MUHAMMAD JUNAID KHAN
(01-134141-079)
MUHAMMAD HAMZA IMTIAZ
(01-134141-072)

Age Invariant Face Recognition

Bachelor of Science in Computer Science

Supervisor: Momina Moetesum

Department of Computer Science
Bahria University, Islamabad

November 2017

Certificate

We accept the work contained in the report titled "Age Invariant Face Recognition", written by Mr. Muhammad Junaid Khan AND Mr. Muhammad Hamza Imtiaz as a confirmation to the required standard for the partial fulfillment of the degree of Bachelor of Science in Computer Science.

Approved by . . . :

Supervisor: Momina Moetesum (Senior Lecturer)

_____

Internal Examiner: ()

_____

External Examiner: ()

_____

Project Coordinator: Dr. Sumaira Kausar (Assistant Professor)

_____

Head of the Department: Dr. Faisal Bashir (Associate Professor)

_____

November 20^{st}, 2017

Abstract

Automated face recognition is a popular area of research due to its wide applicability. However, there are several challenges which obscure performance of such systems. These include pose, lightning, facial expressions, hair and accessories. Recently there is another aspect i.e. 'Age invariance', which is attracting researchers and developers. The main objective of this project is to build a desktop application which can detect same face across different age groups. By applying image analysis and recognition techniques on various facial features i.e. (nose, eyes, mouth etc.), we have achieved considerable recognition results. Training was done on a database of 1600 images is used consisting of celebrity images across various age groups.

Acknowledgments

We are thankful to our respected supervisor Mrs Momina Moetesum who helped us shape this project out in its proper form. From the initiation of this project she always the source of guidance and lent us our support on every problem we faced. We would like to express our utmost gratitude for her support in completing this project. We also like to extend our sincere thanks to Dr. Imran Siddiqui for his cordial support.

MUHAMMAD JUNAID KHAN & MUHAMMAD HAMZA IMTIAZ
Islamabad, Pakistan

November 2017

"We think someone else, someone smarter than us,
someone more capable, someone with more resources will solve that problem.
But there isn't anyone else."

Regina Dugan

Contents

1 Introduction **1**
- 1.1 Overview . . . 1
- 1.2 Problem Description . . . 2
- 1.3 Project Objectives . . . 2
- 1.4 Proposed Methodology . . . 2
- 1.5 Project Scope . . . 2

2 Literature Review **5**
- 2.1 Face Recognition Techniques . . . 5
  - 2.1.1 Traditional Approach . . . 5
  - 2.1.2 3-Dimensional Face Recognition . . . 5
  - 2.1.3 Skin Texture Analysis . . . 6
  - 2.1.4 Thermal Cameras . . . 6
  - 2.1.5 Limitation of Existing Techniques . . . 6

3 Requirement Specifications **9**
- 3.1 Existing System Overview . . . 9
- 3.2 Proposed System Overview . . . 9
- 3.3 Requirement Specification . . . 10
  - 3.3.1 Functional Requirements . . . 10
  - 3.3.2 Non-Functional Requirements . . . 10
- 3.4 User and Interface Requirements . . . 11
  - 3.4.1 User Characteristics . . . 11
- 3.5 Hardware and Software Requirements . . . 11
- 3.6 Use Case Diagrams . . . 11

4 Design **19**
- 4.1 System Architecture . . . 19
- 4.2 System Design . . . 19
  - 4.2.1 Design Constraints . . . 19
  - 4.2.2 High Level Design . . . 20
  - 4.2.3 System Sequence Diagram . . . 21
  - 4.2.4 Low Level Design . . . 21
  - 4.2.5 Graphical User Interface Design . . . 22

5 System Implementation **25**
5.1 Tools Used . . . 25
5.1.1 Visual Studio . . . 25
5.1.2 EmguCV . . . 25
5.1.3 C-sharp Language . . . 26
5.2 Methodology . . . 26
5.2.1 Proposed Algorithm . . . 26
5.3 Dataset . . . 26
5.4 Techniques Applied . . . 28
5.4.1 Preprocessing . . . 28
5.4.2 Face Detection . . . 28
5.4.3 Face Region Segmentation . . . 29
5.4.4 Matching . . . 29
5.4.5 Retrieval . . . 30

6 System Testing and Evaluation **31**
6.1 Software Testing Techniques . . . 31
6.1.1 Function Testing . . . 31
6.1.2 Performance Testing . . . 32
6.1.3 Acceptance Testing . . . 33
6.2 Test Cases . . . 33

7 Conclusions **37**

References **39**

List of Figures

1.1 Image of a Person across Two Age Groups . . . 1
1.2 Flow Chart of Proposed Methodology . . . 3

2.1 Geometric Approach [1] . . . 6
2.2 3D-recognition [2] . . . 7
2.3 Skin Texture [3] . . . 7
2.4 Thermal Cameras [4] . . . 8

3.1 Main usecases . . . 12
3.2 Usecase-1 . . . 13
3.3 Usecase-2 . . . 13
3.4 Usecase-3 . . . 14
3.5 Usecase-4 . . . 14
3.6 Usecase-5 . . . 15
3.7 Usecase-6 . . . 15

4.1 Overall System Architecture . . . 20
4.2 High Level Design . . . 20
4.3 Sequence Diagram . . . 21
4.4 System Module Diagram . . . 22
4.5 First Screen Shot of GUI . . . 22
4.6 Second Screen Shot of GUI . . . 23
4.7 Third Screen Shot of GUI . . . 23

5.1 Complete Methodology Flow Chart . . . 27
5.2 Sample Images from Dataset [5] . . . 27
5.3 RGB to Grayscale Conversion . . . 28
5.4 Image Enhancement using Histogram Equilization . . . 28
5.5 Haar Cascade for Face Detection [6] . . . 29
5.6 Haar Cascade for Face Component Detection [6] . . . 29
5.7 Face Component Segmentation . . . 29
5.8 Top Three Closest Matching Retrieved Images . . . 30

List of Tables

3.1 UseCase-1 16
3.2 UseCase-2 16
3.3 UseCase-3 16
3.4 UseCase-4 16
3.5 UseCase-5 17
3.6 UseCase-6 17

6.1 TestCase-1 34
6.2 TestCase-2 34
6.3 TestCase-3 34
6.4 TestCase-4 34
6.5 TestCase-5 34
6.6 TestCase-6 35
6.7 TestCase-7 35
6.8 TestCase-8 35

Chapter 1

Introduction

1.1 Overview

Face detection and recognition techniques have enjoyed tremendous research interest over the last three decades [7]. Significant research has been carried out on face detection and recognition targeting a wide variety of applications. The key application areas include biometric systems, surveillance and image retrieval [8]. While the traditional face recognition systems have been is practice for a long time, an emerging research area is age invariant face recognition [9] where the challenge is to recognize an individual in scenarios where training and test images have been taken at different ages of the individual. This age invariant face recognition makes the subject of our study and is particularly targeted towards 'lost and found' cases of children. Sample images of an individual at two different ages are represented in Figure 1.1.

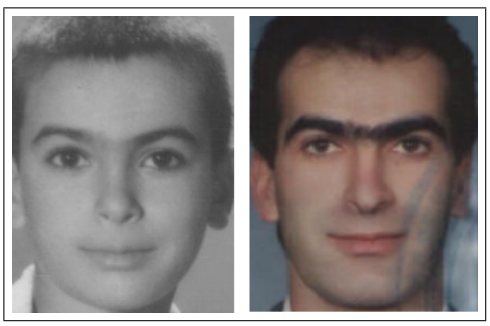

Figure 1.1: Image of a Person across Two Age Groups

1.2 Problem Description

Face recognition systems have been developed for a long time and report high classification rates. All these traditional systems do not take into account the facial features that vary with age. In some cases, for instance, the lost children, age invariant facial recognition is desirable and conventional recognition systems are not likely to work. The proposed system is intended to extract age invariant features from facial images and recognize an individual irrespective of the age.

1.3 Project Objectives

The primary objective of this project is to recognize individuals using frontal facial images taken at different ages. System objectives are as follows:

- Detection of face region from images
- Segmentation of facial features from images
- Feature extraction from each facial region
- Feature matching across various ages

1.4 Proposed Methodology

The overall block diagram of the proposed system is presented in Figure 5.1. Preprocessing is applied to the input facial image and regions of interest including eyes, mouth and nose are localized. Feature extraction is applied on each region of interest on the facial image. Features extracted from all the images in training set are stored in a reference base. For classification, a direct comparison of query features with those stored in the database is carried out. The system retrieves closest matches and displays them.

1.5 Project Scope

It is not a real time recognition system and requires the images to be stored before hand. Frontal pose is required for complete capture of all facial features. Side pose can degrade the performance. Currently the system is trained and tested on a small portion of CACD dataset [5].

