Emotion Recognition

A Project Report

Submitted in partial fulfilment of the

Requirements for award of the Degree of

BACHELOR OF SCIENCE (COMPUTER SCIENCE)

By

Zainabbas Merchant

Seat No:______

Under the Esteemed Guidance of

Prof.Javed Pathan

Assistant Professor

DEPARTMENT OF COMPUTER SCIENCE

RIZVI COLLEGE OF ARTS SCIENCE AND COMMERCE

(Affiliated to University of Mumbai)

Mumbai, Pin Code-400050

MAHARASHTRA

2022-2023

RIZVI COLLEGE OF ARTS SCIENCE AND COMMERCE

(Affiliated to University of Mumbai)

MUMBAI MAHARASTRA-400050

DEPARTMENT OF COMPUTER SCIENCE

CERTIFICATE

This is to certify that the project entitled: **“Emotion Recognition”,** is benefited work of **Zainabbas Merchant** bearingseat No: _ **Roll no.** 14 Submitted in partial fulfilment of the requirements for the award of Degree of BACHLOR OF SCIENCE inCOMPUTER SCIENCE from UNIVERSITY OF MUMBAI.

Internal Guide **HOD**

External Examiner

Date: _________ **College Seal**

ACKNOWLEDGEMENT

I owe special thanks to the Department of COMPUTER SCIENCE of **RIZVI COLLEGE OF ARTS SCIENCE AND COMMERCE** for giving me a change to prepare this project dissertation.

I thank to the principal, **Dr. (Mrs) Anjum Ara Ahmad** for her Leadership and Management.

I thank to the coordinator and Head of the Department **Professor Arif Patel** for providing us the required facilities and guidance throughout the course which culminated into this thesis.

Last but not the least to the Project guide this semester – **Professor Javed Pathan sir.** Deep gratitude to the staff and faculty of **RIZVI COLLEGE OF ARTS SCIENCE AND COMMERCE** for their help and support.

ABSTRACT

Emotion recognition is a rapidly growing field that aims to develop artificial intelligence systems capable of detecting and interpreting human emotions from various sources, including facial expressions, voice, physiological signals, and textual data. This technology has a wide range of applications, including healthcare, marketing, and human-computer interaction. Emotion recognition systems employ various machine learning algorithms such as deep learning, neural networks, and support vector machines, to analyze and classify emotional states. The accuracy of these systems is improving as the datasets and algorithms become more sophisticated. However, challenges still exist in developing robust emotion recognition systems that can accurately identify emotions across different cultures, languages, and contexts.

DECLARATION

I hereby declare that the project entitled, **“Emotion Detection”** done at **Rizvi College of Arts, Science and Commerce**, has not been in any caseduplicated to submit to any other university for the award of any degree. To the best of my knowledge other than me, no one has submitted to any other university.

The project is done in partial fulfilment of the requirements for the award of degree of **BACHELOR OF SCIENCE (COMPUTER SCIENCE)** to be submitted as final semester project as part of our curriculum.

Zainabbas Merchant

Table of Contents

CHAPTER 1. INTRODUCTION

1.1 Introduction to the project

Emotion detection, also known as emotion recognition or affective computing, is an emerging technology that seeks to develop computer systems capable of identifying and interpreting human emotions. This technology is becoming increasingly important in various fields, including healthcare, marketing, and human-computer interaction, where understanding and responding to human emotions can greatly enhance the user experience.

Emotion detection systems rely on various sources of data to identify emotional states, including facial expressions, voice, physiological signals, and textual data. These systems employ machine learning algorithms such as deep learning, neural networks, and support vector machines to analyze and classify emotional states.

Emotion detection has the potential to revolutionize how we interact with technology and how technology interacts with us. For instance, in healthcare, emotion detection can be used to monitor patients' emotional states and alert caregivers in case of potential mental health issues. In marketing, emotion detection can help companies to better understand their customers' needs and preferences and to develop more effective advertising strategies.

Despite the progress made in this field, several challenges still exist in developing robust emotion detection systems that can accurately identify emotions across different cultures, languages, and contexts. This paper will explore the current state of emotion detection technology, its applications, and the challenges that need to be addressed to further advance this field.

1.2 Problem Definition

The project deals with emotion recognition to find the facial expression of the user.

1.3 Aim

This project is built to detect facial emotions of a user which can be used in several other applications dealing with alert systems and music suggestions.

1.4 Objective

To build a system that recognizes the emotions of a user and is able to display it on the screen.

1.5 Goal

1. Improving healthcare: Emotion detection can be used in healthcare to monitor patients' emotional states and alert caregivers in case of potential mental health issues.
2. Enhancing marketing: Emotion detection can help companies to better understand their customers' needs and preferences and to develop more effective advertising strategies.
3. Improving human-computer interaction: Emotion detection can help to create more natural and intuitive interfaces for human-computer interaction, making technology more responsive and user-friendly.
4. Enhancing education: Emotion detection can be used in education to monitor students' emotional states and provide personalized learning experiences based on their emotional needs.

1.6 Need of the System

Emotion detection is becoming increasingly important in various fields because of the need to understand and respond to human emotions. To be able to find out if a users likes and dislikes.

CHAPTER 2. REQUIREMENT AND ANALYSIS

2.1 Introduction: -

In this chapter we mentioned the software and hardware requirements, which are necessary for successfully running this system. The major element in building systems is selecting compatible hardware and software. The system analyst has to determine what software package is best for the "Emotion Recognition" and, where softwareis not an issue, the kind of hardware and peripherals needed for the final conversion.

2.2 System Environment: -

An emotion detection system that uses facial expressions as a source of data can operate in a variety of environments, including public spaces, medical settings, and social media platforms. The system can use cameras or other sensors to capture facial expressions and then analyze them to determine the emotional state of the person. Software and real system are identified. According to the provided functional specification all the technologies and its capacities are identified. Basic functions and procedures and methodologies are prepared to implement. Some of the Basic requirements such as hardware and software are described as follows: -

2.3 Software Requirement: -

- Technology: Python
- IDE: VS Code
- Client-Side Technologies: OpenCV
- Operating System: Microsoft Windows/Linux

2.4 Hardware Requirement: -

- Processor: Pentium-III (or) Higher
- Ram: 64MB (or) Higher
- Hard disk: 80GB (or) Higher

2.5 Methodology: -

The Software development model or system development methodology in software engineering is a framework that is used to structure, plan as well as control the process of developing an information. For any project to be completed, it has to go through stages called Development Life Cycle.

