

FACE DETECTION AND RECOGNITION **SYSTEM**

Software Engineering

Project Report

Submitted by:

Himanshu Sharma(1602035)

Anuj Agarwal(1602036)

Under the supervision of:

Ms. Arti Dua

2018

Bhaskaracharya College of Applied Sciences

University of Delhi

ACKNOWLEDGEMENT

On the successful completion of our project **FACE DETECTION AND RECOGNITION**, we would like to express our sincere gratitude to everyone who helped us in the completion of the project.

We are sincerely thankful to our project guide **Mrs. ARTI DUA** for her interest, guidance and suggestions throughout the course of the project. We feel honoured and privileged to work under her. She shared his vast pool of knowledge with us that helped us steer through all the difficulties with ease. This project would not have been possible without her guidance.

CERTIFICATE

This is to certify that the project entitled “FACE DETECTION AND RECOGNITION” submitted by Himanshu Sharma, Anuj Agarwal, Shivam Dixit has been carried out under our supervision. The project has been submitted as per the requirements in the fourth semester of B.Sc. Computer Science.

Mentor

Mrs. Arti Dua

TABLE OF CONTENTS

1 Introduction	6
1.1 Purpose	6
1.2 scope	6
1.3 Definitions, acronyms, and abbreviations	7
1.4 overview	7
1.5 Process Model	7
2 Software Requirement Specification	7
2.1 Overall description	7
2.1.1 product perspective	7
2.1.2 Product Function	8
2.1.2 User characteristics	8
2.1.3 General Constraints	8
1. IMAGE QUALITY:	8
2. IMAGE SIZE:	8
3. FACE ANGLE:	8
2.1.4 Assumptions and Dependencies	8
2.2 Specific requirements	8
3.1 User interfaces	8
3.2 Hardware interfaces	8
3.3 Software interfaces	8
3.4 Software Requirements	8
3.5 Hardware Requirements	9
3.6 Modules	9
2.3 Data flow Diagram	9
2.3.1 Level 0 DFD	9
2.3.2 Level 1 DFD	10
2.3.2 DFD level 2	10
2.3.4 Data Dictionary	11
3 Project Management	11
3.1 Cost Estimations	11
3.1.1 Functional Point Estimation	11
3.1.2 Efforts	12
3.2 Risk Table	13

3.3 Timeline.....	14
.....	15
4 Design Engineering.....	15
4.1 Architectural Design.....	15
4.2 data design.....	15
4.3 Component Level Design	15
4.3.1CODE	15
4.3.2 OUTPUT:.....	17
4.3.2 FLOW GRAPH	17
5 Testing.....	18
6 References.....	18

Problem Statement

The main aim of this project named “Face Detection and Recognition” is to use face as identity, instead of alphanumeric identity or passwords, for purpose of accessing his/her personal data. A general statement of face recognition problem can be formulated as follows: Given still or live image of a scene, identify or verify person in the scene using a stored database of faces. Available collateral information such as race, age, gender, and facial expression may be used in narrowing the search (enhancing recognition). In identification problems, the input to the system is an unknown face, and the system reports back the decided identity from a database of known individuals, whereas in verification problems, the system needs to confirm or reject the claimed identity of the input face.

The strong demand for user friendly systems which can secure our assets and protect our privacy without losing our identity in a sea of numbers is obvious. At present, one needs:

1. a PIN to get cash from an ATM,
2. a password for a computer,
3. a dozen others to access the internet services
4. Alphanumeric id

Although extremely reliable methods of biometric personal identification exist, e.g., Fingerprint analysis and retinal or iris scans, these methods have yet to gain acceptance by the general population and also need special hardware which accounts for high initial cost. But now a days most or almost every premises are installed with CCTV cameras, have web-cam, or every mobile device comes with camera. A personal identification system based on analysis of frontal or profile images of the face is non-intrusive and therefore user friendly.

1 Introduction

The Software Requirements Specification is designed to document and describe the agreement between the customer and the developer regarding the specification of the software product request. Its primary purpose is to provide a clear and descriptive “statement of user requirements” that can be used as a reference in further development of the software system. This document is broken into a number of sections used to logically separate the software requirements into easily referenced parts.

