SKIN TEXTURE FEATURES FOR FACE RECOGNITION

A Thesis Submitted to The

University of Kent

For the Degree of Ph.D.

In Electronic Engineering,

specializing in Computer Vision

by

Garsah Farhan Al-Qarni

November 2012

Abstract

Face recognition has been deployed in a wide range of important applications including surveillance and forensic identification. However, it still seems to be a challenging problem as its performance severely degrades under illumination, pose and expression variations, as well as with occlusions, and aging. In this thesis, we have investigated the use of local facial skin data as a source of biometric information to improve human recognition.

Skin texture features have been exploited in three major tasks, which include (i) improving the performance of conventional face recognition systems, (ii) building an adaptive skin-based face recognition system, and (iii) dealing with circumstances when a full view of the face may not be available. Additionally, a fully automated scheme is presented for localizing eyes and mouth and segmenting four facial regions: forehead, right cheek, left cheek and chin. These four regions are divided into non-overlapping patches with equal size. A novel skin/non-skin classifier is proposed for detecting patches containing only skin texture and therefore detecting the pure-skin regions. Experiments using the XM2VTS database indicate that the forehead region has the most significant biometric information. The use of forehead texture features improves the rank-1 identification of Eigenfaces system from 77.63% to 84.07%. The rank-1 identification is equal 93.56% when this region is fused with Kernel Direct Discriminant Analysis algorithm.

The second proposed algorithm presents an adaptive strategy for combining distance scores between images with different numbers of usable skin patches. The results are compared with the results of other methods. It is evident that the choice of the distance metric is important as it affects the classification performance. For instance, the mean of rank-1 identification rates of the proposed method is equal to 87.12% using l_1norm distance and 80.25% using l_2norm distance.

Finally, a scheme is suggested for face recognition in scenarios where only a partial image of the face is available for classification. A novel segmentation algorithm is presented for extracting the forehead region without the use of eye locations. This system is tested on FRGC data and presents promising recognition results for a very challenging problem.

Dedication

- *To my brother Saad .. Without his sacrifice, support and encouragement during our stay away from home, there would never have been any chance for this thesis to happen ..*

- *To the memory of my beloved Mother .. The light in my life .. For instilling in me the love of reading and the pursuit of knowledge from my early sapling years .. This degree was one of her dreams but unfortunately it will happen after she left ..*

- *To my Father .. without his warm love, care, sincere prayers and support, it would have been impossible for me to be myself and to continue learning ..*

- *To my wonderful sisters .. Saada, Fauziah, Munirah, Fatimah and their kids .. I deeply appreciate their unlimited and incredible support, sincere prayers and amazing friendship ..*

I owe an impressive dept and feel unable to
sufficiently thank all of you .. With all the love ♡

Garsah ..
Canterbury
28 October 2012

Acknowledgements

... and they shall say: "Praise be to Allah, who hath guided us to this: never could we have found guidance, had it not been for the guidance of Allah"

The Holy Quran (Alaaraf-verse 43)

All praise and thanks be to Allah Almighty for giving me the knowledge, strength and patience to complete this work.

I would like to express my gratitude to my supervisor Dr.Farzin Deravi for his excellent supervision over the past three years. I sincerely thank him for his wisdom, invaluable comments and encouragement as well as his cooperation.

I would like to thank Dr.Albandari Alyousif for nominating me for this scholarship. I also would like to thank and Dr.Konstantinos Sirlantzis and Dr.Sanaul Hoque for their useful lectures in my first year. Thanks to Dr.Ahmed Shihab and my colleagues in the lab Shivanand Gunnes and Sam Chindaro for interesting discussions.

Finally, thanks to my friends Eklas, Nassim, Esmat and the Islamic society members in Canterbury who make the moments happy.

Contents

List of Tables

List of Figures

List of Abbreviations

AAM	Active Appearance Model
Adaboost	Adaptive boosting
AIC	Akaike's Information Criterion
ANL	Adaptive Non-Local technique
ANN	Artifficial Neural Networks
ASM	Active Shape Model
BS	Backward Selection
CBIR	Content-Based Image Retrieval
DCT	Discrete Cosine Transform
DET	Detection Error Trade-Off
DHT	Discrete Hadamard Transform
DLA	Dynamic Link Architecture
D-LDA	Direct LDA
DoG	Difference of Gaussian
DR	Dimensionality Reduction
DST	Discrete Sine Transform
DVFs	Distance Vector Fields
EBGM	Elastic Bunch Graph Matching
EBP	Elliptical Binary Pattern
EER	Equal Error Rate
ELBP	Elongated LBP

FA	False Acceptance
FAR	False Acceptance Rate
FR	False Rejection
FRR	False Rejection Rate
FRVT	Face Recognition Vendor Test
FS	Forward selection
GA	Genetic Algorithms
GDA	Kernel LDA
GEF	Gabor Elementary Functions
GLC	Grey-Level Co-occurrence features
GLCM	Gray-Level Co-occurrence Matrices
GLD	Gray-Level Difference
GLOH	Gradient Location and Orientation Histogram
HGPP	Histogram of Gabor Phase Patterns
IS	Istoropic-diffusion based normalization
KDDA	Kernel D-LDA
KLT	Karhunen-Loeve Transform
kNN	k- Nearest Neighbour
KPCA	Kernel PCA
LBP	Local Binary Pattern
LDA	Linear Discriminative Analysis
LEP	Local Edge Patterns
LEPINV	Local Edge Patterns for Image Retrieval
LEPSEG	Local Edge Patterns for Segmentation

LGBPHS	Local Gabor Binary Pattern Histogram Sequence
LLT	Local Linear Transform
LoG	Laplacian-of-Gaussian
LPCA	Laplacian PCA
LPP	Locality Preserving Projection
LTP	Local Ternary Pattern
MB-LBP	Multi Block LBP
MBP	Median Binary Pattern
MSMD	Multi-Scale/Multi-Directional
MSR	Multi Scale Retinex
NCC	Normalized Cross Correlation
OSH	Optimal Separating Hyper
PCA	Principal Component Analysis
PFR	Partial Face Recognition
ROC	Receiver Operating Characteristic
SGLDM	Spatial Gray Level Dependence Method
SIFT	Scale-Invariant Feature Transform
SPR	Statistical Pattern Recognition
SRC	Sparse Representation Classiffier
SSS	Small Sample Size problem
SURF	Speeded Up Robust Features
SVMs	Support Vector Machines
WA	Wavelet-based normalization technique
ZS	Z-Score Normalization

Chapter 1

Introduction

With the advent of e-commerce, smart cards and electronic banking and increased concerns regarding security and privacy of information stored in various databases, automatic personal authentication has become a desirable capability. Secure access to computer networks, buildings, laptops, automatic teller machines, and the use of machine readable identity document such as passports and driver licenses are some examples where establishing a persons identity is important.

Nowadays, the access to restricted systems has mostly been managed by token-based security (ID cards) or by knowledge-based security (passwords). However, such security measures can easily fail to verify the validity of the user when a password is forgotten or a card is lost. The emergence of *biometrics technology* has addressed such weakness in the traditional verification methods. Biometrics refers to "*Biometrics refers to automated methods of recognizing a person based on physiological or behavioral characteristics*" [121].

Various biometric modalities have been successfully used in recognition systems, such as, face, fingerprint, voice, iris, hand geometry and signature. Despite the fact that other methods of recognition (such as fingerprint, or iris scans) can be more accurate, face recognition has always remained a major focus of research and application because it is non-intrusive, natural and easy to use. Table 1.1 lists some applications of facial recognitions systems.

Areas	**Specific Application**
Civil applications and law enforcement	• National ID cards, passport, drivers license, border control • Surveillance of public places (airports, metro stations, etc) • Forensic applications
Security applications for electronic transactions and access control	• Physical access • Secure access to networks and infrastructures • e-health, e-commerce, e-banking (and now mobile)
Ambient Intelligence	• Smart homes • Natural human-machine interaction
Wearable systems	• Memory aids and context-aware systems
Entertainment	• Interactive movies, computer games
Search	• Applications for finding and managing photos

Table 1.1: Various applications of facial recognition systems [210]

Human identification using fingerprint or palmprint is a widely used technology. This technology is based on the skin appearance which represents the outer tissue of the human body. Other applications of skin appearance in human identification include recent technologies that recognize the pattern of blood vessels in the palm and the finger. Figure 1.1 shows that the morphology of fine wrinkles tends to vary between body regions [78]. Thus, skin appearance has been a subject of great attention in various fields of technology and science.

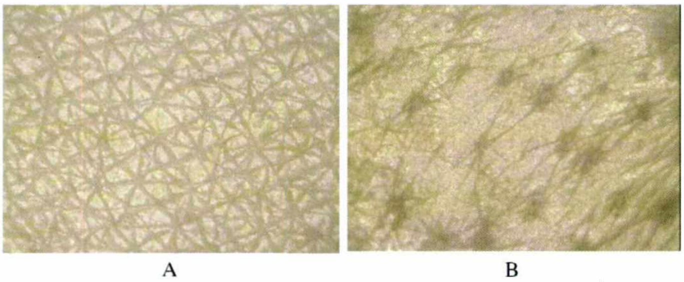


Figure 1.1: Fine wrinkles on (A) cheek and (B) forearm of the same person [78]

Figure 1.2 lists samples of skin appearance applications in four different research fields. There have been a considerable number of research studies utilizing human skin texture for topical drug efficacy testing for the pharmaceutical industry, automatic diagnosis for dermatology, facial animation, facial image synthesis and fingerprint identification. However, there have been only a few studies reported in the literature on investigating skin texture for face recognition, which is the topic of this study.

Section 1.1 and Section 1.2 will discuss the research motivations and aims respectively. The organization of the thesis will be presented in Section 1.3. Finally, Section 1.4 and Section 1.5 respectively will present the major contributions of the present research and the publications that have resulted from thesis so far.

1.1 Research Motivations

The motivations of this research can be summarized in the following points:

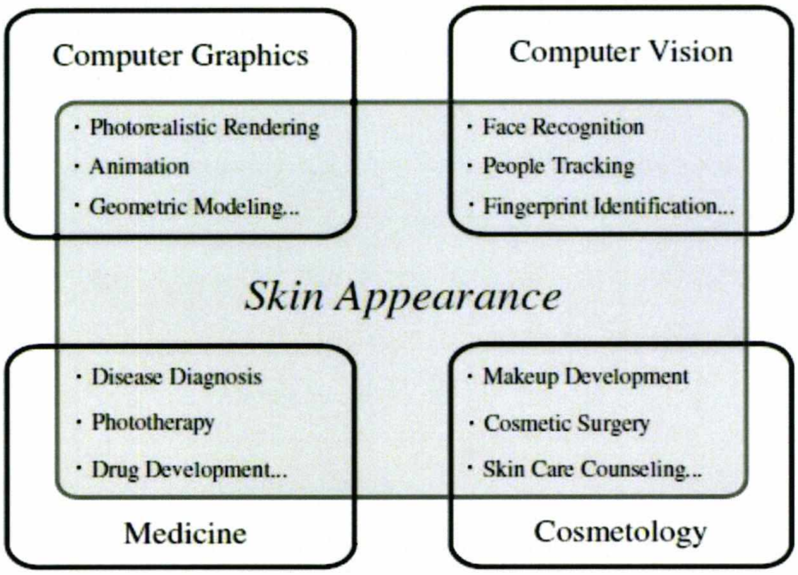


Figure 1.2: Examples of skin appearance applications in four research fields [78]

1. The face images in the conventional face recognition algorithms are scaled to 30-60 pixels between the centers of eyes [121]. Therefore, important information embedded in the micro-structures of skin texture may be lost because only a global appearance of the faces can be seen, while subtle details are lost.

2. The face images used for skin texture methods should be high-resolution images. According to Stan and Jain [121] the inter-eye distance should be at least 90 pixels to obtain reasonable performance. However, recent advancement of high quality digital cameras and web cams makes high-resolution face analysis possible and consequently it has catalyzed the advance of skin texture technology [121].

3. According to Tan and Triggs [204], since the problem of face recognition is a complex task, it is often that no single class of features is rich enough to capture all of the available facial information. Thus, skin details information could be

used to improve the performance of face recognition systems.