The five phases of the project are as follows:

Scoping and planning

This phase focus on the planning of the project's overall direction, including the definition of the project's scope, objectives, and timelines. The deliverable from this phase is this Design Plan.

Conceptual design and research

In this phase, the conceptual design of the methodology is developed and research on existing methodologies is conducted. Research is performed from independent research firms, such as the Gartner Group, Forrester Research, and CIO.com. These research firms sometimes publish the methodologies that consulting firms use. Consulting firms' websites are another source for researching E-commerce strategy methodologies.

Development of methodology

The actual methodology is developed in this phase. Detailed descriptions of each task in the methodology are documented, including the objectives, inputs, approach, relevant models, applicable tools and techniques, outputs, and any references. The methodology is to be documented in an appropriate format, be it a Word document or HTML pages.

Implementation of methodology

The methodology will be implemented with a client. This phase includes the marketing of E-commerce strategy development services and the closing of the sale, followed by the actual implementation.

Revision of methodology

Final touches and revisions to the methodology are made in this phase. The majority of these revisions come from experiences on the client project. Sample reports and any additional references are added to the methodology.

CHAPTER 3. SYSTEM ANALYSIS

3.1 Purpose: -

To detect facial emotions and show them on screen. To aquire a data set that can help in various applications such as user satisfaction and suggestions.

3.2 Project Scope: -

The project has a wide scope, as it is not intended to a particular organization. This project is going to develop generic software, which can be applied by any business's organization. Moreover, it provides facility to its customer. Also, the software is going to provide a huge amount of summary data.

3.3 Project Scheduling: -

Project scheduling is a mechanism to communicate what tasks need to get done and which organizational resources will be allocated to complete those tasks in what timeframe. A project schedule is a document collecting all the work needed to deliver the project on time.

A project is made up of many tasks, and each task is given a start and end (or due date), so it can be completed on time. Likewise, people have different schedules, and their availability and vacation or leave dates need to be documented in order to successfully plan those tasks.

3.4 Project Planning: -

Project planning is part of project management, which relates to the use of schedules such as Gantt charts to plan and subsequently report progress within the project environment.

Initially, the project scope is defined and the appropriate methods for completing the project are determined. Following this step, the durations for the various tasks necessary to complete the work are listed and grouped into a work breakdown structure. Project planning is often used to organize different areas of a project, including project plans, workloads and the management of teams and individuals.

The logical dependencies between tasks are defined using an activity network diagram that enables identification of the critical path. Project planning is inherently uncertain as it must be done before the project is actually started. Therefore, the duration of the tasks is often estimated through a weighted average of optimistic, normal, and pessimistic cases.

The critical chain method adds "buffers" in the planning to anticipate potential delays in project execution. Float or slack time in the schedule can be calculated using project management software. Then the necessary resources can be estimated and costs for each activity can be allocated to each resource, giving the total project cost. At this stage, the project schedule may be optimized to achieve the appropriate balance between resource usage and project duration to comply with the project objectives. Once established and agreed, the project schedule becomes what is known as the baseline 11 schedule.

Progress will be measured against the baseline schedule throughout the life of the project. Analysing progress compared to the baseline schedule is known as earned value management. The inputs of the project planning phase 2 include the project charter and the concept proposal. The outputs of the project planning phase include the project requirements, the project schedule, and the project management plan.

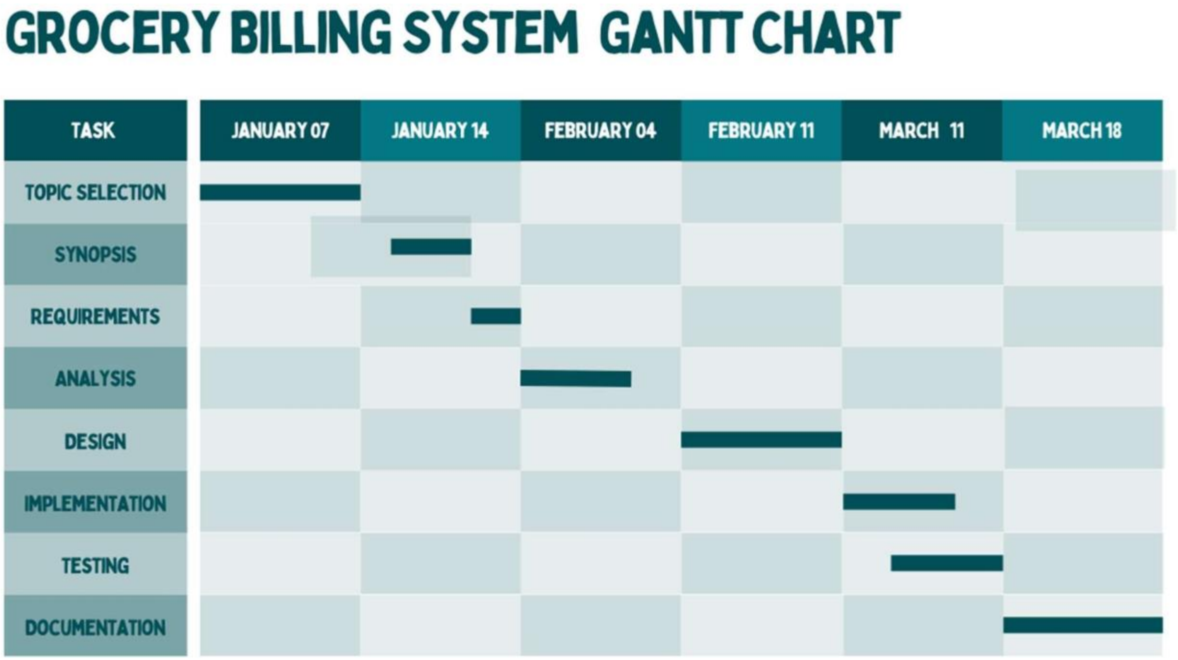
3.5 Gantt Chart: -

A Gantt chart is a type of bar chart that illustrates a project schedule. This chart lists the tasks to be performed on the vertical axis, and time intervals on the horizontal axis.

