**The Islamia University of Bahawalpur**

**Department of Software Engineering**

**Faculty of Computing**

****

**Software Requirement Specification (SRS)**

**for**

**Facial Emotion Detection**

**Table of Contents**

1. Introduction 1

1.1 Purpose 1

1.2 Scope 1

1.3 Definitions, Acronyms, and Abbreviations. 1

1.4 References 1

1.5 Overview 1

2. The Overall Description 2

2.1 Product Perspective 2

2.1.1 Operations 2

2.1.2 Site Adaptation Requirements 3

2.2 Product Functions 3

2.3 User Characteristics 3

2.4 General Constraints 4

2.5 Assumptions and Dependencies 4

3. Specific Requirements 4

3.1 External Interface Requirements 5

3.1.1 System Interfaces 5

3.1.2 Interfaces 5

3.1.3 Hardware Interfaces 5

3.1.4 Software Interfaces 5

3.1.5 Communications Interfaces 6

3.2 Functional Requirements 6

3.2.1 <Functional Requirement or Feature #1> 6

3.2.2 <Functional Requirement or Feature #2> 7

3.3 Use Cases 7

3.3.1 Use Case #1 7

3.3.2 Use Case #2 7

3.4 Classes / Objects 7

3.4.1 <Class / Object #1> 7

3.4.2 <Class / Object #2> 7

3.5 Non-Functional Requirements 7

3.5.1 Performance 8

3.5.2 Reliability 8

3.5.3 Availability 8

3.5.4 Security 8

3.5.5 Maintainability 8

3.5.6 Portability 8

3.6 Inverse Requirements 8

3.7 Logical Database Requirements 8

3.8 Design Constraints 8

3.8.1 Standards Compliance 8

4. Analysis Models 9

4.1 Sequence Diagrams 9

4.2 Data Flow Diagrams (DFD) 9

4.3 State-Transition Diagrams (STD) 9

5. Supporting Information 9

Appendix A – Background Research on: 9

Appendix B – Data Dictionary 9

# 1. Introduction

Facial emotion recognition has emerged as an important research area in the field of computer vision and artificial intelligence. It involves detecting and classifying facial expressions in real time, which can be used in various applications such as human-computer interaction, mental health monitoring, and marketing research.

The goal of this project is to develop a Facial Emotion Recognition system using the deepface library. Deepface is an open-source deep learning library that enables facial analysis through deep neural networks. The system will take input from a camera, detect faces in the image or video stream, and classify the detected facial expressions into one of the basic emotions such as happiness, sadness, anger, fear, surprise, or neutral.

This project will build upon existing research in the field of facial emotion recognition. Several recent studies have focused on using deep learning techniques for emotion recognition from facial expressions. For instance, a study by K. Khorrami et al. [1] proposed a deep convolutional neural network (CNN) architecture for facial expression recognition, achieving state-of-the-art performance on several benchmark datasets. Another study by X. Liu et al. [2] proposed a multi-task deep neural network that can recognize both facial expressions and facial action units.

The proposed system has potential applications in various domains such as mental health diagnosis, marketing research, and human-computer interaction. It can also be extended to recognize emotions in videos, which can be used in areas such as surveillance and movie analysis.

## Purpose The project is intended to provide an automated system for analyzing emotions, which is a critical component in many fields, such as psychology, marketing, and human-computer interaction.

## Scope The scope of the Facial Emotion Recognition using Deepface library in Python with Jupyter Compiler project is to develop a software system capable of analyzing human emotions based on their facial expressions. The software will be designed to capture the facial expressions of individuals in real-time using a camera feed and classify them into one of the seven basic emotions, including happy, sad, angry, surprise, fear, disgust, or neutral.

## 1.3 Definitions, Acronyms, and Abbreviations

* Software Engineering is abbreviated as SE.
* Facial Emotion Recognition is abbreviated as FER.
* DeepFace is a deep learning facial recognition system developed by Facebook's AI research team. It uses a deep convolutional neural network (CNN) to recognize and verify faces in images and videos. The network is trained on a large dataset of labeled faces, which enables it to achieve high accuracy in face recognition tasks.
* AI: Artificial Intelligence
* API: Application Programming Interface
* CNN: Convolutional Neural Network

