Component-Based Face Recognition

by

Jean Vincent Fonou Dombeu

Submitted in fulfillment of the academic requirements of the degree of
Master of Science

in the

School of Computer Science, University of KwaZulu-Natal
Westville Campus, Durban 4000, South Africa
November, 2008

As the candidate's supervisor, I have approved the dissertation for submission

Signed:

Prof. Jules-Raymond Tapamo

Date: November 2008

© Copyright by Jean Vincent Fonou Dombeu, 2008

UNIVERSITY OF KWAZULU-NATAL

FACULTY OF SCIENCE AND AGRICULTURE

The research described in this dissertation was conducted in the School of Computer Science of the University of KwaZulu-Natal under the supervision of Professor Jules-Raymond Tapamo. I, Jean Vincent Fonou Dombeu declare that

1. The research reported in this dissertation, except where otherwise indicated, is my original research.
2. This dissertation has not been submitted for any degree or examination at any other university.
3. This dissertation does not contain other persons' data, pictures, graphs or other information, unless specifically acknwoledged as being sourced from other persons.
4. This dissertation does not contain other persons' writing, unless specifically acknowledged as being sourced from other researchers. Where other written sources have been quoted, then:
   (a) Their words have been re-written but the general information attributed to them has been referenced.
   (b) Where their exact words have been used, then their writing has been placed in italics and inside quotation marks, and referenced.
5. This dissertation does not contain text, graphics or tables copied and pasted from the Internet, unless specifically acknowledged, and the source being detailed in the dissertation and in the references sections.

Signed:

Jean Vincent Fonou Dombeu

Date: November 2008

PUBLICATIONS

The following publications have resulted from this work.

1. J.V. Fonou Dombeu and J.R. Tapamo, "A Component-Based Face Recognition Combining Gabor Filters and Zernike Moments", *International Journal of Computer Application in Technology (IJCAT)*, Submitted.
2. J.V. Fonou Dombeu and J.R. Tapamo, "Validation of Detected Facial Components for and Accurate Face Recognition", *In Proceedings of the* 18^{th} *Annual Symposium of Pattern Recognition Association of South Africa (PRASA)* , Pietermaritzburg, South Africa, pp. 141-146, November 28-30, 2007.
3. J.V. Fonou Dombeu and J.R. Tapamo, "Texture and Shape-Based Face Recognition by Mixture of Gabor Filters and Zernike Moments", *In Proceedings of the International Conference on Information Technology and Applications (ICITA 2008)*, Cairns Gueensland, Australia, pp. 783-788, June 23-26, 2008.
4. J.V. Fonou Dombeu and J.R. Tapamo, "Adaptive Detection of Regions Of Interest for Face Recognition", *In Proceedings of the International Conference on Image Processing, Computer Vision, and Pattern Recognition (IPCV'2008)*, Monte Carlo Resort, Las Vegas, Nevada, USA, July 14-17, 2008.

The author's contributions in each of the papers are as follows:
1^{st}*Author* : Literature review, design and implementation of algorithms, edition of papers.
2^{nd}*Author* : Giving ideas, providing advice, discussing issues on models and algorithms, proof-reading manuscripts.

Signed:

Jean Vincent Fonou Dombeu

Date: November 2008

To Helene Ngambeu and Gabriel Dombeu,

To Nyanine Chuele Mawe Mpombo,

To the Lord Jesus Christ

Table of Contents

List of Tables . . . . . **viii**

List of Figures . . . . . **ix**

Abstract . . . . . **xiii**

Acknowledgements . . . . . **xiv**

Chapter 1 Introduction **1**

1.1 Motivation 1

1.2 Research Objectives 3

1.3 Dissertation Outline 3

1.4 Original Contribution in the Dissertation 4

Chapter 2 Literature Review **5**

2.1 Introduction 5

2.2 Human Biometric Traits 5

2.2.1 Behavioral Traits 5

2.2.2 Physiological Traits 9

2.3 Face Recognition 13

2.3.1 Background 13

2.3.2 Component-Based Face Recognition 14

2.4 Conclusion 17

Chapter 3 Materials and Methods **18**

3.1 Introduction 18

3.2 Basic Operations and Entities 18

3.2.1 What is a Pixel ? 18

3.2.2 Segmentation and Binarization 18

3.2.3 Connected Components 19

3.2.4 Convex Hull 21
3.2.5 Similarity Measures 22
3.3 Feature Extraction Techniques 25
3.3.1 Eigenface Decomposition 25
3.3.2 Gabor Filters 27
3.3.3 Zernike Moments 28
3.4 Learning and Recognition 30
3.4.1 Nearest-Neighbor 30
3.4.2 K-means Clustering 30
3.4.3 Support Vector Machine (SVM) 32
3.4.4 Recognition, Authentication and Identification 35
3.5 Conclusion 36

Chapter 4 System Overview 37
4.1 Introduction 37
4.2 Structure of the System 37
4.3 Adaptive Detection of Regions of Interest 38
4.4 Computation and Representation of Features 47
4.4.1 Gabor Feature Extraction 47
4.4.2 Zernike Moments Feature Extraction 53
4.5 Features Fusion and Normalization 56
4.6 Classification/Recogntion 58
4.6.1 Storage of Known Persons in the Database 59
4.6.2 Identification of a Candidate Face 59
4.7 Conclusion 60

Chapter 5 Experimental Results and Discussion 61
5.1 Introduction 61
5.2 Programming Environment 61
5.3 Data Set 61
5.4 Experiment1: Components Extraction and Validation 62
5.5 Experiment2: Feature Extraction and Identification 64

5.6 Conclusion 68

Chapter 6 Conclusion and Future Works 69

6.1 Summary of Work 69

6.2 Limitations of the System, Recommendations and Future Work . . . 70

6.3 Conclusion 71

Bibliography 72

Appendix A UML Class Diagram of The System and Samples Outputs Results 81

List of Tables

Table 4.1 Dimensions of Eyes Components detected in the Bank of Images in Fig. 4.10, numbered from Top to Bottom. 51

Table 4.2 Example of a Gabor Feature Vector for the Left Eye of the Top Image in Fig. 4.10. 54

Table 4.3 Repetitions of Zernike Moments of Order 4. 55

Table 4.4 Zernike Moments Features at Order 4. 56

Table 4.5 Number of Zernike Moments Features at Orders from 0 to 10. . 56

Table 4.6 Example of Nose Zernike Moment Features. 57

Table 5.1 Triplets extracted from both Training and Sample Face Images. 65

Table 5.2 K-means Classification of a New Triplet from a Sample Face Image. 66

Table 5.3 Comparative Table of Facial Components Detected. 67

Table 5.4 Comparative Table of Components Detection Success Rate in Different Orientation of Face. 67

Table 5.5 Comparative Table of Recognition Rate. 68

List of Figures

Figure 3.1 A Pixel in an Image Array. 19

Figure 3.2 Exemple of Image Binarization:(a):Original Image; (b):Binarized Image. 20

Figure 3.3 Connected Components:(a):4-Connectivity of the Center Pixel; (b):8-Connectivity of the Center Pixel. 21

Figure 3.4 Practical Example of Finding Connected Components in an Image:(a):Binary Image; (b):Four Connected Components found with 4-Connectivity; (c):One Connected Component found with 8-Connectivity. 22

Figure 3.5 Example of Connected Components of a Binary Image ; (a):Original Image; (b):Binary Image;(c):Connected Components of the Binary Image. 23

Figure 3.6 Example of a Convex and Non-convex Polygon: (a):Convex Hull of the Set of Points A,B,C,D,E; (b):Non-convex Hull Polygon of the Set of Points A,B,C,D,E. 23

Figure 3.7 Example of Convex Hull of the Centroid of Face Components: Eyes, Nose and Mouth. 24

Figure 3.8 Sample Feature Vector derived from the Matrix of Gray Levels of an Image. 26

Figure 3.9 Sample Training Data Set. For example: Face and Non-face Components Features Vectors ($F_{+} and\ F_{-}$). 32

Figure 3.10 Example of Hyperplanes Separating Two Classes of Feature Points. 33

Figure 3.11 Optimal Separating Hyper-plane, Margin, Support Vectors. . . 33

Figure 4.1 Flow Chart of the Entire Face Recognition System. 38

Figure 4.2 Flowchart of the Components Detection and Validation System. 39

Figure 4.3 Step by Step Results of the Approach. a) Original Face Images, b) Binary Images, c) Connected Components of Binary Images, d) Detected Region of Interest, e) Detected Region of Interest with Rectangular Boundaries, f) Original Images with the Detected Regions of Interest. 41

Figure 4.4 Coordinates of a Detected Region of Interest in the Face Image. 43

Figure 4.5 Facial Components Detected in Different Orientations. 44

Figure 4.6 Examples of Face Images with Convex Hull of Detected Facial Components. 45

Figure 4.7 Examples of Face Images with the Angles (θ_0, θ_1) at the Side of the two Centroids with Lowest y-coordinates (as describe in Fig 3). The Absolute Value of the Difference of these Angles is $\Delta\theta = \mid \theta_0 - \theta_1 \mid$ and the Triplet is defined as: $(\theta_0, \theta_1, \Delta\theta)$; (a) Face with successful detected Components; (b) Face with an undetected Component. 46

Figure 4.8 Flowchart of the Computation and Representation of Feature Module. 48

Figure 4.9 The Framework of the Gabor Feature Extraction. 51

Figure 4.10 Examples of Imaginary Parts of Gabor Filters:(a):Input Images with detected and validated Facial Components;(b):Extracted Facial Components;(c,d):Original Image with Imaginary Parts of Gabor Filters at Scales $\pi/4$ and $\pi/2\sqrt{2}$, and Orientations $5\pi/8$ and $3\pi/8$, respectively. 53

Figure 4.11 The Framework of the Zernike Moments Feature Extraction. . 55

Figure 4.12 The Framework of the Feature Fusion Strategy. 58

Figure 4.13 Framework of a Typical Face Recognition System. 59

Figure 5.1 Chart of Detection Rate of Face Components in Frontal, Left, and Right View. 62

Figure 5.2 Curves of Detection Rate of the System in Different Orientations. 63

Figure 5.3 Result of SVM Classification of Triplets extracted from the Sets of detected Facial Components; the Top Left Feature Points form the Cluster of Triplets extracted from Facial Components incorrectly detected and the Bottom Right Feature Points the Cluster of correctly detected Facial Components, as shown in Table 5.2. 64

Figure 5.4 Examples of Challenging Face Images with Hair Covering a Portion of the Eye and with Beard. 67

Figure 5.5 Histograms of Recognition Rates of in Frontal, Left, and Right View, and the Average Recognition Rate in different Orientations of Face. 68

Figure A.1 UML Class Diagram of the System. 81

Figure A.2 Sample Output Results of Plugins; (a): Original Images; (b): Binary Images; (c): Connected Components of Binary Images; (d): Detected Components; (e): Detected Components with Bounding Box and Convex Hull; (f): Original Images with Bounding Box of Detected Components; (g): Original Images with Bounding Box and Convex Hull of Detected Components. 83

Figure A.3 Sample Output Results of Plugins; (a): Input Images with Detected and Validated Facial Components; (b): Extracted Facial Components; (c,d):Original Images with Imaginary and Real Parts of Gabor Filters at Scale $\pi/4$ and Orientation $5\pi/8$; (e,f):Original Images with Imaginary and Real Parts of Gabor Filters at Scale $\pi/2\sqrt{2}$ and Orientation $3\pi/8$. 84

Figure A.4 Sample Output Results of Plugins; (a): Original Images; (b): Binary Images; (c): Connected Components of Binary Images; (d): Detected Components; (e): Detected Components with Bounding Box and Convex Hull; (f): Original Images with Bounding Box of Detected Components; (g): Original Images with Bounding Box and Convex Hull of Detected Components. 85

Figure A.5 Sample Output Results of Plugins; (a): Input Images with Detected and Validated Facial Components; (b): Extracted Facial Components; (c,d):Original Images with Imaginary and Real Parts of Gabor Filters at Scale $\pi/4$ and Orientation $5\pi/8$; (e,f):Original Images with Imaginary and Real Parts of Gabor Filters at Scale $\pi/2\sqrt{2}$ and Orientation $3\pi/8$. 86

Figure A.6 Sample Output Results of Plugins; (a): Original Images; (b): Binary Images; (c): Connected Components of Binary Images; (d): Detected Components; (e): Detected Components with Bounding Box and Convex Hull; (f): Original Images with Bounding Box of Detected Components; (g): Original Images with Bounding Box and Convex Hull of Detected Components. 87

Figure A.7 Sample Output Results of Plugins; (a): Input Images with Detected and Validated Facial Components; (b): Extracted Facial Components; (c,d):Original Images with Imaginary and Real Parts of Gabor Filters at Scale $\pi/4$ and Orientation $5\pi/8$; (e,f):Original Images with Imaginary and Real Parts of Gabor Filters at Scale $\pi/2\sqrt{2}$ and Orientation $3\pi/8$. 88

Figure A.8 Sample Output Results of Plugins; (a): Original Images; (b): Binary Images; (c): Connected Components of Binary Images; (d): Detected Components; (e): Detected Components with Bounding Box and Convex Hull; (f): Original Images with Bounding Box of Detected Components; (g): Original Images with Bounding Box and Convex Hull of Detected Components. 89

Figure A.9 Sample Output Results of Plugins; (a): Input Images with Detected and Validated Facial Components; (b): Extracted Facial Components; (c,d):Original Images with Imaginary and Real Parts of Gabor Filters at Scale $\pi/4$ and Orientation $5\pi/8$; (e,f):Original Images with Imaginary and Real Parts of Gabor Filters at Scale $\pi/2\sqrt{2}$ and Orientation $3\pi/8$. 90

Figure A.10 Sample Output Results of Plugins; (a): Original Images; (b): Binary Images; (c): Connected Components of Binary Images; (d): Detected Components; (e): Detected Components with Bounding Box and Convex Hull; (f): Original Images with Bounding Box of Detected Components; (g): Original Images with Bounding Box and Convex Hull of Detected Components. 91

Figure A.11 Sample Output Results of Plugins; (a): Input Images with Detected and Validated Facial Components; (b): Extracted Facial Components; (c,d):Original Images with Imaginary and Real Parts of Gabor Filters at Scale $\pi/4$ and Orientation $5\pi/8$; (e,f):Original Images with Imaginary and Real Parts of Gabor Filters at Scale $\pi/2\sqrt{2}$ and Orientation $3\pi/8$. 92

Abstract

Component-based automatic face recognition has been of interest to a growing number of researchers in the past fifteen years. However, the main challenge remains the automatic extraction of facial components for recognition in different face orientations without any human intervention; or any assumption on the location of these components. In this work, we investigate a solution to this problem. Facial components: eyes, nose, and mouth are firstly detected in different orientations of face. To ensure that the components detected are appropriate for recognition, the Support Vector Machine (SVM) classifier is applied to identify facial components that have been accurately detected. Thereafter, features are extracted from the correctly detected components by Gabor Filters and Zernike Moments combined. Gabor Filters are used to extract the texture characteristics of the eyes and Zernike Moments are applied to compute the shape characteristics of the nose and the mouth. The texture and the shape features are concatenated and normalized to build the final feature vector of the input face image. Experiments show that our feature extraction strategy is robust, it also provides a more compact representation of face images and achieves an average recognition rate of 95% in different face orientations.

Acknowledgements

First and foremost, it couldn't be possible to handle a topic in such a cutting edge research field without excellent supervision. I would like to express my deep and sincere gratitude to Prof. Jules-Raymond Tapamo for all the time he has spent to look after what I was doing, to discuss possible and alternative solutions at each phase of my project, to correct my manuscripts, and to provide me with advice and encouragement. Your simplicity, your sense of endeavor, and your research skills have inspired me forever.

The members of Image Processing, Computer Vision and Data Mining Research Group for the stimulating environment they have provided me with while conducting the research. It was always an ongoing pleasure for me to present my research reports to you and receive your precious and constructive comments.

I am also grateful to my friends Zygmunt Szpak, Michael Da Silva, Wayne Chelliah, and Brendon Clyde McDonald for their friendship, co-operation in sharing ideas, and help towards improving the quality of this work by making observations on the manuscripts. Particular thanks go to Zygmunt Szpak, for sharing his deep knowledge of our programming environments with me.

I am also indebted to my family members, particularly my wife Nyanine Chuele Mawe Mpombo, my sisters and brothers Madeleine Meyou, Jeannette Youmbi, Irene Domtchouang, Leacadie Eleonore Mabekam, Adeline Makougang, Valery Oscar Dzeukeng, and Ines Isabelle Beukam for their patience support and encouragements during my studies. Beside, I wish to express my gratitude to Ludovic Tongouo Tchatchouang for his advice and assistance during my times of difficulty .

Finally, I am grateful to the School of Computer Science and its staff members, for having provided me with a good work environment for my studies.

Chapter 1

Introduction

1.1 Motivation

The main security concern of governments and organizations all over the world nowadays is the fight against organized crime. This mobilisation reached a turning point with the 11 September 2001 attack in the United States where terrorists have disclosed to the world, the weaknesses of all the security techniques that have been implemented so far for identifying people at the entrance and the exit of sensitive areas.

Traditionally, human beings are identified in several situations in their daily life; either when using a key to open the door of their houses, log in a system via a password, or gain access to automated systems such as cell phone, online banking devices, offices or secure spaces in certain buildings with a PIN code. However, these means of identifying people have shown their limits. In fact, keys to open a door, as password and PIN code can be lost or stolen. Particularly, password and PIN code can be forgotten or neglected by their owners. These weaknesses make the security measures based on such tools less reliable and inefficient nowadays. The direct consequence is that, security mechanisms based on biometric technologies are gaining acceptance by individuals, companies, organisations, and governments, as alternative security means for protecting their systems. The advantage of biometric technologies over the traditional security means based on keys, passwords, and PIN codes is that a biometric signature cannot be borrowed, stolen, or forgotten, and it is practically impossible to forge it. Biometric technologies are being used in security banking systems, mobile phones, immigration, health, authentication systems, and many other applications.

Biometrics is the study of methods or techniques for uniquely identifying or recognizing a person, based on one or more physiological or behavioral traits. The physiological traits are related to the shape of the human body, whereas the behavioral traits refer to the human behavior. The widely used human physiological biometric traits include: fingerprint, face, iris, retina, and gait. The main human behavioral biometric traits are : dynamic signature, voice, Deoxyribo Nucleic Acid (DNA), and keystroke. Basically, a human physiological or behavioral trait is used as a biometric characteristic as long as it satisfies the following requirements [48]:

- Universality (Everyone should have it),
- Distinctiveness (No two should be the same),
- Permanence (It should be invariant over a given period of time) and
- Collectability (Could be collected easily).

In the real life applications, three additional factors should also be considered:

- Performance (accuracy, speed, resource requirements),
- Acceptability (it must be harmless to users) and
- Circumvention (it should be robust against various fraudulent method).

Although many researches in biometric recognition have demonstrated that the best biometric solutions nowadays are those combining at least two biometric traits, also called multimodal biometric systems [5, 32, 74, 78, 98], human face have been a subject of interest to a growing number of researches in biometric recognition over the recent years. The reasons being that, not only does the human face meet the above criteria of a good human biometric trait, but also face recognition systems are useful in many applications including: public security, law enforcement and commerce, such as mug-shot database matching, identity authentication for credit card , passport and driver license, access control, information security, and intelligent surveillance. Furthermore, face recognition taken alone has a great advantage over other biometric technologies in that it is non-intrusive and user-friendly [94]. In fact, face images can be captured at a distance without any cooperation from the user. Therefore, face

recognition technologies need an ongoing improvement in order to fit the needs of their broad application domains.

1.2 Research Objectives

Many authors [94, 40] argue that the main problems with the existing face recognition systems are that they are still highly sensitive to environmental factors such as: variations in facial orientation, expression and lighting conditions during image acquisition. It has been reported that designers of face recognition systems could alleviate these problems by focusing their feature extraction algorithms on the facial components instead of the whole face image [19, 38, 80, 93].

In this dissertation, we aim to:

1) Investigate how to detect accurately facial components (eyes, nose, and mouth), given inputs gray scale face images taken in different orientations, for an accurate face recognition and the most robust feature extraction algorithms that could be applied on the facial components once detected, so that we could represent more accurately and compactly each input face image.

2) Apply an efficient classification/recognition technique on the feature vectors obtained in order to assess the feasibility of the implemented model.

3) Design and implement a robust face recognition system based on the key components of the face.

4) Compare the performance of the implemented model with existing one.

1.3 Dissertation Outline

Chapter 2 presents the background on human biometric characteristics and the state of the art in components-based face recognition. In chapter 3, feature extraction, learning and classification methods are discussed. Basic image processing operations are presented as well. Chapter 4 presents the overview of the face recognition system and provides detailed information on its implementation. Experimental results and

discussions are carried out in chapter 5. A conclusion and discussion of possible future works are presented in chapter 6.

1.4 Original Contribution in the Dissertation

The original contributions in the dissertation are presented in Chapter 4 and include:

1. In Section 4.3, we propose an adaptive strategy for detecting facial components (the eyes, the nose and the mouth) for an accurate face recognition. An input gray scale face image is first binarized and the connected components of the resulting image are computed. Thereafter, an iterative strategy is employed to remove the irrelevant components. The iteration terminates when the remaining components are most probably the targeted components. This work has been published in the proceedings of *IPCV'08* in July 2008 [25].

2. In Section 4.3, we investigate the validation of the facial components once detected. The centroid of each detected facial component is computed. The angles at the sides of the two first centroid with lower y-coordinates in the face space are computed. The difference of these angles is further calculated. The two angles and their difference are used to construct a triplet. Finally, the triplet is used to classify the set of facial components detected into the class of correctly detected facial components or the one of the wrongly detected facial components. This work has been published in the proceedings of *PRASA 2007* [24].

3. In Section 4.4, we propose a robust feature extraction framework for face recognition. The textures of the eyes are extracted with Gabor Filters and the shapes of the nose and the mouth are computed with Zernike Moments. The texture features and the shape features are concatenated and normalized to build the final biometric signature of the input face image. This work has been published in the proceedings of *ICITA 2008* [26].

Chapter 2

Literature Review

2.1 Introduction

Biometric field studies the identification of people based on their behavioral or physiological characteristics. In this chapter, we provide a brief presentation of biometric traits. Thereafter, we review the state of the art in components-based face recognition in order to highlight the current challenges in the field.

2.2 Human Biometric Traits

2.2.1 Behavioral Traits

Voice/speaker

Speaker recognition or voice recognition is the task of recognizing people from their voices. It has been proven that human speech contains information about the identity of the speaker [8]. In [39] Kim et al. have classified this information into two categories: Low-level and high-level. High-level information include, the language spoken, the speech pathologies, the physical and emotional state of the speaker. Referring to the actual state of researches in the field, these features are currently only recognized and analyzed by humans. Low-level information denote the information like pitch period, rhythm, tone, spectral magnitude, frequencies, and bandwidths of an individual's voice. These features are used by automatic speaker recognition (ASR) systems. The first work carried out in the area was done by Lawrence Kersta at the Bell *Labs* [8] in the early 1960*s*. He used an electro-mechanical device to produce a speech template called voiceprint from a speech of a person. Modern speaker recognition systems use a standard microphone to capture the speech signal. From the speech signal captured, the voice feature is extracted and the speech modelling is carried out. Once the modelling is done, the pattern classification is performed

prior to the recognition [39]. In the literature, the most robust algorithms used for feature extraction in speaker recognition are principal component analysis (PCA)and independent component analysis (ICA) [39, 52, 47]. The main technique employed for speech modelling is the Hidden Markov Models (HMM) [8, 39, 52]. The pattern classification is based on pattern matching algorithms.

Typically, a speaker recognition system employs three styles of spoken input: text-dependent, text-prompted, and text-independent [8, 47]. The text-dependent approach requires a user to pronounce the same text as the training data. In this case, the text to be spoken is known since the enrollment and could be a fixed text or a phrase. In text-prompted systems, the speaker is asked to speak a prompted text which could be any kind of text. This type of speaker recognition systems involves a much more elaborate recognition model than text-dependent systems where the text is always the same. Text-independent approach requires little or no cooperation from the user. In fact, the enrollment may happen without the user's knowledge; some recorded pieces of speech may suffice. This approach is more often used for speaker identification, whereas the two previous approaches are more suitable for verification. Text-independent is also completely language independent. However, since only general speaker-specific properties of the speaker's voice are used, the accuracy of the recognition is reduced.

The advantage of voice/speaker recognition technology is that very affordable hardware is needed. In fact, in most computers a soundcard and a microphone are implemented. However, there are some disadvantages too. Firstly, the human voice is variant in time, then the template needs to be reacquired after a certain time, which is cumbersome practically. Secondly, the human voice could be influenced by factors such as cold, hoarseness, stress, emotional states or puberty vocal change. Also, the human voice is not as unique as strong biometric traits like fingerprint and iris [8].

Keystroke

Keystroke recognition also called keystroke dynamics by certain authors is the biometric field that studies the identification of individuals based on the manner they type on the keyboard of a computer. As almost all sensitive information of companies and organizations are accessed online nowadays, keystroke recognition is the

means of providing strong authentication protection against online fraud and theft for access to networks, systems or internet based applications [54]. Keystroke feature is extracted from the typing behavior of a person, based on parameters such as the latencies between successive keystrokes, keystroke durations, finger placement and the way fingers are pressed on the keys [31]. Once the features are extracted, they are processed through an algorithm that compares the person's typing behavior to a sample collected in a previous session; the output of the comparison is a score, which, when greater than a conveniently chosen threshold, leads to positive identification or verification of a user [31, 54]. Algorithms used in keystroke dynamics encompass K-means, Expectation Maximization on Gaussian Mixtures, Hidden Markov Model (HMM), and probabilistic neural networks [53].

Over other biometrics solutions, keystroke dynamics has the advantages of being easy to implement, affordable and user-friendly. In fact, it is software-based and does not require additional hardware[54, 66]; thus, its cost is lower. Furthermore, users being authenticated are not aware of any difference in their habits as the same keyboard and the login process used before are used for authentication [31, 54]. The disadvantage is that its use did not solve the problem of users having to remember their passwords for any access. Furthermore, the technology is in its early stage and has not been tested on a wide scale [66].

Gait

The way a living person walks can be used to determine his/her identity. This assertion is supported in [9, 42, 56] where it is reported that the gait of a living person contains certain parameters such as the body mass, the limb length, and the habitual posture, that are unique to every individual. Computer vision community has classified the gait as a biometric signature that can be used to automatically identify people. The corresponding research area is called automated gait recognition. In recent years, the topic has been very attractive judging from the number of papers that are found in the literature. This effervescence is due to the fact that automated gait recognition seems to be more suitable for passive surveillance than most biometric technologies, as the gait can be measured at a distance without any cooperation from the subject, even in low resolution video [4, 56].

