EMOTION DETECTION USING MACHINE LEARNING

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

Bachelor of Technology
in
Computer Science & Engineering

By

B.SIVA PRASAD REDDY	(19UECS0104)	**(14477)**
B.ANIRUDH	(19UECS0156)	**(15272)**
B.GOPICHANDU	(19UECS0102)	**(14811)**

Under the guidance of
Dr. K. SEETHALAKSHMI
ASSISTANT PROFESSOR

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SCHOOL OF COMPUTING

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

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

June , 2022

CERTIFICATE

It is certified that the work contained in the project report titled ”EMOTION DETECTION USING MACHINE LEARNING” by ”B.SIVA PRASAD REDDY (19UECS0104), B.ANIRUDH (19UECS0156), B.GOPICHANDU (19UECS0102)” has been carried out under my supervision and that this work has not been submitted elsewhere for a degree.

Signature of Supervisor
Dr.K.Seethalakshmi
Assistant Professor
Computer Science & Engineering
School of Computing
Vel Tech Rangarajan Dr.Sagunthala R&D
Institute of Science & Technology
June,2022

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

DECLARATION

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

(Signature)

(B.SIVA PRASAD REDDY)

Date: / /

(Signature)

(B.ANIRUDH)

Date: / /

(Signature)

(B.GOPICHANDU)

Date: / /

APPROVAL SHEET

This project report entitled EMOTION DETECTION USING MACHINE LEARNING by B.SIVA PRASAD REDDY (19UECS0104),B.ANIRUDH (19UECS0156), B.GOPICHANDU (19UECS0102) is approved for the degree of B.Tech in Computer Science & Engineering.

Examiners **Supervisor**

Dr.K.Seethalakshmi

Date: / /

Place:

ACKNOWLEDGEMENT

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

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

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

We also take this opportunity to express a deep sense of gratitude to our Internal Supervisor **Supervisor name,degree.,(in capital letters)** for his/her cordial support, valuable information and guidance, he/she helped us in completing this project through various stages.

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

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

B.SIVA PRASAD REDDY (19UECS0104)
B.ANIRUDH (19UECS0156)
B.GOPICHANDU (19UECS0102)

ABSTRACT

Emotion recognition based on facial expression is an interesting research field, which has presented and applied in several areas such as Emotion recognition , health and in human machine interfaces. Researchers in this field are interested in developing techniques to interpret, code facial expressions and extract these features in order to have a better prediction by computer. Generally, the technology works best if it uses multiple in context.Emotion Recognition plays a crucial role in the era of Artificial intelligence. The use of this model is to perform different tasks is persistently increasing in society. Facial expression recognition is the part of Facial recognition which is gaining more importance and need for it increases tremendously. In this paper, machine learning is used to recognise human emotions through facial expressions. Here, in order to experiment with and train a deep convolutional network dataset has been used. Facial expressions play a vital role in nonverbal communication which appears due to internal feelings of a person that reflects on the faces. To computer modeling of human emotion, plenty of research has been accomplished by different kinds of researchers. Yet still far behind from the human visual system. This paper has been used the Viola-Jones algorithm to detect the eye and lips region from a face and then with the help of the neural network. Also, Machine Learning techniques, Deep Learning models, and Neural Network algorithms are used for emotion recognition. This paper detected emotion from those features from the positioning of the mouth and eyes.

Keywords: Emotion recognition, Artificial intelligence, Machine Learning,Emotion Detection, Facial Recognition

LIST OF FIGURES

TABLE OF CONTENTS

Chapter 1

INTRODUCTION

1.1 Introduction

Recognizing human expressions and emotions has drawn the attention of researchers, as the capability of recognizing one's expressions helps in human-computer interaction, to right advertising campaigns, and crowning with an augmented and enhanced human communication ,by amending the emotional intelligence of humans. There are many ways to inspect the recognition of human expressions, ranging from facial expressions, body posture, voice tone etc. In this paper we have focused on facial expression recognition

Facial Emotion Recognition is a thriving research area in which lots of advancements like automatic translation systems, machine to human interaction are happening in industries. Human-computer interaction technology refers to a kind of technology which takes computer equipment as the medium, so as to realize the interaction between human and computer. In the recent years, with the rapid development of pattern recognition and artificial intelligence, more and more research has been conducted in the field of human-computer interaction technology.

1.2 Aim of the project

The aim of emotion detection is identifying emotions of a human. The emotion can be captured either from face or from verbal communication. In this work we focus on identifying human emotion from facial expressions. To Propose the development of android applications that can be used for sensing the Emotions of people for their better health. To provide better services and also better Human machine interactions.

1.3 Project Domain

Human emotions are mental states of feelings that come off spontaneously rather than through conscious effort and are accompanied by physiological changes in facial muscles which imply expressions on the face. Non-verbal communication methods such as facial expressions, eye movement, and gestures are used in many applications of human-computer interaction, which among them facial emotion is widely used because it conveys the emotional states and feelings of persons. Emotion recognition is not an easy task because there is no blueprint distinction between the emotions on the face and also there is a lot of complexity and variability. In the machine learning algorithm some important extracted features used for modeling the face, so, it will not achieve a high accuracy rate for recognition of emotion because the features are hand-engineered and depend on prior knowledge.

For facial emotion recognition, the traditiona approaches usually consider a face image that is distinguished from an information picture, and facial segments or milestones are recognized from the face districts. After that different spatial and worldly highlights are separated from these facial segments. At last dependent on the separated highlights a classifier, for example, Keras library, random forest, is trained to produce recognitions results.his work consider the leading challenge faced by machine learning and the entire system is the training part. Where the system has to train by using real data of human face reactions. For example, if the system has to detect an angry face then the first system has to be acquainted with the angry face.

1.4 Scope of the Project

Emotion detection recognition is a method used for detection and recognition of human emotions with the incorporation of technological capabilities, such as facial recognition, speech and voice recognition, biosensing, machine learning, and pattern recognition. The study under consideration explores the global market for both EDR software and services that are capable of recognizing basic emotions that are anger, contempt, disgust, fear, joy, sadness, and surprise.

Chapter 2

LITERATURE REVIEW

[1] T. Cao and M. Li, ”Facial Expression Recognition Algorithm Based on the Combination of CNN and K-Means,” presented at the Proceedings of the 2019 11th International Conference on Machine Learning and Computing, Zhuhai, China, 2019.

[2]Wei L, Tsangouri C, Abtahi F, et al. A recursive framework for expression recognition: from web images to deep models to game dataset[J]. Machine Vision Applications, 2018, 29(3):489-502.

[3]Avigyan Sinha Indian Institute of Technology Kharagpur, India, Aneesh R P Regional Centre IHRD Thiruvananthapuram, India, IJIIE- International Journal of Innovations Implementations in Engineering ISSN 2454- 3489- 2019 December Edition Volume 1.

Emotion is one of the fundamental expression of human beings. It is an integral part of our nonverbal communication. Emotion recognition refers to identifying human emotions typically from facial expressions. It is a skill that comes naturally to us.

An important role is played by human emotion recognition in the interpersonal relationships. Emotion is what separates us from other living beings. Its classification is essential for human computer interaction.

Facial Recognition is the technology that deals with methods and techniques to identify the emotions from the facial expression. Various technological developments in the area of Machine Learning and artificial Intelligence, made the emotion recognition easier.

Chapter 3

PROJECT DESCRIPTION

3.1 Existing System

To detect a face from a given input image or video, Extract facial features such as eyes, nose, and mouth from the detected face Divide facial expressions into different categories such as happiness, anger, sadness, fear, disgust and surprise. Face detection is a special case of object detection. It also involves illumination compensation algorithms and morphological operations to maintain the face of the input image. Further some systems started using detection data to recognize human emotion but they were needed to be worn by the user which limited the scope of activity recognition in open environment in general.