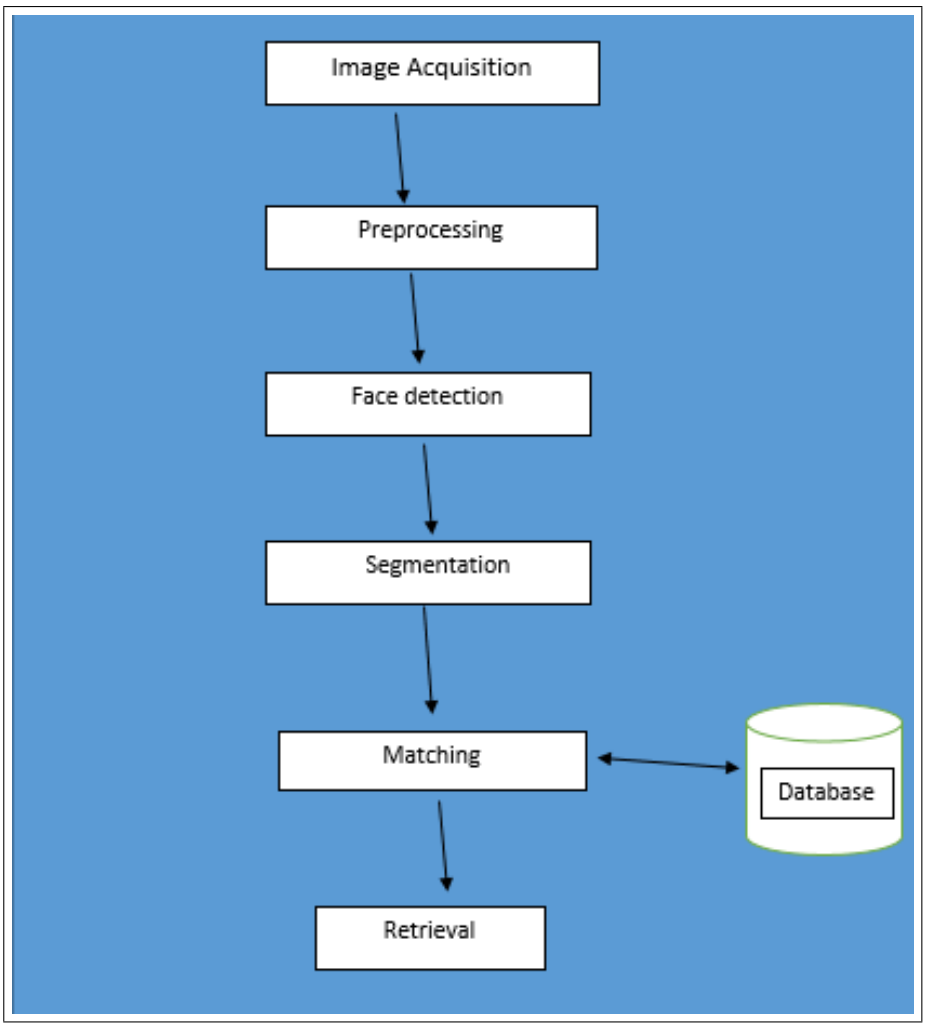

Figure 1.2: Flow Chart of Proposed Methodology

Chapter 2

Literature Review

In this chapter we will discuss the literature review of the face recognition systems.

2.1 Face Recognition Techniques

Face detection and recognition techniques have enjoyed tremendous research interest over the last three decades [10]. There are different techniques for face recognition some of which are discussed below:

2.1.1 Traditional Approach

Most commonly used approach is to identity facial features by extracting features from the person's face [1]. Points which are extracted from the face are size, shape of eyes, nose, cheekbones and jaws. These selected points are than matched with other images which have similar image points. The recognition algorithms are mainly divided into two:

- Geometric: These algorithms looks at distinguishing features on the face (Figure 2.1). However pose change may effect accuracy.
- Photometric: It is statistical approach which breaks the image into values and then compare it with the template, which eliminates the variances.

2.1.2 3-Dimensional Face Recognition

This is the new technique which claims that it will give more accurate results [2]. This technique uses 3D sensors to capture information about shape of face. Further this information is used to detect features on the surface of the face such as eyes sockets, nose etc. 3D features are not effected by illumination. 3D technique uses sensors which are great to capture a 3D image as shown in Figure 2.2. These 3D sensors using structure light onto face to detect the image. The technique uses 3 cameras, 1st camera take pic

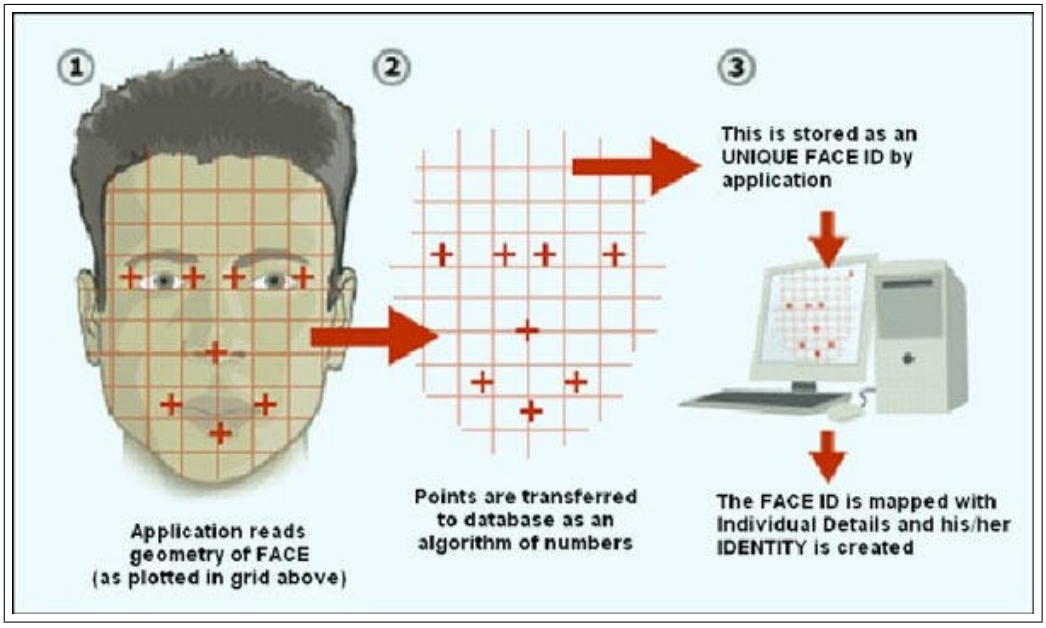

Figure 2.1: Geometric Approach [1]

from front 2nd from side and 3rd on at an angle these camera take image in real time and recognize the face. However 3D technique for matching could be sensitive for expressions.

2.1.3 Skin Texture Analysis

:-This technique uses skin details of the person captured in the digital photographs (Figure 2.3) [3]. Basically this technique turns the spots, unique lines pattern in the skin into mathematical space. Adding skin analysis improves the performance of face recognition up to 20-25 percent.

2.1.4 Thermal Cameras

In this face recognition technique [4], cameras only detect the shape of the head and ignore the items around the head such as hat, glasses, makeup (Figure 2.4) but problem with this approach is that its database is not much developed. It is well used in real life and operation scenarios and in that very time it builds a thermal face database. This technique uses low sensitive and low resolution ferro-electric sensors.

2.1.5 Limitation of Existing Techniques

In mentioned face recognizing applications, system can only recognize two images of a same person in same age groups, if age group of the same person varied, the application would have difficulty detecting the person. We have therefore tried to implement age invariant face recognition technique inspired by [8].

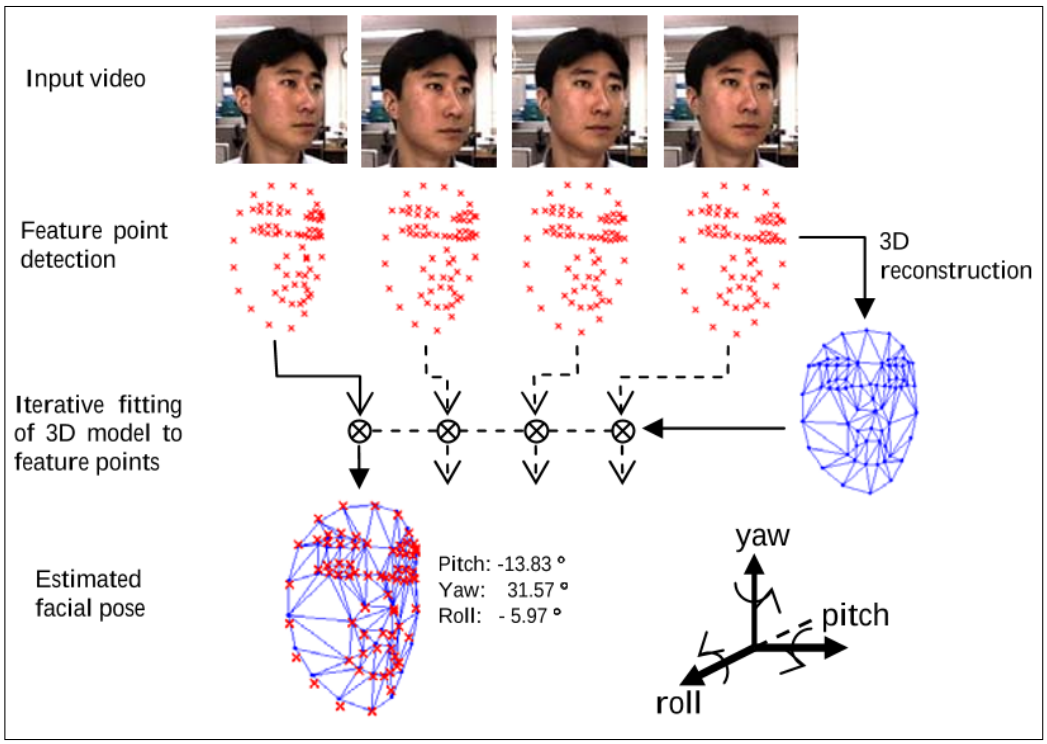

Figure 2.2: 3D-recognition [2]