Terminal elements and summary elements constitute the work breakdown structure of the project. Modern Gantt charts also show the dependency relationships betweenactivities.

Gantt charts can be used to show current schedule status using percent-complete shadings and a vertical "TODAY" line as shown here. Gantt charts are sometimes equated with bar charts.

Gantt charts are usually created initially using an early start time approach, where each task is scheduled to start immediately when its prerequisites are complete. This method maximizes the float time available for all tasks. Gantt chart of this project is on next page.



CHAPTER 4. SURVEY OF TECHNOLOGY

4.1 Python: -

Learned Python: Learning Python does not have to be a difficult. In fact, if you are using the right resources, it can actually be easy.

Set up Python: I downloaded and installed Python on my computer. I wrote a few basic scripts to test my installation and confirmed that it was working. We have import few functions of python to run this project.

Design and Create Database: For better, fast & easy to understand database, it is begin created in a local disk, in the local machine.

Why Python

- It has a **simple syntax** that mimics natural language, so it’s easier to read and understand. This makes it quicker to build projects, and faster to improve on them.
- It is **versatile**. Python can be used for many different tasks, from web development to machine learning.
- It is **beginner friendly**, making it popular for entry-level coders.
- It is **open source**, which means it’s free to use and distribute, even for commercial purposes.
- Python’s archive of **modules and libraries**—bundles of code that third-party users have created to expand Python’s capabilities—is vast and growing.
- Python has a **large and active community** that contributes to Python’s pool of modules and libraries, and acts as a helpful resource for other programmers. The vast support community means that if coders run into a stumbling block, finding a solution is relatively easy; somebody is bound to have encountered the same problem before.

What is Python?

Python is a computer programming language often used to build websites and software, automate tasks, and conduct data analysis. Python is a general-purpose language, meaning it can be used to create a variety of different programs and isn’t specialized for any specific problems. This versatility, along with its beginner-friendliness, has made it one of the most-used programming languages today. A survey conducted by industry analyst firm RedMonk found that it was the second-most popular programming language among developers in 2023

Python is commonly used for developing websites and software, task automation, data analysis, and data visualization. Since it’s relatively easy to learn, Python has been adopted by many non-programmers such as accountants and scientists, for a variety of everyday tasks, like organizing finances.

CHAPTER 5. SYSTEM DESIGN

5.1 Basic Module: -

Camera Display: -

The camera display an instance of the camera to display the face of the user. The system draws a square around the recognized face. The emotions r shown about the square. Multiple users emotions can be displayed at a time.

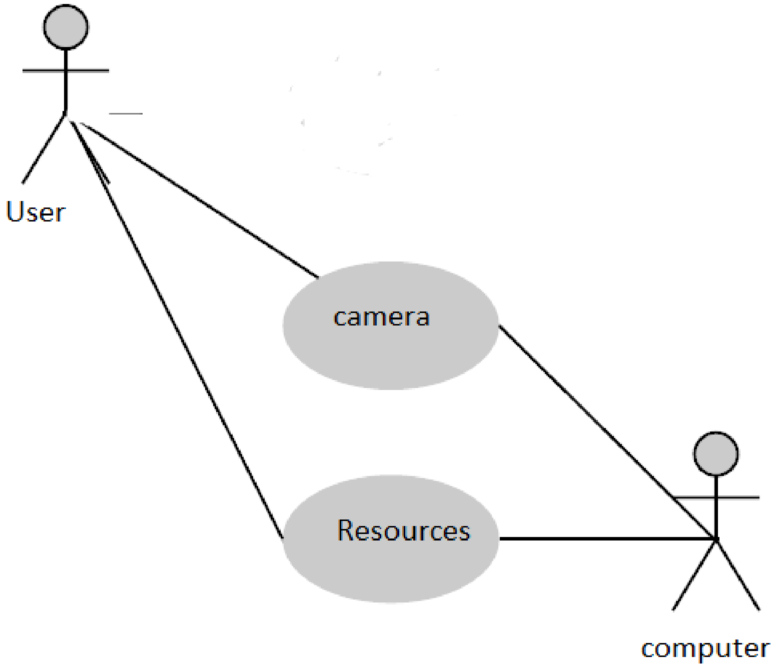
5.2 Use case diagram: -

Use case diagram consists of use cases and actors and shows the interaction between them.

The main purpose is to show the interaction between the use cases and the actor.

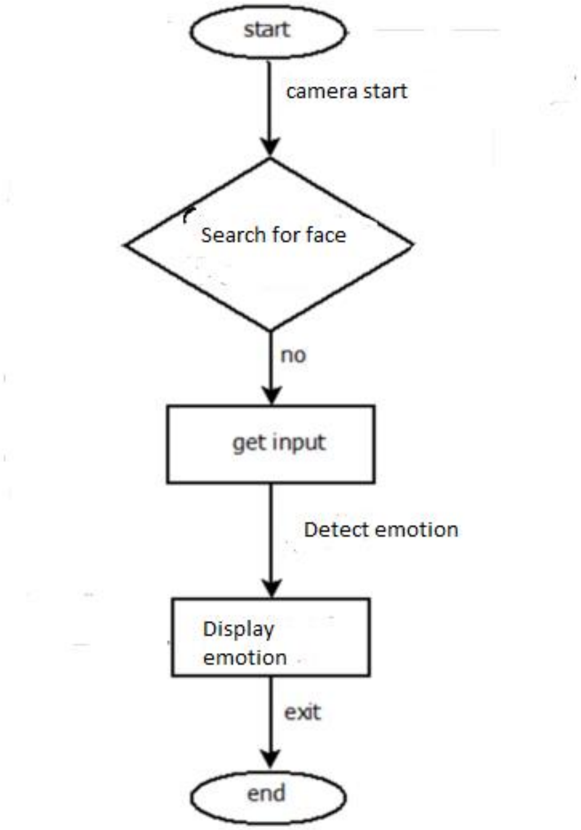
To represent the system requirement from user’s perspective.

