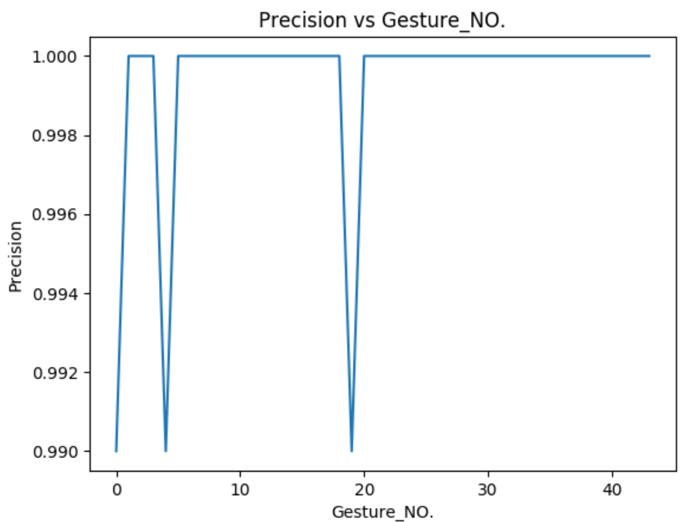


Figure 6.2 : Precision vs Gesture Number

6.1.3 F-1 SCORE

It is defined as the harmonic mean of the Recall and the Precision.In this, even the false negative and false positive are crucial. It tells the accuracy of the classifier in classifying the data points in that particular class compared to all other classes.

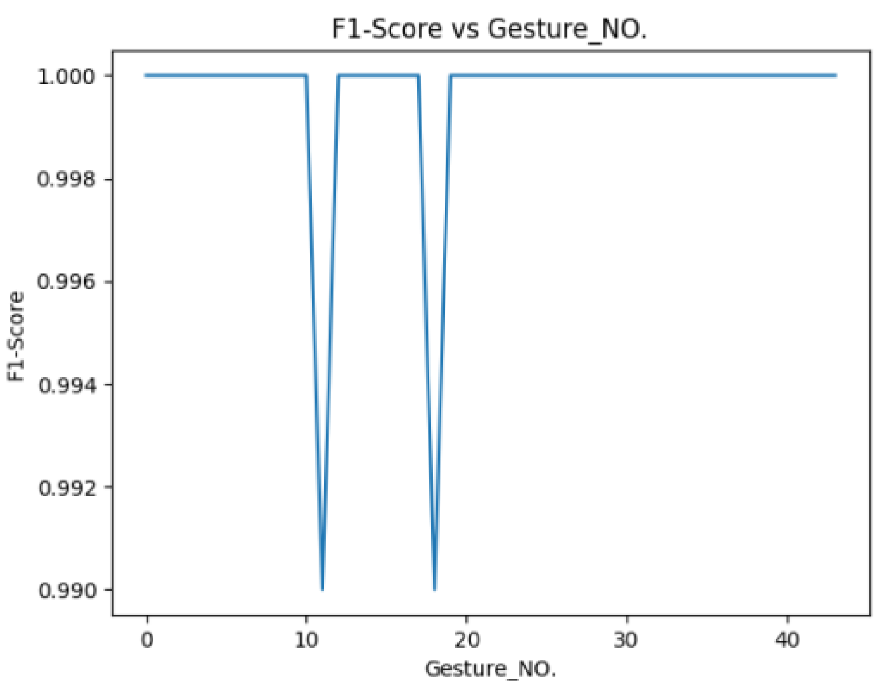


Figure 6.3 : F1-Score vs Gesture Number

6.2 RESULTS

6.2.1 CONFUSION MATRIX

Classification was using Convolutional Neural Network. The accuracy obtained on the test data was 0.9997 and misclass was 0.0003.