Practically, automated gait recognition consists of capturing the image of a walking person, localize and extract the silhouette image. From the silhouette image, parameters such as the width of the outer contour and the optical flows are computed [99] and techniques based on linear and stationary analysis of the gait marker trajectories [51] are applied in order to derive the gait signature. In [99], Rong et al. have explained that the derived signature contains the temporal dynamics of the gait of the subject, whereas the trajectories of the corresponding joint position reveal the spatial-temporal history. Once the gait signature is obtained, it is used to compute features such as knee stride width and knee elevation, ankle stride width and ankle elevation, which will be used for classification and recognition. Only few algorithms are found in the literature focusing on gait classification/recognition; they encompass optical flow structure from motion approach [4], self-correlation, principal component analysis [56], and Hidden Markov Model (HMM) [99].
The main problem in automated gait recognition that is also encountered in face recognition, is that of the position of the subject during the image acquisition. In fact, like with automated face recognition, the gait recognition performs very well when the image of the subject is captured when he/she is placed in a frontal view. However, the case where the subject appears in a side view remains a challenging problem in the area.

Signature

Affixing his/her signature on a document is indubitably among the most habitual administrative tasks every individual performs in his/her day to day activities, as it is the commonly acceptable means to endorse his/her responsibility. In the biometric research community, this type of signature also called off-line signature [50, 88] has been an active subject of research in the past 30 years [50]. The identification process was carried out by scanning a signature of a person from a paper, extract its shape and exploit it to either accept or reject the claimed identity. However, the results obtained were far from being perfect and do not provide the required accuracy for many security problems; the reasons being the fact that a signature written on a paper can be forged or mimicked by intruders to fool the system, and two successive signatures of the same person can be different due to noise introduced by the scanning

device or a difference in pen width. Nowadays, the research endeavors in the field focus on the dynamic signature or on-line signature, encouraged by the emergence of modern portable computers and personal digital assistants (PDA_S) in business environments, which employ handwritten input devices. In the literature [50, 88], it is recognized that on-line signature is more unique and difficult to forge than off-line signature. In fact, in addition to the shape information used in off-line signature, dynamic features like speed, pressure and capture time of each point on the signature trajectory are involved in the classification [50, 70, 88]. One important application of on-line signature recognition today is its use in intrusion detection in computer networks systems, that allows reducing the rate of fraud in online bank transactions based on credit card and checks [88, 95].

In practice, the signature of a person is acquired using special equipment; the most common is a digitalizing tablet [50]. Afterwards, parameters of the signature are extracted and compared to the templates signatures stored in the database to recognize or reject the given signature.

The most popular algorithms used in dynamic signature or on-line signature recognition are Time Warping and Dynamic Matching (DTW)[50, 70, 88], Neural Networks [50, 70], and Hidden Markov Model (HMM) [50, 70, 88].

The drawback of on-line signature is that, a human signature can change over time and it is not nearly as unique or difficult to forge as iris patterns and fingerprint. However, the signature's widespread acceptance by the public makes it more suitable for certain low-security authentication needs [88].

2.2.2 Physiological Traits

Ear

Ears are among the most visible human body traits. Investigations have concluded that the shape and the appearance of the human ear are unique enough to each individual and relatively remain unchanged from birth to old age [17, 96]. Furthermore, by nature each individual has ears. These arguments are in favor of the use of ear as a biometric signature. Some researchers have gone as far as comparing the ear to the most popular used biometric trait which is the face; they have discovered that ears have several advantages over complete faces: reduced spatial resolution, a more

uniform distribution of color, and less variability with expressions and orientations of the face[41].

Until now, only few applications have used ear in identifying people. The earlier application of the ear in recognition has occurred in crime investigation,where earmarks have been used as evidence in court[41]. An interesting application of ear recognition in the near future, which will be useful for a large public, is the design and implementation of an ear recognition technology that will be incorporated in mobile phones to identify callers.

As ear recognition is a recent subject in biometric recognition, there are only few algorithms found in the literature about the subject. The most robust are: Principal Component Analysis, also called eigenear (PCA)[41], Force Filed Transformation (FFT)[20], Iterative Closest Point (ICP)[96], neutral network, generic search, local surface shape descriptor and edge-based. In [96], Ping Yan and Kevin W. Bowyer conducted an empirical evaluation of these algorithms; from their experiments, they deduced that the ICP algorithm yields the best performance.

Despite the fact that ear is recognized by biometric scientists as a potential biometric signature, there still exists one major problem. In fact, ear recognition cannot be used in passive recognition, that is, without the participation of the person to be identified; the reason is that ears can be partially occluded by the hair or the hat, in which case, it will be difficult to capture discretionally the ear of a person with a camera. One solution to this problem is the use of the thermogram imagery to mask out of the captured ear image, the hair or the hat [41].

Hand Geometry

Research efforts in identifying people based on hand geometry have been done since a few decades. One result of these endeavours has been the use of hand geometry systems in controlling and protecting the access of people in the Olympic village, during the 1996 Olympic game. Moreover, one of the widely used application of the hand geometry technology is its use for time and attendance purposes in association with the time clocks by companies [82]. The common features used in hand geometry recognition are the following geometrical characteristics of the hand: length, width,

thickness, and the surface [3, 82, 86]. More often, the enrolment consists of capturing both the top surface of the hand and its side using a camera and an angled mirror [82]. Images captured are then analyzed and measurements of the geometrical parameters cited above are taken and stored as template. The verification of an identity consists of acquiring a hand of an individual and processing a new set of geometrical parameters. Thereafter, the new parameters are compared to the corresponding template parameters from the template database of hand characteristics. However, there is a problem in using the geometrical characteristics of the hand in hand geometry recognition. In fact, the measurement of the hand parameters is done by means of pegs; which dictate how the hand should be placed and fixed on a plate. As a consequence, the hand and the disposition of fingers are deformed at different instants [65]; which can affect considerably the quality of the feature to be acquired and degrade the performance of the hand verification. To overcome these drawbacks, techniques of hand geometry recognition based on the contours of the hand [86] have been developed.

Iris

The iris is a muscle within the eye that regulates the size of the pupil, controlling the amount of light that enters in the eye; it is the colored portion of the eye with coloring based on the amount of melatonin pigment within the muscle [28]. As defined, the iris is a complex organ with a great number of distinctive characteristics that make it a rich source of biometric data. The automated recognition of individuals based on their iris is a relatively young technique. In fact, the first iris recognition technology was introduced in the market only in 1994 [28]. In the literature, it is widely acknowledged that the iris recognition is becoming the best in term of efficiency and accuracy, when compared to face and fingerprint recognition. The reason is that, although the irises of a person are genetically identical, they are structurally distinct to each other and are unique to every living person; which make the iris a best candidate for biometric recognition [28, 57].

Practically, a subject being identified presents his/her eye to a high quality digital camera at the entrance of a sensitive space. The image of the eye is then acquired. Afterwards, the iris is located using landmark features. These landmark features and

the distinct shape of the iris allow imaging, feature isolation, and extraction [28]. The recognition is then carried out by comparing the training features to the templates stored in the database. Daugman's technique and Wildes' system are two of the earliest and the best known iris recognition algorithms [57] found in the literature.
Despite the success of this technique, there are still two main problems associated with it. The first one is the localization of the iris from an eye image acquired by a camera. In fact, if the localization of the iris is done improperly, resulting noise like eyelash, reflections, pupils, and eyelids may lead to poor performance [28]. The second problem is related to the placement of the camera. In fact, the recognition process works very well when the camera is placed indoor; however, the case where the camera is located outdoor remains a challenging problem, as the glare from the sun and the direction of the light can produce significant camera error rates [21].

Fingerprint

A fingerprint is a pattern of ridges and furrows located on the tip of each finger. It is one of the oldest biometric traits used to identify people. The techniques for extracting the patterns are continually being improved . Initially, patterns were extracted by creating an inked impression of the fingertip on a paper. Thereafter, compact sensors have been designed to produce digitalized images of patterns for automated identification. Meanwhile, the repeated use of a sensor has revealed some practical imperfections. In fact, the sensitivity and the reliability of a sensor can be reduced due to the fact that its surface can become oily and cloudy after repeated use. Then, solid state devices that sense the ridges of the fingerprint using electrical capacitance, have been designed to overcome these technical difficulties [29].
In an operational point of view, the images extracted by the sensors are used by the recognition module to compute the feature data that corresponds to the various relevant points of the acquired fingerprint, called minutiae. Each minutiae is defined by a position and an orientation in the feature data extracted. The recognition then consists of matching the acquired minutiae patterns with the stored minutiae template.

Fingerprint matching techniques can be grouped into two categories: minutiae-based

[29] and correlation based [72]. Minutiae-based techniques first find minutiae points and then map their relative placement on the finger. However, some difficulties arise when using this approach. It is difficult to extract the minutiae points accurately when the fingerprint is of low quality. Besides, this method does not take into account the global pattern of ridges and furrows. The correlation-based method is able to overcome some of the difficulties of the minutiae-based approach. Even with this technique, some imperfections persist. For instance, correlation-based techniques require the precise location of a registration point and are affected by image translation and rotation [72].

2.3 Face Recognition

2.3.1 Background

The face is the natural way human beings use to recognize each other. People identification based on face has been a subject of attraction for centuries. The aim is to find how a camera hidden somewhere in an open area or at the entrance of a building can capture a person's face image and send it to a connected computer system in order to identify the individual. This technique is simply called automated face recognition. The pioneer research endeavor in that area was carried out by Francis Galton in 1888. He proposed a face recognition technique consisting of collecting facial profiles as curves of an individual, finding their norm and classified them using their deviations from the norm to identify the person [92]. Thereafter, series of face recognition approaches that use normalized distances and ratio among features points have been proposed to model and classify faces. The drawbacks of these approaches were revealed by Carey and Diamond [59] who demonstrated that, features points and their relationships provide a lower identification performance with faces of adult persons. Since 1990, computer graphics and machine vision, automated face recognition has been of interest to a growing number of research groups [92]. As a result, several automated face recognition applications have been developed and integrated in public security, law enforcement and commerce, such as mug-shot database matching, identity authentication for credit card, passport, and driver license, access

control, information security, and intelligent surveillance [80]. Obviously, several algorithms have been proposed for these applications and can be classified into three categories:holistic methods, feature-based methods and hybrid methods [19, 38, 93]. Holistic methods are characterized by the use of the whole face image for recognition and are all based on principal component analysis (PCA) decomposition, which aim to reduce a higher dimensional training data set of face images to a lower dimensional one, when preserving the key information contained in the data set. Several face recognition algorithms fall in this category. The most popular of them are: eigenfaces, probabilistic eigenfaces, fisherfaces, support vector machine (SVM), nearest feature lines and independent component analysis (ICA) [38].
Feature-based approaches exploit the face regions/components such as nose, eyes or mouth for recognition. Some algorithms of this category include pure geometry, dynamic link architecture, Hidden Markov Model [38], elastic bunch graph matching and local feature analysis [80].Finally, the hybrid approach uses both local regions and the whole face. Modular eigenfaces, hybrid local features, and shape-normalized methods belong to this category [38].

2.3.2 Component-Based Face Recognition

Several previous studies show that feature-based approaches are faster and more robust against variation in face orientation and illumination than holistic techniques [19, 38, 80, 93]. However, one of the main problems with the component-based face recognition methods remains the automatic extraction and validation of face components without human intervention as well as any assumptions on the location of faces components.

In [11], a component based face detection system is presented. It uses two level Support Vector Machines (SVM) to detect and validate facial components. Learned face images are automatically extracted from 3-D head models that provide the expected positions of the components. These expected positions are used to match the detected components to the geometrical configuration of the face. Loulia and Veikko [34] propose a method for detecting facial landmarks. In their method, edge

orientations are used to construct edge maps of the image. The estimation of the orientation of local edges was done by means of a kernel with maximum response. The local oriented edges were extracted and grouped into regions representing candidates for facial landmarks. The detected candidates were further classified manually into noise or facial landmark categories. In [13] and [15], facial components are used to detect the face in an image. The components are either assumed to be the holes in the detected facial regions, features computed in given color spaces, or the darkest region of the face. A geometrical technique is used by Zoltan and Tamas [102] to detect and extract facial components. They compute the facial symmetry axis and use it to deduce the nose region based on the assumption that the region of the nose is the most vertically detailed region on a face. Afterward, the positions of the eyes and mouth are estimated from the chosen nose region. A facial segmentation method based on dialation and erosion operations is presented in [91]. Facial symmetry and relative positions among the facial features are used to locate the face contour, mouth, nostrils and eyes. Tian and Bolle [84] present a method of detecting a neutral face. In their approach, six facial points are chosen as being the most reliable that could be extracted from a face. Thereafter, the normalized distances between them are computed and used as discriminating features. Two preprocessing operations named Skin Color Similarity Map (SCSM) and Hair Color Similarity Map (HCSM) are employed in [97] to compute the coordinates of face and head regions. The SCSM is projected onto the x-axis to determine the x-coordinate of the facial region. The y-coordinate of the face and head regions are determined by projecting the SCSM and HCSM on to the y-axis. Afterward, the positions and the sizes of the facial features are estimated based on the computed coordinates of the face. In [12], a method of detecting facial features such as eyebrows, eyes, nose, mouth, and ears is suggested. Facial features are determined by searching for minima in the grey value relief of the segmented facial region based on the assumption that each facial feature generates a minimum in the projection of the grey value relief or pixel grey level and the expectation that eyebrows, eyes, nostrils, mouth, and chin are ranked respectively as the first, second, third, fourth, and fifth significant minimum on the horizontal relief. Furthermore, as the number of minima is usually greater than the number of features, a geometrical technique is employed to get clues about the relative positions of facial features.

Heisele et al. [10] have proposed a method that enables to automatically learn face components for detection and recognition. Initially, an object window of fixed size is slid over the input image. Afterward, 14 referenced points are manually selected in the object window based on their 3D correspondences from a morphable model. The learning algorithm then iteratively grew small rectangles around the manually preselected reference points. The detection of facial components was carried out by searching for maximum output within the rectangular region around the expected location of the component with component classifiers of linear SVM. In the successful case, the result of the approach yields the detection of both eyes and the mouth. The technique employed in [10] is used in [14] for face detection. Furthermore, the coordinates of the position of the maximum output of each component classifier is recorded along with the value at that position. Then, each detected component is represented by a triplet formed from x and y coordinates as well as the value of the related position. Thereafter, the set of triplets is used as input to the higher level classifier for classification. The output of the upper-level classifier is recorded in the final resulting image. The approach is further applied to detect eyes. The Haar algorithm is applied on frontal faces in [7]. Each facial image is divided into five blocks that are further used as feature vectors to a one-class SVM classification.

The above approaches either involve human intervention in detecting and extracting facial components, employ geometrical considerations and/or assumptions about the location of face components. Furthermore, most of these approaches perform only with faces taken in frontal view. We further remark that, in some cases, components are not detected accurately (detected region larger than the component, detected region includes two components e.g. the eye region includes nose, the mouth region includes nose, etc.) or the detection is limited to only certain components such as both eyes and mouth as in [10] and eyes as in [14]. Also, the large number of techniques employed to learn the location of facial components lead to expensive computational time. An adaptive approach of detecting facial components in different orientations of the face is proposed in chapter 4 as a solution to the above mentioned problems.

2.4 Conclusion

In this chapter we have provided a comprehensive description of biometric recognition based on the most popular human biometric traits. A great emphasis was placed on the face characteristics, particularly on the state of the art of the component-based face recognition. Pathways of our research have been presented with the identification of the shortcomings of the previous works in component-based face recognition. In the next chapter, we define and describe algorithms and techniques of image processing and computer vision, that we have used in our investigation.

Chapter 3

Materials and Methods

3.1 Introduction

In general, face recognition involves three main steps: segmentation/detection of face in a scene, feature extraction, and classification/recognition. This chapter presents some key algorithms used in face recognition for feature extraction and classification/recognition. The chapter commences by reviewing some common operations performed in image processing and computer vision, that have been used in this work.

3.2 Basic Operations and Entities

3.2.1 What is a Pixel ?

An image is a matrix of elements called pixels. A pixel is the smallest entity of an image. Fig.3.1. shows a pixel in an image array. A pixel in an image is characterized by two kinds of information: its position represented by its x and y coordinates in the space spanned by the image, and its value or gray level. The coordinates and the gray level of a pixel are called the feature of the pixel, meaning, the parameters that characterize that pixel uniquely in the image.

3.2.2 Segmentation and Binarization

Segmenting an image is an important step in many image processing problems. The segmentation is the process of partitioning a digital image in several regions or groups of related pixels. The relationship amongst pixels of a region is defined by parameters such as color, intensity, or texture of the region. The resulting image then provides more meaningful information that is useful in analyzing the image. Several algorithms

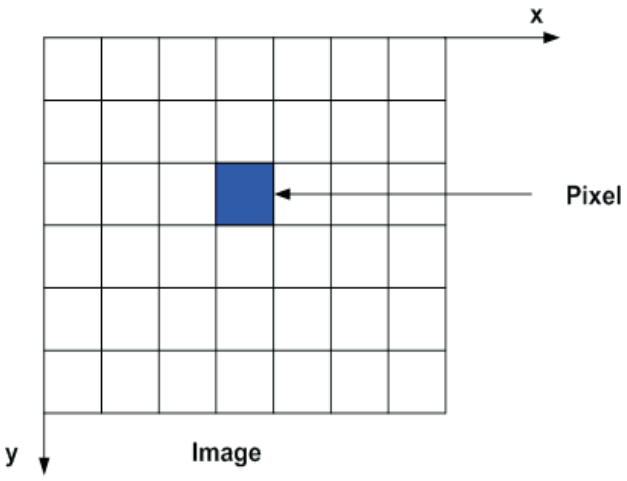


Figure 3.1: A Pixel in an Image Array.

and techniques have been developed for image segmentation. They can be classified into three main groups: thresholding, edge-based, and region-based [35, 62].

The thresholding, also called binarization, separates an image into two parts: the relevant part or foreground and the irrelevant part or background. It refers to setting all the gray levels below a chosen threshold to zero and the gray levels above the threshold to one. Mathematically, the thresholding or binarization transforms an image f to an output binary image or segmented image g as follows [62]:

$$g(i,j) = \begin{cases} 1 & \text{if } f(i,j) \geq T \\ 0 & \text{otherwise,} \end{cases} \tag{3.1}$$

where T is the threshold, $g(i,j) = 1$ for image elements of the foreground, and $g(i,j) = 0$ for image elements of the background. Fig.3.2. shows an original image along with the corresponding binarized version.

3.2.3 Connected Components

Within an image, a pixel (x,y) has four horizontal and vertical neighbors $(x+1,y)$, $(x-1,y)$, $(x,y+1)$, and $(x,y-1)$. This group of pixels defines what is called the

Figure 3.2: Exemple of Image Binarization:(a):Original Image; (b):Binarized Image.

4-connectivity of the pixel (x, y). The pixel (x, y) also has four diagonal neighbors $(x+1, y+1), (x+1, y-1), (x-1, y+1)$, and $(x-1, y-1)$. The four diagonal neighbors of the pixel (x, y) together with its four horizontal and vertical neighbors define its 8-connectivity.

One can establish whether two pixels in an image are connected by determining if they are adjacent, that is, they are 4 or 8-connected, and their gray levels satisfy a predefined criterion of similarity. The criterion could be the equality of their gray levels. Fig. 3.3. shows examples of 4 and 8-connectivity of a the center pixel.

Connectivity between pixels is an important concept in image processing. It is useful in establishing objects boundaries and components of regions in an image [35]. Once the regions are determined, they are further differentiated by assigning a unique label to pixels of each region. The labelling of regions builds what is called connected components or regions with similar pixels in the image. A simple example of connected component finding is shown in Fig.3.4. The application of 4-connectivity yields four connected components, whereas the 8-connectivity yields one connected component. The connected component algorithm is more often applied on binary images but could also be applied to gray level images . Fig.3.5. depicts the result of the connected component algorithm on a gray scale face image.

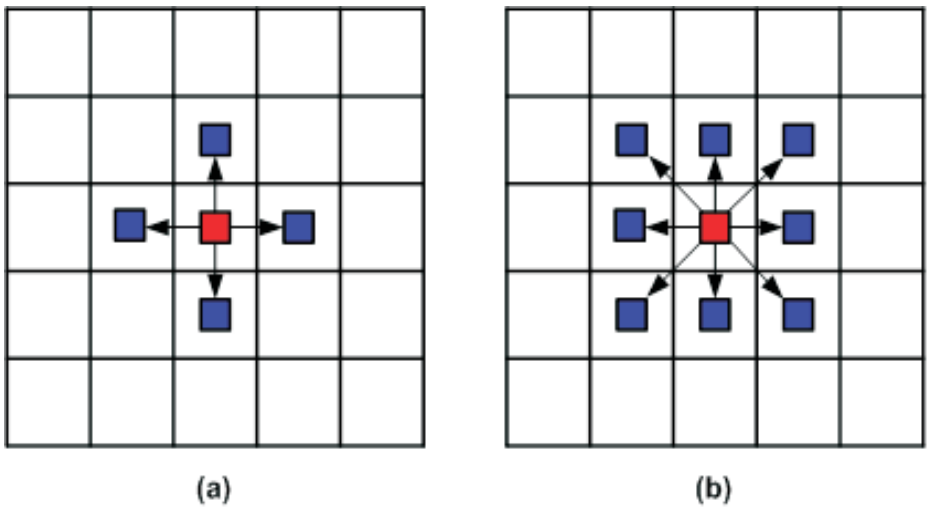


Figure 3.3: Connected Components:(a):4-Connectivity of the Center Pixel; (b):8-Connectivity of the Center Pixel.

3.2.4 Convex Hull

In certain situations when solving image processing problems, the shape of a set of points in the image can give some clues about the step forward. Computing the shape of a set of points in an image could be done by means of a Convex hull. The convex hull of a set of points S in n dimensions is defined as the smallest convex polygon containing S. A polygon is said to be convex if, for each couple of points A and B belonging to the polygon, every point on the line segment connecting A and B is in the polygon. Fig.3.6.(a) is an example of convex hull, whereas Fig.3.6.(b) is not. In Fig.3.6.(b), all the points on the line segment connecting F and G are not in the polygon ABCDE. Fig.3.7. shows the convex hull of the centroid of the facial components: the two eyes, the nose and the mouth. It is a triangle with the vertices situated at the side of the two eyes and the mouth.

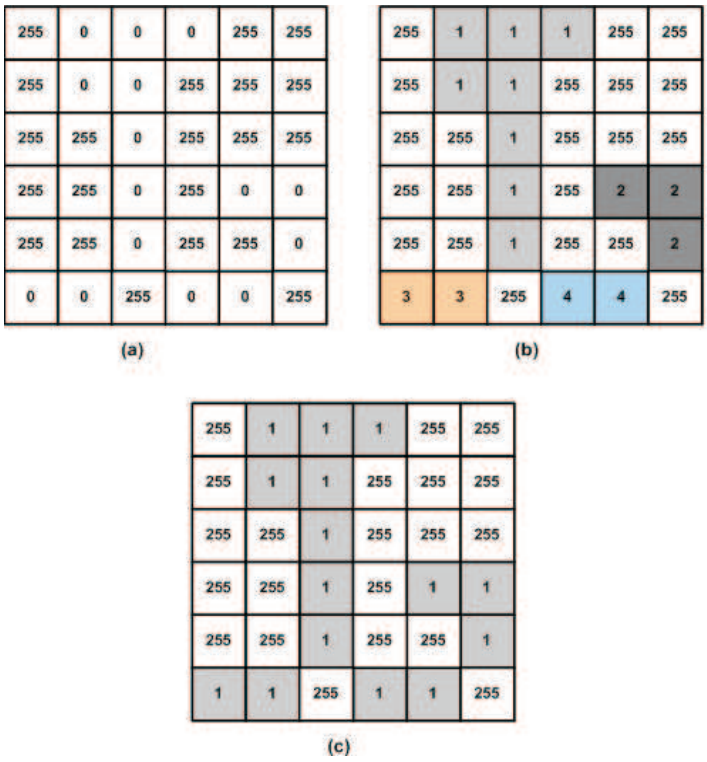


Figure 3.4: Practical Example of Finding Connected Components in an Image:(a):Binary Image; (b):Four Connected Components found with 4-Connectivity; (c):One Connected Component found with 8-Connectivity.

3.2.5 Similarity Measures

Pattern recognition tasks requires comparing features of sample patterns to be recognized with those of templates from the database. The common way to accomplish the comparison is the use of similarity measures between sample and template features. The recognition could then be carried out by inspecting the distance obtained. Three similarity measures are commonly used in pattern recognition: the $City-block$ or $Manhattan$ distance, the Euclidian distance, and the $Mahalanobis$ distance [2, 61, 85, 87]. The first two are deduced from the $Minkowski$ distance of order p in an Euclidian space R^N of dimension N. Let's consider two vectors $X = (x_1, x_2, ..., x_N)$ and $Y = (y_1, y_2, ..., y_N)$. The $Minkowski$ distance of order p denoted $L_P(X, Y)$ is defined as follows:

Figure 3.5: Example of Connected Components of a Binary Image ; (a):Original Image; (b):Binary Image;(c):Connected Components of the Binary Image.

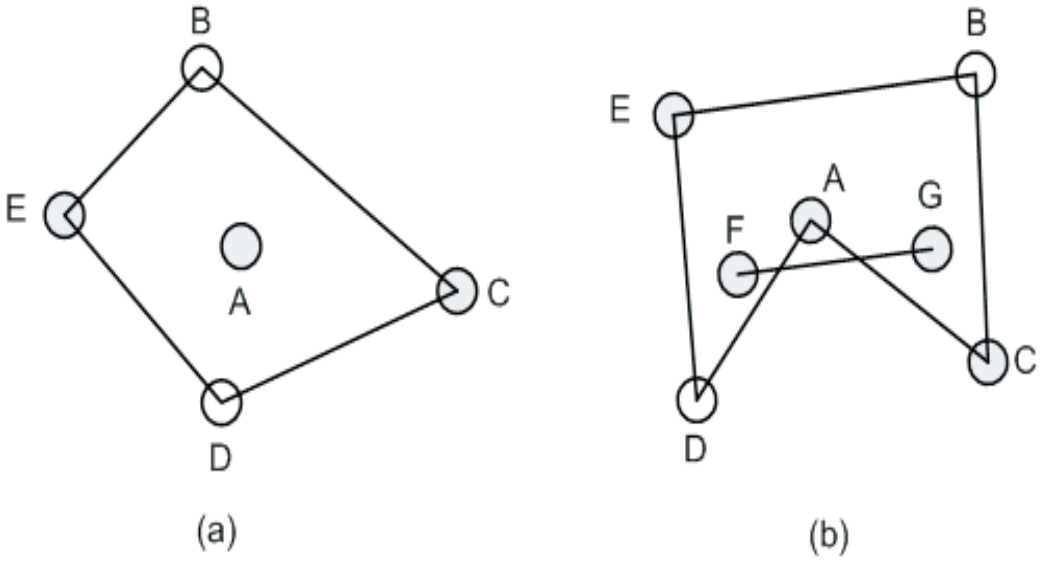


Figure 3.6: Example of a Convex and Non-convex Polygon: (a):Convex Hull of the Set of Points A,B,C,D,E; (b):Non-convex Hull Polygon of the Set of Points A,B,C,D,E.

$$d(X,Y) = L_p(X,Y) \quad = \quad \left(\sum_{i=1}^{N} |x_i - y_i|^p\right)^{\frac{1}{p}} . \tag{3.2}$$

From equation 3.2, most of the similarity measures used in pattern recognition

Figure 3.7: Example of Convex Hull of the Centroid of Face Components: Eyes, Nose and Mouth.

could be derived. For $p = 1$, the $City - block$ or $Manhattan$ distance is written as follows:

$$d(X,Y) = L_1(X,Y) \quad = \quad \sum_{i=1}^{N} |x_i - y_i| \ . \tag{3.3}$$

.

