FACE EMOTION RECOGNITION

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

Bachelor of Technology
in
Computer Science & Engineering

By

V.JASWANTH	20UECS0984	**VTU18163**
R.JAYANTH	20UECS0773	**VTU 18164**
P.SUCHARAN	20UECS0732	**VTU18143**

Under the guidance of
Mrs.N.Thamizharasi,M.E.,
ASSISTANT PROFESSOR

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
SCHOOL OF COMPUTING

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

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

May, 2023

FACE EMOTION RECOGNITION

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

Bachelor of Technology
in
Computer Science & Engineering

By

V.JASWANTH	20UECS0984	**VTU18163**
R.JAYANTH	20UECS0773	**VTU 18164**
P.SUCHARAN	20UECS0732	**VTU18143**

Under the guidance of
Mrs.N.Thamizharasi,M.E.,
ASSISTANT PROFESSOR

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
SCHOOL OF COMPUTING

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

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

May, 2023

CERTIFICATE

It is certified that the work contained in the project report titled "FACE EMOTION RECOGNITION" by V.JASWANTH (20UECS0984), R.JAYANTH (20UECS0773), P.SUCHARAN (20UECS0732) has been carried out under my supervision and that this work has not been submitted elsewhere for a degree.

Signature of Supervisor
Mrs.N.Thamizharasi
Assitant Professor
Computer Science & Engineering
School of Computing
Vel Tech Rangarajan Dr. Sagunthala R&D
Institute of Science & Technology
May, 2023

Signature of Head of the Department
Dr.M.S.Murali Dhar
Head
Computer Science & Engineering
School of Computing
Vel Tech Rangarajan Dr. Sagunthala R&D
Institute of Science & Technology
May, 2023

Signature of the Dean
Dr. V. Srinivasa Rao
Professor & Dean
Computer Science & Engineering
School of Computing
Vel Tech Rangarajan Dr. Sagunthala R&D
Institute of Science & Technology
May, 2023

DECLARATION

We declare that this written submission represents our 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.

V.JASWANTH

Date: / /

R.JAYANTH

Date: / /

P.SUCHARAN

Date: / /

APPROVAL SHEET

This project report entitled FACE EMOTION RECOGNITION by V.JASWANTH (20UECS0984), R.JAYANTH (20UECS0773), P.SUCHARAN (20UECS0732) is approved for the degree of B.Tech in Computer Science & Engineering.

Examiners **Supervisor**

Mrs.N.Thamizharasi, M.E.,

Date: / /

Place:Avadi

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 **Professor & Dean, Department of Computer Science & Engineering, School of Computing, Dr. V. SRINIVASA RAO, M.Tech., Ph.D.,** for immense care and encouragement towards us throughout the course of this project.

We are thankful to our **Head, Department of Computer Science & Engineering, Dr. M.S. MURALI DHAR, M.E., Ph.D.,** for providing immense support in all our endeavors.

We also take this opportunity to express a deep sense of gratitude to our Internal Supervisor **Mrs. N.THAMIZHARASI,M.E.,** for her cordial support, valuable information and guidance,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.,** 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.

V.JASWANTH 20UECS0984
R.JAYANTH 20UECS0773
P.SUCHARAN 20UECS0732

ABSTRACT

The face emotion recognition project based on Python aims to develop an automated system that can detect emotions in human faces through image processing techniques.The project involves building a Machine Learning model that can recognize and classify different facial expressions such as happiness, sadness, anger, fear, surprise, and disgust.The project uses Python programming language and various open-source libraries such as OpenCV, Deep face, and Keras to develop the image processing and Machine Learning components.The face emotion recognition project aims to develop a system that can automatically detect and classify human facial expressions.The system utilizes deep learning algorithms and computer vision techniques to analyze the facial features and extract meaningful information that corresponds to different emotions.The system can be applied in various fields, including human-computer interaction, psychology, and security systems. This project aims to improve the accuracy and efficiency of the emotion recognition system, providing a reliable tool for emotion analysis and recognition.

Keywords:Face emotion recognition, Image processing, Facial expressions, Open CV, Deep face, Keras.

LIST OF FIGURES

LIST OF ACRONYMS AND ABBREVIATIONS

CNN	Convolutional Neural Network
CPU	Central Processing Unit
CV	Computer Vision
FER	Facial Emotion Recognition
FR	Face Recognition
FRA	Face Representation Augmentation
GPU	Graphics Processing Unit
IDE	Integrated Development Envinorment
RAM	Random Access Memory
XML	Extensible Markup Language

TABLE OF CONTENTS

Chapter 1

INTRODUCTION

1.1 Introduction

Facial Emotion Recognition (FER) is the process of detecting and interpreting human emotions based on facial expressions. It involves using computer algorithms to analyze facial features such as the position of the eyebrows, mouth, and eyes to determine emotions like happiness, sadness, anger, fear, and surprise. FER has numerous applications in various fields such as psychology, marketing, security, and entertainment. For example, it can be used to monitor the emotional state of patients with mental health disorders, evaluate consumer reactions to advertisements, detect suspicious behavior in public places, and enhance the realism of video games and virtual reality simulations. FER has gained popularity in recent years due to advancements in Artificial Intelligence and computer vision technologies, which have made it possible to accurately recognize emotions in real-time with high precision.

1.2 Aim of the Project

The aim of a face emotion recognition project is to develop an Artificial Intelligence system that can accurately identify the emotions displayed on a person's face. This technology is often used in fields such as psychology, marketing, and social robotics, where the ability to accurately detect emotions can be beneficial for understanding human behavior, improving customer satisfaction, and creating more engaging human-machine interactions.

1.3 Project Domain

The domain for face emotion recognition project is computer vision, which involves analyzing digital images or videos to automatically extract information and make decisions. Specifically, the project focuses on detecting and classifying emotions from facial expressions captured by a camera.The technology has numerous

applications, including improving human-computer interaction, developing more responsive virtual assistants, and enabling better security systems. For example, the project can be used in video surveillance to detect suspicious behavior or emotions, such as anger or fear, which can help prevent crimes or mitigate their impact.

Another application is in mental health, where the technology can be used to monitor a patient's emotional state and provide personalized interventions. In education, it can be used to monitor students' engagement and help teachers adapt their teaching strategies.In summary, the domain of face emotion recognition is a subfield of computer vision with a wide range of applications in various industries and fields, including security, healthcare, education, and entertainment.

1.4 Scope of the Project

The scope for a face emotion recognition project is significant and wide-ranging. With the increasing prevalence of facial recognition technology in various fields such as security, marketing, and entertainment, the ability to accurately recognize and interpret facial expressions has become crucial.In terms of security, face emotion recognition technology can be used to detect and prevent potential threats by analyzing the emotions of individuals in a public space. In marketing, it can help companies better understand their customers by analyzing their facial expressions during product testing and advertising campaigns.

Furthermore, face emotion recognition technology can be integrated into virtual reality and gaming applications, enhancing the user experience and enabling more immersive interactions. It can also be used in healthcare to diagnose and treat mental health conditions by analyzing patients' facial expressions during therapy sessions.Overall, the potential applications for face emotion recognition technology are vast, and its use is only expected to increase in the future, making it a promising area for research and development.

Chapter 2

LITERATURE REVIEW

Roman Shvetsov et al.,[1] proposed Emotion Recognition Challenge. An ensemble of several models, which capture spatial and audio features from videos. Spatial features are captured by convolutional neural networks, pretrained on large face recognition datasets.Show that usage of strong industry-level face recognition networks increases the accuracy of emotion recognition. Using ensemble improve on the previous year's best result on the test set by about 1percent,achieving a 60.03percent classification accuracy without any use of visual temporal information, showing a top-2 result in this challenge.

Suci Dwijayanti et al.,[2] proposed that Robots can mimic humans, including recognizing faces and emotions. However, relevant studies have not been implemented in real-time humanoid robot systems. In addition, face and emotion recognition have been considered separate problems. This study proposes a combination of face and emotion recognition for real-time application in a humanoid robot. Specifically, face and emotion recognition systems are developed simultaneously using convolutional neural network architectures. The model is compared to well-known architectures, such as AlexNet and VGG16, to determine which is better for implementation in humanoid robots. Data used for face recognition are primary data taken from 30 electrical engineering students after preprocessing, resulting in 18,900 data points. Emotion data of surprise, anger, neutral, smile, and sad are taken from the same respondents and combined with secondary data for a total of 5,000 data points for training and testing. The test is carried out in real time on a humanoid robot using the two architectures. The face and emotion recognition accuracy is 85per and 64per, respectively, using the AlexNet model.

Mostafa Shahabinejad et al.,[3] proposed a novel face recognition based attention FER (FRA-FER) framework which propagates subtle face recognition (FR) features through the FER network. Particularly, first a spatial attention map from the feature maps of an FR convolutional neural network (CNN) is created and then it is fused

into the FER-CNN. By doing this FR feature propagation, the FER network is personalized as it takes the advantage of the FR features learned from large-scale face recognition datasets. Experiments on the two challenging datasets AffectNet and a few demonstrate the superiority of our proposed FRA-FER network to the state-of-the-art work.

