MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR

(A Govt. Aided UGC Autonomous & NAAC Accredited Institute Affiliated to RGPV, Bhopal)

Project Report

on

Emotion Detection using facial recognition

A project report submitted in partial fulfilment of the requirement for the degree of

BACHELOR OF TECHNOLOGY

in

INFORMATION TECHNOLOGY

Submitted by:

Prathmesh Soni

(0901IO201048)

Shubham Sahu

(0901IO201060)

Faculty Mentor:

Prof. Abhishek Dixit

Assistant Professor , Information Technology

Submitted to:

DEPARTMENT OF INFORMATION TECHNOLOGY

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE

GWALIOR – 474005

July-December 2022

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR

(A Govt. Aided UGC Autonomous & NAAC Accredited Institute Affiliated to RGPV, Bhopal)

CERTIFICATE

This is certified that **Prathmesh Soni** (0901IO201048) has submitted the project report titled Emotion Detection using facial recognition under the mentorship of Prof. Abhishek Dixit in partial fulfilment of the requirement for the award of degree of Bachelor of Technology in **Information Technology** from Madhav Institute of Technology and Science, Gwalior.

Prof. Abhishek Dixit
Assistant Professor
Information Technology

Dr. Akhilesh Tiwari
Professor and Head,
Department of IT

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR

(A Govt. Aided UGC Autonomous & NAAC Accredited Institute Affiliated to RGPV, Bhopal)

CERTIFICATE

This is certified that **Shubham Sahu** (0901IO201060) has submitted the project report titled Emotion Detection using facial recognition under the mentorship of Prof. Abhishek Dixit in partial fulfilment of the requirement for the award of degree of Bachelor of Technology in **Information Technology** from Madhav Institute of Technology and Science, Gwalior.

Prof. Abhishek Dixit
Assistant Professor
Information Technology

Dr. Akhilesh Tiwari
Professor and Head,
Department of IT

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR

(A Govt. Aided UGC Autonomous & NAAC Accredited Institute Affiliated to RGPV, Bhopal)

DECLARATION

I hereby declare that the work being presented in this project report, for the partial fulfilment of requirement for the award of the degree of Bachelor of Technology in Information Technology at Madhav Institute of Technology & Science, Gwalior is an authenticated and original record of my work under the mentorship of **Prof. Abhishek Dixit, Assistant Professor,** Information Technology.

I declare that I have not submitted the matter embodied in this report for the award of any degree or diploma anywhere else.

Date:
Place: Gwalior

Prathmesh Soni
0901IO201048
III Year
Information Technology,

Shubham Sahu
0901IO201060
III Year
Information Technology

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR

(A Govt. Aided UGC Autonomous & NAAC Accredited Institute Affiliated to RGPV, Bhopal)

ACKNOWLEDGEMENT

The full semester project has proved to be pivotal to my career. I am thankful to my institute, **Madhav Institute of Technology and Science** to allow me to continue my disciplinary/interdisciplinary project as a curriculum requirement, under the provisions of the Flexible Curriculum Scheme (based on the AICTE Model Curriculum 2018), approved by the Academic Council of the institute. I extend my gratitude to the Director of the institute, **Dr. R. K. Pandit** and Dean Academics, **Dr. Manjaree Pandit** for this.

I would sincerely like to thank my department, **Department of Information Technology,** for allowing me to explore this project. I humbly thank **Dr. Akhilesh Tiwari**, Professor and Head, Department of Information Technology, for his continued support during the course of this engagement, which eased the process and formalities involved.

I am sincerely thankful to my faculty mentors. I am grateful to the guidance of **Prof. Abhishek Dixit** , Assistant Professor , Information Technology, for his continued support and guidance throughout the project. I am also very thankful to the faculty and staff of the department.

Prathmesh Soni
0901IO201048
III Year
Information Technology,

Shubham Sahu
0901IO201060
III Year
Information Technology

TABLE OF CONTENTS

PAGE NO.