3.2 Proposed System

To improve the recognition rate of the system, further modification in the third phase is done using Artificial Neuro-Fuzzy Inference System (ANFIS). In this method, the static images as well as video input can be given for testing the expressions. Here, neuro-fuzzy based automatic facial expression recognition system to recognize the human facial expressions like happy, fear, sad, angry, disgust and surprise has been proposed. Initially a video showing different expressions is framed into different images. Then the sequence of selected images is stored in a database folder. Using AAM method, the features of all the images are located stored in the form of .ASF files. Then a mean shape is created for all the images in data folder. The change in the AAM shape model according to the change in facial expressions measures the distance or the difference (6)between Neutral and other facial expressions. These values are stored in a .MA T file a specific value is assigned for each individual expression for training the ANFIS. These difference values are then given as input to the ANFIS (Artificial Neuro-Fuzzy Inference System). Using the ANFIS tool available in Mat lab, the system is trained for the different images and their video

input sequences for different expressions

3.3 Feasibility Study

Human Emotion Recognition has been studied intensively in the past decade. Existing approaches typically rely on data from inertial sensors. This paper explores the potential of using point data gathered from wearable depth cameras for on human emotion recognition. We discuss effects different granularity in the depth information and compare their performance to inertial sensor based on the emotion classification.We evaluated our approach participants performing distinct activity classes in environments

3.3.1 Economic Feasibility

Emotion recognition model systems are a large field of research and development, currently with a focus on advanced machine learning algorithms, innovations in the field of hardware architecture, and on decreasing the costs of monitoring while increasing safety. This report concentrates on the applications of activity recognition systems. We categorize such applications into active and assisted, monitoring and surveillance systems for indoor and outdoor activities. Within these categories, the applications are classified according to the methodology used for recognizing human behavior, namely, based on visual, non-visual. We provide an overview of these applications and discuss the advantages and limitations of each approach.
Additionally, we illustrate public data sets that are designed for the evaluation of such recognition systems. The article concludes with a comparison of the existing methodologies which, when applied to real-world scenarios, allow to formulate research questions for future approaches.

3.3.2 Technical Feasibility

Our target increase the accuracy. Focusing on the alogirithm, this define a model that represents conventional neural network used in image recognition and processing that is specifically designed to process pixel data. It have three stage will takes in algorithm and applied to bring out accuraxy of the attributes which we have define

3.3.3 Social Feasibility

Our project will recognize the emotion through facial expression.It will an overview of your workout.And it will provide you the what actions the person is doing.It is very useful for detecting the emotion of the person. By this way we can easily find the emotion of the human And if we use it and make and advance project throught the given data sets where the machine trains through the data sets.Our project is useful for the detecting the person's emotion through image classification

3.4 System Specification

3.4.1 Hardware Specification

- Microsoft Visual Studio
- Intel Core i5
- M560 2.6GHz CPU
- 4G RAM.

3.4.2 Software Specification

- We have used colab and our system windows 10.
- Import the packages.
- Download the data sets.
- Import the dataset by using google drive.

3.4.3 Standards and Policies

Sample attached

Anaconda Prompt

Anaconda prompt is a type of command line interface which explicitly deals with the ML(MachineLearning) modules.And navigator is available in all the Windows,Linux and MacOS.The anaconda prompt has many number of IDE's which make the coding easier. The UI can also be implemented in python.

Standard Used: ISO/IEC 27001

Jupyter

It's like an open source web application that allows us to share and create the documents which contains the live code, equations, visualizations and narrative text. It

can be used for data cleaning and transformation, numerical simulation, statistical modeling, data visualization, machine learning.

Standard Used: ISO/IEC 27001

Chapter 4

METHODOLOGY

4.1 General Architecture

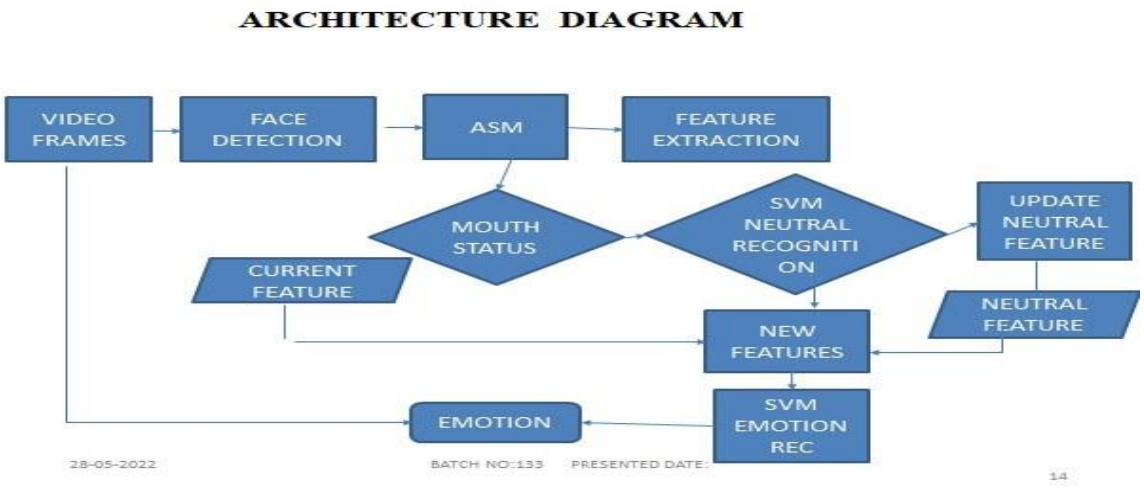


Figure 4.1: **GENERAL ARCHITECTURE**

4.1.1 Description

The data set of the emotion detection has been collected.Then after that the all data is extracted thedata from the dataset. After that the data is sent to the system then after the data set is sent to the ML model or the application of our project after that the dataset is runs and their the result is produced or the result of the operations are sent back to the system from there the result is produced.

4.2 Design Phase

4.2.1 Data Flow Diagram

Figure 4.2: **DATA FLOW**

4.2.2 Description

The data set is collected and that we will extract the data from the data set and then we will categorized which type of emtion is performed by the human and if the emtion of the human is suspicious or identified then the data is sent. And the result is generated and if the emtion is not suspicious or not identified then the again the data application will goes to first then it will again do all the operations from the first onward. If the emtion is identified then it will produce the output or it will display the result.