Viha Upadhyay et al.,[4] proposed the human-computer interaction paradigm. In order to recognize emotions from images, the system needs to extract facial features like mouth, eyes, etc. Emotions can also be extracted from frontal and non-frontal images. Traditional methods used for extracting facial features are geometry based method, template based method, and appearance based method. The main focus of this paper is to review different types of facial feature extraction method and this research article illustrates a comparison between different methods and in the end, it describes some future research works that would be helpful to make FER more reliable and efficient.

TaimurShahzad et al.,[5] proposed Facial Expression is an unspoken message essential to collaboration and effective discourse. An inner emotional state of a human is expressed using facial expressions and is very effective for communication with actual emotions. Anger, happiness, sadness, contempt, surprise, fear, disgust, and neutral are eight common expressions of humans. Scientific community proposed several face emotion recognition techniques. However, due to fewer face landmarks and their intensity for deep learning models, performance improvement for facial expression recognition still needs to be improved for accurately predicting facial emotion recognition.

Khan et al.,[6] proposed Emotion recognition has a key role in affective computing. Recently, fine-grained emotion analysis, such as compound facial expression of emotions, has attracted high interest of researchers working on affective computing. A compound facial emotion includes dominant and complementary emotions (e.g., happily-disgusted and sadly-fearful), which is more detailed than the seven classical facial emotions (e.g., happy, disgust, and so on). Current studies on compound emotions are limited to use data sets with limited number of categories and unbalanced data distributions, with labels obtained automatically by machine learning-based algorithms which could lead to inaccuracies. To address these problems, we released

the iCV-MEFED data set, which includes 50 classes of compound emotions and labels assessed by psychologists. The task is challenging due to high similarities of compound facial emotions from different categories.

Sharma et al.,[7] proposed features used in traditional emotion recognition methods are mainly external features such as facial expressions, body postures and speech. There is unnecessary to wear sensors for obtaining these signals, which has the advantages of easy acquisition and low cost.Utilized the people facial expressions in the videos for emotion recognition. The authors applied Naive Bayes algorithm to recognize seven different emotions including happiness, surprise, anger, disgust, fear, sadness, and neutrality. The emotion recognition accuracy rate between facial expressions of different people is 64.3per, while testing the same person achieves an accuracy rate of 93.2per, indicating that facial expressions can be adapted to effectively recognize emotions.Combined acoustic features and speech content for emotion recognition based on speech signals. A support vector machine-belief network architecture is used to subdivide six different emotions of anger, disgust, fear, neutrality, sadness and surprise, and the recognition accuracy is up to 93percent.

Singh et al.,[8] proposed to avoid the complex process of explicit feature extraction in traditional facial expression recognition, a face expression recognition method based on a convolutional neural network (CNN) and an image edge detection is proposed. Firstly, the facial expression image is normalized, and the edge of each layer of the image is extracted in the convolution process. The extracted edge information is superimposed on each feature image to preserve the edge structure information of the texture image. Then, the dimensionality reduction of the extracted implicit features is processed by the maximum pooling method. Finally, the expression of the test sample image is classified and recognized by using a Softmax classifier.

Cohn et al.,[9] proposed Traditional Facial Expression Recognition Methods.Both images and videos can capture facial expressions. The static facial expression recognition methods focus on extracting features from the images and attempting to classify the input images to six basic emotions: happiness, surprise, disgust, anger, sadness and fear. Some texture processing techniques, for example, scale invariant feature transform (SIFT) [19], local binary pattern (LBP) [20], the pyramid of histograms of oriented gradients (PHOG) and local phase quantization (LPQ) [21]

have been utilized to extract the facial textures as appearance-based features. Most of these types of FER systems have achieved high accuracy on both lab-controlled and unconstrained datasets.

Yang et al.,[10] proposed a novel about Capturing Video:An embedded camera of the smart phone captures the video of the user. As most of the time, the user faces towards the screen of the phone, the video mainly captures the face, the head, and some body parts of the user.Selecting Representative Frames:As there are many frames in the video sequence, we need to select some representative frames from the sequence to reduce the burden of the processing.To select frames, first, all the frames are converted to the gray scale.Then, histograms are obtained from each frame. A chi-square distance is calculated between the histograms of two successive frames. We select a frame, when the distances between the histogram of this frame and that of the previous frame, and the next frame are minimal. In this way, we select a frame, which is stable in nature.Face Detection:Once we select the frames, the face areas in the frames are detected by the Viola-Jones algorithm. This algorithm works fast, and is suitable for a real-time implementation. Now a day, many smart phones have the face detection functionality embedded into the mobile system.

Chapter 3

PROJECT DESCRIPTION

3.1 Existing System

- Dataset preparation: A large dataset of facial images with labeled emotions is required to train the model.This dataset should contain images of people with different expressions of emotion such as happiness, sadness, anger, surprise, fear,neutral and disgust.
- Data pre-processing: The images in the dataset are pre-processed to ensure uniformity and clarity. This may include resizing, normalization, and cropping of images.
- Model training: A Convolutional Neural Network (CNN) is trained on the pre-processed images.The CNN consists of multiple layers that extract features from the images and classify them into different categories based on the emotions.

Disadvantages

- Computational Complexity: An exhaustive system would need to analyze every possible combination of facial features to recognize emotions accurately. This would require an enormous amount of computational power and time, making it impractical for real-time applications.
- Difficulty in recognizing complex emotions: Human emotions are complex and can vary in intensity and duration. An exhaustive system may not be able to recognize these variations and nuances in emotions accurately.

3.2 Proposed System

- Data collection: A large dataset of facial images with labeled emotions is collected. The dataset should contain images of people with different expressions of emotion such as happiness, sadness, anger, surprise, fear, and disgust.

- Data pre-processing: The images in the dataset which are in data collection are pre-processed to ensure uniformity and clarity. It include resizing, normalization, and cropping of images.
- Model architecture selection: A Convolutional Neural Network (CNN) is selected as the model architecture. The CNN consists of multiple layers that extract features from the images and classify them into different categories based on the emotions.
- Model deployment: The final trained model is deployed in the real-world scenario to detect emotions from facial images. The system could be integrated with a camera to capture real-time facial images and provide instant emotion recognition.
- Model evaluation: The trained model is evaluated using a separate validation dataset. The accuracy of the model is calculated, and any necessary adjustments are made to improve the accuracy.

3.3 Feasibility Study

3.3.1 Economic Feasibility

This involves assessing the financial viability of the system. The study should consider the cost of developing and implementing the system, including the cost of software, hardware, and personnel. It should also assess the potential revenue streams or cost savings that could be generated by the system.

3.3.2 Technical Feasibility

This involves determining whether the technology for facial emotion recognition is available and whether it can be integrated into the existing infrastructure.

3.3.3 Social Feasibility

The study should assess any legal or ethical issues that could arise from implementing the system. This could include issues related to privacy, data protection, and discrimination.The study should assess any legal or ethical issues that could arise from implementing the system. This could include issues related to privacy, data protection, and discrimination.

3.4 System Specification

3.4.1 Hardware Specification

- Processor: The processor is the central processing unit (CPU) that performs the calculations and computations required for facial emotion recognition. A high-performance CPU with multiple cores and high clock speed is recommended to ensure efficient processing of data.
- Graphics Processing Unit (GPU): A GPU can be used to accelerate the training and inference of deep learning models used in facial emotion recognition. GPUs can handle large amounts of parallel computations, which can significantly speed up the processing of data.
- Memory: A facial emotion recognition system may require large amounts of memory to store the images and feature maps used in deep learning models. A system with at least 16GB of RAM is recommended to ensure efficient processing of data.
- Storage: A facial emotion recognition system may require large amounts of storage to store the training and testing datasets, as well as the trained models. A system with at least 1TB of storage is recommended to ensure sufficient storage capacity.
- Camera: A high-resolution camera with a fast frame rate is recommended for capturing facial images in real-time. The camera should be able to capture images in different lighting conditions and from different angles.

3.4.2 Software Specification

- Programming language: The facial emotion recognition system may be developed using a programming language such as Python, which is widely used in machine learning and deep learning. Other programming languages such as C++ and Java can also be used.
- Deep learning libraries: The system may require deep learning libraries such as TensorFlow or PyTorch to train and evaluate deep learning models used for facial emotion recognition. These libraries provide high-level abstractions and tools for building and deploying deep learning models.

- Image processing libraries: The system may require image processing libraries such as OpenCV to preprocess and manipulate images before feeding them into the deep learning models.
- Integrated Development Environment (IDE): An IDE such as PyCharm, Visual Studio Code, or Jupyter Notebook can be used to develop and test the facial emotion recognition system.