Abstract

सार

List of Figures

Chapter 1: Project Overview

1.1 Introduction

1.2 Objectives and Scope

1.3 Project Features

1.4 System Requirement

1.4.1 Hardware Specifications

1.4.2 Software Specifications

Chapter 2: Literature Review

Chapter 3: Preliminary Design

3.1 Tools And Technologies Used

3.1.1 Technologies Used

3.1.1.1 Artificial Neural Networks

3.1.1.2 Data Science

3.1.1.3 Transfer Learning

3.1.2 Tools

3.1.2.1 Python

3.1.2.2 Jupyter Notebook

3.1.2.3 Anaconda Prompt

3.1.2.4 Firefox

3.1.3 Modules

3.1.3.1 Numpy

3.1.3.2 Pathlib

3.1.3.3 Matplotlib

3.1.3.4 Opencv

3.1.3.5 Tensorflow

Chapter 4: Final Analysis and Design

4.1 Model Implementation

4.1.1 Data Gathering

4.1.2 Data Preprocessing

4.1.2.1 Resizing

4.1.2.2 Scaling

4.1.3 Building ANN model

4.1.4 Training

4.1.5 Interactive Interface

4.1.6 Testing

4.2 Flowchart

4.3 Code Snippets

4.4 Result

4.5 Result Analysis

4.6 Application

CONCLUSION

REFERENCES

LIST OF FIGURES

Figure Number	Figure caption	Page No.
1	Face Emotion Classification Stages	
2	Artificial Neural Network	
3	Working Flowchart	
4	Data collection code snippet	
5	Data preprocessing code snippet	
6	Building ANN code snippet	
7	Training code snippet	
8	Interactive code snippet	
9	Emotion detection Software	

ABSTRACT

Emotion detection has been an area of interest for researchers for several decades. Since emotions can influence our decisions in life, they have been defined as an essential part of every human behaviour predictor. In this report, we have developed an emotion detection system using facial recognition. For this project , we have used CNN through transfer learning for Image Classification. Unlike other deep learning algorithms, our algorithm is designed in such a way that it can be implemented in any device with low software and hardware specifications.

Keyword: CNN, Deep Learning, Transfer Learning.

सार:

भावना का पता लगाना कई दशकों से शोधकर्ताओं के लिए रुचि का क्षेत्र रहा है। चूंकि भावनाएं जीवन में हमारे निर्णयों को प्रभावित कर सकती हैं, इसलिए उन्हें हर मानव व्यवहार भविष्यवक्ता के एक अनिवार्य हिस्से के रूप में परिभाषित किया गया है। इस रिपोर्ट में, हमने चेहरे की पहचान का उपयोग करके एक भावना पहचान प्रणाली विकसित की है। इस परियोजना के लिए, हमने छवि वर्गीकरण के लिए स्थानांतरण सीखने के माध्यम से सीएनएन का उपयोग किया है। अन्य डीप लर्निंग एल्गोरिदम के विपरीत, हमारे एल्गोरिदम को इस तरह से डिज़ाइन किया गया है कि इसे कम सॉफ्टवेयर और हार्डवेयर विनिर्देशों वाले किसी भी डिवाइस में लागू किया जा सकता है।

कीवर्ड: सीएनएन, डीप लर्निंग, ट्रांसफर लर्निंग।

CHAPTER-1: PROJECT OVERVIEW

1.1 Introduction :-

Facial emotion recognition has become an important issue in many application nowadays. In recent years, the research on facial emotion recognition has become extensive. The aim of facial emotion recognition is to help identify the state of human emotion (eg; neutral, happy, sad, surprise, fear, anger, disgust, contempt) based on particular facial images. The challenge on facial emotion recognition is to automatically recognize facial emotion state with high accuracy.

1.2 Objective & Scope of Project :-

Emotion recognition based on facial expression is an intriguing research field, which has been presented and applied in various spheres such as safety, health and in human machine interfaces. So through this project, we classify different human emotions with the help of some Deep Learning techniques.

1.3 Project Features :-

- Sharp Perception
- Achievable
- Unique
- Easy to understand

1.4 System Requirements:-

1.4.1 Hardware Specification :

Finding the hardware requirements is important because the hardware is a key component in the development of a web project.

1.4.1.1 Processor:

For this platform, a Pentium IV processor with an 800 MHz processing speed is the bare minimum. It is crucial to select the appropriate processor because the amount of time needed to perform an instruction depends on the processor's power.

1.4.1.2 RAM:

The memory plays a role in processing speeds that are faster. Therefore, 1 GB of RAM should be the minimum for better performance.

1.4.1.3 Hard disk:

Today's internet platforms create enormous amounts of data every day. As a result, a large hard disc is needed to store the processed data.

HARDWARE TOOLS	MINIMUM REQUIREMENTS
Processor	I5 (10^{th} gen) or above
RAM	8 GB or above
Monitor	15.6” coloured
GPU	4 GB or above
Keyboard	122 keys
Hard Disk	128 GB

1.4.2 Software Specification :

SOFTWARE TOOLS	MINIMUM REQUIREMENTS
Platform	Windows
Operating System	Windows 10 or above
Technology	Deep Learning through Tensorflow
Scripting Language	Python (version 3.10)
IDE	Jupyter notebook

CHAPTER 2: LITERATURE REVIEW

The challenge on facial emotion recognition is to automatically recognize facial emotion state with high accuracy. It is challenging to find the similarity of the same emotion state between different person since they may express the same emotion state in various ways.