## 1.4 References

1. *Khorrami, P., Luan, S., and Savvides, M. "Deep Convolutional Neural Networks for Facial Expression Recognition." IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015.*
2. *Liu, X., Han, Z., and Chang, S. "Deep Learning for Facial Action Unit Intensity Estimation." IEEE International Conference on Automatic Face and Gesture Recognition (FG), 2017.*
3. *DeepFace Library: https://github.com/serengil/deepface*
4. *Python: https://www.python.org/*
5. *Jupyter: https://jupyter.org/*
6. *Mollah, M. M. H., & Rahman, M. S. (2021). Emotion Detection and Recognition using Deep Learning Techniques: A Review. Journal of Cognitive Computing, 3(3), 304-317.*
7. *Singh, A., Sharma, N., & Gupta, M. K. (2020). Facial Emotion Recognition using Deep Learning Techniques: A Review. International Journal of Machine Learning and Computing, 10(1), 25-33.*
8. *Dhall, A., Goecke, R., Joshi, J., & Sikka, K. (2016). The Emotion Recognition in the Wild Challenge 2016: A benchmark for Emotion Recognition and Affect Analysis in-the-wild. Proceedings of the 18th ACM International Conference on Multimodal Interaction, 501-507.*

## 1.5 Overview The rest of this SRS document outlines the overall description of the project, including product perspective, product functions, user characteristics, general constraints, assumptions and dependencies, external interface requirements, functional requirements, non-functional requirements, design constraints, and standards compliance.

# 2. The Overall Description

## 2.1 Product Perspective

### The Facial Emotion Detection Model using DeepFace Library will be a standalone software that can be used by anyone who wants to detect basic facial emotions in real-time. The model will use a pre-trained Convolutional Neural Network (CNN) model from the DeepFace Library to detect emotions in images.

### 2.1.1 Operations

The model will have a simple user interface that displays the detected emotion on the screen. The following are the normal and special operations required by the user:

1. *The user will provide an image containing a face to the software*
2. *The software will analyze the image and detect the facial emotion*
3. *The software will return the detected emotion to the user*

### 2.1.2 Site Adaptation Requirements

The model does not have any site adaptation requirements.

## 2.2 Product Functions

The product functions of the Facial Emotion Detection Model using DeepFace Library include:

1. *Detect basic seven facial emotions in real-time*
2. *Display the detected emotion on the screen*

## 2.3 User Characteristics

The users of the Facial Emotion Detection Model using DeepFace Library may include anyone who wants to detect basic facial emotions in real-time..

## 2.4 General Constraints

The general constraints of the model include:

1. *The model will only be able to detect basic seven facial emotions*
2. *The model will be developed using the DeepFace Library*
3. *The model will be coded in Python using Jupyter as the compiler*.

## 2.5 Assumptions and Dependencies

The assumptions and dependencies of the Facial Emotion Detection Model using DeepFace Library include:

* *The system will be run on a machine with sufficient processing power and memory to handle real-time image processing*
* *The DeepFace Library will be available and functional*
* *The Python programming language and Jupyter compiler will be available and functional.*

# 3. Specific Requirements

## 3.1 External Interface Requirements

### 3.1.1 System Interfaces

### The AI model will be designed to work with standard operating systems, including Windows, macOS, and Linux.

### 3.1.2 Interfaces

### The user interface will consist of a simple and intuitive graphical user interface (GUI) that allows users to input facial images or video streams and view the results of the emotion detection.

### 3.1.3 Hardware Interfaces

### The AI model will work with standard hardware components, including webcams, microphones, and standard computer systems

### 3.1.4 Software Interfaces The system will be developed using the Python programming language and will require the following software components:

### Python 3.7 or above.

### Jupyter Notebook.

### DeepFace library

### OpenCV library.

### Matplotlib library.

### 3.1.5 Communications Interfaces The system does not require any network communication interfaces.

## 3.2 Functional Requirements

## The following functional requirements need to be met by the system: *(1) The system should be able to detect the seven basic facial emotions: happiness, sadness, surprise, anger, fear, disgust, and neutral. (2) The system should be able to analyze the facial expression of an individual in real-time. (3) The system should be able to capture the facial expressions of an individual using a camera. (4) The system should be able to display the results of the emotion detection in real-time. (5) The system should be able to store the emotion detection results in a local file.*

## 3.3 Use Cases

The following use cases have been identified for the system:

### 3.3.1 Use Case #1 Real-time Emotion Detection

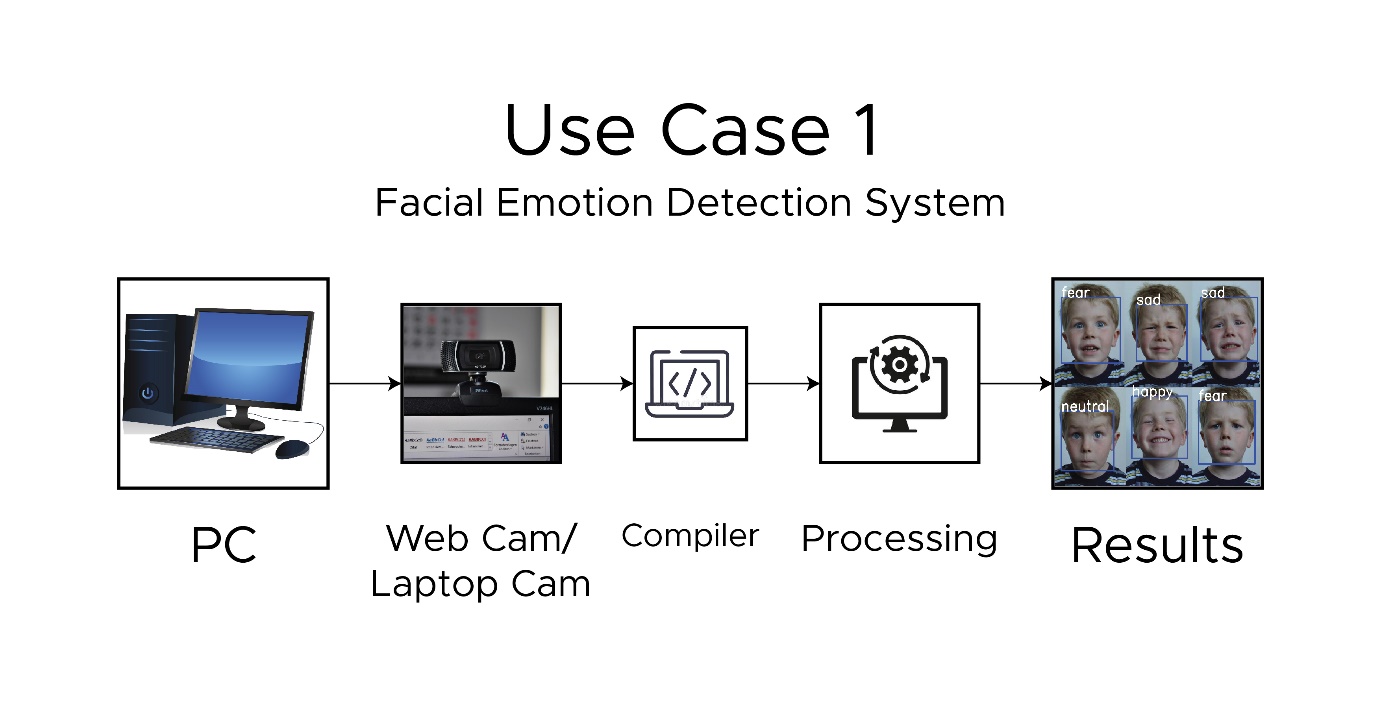


Figure:3.3.1.1 Use Case by Camera Data Diagram of Facial Emotion Detection

### 3.3.2 Use Case #2 Saved Images Emotion Detection

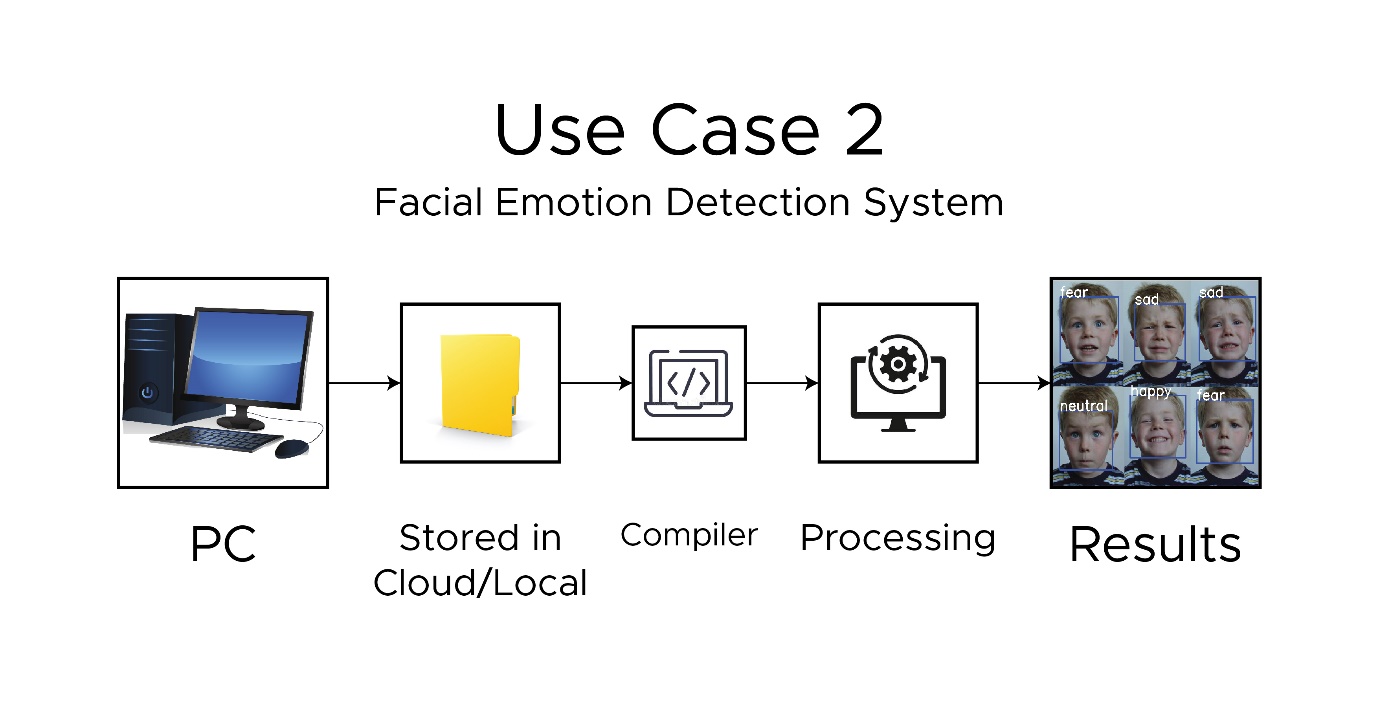


Figure:3.3.1.2 Use Case by Image Data Diagram of Facial Emotion Detection

## 3.4 Classes / Objects

* Camera: This class is responsible for capturing the facial expressions of an individual using the camera.
* Emotion Detector: This class is responsible for detecting the facial emotions of an individual in real-time.
* Result Saver: This class is responsible for saving the results of the emotion detection in a local file.

## 3.5 Non-Functional Requirements

### The following non-functional requirements need to be met by the system:

### Performance: The system should be able to perform real-time emotion detection without any noticeable lag.

### Reliability: The system must be reliable and give accurate results.

### Availability: The system should be available to the user at all times.

### Security: The system should not collect or store any sensitive information about the user.

### Maintainability: The system should be easy to maintain and update.

### Portability: The system should be portable and able to run on different platforms.

## 3.6 Inverse Requirements

There are no such inverse requirements of this system.

## 3.7 Logical Database Requirements

This section summarizes the requirements for all information that must be included in the database. This may include:

1. Types of information used by different functions
2. Frequency of use
3. Accessing capabilities
4. Data entities and their relationships
5. Integrity constraints
6. Data retention requirements

If the customer provides you with a sample file, you can post the sample here. ER diagrams (or static diagrams) can be used here to show relationship information. Remember, a picture is worth a thousand confusing words.

## 3.8 Design Constraints

The following design constraints need to be met by the system:

### 3.8.1 Standards Compliance

The system should follow the best coding practices and design patterns recommended by the Python community.

# 4. Analysis Models

List all analytical methods used to develop the specific requirements provided in this SRS. Each model should have instructions and descriptions. In addition, all models must comply with SRS requirements.