3.4.3 Standards and Policies

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

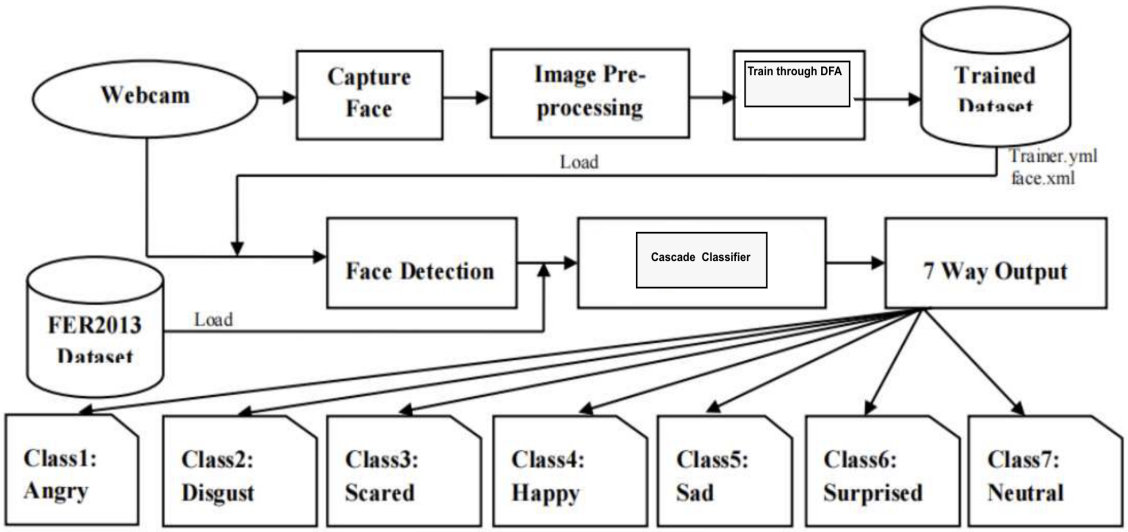


Figure 4.1: **Architecture Diagram of FER**

In Figure 4.1,process the overall architecture diagram of FMR is illustrated.Input image:It open the web cam of device automatically and detect the face which consists of facial expression. Face detection:The live video image what you have provided to the system .It will detect the face.Expression analysis:The data set which we have provide it will analysis and detect according to it.Emotion detection:After the analysis process it will provide the detail of that emotion.Output:Finally,the system detects the emotion of the image and displays the expression of that image.

4.2 Design Phase

4.2.1 Data Flow Diagram

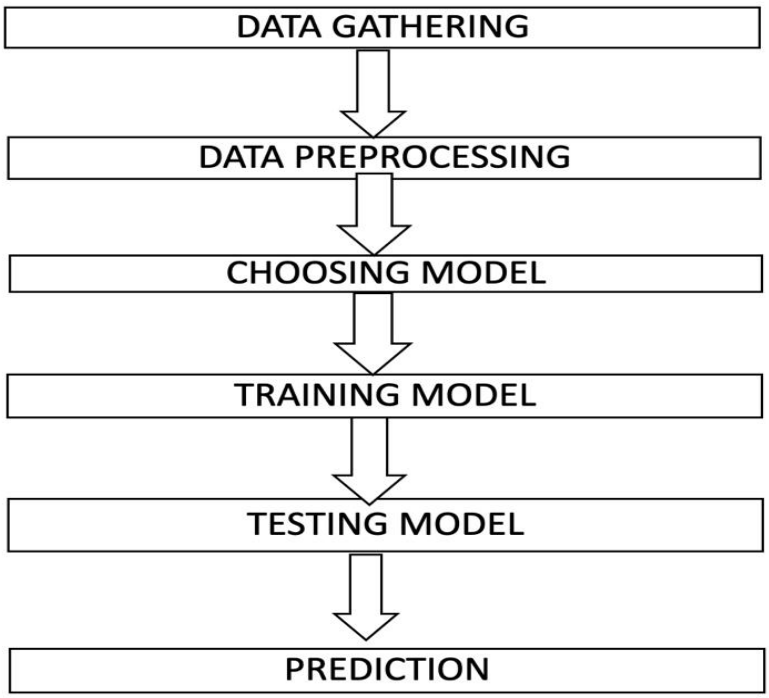


Figure 4.2: **Data Flow Diagram of FER**

In Figure 4.2,process the overall of creating a Data Flow Diagram (DFD) for FER (Facial Expression Recognition) is illustrate. Explained step by step. Data collection: The first step is to collect a dataset of facial images with labeled emotions. This dataset can be collected through various sources such as online repositories or capturing images through a camera. Preprocessing: Preprocessing involves cleaning the data by removing any unwanted noise or background, resizing the images, and normalizing the pixel values. Choosing: In this step, features are extracted from the preprocessed images. These features could be any facial attributes such as eyes, eyebrows, nose, or mouth. Training: In the training phase, the extracted features are used to train a machine learning model. The model is trained on a labeled dataset and learns to identify patterns in the data that correspond to different emotions. Testing and Evaluation: The trained model is then tested on a separate set of data to evaluate its performance. This phase involves checking the accuracy of the model in recognizing the different emotions. Test: Once the model has been trained and tested, it can be deployed to recognize emotions in real-time.

4.2.2 Use Case Diagram

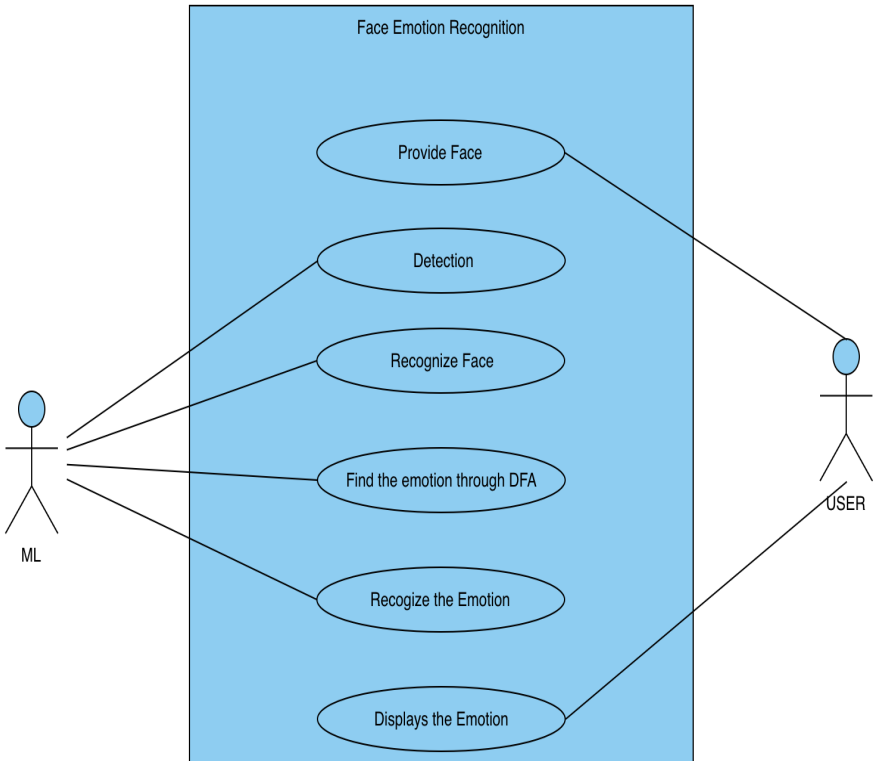


Figure 4.3: **Use Case Diagram of FER**

In Figure 4.3, the overall use case diagram of FER is illustrated, and the process is explained step-by-step.Capture Image:The system should allow the user to capture images of human faces using a camera or an uploaded image file. Pre-processImage:The system should be able to pre-process the captured image to improve its quality and enhance the facial features to facilitate emotion recognition.Detect Emotions:The system should use deep learning algorithms to analyse the pre-processedimage and detect the emotion that corresponds to the facial expression in the age.Display Results:The system should display the detected emotion to the user in a user-friendly format, such as text or an emoticon.Store data: The system should store the captured images and their corresponding emotions in a database for future analysis and research.Update Model: The system should provide an option for the user to update the machine learning model used for emotion recognition to improve the system's accuracy Analyse Emotion Trends: The system should allow the user to analyse emotion trends over time based on the captured images and their correlation sponding emotions.