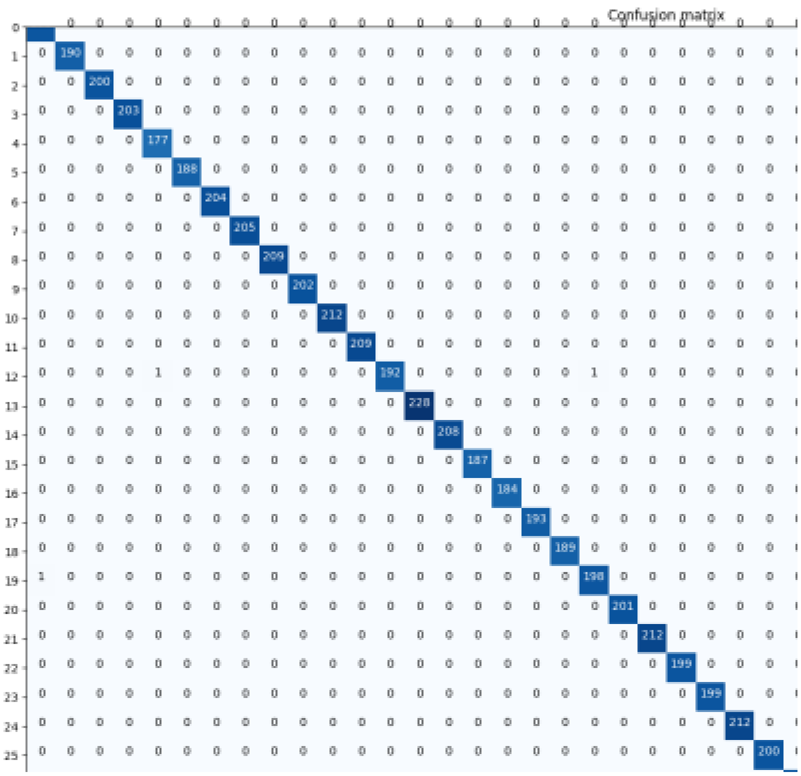