For $p = 2$, one obtains the euclidian distance:

$$d(X,Y) = L_2(X,Y) \quad = \quad \sqrt{\sum_{i=1}^{N} (x_i - y_i)^2} \ . \tag{3.4}$$

.

The Euclidian measure assumes that all the components of the vectors X and Y contribute equally to the similarity measure [2]. However, the performance of this distance measure can be greatly improved if an expert knowledge about the nature of the data is available. If it is known that some values in the features vector hold more discriminatory information with respect to others, it is possible to assign proportionally higher weights to such vector components in order to influence the final outcome of the similarity measure [87]. Thus, the weighted Euclidian distance is defined as:

$$d(X,Y) = L_2(X,Y) \quad = \quad \sqrt{\sum_{i=1}^{N} w_i (x_i - y_i)^2} \ , \tag{3.5}$$

where w_i is the weighting component for the ith vector component.
The Mahalanobis distance is defined as follows [68, 87]:

$$D_M(X) \quad = \quad \sqrt{(X-\mu)^T \Sigma^{-1} (X-\mu)} \ , \tag{3.6}$$

with mean $\mu = (\mu_1, \mu_2, ..., \mu_p)$ and Σ the covariance matrix of the multivariate vector $X = (x_1, x_2, ..., x_P)$. The Mahalanobis distance is scale invariant and it takes into account the correlations of the data set [2].

3.3 Feature Extraction Techniques

In this work, we automatically detect the key facial components that are: the eyes, nose and mouth, from an input face image. Thereafter, the textures of the eyes and the shapes of the nose and the mouth are extracted to characterize the individual. The eyes textures and the shapes of the nose and the mouth are the feature data or simply features, that will be used for recognition. The process carried out from the acquisition of the face image to obtain the feature data is called feature extraction. In biometric recognition, the feature extraction consists of applying algorithms on image pixels to compute features that characterize the image. An image with a feature vector computed by mean of principal component analysis (PCA) algorithm is depicted in Fig.3.8.

3.3.1 Eigenface Decomposition

A considerable among of work has been done in face recognition using eigenface decomposition. A comprehensive literature on this technique could be found in [59, 60, 61]. eigenface is the most popular dimensionality reduction technique used in face recognition. It is often used for feature extraction and classification. Its principle

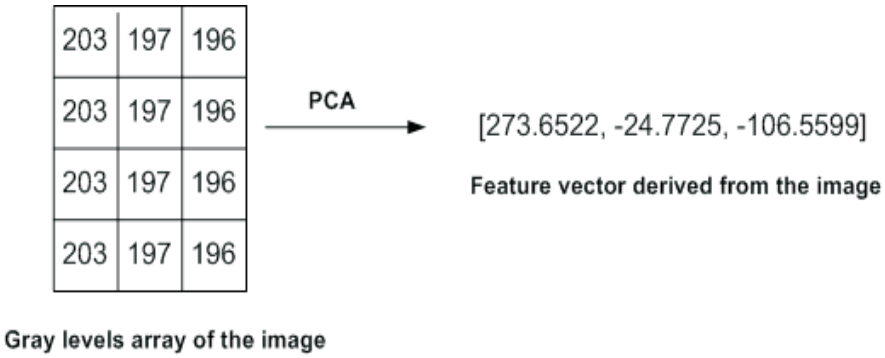


Figure 3.8: Sample Feature Vector derived from the Matrix of Gray Levels of an Image.

can be summarized as follows. The input to the algorithm is a training set of face images, all centered and with the same size. Each image I is represented as an $N \times N$ vector Γ_i obtained by concatenating the columns of the image matrix. Thereafter, the average face Ψ is computed as:

$$\Psi = \frac{1}{M}\sum_{i=1}^{M}\Gamma_i \ , \tag{3.7}$$

where M is the number of face images in the training set. The mean face is used to adjust or normalize each face as:

$$\Phi_i = \Gamma_i - \Psi \ . \tag{3.8}$$

The mean face images allow the computation of the covariance matrix C as:

$$C = \frac{1}{M}\sum_{i=1}^{M}\Phi_i\Phi_i^T = AA^T \ , \tag{3.9}$$

where $A = [\Phi_1, \Phi_2 ... \Phi_M]$.
eigenvectors u_i and corresponding eigenvalues λ_i of the covariance matrix C can be evaluated by using a Singular Decomposition method(SDM)[55]:

$$Cu_i = \lambda_i u_i \ . \tag{3.10}$$

Because matrix C is usually very large ($N^2 \times N^2$), evaluating eigenvectors and eigenvalues is computationally very expensive. Instead, eigenvectors v_i and corresponding eigenvalues μ_i of the matrix $A^T A(M \times M)$ can be computed. After that, u_i can be deducted from v_i as follows:

$$u_i = Av_i, j = 1, ..., M \ . \tag{3.11}$$

The dimensionality reduction is done by selecting only a smaller number of eigenvectors $K(K << M)$ corresponding to the largest eigenvalues. The selected eigenvectors form the eigenspace or face space. A new face image Γ, after subtracting the mean ($\Phi = \Gamma - \Psi$) can then be reconstructed in eigenspace by the formula:

$$\tilde{\Phi} \ = \ \sum_{i=1}^{K} w_i u_i \ , \tag{3.12}$$

where $w_i = u_i^T \Psi$ are coefficients of the projection and can be considered as a new representation of the original face in the eigenspace.

3.3.2 Gabor Filters

Gabor Filters are a powerful tool for texture analysis. They have been widely used by many authors, for feature extraction in recent years. In fact, it has been proven that the Gabor receptive field can extract the maximum information from local image regions [37, 83] and that, when appropriately designed, Gabor features are invariant against illumination, translation, rotation, and scale[71, 83, 100]. This makes it an ideal technique for face recognition. Furthermore, Gabor filtering has been successfully deployed in many applications including texture analysis, character recognition, fingerprint recognition, and face recognition [71]. The formulation of Gabor Filters used in this dissertation is adopted from [37] and [100] . In the spatial domain, a Gabor Filter is a complex exponential modulated by Gaussian function, which can be defined as follows:

$$\Psi(x, y, \omega, \theta_k) = \frac{1}{2\pi\sigma^2} e^{-(\frac{x'^2+y'^2}{2\sigma^2})} [e^{i\omega x'} - e^{-(\frac{\omega^2\sigma^2}{2})}] , \quad (3.13)$$

where $x' = x\cos\theta_k + y\sin\theta_k$, $y' = -x\sin\theta_k + y\cos\theta_k$.

(x, y) denotes the pixel position in a face image, ω is the radial center frequency, θ_k shows the orientation of Gabor Filter, and σ stands for the standard deviation of the Gaussian function along the x- and y-axes. A rotation of the $x - y$ plane by an angle θ_k results in a Gabor Filter at orientation θ_k. θ_k is defined by:

$$\theta_k = \frac{\pi}{n}(k-1) \; k = 1, 2, ..., n , \quad (3.14)$$

where n denotes the number of orientations. The maximum value of the frequency used by many authors is $\omega_{max} = \pi/2$ [37, 63, 100]. Then, the relationship defining the different frequencies is given by the equation:

$$\omega_m = \omega_{max}\lambda^{(m-1)} , \quad (3.15)$$

where $m = 1, 2, ..., 5$, and $\lambda = \sqrt{2}$ [37, 63, 83] is the spacing factor between different frequencies. According to [37] and [100], the relationship between σ and ω is:

$$\sigma \approx \pi/\omega . \quad (3.16)$$

3.3.3 Zernike Moments

Many feature extraction methods for image analysis based on moments have been used in the recent years [6, 30, 81]. These methods encompass: Hu Moments, Legendre Moments, and Zernike Moments [81]. The main characteristic of these moments is that they are translation, scale, rotation invariant and are robust in the presence of noise. Hence, they may be chosen for image analysis and pattern recognition applications. Zernike Moments are a class of orthogonal moments that are effective in representing images, based on orthogonal Zernike radial polynomials. These moments are effectively used for pattern recognition since their rotational invariance can

be easily obtained at an arbitrary order [81]. The algorithm of Zernike Moments implemented in this work is adopted from [6]. For a discrete image function $I(i,j)$ with spatial dimension $M \times N$, their Zernike Moments of order n with repetition l are given by:

$$A_{nl} = \frac{n+1}{\pi} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} I(i,j).R_{nl}(r_{ij}).e^{-il\theta_{ij}} , \tag{3.17}$$

where the discrete polar coordinates r_{ij} and θ_{ij} are respectively defined as follows:

$$r_{ij} = \sqrt{x_j^2 + y_i^2} \tag{3.18}$$

$$\theta_{ij} = \arctan(\frac{y_i}{x_j}) . \tag{3.19}$$

The Cartesian coordinates x_j and y_i are given by:

$$x_j = c + \frac{j.(d-c)}{N-1} \quad 0 \leq j \leq N-1 \tag{3.20}$$

$$y_i = d - \frac{i.(d-c)}{M-1} \quad 0 \leq i \leq M-1 , \tag{3.21}$$

where c and d are real numbers chosen according to whether the image function is mapped outside or inside a unit circle. Outside a circle $c = -1$ and $d = 1$. Inside a circle, $c = -1/\sqrt{2}$ and $d = 1/\sqrt{2}$. The real values radial polynomials $R_{nl}(r)$, are given by:

$$R_{nl}(r) = \sum_{s=0}^{\frac{n-|l|}{2}} (-1)^s \frac{(n-s)!}{s!(\frac{n+|l|}{2}-s)!(\frac{n-|l|}{2}-s)!} , \tag{3.22}$$

where $|l| \leq n$ and $n - |l|$ is always even.

3.4 Learning and Recognition

Learning in pattern recognition consists of extracting a priori knowledge or statistical information from patterns. The patterns to be classified or recognized are usually groups of measurements or observations. Learning can be supervised or unsupervised. A learning is said to be supervised if the different classes of patterns are known *a priori* and if the learning task is guided by a supervisor. The supervisor means the designer who indicates the label of the class to which each candidate pattern sample belongs. For example, in our case, the learning in face recognition is a supervised learning because patterns to be recognized are labelled features of acquired face images. Learning is unsupervised when there is no *a priori* labelling patterns given to the system; instead, classes are established based on the statistical regularities of the pattern. Several learning and recognition algorithms are used for face recognition. We present in the following sections learning and recognition methods used in our study.

3.4.1 Nearest-Neighbor

The nearest neighbor is a classification technique which is used in pattern recognition to compare feature vectors of sample patterns to template vectors stored in a database. The comparison is done by computing the distance between the sample patterns and the templates from the database. Templates are grouped into classes of similar data. Let $C_1, C_2, ..., C_k$ be such classes of data in the database. The class $C_j (j = 1, ..., k)$ of a new sample X is found by measuring the distances $d(X, C_j)$ between X and the centers of all the template classes. The feature vector X is assigned to the class to which it is more closer. This closest class is defined as:

$$C_k \quad = \quad \arg min_{(C_i)_{i=1,...,k}} d(X, C_i) \ . \tag{3.23}$$

3.4.2 K-means Clustering

Clustering involves splitting a set of data into classes or clusters, such that, data belonging to the same class are alike and those belonging to two different classes

are different. Thus, clustering algorithms are appropriate for pattern classification/recognition. Some clustering methods are: K-means, fuzzy C-means, hierarchical, and agglomerative algorithms [2, 46, 64, 73]. K-means method has been shown to be effective in producing clustering results for many practical applications. Basically, the K-means clustering algorithm starts with some known parameters as the data set, the number of clusters, and the randomly chosen cluster centroids. Thereafter, data points are iteratively assigned to clusters until reaching a predefined stopping criterion. The stopping criterion could be the convergence point of the algorithm or a predefined maximum number of iterations. At each iteration, new data points are assigned to clusters based on the one nearest neighbor rule of the data points to the cluster centroids, and the class centroids are updated based on the resulting clusters. Mathematically, the K-means algorithm [2, 46, 64, 73] could be described as follows. Let's consider $X = \{x_1, x_2, ...x_N\}$, a set of N patterns to be clustered. The K-means algorithm aims to minimize an objective function J with variables U and C by partitioning X into K clusters:

$$J(U, C) = \sum_{l=1}^{k} \sum_{i=1}^{N} \mu_{il} d(x_i, c_l) \tag{3.24}$$

subject to the condition

$$\sum_{l=1}^{k} \mu_{il} = 1, 1 \leq i \leq N \ , \tag{3.25}$$

where

U is an $N \times K$ matrix and μ_{il} is a binary variable. Equation 3.34 indicates that pattern i is in the cluster l.

$C = \{c_1, c_2, ..., c_k\}$ is a set of K prototypes representing the K clusters.

$d(x_i, c_l)$ is a similarity measure (for example the Euclidean distance) between object i and prototype l.

The above optimization problem is solved iteratively using the following algorithm:

First, set the number of clusters K.

Second, randomly initialize the cluster centroids c_i for $i = 1, ..., K$.

Third, assign the pattern x to the nearest cluster c_i. $\mu_{il} = 1$ if $d(x_i, c_l) \leq d(x_i, c_t)$ for $1 \leq t \leq K$; otherwise $\mu_{il} = 0$ for $t \neq l$.
Fourth, update the centroids c_i

$$c_l = \frac{\sum_{i=1}^{N} \mu_{il} x_i}{\sum_{i=1}^{N} \mu_{il}} \tag{3.26}$$

for $1 \leq l \leq K$, and
Fifth, repeat steps 3 and 4 until the centroids do not change or the maximum number of iterations have been reached.

The drawback of the K-means algorithm is that, it is sensitive to the initial configuration. In fact, as every iteration of the algorithm improves on the previous configuration, the solution depends on the initial configuration. However, it converges quickly and is computationally efficient.

3.4.3 Support Vector Machine (SVM)

Support Vector Machine is a recent classification technique that is used more and more for face recognition and has been recognized to be more powerful than previous approaches such as principal component analysis (PCA), neural network and example-based learning [49, 90]. The idea of SVM is to separate geometrically a data set (see Fig.3.9) into classes, using an optimal separating hyperplane (OSH), in the input space. The OSH is the hyperplane that minimizes the upper bound on the expected classification error [49] or the generalization error [90], thus providing the best generalization capabilities. Each class is assigned a unique label. SVM of two

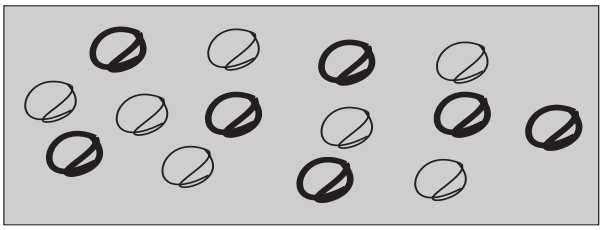


Figure 3.9: Sample Training Data Set. For example: Face and Non-face Components Features Vectors ($F_+ and\ F_-$).

data classes with opposite labels sign is widely used for classification. Moreover, the data points are assumed to be linearly separable. The data set is defined as:

$$(x_i, y_i) \in \mathbb{R}^N \times \{1, -1\}, i = 1, 2, ..., l \ , \tag{3.27}$$

where x_i is a data point to be classified and y_i the label of its potential class.

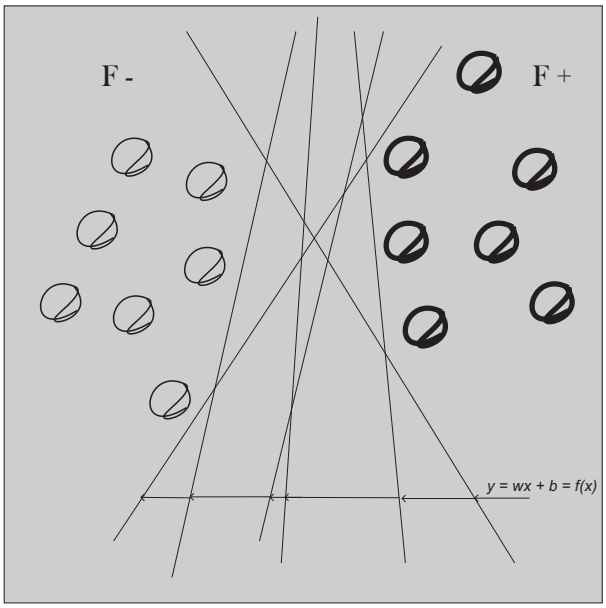


Figure 3.10: Example of Hyperplanes Separating Two Classes of Feature Points.

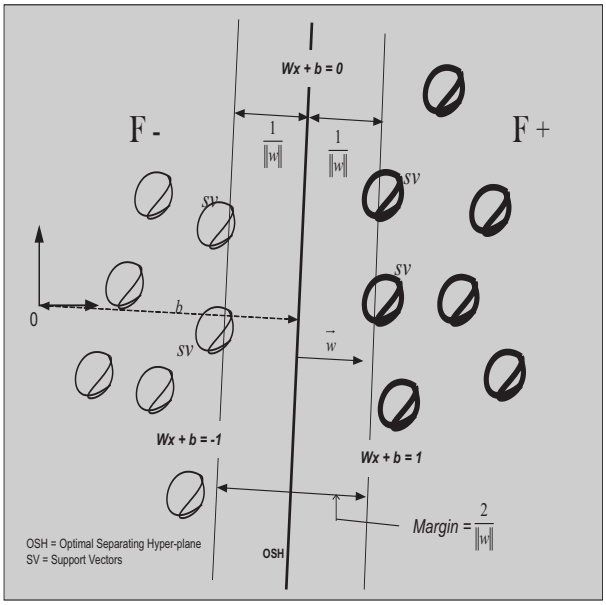


Figure 3.11: Optimal Separating Hyper-plane, Margin, Support Vectors.

SVM technique searches for the OSH among the possible hyperplanes separating the classes in the input space (see Fig.3.10). The equation of all possible hyperplanes is given in equation 3.28.

$$f(x) \quad = \quad (w.x) + b, \ w \in \mathbb{R}, b \in \mathbb{R} \ . \tag{3.28}$$

The classification of a new data point x is done based on the decision function:

$$f(x) \quad = \quad sgn((w.x) + b) \ . \tag{3.29}$$

In other words, the sign of the function f determines the label of the class of the new data point x. The OSH corresponds to $f(x) = 0$ as shown in Fig.3.11, that is:

$$(w.x) + b \quad = \quad 0 \ . \tag{3.30}$$

The data points located on the hyperplane situated at the same distance from the OSH are called support vectors (see Fig.3.11) [18, 33, 58]. These data points are called support vectors because they are the data which provide more information about the localization of data classes in the input space. The support vectors allow computing what is called the margin. The margin (see Fig.3.11) is the perpendicular distance between the OSH and the hyperplane through the support vectors. In other words, the margin is the region between the hyperplane on both sides of the OSH [18, 58]. The margin is defined as follows:

$$margin \quad = \quad \frac{2}{\|w\|} \ . \tag{3.31}$$

The importance of the margin is that, it dictates the choice of the OSH. In fact, the OSH is chosen such that the value of the corresponding margin is the maximum among all possible margins. The classification of a new data point x then consists of finding to which data class it belongs. This is done by calculating its distance from the OSH. In [36], Bernd et al. have explained that, the reliability of the classification depends on the length of that distance. In other words, the larger the distance

between the new data point and the OSH, the more reliable the classification is.
When the data points are not linearly separable, a kernel function is used to map them in a higher dimensional space called feature space [16, 18, 36, 49, 58, 67, 90]. The decison function in equation (3.28) is shown [67] to take the form:

$$f(x) \quad = \quad sgn(\sum_{i=1}^{m} \alpha_i y_i k(x, x_i) + b) \ , \tag{3.32}$$

where $k(x, x_i)$ is the kernel function. The coefficients α_i and b are determined by solving the quadratic programming problem:

$$\max_{\alpha \in \Re^m} w(\alpha) = \sum_{i=1}^{l} \alpha_i - \frac{1}{2} \sum \alpha_i \alpha_j y_i y_j k(x_i, x_j) \tag{3.33}$$

subject to the constraints: $\sum_{i=1}^{l} \alpha_i y_i = 0, 0 \leq \alpha_i \leq C$, for $i = 1, ..., l$.
The parameter C is called the regularization parameter and is selected by the user.

3.4.4 Recognition, Authentication and Identification

Basically, a biometric recognition system performs three main operations. These include the acquisition of biometric data from an individual, extraction of feature from the acquired data and comparison of the feature against the template in the database. From an operational point of view, a biometric system may operate in two different modes: the verification mode or the identification mode [1, 43, 77]. In the verification mode, the identity of an individual is authenticated. The authentication is done by comparing the individual's features only with his/her template(s) in the database [43]. In other words, the system conducts a one-to-one comparison to determine whether the claimed identity is true or not. In the identification mode, the biometric system recognizes a person by searching the entire template database; meaning that, it conducts a one-to-many comparisons to establish the identity of an individual.
In general, it is more difficult to design an identification system than to design a verification system [43]. In fact, only a one-to-one comparison provides a good speed or response time of a verification system. Here, the major challenge remains the accuracy of the system. On the other hand, both accuracy and speed are critical

for an identification system. The identification system needs to explore the entire database to establish an identity. Thus, more requirements are imposed on the feature extraction and matching techniques employed.

Some biometric approaches are more suitable for operating in the identification mode than the others. For example, fingerprint is suitable for biometric verification, whereas the face is appropriate for the identification [43]. Furthermore, it is feasible to design a face recognition system operating in the identification mode, because face comparison requires less expensive operations and there exist efficient indexing techniques that the performance have been demonstrated [43].

3.5 Conclusion

Image processing methods and algorithms used in this research have been defined and described. The various operational modes of a biometric system have been discussed. The next chapter provides detailed information about their use for components-based face recognition. The structure of the entire system and its sub-components are presented as well.

Chapter 4

System Overview

4.1 Introduction

This chapter presents the design of our entire face recognition system. Furthermore, it sequentially provides details about the implementation of the various modules of the system. The first module of the system adaptively detects facial components: eyes, nose and mouth. The detected components are further validated [24] to ensure that they are the appropriate components required for recognition. Thereafter, Gabor Filters are applied on the two eyes to extract their texture features. Similarly, the shape of the nose and the mouth are computed by means of Zernike Moments. Both partial feature vectors are further concatenated and normalized by the mean and the standard deviation to produce the final feature vector of the input face image. The chapter ends with the description of the identification technique.

4.2 Structure of the System

Our system is made of two main modules: feature extraction and classification/recognition. Feature extraction is carried out in two steps. First, the facial components (the two eyes, the nose and the mouth) are detected and validated. Thereafter, Gabor Filters and Zernike Moments features are computed and fused to form the signature of the input face image. In the training phase, the final feature vector obtained is stored in the database. The recognition consists of comparing the feature vector of the candidate's face to the features of known faces previously learned. Fig. 4.1 shows the flowchart of the entire system. Details about each component of the flowchart are provided in the subsequent sections.

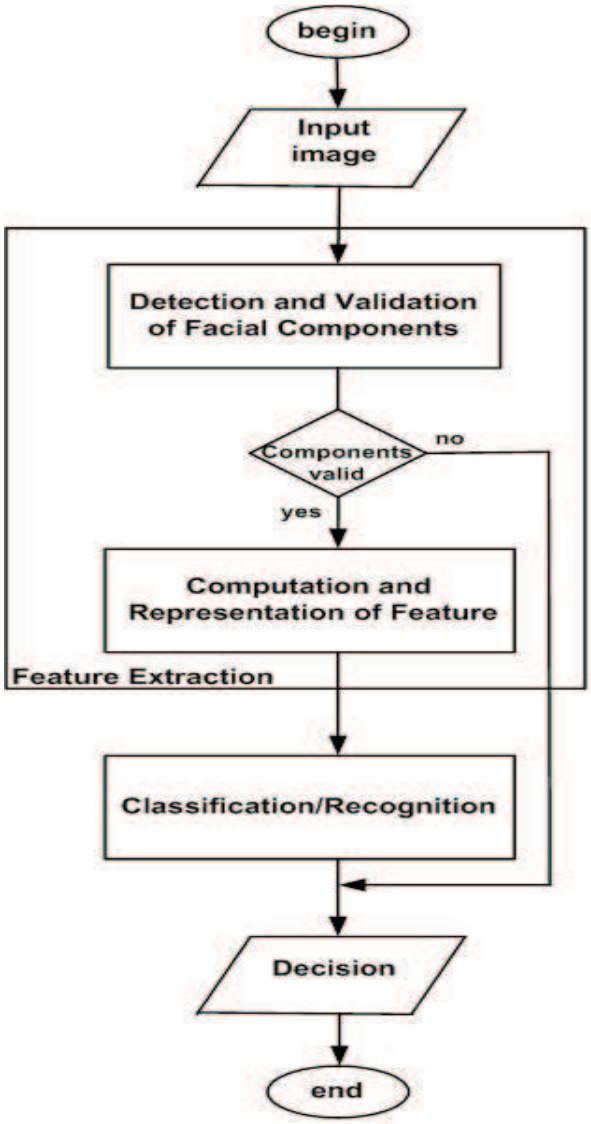


Figure 4.1: Flow Chart of the Entire Face Recognition System.

4.3 Adaptive Detection of Regions of Interest

Description of the System

The components detection and validation module consists of five sub-modules: facial components detection, coordinates computation, bounding box extraction, convex hull computation, and detected components validation. The flowchart of the complete process is presented in Fig. 4.2.