Generally FER is divided into three major stages as shown in Figure : (i) Face Detection, (ii) Feature Extraction, and (iii) Emotion Classification. At first stage, which is a pre-processing stage, an image of a face is detected and facial components of the face will be detected from the region. The facial components can be an eyes, brows, nose, and mouth. In the second stage, an informative features will be extracted from different parts of the face. In the last stage, a classifier need to be trained before been used to generate labels for the Emotions using the training data.

Facial Emotion Classification Stages.

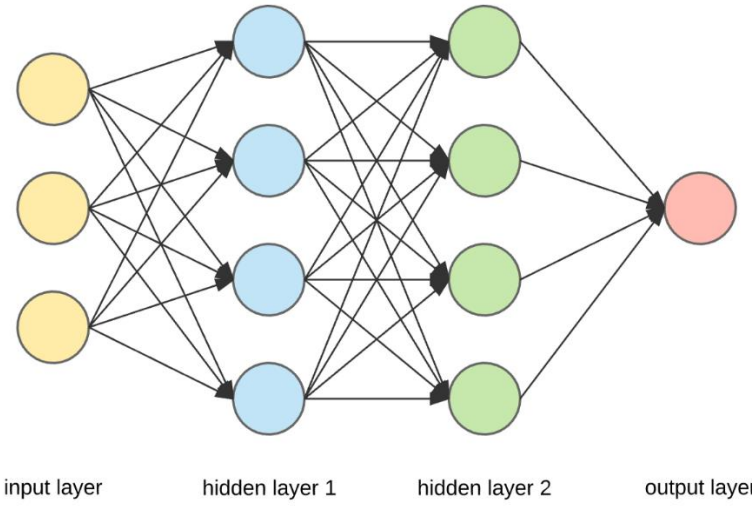
CHAPTER 3: PRELIMINARY DESIGN:-

3.1 Tools And Technologies Used:-

3.1.1 Technologies Used:-

3.1.1.1 Artificial Neural Networks:

An Artificial Neural Network in the field of Artificial intelligence where it attempts to mimic the network of neurons makes up a human brain so that computers will have an option to understand things and make decisions in a human-like manner. The artificial neural network is designed by programming computers to behave simply like interconnected brain cells.



3.1.1.2 Data Science:

Data science is the field of study that combines domain expertise, programming skills, and knowledge of mathematics and statistics to extract meaningful insights from data. Using Data Science we prepare our data to feed it into a deep learning model.

3.1.1.3 Transfer Learning:

Transfer learning is a machine learning technique where a model trained on one task is re-purposed on a second related task. We have used the concept of transfer learning to improve our accuracy while saving time and resources at the same time.

3.1.2 Tools :-

3.1.2.1 Python:

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. This whole project is developed using python and its module. We used python as it is the most promising and easy implementing language for ML and Deep learning Algorithms.

3.1.2.2 Jupyter Notebook:

The Jupyter Notebook is the original web application for creating and sharing computational documents. It offers a simple, streamlined, document-centric experience. We have used this as our Integrated Development Tool (IDE) for writing python codes since it provides an interactive interface.

3.1.2.3 Anaconda Prompt:

Anaconda Prompt is a command line interface with Anaconda Distribution. Terminal is a command line interface that comes in Linux. We have used anaconda prompt as a terminal for running linux command to perform different operations like connecting to TensorBoard, installing different python modules, etc.

3.1.2.4 Firefox :

Mozilla Firefox, or simply Firefox, is a free and open-source web browser developed by the Mozilla Foundation. In our project, Firefox is used as a software for running jupyter notebook, and as an internet explorer to gather information regarding the project.

3.1.3 Modules:

3.1.3.1 NumPy:

NumPy is a Python library used for working with arrays. It also has functions for working in domain of linear algebra, matrices. We have used numpy for storing our image data in 3-d array format.

3.1.3.2 Pathlib:

This module offers classes representing filesystem paths with semantics appropriate for different operating systems. Path classes are divided between pure paths, which provide purely computational operations. We used pathlib to get pure paths so as to extract images data in a much more easy and reliable way.

3.1.3.3 Matplotlib:

Matplotlib is a comprehensive library for creating static, animated, and interactive visualizations in Python. We used matplotlib for visualizing graphs and images to analyse the best parameter fit for our model.