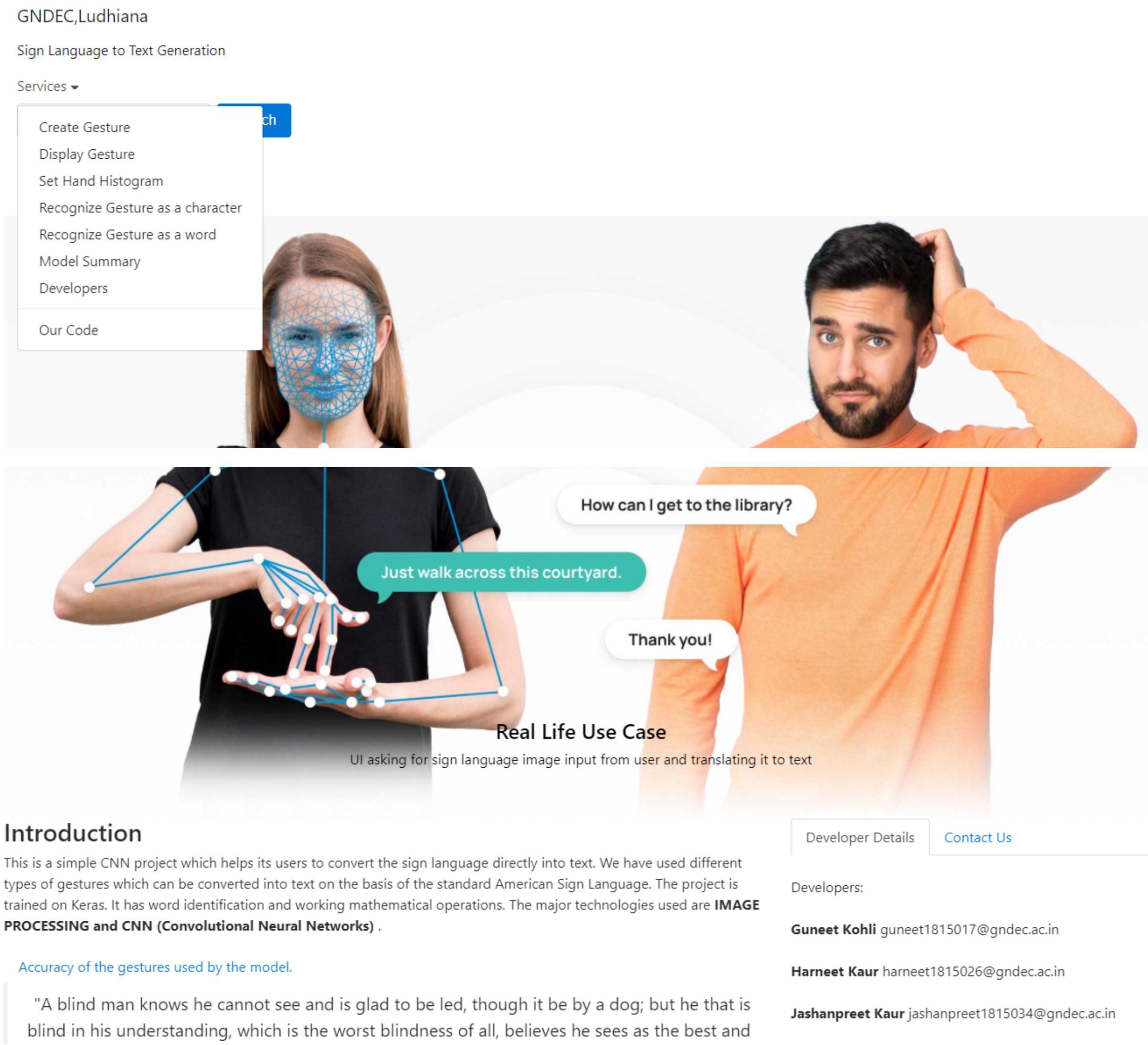