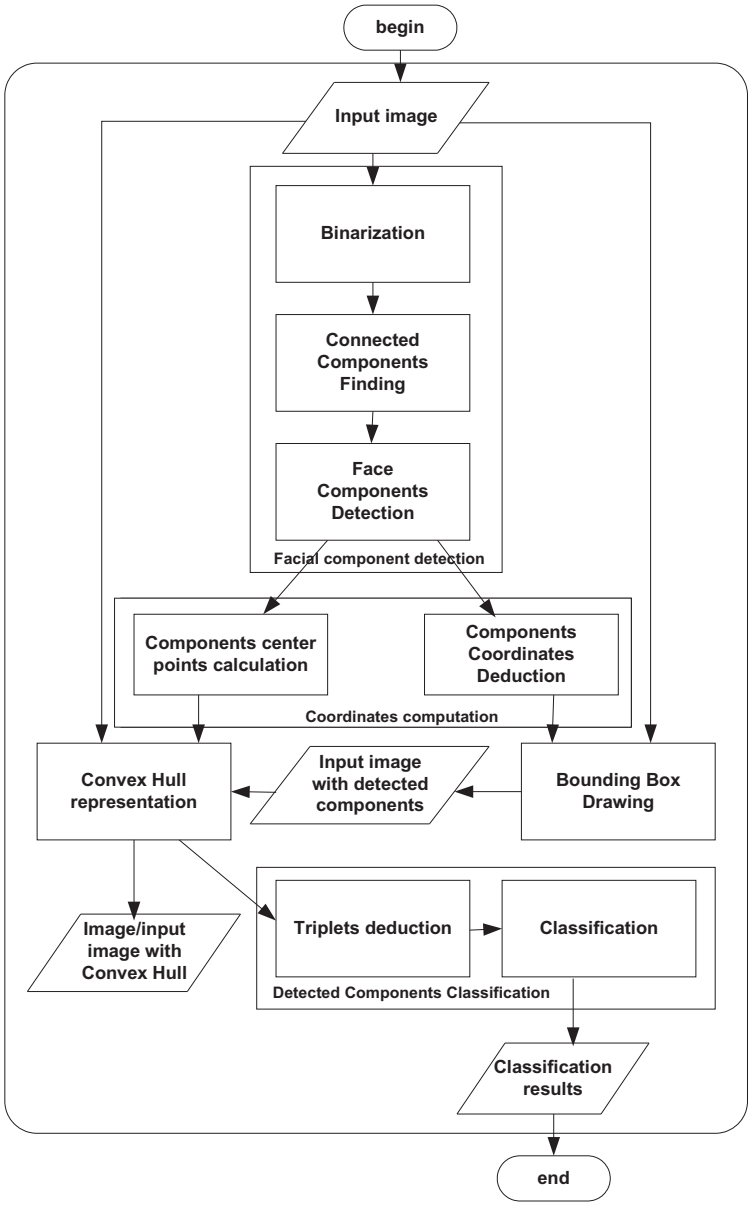


Figure 4.2: Flowchart of the Components Detection and Validation System.

The Binarization Method

The first step of our regions of interest detection is the binarization of the input face image. The binarization process we use is based on the algorithm adopted from [76]. The standard deviation and the mean are used to compute a threshold that is used to split the image into two parts: the foreground and the background.
An image I_{mn} with m rows and n columns, is defined as:

$$
\begin{aligned}
I_{mn} \;=\; & \Big\{(i,j\ ,\ x_{ij}),\ x_{ij} \in \{0,1,2,...,255\} \\
& 0 \le i \le m-1,\ 0 \le j \le n-1, \\
& where\ n,m \in \mathbb{Z}\Big\}\ .
\end{aligned} \tag{4.1}
$$

The mean of the image is defined as follows:

$$
\mu \;=\; \frac{1}{mn}\sum_{i=1}^{m}\sum_{j=1}^{n} x_{ij}\ . \tag{4.2}
$$

The standard deviation of the image is given by the formula:

$$
\sigma \;=\; \sqrt{\frac{1}{mn}\sum_{i=1}^{m}\sum_{j=1}^{n}(x_{ij}-\mu)^2}\ . \tag{4.3}
$$

Given σ and μ, the threshold is computed as follows:

$$
\tau \;=\; k_1\sigma + k_2\mu\ . \tag{4.4}
$$

In [76], it is specified that the values of k_1 and k_2 should be chosen between 0 and 2, depending on the resolution quality needed. In our work, we achieved a good binarization result with $k_1 = k_2 = 0.5$.

Using τ, the obtained binary image can be defined as:

$$
\begin{aligned}
B_{mn} \;=\; & \{B_{ij},\ \text{where}\ , 0 \le i \le m-1 \\
& \text{and}\ 0 \le j \le n-1\}
\end{aligned} \tag{4.5}
$$

is extracted from the image I_{mn}, where B_{ij} is defined as follows:

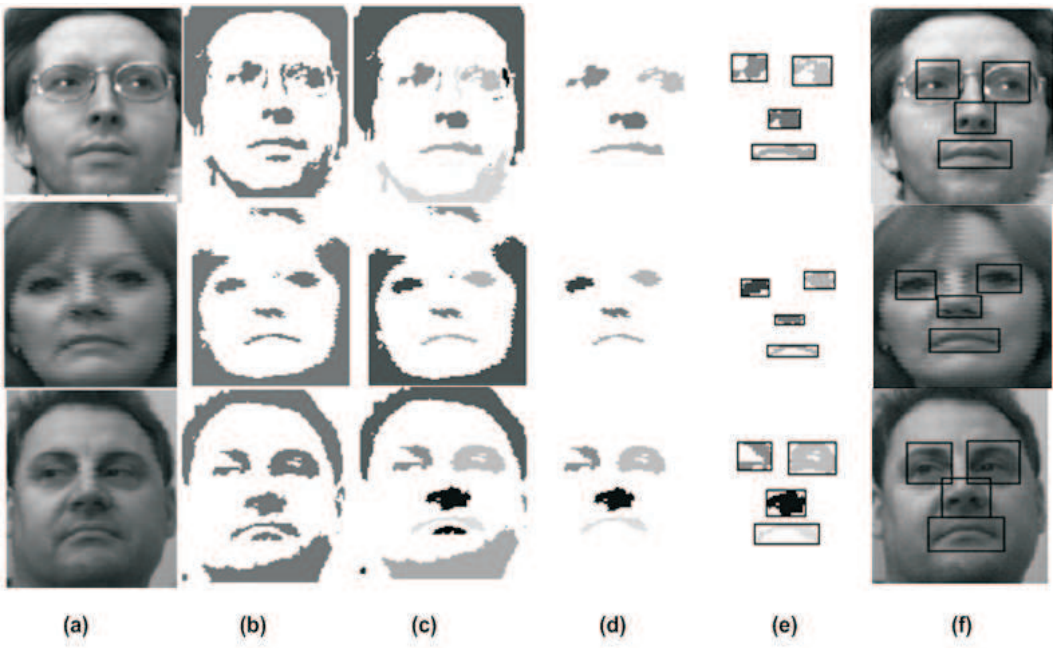


Figure 4.3: Step by Step Results of the Approach. a) Original Face Images, b) Binary Images, c) Connected Components of Binary Images, d) Detected Region of Interest, e) Detected Region of Interest with Rectangular Boundaries, f) Original Images with the Detected Regions of Interest.

$$B_{ij} = \begin{cases} 255 & \text{if } x_{ij} > \tau \\ 0 & \text{otherwise .} \end{cases} \tag{4.6}$$

The face components then appear on the foreground of the binary image as shown in Fig. 4. 3. b). Thereafter, the algorithm searches, detects and labels the connected components [35],[62] of the binary image.

Connected Components Finding

Once an image has been binarized, both the components on the foreground and the background could be viewed as a set of classes of connected and similar pixels. Using this idea, we applied the 4-connectivity to find the connected components of the binary image B_{mn} into classes of related pixels. The pixels of each class are assigned a unique label to differentiate classes. It entails grouping the binary image B_{mn} into classes $c_0, c_1, c_2, ..., c_{k-1}$, where k is the number of connected components. Let's call

the resulting image C_{mn}. C_{mn} is defined as :

$$C_{mn} = \bigcup_{i=0}^{k-1} c_i \ , \tag{4.7}$$

where

$$\begin{aligned} c_i &= \{(l,k,i), \text{ for} \\ &\quad 0 \leq l \leq m-1, 0 \leq k \leq n-1\} \ . \end{aligned} \tag{4.8}$$

Fig. 4.3. c) shows some examples of connected components of binary images. An iterative strategy is used to remove the irrelevant components. First, our algorithm searches all the components with pixels touching the outer border of the image. The pixels corresponding to the labels of these components are set to the background. Then, the remaining components situated in the inner face space are the most probable face components. Thereafter, the size of each remaining component is computed as its total number of pixels. Based on the fact that, in the interior face space, the biggest components are most probably the eyes, the nose and the mouth, the number of remaining components is tested. If this number is greater than the number of targeted components (which is four in our work), we successively select the smallest components and set the pixels corresponding to their labels to the background until the threshold of four remaining components is reached. In the successful case, the iteration stops when these remaining components are the two eyes, the nose, and the mouth.

Representation of the Regions of Interest

Instead of using a geometrical estimation or an assumption about the location of face components as done in previous works [12, 91, 102, 97, 84], our approach exploits the pixel coordinates of each detected component to determine its location in the face. In other words, our algorithm searches for the maximum x and y coordinates and the minimum x and y coordinates belonging to the component, in the two dimensional

space spanned by the image. Fig. 4.4. shows a geometrical representation of the targeted coordinates around a detected component.

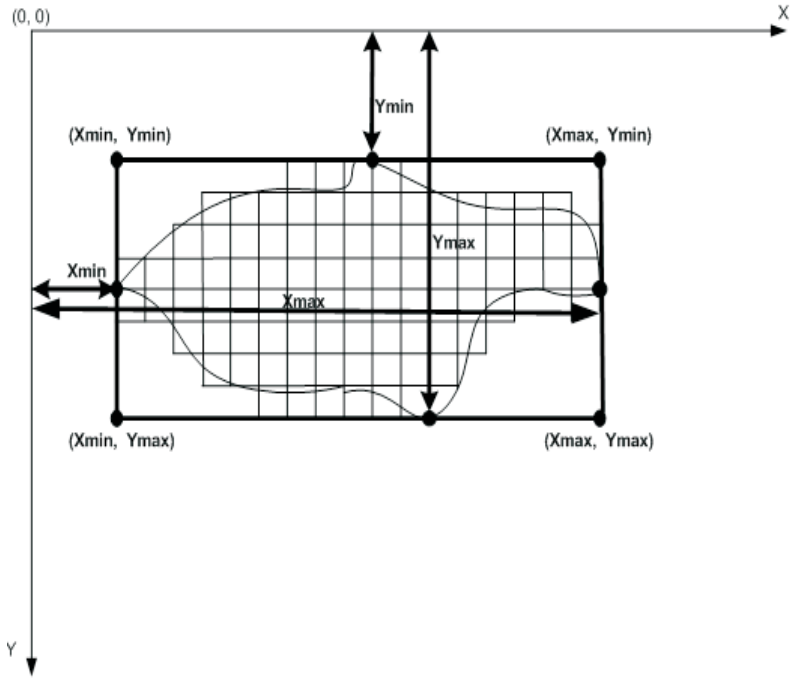


Figure 4.4: Coordinates of a Detected Region of Interest in the Face Image.

In Fig. 4.4, the coordinates X_{min}, X_{max}, Y_{min}, and Y_{max} are found and serve to deduce the coordinates of the four points at the corners of the rectangular boundary containing the component. Afterwards, these coordinates are used to identify the rectangular boundary of the region containing the detected component as shown in Fig. 4.3. e). The coordinates are later shifted to accurately represent the detected regions of interest in the original image for the purpose of feature extraction. By exploiting the coordinates of pixels in the detected components to determine their location in the face, we made no assumption or geometrical estimation of the location of components. Then, as shown in Fig. 4.5., our approach is able to detect face components in different orientations perfectly.

Figure 4.5: Facial Components Detected in Different Orientations.

Facial Component Validation Technique

The inputs to our validation model are the centroids of the detected facial components. Consider the finite set of points, S_n.

$$S_n = \{P_i, i = 1, .., n\} \ , \tag{4.9}$$

where n is the number of detected components, and P_i the centroid of the i^{th} detected component. The convex hull of S_n is defined as the smallest 2D convex polygon Ω that contains S_n [69].

After testing our algorithm on the training set, we found that in the case where components are successfully detected, their convex hull is a triangle with the vertices

placed at the centroids of the two eyes and the mouth. It is also observed that the magnitude of angles at the two eyes are most likely between 50^0 and 95^0 (See Fig.4.7). Once the convex hull has been computed, we determine the triplet formed by the angles on the two first centroids with the lowest y-coordinates and the absolute value of their difference (Fig. 4.7). In successful cases, the two centroids with the lowest y-coordinates are the two eyes. The set of triplets is further classified using a two-class SVM, one class representing the class of the set of successful detections, and the other the set of incorrect detections. For a component-based face recognition system, this classification step is important as the earlier detection of incorrectly detected facial components would increase the efficiency.

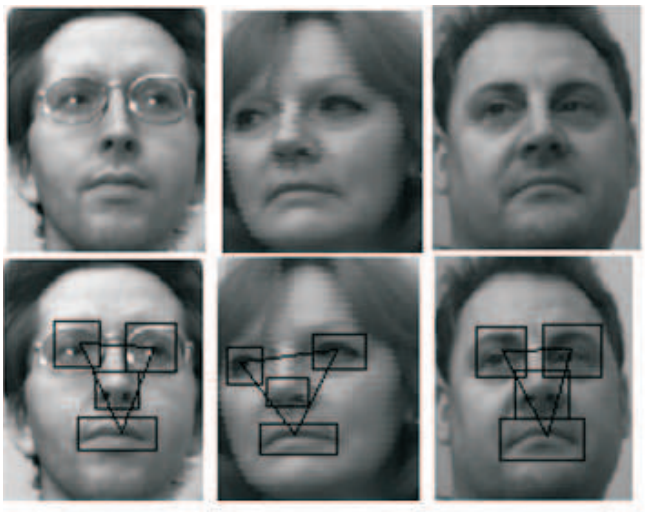


Figure 4.6: Examples of Face Images with Convex Hull of Detected Facial Components.

Components Detection and Validation Algorithm

The algorithm could be briefly described as follows :
Given the connected components of the binarized image of the face.

1) Search through the image, detect each connected component with pixels region touching the outer border of the image. Then, set the pixels corresponding to the label of such components to the background.

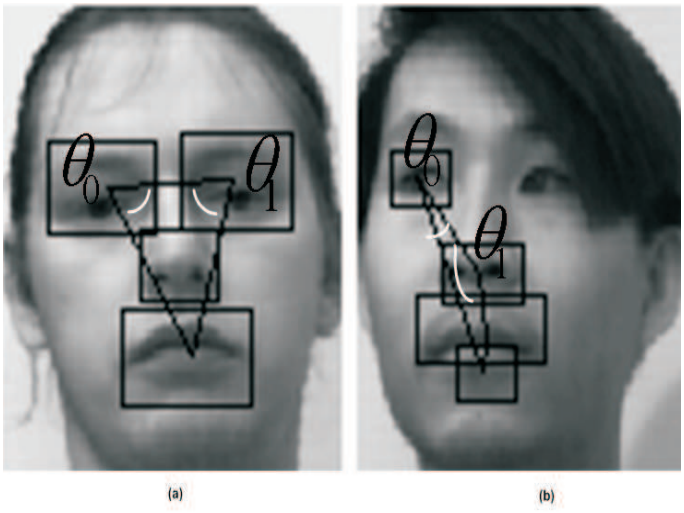


Figure 4.7: Examples of Face Images with the Angles (θ_0, θ_1) at the Side of the two Centroids with Lowest y-coordinates (as describe in Fig 3). The Absolute Value of the Difference of these Angles is $\Delta\theta = \mid \theta_0 - \theta_1 \mid$ and the Triplet is defined as: $(\theta_0, \theta_1, \Delta\theta)$; (a) Face with successful detected Components; (b) Face with an undetected Component.

2) Localize, compute and save the size of each remaining components along with its label.

3) If the number of remaining components is greater than four (the number of targeted components), successively choose the smallest size and set the pixels of their corresponding components through the image to the background until the four remaining components are reached. After this step, the remaining components include the targeted face components.

   In successful cases, the four remaining components are the two eyes, the nose and the mouth.
   Fig. 4.3. d) are examples of images with only the detected face components.

4) Compute the center of gravity of each detected components.

5) Apply the convex hull algorithm to the set of center points to extract the convex hull of the centroids of the detected facial components.

6) Determine the triplet constituted by the angles at the site of the two first points

with the minimum y-coordinates and the difference of these angles to obtain the vector $(\theta_0, \theta_1, \Delta\theta)$, where θ_0, θ_1, and $\Delta\theta$ are defined as in Fig. 4.7.

7) Process the triplet for components validation.

4.4 Computation and Representation of Features

The flowchart of the computation and representation of features is given in Fig. 4.8. The input to the module is an image with the detected and validated facial components: the two eyes, the nose and the mouth. The two eyes are extracted and fed into the Gabor feature extraction module that computes and outputs a partial features vector that is the concatenation of the features from the two eyes. Similarly, the nose and the mouth are fed into the Zernike Moments feature extraction module and the corresponding partial features vector is obtained. Both partial features vectors are further concatenated and normalized by the mean and the standard deviation to produce the final feature vector of the input image.

4.4.1 Gabor Feature Extraction

Background and Related Works in Face Recognition

Over the recent years, Gabor Filters have been of interest to a growing number of researches, for feature extraction in face recognition. In [89], Chen et al. proposed a face recognition system where the feature extraction is done by combining Gabor Filters with Local Binary Patterns (LBP). In their approach, each input face image is convoluted with 40 Gabor filters (5 scales/frequencies and 8 orientations). Then, 40 Gabor magnitude maps and 40 Gabor phase maps, both of the same size as the original face image, are obtained. Thereafter, the maps are processed by LBP respectively to model the holistic distribution of local regions in the images, which results in 80 LBP images of all the magnitude and phase maps. Spatial histograms are further estimated from the LBP to form the final representation of the input image. Gabor Filters of 6 scales and 8 orientations are applied in [23] on 20 detected facial points of size 13x13 each for feature extraction. The feature vector of each point is represented by the magnitude of the Gabor coefficients which leads to feature of 13x13x48=8112 dimensions. Consequently, an input face image is represented by a

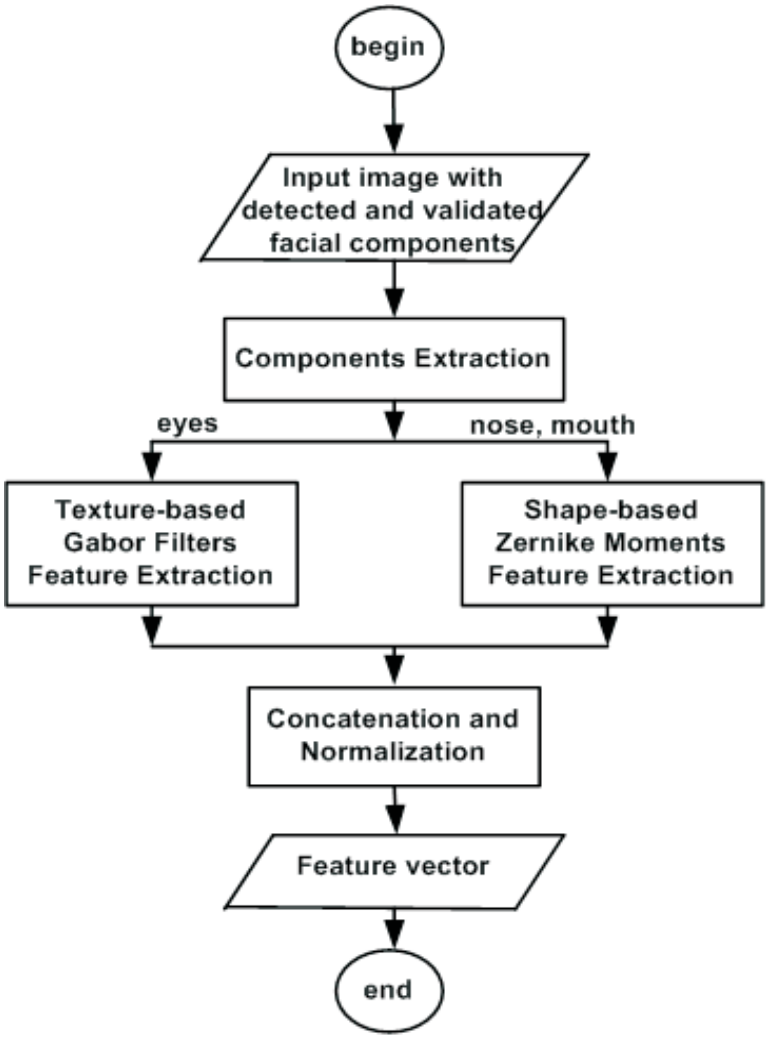


Figure 4.8: Flowchart of the Computation and Representation of Feature Module.

feature vector of 20x8112 dimensions. The dimensionality reduction is further done by means of GentleBoost technique for classification. In [71], Lim et al. manually select four landmark points on a face image (Center of the eyes, nose and mouth). Gabor Filters of 4 frequencies and 8 orientations are used to encode each manually selected landmark point(X,Y). Thus, 32 Gabor Filters responses are used to represent a landmark point, and 32x4=128 Gabor Filters for an input face image. Thereafter, Genetic Algorithm (GA) is applied for a more optimal encoding of the face landmark point. Erik [27] presents a feature extraction based on the combination of Gabor Filters and Gaussian weighting technique. Each input face image is first filtered with

Gabor Filters of 3 scales and 8 orientations. Thereafter, the filtered image is convoluted with 2-D Gaussian to focus on the center of the face. To reduce the dimension of the feature space, the Gabor Filtered and Gaussian weighted image is searched for peaks, which are considered as the interesting feature points for face recognition. As a result of the search, the four peaks found are the center of the eyes, the nose and the chin. In [100], Gabor Filters of 3 frequencies and 8 orientations are applied on frontal face images of size 64x64 each, which leads to feature vectors of 64x64x8x3=98304 dimensions. A sub-sampling strategy based on the Principal Component Analysis (PCA) and the Direct fractional-Step Linear Discriminant Analysis (DF-LDA) is further used for dimensionality reduction and recognition. In [101], Zhang represents fiducial points of a face with Gabor coefficients. First, his approach manually selects 34 fiducial points on the input face image. Gabor Filters of 3 scales and 6 orientations are applied at each manually preselected points to compute 18 Gabor coefficients. An input face image is then encoded with a feature vector of 18x34=612 dimensions. In [83], input face images of size 64x64 each, are convoluted with Gabor Filters of 5 frequencies and 8 orientations, for feature extraction. As the dimension of the resulting feature vectors is very high (64x64x8x5=163840), a dimensionality reduction strategy based on a conditional mutual information theory is used to select a set of informative and non redundant Gabor features for face recognition. The set of selected Gabor features is further subjected to generalized discriminant analysis (GDA) for class enhancement. As a result, significant computation and memory efficiency have been achieved since the dimension of features has been reduced from 163840 to 200 for 64x64 images. Den et al.[37] manually detect facial features points including eyes, nose, and mouth, to normalize and fix the size of an input image to 128x96 pixels, for feature extraction. Afterward, Gagor Filters of 5 scales and 8 orientations are applied on the normalized image and a high dimensional feature vector of 128x96x40 = 491520 dimensions is obtained. To cope with the problem of dimension of the feature vector, a dimensionality reduction strategy, based on the combination of PCA and LDA is applied to select and compress the Gabor features for classification or recognition. In [63], Zhou et al. present a feature extraction approach using Gabor wavelets and AdaBoost. First, they manually measure the center of the two eyes for each face image. The result of the measurement is used to normalize and set the size of the image to

25x24 pixels. Thereafter, features of the normalized face image are extracted with Gabor wavelets of 5 scales and 8 orientations. The dimensionality reduction is further carried out by selecting a small set of significant feature with the Adaboost algorithm.

Most of the works presented above are based on Gabor Filters and the search for features on or around the key facial components that are: the two eyes, the nose and the mouth [23, 71, 63, 37]. However, none of these studies has attempted to detect accurately and automatically these salient regions of the face as it is done in [24], for feature extraction. Moreover, a lot of time is wasted in these previous works for dimensionality reduction on the Gabor features vectors obtained, which could impact negatively on the speed of the feature extraction. Furthermore, there is no guarantee that the techniques used for dimensionality reduction are numerically stable to avoid drastic changes of feature data.
In the next two sections, we present and apply Gabor Filters to extract the texture of the detected eyes components. The dimensionality reduction on the Gabor features is avoided by considering the mean and the standard deviation of the magnitude of Gabor coefficients as the texture features of the eyes, instead of the high dimensional feature of magnitude itself; which speeds-up the feature extraction, while preserving the accuracy of the initial feature data.

Gabor Feature Represenation

Our Gabor feature extraction structure is explained in Fig. 4.9. First, the detected and validated facial components are extracted from the input image. Afterward, Gabor Filters are applied successively on the two eyes and their texture features are extracted as vectors of 60 dimensions each. Both texture features are fused or concatenated to form the sub-feature vector of the face based on the eyes only.

After the design of Gabor Filters, image features at different locations, frequencies, and orientations can be extracted by convolving the image $I(x, y)$ of size $M \times N$ with the filters. In this work, the image $I(x, y)$ is an extracted eye as shown in Fig. 4.9. Table 4.1 presents the dimensions of different eyes components detected from the bank of images in Fig. 4.10, numbered from top to bottom. At each point (x, y) of

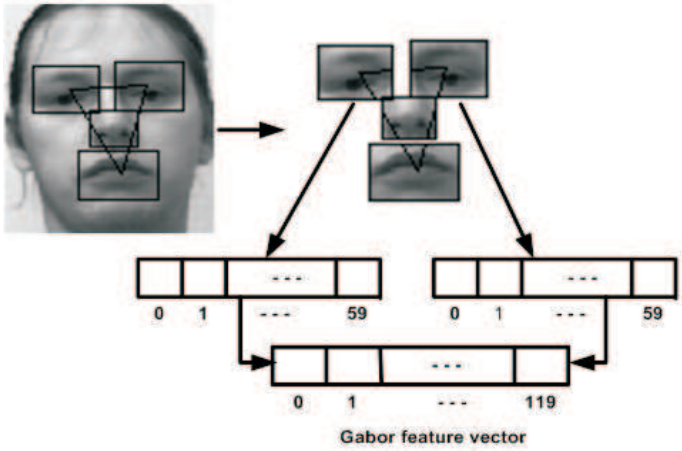


Figure 4.9: The Framework of the Gabor Feature Extraction.

Table 4.1: Dimensions of Eyes Components detected in the Bank of Images in Fig. 4.10, numbered from Top to Bottom.