3.1.3.4 OpenCV:

OpenCV (Open Source Computer Vision Library) is an open source computer vision and machine learning software library. OpenCV was built to provide a common infrastructure for computer vision applications and to accelerate the use of machine perception in the commercial products. We used this module to for rescaling , img to rgb 3d matrix conversion, for getting live image from camera etc.

3.1.3.5 Tensorflow:

TensorFlow is an open-source library developed by Google primarily for deep learning applications. It also supports traditional machine learning. We have used tensorflow's different sub-modules like keras Sequential for developing our neural network , tensorflow-hub for loading the feature vector for transfer learning, TensorBoard to visualize the performace of Deep learning model through Bar graphs, historgrams and other analysing tools.

CHAPTER 4: FINAL ANALYSIS AND DESIGN :-

4.1 Model Implementation :-

4.1.1 Data Gathering :

For our project we imported a image data set from Kaggle. The data set we get is divided into two classes : Training data , Validation . Further it is divided into seven sub classes : i.e. Anger, Fear, Happy, Sad, Disgust, Surprise, Neutral. Our dataset contains approx. 32000 raw images which is pre classified in the categories shown above.

4.1.2 Data Preprocessing :

4.1.2.1 Resizing:

Resizing images into a standard dimention is very essential, since our deep learning model needs to create a generalized model to handle all the images, we reduced the size of image from random sizes to (224 X 224 X 3) using OpenCV module.

4.1.2.2 Scaling:

Through OpenCV module we get an image in the form of 3-D vector with RGB value ranges between (0-255), We then Scale it within the range (0-1). We scale all the image vector before feeding it to a Deep Learning model because it reduces the complexity and tends to perform well on when feeded with data that ranegs between 0 to 1.

4.1.3 Building ANN Model :

For building an ANN model, we apply transfer learning to form CNN (Convolutional Neural Network) to classify image.

4.1.4 Training :

Training data is the initial dataset you use to teach a machine learning application to recognize patterns or perform to your criteria.

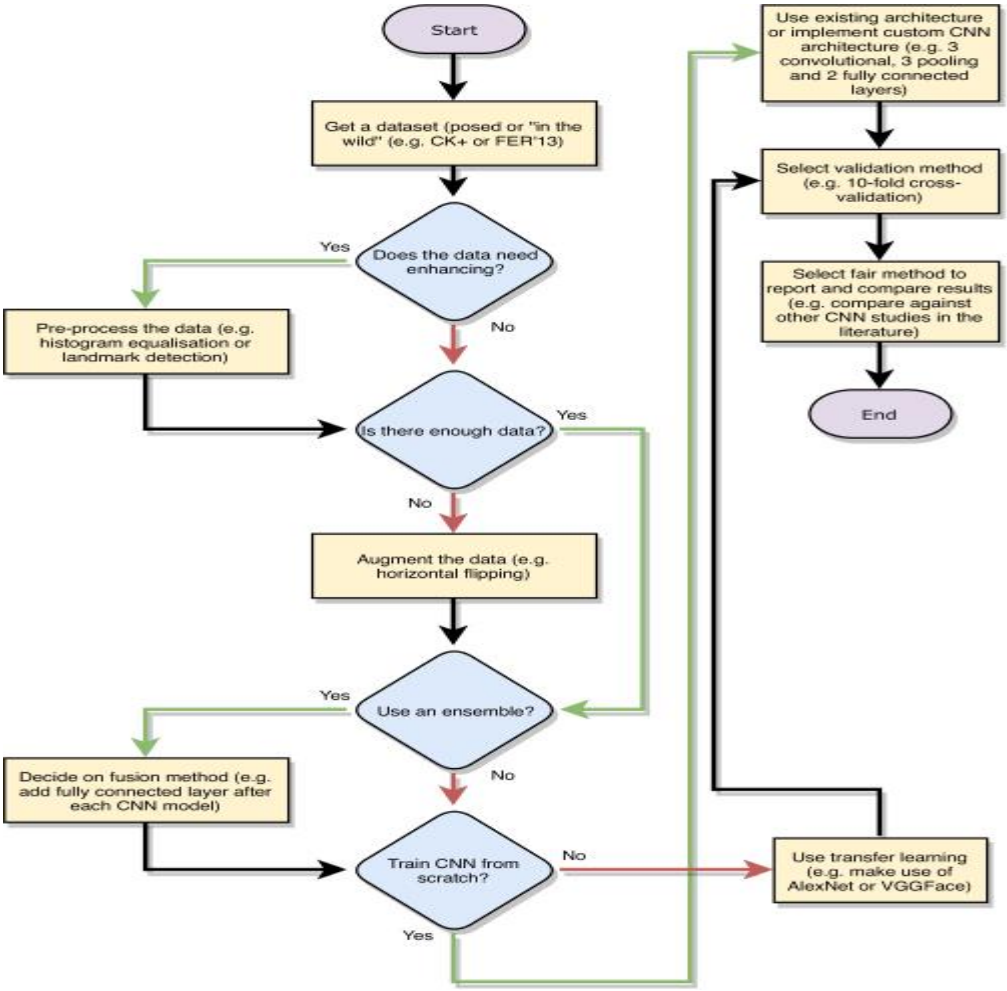
4.1.5 Interactive Interface :

With the help of OpenCv, we designed an interactive live video capturing software that detects the emotion on the basis of image.