This Software Requirements Specification aims to describe the Functionality, External Interfaces, Attributes and Design Constraints [4] imposed on Implementation of the software system described throughout the rest of the document. Throughout the description of the software system, the language and terminology used should unambiguous and consistent throughout the document.

1.1 Purpose

The purpose of this document is to give a detailed description of the requirements for the “FACE DETECTION AND RECOGNITION SYSTEM” software. It will illustrate the purpose and complete declaration for the development of system. It will also explain system constraints, interface and interactions with other external applications. This document is primarily intended to be proposed to a customer for its approval and a reference for developing the first version of the system for the development team.

1.2 scope

The software system being produced is called FACE DETECTION AND RECOGNITION SYSTEM for user friendly systems which can secure our assets and protect our privacy without losing our identity in a sea of numbers is obvious. This system is largely cross-platform and is available to anyone. The system will be run on a central server with each user having a remote user interface through a web browser to interact with it. An administrator also uses the web-portal in order to administer

the system and keep the information accurate. The FACE DETECTION AND RECOGNITION SYSTEM will allow any user to login an account/access information using face. The system will allow users to browse, access, and edit their information.

The FACE DETECTION AND RECOGNITION SYSTEM will have numerous constraints on what it can do.

1.3 Definitions, acronyms, and abbreviations

TERMS	DEFINITION
User	Someone who interact with the software application to get some information
Administration	System administrator who is given specific permission for controlling and managing the System.
FDRS	FACE DETECTION AND RECOGNITION SYSTEM
DFD	Data Flow Diagram

1.4 overview

A FDRS is a technology capable of identifying or verifying a person from a live stream from camera. There are multiples methods in which facial recognition systems work, but in general, they work by comparing selected facial features from given image with faces within a database.

While initially a form of computer application, it has seen wider uses in recent times on mobile platforms and in other forms of technology, such as robotics.

Recently, it has also become popular as a commercial identification and marketing tool.

1.5 Process Model

We choose Waterfall model because of the following reasons:

1. It is easy to manage due to the rigidity of the model – each phase has specific deliverables and a review process.
2. In this model phases are processed and completed one at a time. Phases do not overlap
3. Waterfall model works well for smaller projects where requirements are very well understood.
4. The requirements are fixed and work can proceed to completion in a linear manner.

2 Software Requirement Specification

2.1 Overall description

This section will give an overview of the whole system. The system will be explained in its context to show how the system interacts with other systems and introduce the basic functionality of it. It will also describe what type of stakeholders that will use the system and what functionality is available for each type. At last, the constraints and assumptions for the system will be presented.

2.1.1 product perspective

The system we are developing is independent and totally self-contained. Paper work is removed and due to facial recognition reduces risk of losing or forgetting identity/pins. Due to automation lot of time is removed.

2.1.2 Product Function

The system will allow access only to authorized users with specified roles (Administrator or Employee). Depending upon the user's role he/she will be able to access only specific modules of the system.

A summary of the major functions that the system will perform:-

- A login facility for enabling only authorized access to the system.
- User will be able to view/modify his/her information.
- User will be able to view/add information regarding oneself.

2.1.2 User characteristics

The FDRS requires user to not possess any specific experience and hence it can be used by any naïve user.

2.1.3 General Constraints

Several factors limit the effectiveness of facial-recognition technology giving its constraints:

1.IMAGE QUALITY:

Image quality affects how well facial-recognition algorithms work.

2. IMAGE SIZE:

When a face-detection algorithm finds a face in an image or in a still from a video capture, the relative size of that face compared with the enrolled image size affects how well the face will be recognized.

3. FACE ANGLE:

The relative angle of the target's face influences the recognition score profoundly. When a face is enrolled in the recognition software, usually multiple angles are used (profile, frontal and 45-degree are common).

2.1.4 Assumptions and Dependencies

It is assumed that user database is already existing and new entry by the user is not scope of this project functionality.