4.2.3 Use Case Diagram

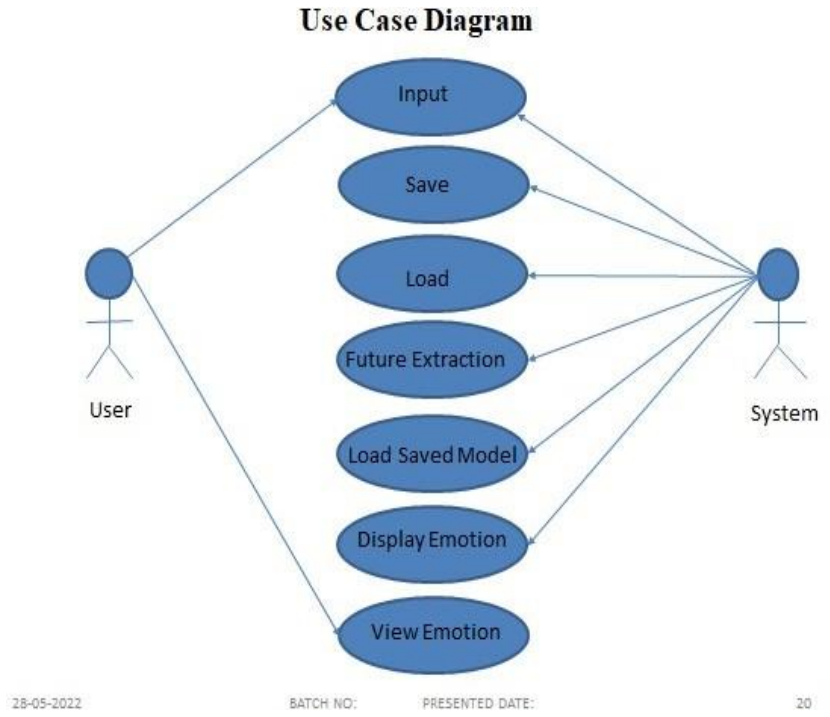


Figure 4.3: **USE CASE DIAGRAM**

4.2.4 Description

Use-case diagrams model the behavior of a system and help to capture the requirements of the system.Use-case diagrams describe the high-level functions and scope of a system. These diagrams also identify the interactions between the system and its actors. The use cases and actors in use-case diagrams describe what the system does and how the actors use it, but not how the system operates internally.Use-case diagrams illustrate and define the context and requirements of either an entire system or the important parts of the system. You can model a complex system with a single use-case diagram, or create many use-case diagrams to model the components of the system. You would typically develop use-case diagrams in the early phases of a project and refer to them throughout the development process.

4.2.5 Class Diagram

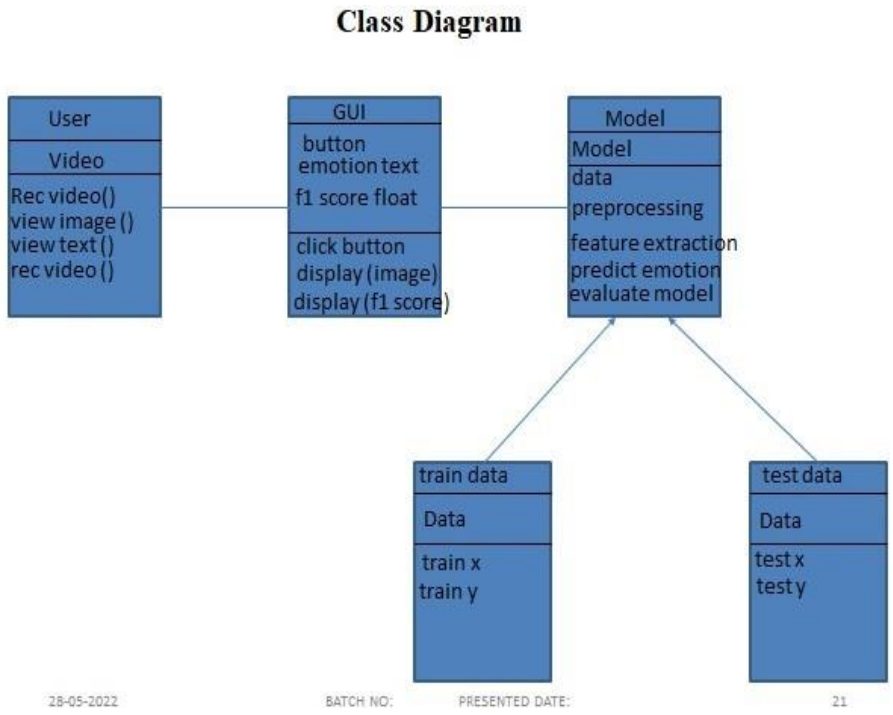


Figure 4.4: **CLASS DIAGRAM**

4.2.6 Description

A class diagram is an illustration of the relationships and source code dependencies among classes in the Unified Modeling Language (UML). In this context, a class defines the methods and variables in an object, which is a specific entity in a program or the unit of code representing that entity. Class diagrams are useful in all forms of object-oriented programming (OOP). The concept is several years old but has been refined as OOP modeling paradigms have evolved.

4.2.7 Sequence Diagram

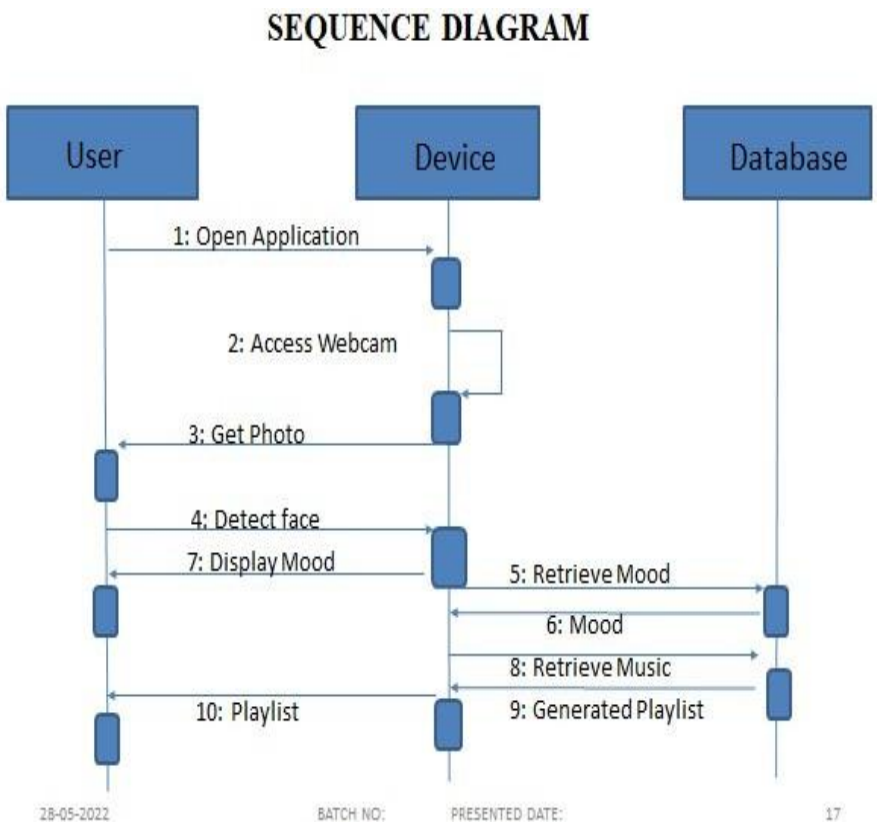


Figure 4.5: **SEQUENCE DIGRAM**

4.2.8 Description

Sequence diagram focuses mainly upon the interaction between objects. It has sequential flow of steps of procedure in which each and every information shared by the two entities can be easily analysed. It can be mainly used for testing of the project especially to know the interactions between the entities