4.1.6 Testing :

Testing or validation data is used to evaluate your model's accuracy.

4.2Flowchart :-



4.3 Code Snippets:

4.3.1 Data Gathering :

```
import pathlib
folder_path = "C:/Users/hp/images"
img = load_img("C:/Users/hp/images/train/fear" + "/" + os.listdir("C:/Users/hp/images/train/fear")[0])
plt.imshow(img)

data_dir = pathlib.WindowsPath(folder_path)
data_dir
```

4.1.2 Data Preprocessing :

4.1.2.1 Resizing:

```
def generate_dateset(Type):
    Emotion_dict = {
        "angry": list(data_dir.glob(Type + '/angry/*.jpg'))[0:1000],
        "disgust": list(data_dir.glob(Type + '/disgust/*.jpg')),
        "fear": list(data_dir.glob(Type + '/fear/*.jpg'))[0:1000],
        "happy": list(data_dir.glob(Type + '/happy/*.jpg'))[0:1000],
        "neutral": list(data_dir.glob(Type + '/neutral/*.jpg'))[0:1000],
        "sad": list(data_dir.glob(Type + '/sad/*.jpg'))[0:1000],
        "surprise": list(data_dir.glob(Type + '/surprise/*.jpg'))[0:1000],
    }

    Emotion_label = {
        "angry": 0,
        "disgust": 1,
        "fear": 2,
        "happy": 3,
        "neutral": 4,
        "sad": 5,
        "surprise": 6,
    }
    X, y = [], []

    for emotion, images in Emotion_dict.items():
        for image in images:
            img = cv2.imread(str(image))
            resized_img = cv2.resize(img, (224, 224))
            X.append(resized_img)
            y.append(Emotion_label[emotion])

    X = np.array(X)
    y = np.array(y)

    return X, y
```

4.1.2.2 Scaling:

```
X_train, y_train = generate_dateset("train")
X_train = X_train / 255.0
```

4.1.3 Building ANN Model :

```
classifier = tf.keras.Sequential([
    hub.KerasLayer("https://tfhub.dev/google/tf2-preview/mobilenet_v2/feature_vector/4",
                   input_shape = (224, 224, 3),
                   trainable = False)
])
classifier.build([None, 224, 224, 3])

model = keras.models.Sequential([
    classifier,
    layers.Dense(350, activation='relu'),
    layers.Dense(7, activation='softmax')
])
model.compile(optimizer='SGD', loss='sparse_categorical_crossentropy', metrics=['accuracy'])
```