2.2 Specific requirements

This section contains all of the functional and quality requirements of the system. It gives a detailed description of the system and all its features

3.1 User interfaces

The application will have a simple and click execute interface(where user clicks on application which takes live stream to recognize and give details).

3.2 Hardware interfaces

Now a days most or almost every premises are installed with CCTV cameras, have webcam, or every mobile device comes with camera. Therefore our system doesn't have any rigid hardware interface

3.3 Software interfaces

This is a simple application software which runs on every operating system provided with python support and requires permission to use camera component.

3.4 Software Requirements

- Any OS.

- sqlite database .
- Python(2.7) official software.
- Opencv(2.4.10)

3.5 Hardware Requirements

1. Any computer device(minimum with RAM-2GB,Processor speed-1.7).
2. Camera(Minimum 8 MP with f/2.2)

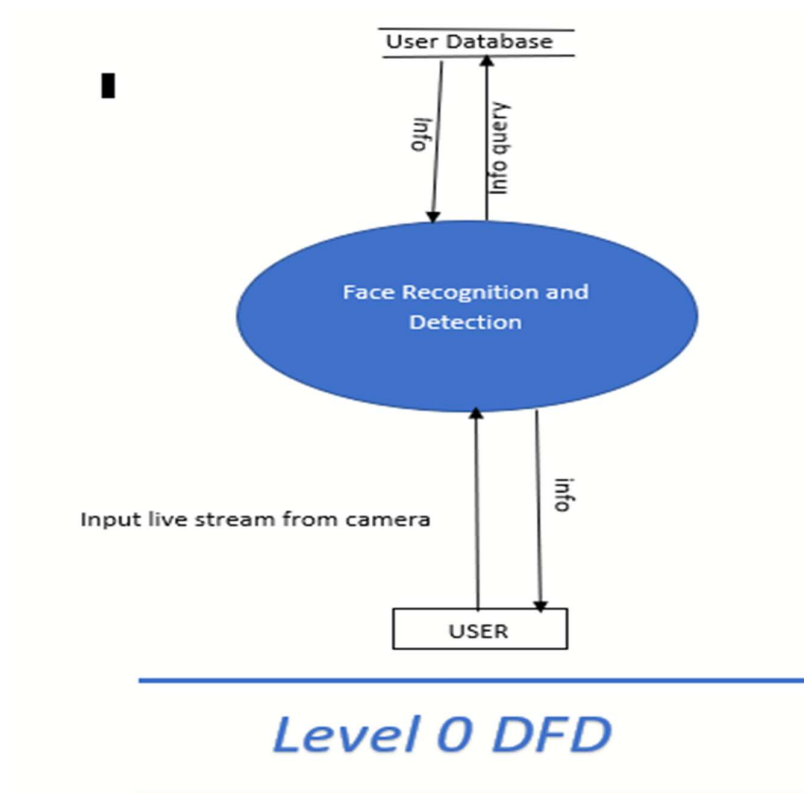
3.6 Modules

A module is a small part of our project.This plays a very important role in the project and in the coding concept.