4. Sometimes only partial images are available for identification as in forensic or surveillance scenario so skin information may be particularly important in theses circumstances.

1.2 Research Objectives

The objective of this work is to investigate the ability of extracting useful biometric information from facial skin texture features and then to explore different approaches to advance and improve the study of face biometrics using these features. Thus the aims can be formulated in finding answers to the following questions:

1. How can one segment the facial skin areas?

2. Do we need to detect any landmarks in the face in order to fix the positions of skin areas?

3. How can we ideally extract the texture features from these areas?

4. Do skin areas contain useful biometric information?

5. If yes, is it long term or short term biometric trait?

6. Are there any parameters which could be tuned to optimize the recognition accuracy?

7. How can we automatically handle the noise in skin areas such as hair covering forehead region, facial skin covering chin region or sometimes dark sun glass covering parts of cheeks?

8. Is there a significant difference between the amount of biometric information in the different regions of the face?

9. If the answer of question Q8 is yes, what is the region which has the most significant biometric information?

10. How can we integrate this biometric trait with the conventional face recognition system?

11. How can one build a face recognition system that depends only on skin data? How can this system cope with the noise in skin areas? Is it possible to adaptively treats with the noise? What is the accuracy of this system?

12. Can we exploit this biometric trait in scenarios where only a partial image of the face is available for classification.

13. Can we compare the performance of the developed systems with the published works on same databases and using the same protocols?

1.3 Organization of the Thesis

Figure 1.3 illustrates the outline of the thesis. It presents the title of every chapter in the thesis and the corresponding objectives that the chapter addresses. More details are below. **Chapter 1 (Introduction):** This chapter introduces the thesis by presenting the research motivations, its aims, the outline of the thesis, the research contributions and the publications related to this research.

Chapter 2 (Texture Analysis: Literature Review): This chapter discusses the canonical problems that are related to texture analysis. It also provides an overview of

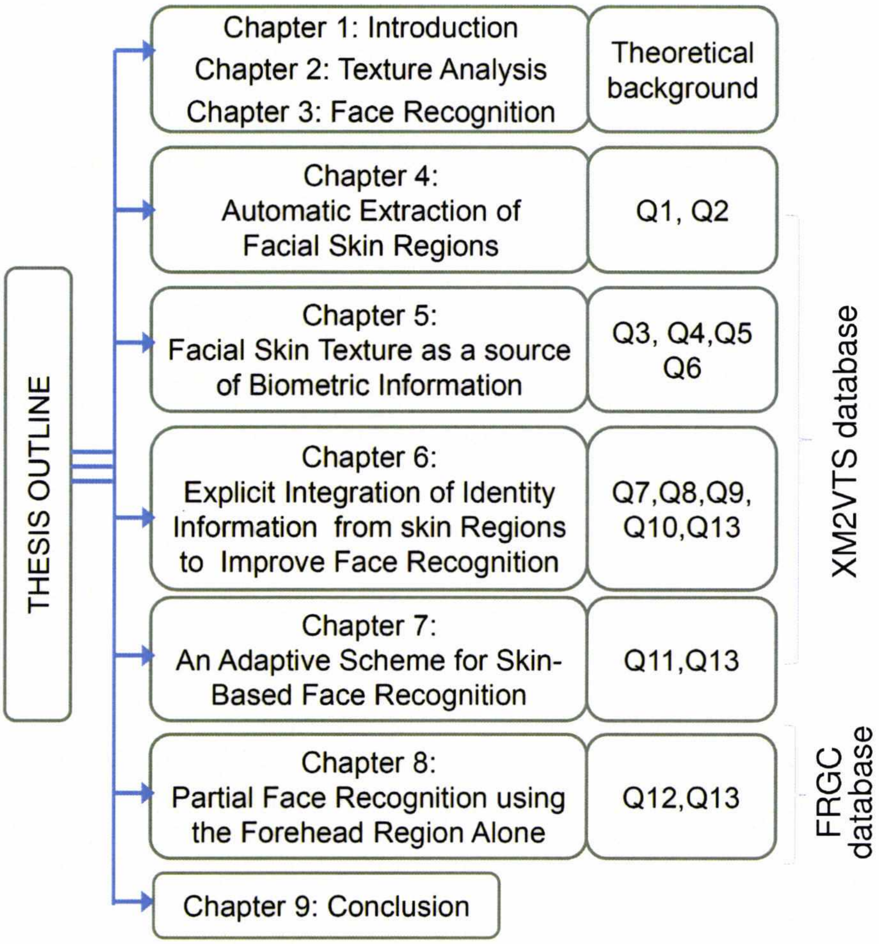

THESIS OUTLINE
Chapter 1: Introduction
Chapter 2: Texture Analysis
Chapter 3: Face Recognition
Theoretical background
Chapter 4: Automatic Extraction of Facial Skin Regions
Q1, Q2
Chapter 5: Facial Skin Texture as a source of Biometric Information
Q3, Q4,Q5 Q6
Chapter 6: Explicit Integration of Identity Information from skin Regions to Improve Face Recognition
Q7,Q8,Q9, Q10,Q13
Chapter 7: An Adaptive Scheme for Skin-Based Face Recognition
Q11,Q13
Chapter 8: Partial Face Recognition using the Forehead Region Alone
Q12,Q13
Chapter 9: Conclusion
XM2VTS database
FRGC database

Figure 1.3: The outline of the thesis (related to questions in Section 1.2)

the most recent approaches that are used to capture the characteristics of a particular texture (the Gabor filter approaches and local binary pattern), the most important and recent classification techniques are reviewed and finally the different information fusion schemes are presented.

Chapter 3 (Face Recognition: State of the art): This chapter discusses the current challenges of face recognition systems, it also describes the structure of a generic face recognition systems. The face detection techniques and some of the basic pre-processing stages are presented. The existing holistic and analytic face recognition approaches are reviewed. Finally, frequently used databases of face images and performance evaluation methods are presented.

Chapter 4 (Automatic Extraction of Facial Skin Regions): This chapter presents two automatic methods for localizing eyes and mouth centers. These two landmarks are used to segment four facial skin regions. The proposed algorithms are evaluated using the XM2VTS database.

Chapter 5 (Facial Skin Texture as a Source of Biometrics Information): This chapter investigates to know the best way for extracting features from the facial skin areas so it starts with the segmentation of usable forehead regions from facial images and their initial pre-processing to provide useful identity information. The method uses a manual annotation scheme to detect the forehead regions which do not contain noise and it investigates the effects of changing the parameters of the Gabor filter-bank and LBP approaches for face feature extraction. It also provides the results of the proposed method for the task of biometric person identification and verification.

Chapter 6 (Explicit Integration of Identity Information from Skin Regions

to Improve Face Recognition): The main objective of this chapter is to combine the conventional face recognition systems with skin texture features from forehead region. This chapter also reports on the development of a novel Skin/Non-Skin classifier based on Support Vector Machines (SVM), in order to automatically exclude non-skin areas in facial images.

Chapter 7 (An Adaptive Scheme for Skin-Based Face Recognition): This chapter presents a fully automated and novel adaptive method for face recognition based on twenty extracted skin patches. The global distance between the probe and gallery image is a weighted sum of the local patch distances adapted in such a way to include the contribution of each usable skin patch.

Chapter 8 (Partial Face Recognition using the Forehead Region Alone): This chapter presents a scheme for face recognition in application scenarios where only a partial image of the forehead region is available for classification.

Chapter 9 (Conclusion and Future work): The chapter provides conclusions and suggestions for future work.

1.4 Contributions

We have developed methods to exploit facial skin texture for human recognition in three ways: (i) The use of facial skin texture as an additional biometric information source to improving the conventional face recognition systems. (ii) An adaptive algorithm to study the information available from all facial regions and cope with the regions that may have partial or complete hair coverage. (iii) A partial face recognition algorithm. The major contributions of this dissertation are summarized as follows:

1. An automated and novel method for eye detection is presented and compared with other published results.

2. A fully automated and accurate segmentation of four facial skin regions in the facial images namely Forehead, Right cheek, Left cheek and Chin using eye and lip centers as landmarks.

3. A novel patch-based technique for first exploring the skin texture information of pure-skin forehead and then utilizing it for biometric verification and identification.

4. A new Skin/Non-Skin classifier to automatically eliminate areas that may be obscured by hair, facial hair or spectacles.

5. A novel combination algorithm that fuses the holistic face recognition approach, and the forehead skin texture matching algorithm has been proposed and implemented. The resulting integrated face matching algorithm delivers better performance than either of these two matching algorithms alone.

6. To verify the feasibility of using facial skin texture alone to perform face recognition, a novel scheme that completely depends on the skin information is proposed and implemented. It uses an adaptive strategy for combining pairwise patch distances between individuals with different numbers of usable skin patches. Experiments using the XM2VTS database show that the proposed method achieves comparable performance to the best of the published results.

7. A new partial face recognition scheme is presented in scenarios where only a partial image of the forehead is available for classification. A scheme for

segmenting the head region in such images is presented. The forehead region is then extracted without the use of facial landmarks such as eye locations. The system is tested on FRGC data and presents promising recognition results for a very challenging problem.

1.5 Publication Arising from the Thesis

This research has resulted in the publications listed below

- **List of Journal publication**

1. G.F.Al-Qarni and F.Deravi, "Facial Skin Texture as a Source of Biometric Information", *International Journal of Signal Processing, Image Processing and Pattern Recognition (IJSIP)*, Vol.5, No.4, 2012.

- **List of Conference publication**

2. G.F.Al-Qarni and F.Deravi, "Explicit Integration of Identity Information from Skin Regions to Improve Face Recognition", In: A. Campilho and M. Kamel (Eds.), *9th International Conference on Image Analysis and Recognition (ICIAR)*, Part II, LNCS, vol. 7325, pp. 30-37, Springer, Heidelberg, June, 2012.

3. G.F.Al-Qarni and F.Deravi, "Partial Face Recognition using the Forehead Region Alone", 5th International Workshop on Computational Forensics (*IWCF'12*), In Conjunction with the International Conference on Pattern Recognition, *ICPR*, Tsukuba, Japan, 811 November, 2012.

4. G.F.Al-Qarni and F.Deravi, "An Adaptive Scheme for Skin-based Face Recognition", *IEEE International Conference on Image Processing, ICIP'13*, Melbourne, Australia.

5. G.F.Al-Qarni and F.Deravi, "Forehead Skin Texture as a Source of Biometric Information", *4th Saudi International Conference (SIC)*, University of Manchester, UK, 3031 July, 2010.

6. G.F.Al-Qarni and F.Deravi, "Exploitation of Pure Skin Regions for Face Recognition", *School Research Conference 2012*, School of Engineering and Digital Arts, University of Kent, Canterbury, UK, 13 January, 2012.

Chapter 2

Texture Analysis and Classification

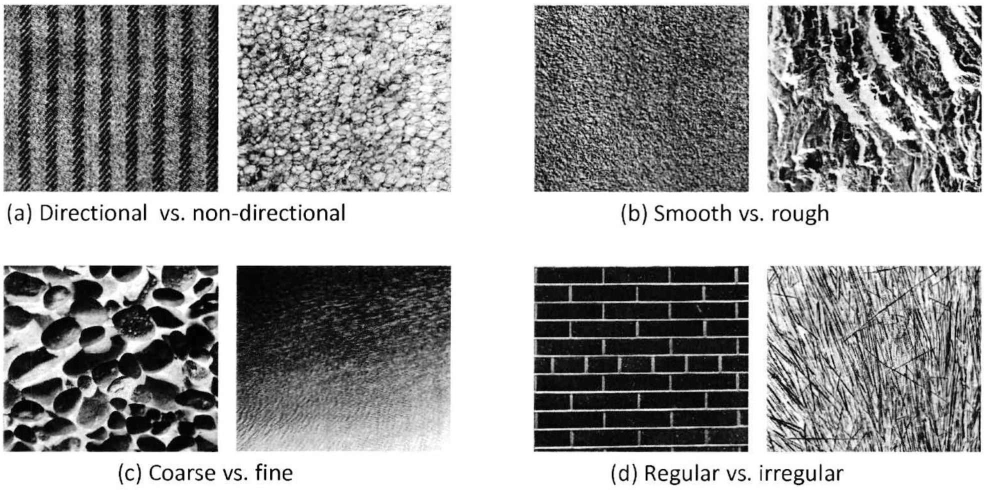
2.1 Introduction

"In biological vision, texture is an important cue allowing humans to discriminate objects. This is because the brain is able to decipher variations in data at scales smaller than those of the viewed objects."[161]

There is no precise definition of the texture, however, it could be defined as *the repetition of a pattern or patterns over a region in an image*[157]. There are two categories of texture image, *stationary* and *non-stationary.* A stationary texture image is an image which contains a single type of texture, i.e. the whole image is filled up by the same texture so its local statistical properties are the same everywhere in it [161]. A non-stationary texture image is a texture image which contains more than one type of texture in it [161]. Texture also may be perceived as being regular or irregular, smooth or rough, coarse or fine, directional or non-directional, etc. Figure 2.1 shows examples of these categories.