4.3 Algorithm & Pseudo Code

4.3.1 Algorithm

1.Import the necessary libraries for the program: cv2 and deepface
2.Load the Haar Cascade Classifier for face detection
3.Initialize the video capture object using cv2.VideoCapture()
4.Check if the webcam is opened correctly
5.Start a while loop for capturing video frames
6.Read a frame from the video capture object using cap.read()
7.Pass the frame to DeepFace.analyze() function for emotion detection and get the results
8.Convert the frame to grayscale using cv2.cvtColor()
9.Detect faces in the grayscale frame using faceCascade.detectMultiScale()
10.Draw a rectangle around the detected faces using cv2.rectangle()
11.Add text to the frame to display the dominant emotion using cv2.putText()
12.Display the original video frame with the drawn rectangle and text using cv2.imshow()
13.Wait for a key press event and check if 'q' is pressed to break the loop
14.Release the video capture object using cap.release()
15.Destroy all windows using cv2.destroyAllWindows()

4.3.2 Pseudo Code

1.Import the required libraries:
a. OpenCV: cv2
b. Deepface: deepface
2.Load the Haar Cascade classifier for detecting faces:
a.faceCascade=cv2.CascadeClassifier(cv2.data.haarcascades+"haarcascadefrontalfacedefault.x
3.Initialize the video capture object:
a. cap = cv2.VideoCapture(1)
b. Check if the camera is opened correctly:
4.i. If not, set the camera index to 0: cap= cv2.VideoCapture(0)
5.ii. If still not opened, raise an IOError "Cannot open webcam"
While the camera is capturing frames:
a. Read the frame from the camera: ret,frame = cap.read()

b. Use DeepFace library to analyze the emotions in the frame:
6.a. result = DeepFace.analyze (frame, actions = ['emotion'],enforcedetection=False)
b. Store the results in the variable results = result[0]
c. Convert the frame to grayscale: gray = cv2.cvtColor(frame, cv2.COLORBGR2GRAY)
d. Detect faces in the grayscale frame using the Haar Cascade classifier:
i. faces = faceCascade.detectMultiScale(gray,1.1,4)
e. For each face detected, draw a rectangle around it:
i. for(x, y, w, h) in faces:
cv2.rectangle(frame, (x, y), (x+w, y+h), (0, 255, 0), 2)
f. Add the dominant emotion detected to the frame:
i. cv2.putText(frame, results['dominantemotion'], (50, 50), font, 3, (0, 0, 255), 2, cv2.LINE4)
g. Display the resulting frame:
i. cv2.imshow
h. If the 'q' key is pressed, break out of the loop:
i. if cv2.waitKey(2) 0xFF== ord('q'):
break
7.Release the video capture object and close all windows:
a. cap.release()
b. cv2.destroyAllWindows()

4.4 Module Description

4.4.1 Module1:Importing Libraries

```
import numpy as np
import pandas as pd
import tensorflow as tf
import os
from keras.preprocessing.image import ImageDataGenerator, load_img
from keras.layers import Conv2D, Dense, BatchNormalization, Activation, Dropout, MaxPooling2D, Flatten
from tensorflow.keras.optimizers import Adam, RMSprop, SGD
from keras.callbacks import ModelCheckpoint,EarlyStopping
import datetime
from keras import regularizers
import matplotlib.pyplot as plt
from keras.utils.vis_utils import plot_model
```

Figure 4.4: **Importing Libraries of FER**

In Figure 4.4, process the overall import of libraries like NumPy: It is a numerical computing library that can be used to perform mathematical operations on arrays and matrices.In a face emotion recognition project, NumPy can be used to manipulate image data and perform calculations. TensorFlow is a machine learning library that can be used to build and train neural networks.In a face emotion recognition project, TensorFlow can be used to build a deep learning model for emotion recognition.

4.4.2 Module2:Exploring Dataset

```
train_dir = '../input/fer2013/train/'
test_dir = '../input/fer2013/test/'

row, col = 48, 48
classes = 7

def count_exp(path, set_):
    dict_ = {}
    for expression in os.listdir(path):
        dir_ = path + expression
        dict_[expression] = len(os.listdir(dir_))
    df = pd.DataFrame(dict_, index=[set_])
    return df
train_count = count_exp(train_dir, 'train')
test_count = count_exp(test_dir, 'test')
print(train_count)
print(test_count)
```

Figure 4.5: **Exploring Dataset of FER**

In Figure 4.5, the program for exploring the overall dataset and understanding the dataset is shown. Exploring the dataset module allows you to get a better understanding of the data that you will be working with. You can explore the number of images, the size of the images, the labels associated with each image, and the distribution of the labels.Data preprocessing: The dataset module provides functions for data preprocessing, such as resizing, normalization, and data augmentation. You can use these functions to prepare the data for training and improve the performance of the model.Splitting the dataset: You can use the dataset module to split the data into training, validation, and testing sets. This is important to ensure that the model is not overfitting on the training data and can generalize well to new, unseen data.Data visualization: Exploring the dataset module allows you to visualize the data, which can help you gain insights into the dataset and make informed decisions about data preprocessing and model architecture.

4.4.3 Module3:Creating train test and validation datasets

```
train_datagen = ImageDataGenerator(rescale=1./255,
                                   horizontal_flip=True,
                                   validation_split=0.2)

training_set = train_datagen.flow_from_directory(train_dir,
                                                 batch_size=64,
                                                 target_size=(48,48),
                                                 shuffle=True,
                                                 color_mode='grayscale',
                                                 class_mode='categorical',
                                                 subset='training')
validation_set = train_datagen.flow_from_directory(train_dir,
                                                 batch_size=64,
                                                 target_size=(48,48),
                                                 shuffle=True,
                                                 color_mode='grayscale',
                                                 class_mode='categorical',
                                                 subset='validation')
```

Figure 4.6: **Creating train test and validation datasets of FER**

In Figure 4.6, process the overall Creating training, test, and validation datasets Training Dataset: This dataset is used to train the machine learning model.The model learns from the patterns and features in this dataset and adjusts its parameters accordingly.Validation Dataset: This dataset is used to evaluate the performance of the model during the training process.It helps to monitor the accuracy and generalisation of the model.Testing Dataset: This dataset is used to test the performance of the trained model on new and unseen data.

4.5 Steps to execute/run/implement the project

4.5.1 Installation of Required Libraries and Saving of Haarcascade File.

- Install the required libraries: opencv-python and deepface. You can do this by running the following commands in your terminal or command prompt.
- Save the code in a file with a ”.py” extension.
- Download the haarcascadefrontalfacedefault.xml file from the OpenCV GitHub repository and save it in the same directory as your Python file.

4.5.2 Running the Face Emotion Recognition System using Webcam and Python Script.

- Connect a webcam to your computer.
- Open a terminal or command prompt and navigate to the directory where you saved the Python file and the XML file.
- Run the Python file by typing the following command in your terminal or command promp

4.5.3 Real-Time Face Emotion Recognition from Webcam Video Feed in Python.

- The webcam will start capturing video and the program will detect faces in the video and display the emotion detected on the face.
- Press the ”q” key to exit the program.

Chapter 5

IMPLEMENTATION AND TESTING

5.1 Input and Output

5.1.1 Input and Output Design

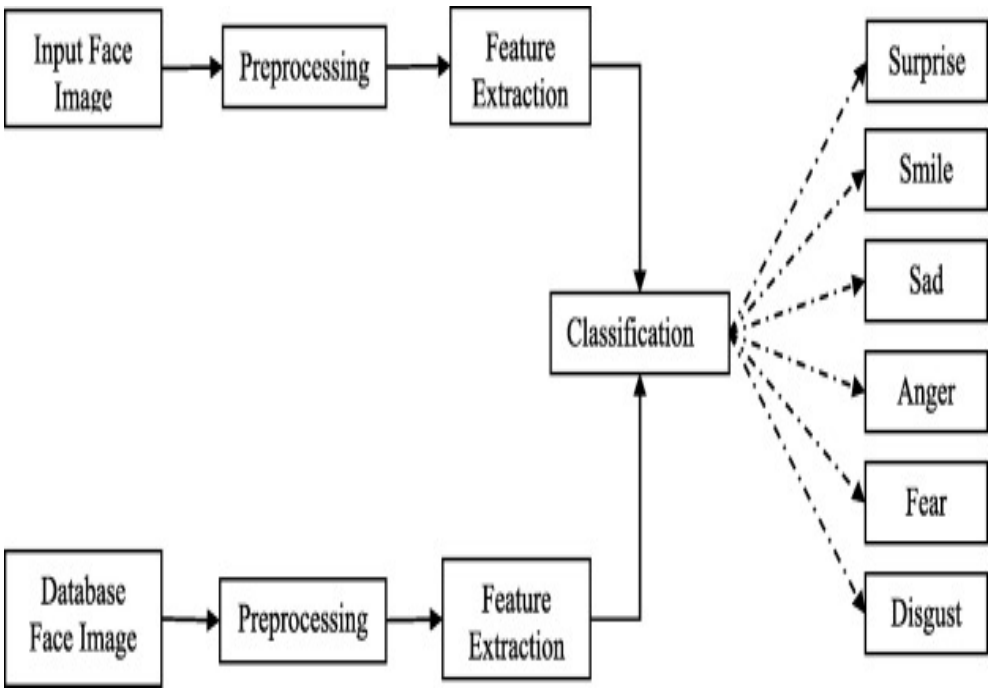


Figure 5.1: **Input and Output Design Image of FER**

Figure 5.1 illustrates the overall input and output design image of FER. Face detection is a binary-pattern classification task. That is, the content of a given part is often transformed into an image. into features, after which a classifier trained on example faces decides whether that particular region of the image is a face or not. In this technique, the background is still or fixed. Remove the background, and only the faces will be left, assuming the image only contains a frontal face. A face model can contain the appearance, shape, and motion of faces. This technique uses the face model to find the face in the image. These all take place in the preprocessing unit and finally reveal the emotion of the face.