The use cases are the functions that are to be performed in the module.



5.3 Data Flow Diagram Notation: -

A data-flow diagram is a way of representing a flow of data through a process or a system (usually an information system). The DFD also provides information about the outputs and inputs of each entity and the process itself. A data-flow diagram has no control flow. there are no decision rules and no loops. Specific operations based on the data can be represented by a flowchart.

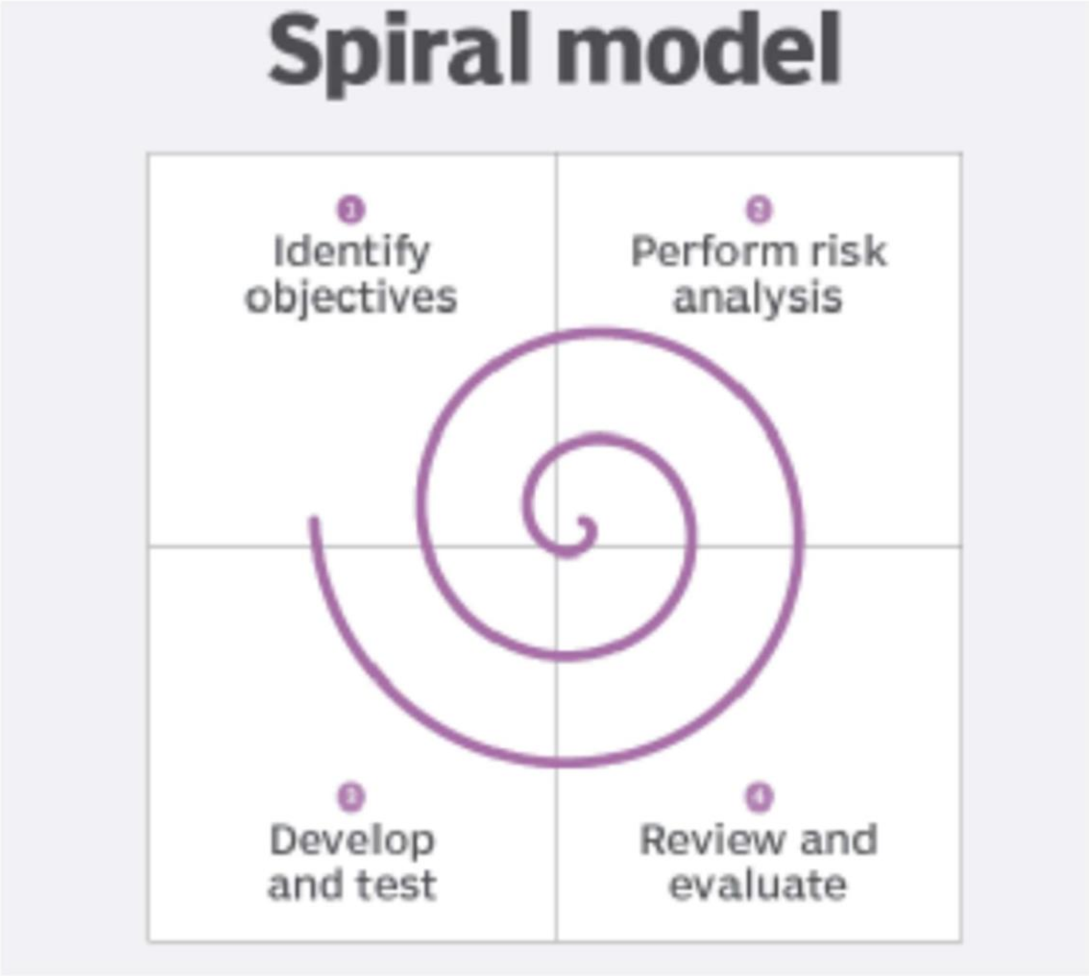


5.4 Spiral Model: -

In order to achieve goals and planned results within a defined schedule and a budget, a methodology in used in a project. Regardless of which field or which trade, there are assortments of methodologies to help managers at every stage of a project from the initiation to implementation to the closure.

A methodology is a model, which project managers employ for the design, planning, implementation, and achievement of their project objectives. There are different project management methodologies to benefit different projects.

The spiral model, initially proposed by Boehm, is an evolutionary softwareprocess model that couples the iterative feature of prototyping with the controlledand systematic aspects of the linear sequential model. It implements the potentialfor rapid development of new versions of the software. Using the spiral model, the software is developed in a series of incremental releases. During the early iterations, the additional release may be a paper model or prototype. During later iterations, more and more complete versions of the engineered system are produced.



Each cycle in the spiral is divided into four parts:

1. **Objective setting:** Each cycle in the spiral starts with the identification of purpose for that cycle, the various alternatives that are possible for achieving the targets, and the constraints that exists.

2. **Risk Assessment and reduction:** The next phase in the cycle is to calculate these various alternatives based on the goals and constraints. The focus of evaluation in this stage is located on the risk perception for the project.

3. **Development and validation:** The next phase is to develop strategies that resolve uncertainties and risks. This process may include activities such as benchmarking, simulation, and prototyping.

4. **Planning:** Finally, the next step is planned. The project is reviewed, and a choice made whether to continue with a further period of the spiral. If it is determined to keep, plans are drawn up for the next step of the project. The development phase depends on the remaining risks. For example, if performance or user-interface risks are treated more essential than the program development risks, the next phase may be an evolutionary development that includes developing a more detailed prototype for solving the risks.

CHAPTER 6. SYSTEM IMPLEMENTATION

Project implementation (or project execution) is the phase where visions and plans become reality. This is the logical conclusion, after evaluating, deciding, visioning, planning, applying for funds and finding the financial resources of a project.