## 

## 4.1 Sequence Diagrams

## 

## Figure:4.1.1 Camera Use Sequence Diagram of Facial Emotion Detection

## 

Figure:4.1.2 Image Data Use Sequence Diagram of Facial Emotion Detection

## 4.2 Data Flow Diagram (DFD)

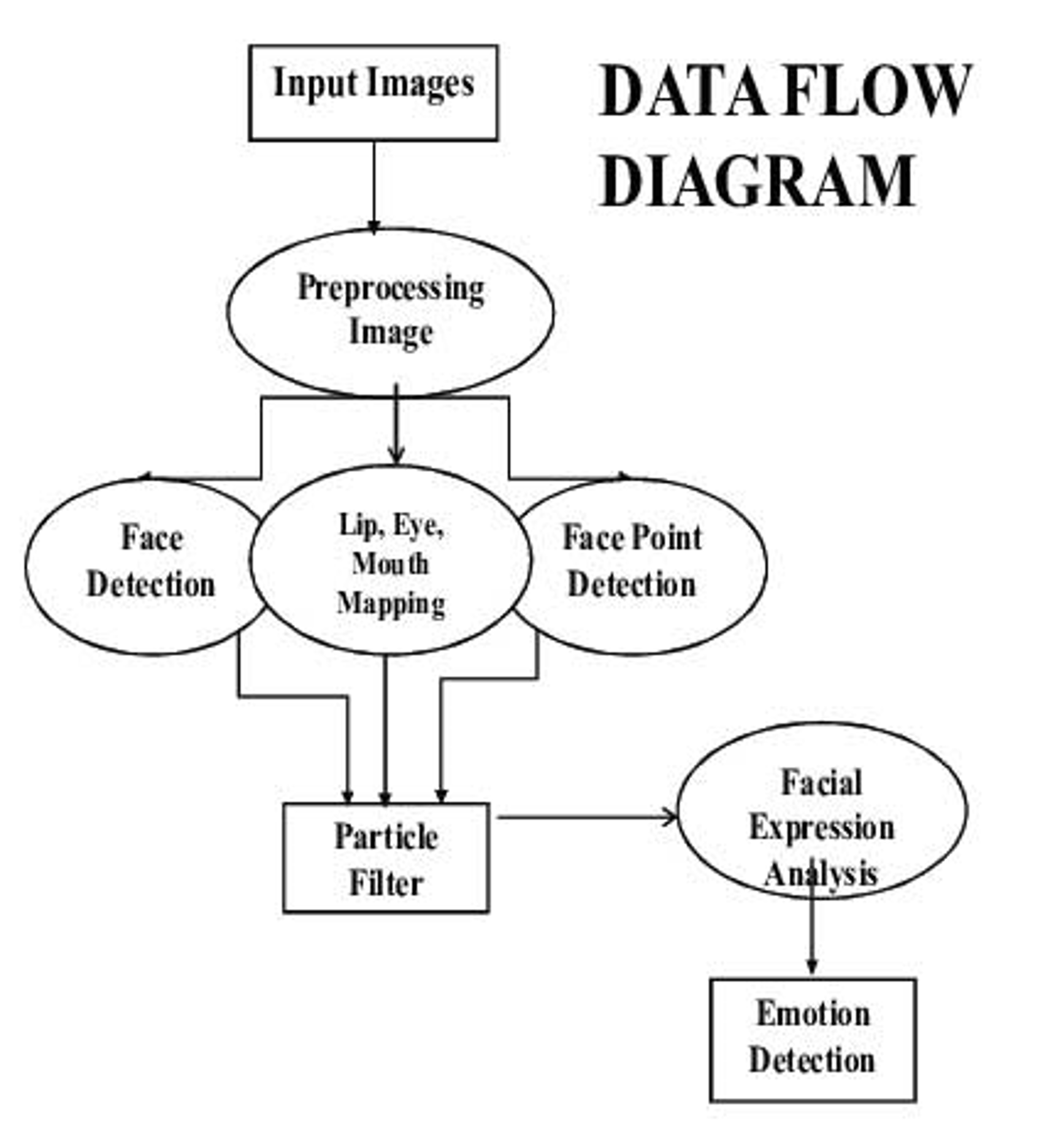


Figure:4.2.1 Basic Data Flow Diagram of Facial Emotion Detection

## 4.3 State-Transition Diagram (STD)

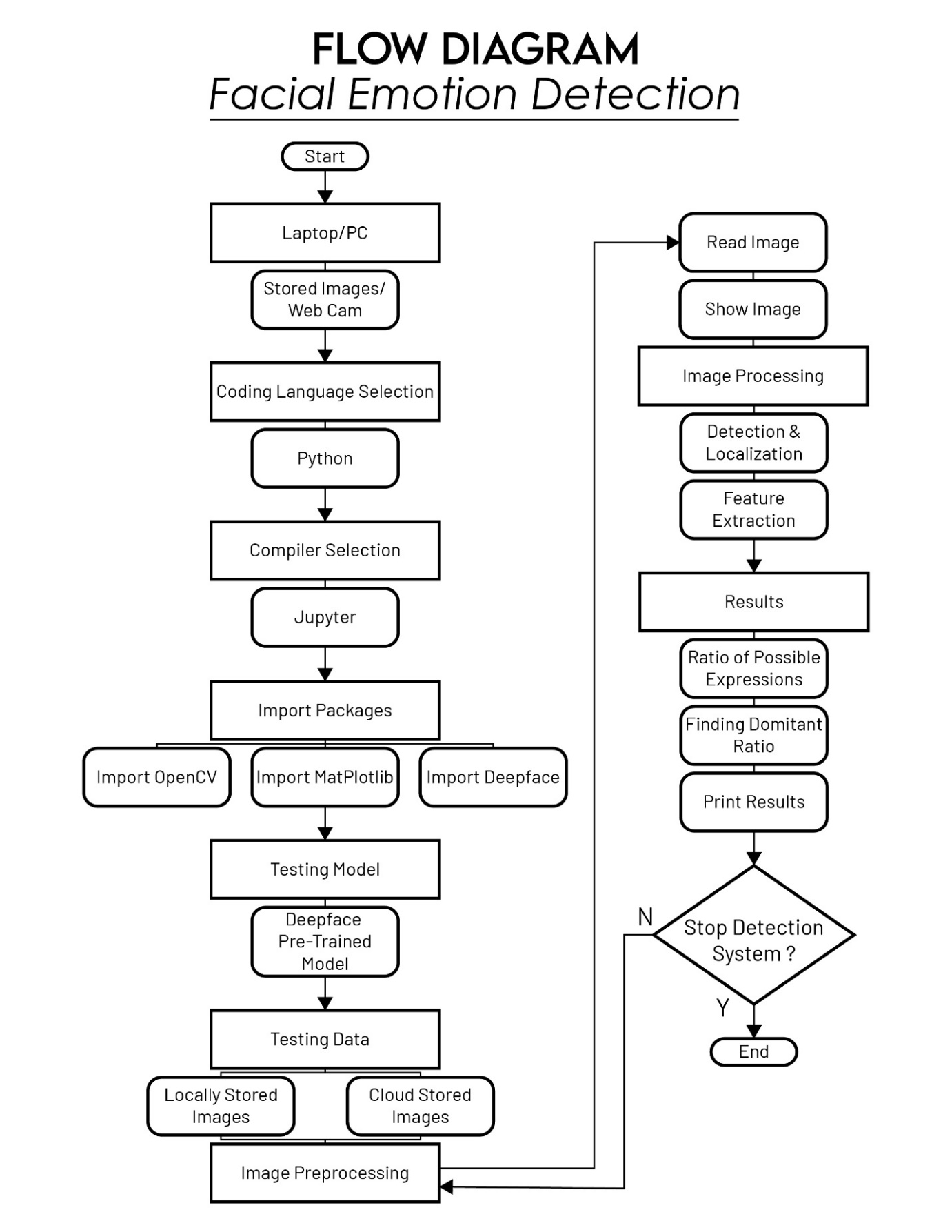
****

Figure:4.3.1 Explained State-Transition Diagram of Facial Emotion Detection

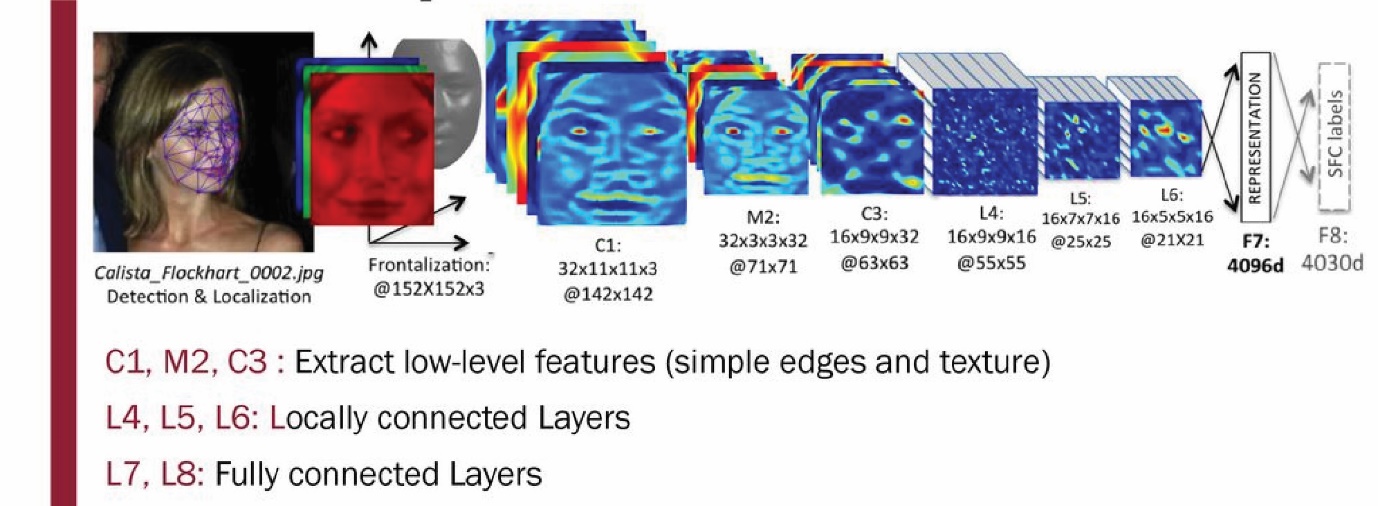


Figure:4.3.2 Working State-Transition Diagram of Facial Emotion Detection

# 5. Supporting Information

Facial emotion detection is a fascinating and rapidly advancing field within the domain of computer vision and artificial intelligence. Emotion detection models can have a wide range of applications, from improving human-computer interaction to enhancing the safety of self-driving cars. The DeepFace library is a state-of-the-art tool for facial emotion detection that is based on deep neural networks. The model is capable of recognizing facial expressions with remarkable accuracy and can be trained on large datasets to improve its performance even further. Using the DeepFace library, developers and researchers can easily implement emotion detection capabilities in their own applications, without