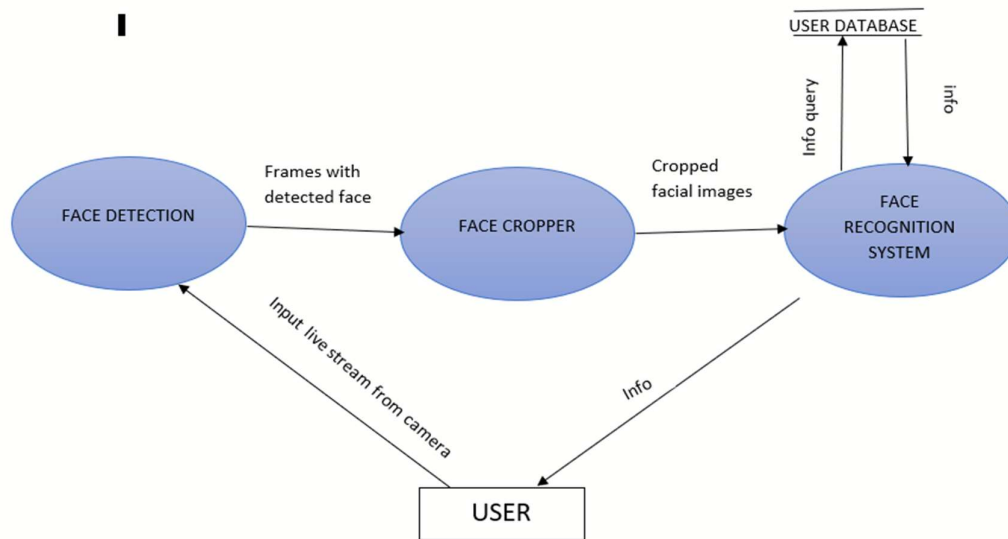
- FACE DETECTION:-It is used to detect whether a given live stream have face or not.
- FACE CROPPER:-This module crops facial images from detected facial images(passed by above module).
- FACE RECOGNIZER:-This module searches in database using cropped facial images(passed by above module).

2.3 Data flow Diagram

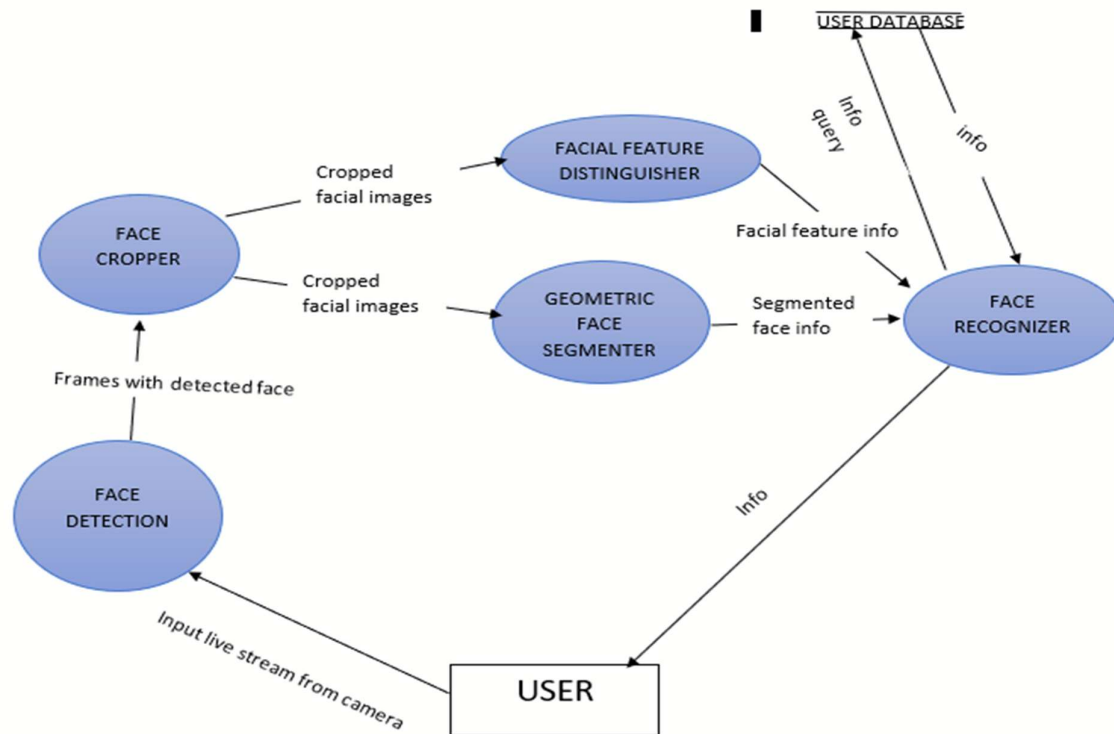
2.3.1 Level 0 DFD



2.3.2 Level 1 DFD



2.3.2 DFD level 2



2.3.4 Data Dictionary

Data	Description
Info	User name+Roll no
Info query	DBMS query string
Input live stream from camera	Input live stream from camera
Frames with detected face	Images containing face
Cropped facial images	Images with face(Background removed)
Facial feature info	Facial feature info to improve searching
Segmented face info	Segmented face info(Coordinates)