Implementation simply means carrying out the activities described in your work plan.

Executing a project in the water and sanitation sector is a very complex mission, as it requires the coordination of a wide range of activities, the overseeing of a team, the management of budget, the communication to the public, among other issues.

The following lines will give you an introduction into the implementation of projects in sustainable sanitation and water management, and highlights key aspects that have to be considered for a successful implementation.

It is important to take into account that independently of the nature of the project, implementation takes time, usually more than it is planned, and that many external constraints can appear, which should be considered when initiating the implementation step (i.e. seasonality in availability of community engagement/resources).

6.1 Coding: -

Under this chapter all the coding of the respective project "**Emotion Detection**" is shown below.

Code: -

Face.py

```
from keras.models import load_model
import time
from keras_preprocessing.image import img_to_array
from keras.preprocessing import image
import cv2
import numpy as np
from pygame import mixer

mixer.init()
face_classifier = cv2.CascadeClassifier(
    './haarcascade_frontalface_default.xml')
classifier = load_model('./Emotion_Detection.h5')

class_labels = ['Angry', 'Happy', 'Neutral', 'Sad', 'Surprise']

cap = cv2.VideoCapture(0)



while True:
    ret, frame = cap.read()
    labels = []
    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
    faces = face_classifier.detectMultiScale(gray, 1.3, 5)
```

```
for (x, y, w, h) in faces:
    cv2.rectangle(frame, (x, y), (x+w, y+h), (255, 0, 0), 2)
    roi_gray = gray[y:y+h, x:x+w]
    roi_gray = cv2.resize(roi_gray, (48, 48), interpolation=cv2.INTER_AREA)

    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)

      preds = classifier.predict(roi)[0]
      print("\nprediction = ", preds)
      label = class_labels[preds.argmax()]
      print("\nprediction max = ", preds.argmax())
      print("\nlabel = ", label)
      label_position = (x, y)
      cv2.putText(frame, label, label_position,
            cv2.FONT_HERSHEY_SIMPLEX, 2, (0, 255, 0), 3)

      if label != 'Angry':
        mixer.music.load("song3.mp3")
        mixer.music.set_volume(0.5)
        mixer.music.play()

    else:
      cv2.putText(frame, 'No Face Found', (20, 60),
            cv2.FONT_HERSHEY_SIMPLEX, 2, (0, 255, 0), 3)
    print("\n\n")
    cv2.imshow('Emotion Detector', frame)
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break

cap.release()
cv2.destroyAllWindows()
```

train.py

```
from keras.applications import MobileNet
from keras.models import Sequential,Model
from keras.layers import Dense,Dropout,Activation,Flatten,GlobalAveragePooling2D
from keras.layers import Conv2D,MaxPooling2D,ZeroPadding2D
from keras.layers.normalization import BatchNormalization
from keras.preprocessing.image import ImageDataGenerator

# MobileNet is designed to work with images of dim 224,224
img_rows,img_cols = 224,224

MobileNet = MobileNet(weights='imagenet',include_top=False,input_shape=(img_rows,img_cols,3))

# Here we freeze the last 4 layers
# Layers are set to trainable as True by default

for layer in MobileNet.layers:
    layer.trainable = True

# Let's print our layers
for (i,layer) in enumerate(MobileNet.layers):
    print(str(i),layer.__class__.__name__,layer.trainable)

def addTopModelMobileNet(bottom_model, num_classes):
    """creates the top or head of the model that will be
    placed ontop of the bottom layers"""
```

```
top_model = bottom_model.output
    top_model = GlobalAveragePooling2D()(top_model)
    top_model = Dense(1024,activation='relu')(top_model)

    top_model = Dense(1024,activation='relu')(top_model)

    top_model = Dense(512,activation='relu')(top_model)

    top_model = Dense(num_classes,activation='softmax')(top_model)

    return top_model

num_classes = 5

FC_Head = addTopModelMobileNet(MobileNet, num_classes)

model = Model(inputs = MobileNet.input, outputs = FC_Head)

print(model.summary())

train_data_dir = '/Users/durgeshthakur/Deep Learning Stuff/Emotion Classification/fer2013/train'
validation_data_dir = '/Users/durgeshthakur/Deep Learning Stuff/Emotion Classification/fer2013/validation'

train_datagen = ImageDataGenerator(
                    rescale=1./255,
                    rotation_range=30,
                    width_shift_range=0.3,
                    height_shift_range=0.3,
                    horizontal_flip=True,
                    fill_mode='nearest'
                                   )
```

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

batch_size = 32

train_generator = train_datagen.flow_from_directory(
                    train_data_dir,
                    target_size = (img_rows,img_cols),
                    batch_size = batch_size,
                    class_mode = 'categorical'
                    )

validation_generator = validation_datagen.flow_from_directory(
                        validation_data_dir,
                        target_size=(img_rows,img_cols),
                        batch_size=batch_size,
                        class_mode='categorical')

from keras.optimizers import RMSprop,Adam
from keras.callbacks import ModelCheckpoint,EarlyStopping,ReduceLROnPlateau

checkpoint = ModelCheckpoint(
                        'emotion_face_mobilNet.h5',
                        monitor='val_loss',
                        mode='min',
                        save_best_only=True,
                        verbose=1)



earlystop = EarlyStopping(
                      monitor='val_loss',
                      min_delta=0,
                      patience=10,
                      verbose=1,restore_best_weights=True)

learning_rate_reduction = ReduceLROnPlateau(monitor='val_acc',
                                    patience=5,
                                    verbose=1,
                                    factor=0.2,
                                    min_lr=0.0001)

callbacks = [earlystop,checkpoint,learning_rate_reduction]