Texture analysis is essential in several problems in image processing and machine vision. It is a well-developed technology and has been used for the analysis of different kinds of images. These range from microscopic images in the biomedical research area to satellite images of the earth's surface.

This chapter discusses the canonical problems that are related to texture analysis. It also provides an overview of the most recent approaches that are used to capture the characteristics of a particular texture, the so called *texture features*. We also highlight the most important and recent classification techniques and the last subsection will be devoted for the different information fusion schemes.



(a) Directional vs. non-directional

(b) Smooth vs. rough

(c) Coarse vs. fine

(d) Regular vs. irregular

Figure 2.1: Perception of textures

2.2 Texture Analysis-related Problems

Texture analysis is an active research area in the field of image processing and machine vision. The various texture analysis techniques for extracting texture features can be applied in five broad categories of problems: texture segmentation, texture classification, texture synthesis, shape from texture, and surface defect detection.

- **Texture segmentation** Texture segmentation is the process of separating an image into regions that have homogeneous properties with respect to texture. It has many applications in medical imaging such as diagnosis, locating tumors and other pathologies, studying of anatomical structure, and measuring tissue volumes. Other important applications include locating objects in satellite or aerial imagery (forests, roads, etc.), traffic control systems, and segmentation of textured regions in document analysis.

  The texture segmentation approaches can typically be divided into two categories, region-based and edge-based techniques [212]. In the Edge-based approach, boundaries of adjacent regions are detected where there are differences in their texture so this technique does not need to know the number of textured areas in an entire image, whereas texture similarities between adjacent regions are used as the basis for region-based texture segmentation. There are some reported research works which take advantage of the complementary nature of both these approaches [159].

- **Texture classification**: The goal of texture classification is to assign an unknown sample image to one of c known texture classes. It can be implemented in a large variety of real world applications that allow the target subjects to be viewed as a specific type of texture. These applications include wood species recognition [208], rock classification [119], fabric classification [13] and etc.

  This process involves two phases: a training or learning phase and a testing or recognition phase. In the learning phase, a set of grey level images are taken from each texture class. Then, the textural properties of the training images are captured with the chosen texture analysis method, which yields a set of textural features for each image. Finally a model of each texture class (or signature) is built. In the recognition phase, any input test image is first described with the same texture analysis method. This feature vector is compared with all of the prototype vectors obtained in the learning stage via a classification algorithm, and the sample is assigned to the class with the best match.

- **Texture defect detection**: The goal of texture defect detection is to decide whether a surface texture is as it is expected to be or contains faults (texture abnormalities). This problem has got applications on a variety of surfaces, e.g. wood, steel, ceramics, etc. Therefore it is highly demanded by industry in order to replace the subjective and repetitive process of manual inspection [7].

- **Shape from Texture**: There exist many properties in images that allow to determine the shape of an object in 3D scene, for example variations in intensity on the surface of objects, the relative positions and orientations of edges and corners, and shadowing effects. Texture is another feature which can be used to estimate the relative orientation of a surface [32].

- **Texture synthesis**: The objective of the texture synthesis is to generate descriptions to build a model of image texture, which can then be used for producing the texture [79]. It has more applications in computer graphics.

2.3 Texture Analysis

Texture analysis has been traditionally performed by extracting descriptors from greyscale texture images, and therefore, colour information is often ignored in this analysis [161]. Numerous approaches have been proposed for texture description. According to Haralick [64], such approaches can be categorized into two main groups: Statistical and Structural methods. Tuceryan and Jain [212] have divided texture description methods into four main classes: geometrical, model-based, statistical and signal processing methods. Figure 2.2 illustrates the categorization of texture description methods according to [212]. The first two categories are briefly reviewed in this section, though the central focus is on the latter two approaches because they are of interest in this thesis.

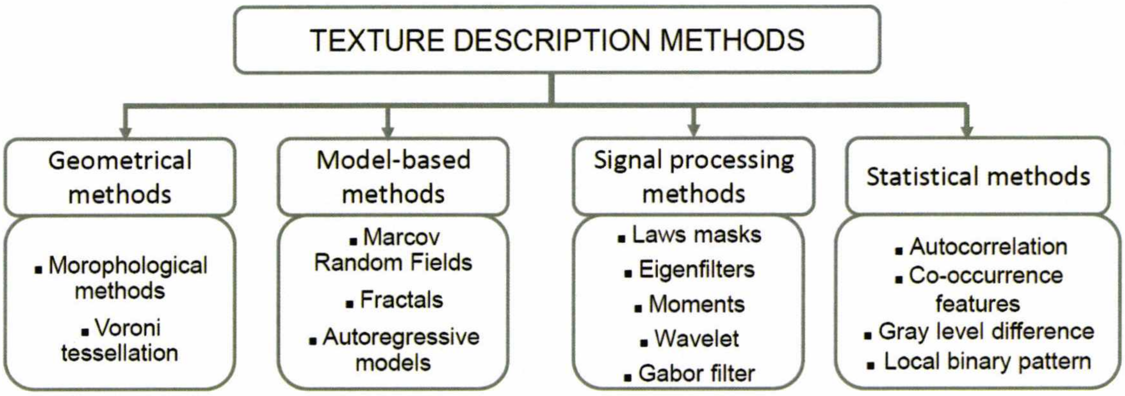


Figure 2.2: Main categories of texture descriptors according to [212]

Nowadays there is a more recent family of texture descriptors, that is called "SIFT (Scale-Invariant Feature Transform)-related" descriptors. We can find, among the methods belonging to this group, SIFT [132], GLOH (Gradient Location and Orientation Histogram)[143] and SURF (Speeded Up Robust Features)[11]. These are based on detecting the key points in the image and then describing the local image region around each key point.

2.3.1 Geometrical methods

Geometrical methods consider texture as being composed of "texture elements" or primitives so they try to describe these primitives and the rules governing their spatial arrangement. In this technique, texture primitives are first extracted from the image using techniques such as mathematical morphological tools ([189], [30]) or using edge detection algorithms such as a Laplacian-of-Gaussian or difference-of-Gaussian filter. For the analysis of the texture, two major approaches can be used [212]. In the first approach, statistical properties of the primitives are computed and used as texture features. In the second approach, the placement rule that describes the texture elements are extracted and used as features [53].

2.3.2 Model-based methods

Model-based texture methods model the mathematical process that generated the texture. The model parameters are estimated and then used as features. Such models can be used for feature synthesis. There are currently three main model-based methods, Markov Random Fields [48], fractals ([160]), and the autoregressive models presented by Mao and Jain [139]. The image models are detailed by Chellappa and Jain in [27].

2.3.3 Statistical methods

Statistical methods analyze the spatial distribution of gray values, by measuring the local features at each pixel in the image, and deriving a set of statistics from the distributions of these features. Based on the number of pixels defining the local feature, these methods can be classified into first-order (one pixel), second-order (two pixels) and higher-order (three or more pixels) statistics. In contrast to the first-order statistics, second- and higher-order statistics do not ignore the spatial interaction between image pixels. Instead, they estimate properties of two or more pixel values occurring at specific locations relative to each other.

The most widely used and popular statistical methods are co-occurrence features [65], autocorrelation function [95], and gray level differences [223]. Inspired by Gray level differences method, a variety of modifications has been proposed by Ojala *et al.* for example signed differences [153] and the LBP (Local Binary Pattern) operator [151], which combine structural and statistical approaches to texture analysis.

Auto Correlation

The autocorrelation function [95] can be used to analyze the regularity and estimation of the fineness/coarseness of the texture. It measures the linear spatial relationship of the structural elements (primitives). Formally, the autocorrelation function of an image $I(x, y)$ is given by Equation 2.1:

$$\rho(x, y) = \frac{\sum_{u=0}^{N} \sum_{v=0}^{N} I(u, v) I(u + x, v + y)}{\sum_{u=0}^{N} \sum_{v=0}^{N} I^2(u, v)} \tag{2.1}$$

Where x, y are the positional differences in the u, v directions. For a coarse texture (with large primitives), the function value will decrease slowly with increasing distance whereas it will decay rapidly if texture is fine. For those textures with a very repetitive nature, the autocorrelation function will become semi-periodic with clear peaks and valleys.

Co-Occurrence features

In 1963 Julesz [94] showed the importance of using second order statistics for texture discrimination experiments. Different tools have been developed to exploit this technique. The original investigation into the grey-level co-occurrence (GLC) features was pioneered by (Haralick,1979) [64]. Typically, these features are extracted in two stages.

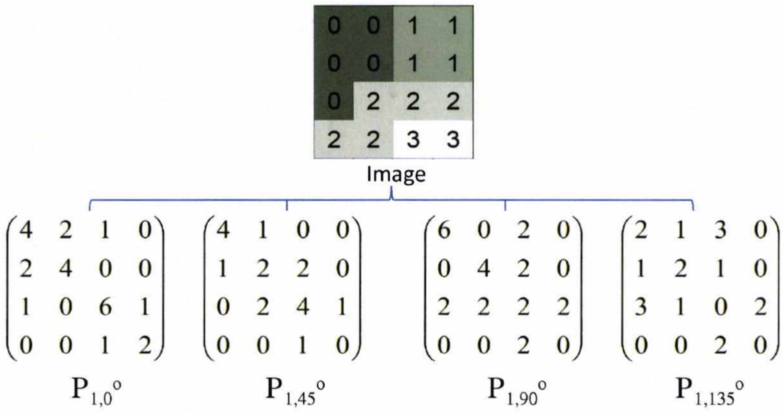
First, a set of gray-level co-occurrence matrices (GLCM) is derived. A GLC matrix represents the joint probability occurrence of pairs of grey levels of pixels separated by a certain distance d and lying in a certain direction θ in the image. Formally, given the image $f(x, y)$ with a set of G discrete intensity levels, the matrix $P_{d,\,\theta}(i, j)$ is defined such that its (i, j)th entry is equal to the number of times that:

$$\begin{aligned} f(x_1, y_1) = i \quad \text{and} \quad f(x_2, y_2) = j, \\ \text{where} \quad (x_2, y_2) = (x_1, y_1) + (d\ cos\theta, d\ sin\theta) \end{aligned} \tag{2.2}$$

This yields, for each distance d and orientation θ, a square matrix of dimension equal to the number of intensity levels in the image. Thus, the size of the matrix is independent of the size of the image, but depends on the number of grey levels in the image. For instance, a GLCM for a dynamic range of 8 bits has 256 × 256 entries.

Regarding the distance, often only the distances $d = 1$ and 2 pixels are considered due to the intensive nature of computations involved. There are two forms of co-occurrence matrix exist, one symmetric where pairs separated by $\mp(d)$ for a given direction θ are counted, and in the non symmetric only pairs separated by distance d are counted. The later case allows us to infer four matrices for a given distance d with angles $\theta = 0^o, 45^o, 90^o$ and 135^o as suggested in [64].