3 Project Management

3.1 Cost Estimations

3.1.1 Functional Point Estimation

S.no	<u>FACTOR</u>	GRADE VALUE
1	Does the system require reliable backup and recovery?	5
2	Are specialized data communications required to transfer information to or from the application?	1
3	Are there distributed processing functions?	3
4	Is performance critical?	3
5	Will the system run in an existing, heavily utilized operational environment?	0
6	Does the system require on-line data entry?	0
7	Does the on-line data entry require the input transaction to be built over multiple screens or operations?	1
8	Are the ILFs updated online?	1
9	Are the inputs, outputs, files, or inquiries complex?	2
10	Is the internal processing complex?	5
11	Is the code designed to be reusable?	5
12	Are conversions and installations included in the design?	2

13	Is the system designed for multiple installations in different organizations?	5
14	Is the application design to facilitate change and for ease of use by the user?	5

VALUE ADJUSTMENT FACTORS, $\Sigma f_i = 38$

VALUE ADJUSTMENT FACTORS

INFORMATION DOMAIN VALUE	EST. COUNT	WEIGHING FACTOR	WEIGHING COUNT
1. EXTERNAL INPUT	1	3	3
2. EXTERNAL OUTPUT	1	4	4
3. EXTERNAL INQUIRIES	1	3	3
4. NUMBER OF LOGICAL FILES	1	7	7
5. EXTERNAL INTERFACE FILES	0	5	0
TOTAL	17		

COMPUTING FUNCTION POINTS:

$$FP_{\text{estimated}} = \text{Count Total} * [0.65 + 0.01 * \Sigma(F_i)] = 17 * (0.65 + 0.01 * 38)$$

$$= 17.51$$

3.1.2 Efforts

Assuming that Average productivity for such a system is 6.5FP/pm i.e. 6.5 functional points per person month, then the estimates are:

$$\text{Effort} = \text{Total Functional Points} / \text{Average Productivity}$$

$$= 17.51 / 6.5 = 2.69 \text{ pm}$$

i.e. 2.69 person month

COST ESTIMATION

Assuming that the cost is ₹ 5000/pm i.e. ₹ 5000 per person month

Cost = Cost per person month * Effort

$$= ₹ 5000 * 2.69 = ₹ 13469$$

3.2 Risk Table

Risks	Category	Probability	Impact	Mitigation
Quality not maintained	DE	40%	3	Take up steps to maintain quality at each stage of development.
Size estimates may be low	PS	30%	2	Past experiences must be considered and a similar task on a smaller scale may be attempted
Requirements not properly documented and understood	CU	50%	1	Regular interaction with the customer and getting the requirements verified before finalising them
Delivery deadline will be tightened	BU	30%	2	Review the progress from time to time and take appropriate steps to keep up with the schedule
Lack of skill	ST	30%	2	External resources might help
Building the wrong product	CU	10%	1	Early and continuous validation is critically important. You need to establish a clear vision and solid justification for the product.