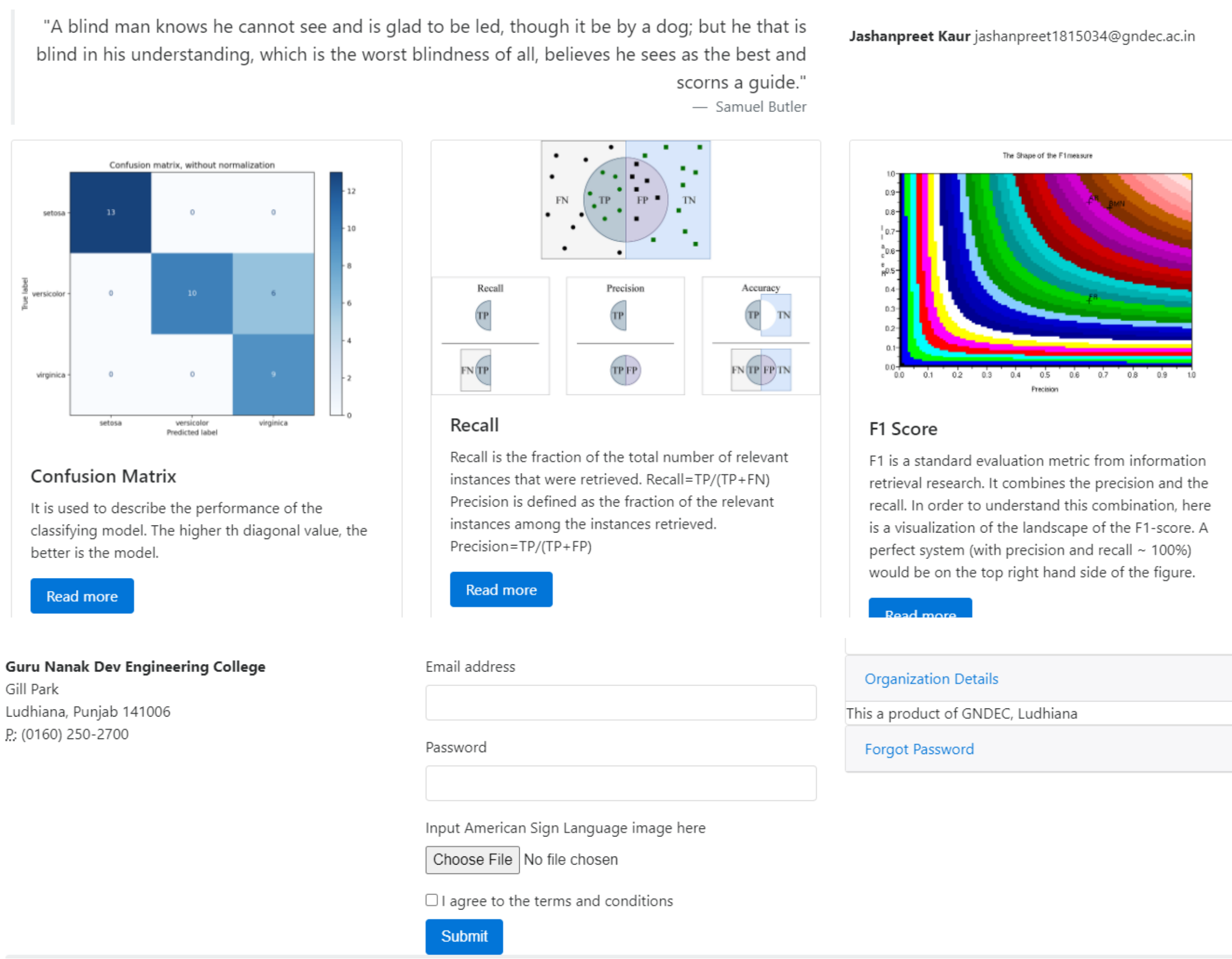