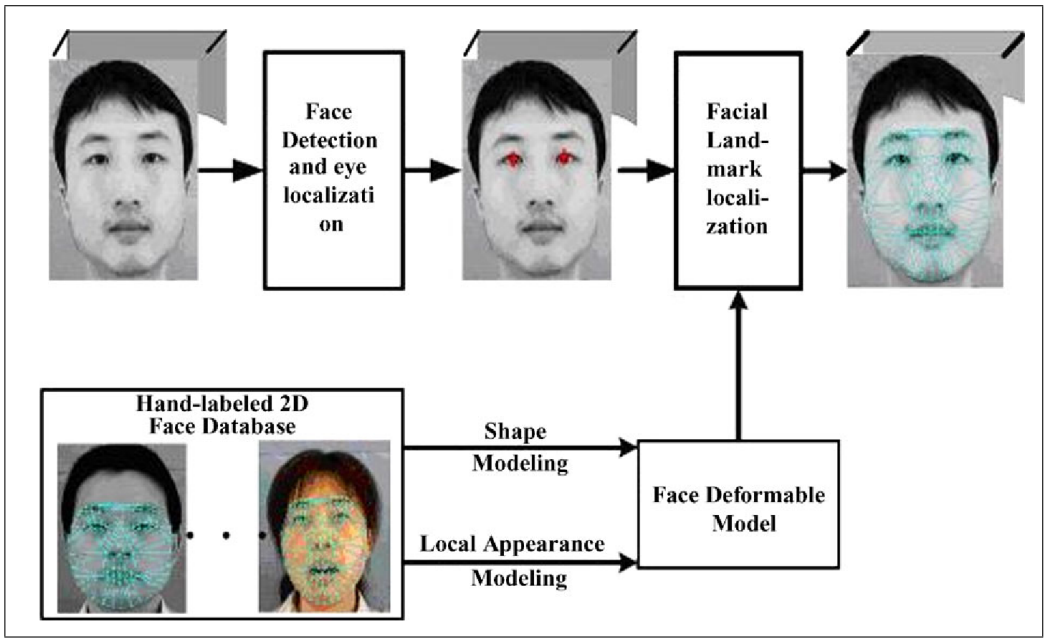

Figure 2.3: Skin Texture [3]

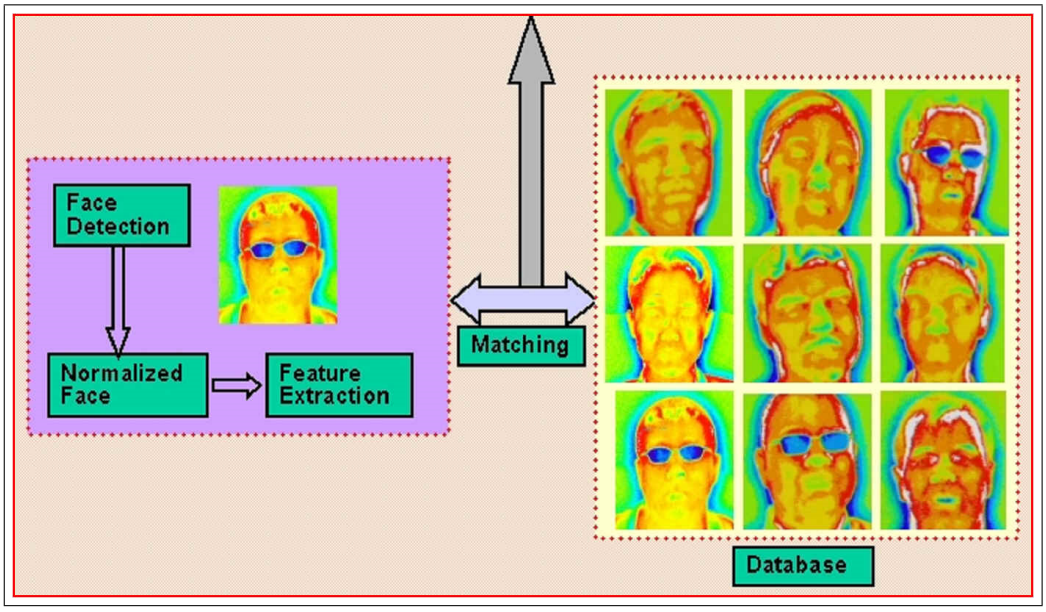

Figure 2.4: Thermal Cameras [4]

Chapter 3

Requirement Specifications

Requirement specification is done to outline the various functional and non-functional requirements of the system which needs to be developed. The functional and non-functional requirements of our proposed system are present in this section.

3.1 Existing System Overview

Although face recognition is a popular research area. There is a lot of work done in this field which has its own importance. Many existing applications are in the market which work on different techniques. Every application has its own benefits and limitations, but the most common limitations is all these applications is that they do not recognize the person if his/her two images are from different age groups. Our system will be able to recognize the person from the images of different age groups.

3.2 Proposed System Overview

The proposed system is a desktop application that could be used by police, forensic labs or any other place where face matching is required. The proposed application will use different image processing techniques for face recognition. When the application is started it will offer the user an option whether to add new face in the database or match the face with the existing database. Our application will not work on run time. For matching the face we must have the picture of that person in our database. User can match the person after detecting face in the image, after detection of face it will start detecting different features of the face to match the face with database. The application will use EmguCV libraries which are used for detection of the face and facial components. Our system will be very useful for security purposes and in police or in forensic field.

3.3 Requirement Specification

The main functionality of our application is to detect the face from the input image, extract different features of the face and save them and then match them with our database and show the results.

3.3.1 Functional Requirements

It defines the main functions of a system and its core components. The functional requirements of our system are given below:

- Application should be installed properly to avoid any failure.
- Images should be clear and of good quality to detect face easily.
- It will not work on run time, user must save the images in the database in order to recognize the person.
- On application start-up, user will choose whether to update the database or recognize the person.
- After choosing recognition process, application will detect the face, extract features from the face and then start doing.

3.3.2 Non-Functional Requirements

Following are the non-functional attributes of the system:

3.3.2.1 Reliability

Once application is installed in the system, user can use it any time to detect and recognize the face.

3.3.2.2 Response time

User will get the accurate recognition after matching the face with the database.

3.3.2.3 Usability

It is very simple and easy to use application, even user with very little knowledge of computer can easily use it.

3.3.2.4 Portability

Its design is easy so that we can enhance the capabilities of our system easily.

3.4 User and Interface Requirements

Interface should be easy for the users to understand. Selection of query images should be easy for any technical and non technical user.

3.4.1 User Characteristics

The user must have the following characteristics in order to use our application.

- They must have basic features knowledge about computer.
- They should know how to install our application.
- They should know how to run the application properly in order to recognize the person or add new image.

3.5 Hardware and Software Requirements

Windows operating system will be needed to operate the application. Optimal performance of the system will require at least 1GB of RAM, as we will be dealing with real time processing. Debugging information can be retrieved from visual studio and device diagnostic information can be read from debugger in visual studio while application is run on visual studio.

3.6 Use Case Diagrams

This section details following subset of all possible use cases of application. The subset includes following key use cases shown in Figures (3.1-3.7) with details specified in corresponding Tables (3.1-3.6).