needing to have in-depth knowledge of machine learning and computer vision. Overall, the DeepFace library provides a powerful and accessible tool for anyone looking to integrate emotion detection capabilities into their projects.

### **Appendix A – Background Research on:**

1. Facial recognition technology and its applications
2. Deep learning and neural networks
3. Convolutional Neural Networks (CNNs)
4. Transfer learning and fine-tuning pre-trained models
5. Theoretical foundations of emotion detection from facial expressions
6. Datasets and benchmarks for emotion detection from facial expressions
7. Evaluation metrics for emotion detection models
8. Ethical considerations around facial recognition technology and privacy concerns
9. State-of-the-art research and developments in emotion detection from facial expressions

### **Appendix B – Data Dictionary**

Facial Emotion Detection : The process of automatically detecting and recognizing emotions from human facial expressions.

DeepFace: A deep learning facial recognition system developed by Facebook that can be used for facial emotion detection.

Preprocessing: The process of preparing input data for a machine learning algorithm, such as normalizing or scaling the data.

CNN: Convolutional neural network, a type of deep learning algorithm commonly used for image recognition tasks.

Training data: A set of input/output pairs used to train a machine learning model. In this case, images of faces labeled with the corresponding emotion they represent.

Test data: A set of input/output pairs used to evaluate the performance of a machine learning model. In this case, images of faces not seen during training labeled with the corresponding emotion they represent.

Accuracy: A measure of how well a machine learning model is able to correctly predict the output given an input.

Precision: A measure of how often a machine learning model correctly predicts a positive class (e.g. a specific emotion), given that it has made a positive prediction.

Recall: A measure of how often a machine learning model correctly predicts a positive class (e.g. a specific emotion), given that the actual class is positive.

F1 score: A measure of a machine learning model's accuracy that takes both precision and recall into account. It is the harmonic mean of precision and recall.

User interface: The part of the system that the user interacts with, which allows them to input an image and receive the predicted emotion.

Web application: A software application that runs on a web server and is accessed via a web browser.

API: Application Programming Interface, framework and software development tools.