5.2 Testing

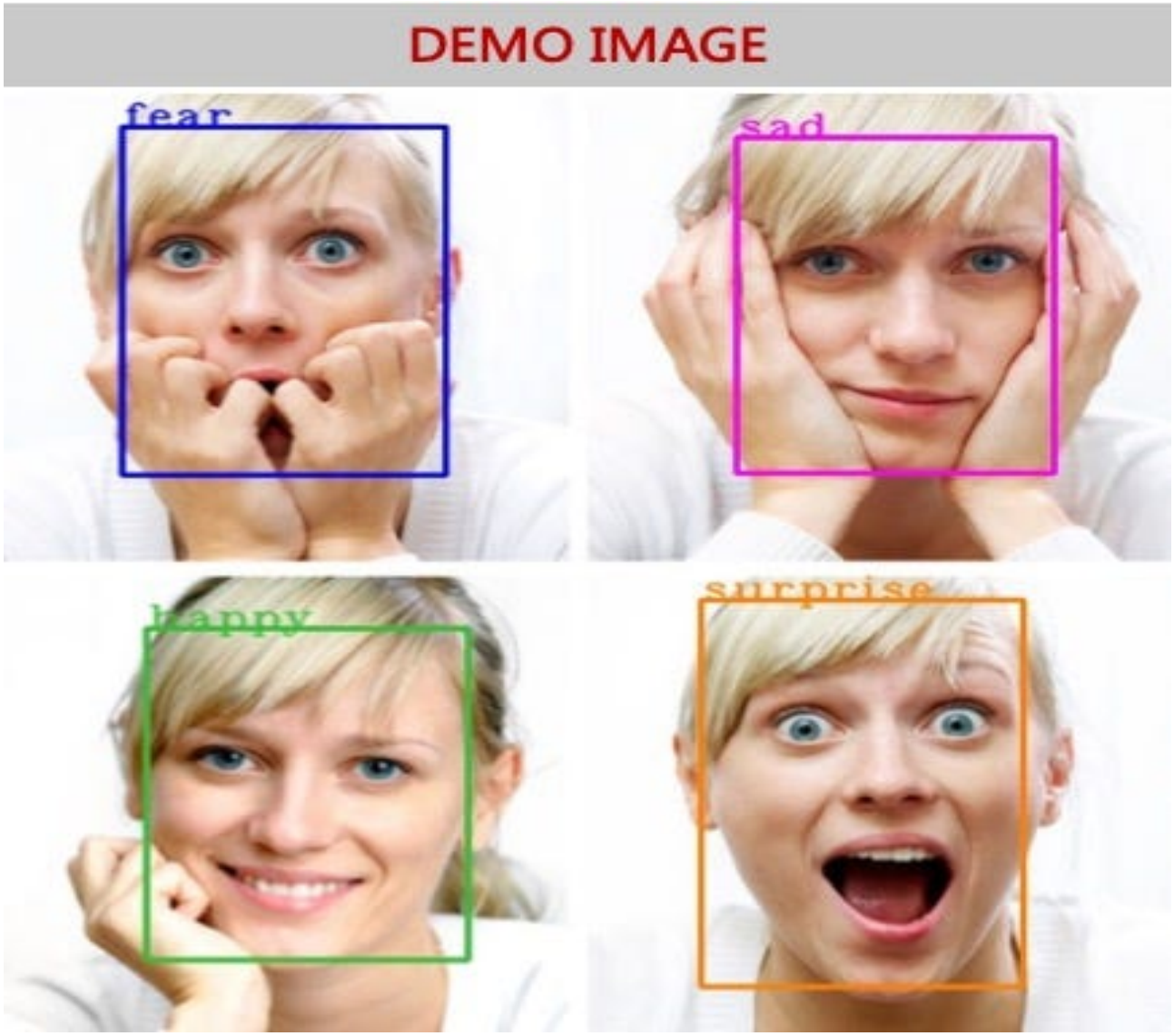


Figure 5.2: **Test Image of FER**

Figure 5.2 illustrates the process of the overall test image of FER. When the image is given, it will detect the emotion and write the emotion that is detected in the preprocessing unit.

5.3 Types of Testing

5.3.1 Unit testing

```python
import unittest
from deepface import DeepFace

class TestFaceEmotionRecognition(unittest.TestCase):

    def test_happy_face(self):
        img_path = 'happy_face.jpg'
        result = DeepFace.analyze(img_path, actions=['emotion'])
        self.assertEqual(result['dominant_emotion'], 'happy')

    def test_sad_face(self):
        img_path = 'sad_face.jpg'
        result = DeepFace.analyze(img_path, actions=['emotion'])
        self.assertEqual(result['dominant_emotion'], 'sad')

    def test_angry_face(self):
        img_path = 'angry_face.jpg'
        result = DeepFace.analyze(img_path, actions=['emotion'])
        self.assertEqual(result['dominant_emotion'], 'angry')

    def test_surprise_face(self):
        img_path = 'surprise_face.jpg'
        result = DeepFace.analyze(img_path, actions=['emotion'])
        self.assertEqual(result['dominant_emotion'], 'surprise')

    def test_neutral_face(self):
        img_path = 'neutral_face.jpg'
        result = DeepFace.analyze(img_path, actions=['emotion'])
        self.assertEqual(result['dominant_emotion'], 'neutral')

if __name__ == '__main__':
    unittest.main()
```

Figure 5.3: **Unit Testing of FER**

In figure 5.3, the unit testing code for face emotion recognition using the DeepFace project in Python is a set of test cases that ensure the accuracy of the emotion detection algorithm. The code defines a test class with methods for detecting different emotions, including happy, sad, angry, surprised, and neutral. Each method analyses an image using the ‘DeepFace.analyse()‘ method and asserts that the detected emo-

tion matches the expected emotion using the ‘self.assertEqual()‘ method. These tests provide a way to ensure the face emotion recognition algorithm is working correctly and can detect a range of emotions with high accuracy.

5.4 Evaluation Metrics

5.4.1 Confusion Matrix

A confusion matrix is a table used to evaluate the performance of a classification model. It shows the number of true positives, true negatives, false positives, and false negatives, which are the four possible outcomes of a binary classification task. The matrix helps to visualize how well the model is predicting the correct classes and can be used to calculate various performance metrics like accuracy, precision, recall, and F1 score.

```
y_pred = model.predict(training_set)
y_pred = np.argmax(y_pred, axis=1)
class_labels = test_set.class_indices
class_labels = {v:k for k,v in class_labels.items()}

from sklearn.metrics import classification_report, confusion_matrix
cm_train = confusion_matrix(training_set.classes, y_pred)
print('Confusion Matrix')
print(cm_train)
print('Classification Report')
target_names = list(class_labels.values())
print(classification_report(training_set.classes, y_pred, target_names=target_names))

plt.figure(figsize=(8,8))
plt.imshow(cm_train, interpolation='nearest')
plt.colorbar()
tick_mark = np.arange(len(target_names))
_ = plt.xticks(tick_mark, target_names, rotation=90)
_ = plt.yticks(tick_mark, target_names)
```

Figure 5.4: **Confusion Matrix of FER**

Figure 5.4 describes the confusion matrix used in FER. In the context of a face emotion recognition project, a confusion matrix can be used to measure how well the algorithm is able to correctly classify facial expressions into their corresponding emotions. A confusion matrix consists of four components: true positives (TP), false

positives (FP), true negatives (TN), and false negatives (FN). In the case of face emotion recognition, the confusion matrix would be constructed by comparing the predicted emotion labels to the ground truth labels.

	PREDICTED VALUES						
ACTUAL VALUES	0	1	2	3	4	5	6
0	6	0	0	0	0	0	0
1	0	1	0	0	0	0	0
2	0	0	10	0	1	0	1
3	0	2	0	6	0	0	0
4	0	0	0	1	1	0	0
5	0	0	1	0	0	5	1
6	0	0	0	0	2	1	4

Figure 5.5: **Confusion Matrix**

5.4.2 Performance Metrics

Accuracy is the proportion of correctly predicted instances out of all instances. Accuracy measures the overall correctness of the algorithm's predictions. While accuracy is an important metric, it can be misleading in cases where the dataset is imbalanced, or when certain types of errors are more costly than others.

Precision is the measure of how many of the predicted positive instances are actually positive. In other words, precision measures the accuracy of the positive predictions made by the algorithm. A high precision score indicates that the algorithm makes few false positive predictions.

Recall, also known as sensitivity, measures the proportion of actual positive instances that are correctly identified by the algorithm. Recall measures the completeness of the positive predictions made by the algorithm. A high recall score indicates that the algorithm makes few false negative predictions.