Number	*Detected Eyes Components Dimensions*	
	left	*right*
1	13x17	15x17
2	29x27	27x26
3	23x19	16x20

the image, the convolution is defined as follows [35]

$$G_{mn}(x,y) = \frac{1}{M \times N} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} I(x-m, y-n)\psi(m,n) \ , \tag{4.10}$$

where ψ is the filter mask of size $m \times n$ and G_{mn} the matrix of Gabor coefficients of the same size as the image $I(x,y)$. In practice, the mask is derived from equation (3.13). A more compact representation of G_{mn} is :

$$G_{mn}(x,y) = I(x,y) * \psi(m,n) \ , \tag{4.11}$$

where * denotes the convolution operator. The result of the convolution should be a complex representation of Gabor features. And the magnitude of those complex vectors would be the basic feature for texture segmentation and recognition [100]. The complex representation of G_{mn} is given by the formula:

$$G_{mn}(x,y) = R_{mn}(x,y) + iI_{mn}(x,y) \ , \tag{4.12}$$

where R_{mn} and I_{mn} are the real and imaginary parts of Gabor coefficients, respectively. Fig. 4.10 shows some examples of imaginary parts of Gabor Filters for the two eyes, at scales $\pi/4$ and $\pi/2\sqrt{2}$, and orientations $5\pi/8$ and $3\pi/8$. Then, the magnitude $|G_{mn}(x,y)|$ of G_{mn} is calculated as follows:

$$|G_{mn}(x,y)| = \sqrt{R_{mn}(x,y)^2 + I_{mn}(x,y)^2} \ . \tag{4.13}$$

For each couple of frequency and orientation, the application of Gabor Filters on an image produces a magnitude of Gabor coefficients as defined above, which is a matrix of the same size as the input image. The sum of components of this magnitude is called the energy of the filter [22] for the given scale and orientation. Let's call the energy of an image extracted with a mask of size $m \times n$, $E(m,n)$. Its value is computed using the following formula:

$$E(m,n) = \sum_x \sum_y |G_{mn}(x,y)| \ . \tag{4.14}$$

Once the Gabor Filters have been applied on an image with different frequencies and orientations, the result is an array of magnitudes. These magnitudes represent the energy of the image at different scales and orientations and could therefore be used to extract the texture properties of the image [22]. As the main purpose of texture-based retrieval is to find images or regions with similar texture, the mean μ_{mn} and standard deviation σ_{mn} of the magnitude of Gabor coefficients (See equations (4.15) and (4.16)) at each scale and orientation could be used to represent the homogenous texture feature of an image region [22].

$$\mu_{mn} = \frac{E(m,n)}{M \times N} \tag{4.15}$$

$$\sigma_{mn} = \frac{\sqrt{\sum_x \sum_y (|G_{mn}(x,y)| - \mu_{mn})^2}}{M \times N} \tag{4.16}$$

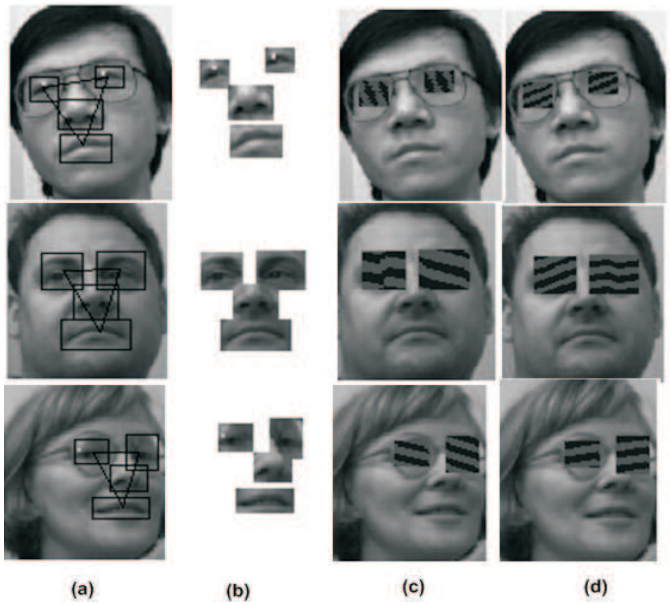


Figure 4.10: Examples of Imaginary Parts of Gabor Filters:(a):Input Images with detected and validated Facial Components;(b):Extracted Facial Components;(c,d):Original Image with Imaginary Parts of Gabor Filters at Scales $\pi/4$ and $\pi/2\sqrt{2}$, and Orientations $5\pi/8$ and $3\pi/8$, respectively.

In our implementation, Gabor Filters of 5 scales and 6 orientations lead to the feature vector f defined below for each detected eye region:

$$f = (\mu_{00}, \sigma_{00}, \mu_{01}, \sigma_{01}, ..., \mu_{45}, \sigma_{45}) \ . \tag{4.17}$$

f is a vector of 60 dimensions. Therefore, the two eyes are represented in our work with a vector of 120 dimensions obtained by concatenating the Gabor features from both eyes regions. An example of Gabor feature vector extracted from the left eye of the top image in Fig. 4.10 is given in Table 4.2.

4.4.2 Zernike Moments Feature Extraction

Zernike Moments have been applied in face recognition in the recent years. Saradha et al. [81], applied Zernike Moments at order 10 as feature extractor on input face images along with other moment descriptors such as: Fourier descriptors, Legendre Moments, and Hu Moments. Thereafter, a comparative study of these methods was

Table 4.2: Example of a Gabor Feature Vector for the Left Eye of the Top Image in Fig. 4.10.

Mean	*Standard Deviation*
$\mu_{00} = 0.08181044091054135$	$\sigma_{00} = 1.3559866487381657$
$\mu_{01} = 0.08337636259426681$	$\sigma_{01} = 0.6676369383831324$
$\mu_{02} = 0.08152348198809901$	$\sigma_{02} = 1.2317149332260285$
$\mu_{03} = 0.08491605886368899$	$\sigma_{03} = 0.23671802276324456$
$\mu_{04} = 0.09014125256830944$	$\sigma_{04} = 0.374231236577378$
$\mu_{05} = 0.08392206781246044$	$\sigma_{05} = 0.6838807554957311$
$\mu_{10} = 0.08341801913017339$	$\sigma_{10} = 0.8492383112337858$
$\mu_{11} = 0.0760509986845547$	$\sigma_{11} = 0.5809793367205929$
$\mu_{12} = 0.08546242648601242$	$\sigma_{12} = 0.4155015960262604$
$\mu_{13} = 0.06806158399022257$	$\sigma_{13} = 1.4646988953798364$
$\mu_{14} = 0.08372040917422031$	$\sigma_{14} = 1.4157061109129838$
$\mu_{15} = 0.08197309262939245$	$\sigma_{15} = 1.0989956761355177$
$\mu_{20} = 0.0769797574020915$	$\sigma_{20} = 0.8821930359446563$
$\mu_{21} = 0.07909270911226945$	$\sigma_{21} = 0.5553540028174903$
$\mu_{22} = 0.07317591440046756$	$\sigma_{22} = 1.3202163797908408$
$\mu_{23} = 0.070023340678377$	$\sigma_{23} = 0.883162900620407$
$\mu_{24} = 0.0802667702163421$	$\sigma_{24} = 0.7949532950158003$
$\mu_{25} = 0.0781800439204664$	$\sigma_{25} = 0.864711246941754$
$\mu_{30} = 0.07897105152519596$	$\sigma_{30} = 0.8294916792905965$
$\mu_{31} = 0.08055107952226649$	$\sigma_{31} = 0.2639497125572756$
$\mu_{32} = 0.08279032564842433$	$\sigma_{32} = 0.7971833467788671$
$\mu_{33} = 0.07928489826607043$	$\sigma_{33} = 0.8576380199533234$
$\mu_{34} = 0.08352822002041937$	$\sigma_{34} = 1.1148694948746472$
$\mu_{35} = 0.07798851717079486$	$\sigma_{35} = 0.7260414785864759$
$\mu_{40} = 0.07722373498036806$	$\sigma_{40} = 0.4986908602924516$
$\mu_{41} = 0.08137228696607556$	$\sigma_{41} = 0.9479364529554858$
$\mu_{42} = 0.0810443339118555$	$\sigma_{42} = 0.8056562413945153$
$\mu_{43} = 0.08701887868529849$	$\sigma_{43} = 0.7500911699666682$
$\mu_{44} = 0.07422434349268941$	$\sigma_{44} = 0.8608944468329163$
$\mu_{45} = 0.0800765682749293$	$\sigma_{45} = 0.9849069988001335$

carried out. In [30], Pseudo Zernike Moments at order 10, and PCA are applied successively to compute features of extracted elliptical shape face images. The resulting feature vectors are further compared to show the classification performance. A similar work was done in [45] based on Pseudo Zernike Moments, and Lengendre Moments. Mohammed et al. [79], combines Zernike Moments and neural network to localize a face in an image. Zernike Moments at order 10 with 5 repetitions are applied to extract features of input face images. This resulting feature vector is further fed into a neural network structure to learn the pixels at the contour of the face region.
The framework of our Zernike Moments feature extraction is given in Fig.4.11.

We defined Zernike moments in Chapter 3. For Zernike Moments of order n with l

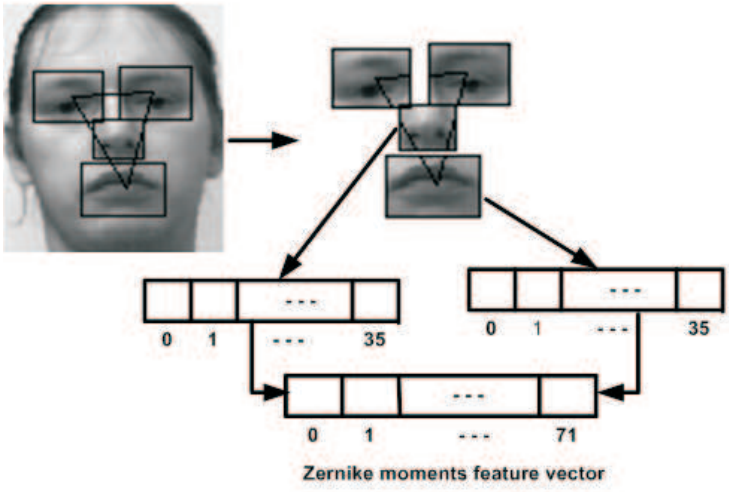


Figure 4.11: The Framework of the Zernike Moments Feature Extraction.

repetitions, the number of moments features could be computed as follows:

$$N\_moments = \sum_{k=0}^{n} \left\{ \lfloor \frac{k}{2} \rfloor + 1 \right\}, \tag{4.18}$$

where $\lfloor u \rfloor$ is the integer part of u. For a given order n, $k = 0, ..., n$ and $l = 0, ..., \pm n$. Let's consider the case of $n = 4$. Table 4.3 shows the various repetitions associated.

Table 4.3: Repetitions of Zernike Moments of Order 4.

Order n	**Repetitions** l								
4	-4	-3	-2	-1	0	1	2	3	4

Once the order n and the l repetitions have been chosen, the Zernike Moments algorithm is executed and the moments features are selected based on the following basic conditions: $|l| \leq n$ and $n - |l|$ is always even. As $|l| = \pm l$, only the positive values of l could be considered as repetitions. Table 4.4 shows the selected moments feature for Zernike Moments at order 4. The number of moments features here is 9.

In practice, the Zernike Moments features A_{nl} are complex numbers. Thus, their magnitudes $|A_{nl}|$ should be taken as the final features values. Table 4.5 shows the values of Zernike Moments features for orders $0 \leq n \leq 10$.

The order 10 is commonly used in the literature [81, 44] to capture the details shape information in images. We have applied it in our case to extract the shape

Table 4.4: Zernike Moments Features at Order 4.

Order n	**Repetitions** l				
	0	1	2	3	4
0	A_{00}				
1		A_{11}			
2	A_{20}		A_{22}		
3		A_{31}		A_{33}	
4	A_{40}		A_{42}		A_{44}

Table 4.5: Number of Zernike Moments Features at Orders from 0 to 10.

Order n	**Numbers of Zernike Moments Features**
0	1
1	2
2	4
3	6
4	9
5	12
6	16
7	20
8	25
9	30
10	36

of the detected nose and mouth facial components, which leads to a feature vector of 36 dimensions for each component and a Zernike Moments feature vector of 72 dimensions for each face image, characterizing the shape of the nose and the mouth only. The Zernike Moments features of the nose component of the top image in Fig. 4.10 is shown in Table 4.6.

4.5 Features Fusion and Normalization

We combine the texture of the eyes and the shapes of the nose and the mouth to characterize a face. The high discriminative power of Gabor Filters in texture representation and Zernike Moments in characterizing shapes, motivated their choice. Our fusion framework is shown in Fig. 4.12.

The partial feature vector of the eyes texture of 120 dimensions, computed by 30 Gabor Filters, is fused with the partial feature vector of 72 dimensions, representing the shapes of the nose and the mouth obtained by means of Zernike Moments at

Table 4.6: Example of Nose Zernike Moment Features.

Number	*Features*
0	9737.770515291975
1	12771.537273400585
2	29213.31154587586
3	443.8493306736061
4	25543.07454680117
5	14426.63377008863
6	48688.852576459794
7	739.7488844560106
8	13760.327525867187
9	38314.611820201724
10	21639.95065513294
11	4736.415372453778
12	68164.39360704363
13	1035.6484382384192
14	19264.458536214082
15	2554.322836635707
16	51086.14909360234
17	28853.26754017726
18	6315.22049660503
19	5233.515152875227
20	87639.93463762765
21	1331.5479920208165
22	24768.589546560976
23	3284.1293613887624
24	11040.633248353282
25	63857.686367002905
26	36066.58442522156
27	7894.0256207562925
28	6541.893941094035
29	11523.441051820959
30	107115.47566821153
31	1627.447545803227
32	30272.720556907807
33	4013.935886141819
34	13494.107303542896
35	799.6831617363044

order 10. The fusion consists of concatenating the two partial feature vectors into a single vector of 120+72=192 dimensions, which is the final signature of the input face image.

The scale of individual feature extracted with Gabor Filters and Zernike Moments differ drastically. To address this problem, given a final feature vector f of p components, where f is defined as:

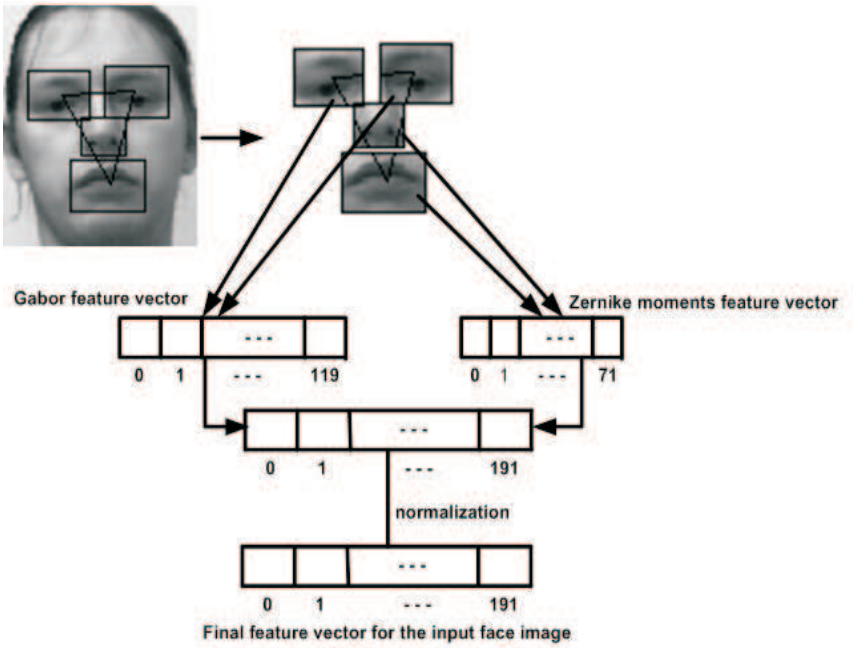


Figure 4.12: The Framework of the Feature Fusion Strategy.

$$f = (x_1, x_2, ..., x_p) , \tag{4.19}$$

we used a statistical method to normalize the vector. Each component x_i is independently transformed into its normalized value $\tilde{x}_i$ as follows:

$$\tilde{x}_i = \frac{x_i - \mu}{\sigma} , \tag{4.20}$$

where μ and σ are the mean and the standard deviation of f coordinates, respectively. The normalization is applied both to the training and the test sets.

4.6 Classification/Recogntion

This module performs essentially two operations: storage of trained feature vectors in the database and identification of a sample face images. Face recognition is fundamentally a pattern recognition problem. Solving such a problem generally involves two phases: the training and the classification/recognition. In the training phase, features of a subset of the population to be recognized are computed and stored in a

storage area called template database (See Fig. 4.13). The classification/recognition phase extracts the feature of a new subject and matches it with those of known subjects in the database. The correct match of the new feature yields the acceptance of the claimed identity, whereas the incorrect match yields its rejection.

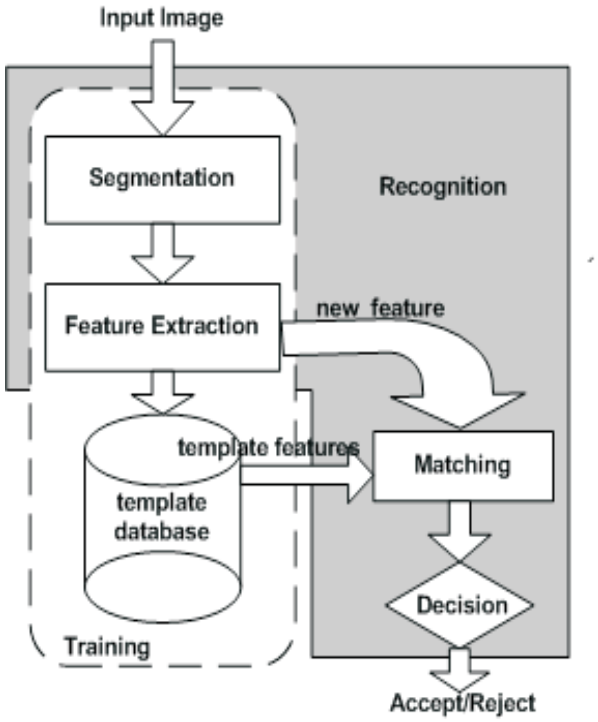


Figure 4.13: Framework of a Typical Face Recognition System.

4.6.1 Storage of Known Persons in the Database

First, we compute and store in a template database, feature vectors of the training set. Each database record represents the centroid of a class of face images of the same person.

4.6.2 Identification of a Candidate Face

The nearest neighbor rule is used to classify new face images. The identification is done by computing the distance between the feature vector of the candidate face and the templates from the database. The unknown face feature vector is extracted and

compared with the feature vectors stored in the template database of known persons using the formula given in (4.21).

$$D(k) \quad = \quad \sqrt{\sum_{j=1}^{n}(f_j(u) - f_j(k))^2}\ , \tag{4.21}$$

where n and $f_j(u)$ are the number of features and the features of an unknown person respectively, while $f_j(k)$ represents the features of the k^{th} known person in the database. Then the unknown person is identified as the k^{th} person in the database if the distance $D(k)$ is the minimum amongst all the persons available in the database and less than a conveniently chosen threshold.

4.7 Conclusion

We have presented the details of the implementation of the system. A literature review has been carried out where appropriate, to support the aims of the implementation. Diagrams have been provided to make the structure of the system more understandable. Resulting images and data tables illustrate the outputs of the implementation. In the next chapter, the programming environment is described, the results are further presented and discussed, and comparison with related works is performed.

Chapter 5

Experimental Results and Discussion

5.1 Introduction

In this chapter, we present experiments and a comparison of our results with related works. The programming environment and the data set are first described and the experimental results are presented and discussed.

5.2 Programming Environment

The system is implemented in the form of plugins in ImageJ [75], a public domain Java image processing program. Separate plugins have been developed for each image processing operation (binarization, connected component, region of interest detection, bounding box drawing, convex hull, Gabor Filters, Zernike Moments). These plugins are sequenced such that the output image of one serves as input to another. A statistical package R is used for SVM classification. The template database is implemented in XML with the Java Document Object Model (DOM) package. The UML class diagram, the description of the various plugins of the system, and some samples outputs results are given in the Appendix.

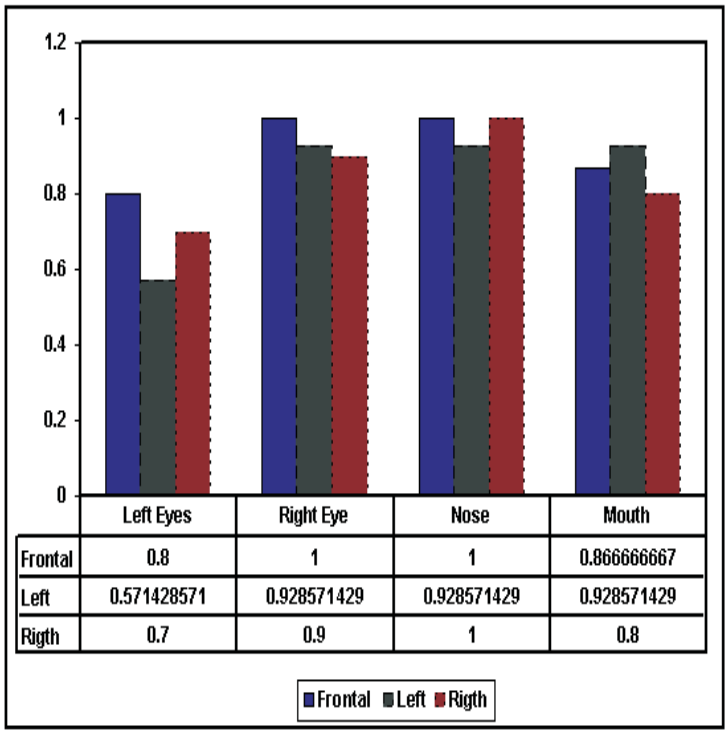
5.3 Data Set

In our study, we assume that the image quality and the resolution is sufficient enough, the illumination is uniform and the input images are gray scale images. However, we made no restrictions on wearing glasses, make-up, hairstyle, beard, and the like.
The images we used to test our system are taken from the Yale database. This database contains 575 face images of 20 persons with 92x112 pixels in size, taken in different orientations.

5.4 Experiment1: Components Extraction and Validation

We carried out our test with 46 images. Fig. 5.1 shows the chart of the detection rate of face components in frontal, left, and right view.

In the frontal view, the detection rate of the nose and the right eye is nearly 100%, the left eye 80%, and the mouth over 86%. In the left view, the detection rate of the nose, the mouth, and the right eye is over 92% and that of the left eye is over 57%. The right view presents a detection rate of 70% for the left eye, 90% for the right eye, 80% for the mouth, and nearly 100% for the nose.
The chart of detection rate in various orientations is shown in Fig. 5.2. It appears that the detection rate is nearly 92% in frontal view, 85% in right view and over 83% in left view.



	Left Eyes	Right Eye	Nose	Mouth
Frontal	0.8	1	1	0.866666667
Left	0.571428571	0.928571429	0.928571429	0.928571429
Rigth	0.7	0.9	1	0.8

Figure 5.1: Chart of Detection Rate of Face Components in Frontal, Left, and Right View.

Table 5.3 compares our approach, in terms of the successfully detected key facial components, against the top performing works in the field. The table shows that the two eyes, the nose and the mouth, that are successfully detected, are the commonly

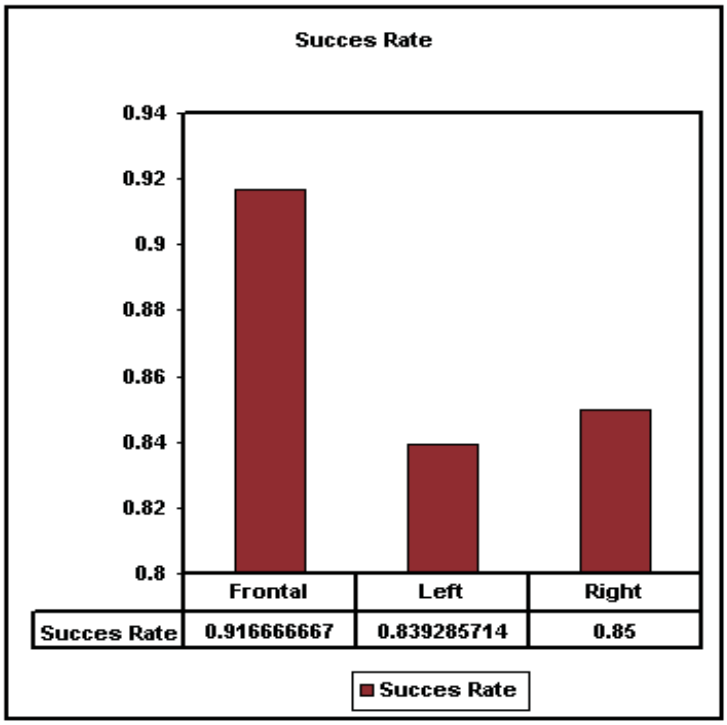


Figure 5.2: Curves of Detection Rate of the System in Different Orientations.

targeted components in most of the previous works. We compare the success rate of our approach at different orientations against related works in Table 5.4. Although our approach doesn't achieve the best result in frontal view, it detects more accurately facial components in different orientations compared to previous works.

For validation purposes, the triplets extracted from the images of the training set (See Fig. 4.7) were further classified. We have used two classifications techniques: K-means clustering and SVM. The triplets extracted from both training and sample images are given in Table 5.1.

We ran the K-means clustering with two classes; one class representing the class of properly detected facial components and the other, the class of bad detected one. A sample K-means classification of a new triplet is shown in Table 5.2. In this table, the triplets corresponding to the bad detected components are classified in the *Cluster* 0, whereas those of the correctly detected facial components are classified in the *Cluster* 1. The row corresponding to the classified triplet appears in bold face in the table.

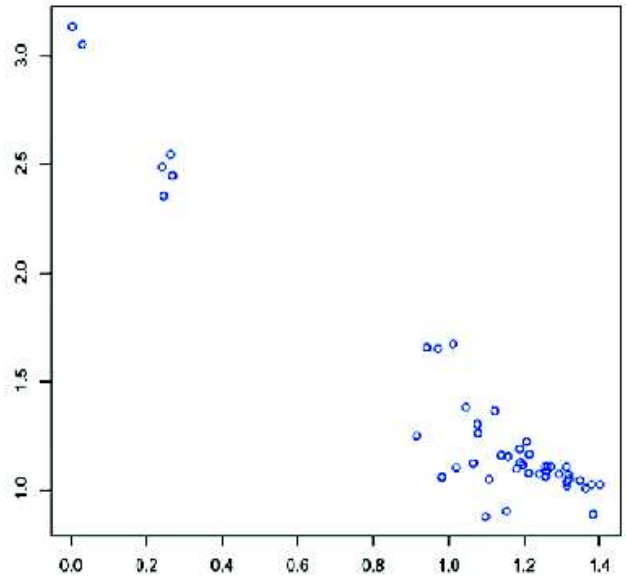


Figure 5.3: Result of SVM Classification of Triplets extracted from the Sets of detected Facial Components; the Top Left Feature Points form the Cluster of Triplets extracted from Facial Components incorrectly detected and the Bottom Right Feature Points the Cluster of correctly detected Facial Components, as shown in Table 5.2.

The result of SVM classification is depicted in Fig. 5.3, where the class of the facial components that are successfully detected and that of those which are wrongly detected are clearly linearly separated. The width of the separating hyperplane between the support vectors of both classes shows the power of the validation method.

Facial images with hair covering a portion of the eye, or with a beard are still slightly inaccurately detected by our system. The detection problems come from the fact that, these categories of face images merge more than one connected component by establishing a link between them as shown in Fig. 5.4.

5.5 Experiment2: Feature Extraction and Identification

Our feature extraction model is tested with 40 images belonging to 10 data classes. The 40 face images were taken in different orientations of the face. Fig. 5.5 shows the histograms of recognition rate in frontal, left, and right view, as well as the average recognition rate in different orientations of the face. Based on Fig. 5.5, our method

Table 5.1: Triplets extracted from both Training and Sample Face Images.