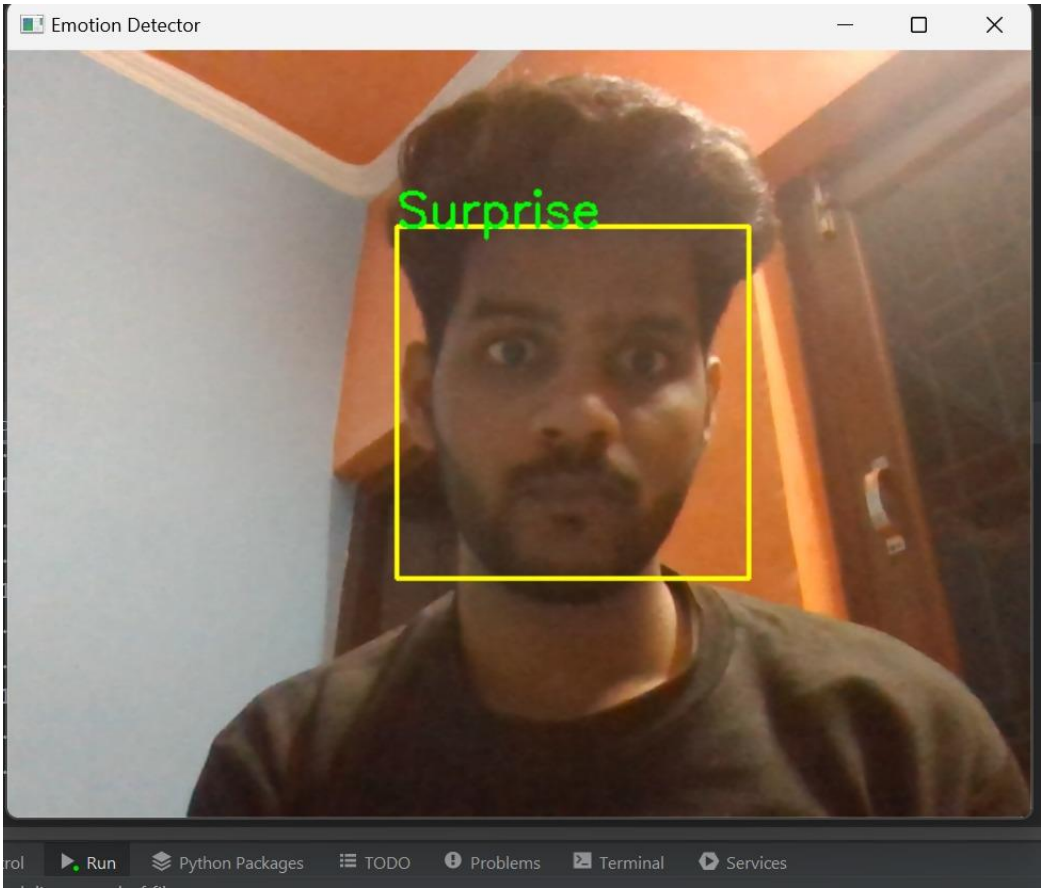
4.1.4 Training :

```
model.fit(X_train, y_train, epochs=40, batch_size = 10 )

model.save('black_box_3.h5')

```

4.1.5 Interactive Interface :

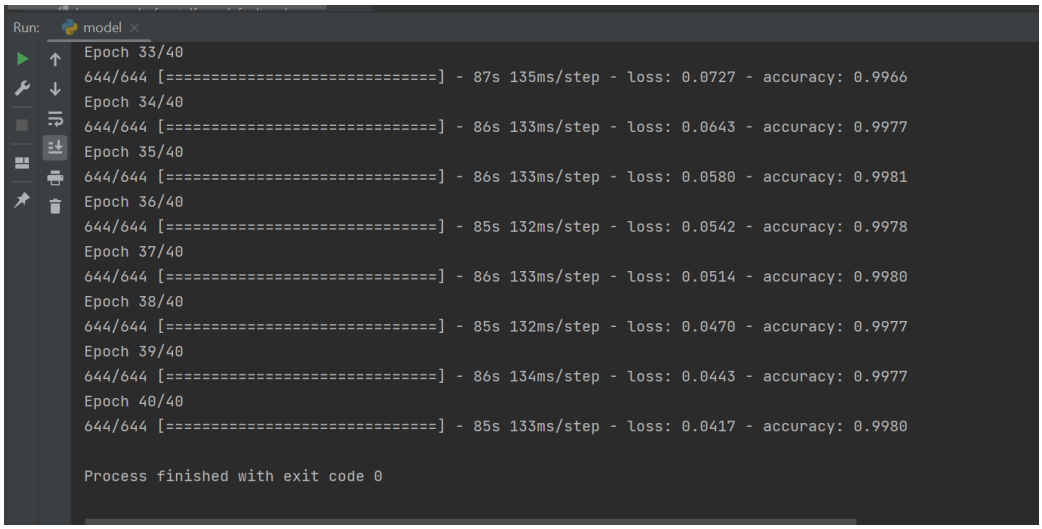
```
face_classifier = cv2.CascadeClassifier(r"C:\Users\hp\IdeaProjects\minor_project\haarcascade_frontalface_default.xml")

classifier = my_reloaded_model = tf.keras.models.load_model(
    r"C:\Users\hp\IdeaProjects\minor_project\black_box_3.h5",
    custom_objects={'KerasLayer': hub.KerasLayer}
)

emotion_labels = ['Angry', 'Disgust', 'Fear', 'Happy', 'Neutral', 'Sad', 'Surprise']

cap = cv2.VideoCapture(0)

while True:
    _, frame = cap.read()
    labels = []
    # gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
    faces = face_classifier.detectMultiScale(frame)

    for (x, y, w, h) in faces:
        cv2.rectangle(frame, (x, y), (x + w, y + h), (0, 255, 255), 2)
        # roi_gray = gray[y:y + h, x:x + w]
        roi_gray = cv2.resize(frame, (224, 224))

        if np.sum([roi_gray]) != 0:
            roi = roi_gray.astype('float') / 255.0
            roi = img_to_array(roi)
            roi = np.expand_dims(roi, axis=0)
```

```
            prediction = classifier.predict(roi)[0]
            print(prediction)
            label = emotion_labels[prediction.argmax()]
            label_position = (x, y)
            cv2.putText(frame, label, label_position, cv2.FONT_HERSHEY_SIMPLEX, 1, (0, 255, 0), 2)
        else:
            cv2.putText(frame, 'No Faces', (30, 80), cv2.FONT_HERSHEY_SIMPLEX, 1, (0, 255, 0), 2)
    cv2.imshow('Emotion Detector', frame)
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break

cap.release()
cv2.destroyAllWindows()