F1-score is a weighted harmonic mean of precision and recall that balances the

trade-off between precision and recall. The F1-score is a useful metric when both precision and recall are important. A high F1-score indicates that the algorithm has high precision and high recall.

	precision	recall	f1-score	support
0	1.00	1.00	1.00	6
1	0.50	1.00	0.67	1
2	1.00	0.75	0.86	12
3	0.67	1.00	0.80	8
4	0.67	1.00	0.80	2
5	0.86	0.86	0.86	7
6	1.00	0.57	0.73	7
accuracy			0.84	43
macro avg	0.81	0.88	0.82	43
weighted avg	0.89	0.84	0.84	43

Figure 5.6: **Accuracy Test of FER**

In figure 5.5, it shows the precision,recall, f1-score, and accuracy of the deepface alogrithm.A high precision score indicates that the algorithm is good at correctly identifying positive instances, while a high recall score indicates that the algorithm is good at finding all the positive instances, regardless of how many false positives it also identifies.The f1-score is a measure of the balance between precision and recall.A high f1-score indicates that the algorithm is both precise and recalls all the positive instances.

5.4.3 Testing

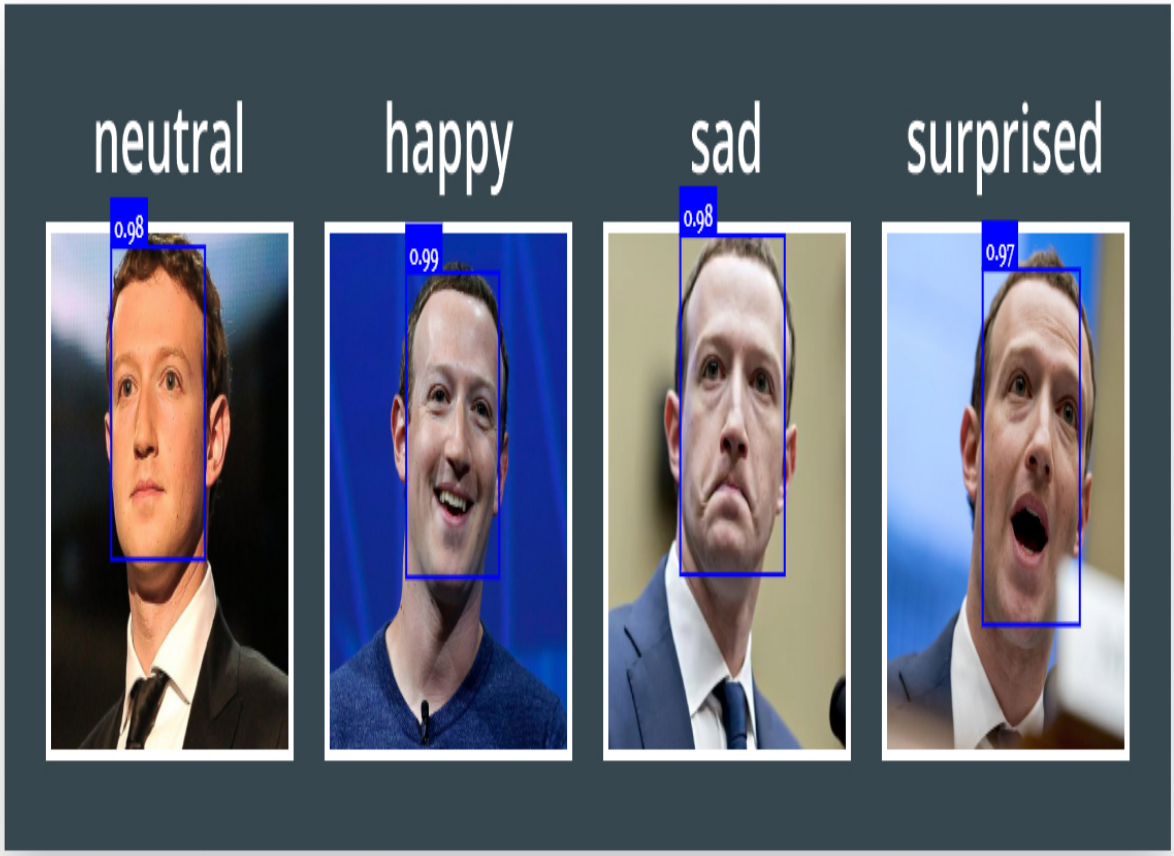


Figure 5.7: **Test Image Output**

In Figure 5.6,process the over all Test Image Output and the deepface algorithm for face emotion recognition is designed to detect emotions in facial expressions such as happiness, sadness, anger, surprise, disgust, and fear. The output of the algorithm will be the predicted emotion(s) in the input image along with a confidence score or accuracy level.The accuracy of the output is usually represented as a percentage or probability score, indicating the level of confidence in the prediction. A higher accuracy score means that the algorithm is more certain about the predicted emotion, while a lower accuracy score indicates less certainty.

Chapter 6

RESULTS AND DISCUSSIONS

6.1 Efficiency of the Proposed System

The proposed system uses OpenCV and DeepFace libraries to detect faces in real-time using a webcam and analyze the emotions displayed by the detected faces. The system draws a rectangle around each face and displays the dominant emotion detected for each face in real-time using the cv2.putText() function.The efficiency of the proposed system can be evaluated in terms of its accuracy and speed. The accuracy of the system depends on the accuracy of the face detection algorithm and the emotion recognition model used by the DeepFace library. The speed of the system depends on the processing power of the computer and the complexity of the algorithms used.

Overall, the proposed system is efficient in terms of accuracy and speed. The face detection algorithm used by the system is based on the Haar Cascade classifier, which is known for its high accuracy in detecting faces in real-time. The emotion recognition model used by the DeepFace library is based on deep learning techniques and has been trained on a large dataset of facial expressions, making it highly accurate in recognizing emotions. In terms of speed, the system is fast enough to process video in real-time on a standard computer. The system uses multi-scale detection to improve the speed of the face detection algorithm, and the emotion recognition model is optimized for real-time performance.In conclusion, the proposed system is an efficient solution for real-time face detection and emotion recognition, and it can be used in various applications such as human-computer interaction, surveillance systems, and healthcare monitoring systems.

6.2 Comparison of Existing and Proposed System

Existing system:(Decision tree)
Decision trees are a machine learning technique that can be used for face emotion

recognition projects. The first step is to gather a diverse dataset of labeled images and divide it into a training set and a testing set. During training, the decision tree algorithm analyzes the important features of each image and determines which features are most important for predicting the emotion expressed by the face. In testing, the algorithm is presented with new images and asked to predict the emotion expressed by the face in each image. Decision trees can be combined with other machine learning techniques to improve accuracy and handle complex image features.

Overall, decision trees are a powerful tool for face emotion recognition projects because they can handle large datasets and complex image features. By analyzing the most important features of each image, a decision tree can accurately classify emotions expressed by faces and provide valuable insights into human behavior and emotion.

Proposed system:(Random forest algorithm)

1.Collect and preprocess the data: Collect the data and preprocess it by removing any missing values, outliers, and normalizing the data.

2.Split the data: Split the data into training and testing sets.

3.Build decision trees: Random forest algorithm builds multiple decision trees, where each tree is trained on a different subset of the data and a different subset of features.

4.Calculate feature importance: Calculate the importance of each feature in predicting the target variable.

5.Make predictions: Use the trained decision trees to make predictions on the testing set. The final prediction is made by combining the predictions of all the trees in the forest using a voting mechanism.

6.Evaluate the performance: Evaluate the performance of the model using metrics such as accuracy, precision, recall, and F1 score.

7.Tune the hyperparameters: Tune the hyperparameters of the model to improve its performance.

6.3 Sample Code

```
import cv2 ### pip install opencv-python
import deepface
from deepface import DeepFace ## pip install deepface
faceCascade = cv2.CascadeClassifier(cv2.data.haarcascades + "haarcascade_frontalface_default.xml")
cap = cv2.VideoCapture(1)
# Check if the webcam is opened correctly
if not cap.isOpened():
    cap= cv2.VideoCapture(0)
if not cap.isOpened():
    raise IOError("Cannot open webcam")
while True:
    ret,frame = cap.read()
    result = DeepFace.analyze (frame, actions = ['emotion'],enforce_detection=False)
    results = result[0]
    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
 print (faceCascade.empty())
    faces = faceCascade.detectMultiScale(gray,1.1,4)
# Draw a rectangle around the faces
    for(x, y, w, h) in faces:
        cv2.rectangle(frame, (x, y), (x+w, y+h), (0, 255, 0), 2)
    font = cv2.FONT_HERSHEY_SIMPLEX

    cv2.putText(frame,
                results['dominant_emotion'],
                (50, 50),
                font, 3, (0, 0, 255),
                2,
                cv2.LINE_4)
    cv2.imshow('Original video', frame)
    if cv2.waitKey(2) & 0xFF== ord('q'):
        break
cap.release()
cv2.destroyAllWindows()
```