	θ_1	θ_2	$\Delta\theta = \lvert\theta_1 - \theta_2\rvert$
0	1.2066982233776784	1.2217952795067433	0.015097056129064867
1	1.2558891215135704	1.107717869696209	0.14817125181736146
2	1.2066982233776784	1.2217952795067433	0.015097056129064867
3	1.3191881245283303	1.0494351041851249	0.2697530203432055
4	1.378095568132511	1.023234090854899	0.3548614772776122
5	1.1902899496825319	1.1273479903751584	0.06294195930737345
6	1.3182420510168371	1.0691922726057759	0.24904977841106124
7	1.0780871346852774	1.2623783156924002	0.18429118100712283
8	1.3145941905370055	1.0182944968590164	0.29629969367798914
9	1.4016951007935197	1.0232340908548991	0.37846100993862053
10	1.347801808213736	1.0424907713701093	0.3053110368436267
11	1.1885137526161473	1.1878800101572655	6.337424588818141E-4
12	1.0460636784877178	1.3817287668823592	0.3356650883946415
13	1.3633001003596938	1.0074800653029286	0.3558200350567653
14	1.3137037188222356	1.0377738252217037	0.2759298936005319
15	1.1076744801010612	1.048542438968385	0.059132041132676116
16	1.012323802372425	1.6733967034009436	0.6610729010285186
17	0.9420000403794635	1.6571610432203796	0.7151610028409161
18	0.9716210038085896	1.6523225404044293	0.6807015365958397
19	1.0653862705510355	1.1228697600933195	0.057483489542283994
20	1.2627435457711202	1.1071487177940904	0.15559482797702984
21	1.2136794984046857	1.1640067094813542	0.04967278892333149
22	1.1226685359165174	1.3657963507913224	0.24312781487480506
23	1.3120254348311988	1.1071487177940904	0.2048767170371084
24	1.2924966677897851	1.0734536104480743	0.2190430573417108
25	1.2578950989145121	1.0612040619859706	0.1966910369285415
26	1.240498971965643	1.0734536104480743	0.16704536151756866
27	1.1807737365666067	1.0975438904299795	0.0832298461366272
28	1.383013136981797	0.8869074027985949	0.49610573418320214
29	0.9819299169548331	1.0579920516931367	0.07606213473830359
30	1.0985674039573494	0.8764321620659624	0.222135241891387
31	1.0206582486156377	1.1028691433501245	0.08221089473448684
32	1.2120256565243244	1.0776006698921967	0.13442498663212765
33	1.0759088843638223	1.3035372286389857	0.22762834427516343
34	1.2711902824587216	1.1080947913055836	0.16309549115313793
35	1.1583858851975095	1.1543577096033049	0.004028175594204653
36	1.139455683143694	1.1598306229354014	0.020374939791707458
37	1.1967457229728653	1.1157216859618786	0.0810240370109867
38	1.153281846253494	0.9015983053976788	0.2516835408558151
39	0.9154001117024694	1.2492522559798749	0.3338521442774055
40	0.24497866312686378	2.3561944901923453	2.1112158270654815
41	0.2687030246351712	2.4511461965351007	2.1824431718999295
42	0.029403288204001142	3.0566908601400713	3.0272875719360703
43	0.0024813844852987037	3.1381913061625824	3.1357099216772837
44	0.2635204939878902	2.5485353331933727	2.2850148392054823
45	0.24031226077257495	2.4901033341767036	2.2497910734041286

achieves 100% recognition rate in frontal and right view, and 85% recognition rate in

Table 5.2: K-means Classification of a New Triplet from a Sample Face Image.

θ_1	θ_2	$\Delta\theta = \lvert\theta_1 - \theta_2\rvert$
———————	———*Cluster 0*———	———————-
0.24497866312686378	2.3561944901923453	2.1112158270654815
0.2687030246351712	2.4511461965351007	2.1824431718999295
0.029403288204001142	3.0566908601400713	3.0272875719360703
0.0024813844852987037	3.1381913061625824	3.1357099216772837
0.2635204939878902	2.5485353331933727	2.2850148392054823
0.24031226077257495	2.4901033341767036	2.2497910734041286
———————	———*Cluster 1*———	———————-
1.3182420510168371	1.0691922726057759	0.24904977841106124
1.0780871346852774	1.2623783156924002	0.18429118100712283
1.3145941905370055	1.0182944968590164	0.29629969367798914
1.4016951007935197	1.0232340908548991	0.37846100993862053
1.347801808213736	1.0424907713701093	0.3053110368436267
1.1885137526161473	1.1878800101572655	6.337424588818141E-4
1.0460636784877178	1.3817287668823592	0.3356650883946415
1.3633001003596938	1.0074800653029286	0.3558200350567653
1.3137037188222356	1.0377738252217037	0.2759298936005319
1.1076744801010612	1.048542438968385	0.059132041132676116
1.012323802372425	1.6733967034009436	0.6610729010285186
0.9420000403794635	1.6571610432203796	0.7151610028409161
0.9716210038085896	1.6523225404044293	0.6807015365958397
1.0653862705510355	1.1228697600933195	0.057483489542283994
1.2627435457711202	1.1071487177940904	0.15559482797702984
new 1.2136794984046857	**1.1640067094813542**	**0.04967278892333149**
1.1226685359165174	1.3657963507913224	0.24312781487480506
1.3120254348311988	1.1071487177940904	0.2048767170371084
1.2924966677897851	1.0734536104480743	0.2190430573417108
1.2578950989145121	1.0612040619859706	0.1966910369285415
1.240498971965643	1.0734536104480743	0.16704536151756866
1.1807737365666067	1.0975438904299795	0.0832298461366272
1.383013136981797	0.8869074027985949	0.49610573418320214
0.9819299169548331	1.0579920516931367	0.07606213473830359
1.0985674039573494	0.8764321620659624	0.222135241891387
1.0206582486156377	1.1028691433501245	0.08221089473448684
1.2120256565243244	1.0776006698921967	0.13442498663212765
1.0759088843638223	1.3035372286389857	0.22762834427516343
1.2711902824587216	1.1080947913055836	0.16309549115313793
1.1583858851975095	1.1543577096033049	0.004028175594204653
1.139455683143694	1.1598306229354014	0.020374939791707458
1.1967457229728653	1.1157216859618786	0.0810240370109867
1.153281846253494	0.9015983053976788	0.2516835408558151
0.9154001117024694	1.2492522559798749	0.3338521442774055

left view. The average recognition rate is 95%.

The low recognition rate in the left view is due to the fact that, face images that were taken with angles greater than 45^o were misclassified by our system. Table 5.5. compares our feature extraction approach in terms of recognition rate against the top

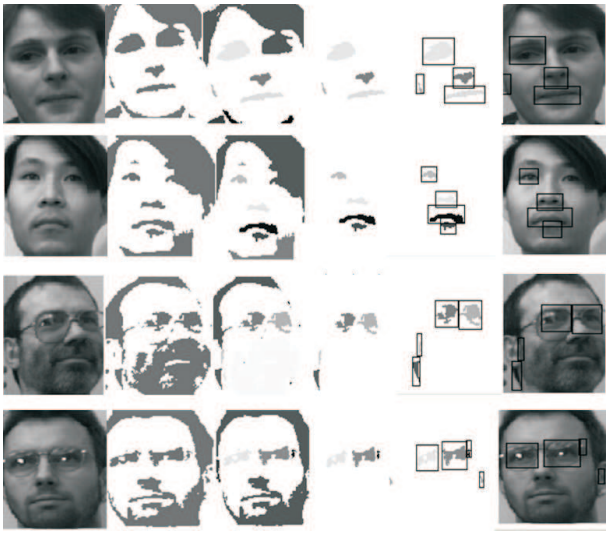


Figure 5.4: Examples of Challenging Face Images with Hair Covering a Portion of the Eye and with Beard.

Table 5.3: Comparative Table of Facial Components Detected.

Methods	*Facial Components Detected*			*Number*
	eyes	*nose*	*mouth*	
(Loulia et al., 2005)	x	x	x	3
(Selin et al., 2002)	x	x	x	3
(Heisele et al., 2006)	x		x	2
Our approach	**x**	**x**	**x**	**3**
(Bao et al., 2006)	x		x	2
(Wang et al., 1999)	x	x	x	3

Table 5.4: Comparative Table of Components Detection Success Rate in Different Orientation of Face.

Methods	*Face Orientation*				*Success Rate*
	Frontal	*Right*	*Left*	*All*	
(Loulia et al., 2005)	x				95%
(Selin et al., 2002)	x				93.76%
(Heisele et al., 2006)				x	89.25%
Our approach	**x**				**91.66%**
		x			**85%**
			x		**83.92%**
				x	**86.82%**
(Bao et al., 2006)	x				90%
(Wang et al., 1999)	x				90%

performing works in the field. From the table, our method performs better than any studied previous work, in different orientations of face.

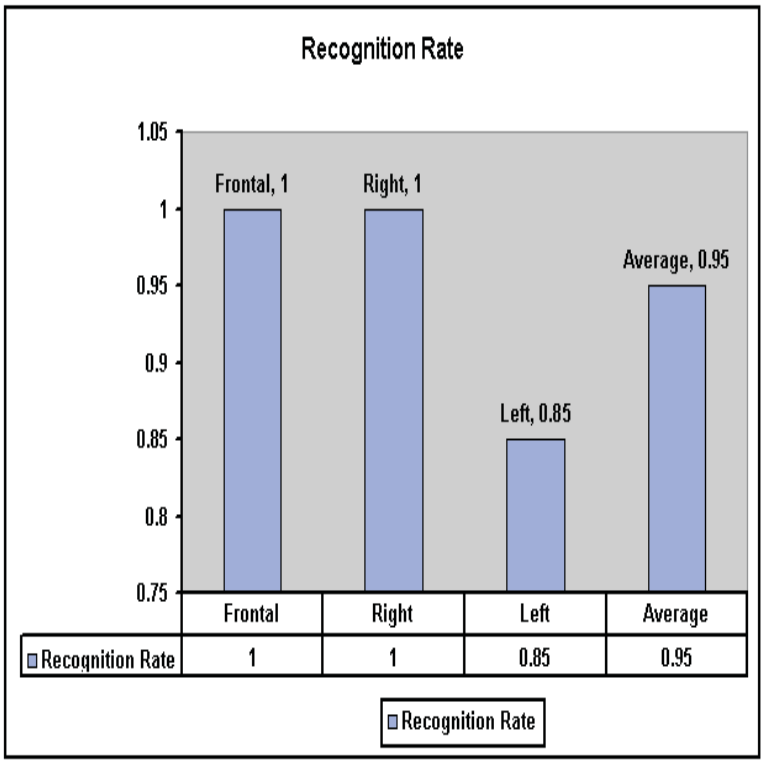


Figure 5.5: Histograms of Recognition Rates of in Frontal, Left, and Right View, and the Average Recognition Rate in different Orientations of Face.

Table 5.5: Comparative Table of Recognition Rate.

Methods	*Face Orientation*				*Recognition Rate*
	Frontal	*Left*	*Right*	*All*	
(Deng et al., 2005)	x				97.33%
(Mei et al., 2005)	x				100%
Our approach	**x**				**100%**
		x			**100%**
			x		**85%**
				x	**95%**
(Shen et al., 2006)	x				96.9%

5.6 Conclusion

This chapter presented and discussed experimental results of the system. Statistics have been drawn to compare the system with related works. In the next chapter, the summary of the work is done. The shortcomings of the system are reported and some pathways for future works are specified. Finally, a conclusion ends the dissertation.

Chapter 6

Conclusion and Future Works

6.1 Summary of Work

In this work, a component-based face recognition system is implemented. We first provided a comprehensive detailed review of the most popular used human biometrics traits. Thereafter, the investigation on the state of the art in component-based face recognition revealed some shortcomings of the previous works. For instance, we identified that facial components were mainly chosen manually, many assumptions were employed to locate facial components and components were detected using geometrical calculations on the face space.

Consequently, facial components were detected inaccurately. Some important components were missing in certain cases, and most of the techniques used were computationally expensive. Our first endeavor has been to find the solutions to the above problems. First of all, we applied simple image processing operations on facial images and attempted to detect automatically and more accurately facial components without any geometrical consideration. We found that, connected components of binary face images could provide clues about how to detect automatically the key facial components: the eyes, the nose and the mouth. Then, we adopted an iterative strategy to remove the irrelevant components from the connected components of an input face image. As a result, our approach detects automatically and with a high accuracy facial components in different orientation of the face.

The detected facial components were further validated by means of two-class SVM. Thereafter, the correctly detected components were extracted and used for feature extraction. The high discriminative power of Gabor Filters in texture representation and Zernike Moments in characterizing the shape motivated their use as feature extraction methods. Gabor Filters are applied to extract the texture features of the two eyes, whereas Zernike Moments are used to compute the shape features of the nose and the mouth. The texture and the shape features were further concatenated

and normalized to build the final feature vector of the input face image. Experiments were carried out on both components detection and feature extraction. Compared to previous studies, they showed that components are more accurately detected in different face orientations and the feature extraction achieves an average recognition rate of 95%.
Although our research endeavors provided solutions to some of the current problems in the component-based face recognition research field, there is still some room for improvement. This aspect is discussed in the next few lines.

6.2 Limitations of the System, Recommendations and Future Work

One of the critical steps of a system development is the testing phase, during which its shortcomings should be reported and eventually fixed. After testing our system, we found some limitations that are described below.

- **Sample Images:** In this work, we use pixel gray levels of face images to detect and extract facial components for recognition; which meant that, a bad illumination condition during the capture of sample face images could affect drastically the performance of the system. As a solution to this problem, our approach should not be used for passive surveillance. Instead it is suitable for biometric identification where the capture time, the agreement of the participants, and the parameters of the camera could be controlled efficiently.

- **Component Detection:** Some face images are still challenging to our system. For instance, facial images partly covered by the hair and some bearded faces are candidates in that category. The problem with such categories of face images is that, more than one connected components are merged into one component, which yield an automatic removal of one or more targeted facial components. These particular cases could be solved by using geometric approaches.

- **Classification/Recognition:** We use one nearest neighbor classifier with simple distance measure to classify candidate face images. Other techniques as the Radial Basic Function Neural network (RBF) and Support Vector Machines (SVM) could be investigated.

6.3 Conclusion

We have presented a component-based face recognition system. Facial components are firstly detected and validated. The feature extraction is carried out by combining the feature data computed by Gabor Filters and Zernike Moments. The identification is done by matching the feature vector of a candidate face to the template features of known persons in the database. A candidate face is accepted, if its minimum distance from the template features is below a conveniently chosen threshold. In terms of application, our system could be a solution to the new emergent biometric recognition in passports, driving licenses, and identity books.

Bibliography

[1] A. K. Arun, A. Ross and S.Prabhakar. An introduction to biometric recognition. *IEEE Transactions on Circuits and Systems for Video Technology, Special Issue on Image and Video-Based Biometrics*, 14(1):1–29, 2004.

[2] A. K. Bachoo. Comparison of segmentation methods for and accurate iris extraction. *Master Thesis, School of Computer Science, University of KwaZulu-Natal, Durban, South Africa*, page 95, 2006.

[3] A. K Jain, A. Ross, and S. Pankanti. A prototype hand geometry-based verification system. In *Appeared in proc. of 2nd International Conference on Audio-and-Video-based Biometric Person Authentication (AVBPA), Washington D. C.*, pages 166–171, March 22-24 1999.

[4] A. Kale, A. K. R. Chowdhury, and C. Rama. Towards a view invariant gait recognition algorithm. *Center for Information Research, University of Maryland, USA*, 2002.

[5] A. Kumar, D. C. Wong, H. C. Shen, and A. K. Jain. Personal verification using palmprint and hand geometry biometric. In *proceeding of* 4^{th} *International Conference on Audio and Video-based Biometric Person Authentication (AVBPA), j. kittler and M. Nixon, Eds.*, volume LNCS 2688, pages 668–678, 2003.

[6] A. Padilla-Vivanco, A. Martinez-Ramirez, and F. Granados-Augustin. Digital image reconstruction by using zernike moments. *Universidad de Puebla, Puebla, Mexico*, 2004.

[7] A. Paolo, M. Baltatu, and D. Rosalia. User authentication based on face recognition with support vector machine. In *Proceedings of the* 3^{rd} *Canadian Conference on Computer and Robot Vision (CRV'06), IEEE Computer Society, USA*, 2006.

[8] G. Alexander and V. Graevenitz. About speaker recognition technology. *Bergdata Biometrics GmbH, Bonn, Germany*, 2000.

[9] B. Chiraz, G. C. Ross, and S. D. Larry. Gait recognition using image self-similarity. In *EURASIP Journal on Applied Signal Processing, Hindawi Publishing Corporation*, number 4, pages 1–14, 2004.

[10] B. Heisele, T. Serre, and T. Paggio. A component-based framework for face detection and identification. *International Journal of Computer Vision (IJCV), Springer Science and Business Media*, 74(2):167–181, 2006.

[11] B. Heisele, T. Serre, M. Pontil, and T. Paggio. Component-based face detection. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, IEEE Computer Society Press*, volume 1, pages 657–662, 2002.

[12] B. Selin, M.M. Bulut, and A. Volkan. Projection based method for segmentation of human face and its evaluation. In *Pattern Recognition Letters*, volume 23, pages 1623–1629, 2002.

[13] P.T. Bao. Fast multi-face detection using facial component based validation by fuzzy logic. In *Proceedings of the International conference on Image Processing and Computer Vision (IPCV'06), Las Vergas, Nevada, USA*, pages 403–407, 2006.

[14] S.M. Bileschi and B. Heisele. Advances in component based face detection. In *Proceedings of the IEEE International Workshop on Analysis and Modeling of Faces and Gestures (AMFG'03), Computer Society, USA*, 2003.

[15] B.S. Venkatesh, S. Palanivel, and B. Yegnanarayana. Face detection and recognition in an image sequence using eigenedginess. In *Proceedings of Indian Conference on Computer Vision, Graphics and Image Precessing, India*, pages 97–101, 2002.

[16] C. W. Hsu, C. C. Chang, and C. J. Lin. A practical guide to support vector classification. *Technical Report, Department of Computer Science and Information Engineering, National Taiwan University*, 2003.

[17] Candid Camera. Engineering advances at the university of notre dame, signatures. biometric research report. *http://www.nd.edu/˜engineer/publications/signatures/2006/candid.html, (URL accessed on July 26, 2008).*

[18] C. Campbell. Algorithmic approaches to training support vector machines: A survey. In *ESANN'2000, proceeding-European Symposium on Artificial Neurals Networks, Bruges, Belgium*, pages 26–28, april 2000.

[19] D. Damien, C. Christopher, R. Jonas, and D. Andrrzej. Multimodal biometric for identity documents, state-of-the-art. *MBioD Research Report PSF 341-08.05, Unil, France*, September 2005.

[20] D. J. Hurley, M. S. Nixon, and J. N. Carter. Automated ear recognition by force field transformations. *Department of Electronic and Computer Science, University of Southampton, SO17, 1BJ,UK*, 2000.

[21] D. U. Craig, R. M.Edward, E. S.Shellie, and G. Megan. Safe kids, safe schools: Evaluating the use of iris recognition technology in new egypt, nj. *21st Century Solutions, Inc., National Institute of Justice (NIJ), U.S. Department of Justice, USA*, December 2004.

[22] D. Zhang, A. Wong, M. Indrawan, and G. Lu. Content-based image retrieval using gabor feature texture features. In *Proceedings of First IEEE Pacific-Rim Conference on Multimedia (PCM"00), USA*, pages 1–9, June 1-3 2001.

[23] V. Danijela and P. Maja. Fully automatic facial point detection using gabor feature based boosted classifiers. In *Proceeding of IEEE International Conference on Systems, Man and Cybernetics*, pages 1692–1698, October 10-12 2005.

[24] J. V. Fonou Dombeu and J. R. Tapamo. Validation of detected facial components for and accurate face recognition. In *Proceedings of the* 18^{th} *Annual Symposium of Pattern Recognition Association of South Africa (PRASA), Pietermaritzburg, South Africa*, pages 41–46, 28-30 November 2007.

[25] J. V. Fonou Dombeu and J. R. Tapamo. Adaptive detection of regions of interest for face recognition. In *Proceedings of the International Conference on Image Processing, Computer Vision, and Pattern Recognition (IPCV'2008), Monte Carlo Resort, Las Vegas, Nevada, USA*, 14-17 July 2008.

[26] J. V. Fonou Dombeu and J. R. Tapamo. Texture and shape-based face recognition by mixture of gabor filters and zernike moments. In *Proceedings of the International Conference on Information Technology and Applications (ICITA 2008), Cairns Gueensland, Australia*, pages 783–788, 23-26 June 2008.

[27] H. Erik. Feature-based face recognition. *In Proceedings of Norwegian Image processing and Pattern Recognition Conference*, 2000.

[28] B. Duane et al. Iris recognition. *National Science and Technology Council (NSTC), committee on Technology, Committee on Homeland and National Security, Subcommittee on Biometric, USA*, 7 August 2007.

[29] F. A. Afsar, M. Arif, and M. Hussain. Fingerprint identification and verification system using minutiae matching. In *National Conference on Emerging Technologies*, pages 141–146, 2004.

[30] F. Hajati, K. Faez, and S. K. Pakazad. An efficient method for face localization and reognition in color images. In *IEEE Conference on Systems, Man, and Cybernetics, Taipei, Taiwan*, pages 4214–4219, October 8-11 2006.

[31] M. Fabian and A. D. Rubin. Keystroke dynamics as a biometric for authentication. *Future Generation Computing Systems (FGCS) Journal: Security on the Web (special issue)*, 16(4):351–359, March 1999.

[32] R. W. Frischhols and U. Dieckmann. Bioid: A multimodal biometric identification system. In *IEEE Computer*, volume 33, pages 64–68, 2000.

[33] G. Guo, S. Z. Li, and K. Chan. Face recognition by support vector machine. *Journal of Image and Vision Computing, Singapore*, 19:631–638, 1998.

[34] G. loulia and S. Veikko. Detection of facial landmarks from neutral, happy, and disgust facial images. In *WSCG 2005 conference proceedings, Finland*, January 31-February 4 2005.

[35] R. C. Gonzalez and R. E. Woods. *Digital Image Processing, Second Edition.* Prentice Hall, 2002.

[36] H. Bernd, H. Purdy, and P. Tomaso. Face recognition with support vector machines: global versus component-based approach. In *proceedings of IEEE International Conference on Computer Vision, Vancouver Canada*, July 2001.

[37] H. Deng, L. Jin, L. Zhen, and J. Huang. A new facial expression recognition method based on local gabor filter bank and pca plus lda. *International Journal of Information Technology*, 11(11):86–96, 2005.

[38] H. H. J. Kim. Survey paper: Face detection and face recognition. *Research Report, University of Saskatchewan, Departement of Computing Science, Canada*, 2007.

[39] H. Kim, E. Berdahl, N. Moreau, and T. Sikora. Speaker recognition using mpeg-7 descriptors. *Eurospeech, Geneva, Switzerland*, 1-4 September 2003.

[40] H. Thomas, P. Nick, A. Jim, and C. Zezhi. Face recognition: A comparison of appearance-based approaches. In *Proc. 7^{th} Digital Image Computing: Techniques and Applications*, 10-12 December 2003.

[41] L. K. Hanna. Ear biometric, 2005. *http://www.it.lut.fi/kurssit/03-04/010970000/seminars/Lammi.pdf, (URL accessed on May 07, 2008).*

[42] Sun Herald. Gait recognition research could make catching osama a walkover. *Media Monitors, General news, Region: Sydney circulation: 516400, type: Capital City Daily, Ref:24666349*, page 40, 22 October 2006.

[43] L. Hong and A. Jain. Integrating faces and fingerprints for personal identification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(12):1295–1307, 1998.

[44] H. Hse and A. R. Newton. Sketched symbol recognition using zernike moments. *Technical Memorandum UCB/ERL M03/49, Electronics Research Lab, Department of Electrical Engineering and Computer Sciences, California, USA*, 2003.

[45] J. Haddadnia, K. Faez, and P. Moallem. Neural network based face recognition with moment invariants. In *Proceedings of IEEE International Conference On Image Processing, Thessaloniki, Greece*, pages 1018–1021, October 7-10 2001.

[46] A. K. Jain and R. C. Dubes. *Algorithms for Clustering.* Prentice Hall, 1988.

[47] P. Joseph and J. R. Campbell. Speaker recognition: A tutorial. In *Proceeding of the IEEE International Conference, USA*, volume 85, September 1997.

[48] K. Delvac and M. Grgic. A survey of biometric recognition methods. *46th International Symposium Electronics in Marine*, ELMAR-2004:16–18, 2004.

[49] K. Jonsson, J. Matas, J. kittler, and Y.P.Li. Learning support vectors face verification and recognition. In *Proceedings of IEEE International Conference on Automatic Face and Gesture Recognition*, pages 208–213, 1999.

[50] D. Kelenova. Personal authentication using signature recognition. *Department of Information Technology, Laboratory of Information Processing, Lappeenranta University of Technology*, 2003.

[51] P. Kuchi and P. Sethuraman. Gait recognition. *CUbic: Center for Cognitive Ubiquitous Computing*, 2007.

[52] L. Juergen, A. T. Neil, and W. B. Steve. Speaker identification by lipreading. In *Proceedings of the 4th International Conference on Spoken Language Processing (ICSLP'96)*, pages 62–65, 1996.

[53] L. Zhuang, F. Zhu, and J. D. Tygar. Keyboard accoustic emanations revised. In *Proceedings of the 12th ACM Conference on Computer and Communications Security, Alexandria, Virginia, USA*, pages 373–382, November 7-11 2005.

[54] W. Larissa. Online fraud and theft under attack with keystroke recognition. *PressRelease,ID Control BV, HEEMSKERK, North-Holland*, pages 24–7, 10 March 2007.

[55] D. Le and S. Satoh. An efficient selection method for object detection. In *Procceedings of the ICARPR, Springer-Verlag Heidelberg*, volume LNCS 3686, pages 461–468, 2005.

[56] L. Lee and W. E. L. Grimson. Gait analysis for recognition and classification. In *Proceedings of International Conference on Automatic Face and Gesture recognition*, pages 155–162, 2002.

[57] X. Liu. Optimizations in iris recognition. *PhD thesis, University of Notre Dame*, 2006.

[58] J. Louradour. Noyaux de sequences pour la verification du locuteur par machines a vecteurs de support. *These de PhD, Ecole Doctoral Informatique et Telecom, Universite de Toulouse III, France*, pages 24–36, 25 janvier 2007.

[59] M. A. Turk and A. P. Pentland. Face recognition using eigenfaces. In *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition, USA*, pages 586–591, 1991.

[60] M. A. Turk and A. P. Pentland. Face recognition using eigenfaces. In *Journal of Cognitive Neuroscience, USA*, volume 3, 1991.

[61] M. Nikolas, E. Thomas, R. Florence, A. Frederic, and A. Amara. Revue des algorithmes pca, lda et ebgm utilises en reconnaissance 2d du visage pour la biometrie. *Institut Superieur de Paris (ISEP), Departement d'Electronique, 21, rue d'Assas 75270,Paris Cedex 06*, 2006.

[62] M. Sonka, V. HLavac, and R. Boyle. *Image Processing, Analysis and Machine Vision*. Pws Publishing, 1999.