4.2.9 Collaboration diagram

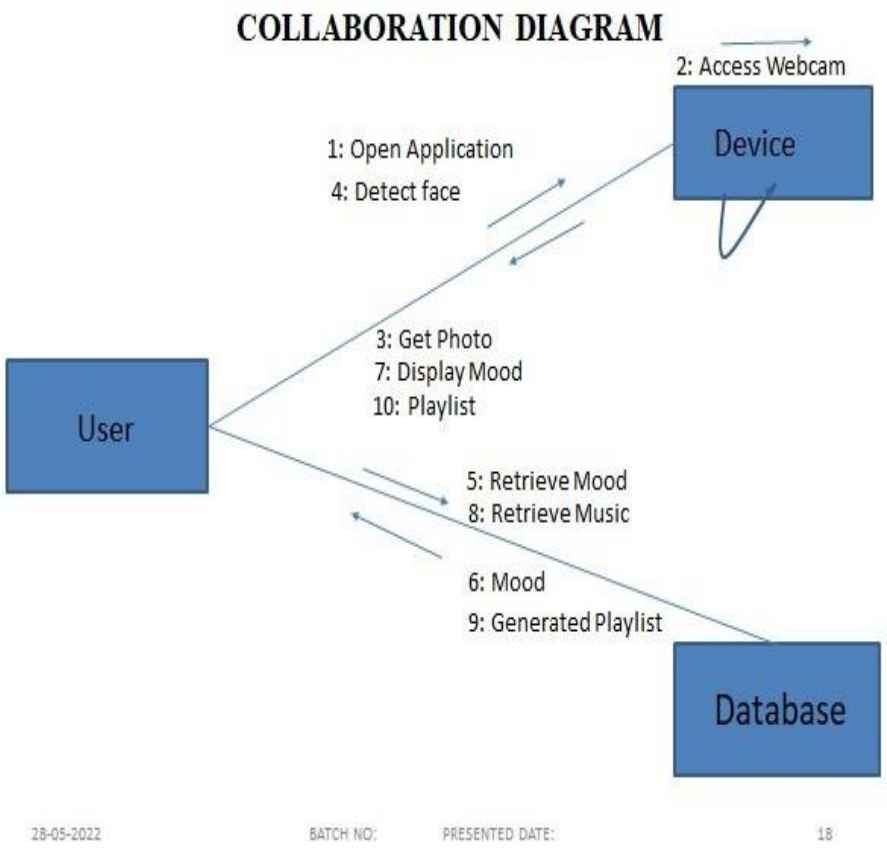


Figure 4.6: **COLLABRATION DIAGRAM**

4.2.10 Description

A collaboration diagram, also known as a communication diagram, is an illustration of the relationships and interactions among software objects in the Unified Modeling Language .The diagrams can be used to portray the dynamic behavior of a particular use case and define the role of each object.

4.2.11 Activity Diagram

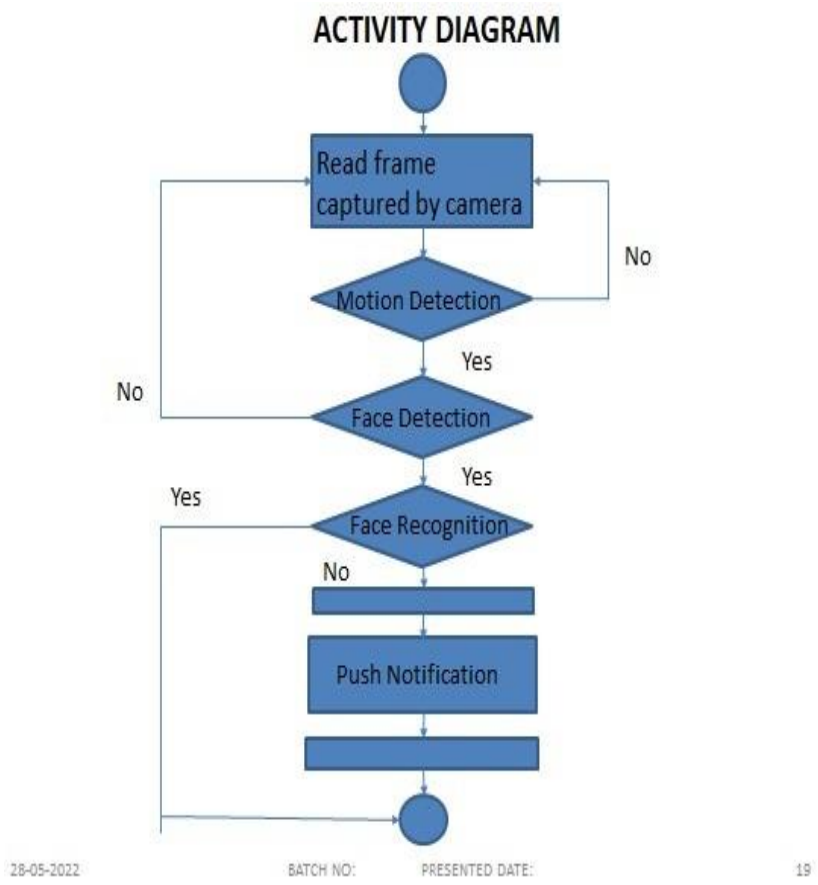


Figure 4.7: **ACTIVITY DIAGRAM**

4.3 Module Description

4.3.1 Module1

Emotion Database: In the data collection steps, this is used both in real-world media and online media to collect as much data as that could. Real-world includes different types of emotional pictures of friends and family members, relatives, some known unknown people's different kinds of facial expressions. They culled data was initially stored for future analysis. From online media, the data is collected data set from kaggle.com. This site uploaded this data set years ago. This site most trusted data set of emotions. This converted the data into pixel grayscale images of faces.

4.3.2 Module2

Emotion Database: In the data collection steps, this is used both in real-world media and online media to collect as much data as that could. Real-world includes different types of emotional pictures of friends and family members, relatives, some known unknown people's different kinds of facial expressions. They culled data was initially stored for future analysis. From online media, the data is collected data set from kaggle.com. This site uploaded this data set years ago. This site most trusted data set of emotions. This converted the data into pixel grayscale images of faces.

4.3.3 Module3

Detection of Emotion: K-means clustering was used with the number of clusters taken as two. Here, the maximum value in all rows is found out and its average is determined. Similarly, the minimum value in all rows is found out and its average is determined. Considering these two points as the base, the pixel values nearer to the maximum average value are grouped into a cluster and the pixel values nearer to the minimum average value are grouped into another cluster. Based on the clustering result, the total number of components in the image is calculated.

4.3.4 Module4

Detection of Emotion: K-means clustering was used with the number of clusters taken as two. Here, the maximum value in all rows is found out and its average is determined. Similarly, the minimum value in all rows is found out and its average is determined. Considering these two points as the base, the pixel values nearer to the maximum average value are grouped into a cluster and the pixel values nearer to the minimum average value are grouped into another cluster. Based on the clustering result, the total number of components in the image is calculated.

Chapter 5

IMPLEMENTATION AND TESTING

5.1 Input and Output

5.1.1 Data Set

```
!git clone https://github.com/misbah4064/facial_expressions.git
```

```
Cloning into 'facial_expressions'...
remote: Enumerating objects: 29, done.
remote: Counting objects: 100% (29/29), done.
remote: Compressing objects: 100% (29/29), done.
remote: Total 14243 (delta 12), reused 0 (delta 0), pack-reused 14214
Receiving objects: 100% (14243/14243), 240.06 MiB | 34.46 MiB/s, done.
Resolving deltas: 100% (235/235), done.
Checking out files: 100% (14004/14004), done.
```

```
[ ] %cd facial_expressions/
    %mkdir -p data_set/{anger,happy,neutral,sad,surprise}
```

```
/content/facial_expressions
```

```
[ ] import cv2
    with open('happy.txt','r') as f:
        img = [line.strip() for line in f]
    for image in img:
        loadedImage = cv2.imread("images/"+image)
        cv2.imwrite("data_set/happy/"+image,loadedImage)
    print("done writing")
```

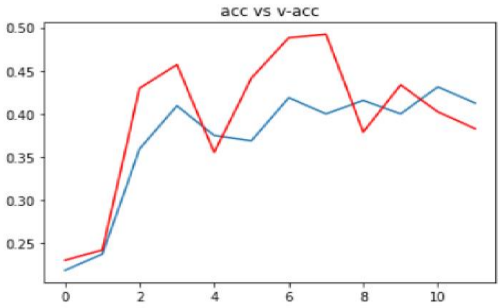
```
done writing
```

```
[ ] %mkdir dataset
```