Example of building cooccurrence matrix is illustrated in Figure 2.3, for $d = 1$.



$$\begin{pmatrix} 4 & 2 & 1 & 0 \\ 2 & 4 & 0 & 0 \\ 1 & 0 & 6 & 1 \\ 0 & 0 & 1 & 2 \end{pmatrix} \quad \begin{pmatrix} 4 & 1 & 0 & 0 \\ 1 & 2 & 2 & 0 \\ 0 & 2 & 4 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \quad \begin{pmatrix} 6 & 0 & 2 & 0 \\ 0 & 4 & 2 & 0 \\ 2 & 2 & 2 & 2 \\ 0 & 0 & 2 & 0 \end{pmatrix} \quad \begin{pmatrix} 2 & 1 & 3 & 0 \\ 1 & 2 & 1 & 0 \\ 3 & 1 & 0 & 2 \\ 0 & 0 & 2 & 0 \end{pmatrix}$$

$$P_{1,0^o} \quad P_{1,45^o} \quad P_{1,90^o} \quad P_{1,135^o}$$

Figure 2.3: Example of Symmetric GLCM for d and different θ

Several texture descriptors are usually derived from the co-occurrence matrix although the direct use of co-occurrence matrix has also been proposed in [146]. The early paper by Haralik suggested fourteen such descriptors which can be categorized into four groups [58]:

- visual textural characteristics: angular second moment, contrast
- statistical measures: sum average, variance, difference variance
- features that are based on information theory: entropy
- Correlation-based measures

2.3.4 Signal-processing methods

Most Signal processing-based techniques try to extract texture features by submitting the image to a linear transform, filter, or filter bank, followed by some energy measurement [173]. They can be divided into spatial domain, frequency domain, and joint spatial/spatial-frequency domain methods [144].

Spatial Domain Filters

A texture can be considered as a mixture of patterns, therefore the early attempts to discriminate different textures concentrated on measuring edge strength and edge frequency. In order to detect edges, lines, dots, etc, in the spatial domain, the image is usually filtered by a gradient filter such as the Robert or Laplacian operators [57].

Moreover, Laws [116] proposed a novel approach to calculate texture energy measures from a greyscale image. The measures are derived from three simple vectors of length 3, which represent the one-dimensional operations of center-weighted local averaging L3=$[\,1 \quad 2 \quad 1\,]$, edge detection E3=$[\,-1 \quad 0 \quad 1\,]$, and spot detection S3=$[\,-1 \quad 2 \quad -1\,]$.

With the convolution pairs of these three vectors with each other and themselves, we obtained the following 1×5 masks, the initial letters are as before with the addition of Ripple detection R5 and Wave detection W5:

$$\mathrm{L5} = \mathrm{L3} * \mathrm{L3} \Rightarrow \;\; \mathrm{L5} = [\,1 \quad 4 \quad 6 \quad 4 \quad 1\,] \tag{2.3}$$

$$\mathrm{E5} = \mathrm{L3} * \mathrm{E3} \Rightarrow \;\; \mathrm{E5} = [\,-1 \quad -2 \;\; 0 \;\; 2 \;\; 1\,] \tag{2.4}$$

$$\mathrm{S5} = \text{-}\mathrm{L3} * \mathrm{E3} \Rightarrow \;\; \mathrm{S5} = [\,-1 \quad 0 \;\; 2 \;\; 0 \quad -1\,] \tag{2.5}$$

$$\text{R5} = \text{-E3} * \text{S3} \Rightarrow \text{R5} = [\,1 \quad -4 \quad 6 \quad -4 \quad 1\,] \tag{2.6}$$

$$\text{W5} = \text{S3} * \text{S3} \Rightarrow \quad \text{W5} = [\,-1 \quad 2 \quad 0 \quad -2 \quad 1\,] \tag{2.7}$$

From these five 1D operators, a total of 25 2D operators can be generated by matrix multiplication of these vectors, considering the first term as a column vector and the second term as row vector, results in 5 × 5 matrix known as Laws Masks. A feature vector that can be used for texture description is derived by applying a convolving the texture image with 5 × 5 masks and calculating energy statistics such as mean, absolute mean and standard deviation.

It is also possible to implement small size spatial domain filters, known as Local Linear Transform (LLT), such as discrete cosine (DCT), discrete sine (DST), or discrete Hadamard (DHT) filters instead of Laws filters for texture analysis and classification [214]. Ade [2] suggested Eigenfilters in the spatial domain, a set of masks obtained from the Karhunen-Loeve transform (KLT) [92] of local image patches, for texture representation. These local linear transforms i.e. KLT, DST, DCT, and DHT are compared for texture classification and found KLT to be the optimum LLT [215].

Another class of spatial filters are Moments [117], the $(p+q)^{th}$ moments over an image region R are computed as follows:

$$m_{pq} = \sum_{(x,y)\in R} x^p \; y^q \; f(x,y) \tag{2.8}$$

These moments are computed around each pixel in the image which is equivalent to filtering the image by a combination of spatial masks. The resulting filtered images are then used as texture features. Tuceryan [211] used $p+q \leq 2$ moment-based features for texture segmentation. The invariance property of moments to scale, translation and rotation is discussed in [57].

Frequency domain analysis

The Fourier transform $F(u,v)$ of an image $f(x,y)$ reveals its frequency and orientation distribution. It can be formed as [57]:

$$F(u,v) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)\ exp\ (-j2\pi(ux+vy)\ )\ dxdy \tag{2.9}$$

The images are practically in digital form so discrete Fourier transform is utilized:

$$F(u,v) = \frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)\ exp\ (-j2\pi(\frac{ux}{M}+\frac{vy}{N})\ ) \tag{2.10}$$

where M is the image width and N is its height. The Fourier power spectrum is given by: $|F|^2 = F\bar{F}$, where $\bar{F}$ denotes complex conjugate of F. In texture analysis, the power spectrum can be used for extracting features in several methods such as *Ring*, *Wedge* or their combined filters as shown in Figure 2.4.

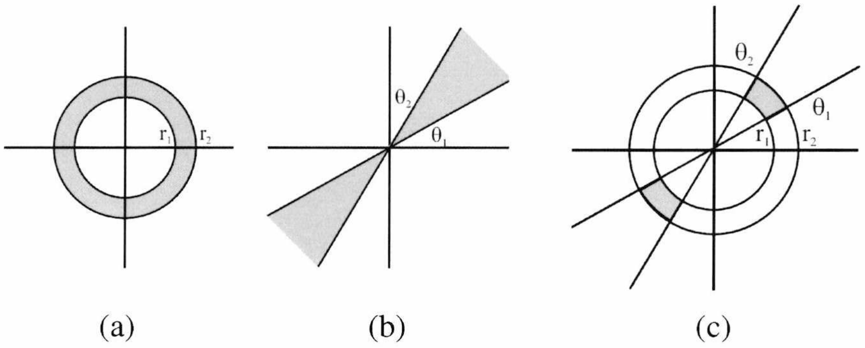


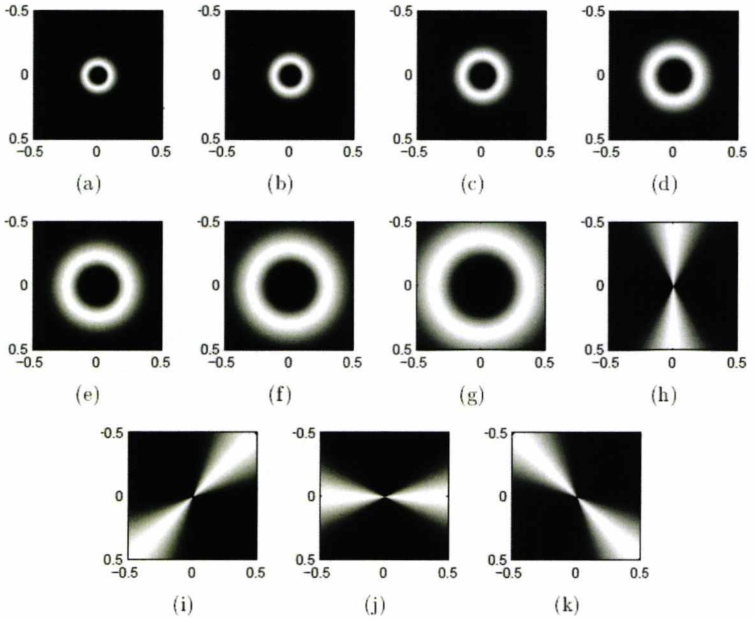
Figure 2.4: An example of (a) a ring (b) a wedge (c) a combined ring-wedge filter

The features based on Ring and Wedge filters are respectively of the forms [223]:

$$f_{r1,r2} = \sum_{\substack{{r_1}^2<(u^2+v^2)<{r_2}^2 \\ 0 \le u \le M-1,\ 0 \le v \le N-1}} |F(u,v)|^2 \tag{2.11}$$

$$f_{\theta1,\theta2} = \sum_{\substack{{\theta_1}^2<\tan^{-1}(\frac{v}{u})<{\theta_2}^2 \\ 0\le u\le M-1,\ 0 \le v \le N-1}} |F(u,v)|^2 \tag{2.12}$$

It is noted that the radial distribution of the spectrum values is sensitive to texture coarseness in f and the angular distribution of the spectrum is sensitive to the directionality of texture in f. Thus, Coggins and Jain [33] employed seven dyadically spaced ring filters and four wedge-shaped orientation filters, as shown in Figure 2.5, for evaluating the coarseness and the directionality of the texture image.



(a) (b) (c) (d)

(e) (f) (g) (h)

(i) (j) (k)

Figure 2.5: Amplitude responses of the ring and wedge filters [33]

Spatial/Spatial-frequency methods

Psychophysiological findings indicated that the human (and other mammals) visual system analyzes the textured images by decomposing the image into various spatial scales and orientations [212]. Motivated by this biological theory, multiresolution analysis, called multi-scale/multi-directional (MSMD) methods were developed and applied for different texture problems such as texture segmentation, classification and synthesis applications [175], [195].

In the MSMD methods, texture description, is done by first filtering the image with a bank of filters, each filter having a specific scale and orientation and texture features are then extracted. Most of these methods are based on either analysis, or Gabor filters.

Wavelet

Wavelets perform a decomposition of a signal as a sum of local bases with finite support and localized at different scales. They are defined in reference to a "mother function" $\psi(t)$ of some real variable t. The mother function is characterized for being a bounded function with zero average:

$$\int_{-\infty}^{\infty} \psi(t)dt = 0 \tag{2.13}$$

This function can be dilated with a scale parameter s and translated by u in order to generate a whole family of wavelets:

$$\psi_{u,s}(t) = \frac{1}{\sqrt{(s)}}\,\psi(\frac{t-u}{s}) \tag{2.14}$$

The wavelet transform of f at scale s and position u is computed by:

$$W_f(u,s) = \int_{-\infty}^{\infty} f(t)\,\frac{1}{\sqrt{s}}\,\bar{\psi}(\frac{t-u}{s})dt \tag{2.15}$$

There are several ways to implement a multiscale wavelet technique. However, the formal and unified approach which was introduced by Mallat [136] is a well established and popular platform. Mallat suggested to use the Dyadic Wavelet Transform. Other popular wavelet transform techniques, that have been applied to texture analysis, include the contourlet transform [120], steerable pyramidal transform [98] and curvelet transform [188].

2.4 Gabor wavelet-based features for texture analysis

Gabor filtering has been successfully used for feature extraction in many image processing and machine vision applications. In this section, the Gabor filter responses are analyzed in one- and two- dimensional spaces. First, the formulation of the Gabor elementary functions (GEF) is covered in the one-dimensional case. Then, the formulation is extended for two-dimensional images with the focus on filter normalization, the advantages of Gabor filtering, filter parameters selection, and the different ways for extracting texture features are discussed. Finally, a literature review of the Gabor-based algorithms and the most important applications are presented.

2.4.1 One-dimensional Gabor filter

For the past few decades, there have been two alternative ways to represent one-dimensional signals. The first one describes signals as a function of frequency and the second as a function of time. The description form can be converted from one to another using the Fourier or inverse Fourier transforms, i.e they carry the same information but in different visualization.

It is, however, hard to simultaneously characterize the two basic properties of such signals: what kind of events a signal contains? and when exactly do they occur? The only well known is uncertainty principle which says: the product of the frequency bandwidth multiplied with the time duration of a signal cannot be less than a certain minimum value. Formally this is expressed as:

$$\Delta t \, \Delta f \geq 1/2 \tag{2.16}$$

where Δt is the duration of the signal that is considered and Δf is its bandwidth in the frequency domain. There are several other forms of the Fourier transform, and thus, the uncertainty value is sometimes replaced by $1/4\pi$ but having the same interpretation [96].