1-Catastrophic

2-Critical

3-Marginal

4-Negligible

PD-Process Definition

ST-Staff-size and experience

DE-Development Environment

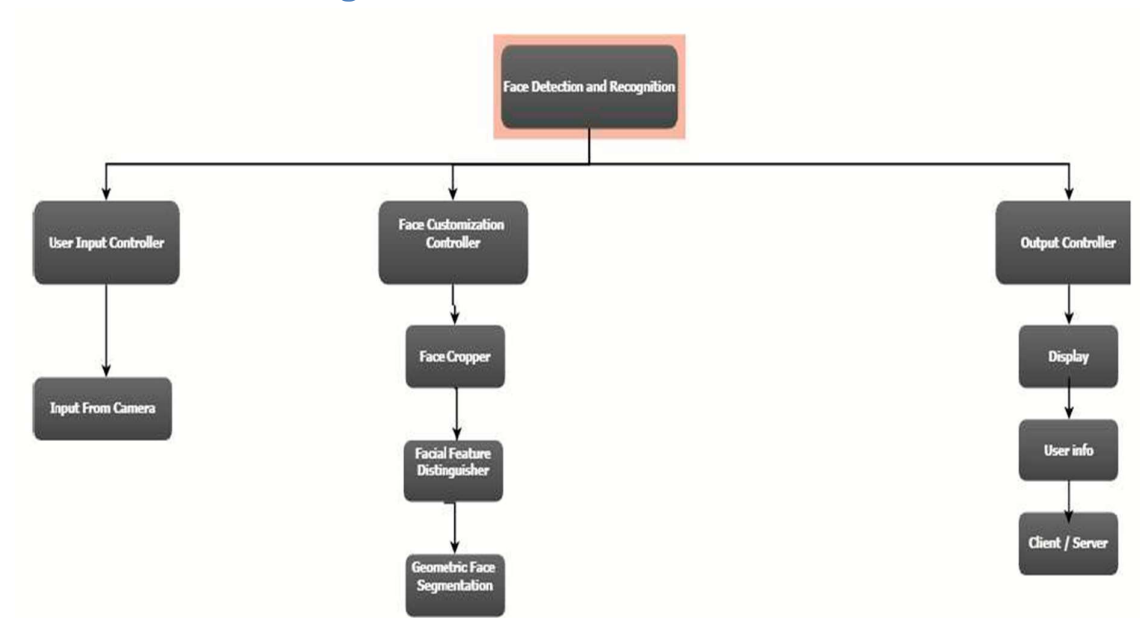
CU-Customer Characteristics

BU-Business Impact

PS-Product Size

4 Design Engineering

4.1 Architectural Design



4.2 Data Design

People

<u>ID</u>	<u>INT</u>
<u>Name</u>	<u>VARCHAR(30)</u>

4.3 Component Level Design

4.3.1 CODE

```
import cv2,os
```

```
import numpy as np
```

```
from PIL import Image
```

```
import pickle
```

```
import sqlite3
```

```

faceDetect=cv2.CascadeClassifier('haarcascade_frontalface_default.xml');

cam=cv2.VideoCapture(0);

recognizer=cv2.createLBPHFaceRecognizer();

recognizer.load('trainer.yml')

font=cv2.cv.InitFont(cv2.cv.CV_FONT_HERSHEY_COMPLEX,0.4,1,0,1)

def getProfile(id):

    conn=sqlite3.connect("FaceBase.db")

    cmd="SELECT * FROM People WHERE ID="+str(id)

    cursor=conn.execute(cmd)

    profile=None

    for row in cursor:

        profile=row

    conn.close()

    return profile

while(True):

    ret,img=cam.read();

    gray=cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)

    faces=faceDetect.detectMultiScale(gray,1.3,5);

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

        cv2.rectangle(img,(x,y),(x+w,y+h),(0,255,0),2)

        id,conf=recognizer.predict(gray[y:y+h,x:x+w])

        profile=getProfile(id)

        if(profile!=None):

            cv2.cv.PutText(cv2.cv.fromarray(img),"ID : "+str(profile[0]),(x,y+h+20),font,(0,255,0));

            cv2.cv.PutText(cv2.cv.fromarray(img),"Name : "+str(profile[1]),(x,y+h+45),font,(0,255,0));

        cv2.imshow("Face detection and recognize",img);

        if(cv2.waitKey(1)==ord('q')):

            break;

    cam.release()

cv2.destroyAllWindows()