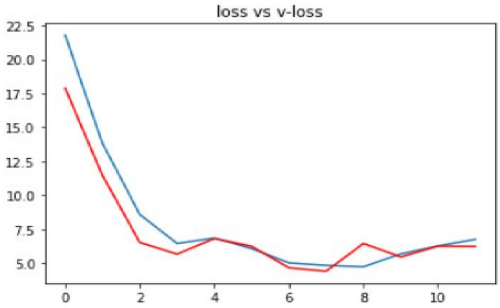
```
import cv2

with open('happy.txt','r') as f:
    images = [line.strip() for line in f]

face_detector = cv2.CascadeClassifier('haarcascade_frontalface_default.xml')

# For each Emotion, enter one numeric face id
face_id = input('\n Enter Emotion id end press <return> ==>  ')

count = 0

for image in images:
    img = cv2.imread("data_set/happy/"+image)
    gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
    faces = face_detector.detectMultiScale(gray, 1.3, 5)

    for (x,y,w,h) in faces:
```

5.1.2 Output Design

In [13]:
```
plt.plot(h['accuracy'])
plt.plot(h['val_accuracy'] , c = "red")
plt.title("acc vs v-acc")
plt.show()
```



In [14]:
```
plt.plot(h['loss'])
plt.plot(h['val_loss'] , c = "red")
plt.title("loss vs v-loss")
plt.show()
```




```
import cv2
import matplotlib.pyplot as plt
%matplotlib inline

image = cv2.imread("dwayne_johnson.jpg")
height, width = image.shape[:2]
resized_image = cv2.resize(image,(3*width, 3*height), interpolation = cv2.INTER_CUBIC)

fig = plt.gcf()
fig.set_size_inches(18, 10)
plt.axis("off")
plt.imshow(cv2.cvtColor(resized_image, cv2.COLOR_BGR2RGB))
plt.show()
```

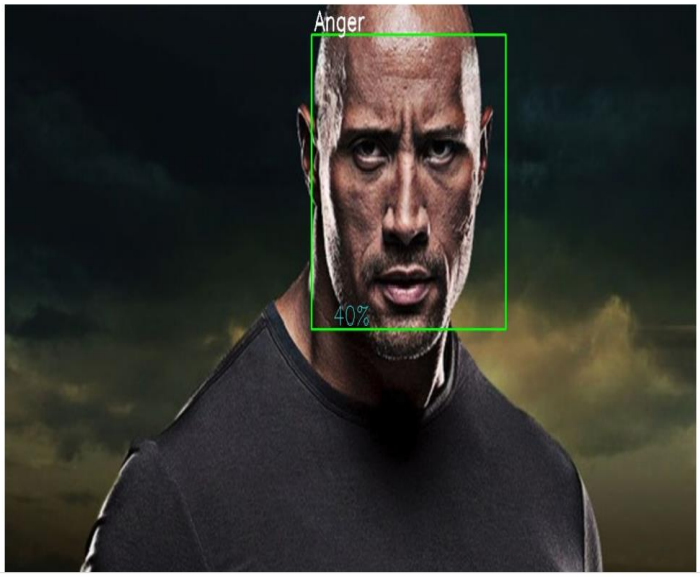


Figure 5.2: **DATA SET**

5.2 Testing

5.3 Types of Testing

5.3.1 Unit testing

Input

```
import cv2
import numpy as np
from PIL import Image
import os

# Path for face image database
path = 'dataset'

recognizer = cv2.face.LBPHFaceRecognizer_create()
detector = cv2.CascadeClassifier("haarcascade_frontalface_default.xml");

# function to get the images and label data
def getImagesAndLabels(path):

    imagePaths = [os.path.join(path,f) for f in os.listdir(path)]
    faceSamples=[]
    ids = []

    for imagePath in imagePaths:

        PIL_img = Image.open(imagePath).convert('L') # convert it to grayscale
        img_numpy = np.array(PIL_img,'uint8')

        id = int(os.path.split(imagePath)[-1].split(".")[1])
        faces = detector.detectMultiScale(img_numpy)

        for (x,y,w,h) in faces:
            faceSamples.append(img_numpy[y:y+h,x:x+w])
            ids.append(id)

    return faceSamples,ids

print ("\n [INFO] Training faces....")
faces,ids = getImagesAndLabels(path)
recognizer.train(faces, np.array(ids))

# Save the model into trainer/trainer.yml
recognizer.write('trainer/trainer.yml')

# Print the numer of Emotions trained and end program
print("\n [INFO] {0} Emotions trained. Exiting Program".format(len(np.unique(ids))))
```

Figure 5.3: **UNIT TESTING**

Test result

```
import cv2
import numpy as np
from PIL import Image
import os

# Path for face image database
path = 'dataset'

recognizer = cv2.face.LBPHFaceRecognizer_create()
detector = cv2.CascadeClassifier("haarcascade_frontalface_default.xml");

# function to get the images and label data
def getImagesAndLabels(path):

    imagePaths = [os.path.join(path,f) for f in os.listdir(path)]
    faceSamples=[]
    ids = []

    for imagePath in imagePaths:

        PIL_img = Image.open(imagePath).convert('L') # convert it to grayscale
        img_numpy = np.array(PIL_img,'uint8')

        id = int(os.path.split(imagePath)[-1].split(".")[1])
        faces = detector.detectMultiScale(img_numpy)

        for (x,y,w,h) in faces:
            faceSamples.append(img_numpy[y:y+h,x:x+w])
            ids.append(id)

    return faceSamples,ids

print ("\n [INFO] Training faces....")
faces,ids = getImagesAndLabels(path)
recognizer.train(faces, np.array(ids))

# Save the model into trainer/trainer.yml
recognizer.write('trainer/trainer.yml')

# Print the numer of Emotions trained and end program
print("\n [INFO] {0} Emotions trained. Exiting Program".format(len(np.unique(ids))))
```

```
[INFO] Training faces....

[INFO] 2 faces trained. Exiting Program
```

Figure 5.4: **UNIT TESTING**

5.3.2 Integration testing

Input

```
base_model = MobileNet( input_shape=(224,224,3), include_top= False )

for layer in base_model.layers:
  layer.trainable = False



x = Flatten()(base_model.output)
x = Dense(units=7 , activation='softmax' )(x)

# creating our model.
model = Model(base_model.input, x)
```

Figure 5.5: **INTEGRATION TESTING**

Test result

```
# Working with pre trained model

base_model = MobileNet( input_shape=(224,224,3), include_top= False )

for layer in base_model.layers:
  layer.trainable = False


x = Flatten()(base_model.output)
x = Dense(units=7 , activation='softmax' )(x)

# creating our model.
model = Model(base_model.input, x)
```

```
Downloading data from https://storage.googleapis.com/tensorflow/keras-applications/mobilenet/mobilenet_1_0_224_tf_no_top.h5
17227776/17225924 [==============================] - 0s 0us/step
17235968/17225924 [==============================] - 0s 0us/step
```

Figure 5.6: **UNIT TESTING**

5.3.3 System testing

Input

```
t_img , label = train_data.next()


def plotImages(img_arr, label):
  """
  input  :- images array
  output :- plots the images
  """
  count = 0
  for im, l in zip(img_arr,label) :
    plt.imshow(im)
    plt.title(im.shape)
    plt.axis = False
    plt.show()

    count += 1
    if count == 10:
      break
plotImages(t_img, label)
```