Gabor (1946) [54] derived the function that minimizes this uncertainty. i.e., turns the inequality into an equality. He demonstrated that the function is a complex sinusoidal wave modulated by a Gaussian probability function:

$$\psi(t) = e^{-\alpha^2 \, (t-t_0)^2} \, e^{j \, 2\pi f_0 t + \phi}, \tag{2.17}$$

where α is the sharpness (time duration/bandwidth) of the Gaussian, t_0 denotes its centroid in time domain, f_0 is the frequency of the harmonic oscillation, and ϕ denotes its phase shift. The 1-D Fourier transform of the Gabor elementary function is given by Equation 2.18:

$$\Psi(u) = \sqrt{\frac{\pi}{\alpha^2}}\; e^{-(\frac{\pi}{\alpha})^2 (f-f_0)^2}\; e^{-j\,2\pi t_0 (f-f_0)+\phi}. \tag{2.18}$$

Since an origin-centered filter is preferred for convolution, the time and phase shift can be removed ($t_0 = 0$, $\phi = 0$), so the Gabor elementary functions in Equation (2.18) can be defined in more compact form:

$$\psi(t) = e^{-\alpha^2\, t^2}\; e^{j\,2\pi f_0 t}. \tag{2.19}$$

To make the time duration of function $\psi(t)$ dependent on the central frequency f_0, an approach from multiresolution analysis [60] is employed to guarantee that the functions on different frequencies behave as scaled version of each other. This can be accomplished by defining a constant ratio:

$$\alpha = \frac{|f_0|}{\gamma} \tag{2.20}$$

From the Fourier transformed Gabor function in 2.18, we can see that the maximum response ($\sqrt{\pi/\alpha^2}$) occurs at $u = f_0$. Thus, its inverse can be used as a normalization factor. Consequently, the normalized 1-D Gabor filter function is defined as:

$$\psi(t) = \frac{|f_0|}{\gamma\sqrt{\pi}}\; e^{-(\frac{f_0}{\gamma})^2\, t^2}\; e^{j\,2\pi f_0 t} \tag{2.21}$$

The 1-D Gabor filter function has a Fourier transform

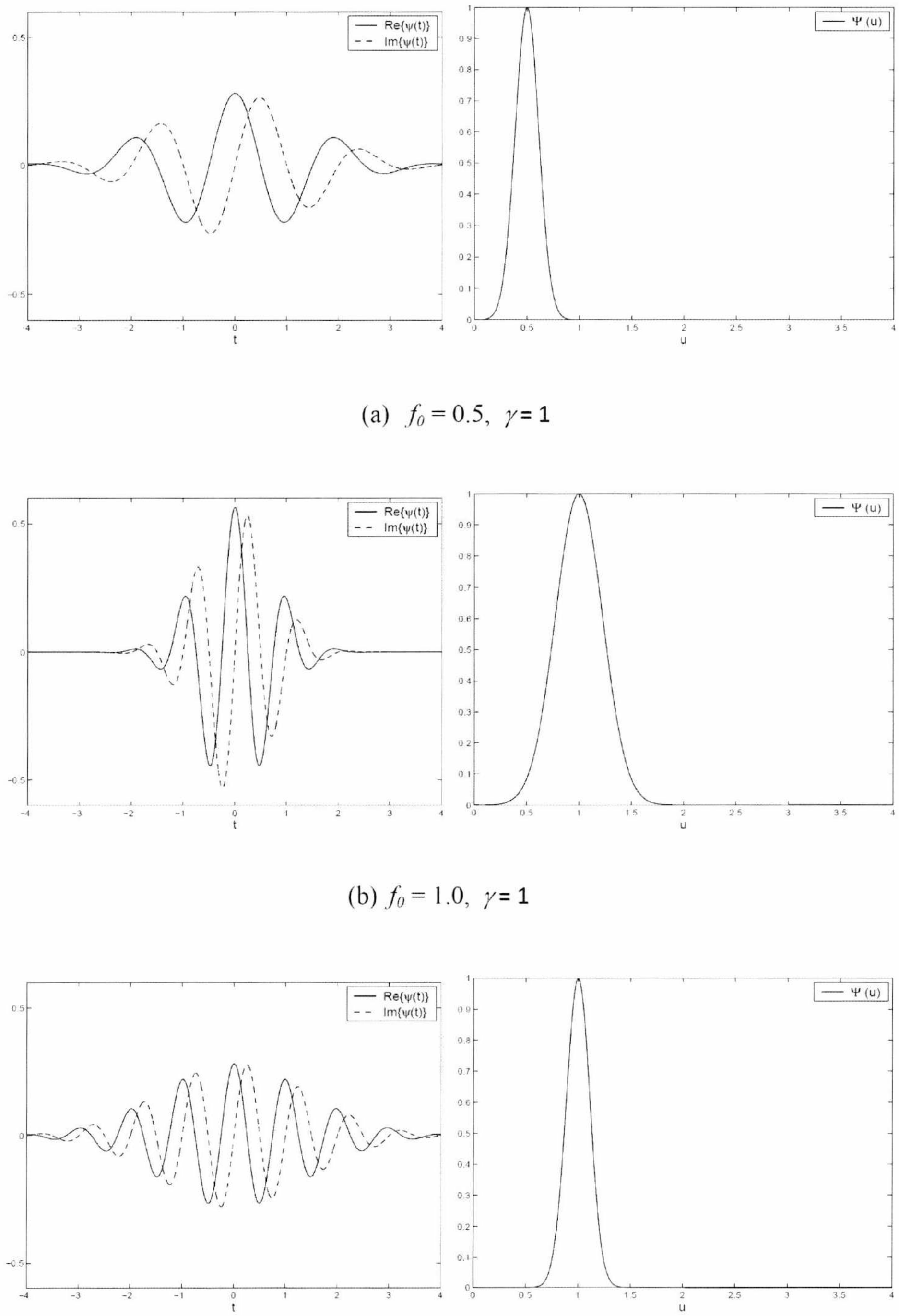
$$\Psi(u) = e^{-(\frac{\gamma\pi}{f_0})^2 \, (u-f_0)^2} \tag{2.22}$$

Figure 2.6 shows the Gabor filter plots in time and frequency domains for different values of f_0 and γ. The parameter γ can be used to adjust the effective width of the Gabor filter, by increasing γ the filter spreads in the time domain and shrinks in the frequency domain and vice versa (Figure 2.6 (a) and (c)) The center frequency f_0 can also change the width of the filter when γ is fixed (Figure 2.6 (a) and (b)).

2.4.2 Two-dimensional Gabor filter

The development and use of 2-D Gabor filters began from Granlund (1978), when he proposed the form of a general picture processing operator and addressed some fundamental properties, such as the octave spacing of the frequencies [59]. Daugman [42], [41] later showed a surprising equivalence between a structure based on the 2-D Gabor functions and the organization and the characteristics of the mammalian visual system. He also defined similar uncertainty measures as in Equation (2.16) for the 2-D case.

The 1-D Gabor filter in Equation (2.19) can be generalized to two dimensions by replacing time variable t with spatial coordinates (x, y) in the spatial domain and the frequency variable f is replaced by the frequency variables (u, v) in the frequency



(a) $f_0 = 0.5$, $\gamma = 1$

(b) $f_0 = 1.0$, $\gamma = 1$

(c) $f_0 = 1.0$, $\gamma = 2$

Figure 2.6: Gabor filter functions in time and frequency domains for different values of parameters f_0 and γ [97]

domain as follows:

$$\begin{aligned} \psi(x,y) &= e^{-(\alpha^2\acute{x}^2 + \beta^2\acute{y}^2)}\, e^{j2\pi f_0 \acute{x}} \\ \acute{x} &= x\cos\theta + y\sin\theta \\ \acute{y} &= -x\sin\theta + y\cos\theta \end{aligned} \tag{2.23}$$

As in the 1-D case, to guarantee that filters in different frequencies are scaled versions of each other, $\alpha = |f_0|/\gamma$ and $\beta = |f_0|/\eta$ are substituted and the corresponding 2-D Gabor filter can be defined as Equation (2.24)

$$\begin{aligned} \psi(x,y) &= \frac{f_0^2}{\pi\gamma\eta}\, e^{-(\frac{f_0^2}{\gamma^2}\acute{x}^2 + \frac{f_0^2}{\eta^2}\acute{y}^2)}\, e^{j2\pi f_0 \acute{x}} \\ \acute{x} &= x\cos\theta + y\sin\theta \\ \acute{y} &= -x\sin\theta + y\cos\theta \end{aligned} \tag{2.24}$$

where f_0 is the central frequency of the filter, θ is the rotation angle of the Gaussian major axis and plane wave (sinusoidal), γ is the sharpness along the major axis, and η is the sharpness along the minor axis. The normalized 2-D Gabor filter function in the frequency domain is given by Equation (2.25)

$$\begin{aligned} \Psi(u,v) &= e^{-\frac{\pi^2}{f_0^2}(\gamma^2(\acute{u}-f_0)^2 + \eta^2\acute{v}^2)} \\ \acute{u} &= u\cos\theta + v\sin\theta \\ \acute{v} &= -u\sin\theta + v\cos\theta \end{aligned} \tag{2.25}$$

The effects of the parameters, that are demonstrated in Figure 2.7 and Figure 2.8 can be summarized as follows:

1. Change the width of the filter by varying the center frequency f_0 (Figure 2.7).

2. Rotate filter by angle θ around origin of coordinates (Figure 2.8).

3. Compress (Stretch) filter in x direction by parameter γ (Figure 2.7(b) and Figure 2.8(a)).

4. Compress (Stretch) filter in y direction by parameter η (Figure 2.7(b) and Figure 2.8(a)).

A filter response for an image function $I(x, y)$ can be calculated at any location (x, y) with the convolution as shown in Equation (2.26)

$$r_I(x, y; f, \theta) = \psi(x, y; f, \theta) * I(x, y) \tag{2.26}$$

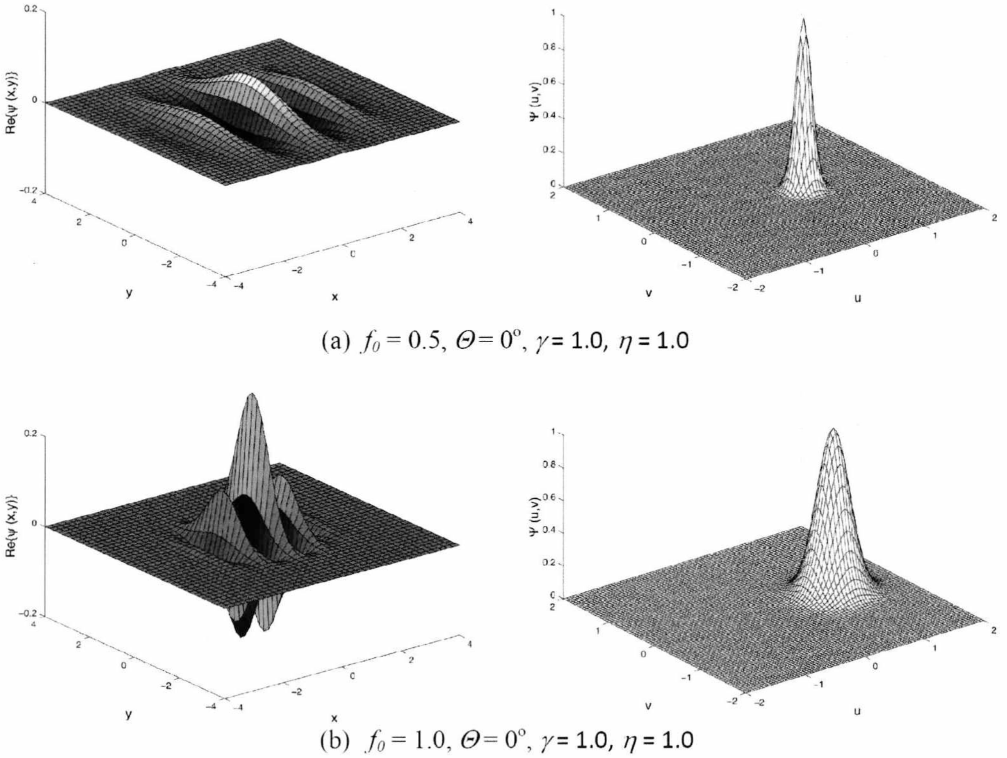
2.4.3 Filter bank and parameters selection

In practice, applications utilizing only a single filter are unusual [209] and it is necessary to apply several Gabor filters to an image. This is done by forming a so called "filter bank". A filter bank needs to be used because relationships between responses provide the basis for distinguishing objects in the classification stage.