```



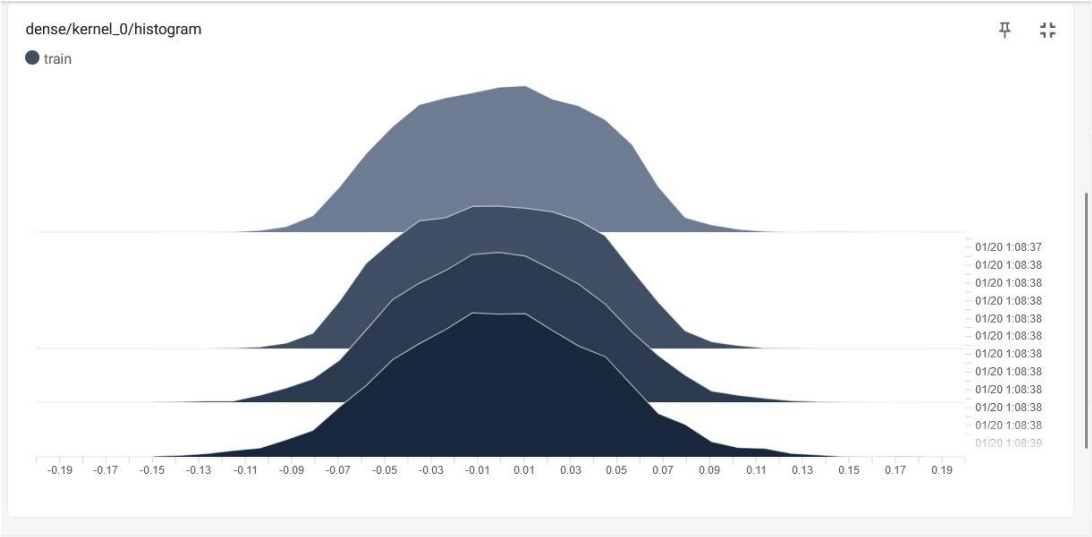
4.4 Result :-

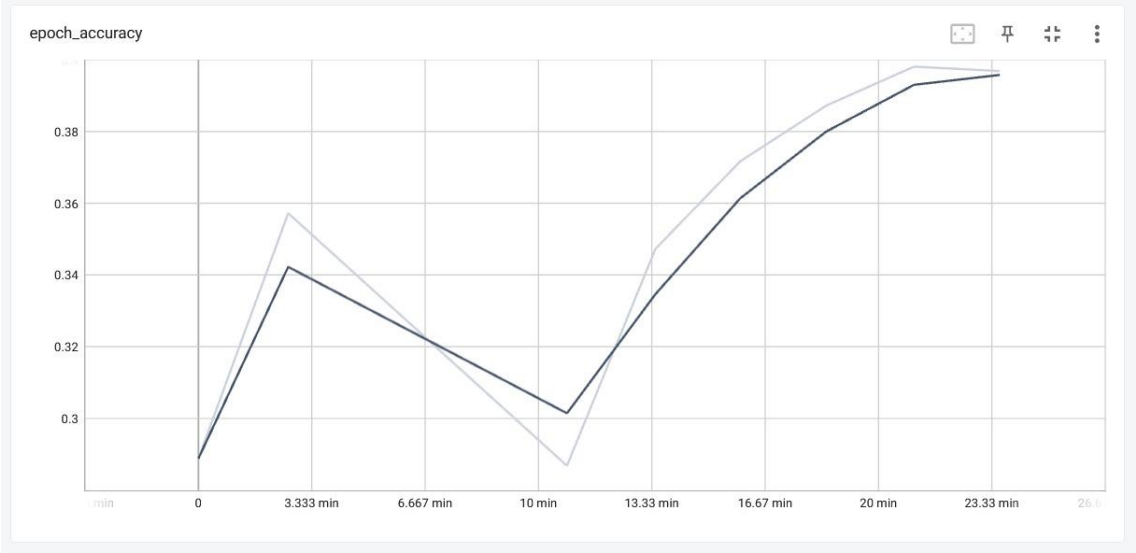
After completion of this project, we have successfully developed a CNN Deep Learning model which can classify human emotions. In our model we have achieved an accuracy of 99.8% on training data and an accuracy of 45% on a random testing data. We have tested the model on all aspects and can asure that this model will outperform in maximum situations.