Figure 5.7: **SYSTEM TESTING**

Test Result

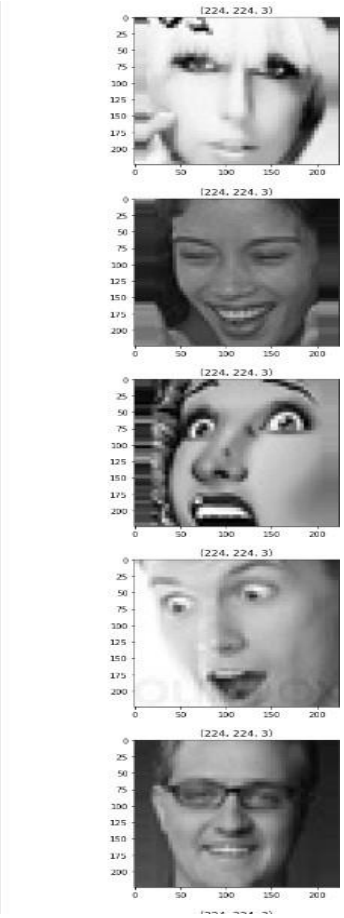


Figure 5.8: **SYSTEM TESTING**

5.3.4 Test Result

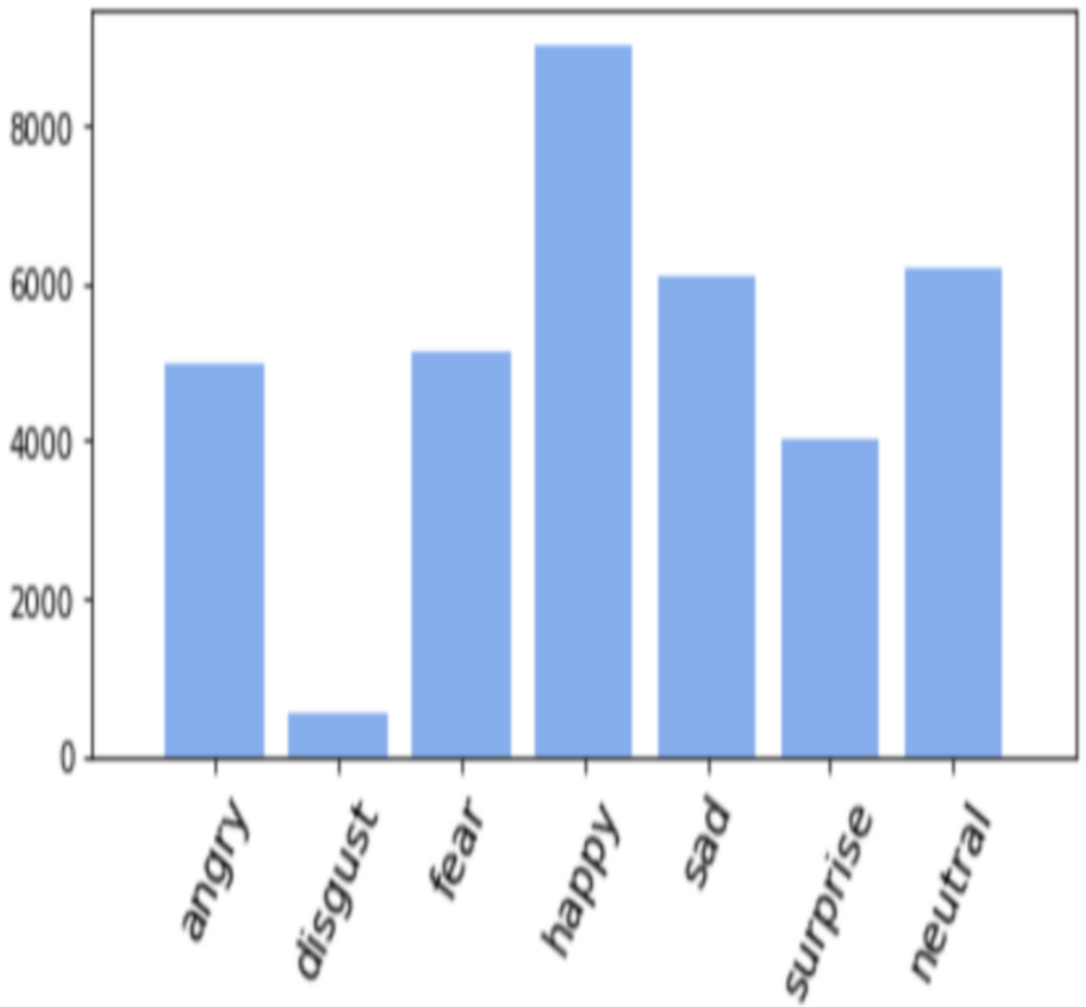


Figure 5.9: **Test Image**

Chapter 6

RESULTS AND DISCUSSIONS

6.1 Efficiency of the Proposed System

Sample attached

We applied support vector machines to our dataset and predicted the results. The results were interpreted using confusion matrix and accuracy metric. The train:test split was 75:25. We also did cross-validation on the dataset to remove any biases. Value of split was chosen as 4 because the resultant splits will have same number of images as our 25In our experiment SVM with linear kernel performed better than other kernels. Rbf gave us the worst performance, whereas poly was as good as linear kernel. We tried to keep the test set cross-validation score was also approximately equal to the accuracy score achieved by the split. Figure 17 shows the heat-map of the confusion matrix from our multi-class classification results. On further analysis of the confusion matrix, we see that the diagonals have higher weights; there are a few misclassifications in every class except class 4: Fear.Our focus is on the cross-validation score as it takes entire database into consideration for training, thus removing any biases. After removing jaw-line features the cross-validation results also increased from 784. Further we analyzed the predicted Vs actual results report for this scenario as in Table 9. From Table 9 we see that the detection rate for anger, disgust and surprise has increased, whereas contempt and fear has been misclassified. Detection rate of sadness has been same as the above experiment. A logical assumption will be that the sadness emotion of the subjects, closely resemble that of anger. To improve the results, we did some fine tuning. We changed the data split to 70:30 and cross validation folds to 5. We use stratified sampling for the split. We also extracted more features from the available 68 facial point landmarks.

6.2 Sample Code

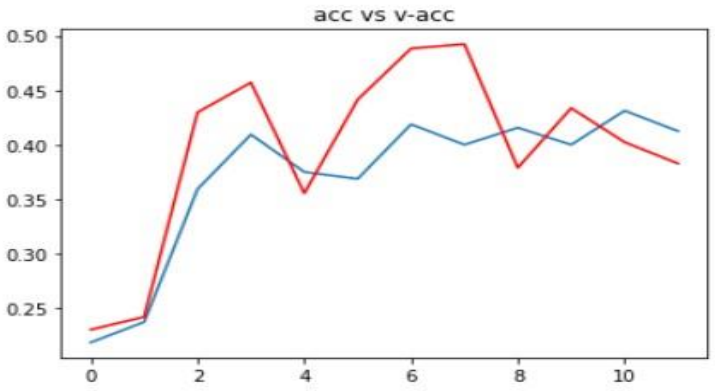
```
!git clone https://github.com/3Anirudh/Emotion-Detection1.git
!unzip dataset.zip?dl=0
```

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

from keras.layers import Flatten, Dense
from keras.models import Model
from keras.preprocessing.image import ImageDataGenerator , img_to_array, load_img
from keras.applications.mobilenet import MobileNet, preprocess_input
from keras.losses import categorical_crossentropy
base_model = MobileNet( input_shape=(224,224,3), include_top= False )

for layer in base_model.layers:
  layer.trainable = False


x = Flatten()(base_model.output)
x = Dense(units=7 , activation='softmax' )(x)


model = Model(base_model.input, x)
train_datagen = ImageDataGenerator(
      zoom_range = 0.2,
      shear_range = 0.2,
      horizontal_flip=True,
      rescale = 1./255
)

train_data = train_datagen.flow_from_directory(directory= "/content/train",
                                               target_size=(224,224),
                                               batch_size=32,
                                    )


train_data.class_indices
val_datagen = ImageDataGenerator(rescale = 1./255 )

val_data = val_datagen.flow_from_directory(directory= "/content/test",
                                           target_size=(224,224),
                                           batch_size=32,
                                    )
t_img , label = train_data.next()


def plotImages(img_arr, label):

  count = 0
  for im, l in zip(img_arr,label) :
    plt.imshow(im)
    plt.title(im.shape)
    plt.axis = False
```