The selection of filter bank parameters i.e. orientations and frequencies in Equation (2.26) is very important. It has been demonstrated in [111] that the orientations must be spaced uniformly as:

$$\theta_k = \frac{k2\pi}{n}, \text{for } k = \{0, \ldots, n-1\} \tag{2.27}$$

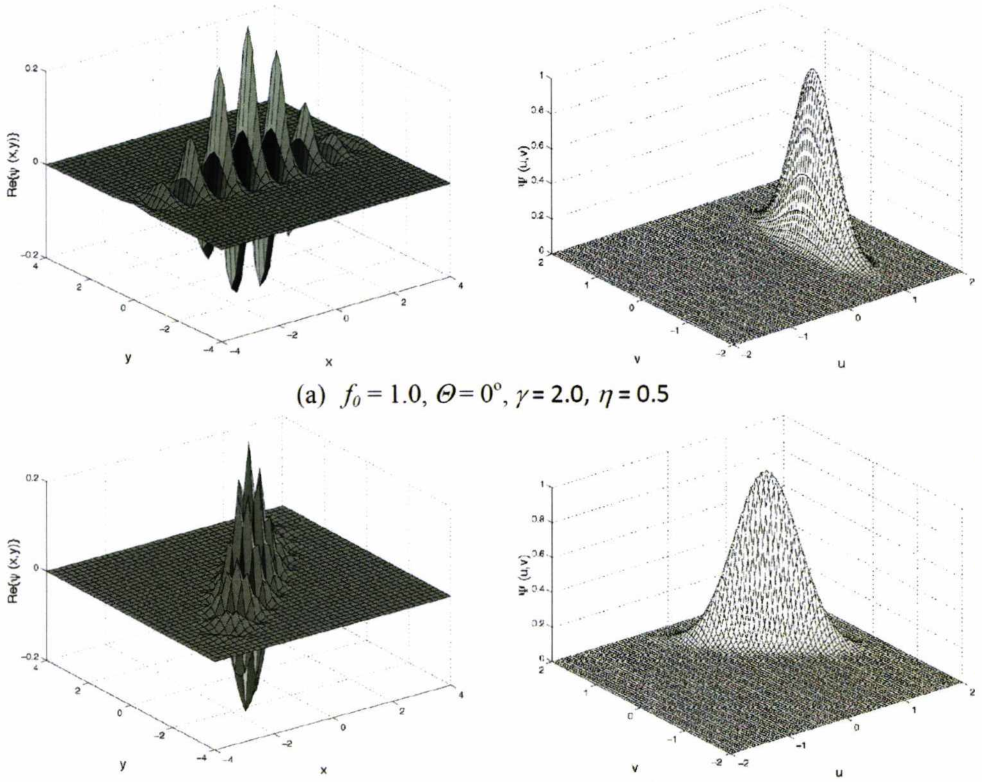
where θ_k is the kth orientation and n is the total number of orientations to be used.



(a) $f_0 = 0.5$, $\Theta = 0^\circ$, $\gamma = 1.0$, $\eta = 1.0$

(b) $f_0 = 1.0$, $\Theta = 0^\circ$, $\gamma = 1.0$, $\eta = 1.0$

Figure 2.7: 2D Gabor filter functions with different values of the parameters f_0, θ, γ and η in the space and spatial-frequency domains [97]



(a) $f_0 = 1.0$, $\Theta = 0°$, $\gamma = 2.0$, $\eta = 0.5$

(b) $f_0 = 1.0$, $\Theta = 45°$, $\gamma = 2.0$, $\eta = 0.5$

Figure 2.8: 2D Gabor filter functions with different values of the parameters f_0, θ, γ and η in the space and spatial-frequency domains [97]

However, the computations can be reduced to half because the responses at $[\pi, 2\pi]$ for real signals are complex conjugates of responses on $[0, \pi]$:

$$\theta_k = \frac{k\pi}{n}, \text{ for } k = \{0, \ldots, n-1\} \tag{2.28}$$

In order to maintain homogeneous spacing between the scales, a logarithmic relation between the frequencies f is typically established ([111]):

$$f_k = c^{-k} f_{max}, \text{for } k = \{0, \ldots, m-1\} \tag{2.29}$$

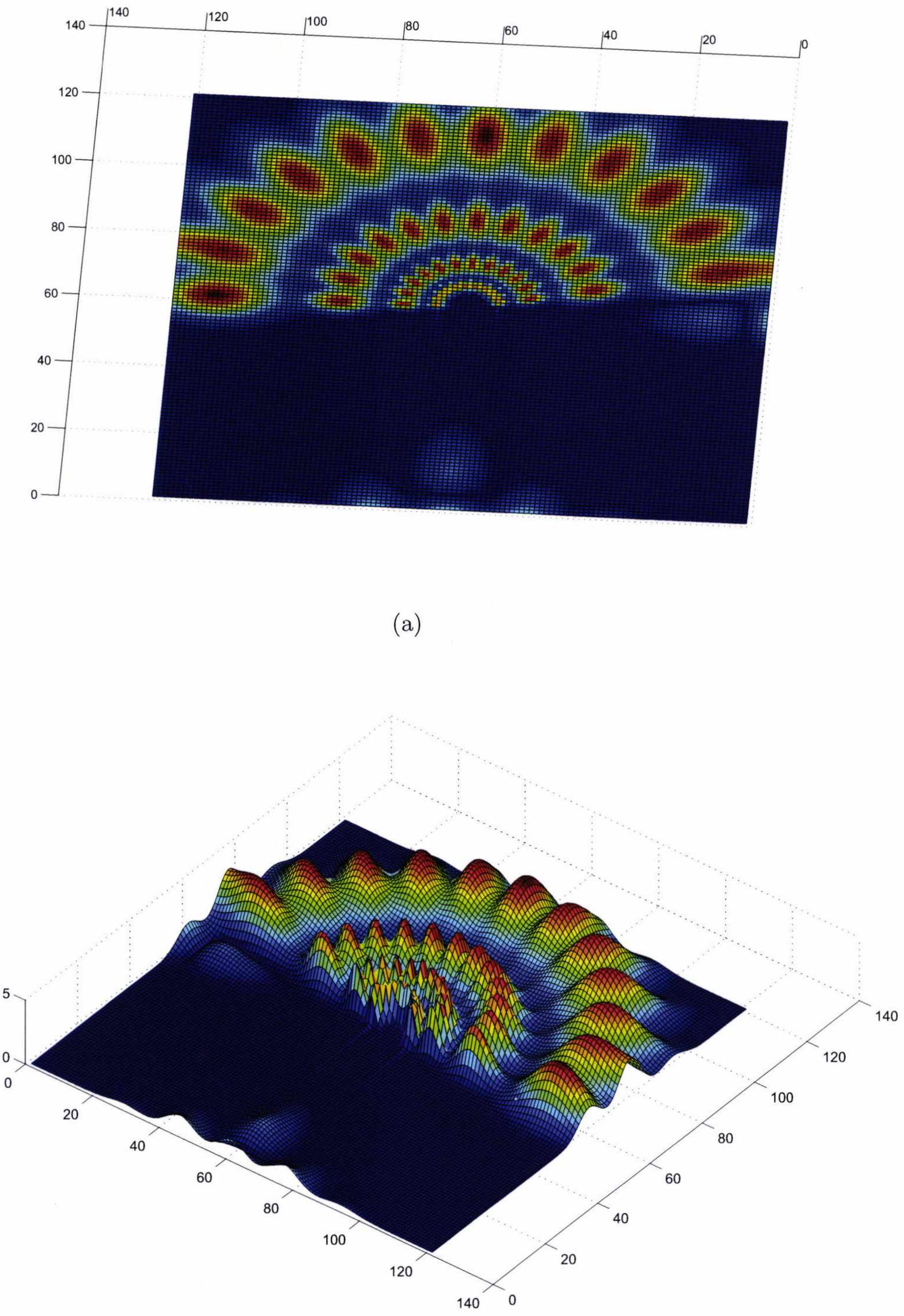
where f_k is the kth frequency, f_{max} is the maximum frequency, and c is the frequency scaling factor. Useful values for c include $c = 2$ for octave spacing and $c = \sqrt{2}$ for half-octave spacing, the later value is chosen in this research. A filter bank in Figure 2.9 is reconstructed using filter responses from 12 orientations and four frequencies.

2.4.4 Gabor-based features extraction

The bank of Gabor filters can be used to analyze the texture image because they have tunable orientation, tunable radial frequency bandwidths, and tunable center frequencies [240].

Texture analysis using Gabor-based methods is done by filtering the image with a bank of filters, each filter having a specific frequency (and orientation), and then extracting texture features from the filtered images. Due to the large feature size resulted from using many scales and orientations, different techniques have been developed to prevent the curse of dimensionality.

For instance, the dimensionality can be reduced by using only the real component



(a)

(b)

Figure 2.9: 2D Gabor filterbank of 4 scales and 12 orientations in frequency domain

[83]. Alternatively, it can be reduced by using the magnitude response [17]. Manjunath and Ma [137] suggested a simple texture features for images retrieval. In that method, mean and standard deviation of the transform coefficient at each scale and orientation were used as texture feature. This technique will be used in this thesis as described in Section 5.2.4.

2.5 Local Binary Pattern for Texture Classification: A Survey

The gray-level difference (GLD) methods closely resemble the co-occurrence approach [223]. The difference is that instead of the absolute gray levels of the pair of pixels, their difference is utilized. As features, Weszka *et al.* [223] (1976) proposed the use of the mean difference, the entropy of differences, a contrast measure, and an angular second moment. This method have inspired a variety of modifications by other including signed differences [153] and the LBP (Local Binary Pattern) operator [151].

Due to its flexibility the LBP method can be easily modified to make it suitable for the needs of different types of problems. The success of LBP methods in various computer vision problems and applications has inspired much new research on different variants [169]. Therefore, several extensions and modifications of LBP have been proposed with an aim to increase its robustness and discriminative power. In this section different variants are divided into such categories that describe their roles in feature extraction.

2.5.1 Basic LBP

The basic Local Binary Pattern (LBP) operator was originally applied for texture description [151]. It labels the pixels of an image by thresholding the 3×3-neighbourhood of each pixel with the centre pixel value and concatenating all these binary values in a clockwise direction starting with the one of its top-left neighbor. The decimal value, in Figure 2.10, is obtained by summing the thresholded differences weighted by powers of two. Finally, central pixel value is replaced with corresponding decimal value, and the 3 × 3 window slides to the next pixel.

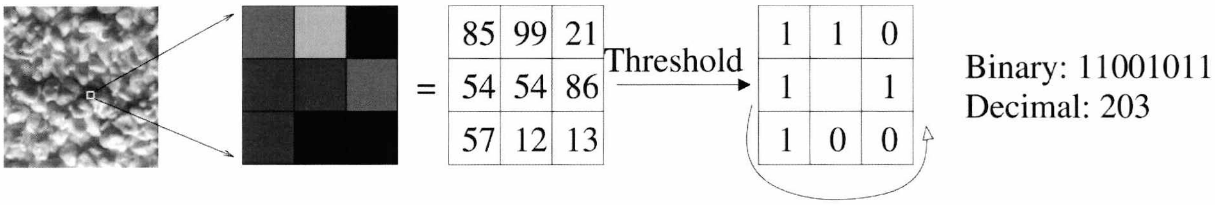


Figure 2.10: The basic LBP operator [4]

2.5.2 Dealing with textures at different scales

One limitation of the basic LBP operator was that its small 3 × 3 neighbourhood could not extract dominant features of large-scale texture primitives. The operator was later extended to use neighbourhoods of different sizes in order to be able to deal with textures at different scales [152].