```

1

12

13

14

15

16

2

3

4

5

6

7

8

9

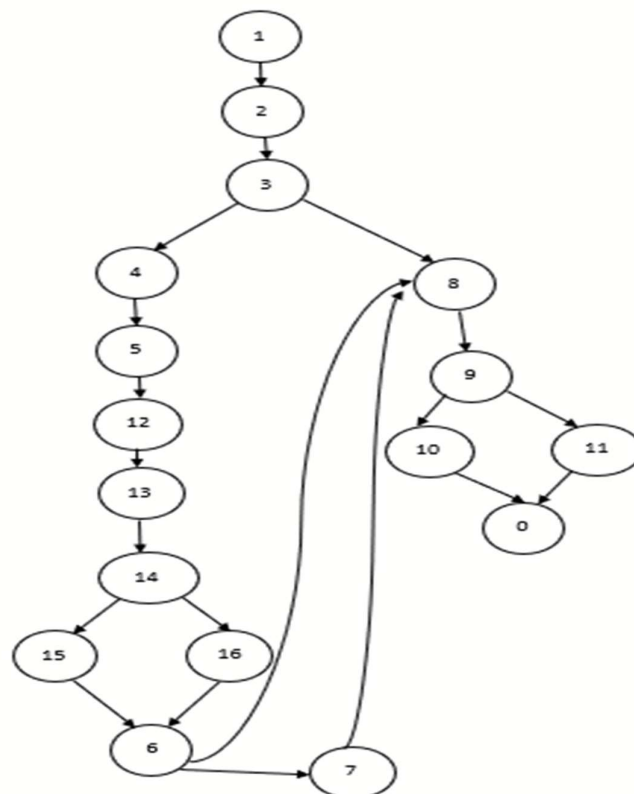
10

11

4.3.2 OUTPUT:



4.3.2 FLOW GRAPH



CALCULATING CYCLOMATIC COMPLEXITY

P = 4

E = 20

N = 17

R=5

WHERE P = NO OF PREDICATE NODE ,E = NO. OF EDGES, N = NO. OF NODES,R = NO. OF REGIONS

CYCLOMATIC COMPLEXITY = $E - N + 2 = 20 - 17 + 2 = 5$

OR

CYCLOMATIC COMPLEXITY = $P + 1 = 4 + 1 = 5$

OR

CYCLOMATIC COMPLEXITY = $R = 5$

INDEPENDENT PATHS

Path 1 – 1,2,3,4,5,12,13,14,15,6,8,9,10,0

Path 2 – 1,2,3,4,5,12,13,14,15,6,7,8,9,10,11,0

Path 3 – 1,2,3,4,5,12,13,14,16,6,8,9,10,11,0

Path 4 – 1,2,3,8,9,10,0

Path 5 – 1,2,3,8,9,11,0

5 Testing

TEST CASES

S.NO.	TEST SUITE	TEST CASE ID	TEST CASE DESCRIPTION	INPUT VALUE	OUTPUT	EXPECTED OUTPUT	RESULT	REMARKS
1	Input Suite	T1	User Is Not Front of Camera	Face1	Red Circle	Red Circle	Not Processed	Pass
2	Input Suite	T2	Picture Is Not Clear	Face2	Coordinates Not Defined	Coordinate Not Defined	Show Error Message	Pass
3	Input Suite	T3	Two objects in one Frame	Face3	Both Object Save With Same Id	Two Object Are There Error Message	Wrong Output	Fail
4	Input Suite	T4	Object Is Not Stable	Face4	Coordinates Not Defined	Coordinate Not Defined	Show Error Message	Pass
5	Input Suite	T5	Picture Is in Database Or Not	Face5	Picture Is Not in Database	Picture is not In Database	Show Error Message	Pass
6	Hardware Suite	T6	Webcam Requirement	Face6	Incompatible Webcam	Incompatible Webcam	Show Error Message	Pass
7	Hardware Suite	T7	Ram Requirement	Face7	Incompatible System	Incompatible System	Show Error Message	Pass
8	Hardware Suite	T8	Operating System	Face8	This System Doesn't Have Python	This System Doesn't Have Python	Show Error Message	Pass

6 References

- Software Engineering: A Practitioner's Approach by Roger S.Pressman
- An Integrated Approach to Software Engineering by P.Jalote
- en.wikipedia.org
- Introduction to Software Engineering by K.K Aggarwal and Yogesh Singh.
- P.Jalote, an Integrated Approach to Software Engineering, 3rd Edition.