model.compile(loss='categorical_crossentropy',
            optimizer=Adam(lr=0.001),
            metrics=['accuracy']
            )
```

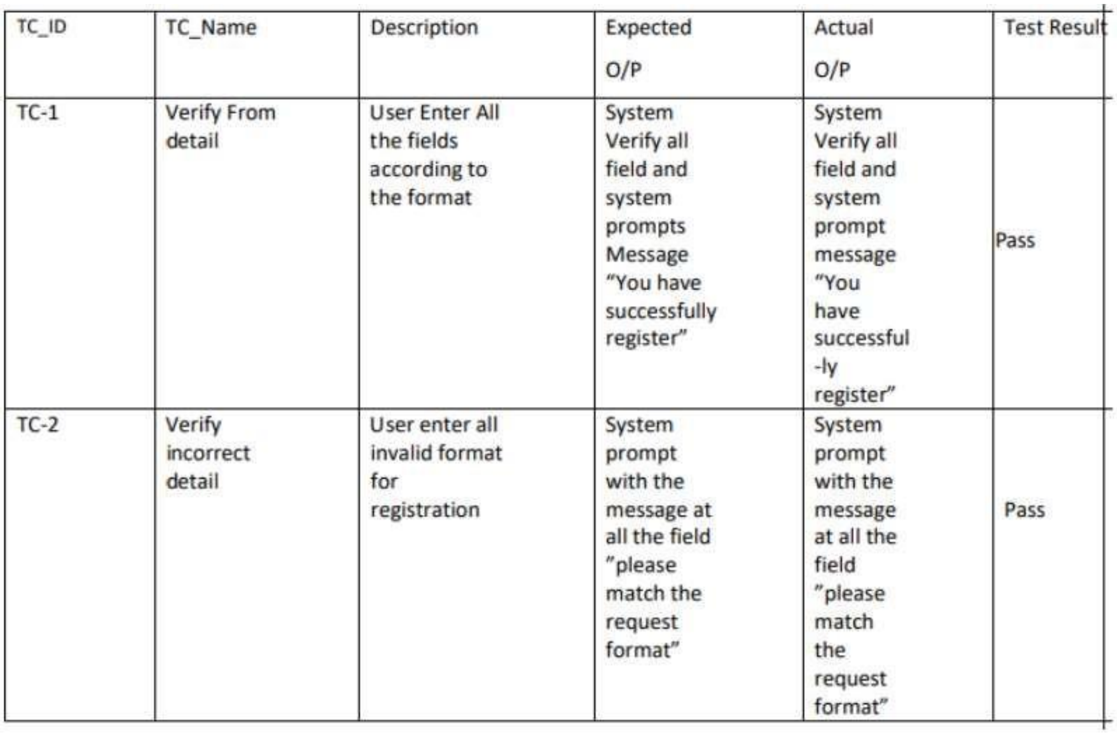
```
nb_train_samples = 24176
nb_validation_samples = 3006