[63] M. Zhou, H. Wei, and S. Maybank. Gabor wavelets and adaboost in feature selection for face verification. *In Proceedings of Applications of Computer Vision 2006 workshop in conjuction with ECCV 2006, Graz, Austria*, pages 101–109, May 2006.

[64] J. McQueen. Some methods for classification and analysis of multivariate observations. In *Proceedings of the* 5^{th} *Symposium on Mathematical Statictics and Probability*, pages 281–197, 1967.

[65] N. L. Alexandra, Wong, and S. Pengcheng. Peg-free hand geometry recognition using hierarchical geometry and shape matching. *Proceedings of the IAPR Conference on Machine Vision Applications (IAPR MVA 2002), Nara-ken New Public Hall, Nara, Japan*, pages 281–284, December 11-13 2002.

[66] S. Nimalan and L. shahram. A survey of unimodal biometric methods. In *Proceedings of the International Conference on Security and Management (SAM'06), Monte Carlo Resort, Las Vegas, Nevada, USA.*, pages 57–63, June 26-29 2006.

[67] P. H. Chen, C. J. Lin, and S. Bernhard. A tutorial on v-suppot vector machines. In *Proceedings of the International Conference on Applied Stochastic Models in Business and Industry*, volume 21, pages 111–136, 2003.

[68] V. Perlibakas. Distance for pca-based face recognition. *Pattern Recognition Letters*, 25:711–724, 2004.

[69] F. P. Pretarata and I. Shamos. *Computational Geometry: An Introduction*. Springer-Verlag, 1991.

[70] Q. Tong, A. E. Saddik, and A. Adler. A stroke based algorithm for dynamic signature verification. In *Proceedings of the 17th IEEE CCECE 2004-CCGEI 2004, Niagara Falls, Ontario, Canada*, May 2004.

[71] R. Lim, M. J. T. Reinders, and Thiang. Facial landmark detection using a gabor filte representation and a genetic search algorithm. In *Proceeding, Seminar of Intelligent Technology and Its Applications (SITIA'2000), Surabaya*, 19 April 2000.

[72] R. M. .Bolle, A. W. Senior, N. K. Ratha, and S. Pankanti. Fingerprint minutiae: A constructive definition. In *Proceedings of ECCV workshop on Biometrics. Springer LNCS*, volume 2359, pages 58–66, June 1 2002.

[73] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification.* Wisley, 2001.

[74] R. Snelick, U. Uludag, A. Mink, M. Indovina, and A. K. Jan. Large scale evaluation of multimodal biometric authentication using state-of-the-art systems. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, volume 27, pages 450–455, 2005.

[75] W. S. Rasband. Imagej. *U.S. National Institutes of Health, Bethesda, Maryland, USA, http://rsb.info.nih.gov/ij/, (URL accessed on May 23, 2008).*

[76] G. X. Ritter and J. N. Wilson. *Handbook of Computer Vision, Algorithms in Image Algebra, Second Edition.* CRC press LLc, 2001.

[77] A. Ross and A. Jain. Information fusion in biometrics. *Pattern Recognition Letters*, 24:2115–2125, 2003.

[78] A. Ross and A. K. Jain. Multimodal biometrics: An overview. In *Proceedings of* 12^{th} *European Signal Processing Conference (EUSIPCO), Vienna, Austria*, pages 1221–1224, September 2004.

[79] S. Mohammed, L. Sylvie, V. Vincent, E. Bedda, and Las. Face detection by neural network trained with zernike moments. In *Proceedings of the 6th WSEAS International Conference on Signal Processing, Robotics and Automation, Corfu Island, Greece*, pages 36–41, February 16-19 2007.

[80] S. Shiguang, G. Wen, and Z. Debin. Face recognition based on face-specific subspace. *Wiley Periodicals, Inc, Institute of Computer Technology, Department of Computer Science, Harbin, China*, 2003.

[81] A. Saradha and S. Annadurai. A hybrid feature extraction approach for face recognition systems. *ICGST International Journal on Graphics, Vision and Image Processing*, 5, 2005.

[82] National Science and Technology Council(NSTC). Hand geometry. 27 March 2006.

[83] L. Shen and L. Bai. Information theory for gabor feature selection for face recognition. *Hindawi Publishing Corporation, EURASIP Journal on Applied Signal Processing*, Article ID 30274:1–11, 2006.

[84] Y. Tian and R.M. Bolle. Automatic detecting neutral face for face authentification and facial expression analysis. *Computer Science Research Report, IBM Thomas J. Watson Research Center*, 2002.

[85] V. Perlibakas. Distance measures for pca-based face recognition. *Pattern Recognition Letters*, 25:711–725, 2004.

[86] V. Raymond, B. Asker, B. Wim, and H. Anne. Hand geometry recognition based on contour parameters. In *SPIE Defense and Security Symposium, Orlando*, 2005.

[87] V. Y. Roman and G. Venu. Similarity measure functions for strategy-based biometrics. *International journal of Biometric Science*, 1(1):224–229, 2006.

[88] B. K. Vinayak. A color code algorithm for signature recognition. In *ELCVIA, ISSN:1577-5097, Computer Vision Center, Barcelona, Spain*, number 6(1), pages 1–12, 16 January 2007.

[89] W. Zhang, S. Shan, X. Chen, and W. Gao. Are gabor phases really useless for face recognition? In *Proceeding of the International Conference on Pattern Recognition (ICPR2006)*, volume 4, pages 606–609, 2006.

[90] W. Zhao, R. Chellapa, and A. Rosenfeld. Face recognition: A literarture survey. *UMD CfAR Technical Report CAR-TR-948, National Institute of Standards and Technology, USA*, 2000.

[91] J.G. Wang and E. Sung. Frontal-view face detection and facial feature extraction using color and morphological operations. *Pattern Recognition Letters*, 20:1053–1068, july 1999.

[92] A. B. William. A survey of face recognition algorithms and testing results. In *Proceedings of the 31th Asilomar Conference on Signals, Systems, and Computers*, pages 301–305, 1998.

[93] X. Tan, S. Chen, Z. Zhou, and F. Zhang. Face recognition from single image per person: A survey. *Pattern Recognition Letters*, 39:1725–1745, September 2006.

[94] D. Xiaoqing and F. Chi. Discussion on some problems in face recognition. In *proceedings of the International Conference on Advance in Biometric Person Authentication, Guangzhou, China*, number LNCS 3338, pages 47–56, 2004.

[95] Y. Nong and L. Xiangyang. A scalable clutering technique for instrusion signature recognition. In *Poceeding of the 2001 IEEE, Workshop on Information Assurance and Security, USA*, 2001.

[96] Y. Ping and W. B. Kevin. Empirical evaluation of ear biometrics. *Empirical Evaluation Methods in Computer Vision, (EEMCV 2005), San Diego*, June 2005.

[97] Y. Taro, H. Wu, and M. Yachida. Automatic detection of facial feature points and contours. In *IEEE International Workshop on Robot and Human Communication*, number 0- 7803-3253-9/96, 1999.

[98] Y. Wang, T. Tan, and A. K. Jain. Combining face and iris biometrics for identity verification. In *proceeding of* 4^{th} *International Conference on Audio and Video-based Biometric Person Authentication (AVBPA), j. kittler and M. Nixon, Eds.*, volume LNCS 2688, pages 805–813, 2003.

[99] Z. Rong, V. Christian, and M. Dimitris. Human gait recognition. In *Proceedings of the Conference on Computer Vision and Pattern Recognition Workshop (CVPRW'04)*, volume 1, page 18, 2004.

[100] Z. Y. Mei, Z. Ming, and G. YuCong. Face recognition based on low diamensional gabor feature using direct fractional-step lda. In *Proceedings of the Computer Graphics, Image and Vision: New Treds (CGIV'05), IEEE Computer Society*, 2005.

[101] Z. Zhang. Feature-based facial expression recognition:experiments with a multi-layer perceptron. *Institut de Recherche en infromatique et en Automatique (INRIA), Rapport de Recherche No. 3354, Sophia Antipolis Cedex, France*, Fevrier 1998.

[102] S. Zoltan and S. Tamas. Face analysis using cnn-um. In *Proceedings of IEEE International Workshop on Cellular Neural Networks and their Applications(CNNA 2004), Budapest*, pages 190–196, December 2004.

Appendix A

UML Class Diagram of The System and Samples Outputs Results

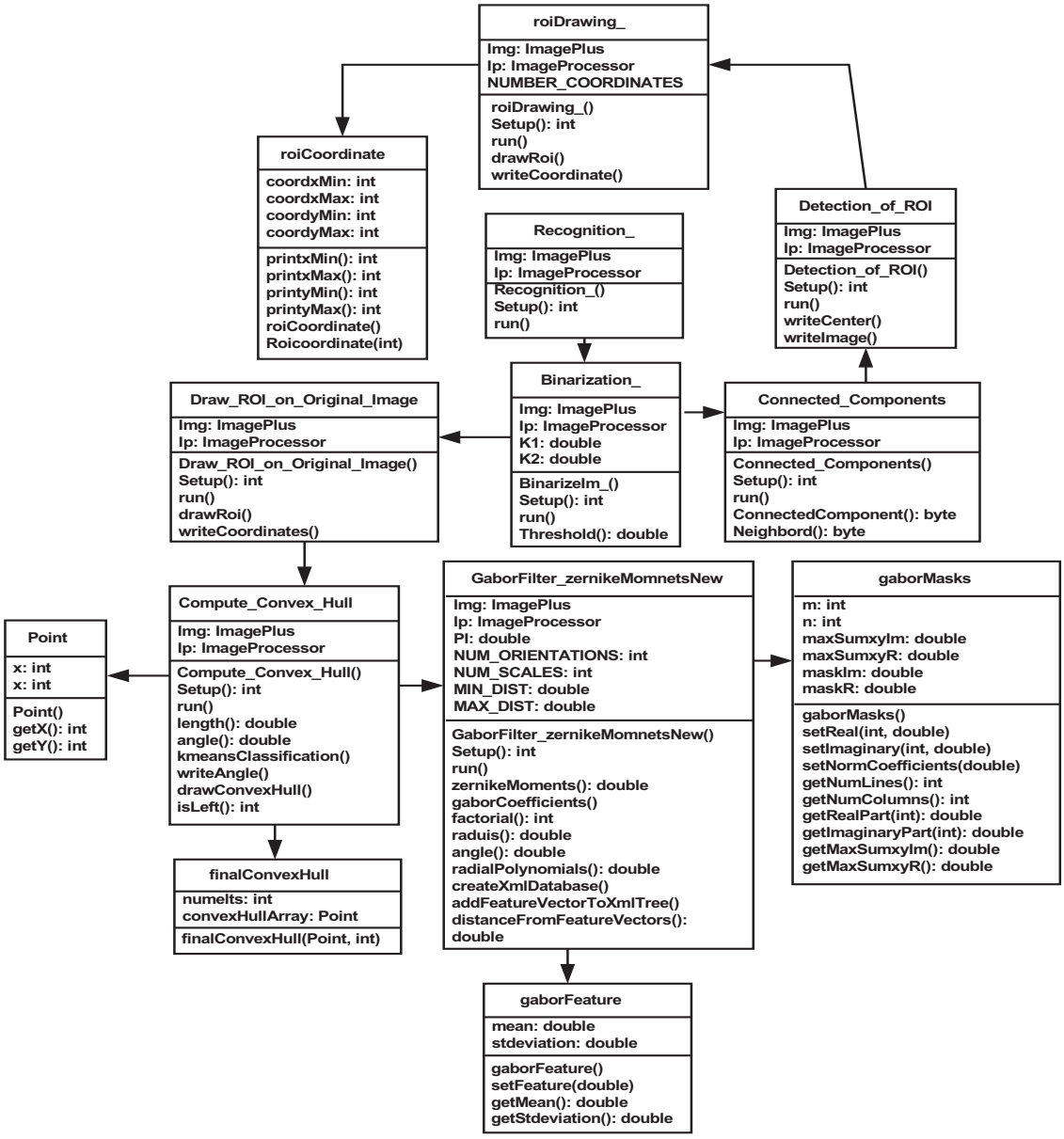


Figure A.1: UML Class Diagram of the System.

List and Roles of Plugins of the System

- **Recognition:** Accept or reject a claimed identity.
- **Binarization:** Binarize the input image.
- **Connected Components:** Construct the connected components of the binary image.
- **Detection of ROI:** Detect iteratively facial components.
- **Draw ROI Bounding Box:** Draw bounding box around detected components.
- **Draw ROI on Original Image:** Draw bounding box of detected components on the original image.
- **Compute Convex Hull:** Compute convex hull of the centroids of detected components.
- **GaborFilter zernikeMomentsNew:** Create temple database; compute feature vectors; classify/recognize.

NB: The space in the name of certain plugins correspond to the character underscore in ImageJ naming rules.

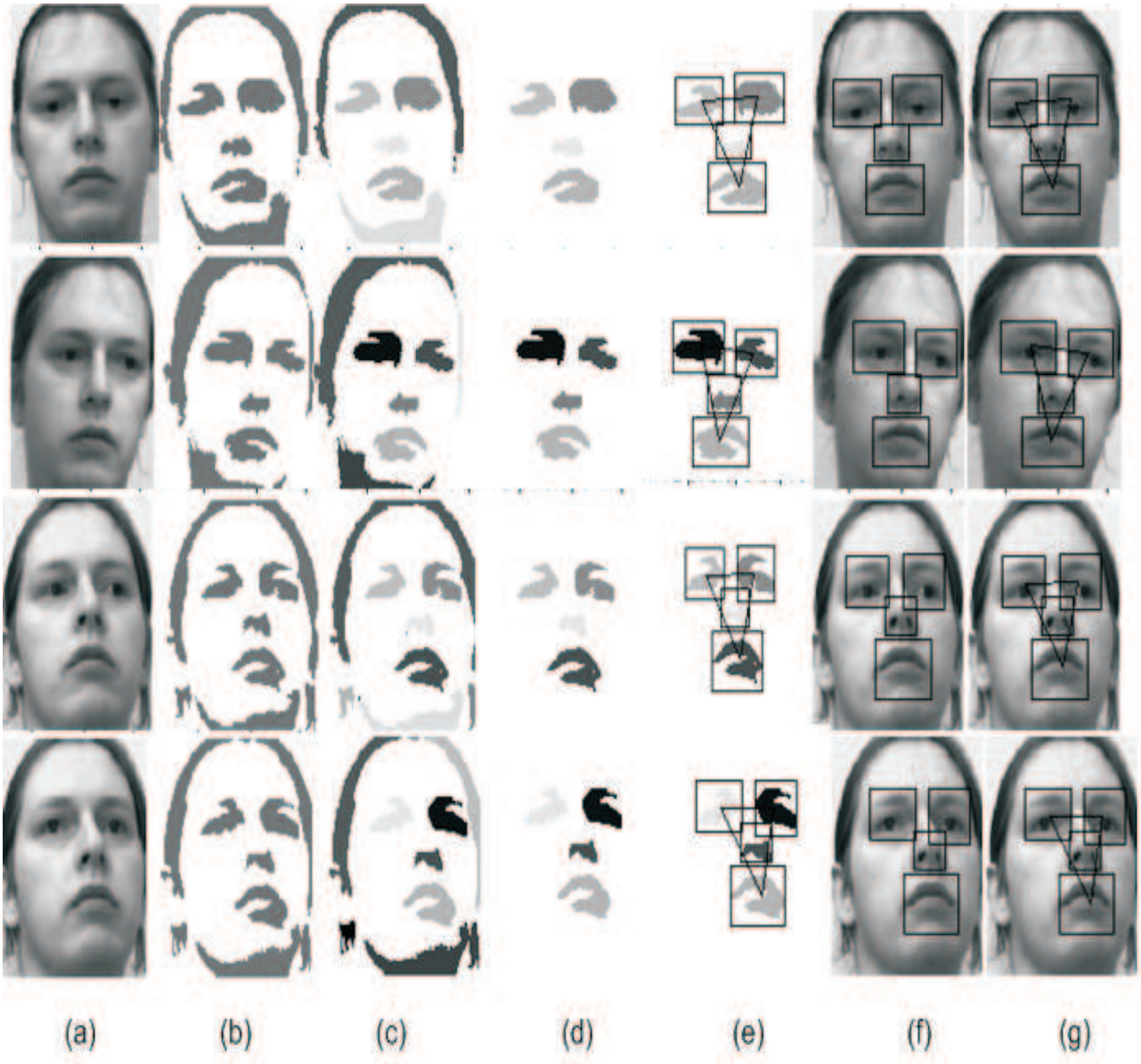


Figure A.2: Sample Output Results of Plugins; (a): Original Images; (b): Binary Images; (c): Connected Components of Binary Images; (d): Detected Components; (e): Detected Components with Bounding Box and Convex Hull; (f): Original Images with Bounding Box of Detected Components; (g): Original Images with Bounding Box and Convex Hull of Detected Components.

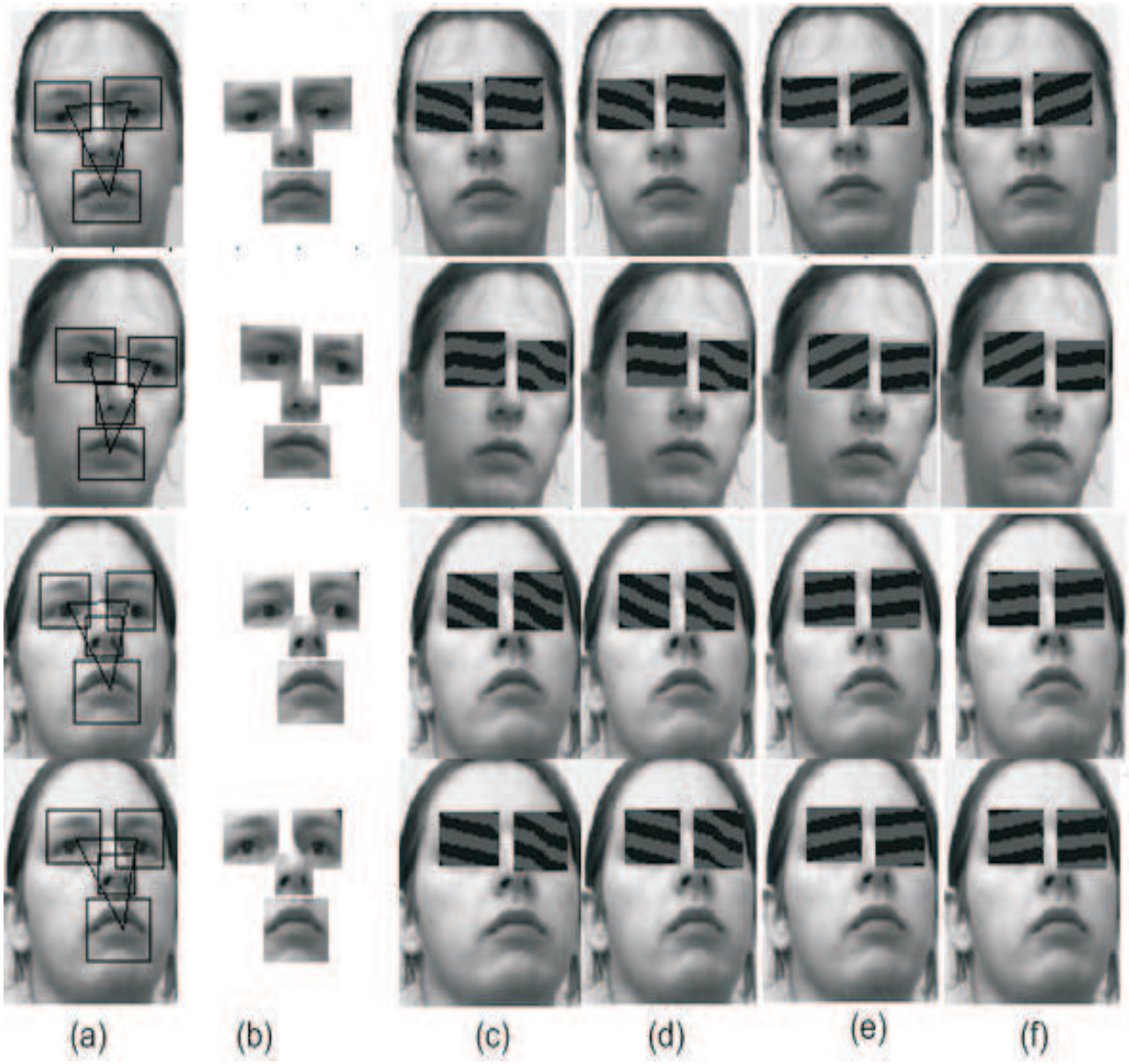


Figure A.3: Sample Output Results of Plugins; (a): Input Images with Detected and Validated Facial Components; (b): Extracted Facial Components; (c,d):Original Images with Imaginary and Real Parts of Gabor Filters at Scale $\pi/4$ and Orientation $5\pi/8$; (e,f):Original Images with Imaginary and Real Parts of Gabor Filters at Scale $\pi/2\sqrt{2}$ and Orientation $3\pi/8$.

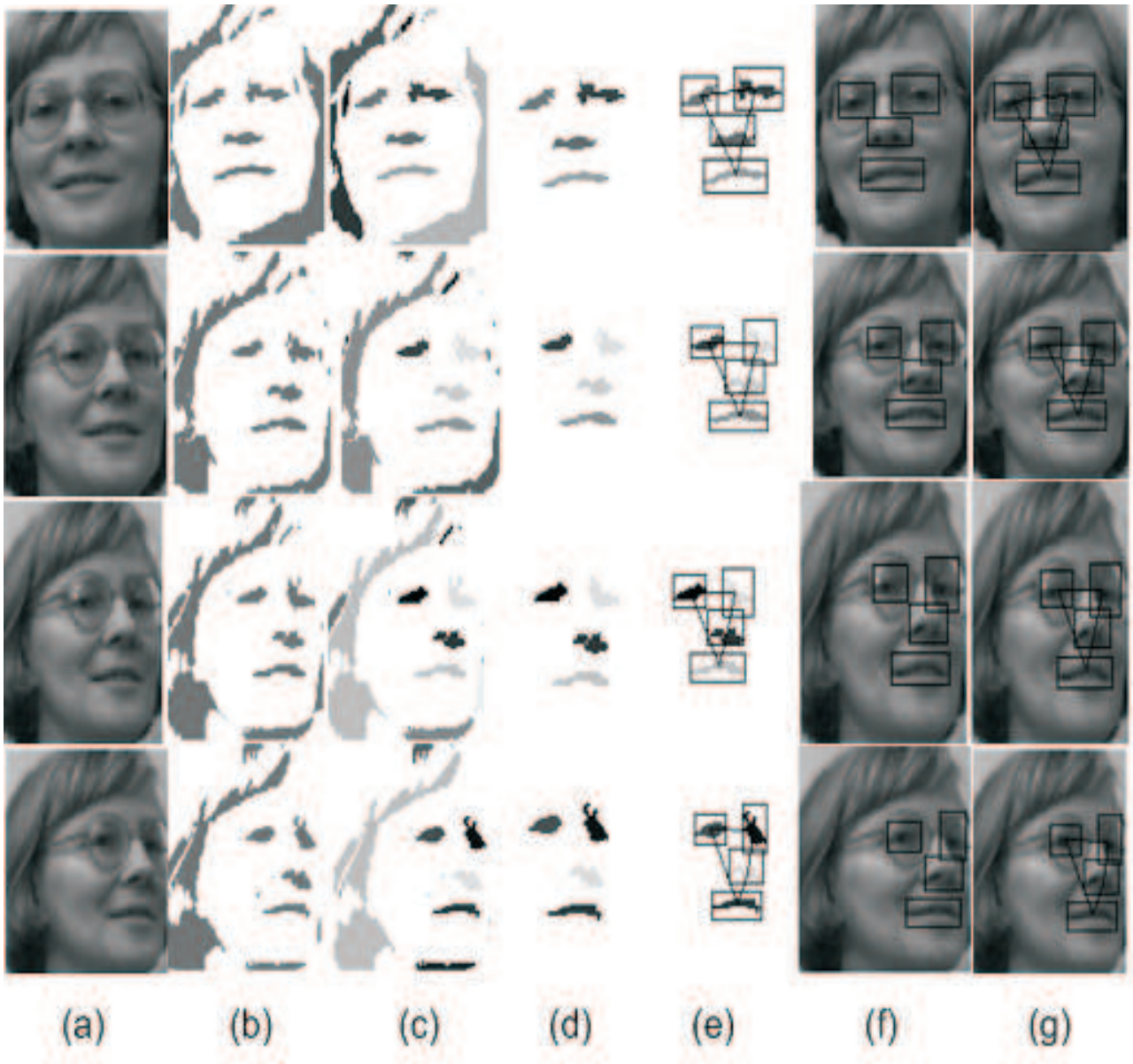


Figure A.4: Sample Output Results of Plugins; (a): Original Images; (b): Binary Images; (c): Connected Components of Binary Images; (d): Detected Components; (e): Detected Components with Bounding Box and Convex Hull; (f): Original Images with Bounding Box of Detected Components; (g): Original Images with Bounding Box and Convex Hull of Detected Components.