```
    plt.show()

    count += 1
    if count == 10:
      break


plotImages(t_img, label)
from keras.models import load_model
model = load_model("/content/best_model.h5")
h =  hist.history
h.keys()
plt.plot(h['accuracy'])
plt.plot(h['val_accuracy'] , c = "red")
plt.title("acc vs v-acc")
plt.show()
plt.plot(h['loss'])
plt.plot(h['val_loss'] , c = "red")
plt.title("loss vs v-loss")
plt.show()
op = dict(zip( train_data.class_indices.values(), train_data.class_indices.keys()))
path = "/content/test/angry/PrivateTest_1054527.jpg"
img = load_img(path, target_size=(224,224) )

i = img_to_array(img)/255
input_arr = np.array([i])
input_arr.shape

pred = np.argmax(model.predict(input_arr))

print(f" the image is of {op[pred]}")


plt.imshow(input_arr[0])
plt.title("input image")
plt.show()
```

Output

In [13]:
```
plt.plot(h['accuracy'])
plt.plot(h['val_accuracy'] , c = "red")
plt.title("acc vs v-acc")
plt.show()
```



In [14]:
```
plt.plot(h['loss'])
plt.plot(h['val_loss'] , c = "red")
plt.title("loss vs v-loss")
plt.show()
```

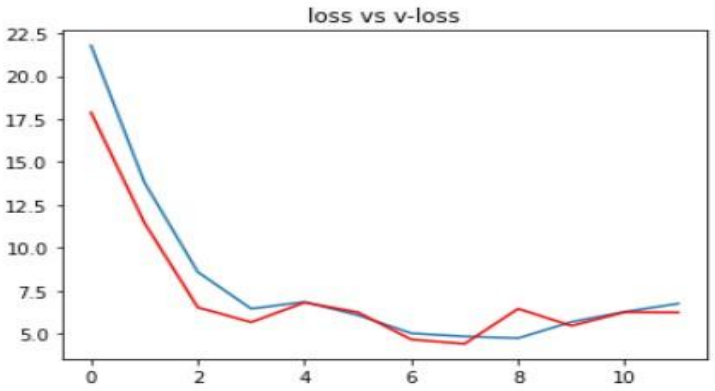


Figure 6.1: **Output 1**

```
import cv2
import matplotlib.pyplot as plt
%matplotlib inline

image = cv2.imread("dwayne_johnson.jpg")
height, width = image.shape[:2]
resized_image = cv2.resize(image,(3*width, 3*height), interpolation = cv2.INTER_CUBIC)

fig = plt.gcf()
fig.set_size_inches(18, 10)
plt.axis("off")
plt.imshow(cv2.cvtColor(resized_image, cv2.COLOR_BGR2RGB))
plt.show()
```

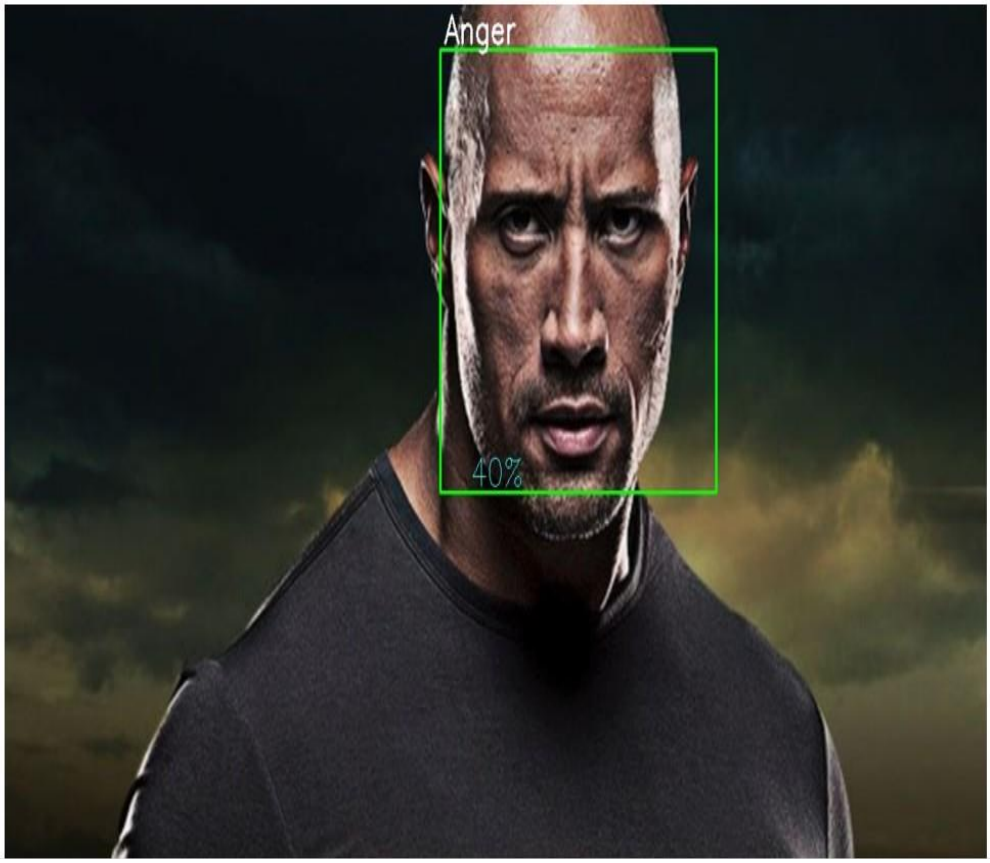


Figure 6.2: **Output 2**

Chapter 7

CONCLUSION AND FUTURE ENHANCEMENTS

7.1 Conclusion

The goal of this work has been to design a deep neural network for facial expression recognition. We have seen how to implement a Convolutional Neural Network capable of predicting emotion and facial expressions. The proposed method directly inputs the image pixel value through training sample image data. Autonomous learning can implicitly acquire more abstract feature expression of the image. Numerous researches and studies about Emotion Recognition, Deep learning techniques used for recognizing the emotions are conducted. It is required in future to have a model like this with much more reliable, which has limitless possibilities in all fields.
This project tried to use inception net for solving emotion recognition problem. The effect of unimportant pixels which is outside facial expressions was reduced. In addition, single-depth placement of the pixels in the pictures to networks did not only result in loss of success rate, but also reduced training time and number of networks.

7.2 Future Enhancements

facial expression recognition, using different techniques for face detection, feature extraction, analysis and classification methods. The paper gives detailed information about existing techniques in all the stages of Facial Expression Recognition FERs. The paper is very useful to both old and upcoming researchers in the field of FER, it presents detail information about existing techniques in all stages of that field to reinforcement their understanding of current trends and assist their future research prospects and directions. Further, the paper discussed about various techniques of their technology, merits and demerits which improves the performance of Facial Expression Recognition in image processing.

Chapter 8

PLAGIARISM REPORT

Report Title:	report
Report Link: (Use this link to send report to anyone)	https://www.check-plagiarism.com/plag-report/5159599a95915b9e45c064925d2781779d7541653718769
Report Generated Date:	28 May, 2022

Total Words:	1696
Total Characters:	12350
Keywords/Total Words Ratio:	0%
Excluded URL:	No
Unique:	88%
Matched:	12%

Figure 8.1: **PLAGIARISM REPORT**

Chapter 9

SOURCE CODE & POSTER PRESENTATION

9.1 Source Code