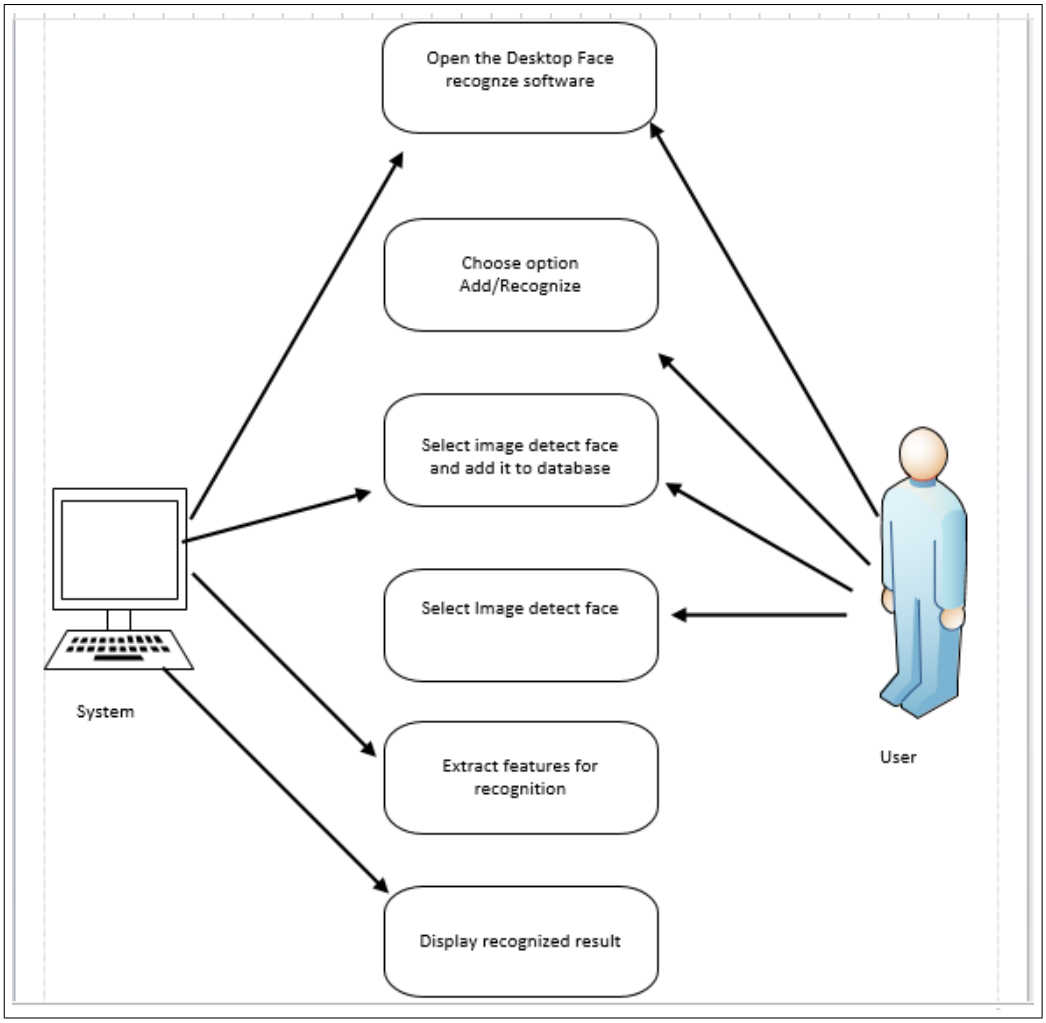

Figure 3.1: Main usecases

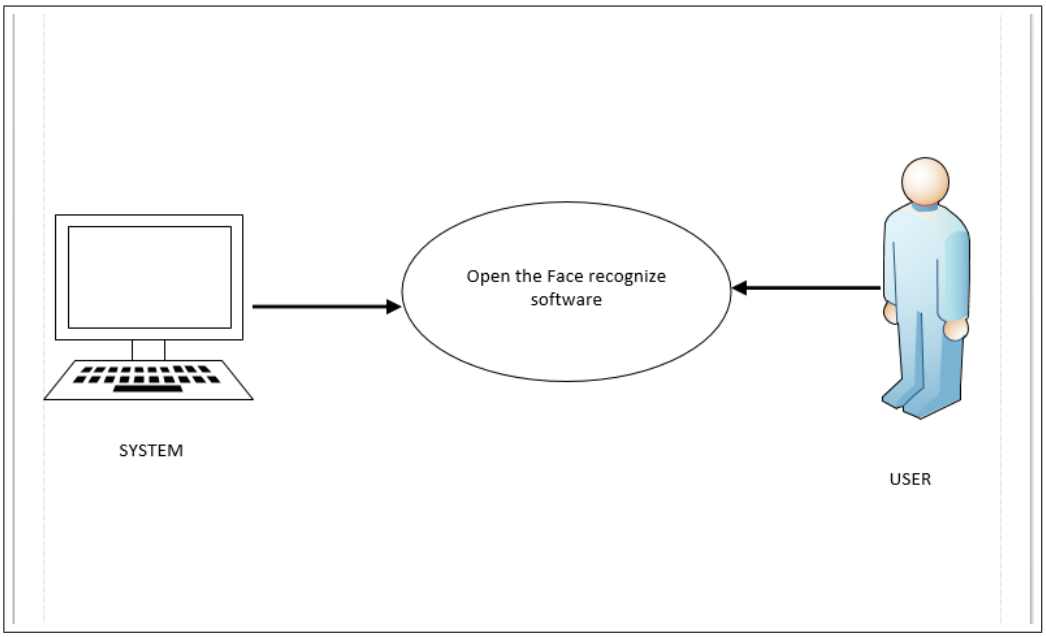

Figure 3.2: Usecase-1

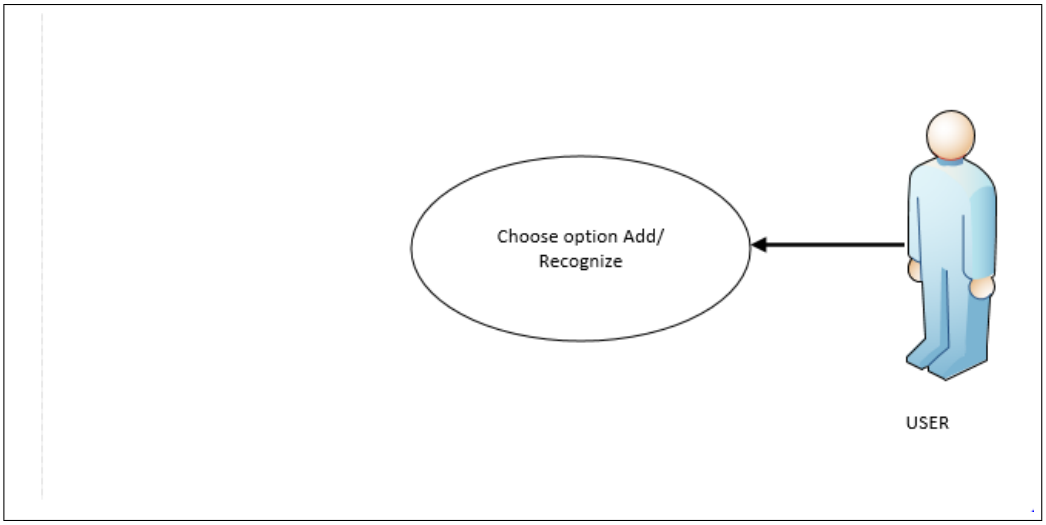

Figure 3.3: Usecase-2

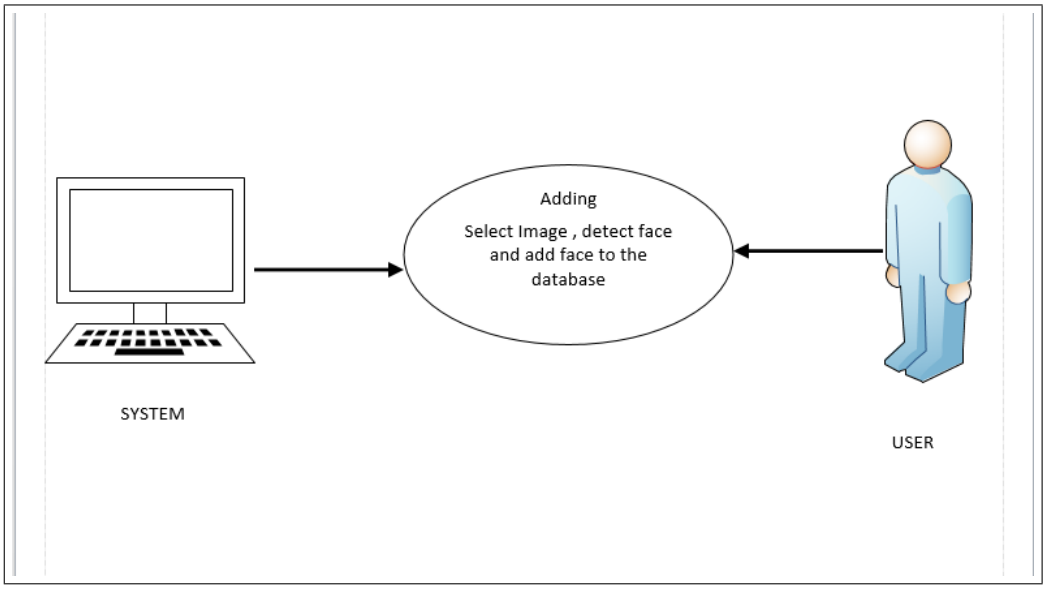

Figure 3.4: Usecase-3

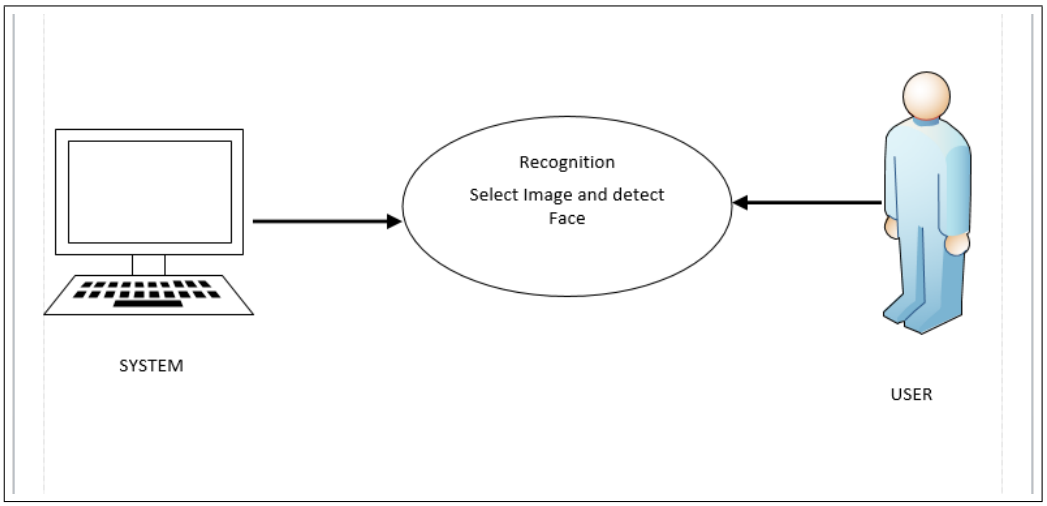

Figure 3.5: Usecase-4

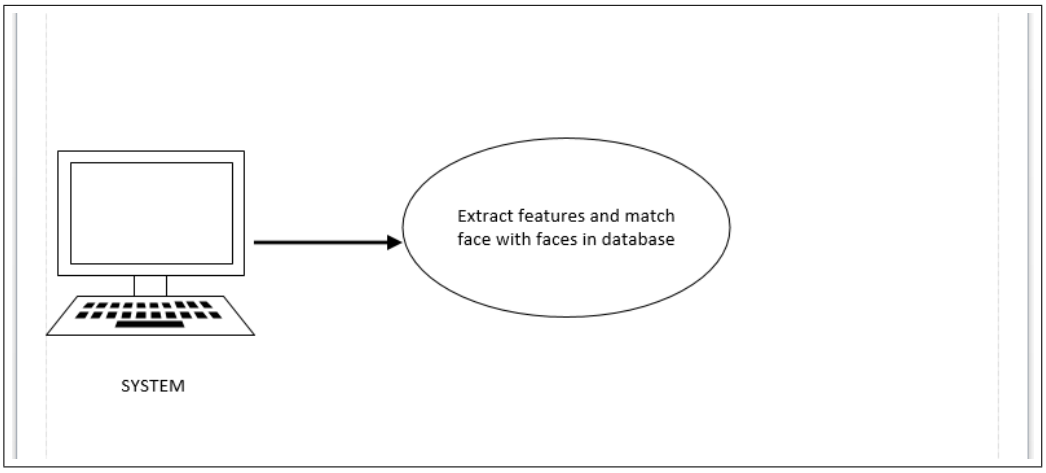

Figure 3.6: Usecase-5

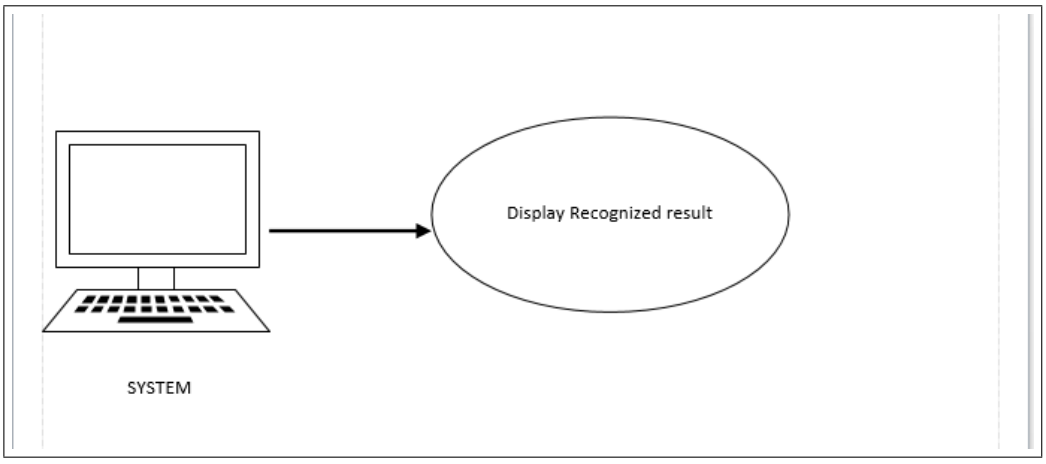

Figure 3.7: Usecase-6

Use Case ID	**UC-1**
Title	Start the application
Description	Start the application from the desktop system.
Primary Actor	System, User
Pre-Condition	Application should installed in desktop system correctly.
Post-Condition	Start the application correctly.
Success scenario	Application opens successfully.
Exception	Not open due to some error.
Assumptions	User should know how to open it correctly

Table 3.1: UseCase-1

Use Case ID	**UC-2**
Title	Select option
Description	Choose weather to add image or go for recognition.
Primary Actor	User
Pre-Condition	User should know What to choose.
Post-Condition	Selected option will open in new dialog box.
Success scenario	opens successfully.
Exception	Not open due to some error.
Assumptions	Choose correctly

Table 3.2: UseCase-2

Use Case ID	**UC-3**
Title	Adding image
Description	Select image, detect face and features and add it to the database.
Primary Actor	User
Pre-Condition	Should choose image having face.
Post-Condition	Detect face successfully.
Success scenario	Add face's image to the database.
Exception	Didn't detect any face.
Assumptions	User should know he select image having face.

Table 3.3: UseCase-3

Use Case ID	**UC-4**
Title	Recognition
Description	Select image, detect face and go to training section for recognition.
Primary Actor	System, User
Pre-Condition	Should select image having face..
Post-Condition	Detect face successfully.
Success scenario	Detect face and detected face to the training set for recognition.
Exception	Didn't detect face due to blur image or not clear image.
Assumptions	Should have image having face clearly.

Table 3.4: UseCase-4

Use Case ID	**UC-5**
Title	Extract Features
Description	Extracting features from the detected face.
Primary Actor	System
Pre-Condition	Should have clear facial image.
Post-Condition	Detect features successfully.
Success scenario	Detect features successfully.
Exception	Didn't detect any of the feature.
Assumptions	Should use image having clear face.

Table 3.5: UseCase-5

Use Case ID	**UC-6**
Title	Recognized result
Description	After checking from the dataset show the best 3 match.
Primary Actor	System
Pre-Condition	Should recognize person.
Post-Condition	Recognized person and show the result.