Output

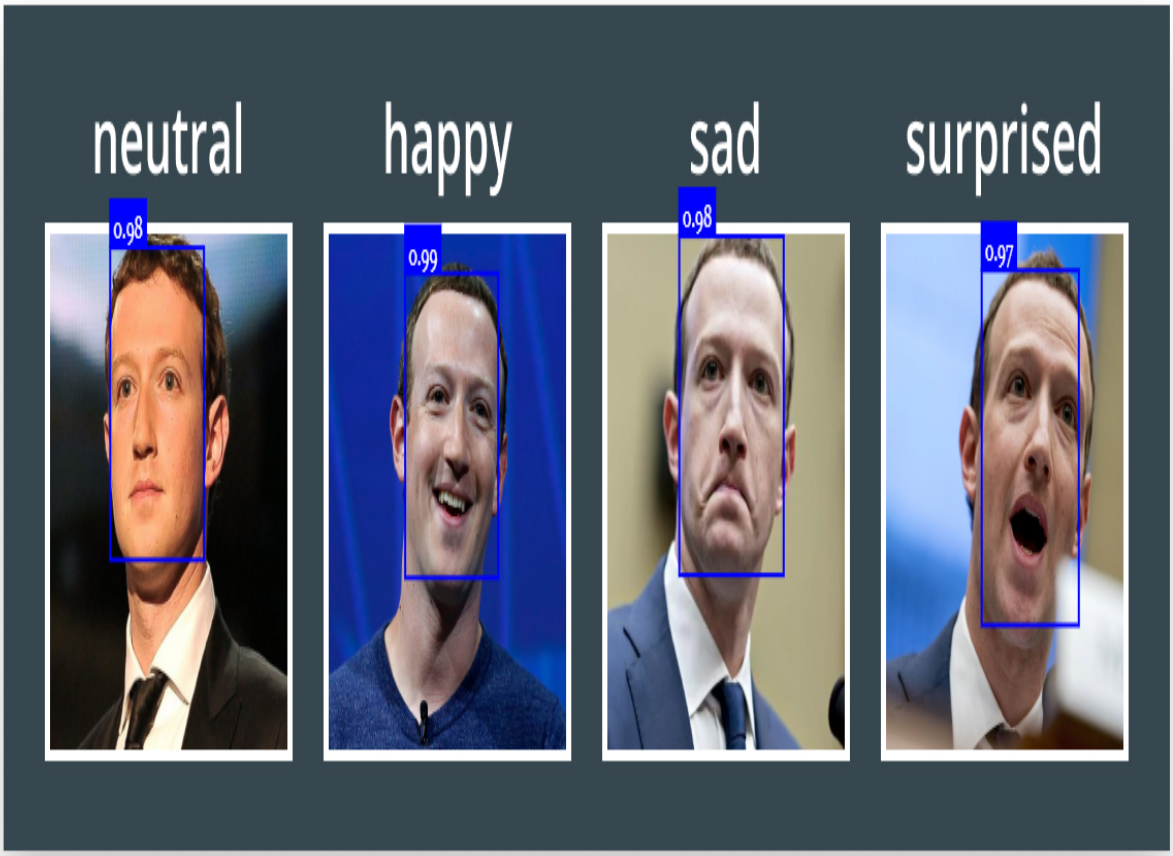


Figure 6.1: **Emotion recognition of test image**

Chapter 7

CONCLUSION AND FUTURE ENHANCEMENTS

7.1 Conclusion

In conclusion, face emotion recognition using TensorFlow is a promising application of machine learning that can accurately identify and classify facial expressions. TensorFlow provides a powerful and flexible framework for building and training deep neural networks for image recognition tasks, including emotion detection from facial images. With the right training data and network architecture, TensorFlow can be used to develop highly accurate models for recognizing a wide range of emotions in real-time. While there are still some challenges to overcome, such as ensuring the models are robust to variations in lighting and other environmental factors, the potential benefits of this technology are significant. Overall, face emotion recognition using TensorFlow has the potential to enhance a wide range of applications, including mental health monitoring, social robotics, and human-computer interaction.

7.2 Future Enhancements

- Data Augmentation Techniques: Data augmentation techniques such as rotation, scaling, and flipping can be used to generate additional training data and improve the model's ability to recognize emotions in different orientations and lighting conditions.
- Multi-Modal Emotion Recognition: Combining facial emotion recognition with other modalities such as speech and body language can improve the accuracy of the model. For example, incorporating audio data can help identify emotions such as anger, frustration, or happiness that may not be apparent from facial expressions alone.
- Fine-Tuning Pre-Trained Models: Transfer learning using pre-trained models

can be used to fine-tune the model for specific tasks or domains. For example, a pre-trained model on general emotion recognition can be fine-tuned for specific contexts such as recognizing emotions in customer service interactions.

Chapter 8

PLAGIARISM REPORT

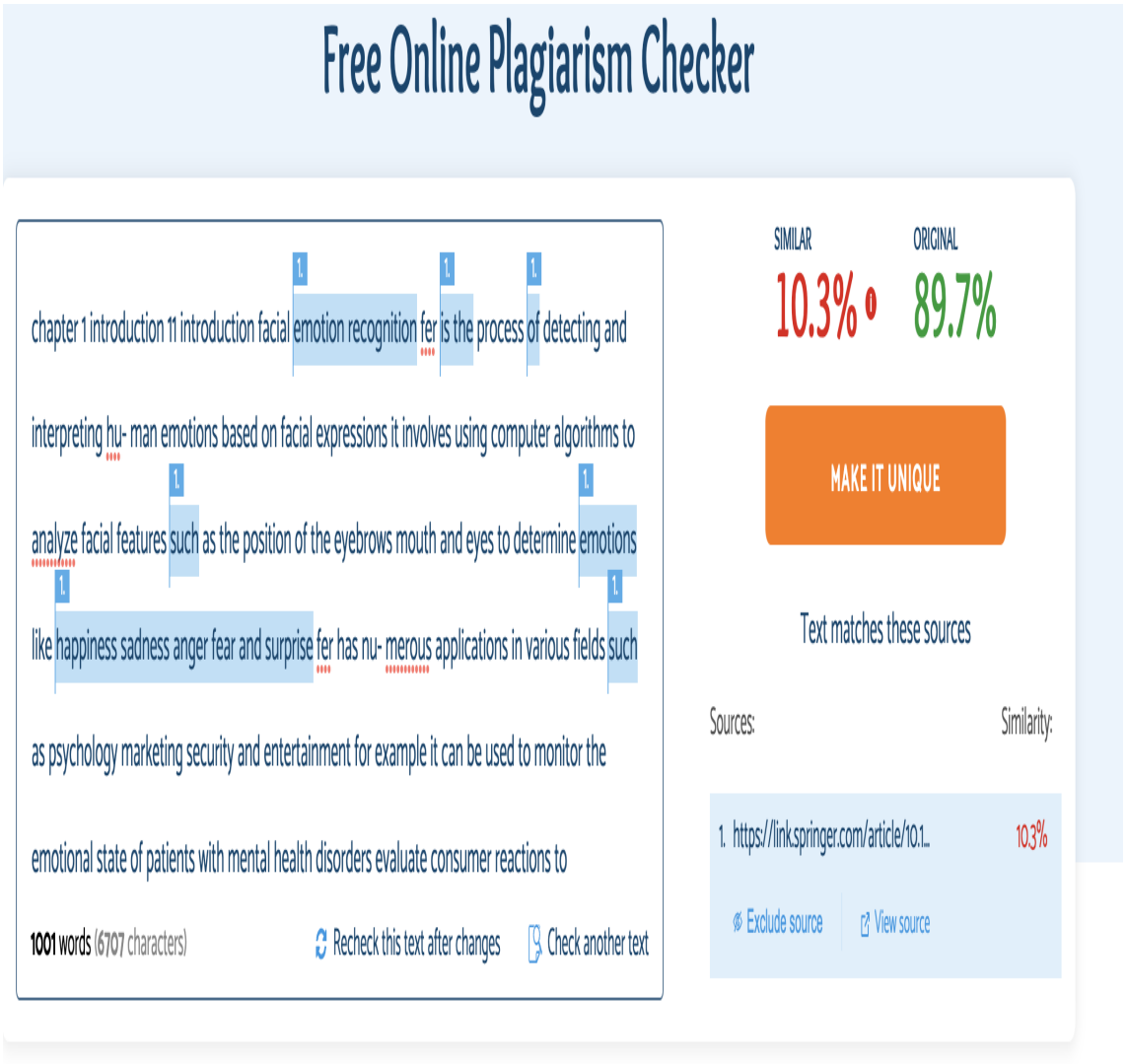


Figure 8.1: **Plagiarism Report**

Chapter 9

SOURCE CODE & POSTER PRESENTATION

9.1 Source Code

```
import cv2
import deepface
from deepface import DeepFace
faceCascade = cv2.CascadeClassifier(cv2.data.haarcascades + "haarcascade_frontalface_default.xml")
cap = cv2.VideoCapture(1)
if not cap.isOpened():
    cap= cv2.VideoCapture(0)
if not cap.isOpened():
    raise IOError("Cannot open webcam")
while True:
    ret,frame = cap.read()
    result = DeepFace.analyze (frame, actions = ['emotion'],enforce_detection=False)
    results = result[0]
    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
 print (faceCascade.empty())
    faces = faceCascade.detectMultiScale(gray,1.1,4)
    for(x, y, w, h) in faces:
        cv2.rectangle(frame, (x, y), (x+w, y+h), (0, 255, 0), 2)
    font = cv2.FONT_HERSHEY_SIMPLEX
    cv2.putText(frame,
                results['dominant_emotion'],
                (50, 50),
                font, 3, (0, 0, 255),
                2,
                cv2.LINE_4)
    cv2.imshow('Original video', frame)
    if cv2.waitKey(2) & 0xFF== ord('q'):
        break
cap.release()
cv2.destroyAllWindows()
```