epochs = 25

history = model.fit_generator(
            train_generator,
            steps_per_epoch=nb_train_samples//batch_size,
            epochs=epochs,
            callbacks=callbacks,
            validation_data=validation_generator,
            validation_steps=nb_validation_samples//batch_size)
```

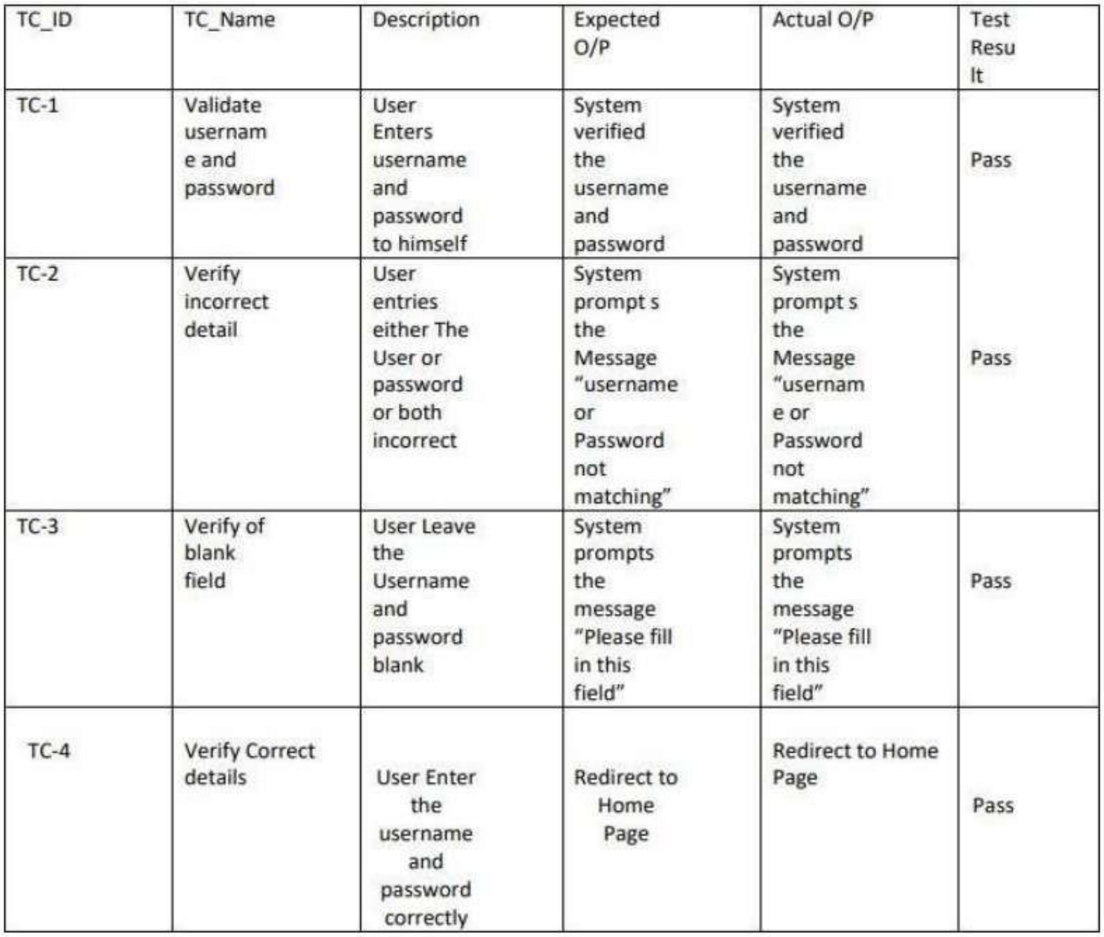
6.2 TEST CASE: -

6.2.1 Main Page: -



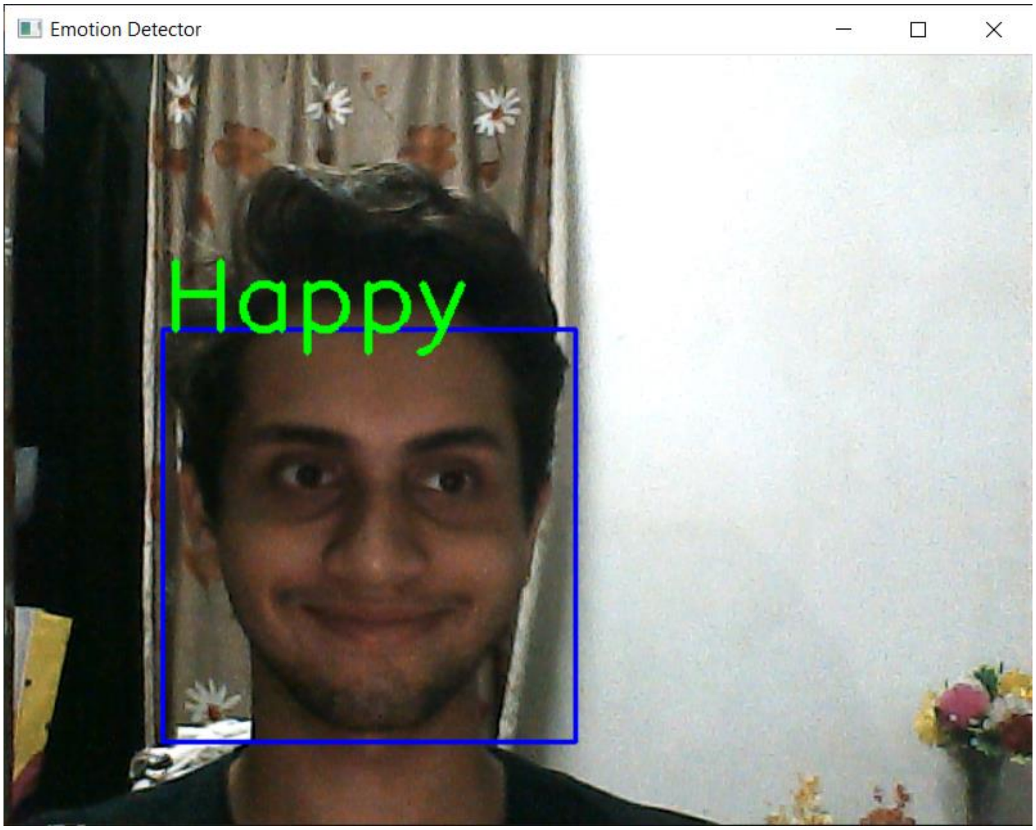
TC_ID	TC_Name	Description	Expected O/P	Actual O/P	Test Resul
TC-1	Verify From detail	User Enter All the fields according to the format	System Verify all field and system prompts Message "You have successfully register"	System Verify all field and system prompt message "You have successful -ly register"	Pass
TC-2	Verify incorrect detail	User enter all invalid format for registration	System prompt with the message at all the field "please match the request format"	System prompt with the message at all the field "please match the request format"	Pass

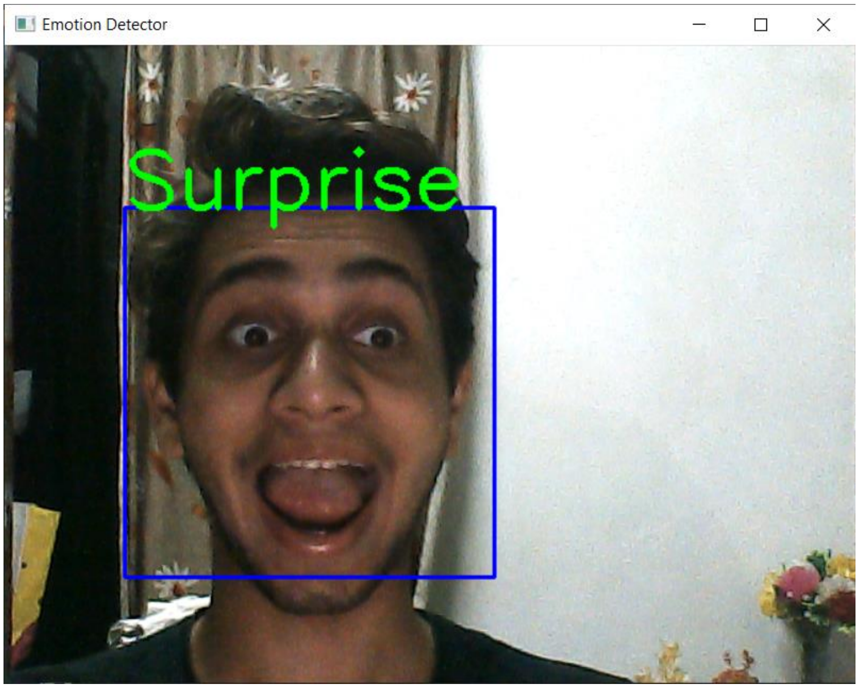
6.2.2 Buttons: -

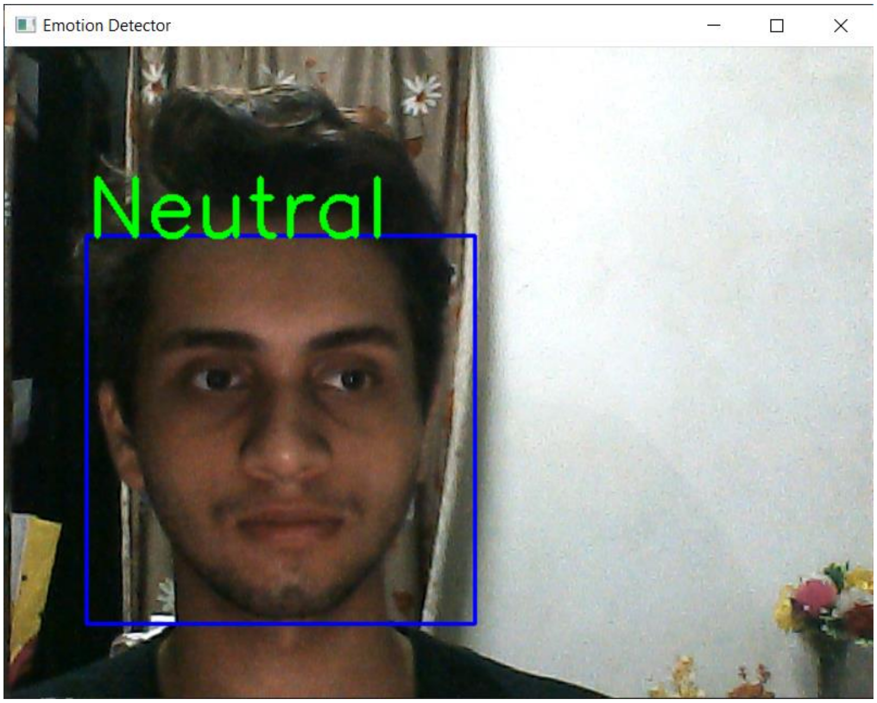


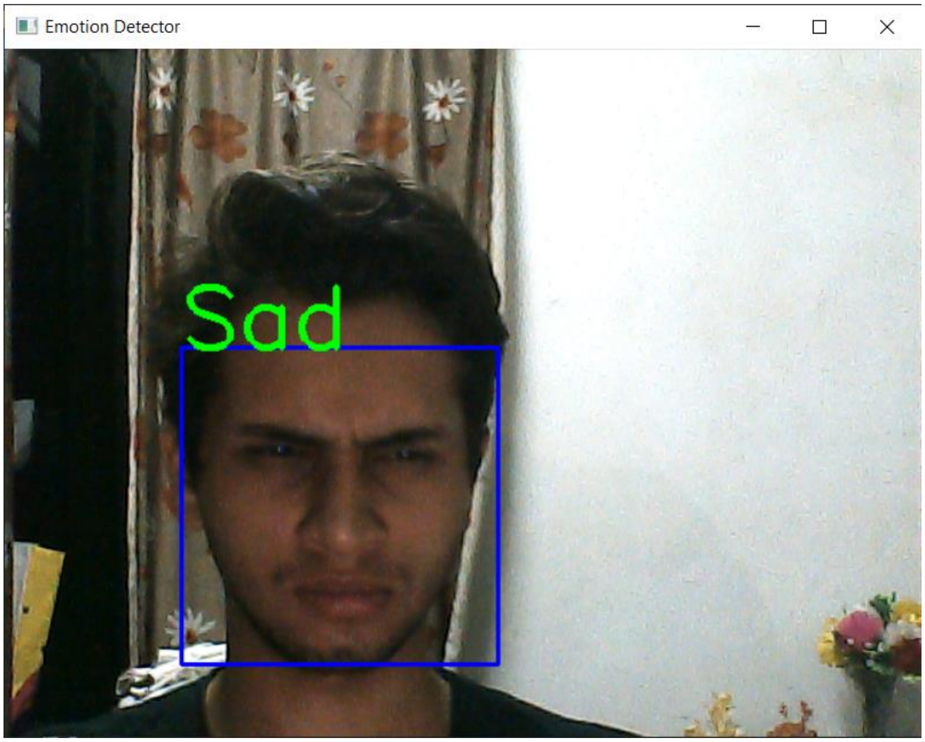
TC_ID	TC_Name	Description	Expected O/P	Actual O/P	Test Result
TC-1	Validate username and password	User Enters username and password to himself	System verified the username and password	System verified the username and password	Pass
TC-2	Verify incorrect detail	User entries either The User or password or both incorrect	System prompt s the Message "username or Password not matching"	System prompt s the Message "username or Password not matching"	Pass
TC-3	Verify of blank field	User Leave the Username and password blank	System prompts the message "Please fill in this field"	System prompts the message "Please fill in this field"	Pass
TC-4	Verify Correct details	User Enter the username and password correctly	Redirect to Home Page	Redirect to Home Page	Pass

CHAPTER 7. RESULT