Success scenario	Show recognized person image with name.
Exception	Didn't recognized image.
Assumptions	Should have person's image in database.

Table 3.6: UseCase-6

Chapter 4

Design

Systems design is the process of defining the architecture, components, modules, interfaces, and data for a system to satisfy specified requirements. In this chapter, a detailed insight into to the design of the system will be provided. It will give an insight to the working of the system and the different processes which it is doing. It will also provide information about how the different processes are being carried out and how do these interact with each other on an intricate level.

4.1 System Architecture

The application devised at the end of this project is for windows based computers. The basic working of the system is described in the portion given below and the system architecture is given in Figure 4.1. The system has two components.

- The desktop application with the dataset.
- The image processing library i.e. Emgu CV.

4.2 System Design

Since the requirements of our system is very well known and very well clear so will prefer to use water fall model for the development of this system. During our design procedure we have used UML(Unified Modeling Languages) diagrams because it helps to keep a proper standard in system development process. Furthermore, it helps in understanding the system to developers to use it in any implementation platform.

4.2.1 Design Constraints

Although the GUI of our system is quite user friendly, however user must be computer literate to be able to use our system.

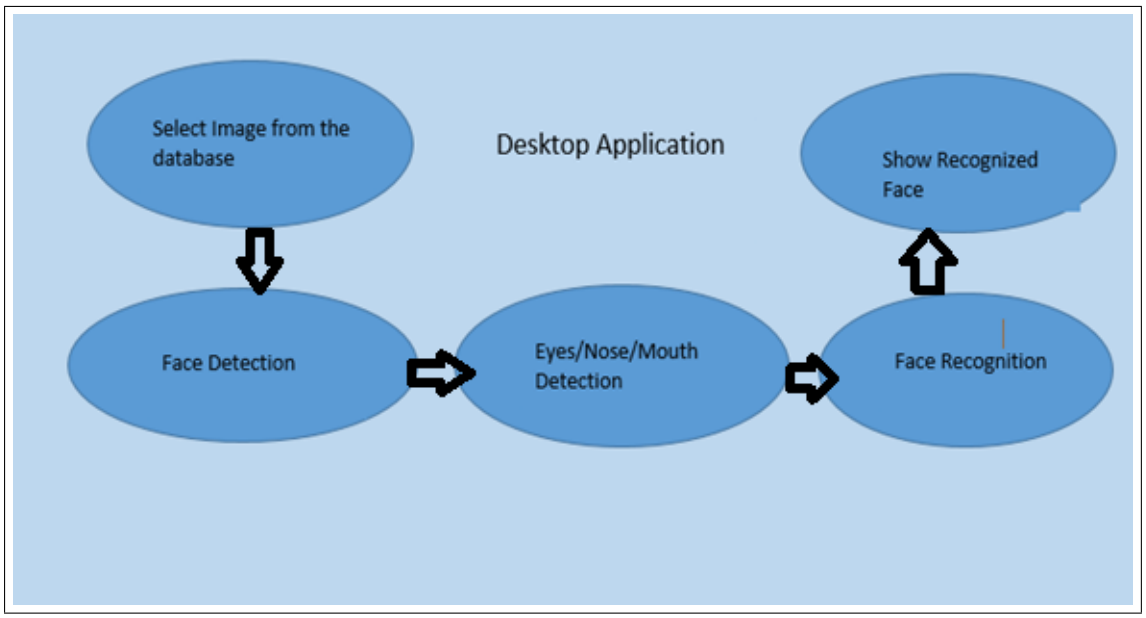

Figure 4.1: Overall System Architecture

4.2.2 High Level Design

The high level design is the overall design covering the system functionality. Figure 4.2 gives a high-level view of the system.

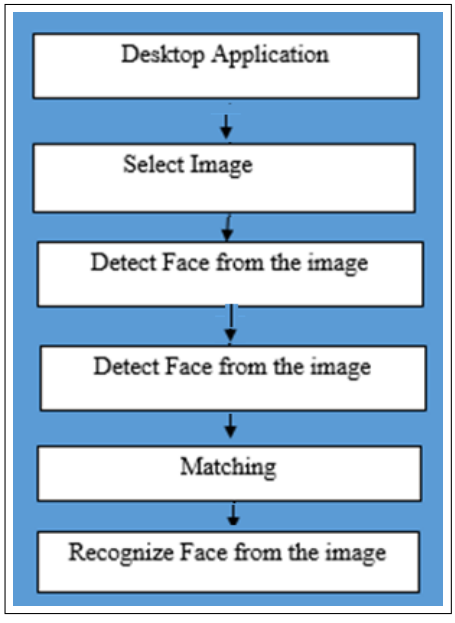

Figure 4.2: High Level Design

A query image can be selected from the dataset, then processing is done on the image and three closely matching output images. Applied methodology of the project is given below:

- Initialization
- Face detection
  - Eyes detection
  - Mouth detection
  - Nose detection
- Output images
- Optimization for built in database

4.2.3 System Sequence Diagram

A sequence diagram gives the interaction of the objects in a timed sequence. It depicts the classes involved in the system. The sequence diagram for the working of the system is given in Figure 4.3. User will start the application and select the image from the database. Image will then be placed in the image holder after clicking the recognizing button the selected image will be recognized and will be shown in second image box.