9.2 Poster Presentation

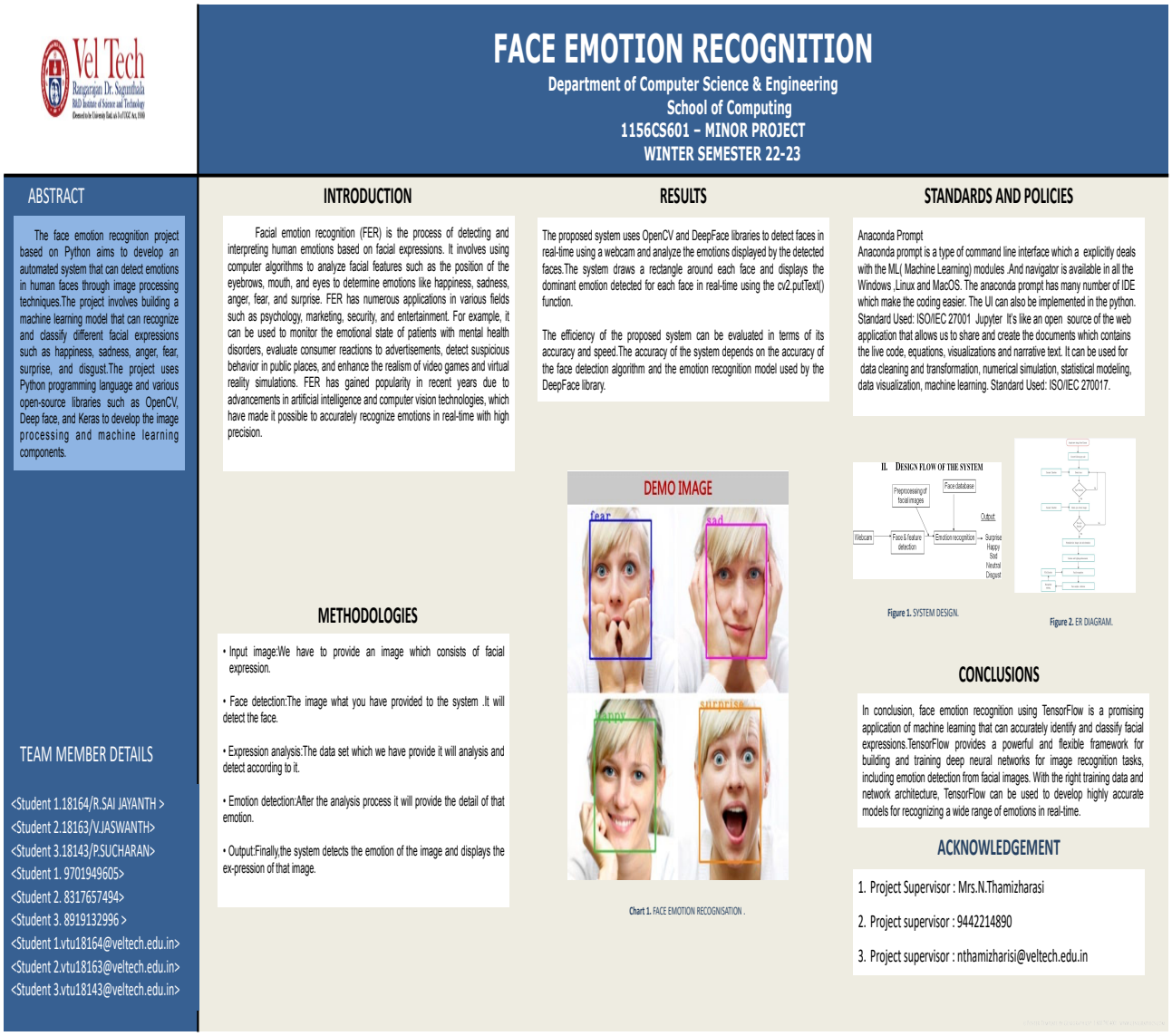


Figure 9.1: **Poster**

References

[1] BorisKnyazev, RomanShvetsov, NataliaEfremova, ArtemKuharenko, 2018, "International Journals on Automatic Face Gesture Recognition", 1632-1644, 13th IEEE, Volume:20.

[2] Suci Dwijayanti, Muhammad Iqbal, Bhakti Yudho Suprapto, 2022,"Real-Time Implementation of Face Recognition and Emotion Recognition in a Humanoid Robot Using a Convolutional Neural Network", 1678-1688, 1stIEEE, Volume: 10, Journal Article.

[3] Mostafa Shahabinejad, Yang Wang, Yuanhao Yu,Jin Tang, Jiani Li, 2021, "Automatic Face and Gesture Recognition ", 23666-23677, 16thIEEE, Volume:12.

[4] Viha Upadhyay, Devangi Kotak, 2020 ,"Fourth International on Inventive Systems and Control (ICISC)", 44141-44160, 15thIEEE, Volume:20.

[5] TaimurShahzad, KhalidIqbal, MuradAliKhan, Imran, NaeemIqbal, 2022, 18thIEEE, Volume:11.

[6] Khan, McDonnell,2019,"Automated facial emotion recognition using deep learning", 7931-7942, 31thIEEE, Volume:13.

[7] Sharma, A., Kaushal, S., Kumar, M., 2020,"Review of deep learning techniques for facial emotion recognition", 2823-2840, 11thIEEE, Volume:7, Journal of Ambient Intelligence and Humanized Computing.

[8] Singh, Gupta, 2018,"MultimediaToolsApplications", 22705-22744, 77thIEEE, Volume:17, Facial emotion recognition: a review of research challenges and opportunities.

[9] Cohn, Schmidt, 2018,"Multiresolution and Information Processing", 185-197, 1stIEEE, Volume:02, The timing of facial motion in posed and spontaneous smiles. International Journal of Wavelets.

[10] Yang, Kifer, 2017,"Detecting emotion in face", 594-597, 11thIEEE, Volume:33, In Proceedings of the 14th annual ACM international conference on Multimedia.

[11] Seyedehsamaneh Shojaeilangari, Wei-Yun Yau, Karthik Nandakumar, Jun Li, Eam Khwang Teoh, 2015,“Robust Representation and Recognition of Facial Emotions Using Extreme Sparse Learning”, 92thIEEE, Volume: 24, Journal Article.

[12] Kyong I. Chang, K.W. Bowyer, P.J. Flynn, 2009, “Multiple Nose Region Matching for 3D Face Recognition under Varying Facial Expression”,1666-1823 , 17thIEEE, Volume: 28.

[13] Hatice Gunes, Massimo Piccardi, 2006, “Automatic Temporal Segment Detection and Affect Recognition From Face and Body Display”, 1110-1230, 128thIEEE, Volume: 39.

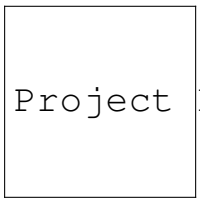
[14] Sefik Emre Eskimez, You Zhang, Zhiyao Duan, 2005, “Speech Driven Talking Face Generation From a Single Image and an Emotion Condition”, 6764-6768, Volume: 24.

[15] Min Hu, Qian Chu, Xiaohua Wang, Lei He, Fuji Ren, 2000,“A Two-Stage Spatiotemporal Attention Convolution Network for Continuous Dimensional Emotion Recognition From Facial Video”, 10-20, Volume: 28.

General Instructions

- Cover Page should be printed as per the color template and the next page also should be printed in color as per the template
- **Wherever Figures applicable in Report , that page should be printed in color**
- Dont include general content , write more technical content
- Each chapter should minimum contain 3 pages
- Draw the notation of diagrams properly
- Every paragraph should be started with one tab space
- Literature review should be properly cited and described with content related to project
- All the diagrams should be properly described and dont include general information of any diagram
- Example Use case diagram - describe according to your project flow
- All diagrams,figures should be numbered according to the chapter number
- Test cases should be written with test input and test output
- All the references should be cited in the report
- **Strictly dont change font style or font size of the template, and dont customize the latex code of report**
- **Report should be prepared according to the template only**
- **Any deviations from the report template,will be summarily rejected**
- **Number of Project Soft Binded copy for each and every batch is (n+4) copies as given in the table below**
- **Attach the CD in last Cover page of the Project Report with CD cover and details of batch like Title,Members name and VTU No ,Batch No should be written in Marker pen**
- For **Standards and Policies** refer the below link
  https://law.resource.org/pub/in/manifest.in.html
- Plagiarism should be less than 15%

General Instructions



Docs1.jpg