Emotion Detector
Neutral


Emotion Detector
Sad

CHAPTER 8. CONCLUSION & FUTURE SCOPE

8.1 Feature: -

Emotion detection involves analyzing various features of human expression and behavior to infer emotional states. Some of the common features used for emotion detection include:

Facial expressions: The movements of the muscles in the face, such as the eyebrows, mouth, and eyes, can convey a range of emotions, including happiness, anger, sadness, and surprise.

Voice tone and pitch: The tone and pitch of a person's voice can also reveal emotions. For example, a high-pitched voice might indicate excitement, while a monotone voice might indicate boredom or indifference.

Body language: The way a person stands, sits, and moves can also reveal emotions. For example, crossed arms might indicate defensiveness or discomfort, while slumped shoulders might indicate sadness or defeat.

Word choice: The words a person chooses to use can also reveal their emotional state. For example, someone might use more negative words when they are feeling sad or angry.

Context: The situation or context in which the person is communicating can also provide clues to their emotional state. For example, someone might be more likely to be feeling happy and excited at a party, and more likely to be feeling sad and tired at a funeral.

8.2 Limitations : -

Besides the above achievements and the successful completion of the project, we still feel

The project has some limitations, listed as below:

1. Limited to five emotions.

2. Camera is needed .

3. System requirements must match to make use of libraries.

8.3 CONCLUSION: -

Emotion detection system is successfully developed in python to detect facial expressions of a user.

CHAPTER 9. REFERENCE

- https://www.youtube.com/watch?v=PulKlAZRoAY&t=500s&ab_channel=PyPowerProjects
- https://pypi.org/project/opencv-python/
- https://pypi.org/project/tensorflow/
- https://keras.io/