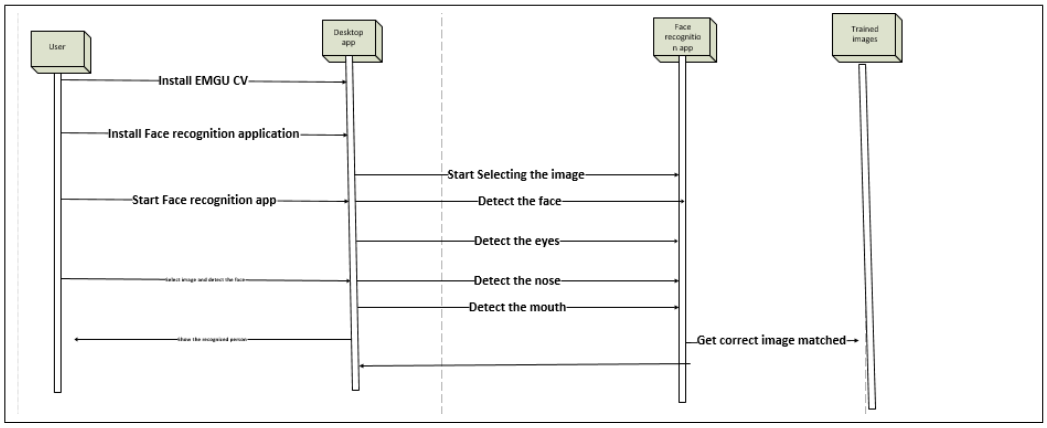

Figure 4.3: Sequence Diagram

4.2.4 Low Level Design

A low level design of the system is the component-level design which follows the step by step process of the component. The module diagram of the system is given in Figure 4.4.

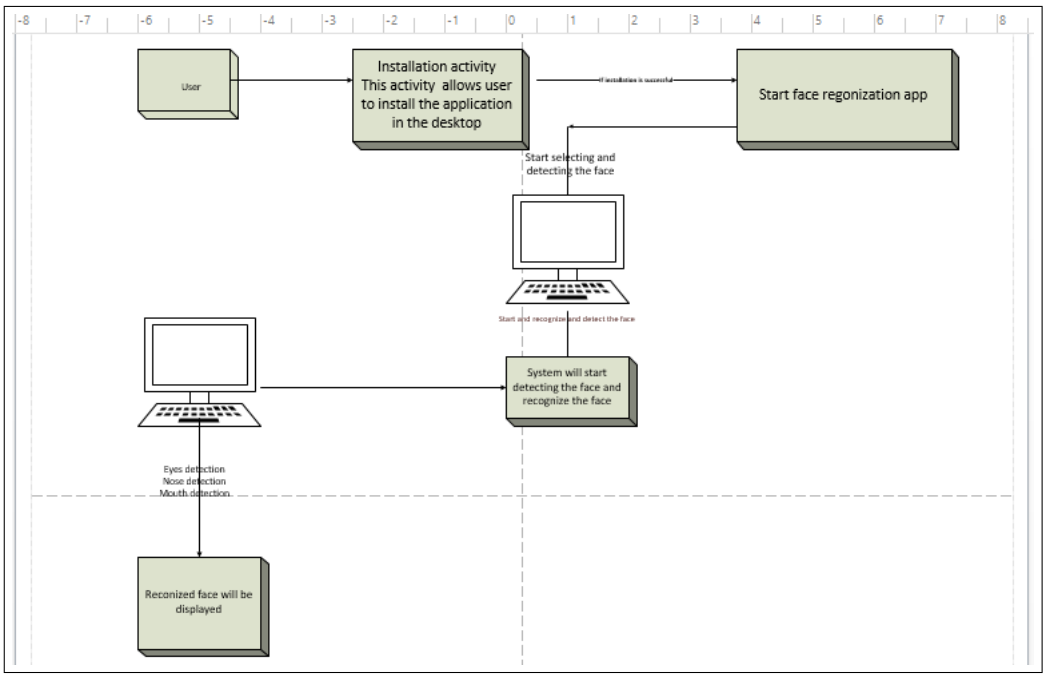

Figure 4.4: System Module Diagram

4.2.5 Graphical User Interface Design

The naming of the different GUI controls is easier so that the user can traverse through the different modules of the system with ease. Figure 4.5, Figure 4.6 and Figure4.7 show the various screen shots of our system's GUIs.

Figure 4.5: First Screen Shot of GUI

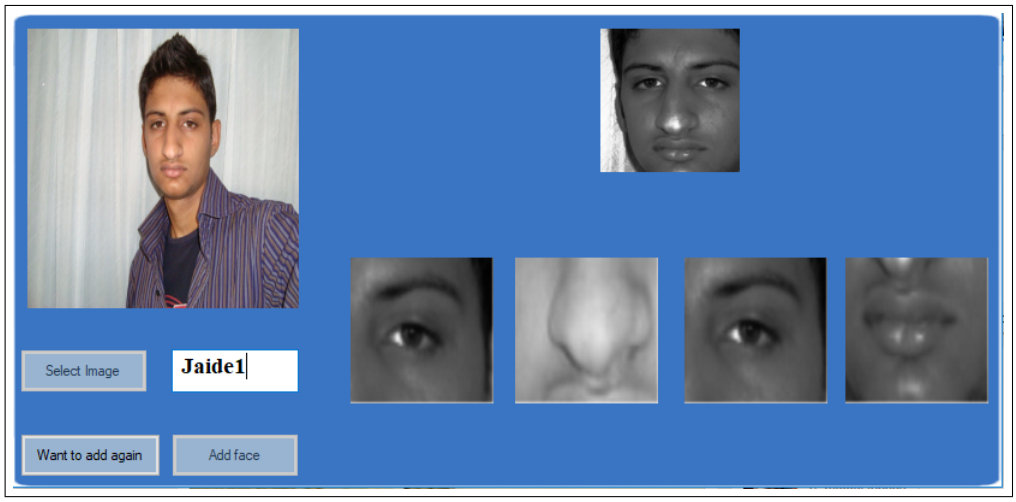

Figure 4.6: Second Screen Shot of GUI

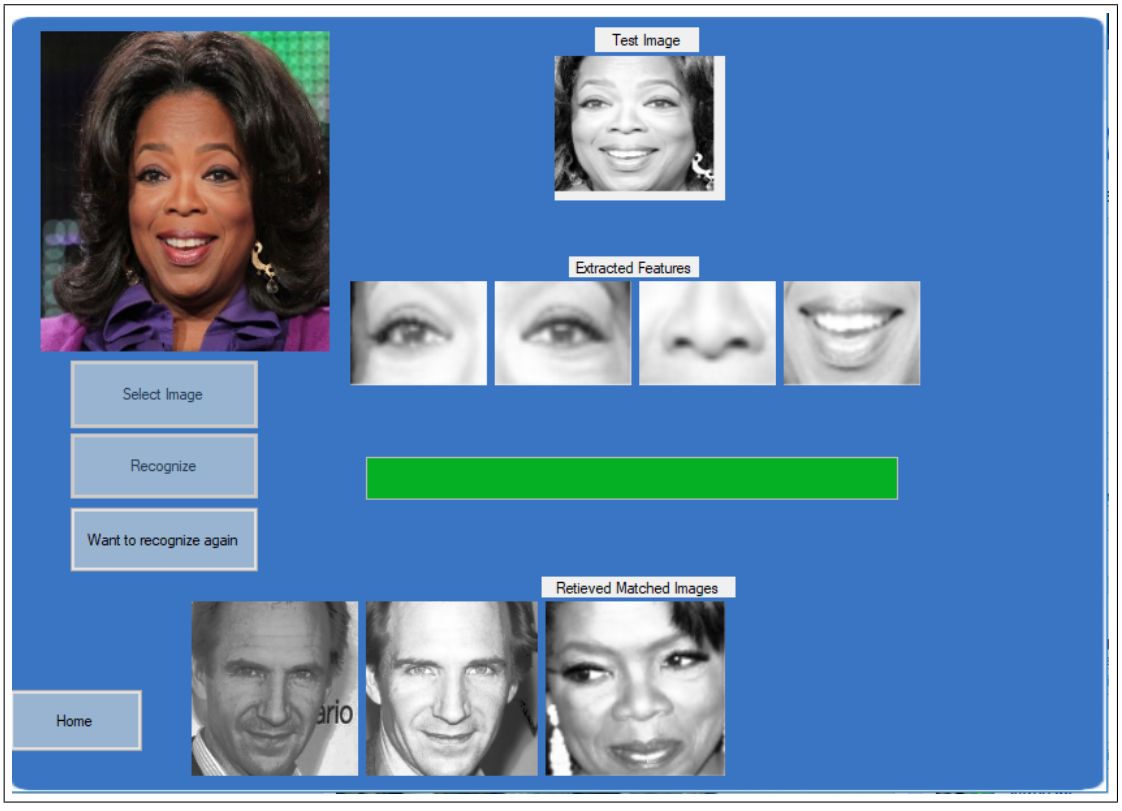

Figure 4.7: Third Screen Shot of GUI

Chapter 5

System Implementation

System implementation involves the conversion of the idea of the proposed system to a reality. This is done by using various tools and techniques available in the software domain. In this chapter we will discuss the tools/technology of the 'Age Invariant Face Recognition' system. This chapter also includes the core methodology of the project.

5.1 Tools Used

Following are the tools used to develop the proposed system.