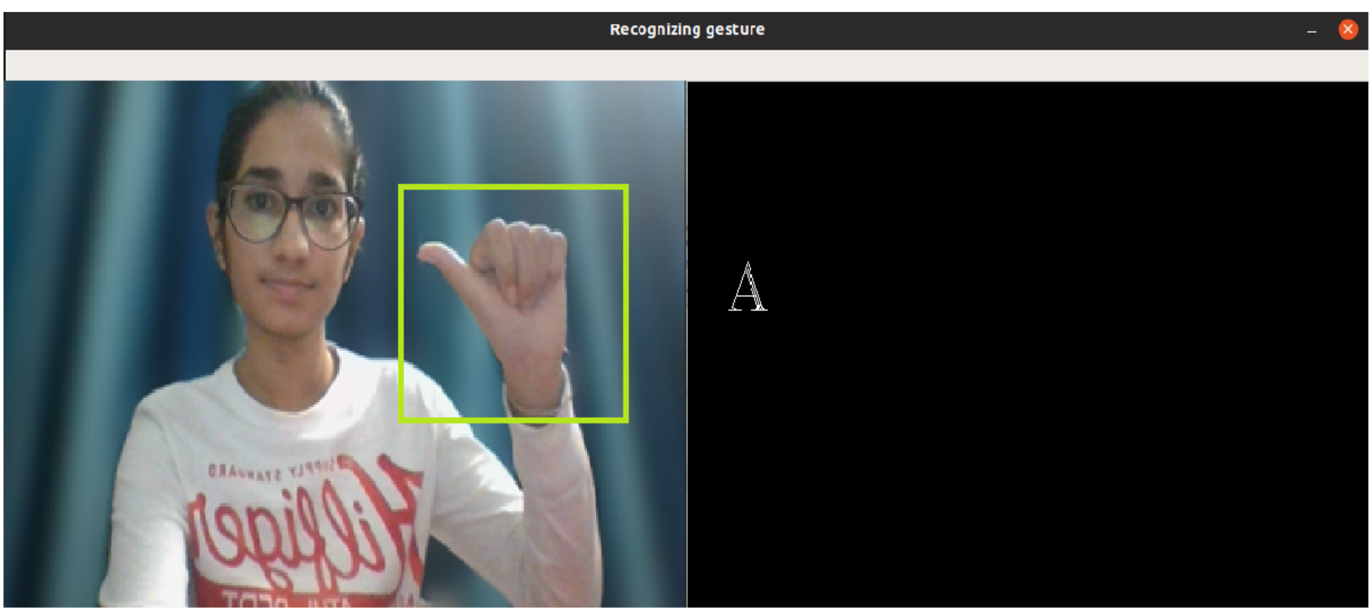
Figure 6.4 : Confusion Matrix

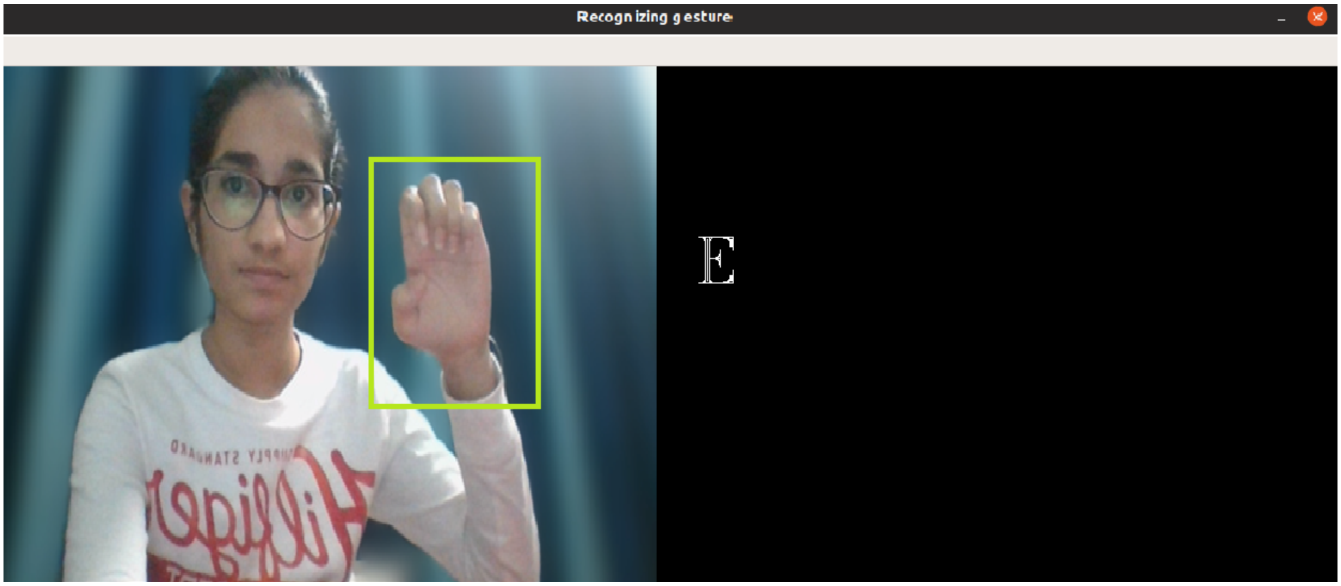
CHAPTER 7
OUTPUT SCREENS
7.1 User Interface