```
!git clone https://github.com/3Anirudh/Emotion-Detection1.git
!unzip dataset.zip?dl=0
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

from keras.layers import Flatten, Dense
from keras.models import Model
from keras.preprocessing.image import ImageDataGenerator, img_to_array, load_img
from keras.applications.mobilenet import MobileNet, preprocess_input
from keras.losses import categorical_crossentropy
base_model = MobileNet( input_shape=(224,224,3), include_top= False )

for layer in base_model.layers:
  layer.trainable=False


x = Flatten()(base_model.output)
x = Dense(units=7 , activation='softmax' )(x)


model = Model(base_model.input, x)
train_datagen = ImageDataGenerator(
      zoom_range = 0.2,
      shear_range = 0.2,
      horizontal_flip=True,
      rescale = 1./255
)

train_data = train_datagen.flow_from_directory(directory= "/content/train",
                                               target_size=(224,224),
                                               batch_size=32,
                                    )


```

```
train_data.class_indices
val_datagen = ImageDataGenerator(rescale = 1./255 )

val_data = val_datagen.flow_from_directory(directory= "/content/test",
                                           target_size=(224,224),
                                           batch_size=32,
                                    )
t_img , label = train_data.next()


def plotImages(img_arr, label):

  count = 0
  for im, l in zip(img_arr,label) :
    plt.imshow(im)
    plt.title(im.shape)
    plt.axis = False
    plt.show()

    count += 1
    if count == 10:
      break


plotImages(t_img, label)
from keras.models import load_model
model = load_model("/content/best_model.h5")
h =  hist.history
h.keys()
plt.plot(h['accuracy'])
plt.plot(h['val_accuracy'] , c = "red")
plt.title("acc vs v-acc")
plt.show()
plt.plot(h['loss'])
plt.plot(h['val_loss'] , c = "red")
plt.title("loss vs v-loss")
plt.show()
op = dict(zip( train_data.class_indices.values(), train_data.class_indices.keys()))
path = "/content/test/angry/PrivateTest_1054527.jpg"
img = load_img(path, target_size=(224,224) )

i = img_to_array(img)/255
input_arr = np.array([i])
input_arr.shape

pred = np.argmax(model.predict(input_arr))

print(f" the image is of {op[pred]}")


```

```
86 plt.imshow(input_arr[0])
87 plt.title("input image")
88 plt.show()
```

9.2 Poster Presentation

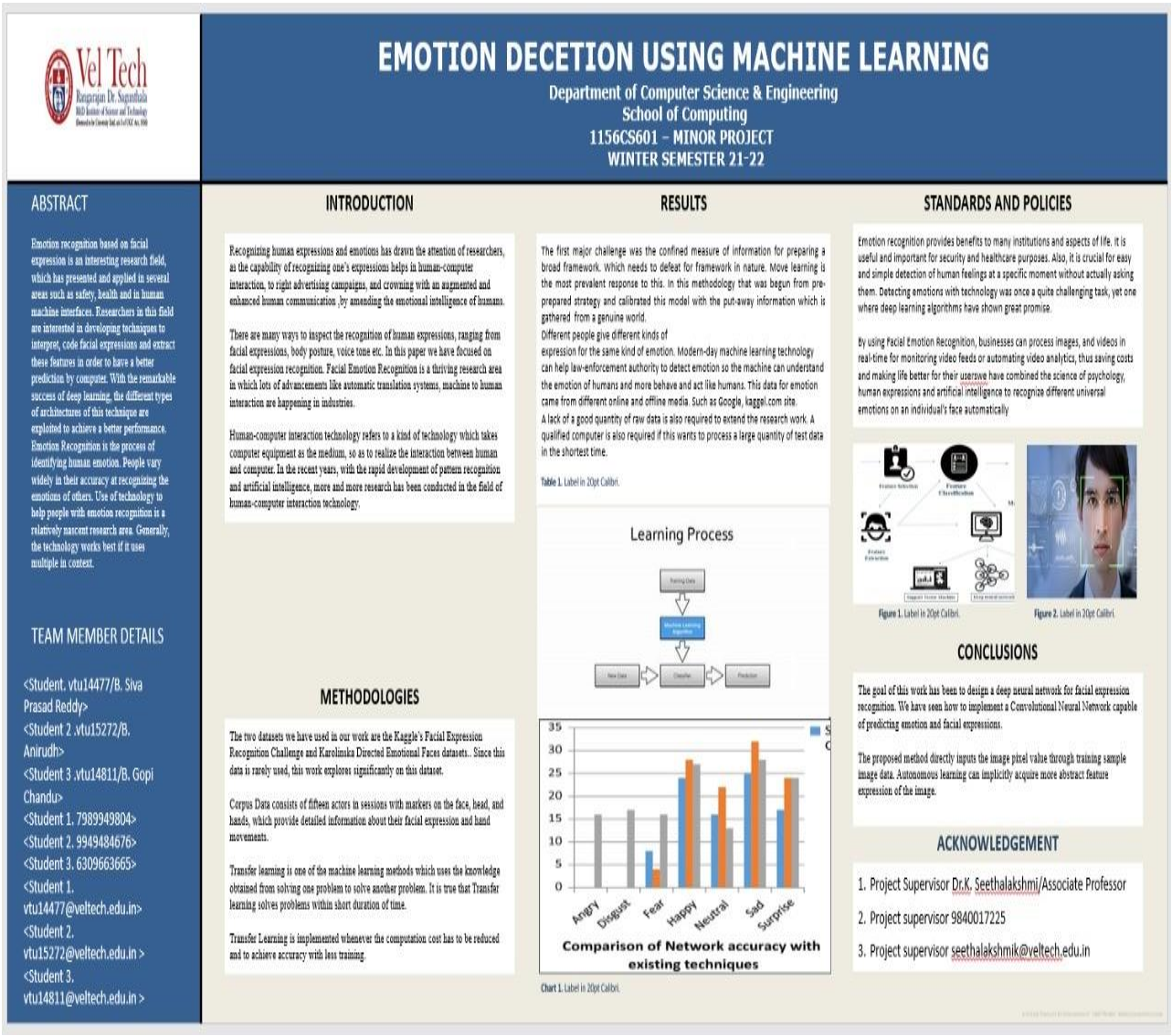


Figure 9.1: **POSTER PRESENTATION**

References

[1] Magudeeswaran V, Singh J F. Contrast limited fuzzy adaptive histogram equalization for enhancement of brain images[J]. International Journal of Imaging Systems Technology, 2017, 27(1):98-103 Issue, Pageno.Year

[2] A. M. Ousmane, T. Djara, F. J. Zoumarou W. and A. Vianou, Automatic recognition system of emotions expressed through the face using machine learning: Application to police interrogation simulation," 2019 3rd International Conference on Bio-engineering for Smart Technologies (BioSMART), Paris, France, 2019, pp. 1-4.

[3] Wei L, Tsangouri C, Abtahi F, et al. A recursive framework for expression recognition: from web images to deep models to game dataset[J]. Machine Vision Applications, 2018, 29(3):489-502

[4] Moeini A, Faez K, Moeini H, et al. Facial expression recognition using dual dictionary learning [J]. Journal of Visual Communication Image Representation, 2017, 45(C): 20-33.

[5]] A. Routrey , M. Swaen and P Kabisetpathy, "Database, features and classifiers for emotion recognition: a review," Int. J. Speech Technol., vol. 0, no. 0, p. 0, 2018.

[6]] Avigyan Sinha Indian Institute of Technology Kharagpur, India, Aneesh R P Regional Centre IHRD Thiruvananthapuram, India, IJIIE- International Journal of Innovations Implementations in Engineering ISSN 2454- 3489- 2019 December Edition Volume 1.

[7]] T. Cao and M. Li, "Facial Expression Recognition Algorithm Based on the Combination of CNN and K-Means," presented at the Proceedings of the 2019 11th International Conference on Machine Learning and Computing, Zhuhai, China, 2019.