5.1.1 Visual Studio

Microsoft Visual Studio is an integrated development environment (IDE) from Microsoft. It can be used to computer application as well application of the web and mobile. It uses various software development platform for its working such as Windows Form, Windows API. It contains a code editor which includes intelligence which completes code. It can also be used for different programming languages (36 as cited on the wiki page of the visual studio) such as C-sharp, C++, python etc. Different types of external APIs can be used with it. In our case we used Emgu-CV which is the OpenCV wrapper for C-sharp.

5.1.2 EmguCV

EmguCV is the crossnet platform for the OpenCV image processing library. It can be used with languages which are compatible with .NET such as C-sharp, VB, VC++. We used EmguCV 3.1.0 version which was the most stable version during the development of our software and for which the community support was rather active.

5.1.3 C-sharp Language

C-sharp language was used to write the code for the software. It is a general purpose object oriented programming language. It was compatible with EmguCV so it was opted as it is easier to use and provides a wide range of functionality.

5.2 Methodology

In this section, the methodology and algorithm used in the application will be explained. The following diagram shows the step by step processing which are done to reach the final results. This section shows in depth processing which was applied on the images for the end result.

5.2.1 Proposed Algorithm

The proposed algorithm of our project performed the following steps.

- The input images are preprocessed.
- Face region is detected and segmented.
- Facial features (eyes, nose and mouth) are extracted.
- Features are extracted from each region.
- System is trained using the extracted features.
- Same process is repeated on test images.
- Matching of features from test and training images is performed.
- Close matches are returned.

In the next step, a detailed insight into the proposed algorithm and its working is given which will dive further into the project and its working. Figure 5.1 gives a flow chart of the overall working of the system.

5.3 Dataset

The images used are from the online available dataset CACD (Cross Age Celebrity Dataset) [5] as shown in Figure 5.2. It consists of 160000 images of 2000 celebrities with ages varying from 16-62 years. However for our project, we have used 2 different aged images of 20 different celebrities. Out of 20 celebrities, images of 15 are used for system training and 5 are used for testing purposes.

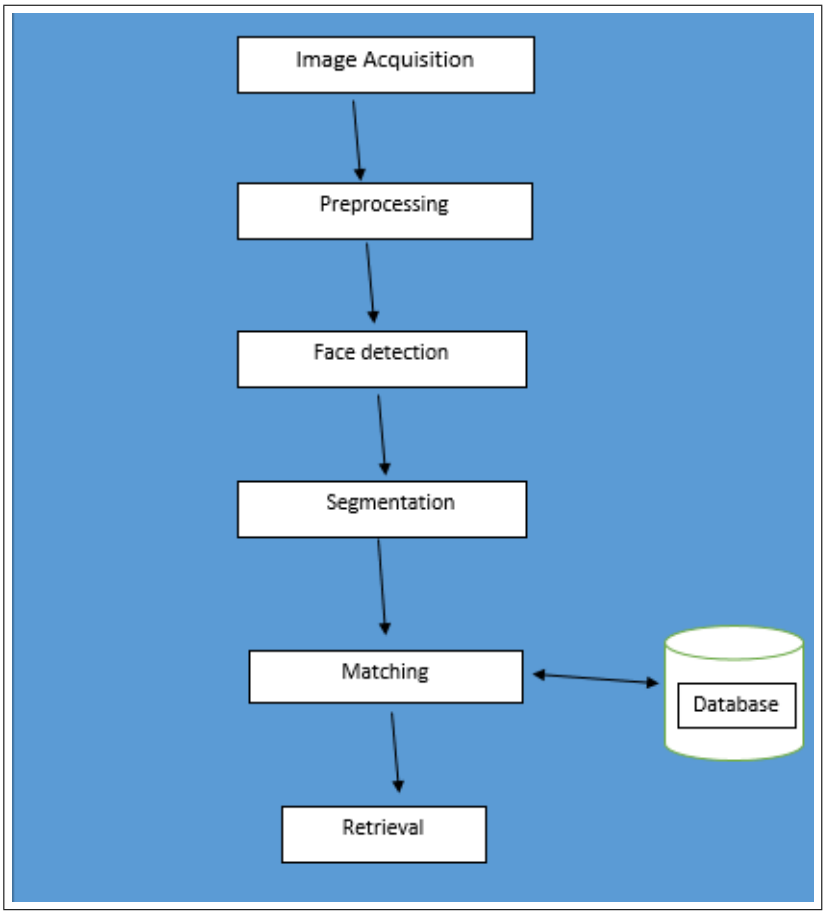

Figure 5.1: Complete Methodology Flow Chart

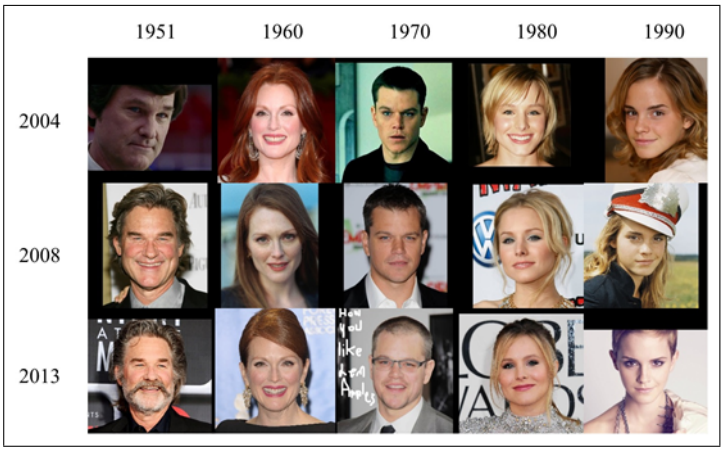

Figure 5.2: Sample Images from Dataset [5]

5.4 Techniques Applied

Following techniques were applied to implement the proposed algorithm steps:

5.4.1 Preprocessing

The input image is first preprocessed. Following preprocessing steps are performed on both the training and testing images:

- Resizing: We manually cropped the images and resized them according to our need.
- RGB to Grayscale Conversion: Images are then converted from RGB to grayscale for further processing as shown in Figure 5.3.
- Histogram Equilization: Finally histogram equalization is applied to enhance the contrast as shown in Figure 5.4.

Figure 5.3: RGB to Grayscale Conversion

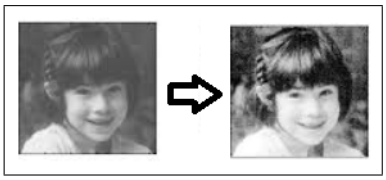

Figure 5.4: Image Enhancement using Histogram Equilization

5.4.2 Face Detection

In this step the face region of the image is detected using 'Haar Cascade' [6]. Object detection using Haar Cascade classifiers is an effective object detection method. Basically it is a learning based approach where cascade function is trained from a lot of positive and negative images as shown in Figure 5.5. First algorithm needs positive(images with faces) and the negative images(images without faces)to train the classifier then the features are extracted from it.

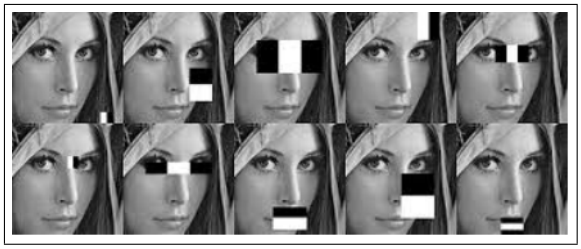

Figure 5.5: Haar Cascade for Face Detection [6]

5.4.3 Face Region Segmentation

Same technique of Haar Cascade is used for facial component i.e. (nose, mouth, left and right eye) detection as shown in Figure 5.6 and Figure 5.7.

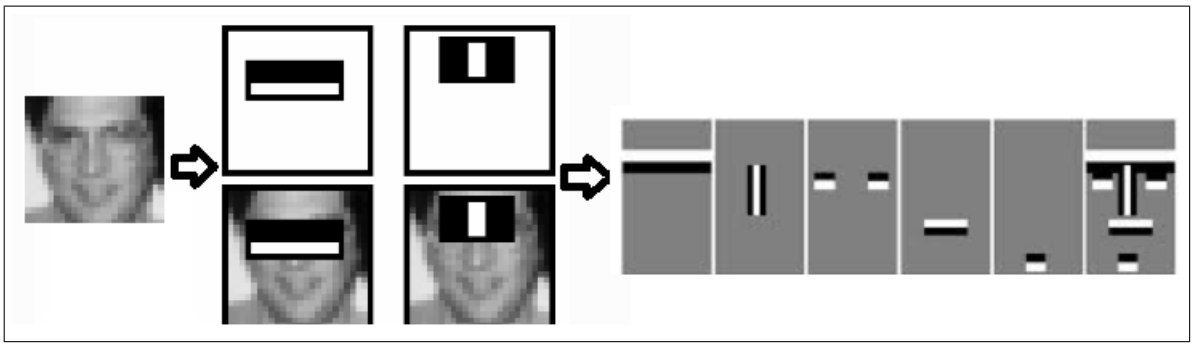

Figure 5.6: Haar Cascade for Face Component Detection [6]

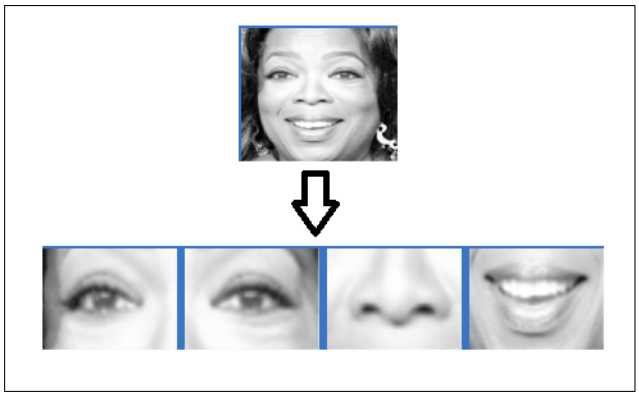

Figure 5.7: Face Component Segmentation

5.4.4 Matching

Once facial components are detection and segmented, templates are generated from each. These templates are then matched with the ones extracted from the query image. This technique is known as 'Template matching'. In this technique sampling of a

large number of points in an image is required. It is possible to reduce the number of samples by reducing the resolution of the search and template images by the same factor and performing the operation on the resultant downsized images, providing a search window of data points within the search image so that the template does not have to search every viable data point, or a combination of both.