A local neighbourhood is defined as a set of sampling points evenly spaced on a circle, which is centered at the pixel to be labeled, thus allowing for any radius and any number of sampling points in the neighbourhood. Bilinear interpolation is used when the sampling point does not fall within the centre of pixel.

Figure 2.11 shows some examples of circular neighbourhoods, (8, 1),(16, 2) and (8, 2), where the notation (P, R) is used for pixel neighbourhoods which means P sampling points on a circle of radius of R. Formally, the resulting LBP can be expressed in decimal form as follows:

$$LBP_{P,R}\,(x_c,\,y_c) = \sum_{p=0}^{P-1} s(g_p - g_c)\,2^p \tag{2.30}$$

where g_c, g_p are, respectively, the gray value of the central pixel $c(x_c, y_c)$ and P surrounding pixels in the circle neighbourhood with a radius R, and function $s(x)$ is defined as:

$$s(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ 0 & \text{if } x > 0 \end{cases} \tag{2.31}$$

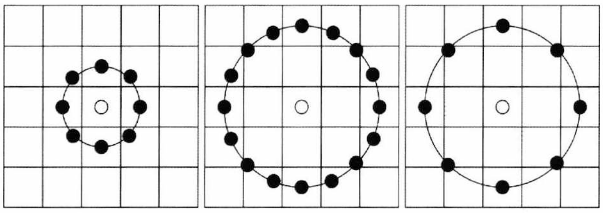


Figure 2.11: The circular (8, 1),(16, 2) and(8, 2) neighbourhoods [4]

2.5.3 Considering Uniform patterns

An LBP_{PR} is called uniform LBP^{U2}_{PR}, if it contains at most two bitwise transitions from 0 to 1 or vice versa when the bit pattern is considered circular. For example, the patterns 00000000 (0 transitions) and 01110000 (2 transitions) are both uniform, whereas 11001001 (4 transitions) and 01010011 (6 transitions) are not. The uniform mapping produces an LBP operator with less than 2^p labels. For instance, it yields only 59 labels with the neighbourhood of 8 pixels but the number of resulted labels with the same neighbourhood with the standard LBP will be 256.

There are two reasons for considering only uniform patterns and omitting others. Firstly, these patterns will be more robust and therefore they practically produced better recognition performance in many applications. Secondly, there are indications that most of the local binary patterns in natural images are uniform [152].

2.5.4 Preprocessing

In some applications, a preprocessing stage such as edge detection has been used prior to LBP feature extraction for enhancing the gradient information. Yao and Chen [235] proposed a combination of colour and local edge patterns (LEP) histograms for colour texture analysis. The Sobel edge detection is first applied and then LBP-based features are extracted, resulting in two types of LEP histograms: one is LEPSEG for image segmentation, and the other is LEPINV for image retrieval.

2.5.5 Topology of the neighbourhood

The LBP features, in the original LBP, are computed from a circle neighbourhood. The neighbourhood is designed in such an isotropic manner to obtain rotation invariance for texture description. Liao and Chung [126] claimed that the anisotropic information, in some applications, could also be an important feature. As a result, they used an elliptical neighbourhood definition, calling their LBP variant the elongated LBP (ELBP).

Figure 2.12 shows two examples of this operator, where A and B denote, respectively, the long axis and short axis of the neighbourhood, and m is the number of sampling points. Like the original LBP, the bilinear interpolation technique is adopted for sampling points that do not fall exactly at the pixels.

Nanni *et al.* [148] studied the effects of using different neighbourhood topologies (circle, ellipse, parabola, hyperbola and Archimedean spiral) for medical image analysis.

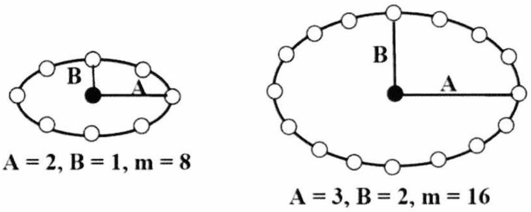


Figure 2.12: Two examples of the elongated LBP operator [126]

2.5.6 Encoding

In binary coding, the differences between the center pixel and its neighbours are encoded by two values (0 and 1). A drawback of the LBP method, is its sensitivity to random and quantization noise in uniform regions. This is due to the exact thresholding at the value of the central pixel [205].

Tan and Triggs [205] proposed a three-level operator called local ternary patterns (LTP) to improve resistance to noise in uniform and near-uniform image regions. In LTP, the difference between a neighbouring pixel and the center pixel is encoded by three values (1, 0 or -1) according to a user-specified threshold t. Formally, function

$s(x)$ in (2.18) is replaced by:

$$s(x) = \begin{cases} 1 & \text{if } x \geq t \\ 0 & \text{if } |x| < t \\ -1 & \text{if } x \leq t \end{cases} \tag{2.32}$$

The *ternary* pattern is divided into two parts: the positive one and the negative one, as illustrated in Figure 2.13. The histograms from these components computed over a region are then concatenated.

The difference between a neighboring pixel and the center pixel, in *quinary coding*, is encoded by five values (-2,-1, 0, 1, 2) according to two thresholds (T1 and T2) [148]. Nanni *et al.* [148] investigated the use of different encodings of the local grayscale differences (binary, ternary, and a quinary) and different neighbourhood topologies with three different types of medical images. It was found that the operator that uses quinary encoding in an elliptic neighbourhood provided the best performance.

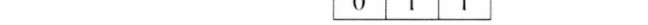


Figure 2.13: Example of the LTP operator [205]

2.5.7 Thresholding

Instead of thresholding the local pixel values against the value of the center pixel within the neighbourhood, other techniques have also been considered. Hafiane *et al.* [63] proposed Median Binary Pattern (MBP) operator by thresholding the local pixel values, including the center pixel, against the median (MBP) within the neighbourhood.

2.5.8 Multiscale analysis

In order to capture not only the micro structures but also the macro structures, a multiblock LBP (MB-LBP) was proposed ([241], [127]), which instead of comparing pixels, compares the average pixel values within smaller blocks. The block can either be a rectangle or a square. Figure 2.14 shows an example of MB-LBP, where each block consists of six pixels. This scheme has gained popularity especially in facial image analysis. For instance, it is exploited for face detection in [241] and for Face recognition in [127].

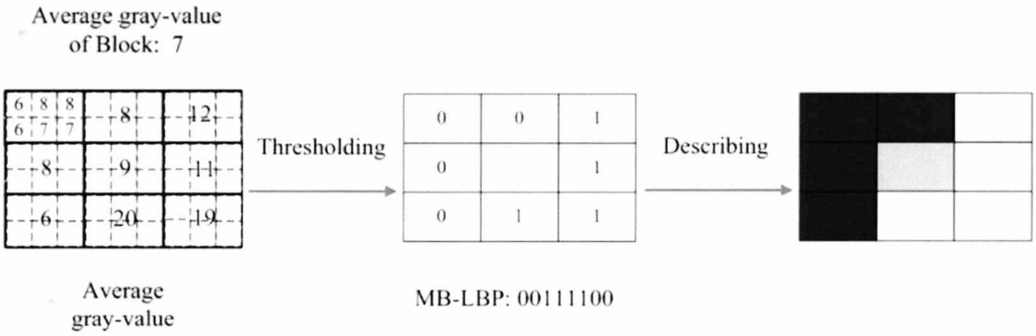


Figure 2.14: Example of the MB-LBP operator [241]

2.6 Colour Textures

Most texture analysis techniques have been proposed for grey-scale images, although the colour is one of the important characteristics that may be used in image content description. All colours can be represented as variable combination of the three primary colours, red, green, blue.

The RGB colour space is an additive model in which red, green, and blue lights are added together in various ways to reproduce the whole range of other colours [187]. There are various colour spaces that have been developed for the sensing, representation, and display of images in different electronic applications such as YIQ, HSV, LAB, etc. These models are extensively discussed in [57] and experimentally compared in [187].

A colour texture can be regarded as a pattern expressed by the relationship between its chromatic and structural distribution [46]. Basically, there could be two different cases of colour textures, two images consisting of the same colour but different texture patterns or the same texture pattern but different colours.

Since colour information is a vector quantity (In contrast to intensity which is scalar value) traditional methods of gray texture could not be extended in a straightforward to the colour domain. It is also still unclear how best to combine these two properties into a composite model. Palm [155], has categorized the methods combining color and texture into three groups: parallel, sequential and integrative approaches as illustrated in Figure 2.15.

In parallel methods, colour and texture are considered as separate phenomena. Texture is extracted by the relationship of the intensities of neighbouring pixels ignoring their colour. Colour is evaluated globally according to the histogram ignoring local neighbouring pixels. The results of both analyses are fused subsequently to a feature vector. These algorithms are most commonly used for image retrieval [141] and segmentation [158], [49] applications.

Sequential methods contain of two steps: a partitioning of the original colour image is first obtained by applying a colour indexing method. Subsequently, texture features are extracted from the indexed image because it can be processed as a gray-scale texture. The sequential methods have been used in applications that related to the structural texture model such as defect detection in granite images [200], and inspection of ceramic tiles [108]. For the sequential approach, the co-occurrence matrices are adapted in [99] and LBP and Gabor-based features are used in [15].

Integrative methods attempt to process colour and texture jointly so the information dependency between both cues are taken into account. The methods are divided into single- and multi-channel strategies. Single-channel methods perform the classical gray-scale texture analysis on each color channel separately. For instance, cooccurrence matrices [207] are adapted for integrative single-channel color texture analysis. [46]. Integrative multi-channel techniques handle simultaneously two (or more) channels. These approaches have already been proposed for different texture analysis methods like Gabor filters [84], wavelet [225], Markov random fields [202], autocorrelation features [68], and the cooccurrence matrices [155].

In [135], the authors discuss pros and cons of using texture and colour descriptor separately or jointly. It was concluded that using colour version of texture descriptors should only be used in controlled environments. Otherwise, extracting feature vectors separately (e.g. texture histograms and colour histograms) and concatenating them offers better performance.

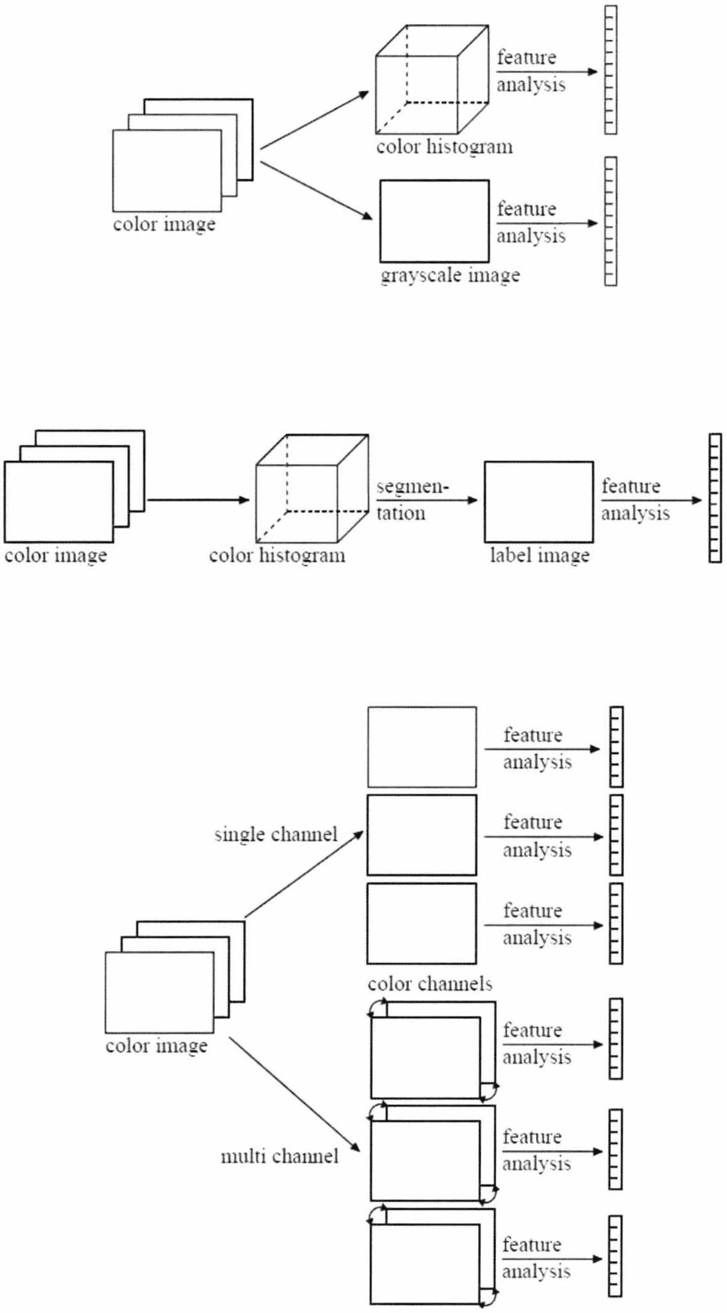


Figure 2.15: Main categories of the the approaches combining color and texture [155]

2.7 Feature Selection

The term *feature selection* is taken to refer to algorithms that select a subset of size m, from an input set of d features, which lead to the smallest classification error. This process has obvious economical benefits in terms of computational requirements and data storage. In addition, it often provides better model understanding and even better classification accuracy [81]. Therefore, it has many applications in engineering (robotics), Internet applications(text categorization), medical applications (diagnosis, prognosis), and in pattern recognition (speech, handwriting, face recognition). Basically, there are four basic steps in a typical feature selection method [113]:

1. *Starting point* to determine, in the space, the direction of search.
2. *The organization of the search.*
3. *Evaluation procedure* to evaluate the subset under examination.
4. *Stopping criterion* to decide when to stop.