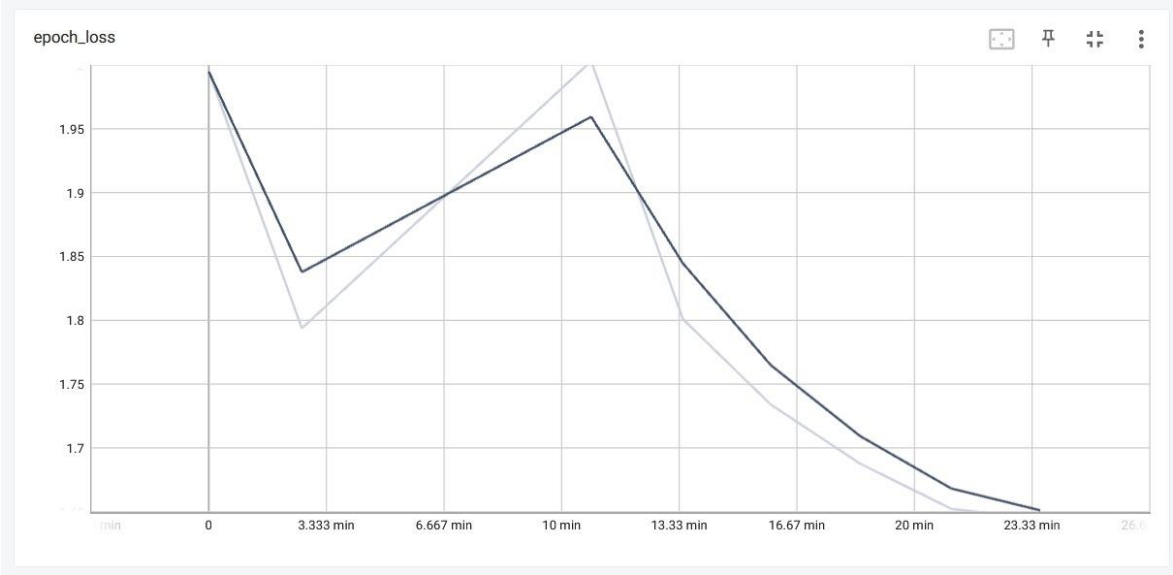


```
Python Console
    ...:
    ...:
    ...: X_test, y_test = generate_dateset("validation")
    ...: X_test = X_test[0:4000] / 255.0
    ...:
    ...: y_pre = classifier.predict(X_test)
    ...: y_predicted = [np.argmax(i) for i in y_pre]
    ...: count = 0
    ...: for i in range(0,len(y_predicted)):
    ...:     if y_test[i] == y_predicted[i]:
    ...:         count += 1
    ...: acc = (count/len(y_predicted))* 100
    ...: print("Accuracy of our Emotion detection modelon test data is:", acc)
    ...:
125/125 [==============================] - 67s 480ms/step
Accuracy of our Emotion detection modelon test data is: 44.375

In [9]:
```

4.5 Result Analysis :-







4.6Application :-

4.6.1 Medicine

4.6.2 E - learning

4.6.3 Entertainment

4.6.4 Marketing

4.6.5 Monitoring

CONCLUSION :-

We have presented an emotion detection algorithm by using facial recognition method. The algorithm composed of three stages. In image processing stage, facial components extraction and emotion classification. The histogram analysis methods are adopted to analyse face region parameters from facial image. The facial components then are extracted by using CNN Deep Learning model. To extract feature for emotion detection, the new feature extraction methods are proposed in feature extraction stage. The emotion of facial image is detected by using CNN classifier using Transfer Learning. In the experimental results, it is shown that the proposed algorithm detect emotion well.

REFERENCES :-

1. lliana Azizan, Fatimah Khalid Facial Emotion Recognition: A Brief Review International Conference on Sustainable Engineering, Technology and Management (ICSETM -2018), Dec. 20, 2018, Negeri Sembilan, Malaysia.
2. Jonathan Oheix, Face Expression Recognition Data Set, KAGGLE.
3. tf2-preview/mobilenet_v2/feature_vector published by GOOGLE with Licence Apache 2.0 based over MobileNet V2 architecture on ImageNet (ILSVRC-2012-CLS) dataset.
4. www.wikipedia.com