5.4.5 Retrieval

Finally top three closest matching images are retrieved from the database. Since the query image is also present in the data base, therefore it will also be returned every time in the search results. To avoid this situation, the image that is an absolute match with zero distance is neglected and remaining top three images are displayed as shown in Figure 5.8.

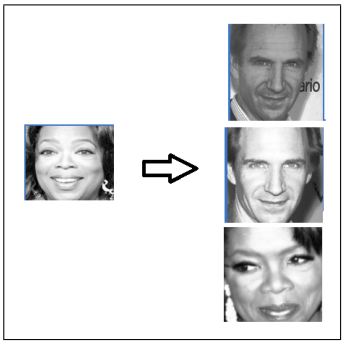

Figure 5.8: Top Three Closest Matching Retrieved Images

Chapter 6

System Testing and Evaluation

System testing of software or hardware is testing conducted on a complete, integrated system to evaluate the system's compliance with its specified requirements. In this chapter, numerous testing techniques are applied to test the functionality the application which can be used for the validity of the system. Testing is a very important phase of software development. It points out the problems in the software.

6.1 Software Testing Techniques

There are several methods which can be used to highlight the problems in the software. Some of these techniques are as follows:

- Function testing
- Performance testing
- Acceptance testing

6.1.1 Function Testing

In function testing, the functionality of the system is tested. This is not the functionality of the system as a whole but that of the units of the system. The system is easier to use and the GUI easily outlines how to use it.

6.1.1.1 Unit Testing

The units of the systems are tested. All of the components are working fine. The API connected appears to be working fine with no issue in their working.

6.1.1.2 Integration Testing

The different units/components are tested while connected with each other. These seem to be working properly. This test produced suitable results.

6.1.1.3 System Testing

The system was working as intended and the recognition of faces across different age group was working with acceptable accuracy.

6.1.1.4 Compatibility Testing

The compatibility testing is done to ensure that the software/system will run on different devices supporting the same platform. This was successful as the software was tested on different Windows based Computers.

6.1.1.5 Load Testing

The load testing is done to test what the software will do when it has to process a lot of information at the same time. Since the matching is done across a database, hence by increasing the size of the database increases the delay.

6.1.1.6 Security Testing

The application works by taking the locally stored images so security is not an issue. If images from an external hard-drive are to be taken, then the external hard-drive can be scanned for malicious activity.

6.1.1.7 Installation Testing

Installation testing is done to determine what type of pre-requirements are needed to be taken into consideration when the software is to be installed on a system. A computer having support for Visual SDK and EmguCV will be able to run the software.

6.1.2 Performance Testing

System performance testing is used to determine the effectiveness of the system. The system is detecting most of the faces but retrieval shows them in top three, there

are some faces which are returned in almost every search amongst the top three matches. Out of 10 query images the system correctly recognized 6, giving an overall system accuracy of 60%. Inclusion of further features can improve the accuracy. Face component detection at times causes errors as shown below:

Image #	Left eye	Right eye	Nose	Mouth
1	Yes	Yes	Yes	Yes
2	Yes	Yes	No	Yes
3	Yes	No	No	Yes
4	Yes	Yes	Yes	Yes

6.1.3 Acceptance Testing

It is important to test system for user acceptance.

6.1.3.1 Graphical User Interface Testing

In this step, the GUI which is the most important part of a system is tested. It is the interface which allows the user to interact with the system. GUI explains to the user how the system is to be used by the user. A GUI therefore must be simple to use and interactive so that a user can use it with ease. Our project contains a simple GUI which is easy to use and it is self-explanatory. Different users who were not familiar with our project were given the project to test the GUI. All of them were able to traverse through the system with ease and had no issue using it.

6.1.3.2 Usability Testing

The usability testing is the testing which is done by the actual users of the system. As mentioned in the GUI testing, users were able to use our system with ease.

6.2 Test Cases

Some of the test cases of the system are detailed in Tables (6.1 - 6.8).

Test Case ID	**TC-1**
Description	The application is to be installed in system
Requirement	Application should be install on system having emgu cv.
Steps to be taken	Run open the app on system.
Expected results	Application is install on the system.
Actual results	Application install successfully.
Status	Success.

Table 6.1: TestCase-1

Test Case ID	**TC-2**
Description	Application should start properly.
Requirement	Application should be launched when selected from the system by the user.
Steps to be taken	Select the right application from the system.
Expected results	Application should be launched without crush.
Actual results	The application was lunched successfully.
Status	Success.

Table 6.2: TestCase-2

Test Case ID	**TC-3**
Description	Select feature extraction from the menu.
Requirement	User should select feature selection properly.
Steps to be taken	Select the feature extraction.
Expected results	New form open where we extract features.
Actual results	New form open where we extract feature.
Status	Success.

Table 6.3: TestCase-3

Test Case ID	**TC-4**
Description	Select the image having human face.
Requirement	Face should be clear.
Steps to be taken	Click image button from gui to select image.
Expected results	A dialog box open for selecting the image.
Actual results	Image will be selected.
Status	Success.

Table 6.4: TestCase-4

Test Case ID	**TC-5**
Description	Extracting feature.
Requirement	The image we select should have clear face.
Steps to be taken	Click on the extract features button.
Expected results	All features will be extracted properly.
Actual results	Features extracted.
Status	Success.

Table 6.5: TestCase-5

Test Case ID	**TC-6**
Description	Select the recognition option from the menu.
Requirement	User should select recognition properly.
Steps to be taken	User should select recognition.
Expected results	New form will be open without crashing.
Actual results	New form will be opened for recognition.
Status	Success.

Table 6.6: TestCase-6

Test Case ID	**TC-7**
Description	Select the image having human face.
Requirement	Face should be clear.
Steps to be taken	Click image button from GUI to select image.
Expected results	A dialog box open for selecting the image.
Actual results	Image will be selected.
Status	Success.

Table 6.7: TestCase-7

Test Case ID	**TC-8**
Description	Recognize the face.
Requirement	The image we select should have clear face.
Steps to be taken	Click on the recognize button.
Expected results	Application should check images to show most matched face..
Actual results	Face will be matched.
Status	Success.

Table 6.8: TestCase-8

Chapter 7

Conclusions

Digital Image Processing is an interesting field to work on, which served as the motivation behind developing this project. It was a new and great experience for us to implement our project using Digital Image Processing techniques. The project 'Age Invariant Face Recognition' is designed to facilitate the users for recognizing persons more easily and in an improved way. The project was successfully developed and met its initial goal. The GUI is very simple and easy to use. There is no need of some type special training for this. It has been a great learning experience for us while developing this. It also came with its own set of difficulties as we were new to image processing using EmguCV, lots of research was needed to understand the working of image processing using EmguCV. By the end of this project, we have learned the EmguCV concept as well as C language and different techniques of image processing. We have also learned how system testing is done using different testing techniques.

Although our project performs well, but there is always a room for improvements. Following are some of the future directions for our work:

- Explore more features to improve accuracy
- Use the entire Dataset for training and testing

References

[1] Bernd Heisele, Purdy Ho, Jane Wu, and Tomaso Poggio. Face recognition: component-based versus global approaches. *Computer vision and image understanding*, 91(1):6–21, 2003. Cited on pp. ix, 5, and 6.

[2] Theodoros Papatheodorou and Daniel Rueckert. 3d face recognition. In *Face Recognition*. InTech, 2007. Cited on pp. ix, 5, and 7.

[3] Christophe Garcia and George Tziritas. Face detection using quantized skin color regions merging and wavelet packet analysis. *IEEE Transactions on multimedia*, 1(3):264–277, 1999. Cited on pp. ix, 6, and 7.

[4] Diego A Socolinsky, Andrea Selinger, and Joshua D Neuheisel. Face recognition with visible and thermal infrared imagery. *Computer vision and image understanding*, 91(1):72–114, 2003. Cited on pp. ix, 6, and 8.

[5] Bor-Chun Chen, Chu-Song Chen, and Winston H Hsu. Cross-age reference coding for age-invariant face recognition and retrieval. In *European Conference on Computer Vision*, pages 768–783. Springer, 2014. Cited on pp. ix, 2, 26, and 27.

[6] R Padilla, CFF Costa Filho, and MGF Costa. Evaluation of haar cascade classifiers designed for face detection. *World Academy of Science, Engineering and Technology*, 64, 2012. Cited on pp. ix, 28, and 29.

[7] Wenyi Zhao, Rama Chellappa, P Jonathon Phillips, and Azriel Rosenfeld. Face recognition: A literature survey. *ACM computing surveys (CSUR)*, 35(4):399–458, 2003. Cited on p. 1.

[8] Anil K Jain and Stan Z Li. *Handbook of face recognition*. Springer, 2011. Cited on pp. 1 and 6.

[9] Unsang Park, Yiying Tong, and Anil K Jain. Age-invariant face recognition. *IEEE transactions on pattern analysis and machine intelligence*, 32(5):947–954, 2010. Cited on p. 1.

[10] Rabia Jafri and Hamid R Arabnia. A survey of face recognition techniques. *Jips*, 5(2):41–68, 2009. Cited on p. 5.