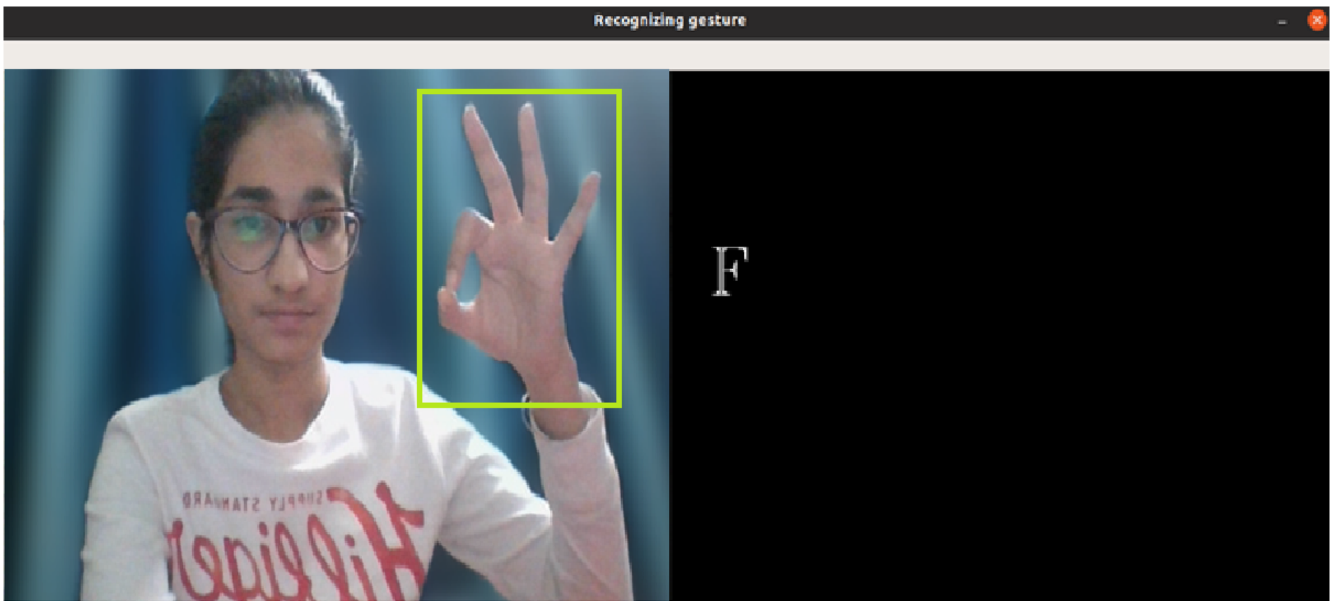



"A blind man knows he cannot see and is glad to be led, though it be by a dog; but he that is blind in his understanding, which is the worst blindness of all, believes he sees as the best and scorns a guide."
— Samuel Butler
Jashanpreet Kaur jashanpreet1815034@gndec.ac.in
Confusion matrix, without normalization
Confusion Matrix
It is used to describe the performance of the classifying model. The higher th diagonal value, the better is the model.
Read more
Recall
Recall is the fraction of the total number of relevant instances that were retrieved. Recall=TP/(TP+FN) Precision is defined as the fraction of the relevant instances among the instances retrieved. Precision=TP/(TP+FP)
Read more
The Shape of the F1measure
F1 Score
F1 is a standard evaluation metric from information retrieval research. It combines the precision and the recall. In order to understand this combination, here is a visualization of the landscape of the F1-score. A perfect system (with precision and recall ~ 100%) would be on the top right hand side of the figure.
Guru Nanak Dev Engineering College
Gill Park
Ludhiana, Punjab 141006
P: (0160) 250-2700
Email address
Password
Input American Sign Language image here
Choose File No file chosen
I agree to the terms and conditions
Submit
Organization Details
This a product of GNDEC, Ludhiana
Forgot Password

7.2 MODEL


Recognizing gesture
A


Recognizing gesture
E


Recognizing gesture
F

CHAPTER 8
REFERENCES

[1] N B, Mahesh K. 2018. “Conversion of Sign Language into Text.” *International Journal of Applied Engineering Research ISSN 0973-4562 Volume 13, Number 9*.
https://www.ripublication.com/ijaer18/ijaerv13n9_90.pdf

[2]Bantupalli, Kshitij, and Ying Xie. 2019. “American Sign Language Recognition using Deep Learning and Computer Vision.” *IEEE*, (January).
https://ieeexplore.ieee.org/abstract/document/8622141.

[3] Arsan, Taner, and Oguz Ulgen. 2015. “SIGN LANGUAGE CONVERTER.” *International Journal of Computer Science & Engineering Survey (IJCSES) Volume* 6, *Number* 4 (August).
https://www.researchgate.net/publication/282839736_Sign_Language_Converter.

[4] Grif, Mikhail G., and Olga O. Korolkova. 2011. “Development of computer sign language translation technology for deaf people.” *IEEE*, (September).
https://ieeexplore.ieee.org/abstract/document/6021115.

[5] Baranwal, Neha, and G. C. Nandi. 2015, "Continuous dynamic Indian Sign Language gesture recognition with invariant backgrounds by Kumud Tripathi", *Conference on Advances in Computing Communications and Informatics*, (September).
https://ieeexplore.ieee.org/abstract/document/7275945

[6] Hu Peng, “Application Research on Face Detection Technology based on Open CV in Mobile Augmented Reality”, International Journal of Signal Processing, Image Processing and Pattern Recognition, *Volume* 8, *Number* 2 (2015)
http://article.nadiapub.com/IJSIP/vol8_no4/22.pdf