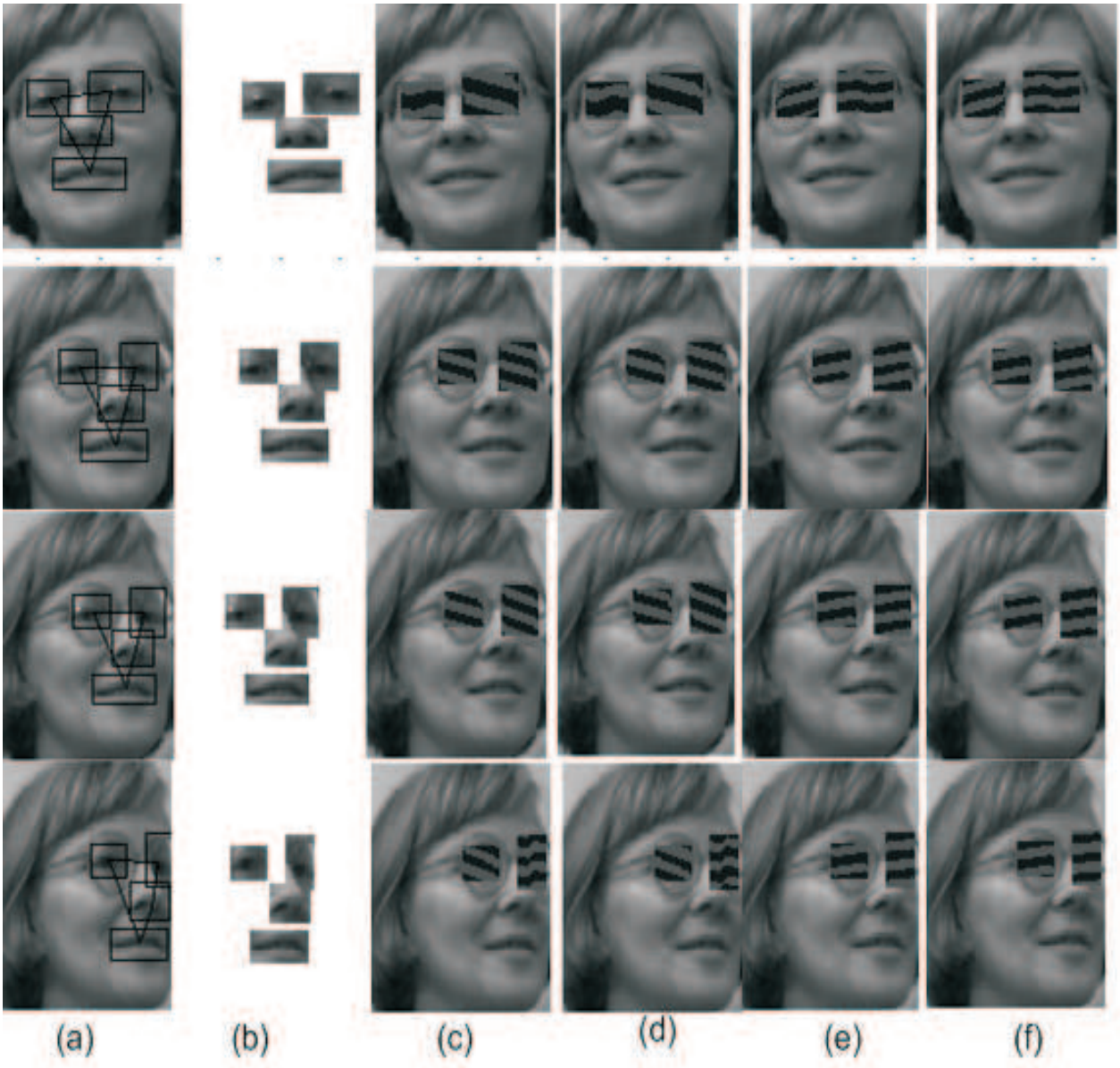


Figure A.5: Sample Output Results of Plugins; (a): Input Images with Detected and Validated Facial Components; (b): Extracted Facial Components; (c,d):Original Images with Imaginary and Real Parts of Gabor Filters at Scale $\pi/4$ and Orientation $5\pi/8$; (e,f):Original Images with Imaginary and Real Parts of Gabor Filters at Scale $\pi/2\sqrt{2}$ and Orientation $3\pi/8$.

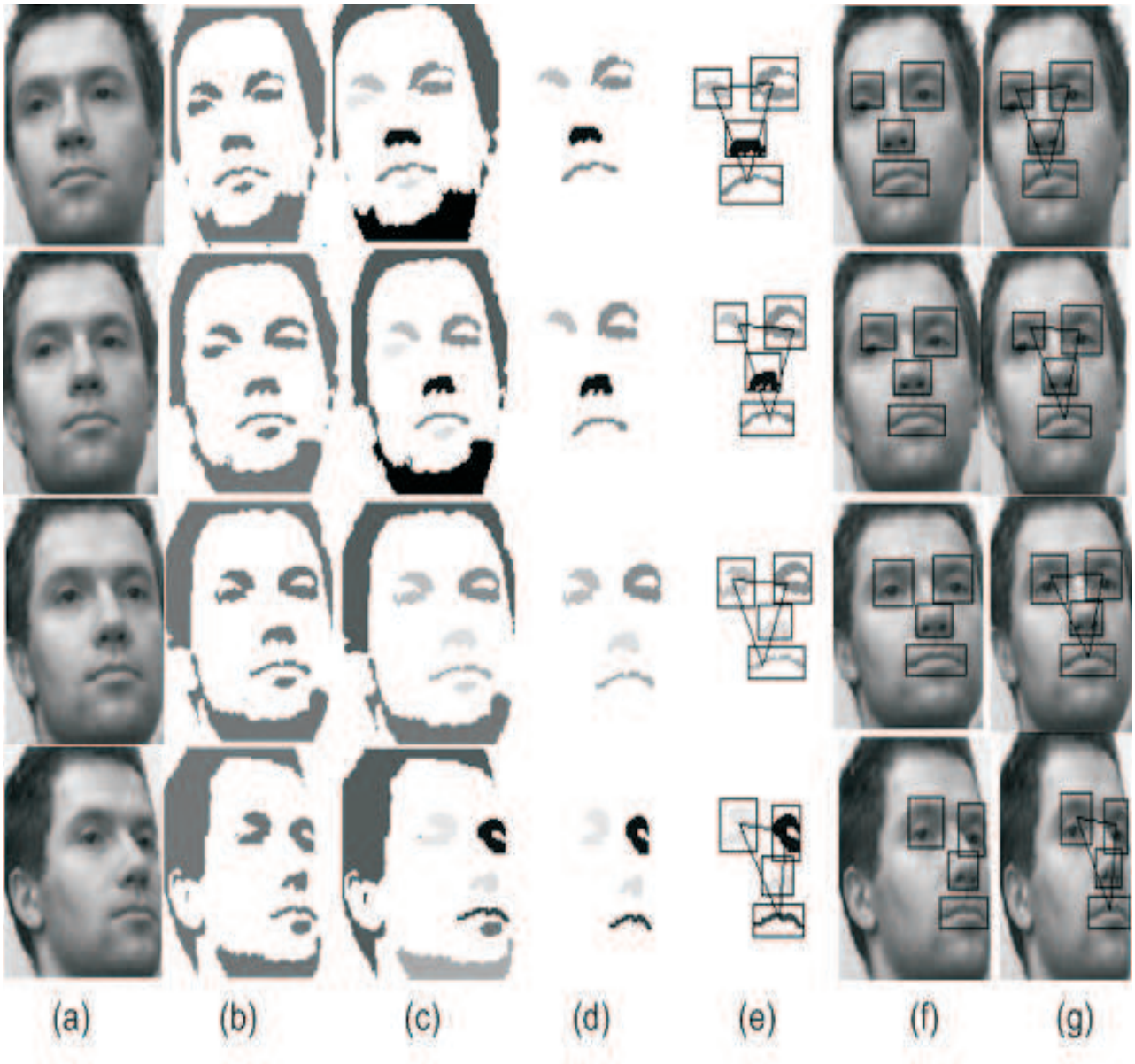


Figure A.6: Sample Output Results of Plugins; (a): Original Images; (b): Binary Images; (c): Connected Components of Binary Images; (d): Detected Components; (e): Detected Components with Bounding Box and Convex Hull; (f): Original Images with Bounding Box of Detected Components; (g): Original Images with Bounding Box and Convex Hull of Detected Components.

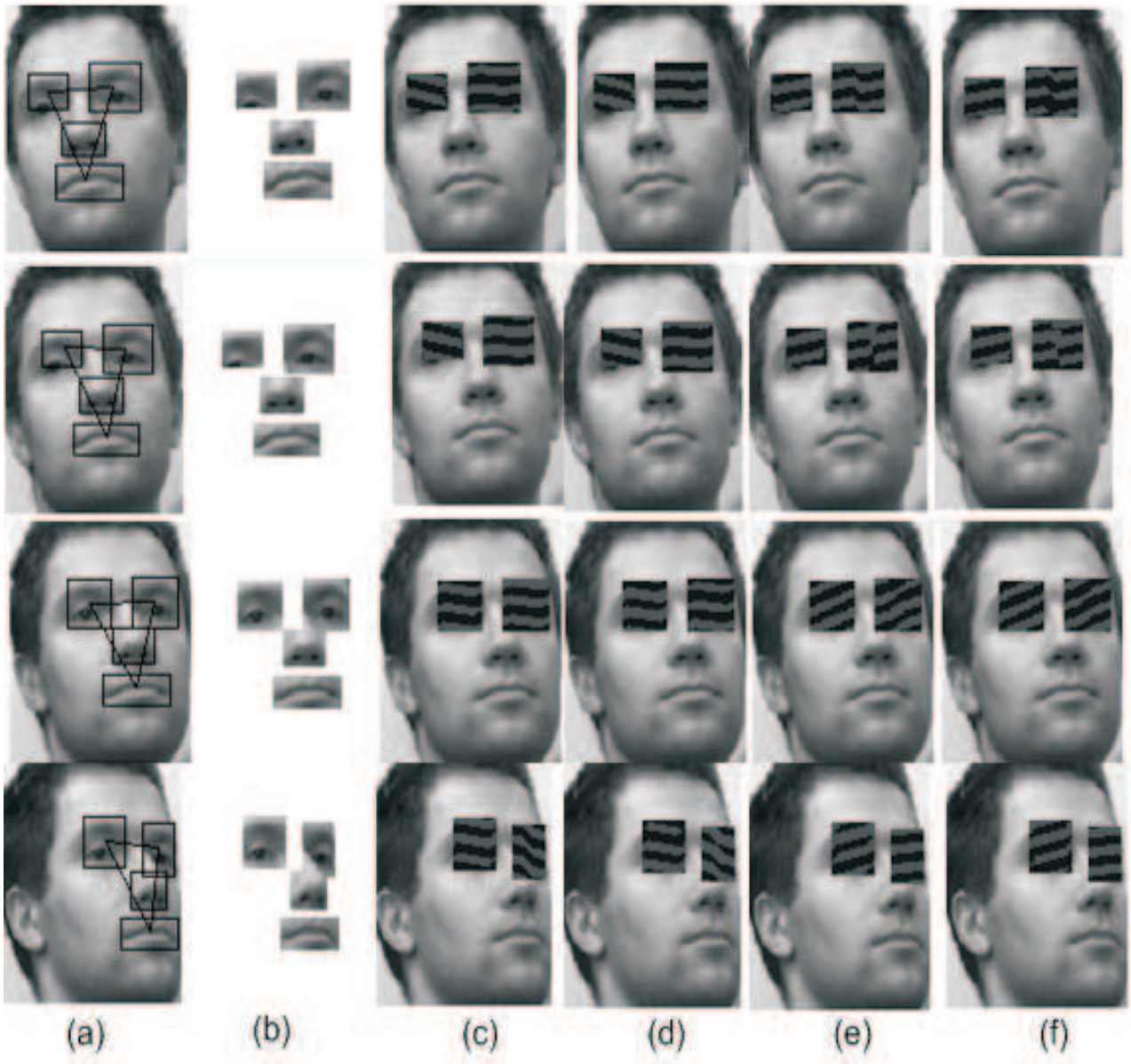


Figure A.7: Sample Output Results of Plugins; (a): Input Images with Detected and Validated Facial Components; (b): Extracted Facial Components; (c,d):Original Images with Imaginary and Real Parts of Gabor Filters at Scale $\pi/4$ and Orientation $5\pi/8$; (e,f):Original Images with Imaginary and Real Parts of Gabor Filters at Scale $\pi/2\sqrt{2}$ and Orientation $3\pi/8$.

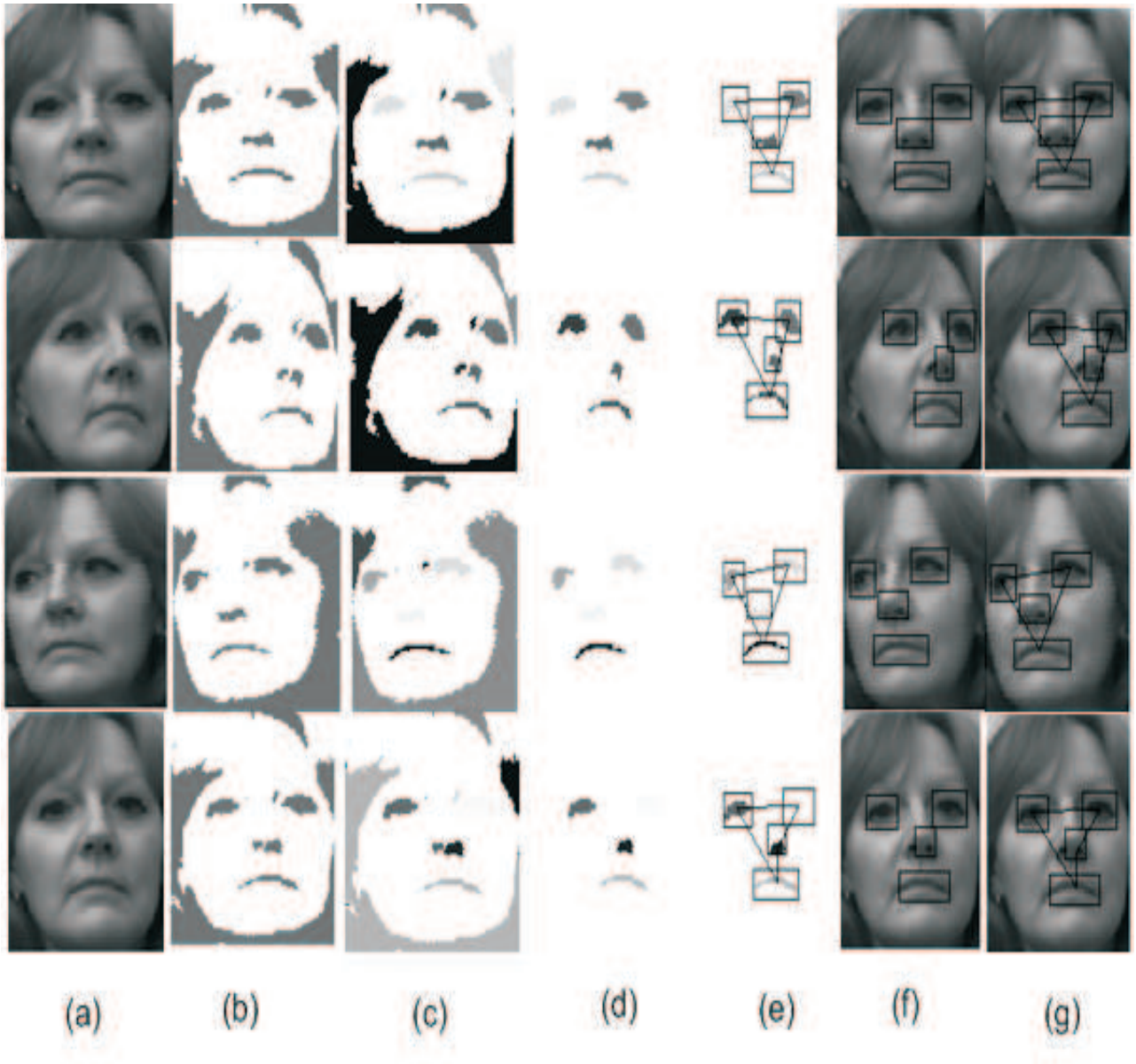


Figure A.8: Sample Output Results of Plugins; (a): Original Images; (b): Binary Images; (c): Connected Components of Binary Images; (d): Detected Components; (e): Detected Components with Bounding Box and Convex Hull; (f): Original Images with Bounding Box of Detected Components; (g): Original Images with Bounding Box and Convex Hull of Detected Components.

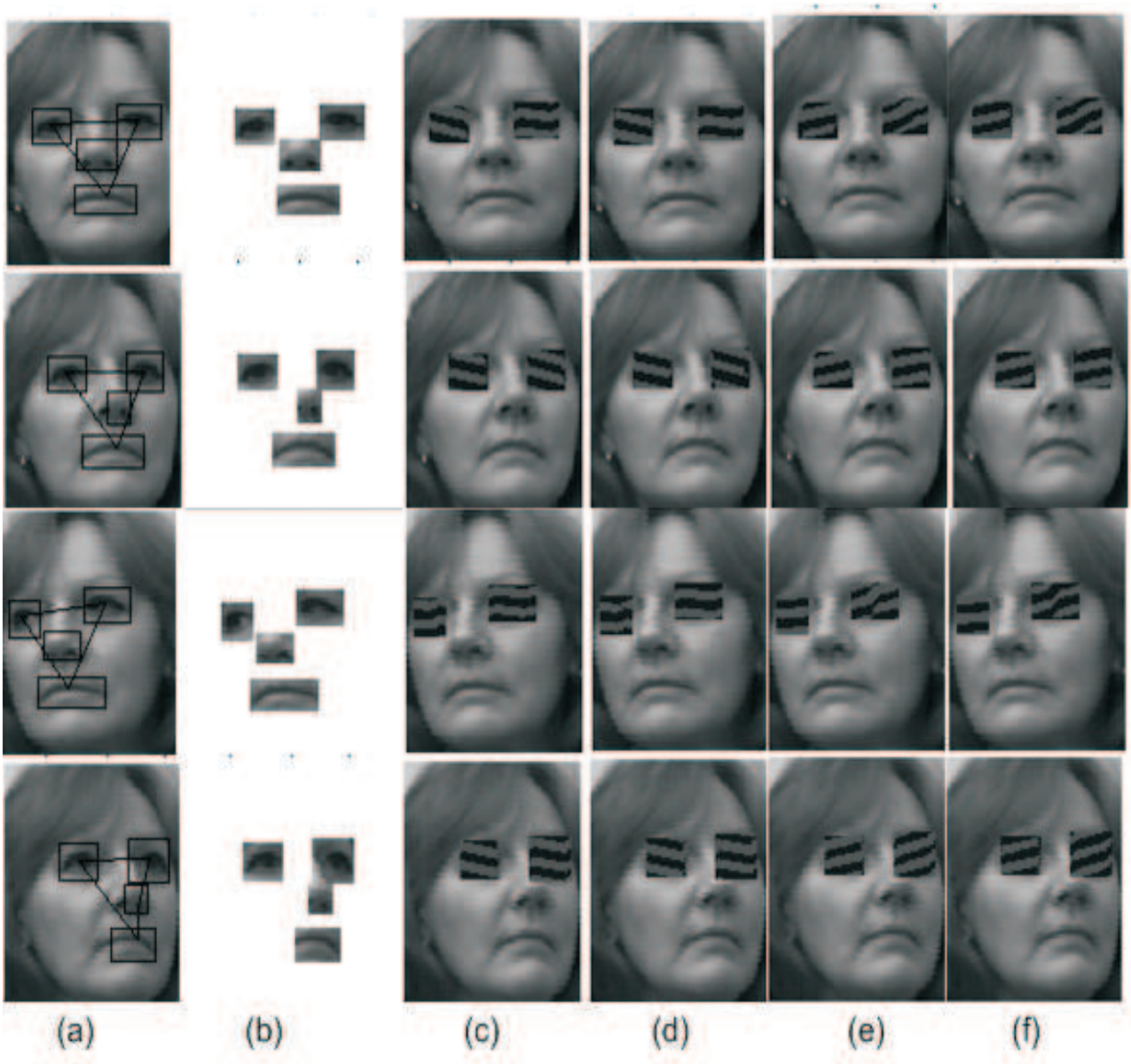


Figure A.9: Sample Output Results of Plugins; (a): Input Images with Detected and Validated Facial Components; (b): Extracted Facial Components; (c,d):Original Images with Imaginary and Real Parts of Gabor Filters at Scale $\pi/4$ and Orientation $5\pi/8$; (e,f):Original Images with Imaginary and Real Parts of Gabor Filters at Scale $\pi/2\sqrt{2}$ and Orientation $3\pi/8$.

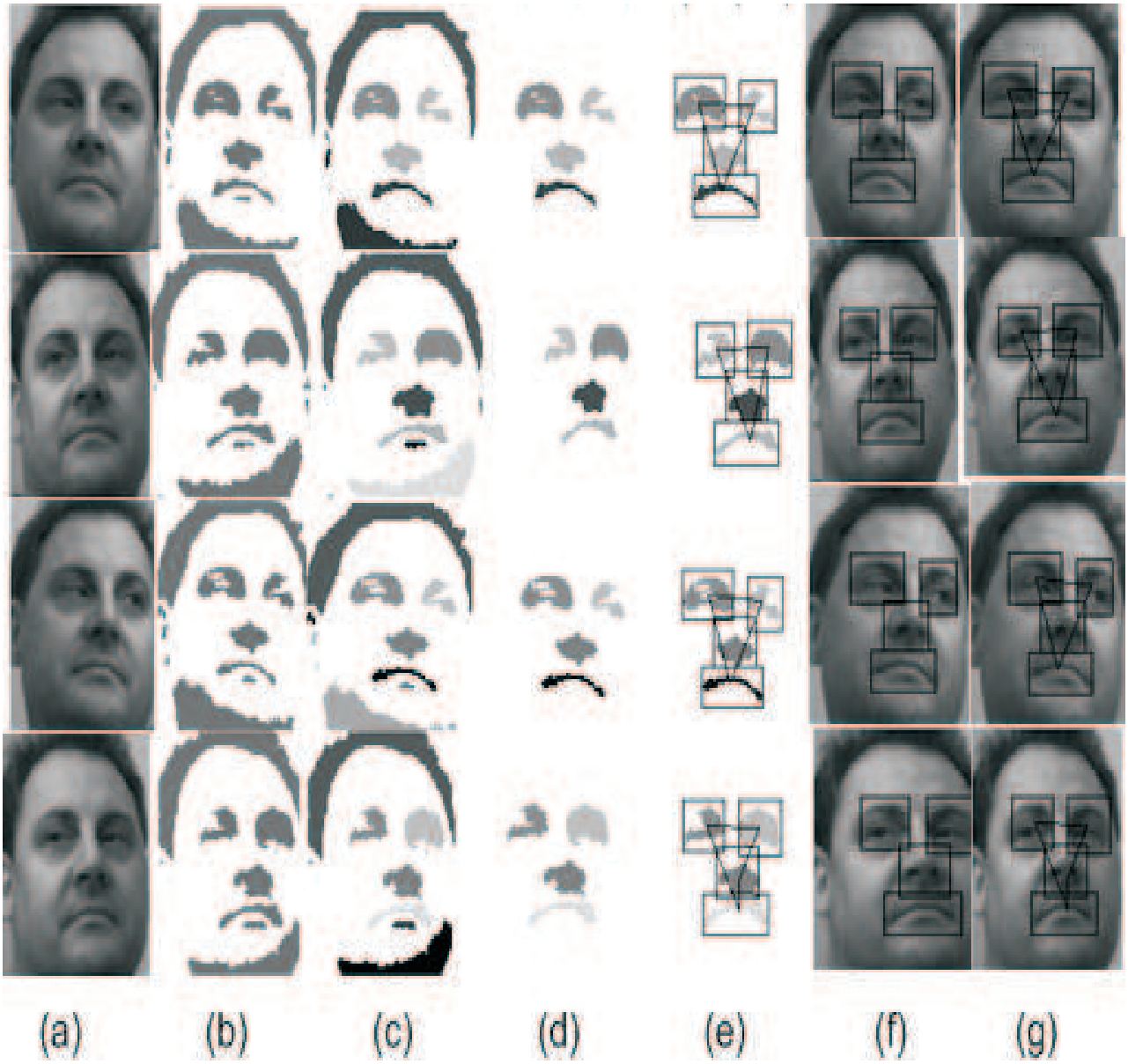


Figure A.10: Sample Output Results of Plugins; (a): Original Images; (b): Binary Images; (c): Connected Components of Binary Images; (d): Detected Components; (e): Detected Components with Bounding Box and Convex Hull; (f): Original Images with Bounding Box of Detected Components; (g): Original Images with Bounding Box and Convex Hull of Detected Components.

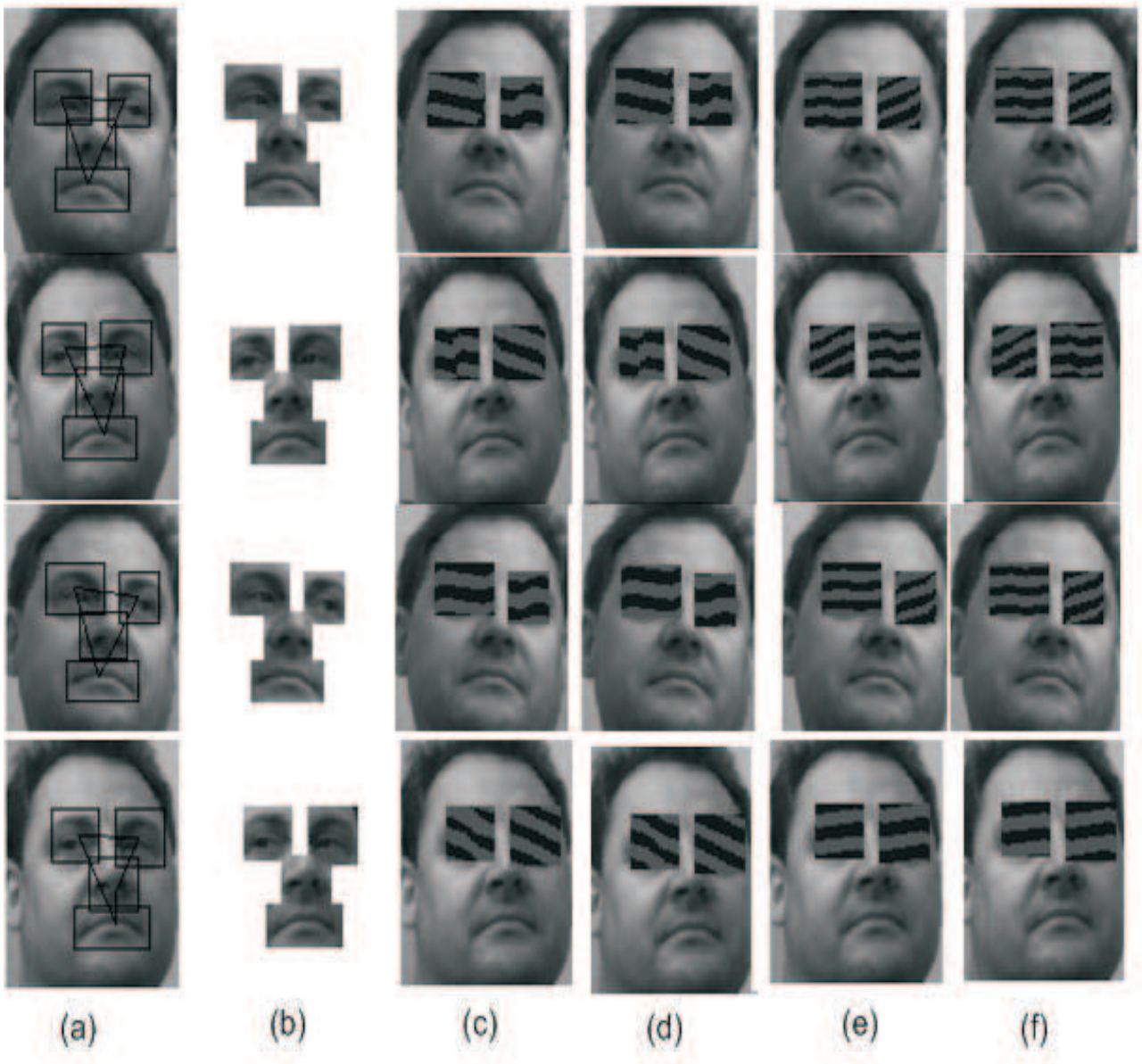


Figure A.11: Sample Output Results of Plugins; (a): Input Images with Detected and Validated Facial Components; (b): Extracted Facial Components; (c,d):Original Images with Imaginary and Real Parts of Gabor Filters at Scale $\pi/4$ and Orientation $5\pi/8$; (e,f):Original Images with Imaginary and Real Parts of Gabor Filters at Scale $\pi/2\sqrt{2}$ and Orientation $3\pi/8$.