In the literature, there are different taxonomies of feature selection methods based on the listed above. For instance, Doack [44] identified three classes depending on search organization: exponential, sequential and randomized. Based on the feature evaluation scheme, Siedlecki and Sklandsky [194] classified feature selection methods into past, present, and future categories while Langley [113] grouped them into *wrapper* and *filter*.

Jain *et al.* [82] presented a taxonomy of feature selection algorithms, dividing them into statistical pattern recognition (SPR) techniques and artificial neural networks (ANN). In a more recent survey, Kudo et.[107] compared (large-scale) feature selection algorithms. Their performance were measured their using a leave-one-out correct-classification rate of a nearest neighbour classifier. Generally, feature selection procedure can be performed by three search strategy groups [40]. These are complete, heuristic and stochastic search strategy.

- **Complete search** is the way to find an optimal feature subset. The first method which can be used to carry out this search is the exhaustive search. It examines all possible feature subsets (feature combinations) and selects the best one among the competing 2^n possible subsets for a given n dimensional feature space. Different heuristic methods, which introduce backtracking in the search, can be used to reduce the search without jeopardizing the chances of finding the optimal subset. (For example *Focus* by Almuallim *et al.* [6] and *Branch and Bound* [149]). However, this strategy is still impractical for high-dimensioned problems because it is time consuming.

- **Heuristic search** The generation of feature subset, in this technique, is basically incremental (either increasing or decreasing). In each iteration, all remaining features yet to be selected or rejected are considered for selection or rejection [40]. Since all possible subsets are not examined, heuristic search is not guaranteed to converge to the global optimum result. However, it is more preferable because it is very simple to implement and very fast. The most popular methods that belong to this technique are *Forward selection (FS)*, *Backward selection (BS)* methods and *stepwise (bi-directional) selection* which uses both (FS) and (BS) in association.

- **Stochastic search** This strategy search fewer number of subsets than 2^n by setting a maximum number of possible iterations. It is based on a random generation of feature subset. For achieving good results, it is essential to assign suitable values to parameters which are required in each random generation. Various methods have been suggested such as *tabu-search* [105], *simulated annealing* [102], and *genetic algorithm* [193].

Siedlecki and Sklansky [193] were the first to propose the use of genetic algorithms (GA) for feature selection. In GA, a feature subset is represented by a binary string, called *a chromosome*, of length n with a zero or one in position i denoting the absence or presence of feature i in the set. The *fitness* of each chromosome is evaluated through an optimization function in order to determine how likely the chromosome is to survive and breed into the next generation. A population is maintained and evolved by creating new offsprings from the fittest chromosomes using the process of: (i) *crossover*, where parts of two different parent are combined to create a child. (ii) *mutation*, where a child is created by randomly selecting bits in a single parent and flipping the bit value from zero to one or vice versa.

2.8 Classification methods

The classification task is to assign a class to the pattern according to the corresponding feature vector. It can be divided into two main categories: supervised classification and unsupervised classification.

In supervised classification (or discrimination), the User has a set of data samples with associated labels or the class types. These are used as exemplars in the classifier design. In unsupervised classification(or clustering), the data are not labeled and the

User seeks to find groups in the data and the features that distinguish one group from another. This section will discuss the different distance metrics and the most popular methods in the supervised classification that have been used in this research.

2.8.1 Distance metrics

In general, there are two ways to compare two patterns; one is the inverse of the other: (i) to compute the distance between the image features. (ii) to evaluate the similarity between them. Several types of functions, known as distance metrics or similarity measure, have been proposed. Duda *et al.*[50] have presented four properties for distance metrics d between vectors a and b:

1. Nonnegativity: $d(a, b) \geq 0$
2. Reflexivity: $d(a, b) = 0, \quad \text{if and only if} \quad a = b$
3. Symmetry: $d(a, b) = d(b, a)$
4. Triangle inequality: $d(a, b) + d(b, c) \geq d(a, c)$

The following distance metrics are used, in this work, because they are the most commonly used [184]:

L1 norm:

$$d(a, b) = \sum_{i=1}^{n} |a_i - b_i| \tag{2.33}$$

L2 norm:

$$d(a, b) = \sqrt{\sum_{i=1}^{n} (a_i - b_i)^2} \tag{2.34}$$

Cos distance (Normalized Correlation):

$$d(a,b) = \frac{a^T b}{||a|| \, ||b||} \tag{2.35}$$

where a_i, b_i are an $i - th$ element of vectors a and b respectively.

2.8.2 k-Nearest Neighbour

It is an easy and efficient, and also known as *lazy classification algorithm*, where there is no work done (if k is known) in training stage and all the work is conducted during testing. The algorithm first calculates the distances between a single query sample and each of the other training samples based on one of the the distance metrics that were discussed in Section 2.8.1. The training samples closest to that sample are defined as its "nearest neighbours". The query sample is then assigned to the class from which a majority of it's k nearest neighbors are from, where k is typically an integer less than 10. Choosing big values for k yields smoother boundaries, which makes the classifier more robust to noise but less discriminative [147]. When k is 1, the algorithm seeks the nearest neighbor and the label of the testing sample is recognized as its nearest neighbor's class.

The other major attraction of the kNN classifier, in addition to having to optimize the selection of only one parameter k, is its asymptotic performance. This fact comes from the famous result of Cover and Hart (1967) [34] which shows that the error rate of the 1-nearest-neighbor classifier is never more than twice the Bayes rate for the same data. The theoretical basis of the kNN classifier is well described in [39].

2.8.3 Support Vector Machines

The Support Vector Machine (SVMs), are among the most powerful methods for data classification. Thus they have attracted extensive interest in the machine learning research community [216] and have been employed in a wide range of classification and regression problems such as face detection [154], face verification [162], Gender classification [145], 3D Object Recognition [170] and bioinformatics [43]. SVM is ideally suited to two class classification problems. In this case, it aims to determine a maximal separating hyperplane between two classes so that the classification error is minimized. Figure 2.16 demonstrates the example of maximum-margin hyperplane for two sets of data points. It is shown that there are many linear hyper planes that separate the data. However only one of these achieves maximum separation. It is called Optimal Separating Hyper plane (OSH) and the samples on the margin are called the support vectors.

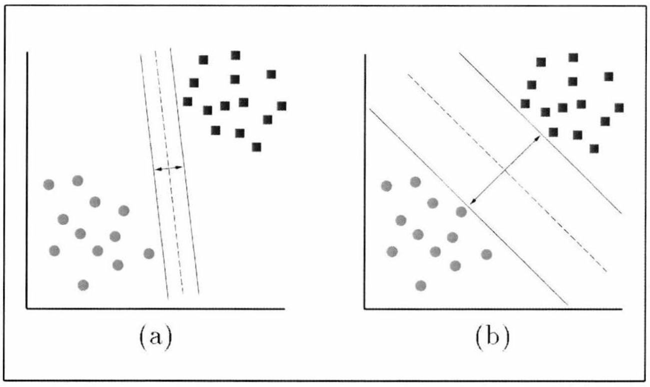


Figure 2.16: Two examples of a separating hyperplanes

Mathematically, Given a training samples (x_i, y_i) with $x_i \in \Re^n$, $y_i \in \{-1, +1\}$ for $1 \leq i \leq l$, where l is the number of instances, so the canonical hyperplanes can be defined as follows:

$$
\begin{aligned}
H_1 &: w^T\ x_i - b = +1 \\
H_2 &: w^T\ x_i - b = -1
\end{aligned}
\tag{2.36}
$$

where b is the offset of the hyperplane from the origin and w is an n dimensional vector and perpendicular to the separating hyperplane (see Figure 2.17).

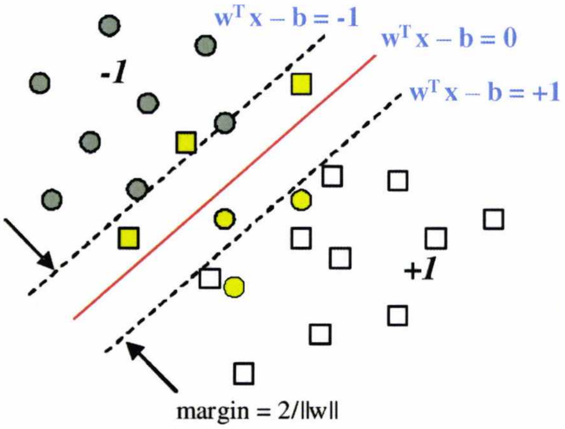


Figure 2.17: Linear separation of the data points into two classes

Based on the above definition, any solution must satisfy the following constraint:

$$
\begin{cases}
w^T\ x_i - b \geq +1 & \forall\ y_i = +1 \\
w^T\ x_i - b \leq -1 & \forall\ y_i = -1
\end{cases}
\tag{2.37}
$$

These can be combined into one set of inequalities:

$$
y_i\ (w^T\ x_i - b) \geq 1,\ i = 1, 2, \cdots, l. \tag{2.38}
$$

Below, the optimal separating hyperplane for the case where the training data are linearly separable will be first considered. Then, we generalize the idea of optimal separating hyperplanes to the case of linearly nonseparable data and kernel trick.

Linearly separable data

For linearly separable data, the separating hyperplane can be written as, $w^T\ x_i{-}b = 0$, The distance between each of canonical hyperplanes and the separating hyperplane is $1/\left\|w\right\|$. Now optimizing the separating margin is equivalent to maximizing the distance between H_1 and H_2. Since the largest width between them is $2/\left\|w\right\|$, the learning problem of SVM can be formulated as follows [222]:

$$\begin{aligned} &\text{minimize} && \tfrac{1}{2}\ \|w\| \\ &\text{subject to} && y_i(w^T\ x_i - b) \geq 1\ \ \forall i \end{aligned} \tag{2.39}$$

The optimization problem (2.39) can be solved by saddle points of the Lagrange's function, which is:

$$\text{minimize}\quad L(w,b,\alpha) \equiv \frac{1}{2}\ \|w\| - \sum_{i=1}^{l} \alpha_i\ \left[y_i\ (w^T\ x_i - b) - 1\right] \tag{2.40}$$

where $\{\alpha_i : 1, \cdots l;\ \alpha_i \geq 0\}$ are Lagrangian multipliers.

To find the values w, b, and $\alpha_i \geq 0$, that minimize L, we partially differentiate Equation 2.40 with respect to b and w and equate the derivate to zero. The dual form of the Lagrangian becomes as follows [222]:

$$\begin{aligned} &\text{maximize} && L_D = \textstyle\sum_{i=1}^{l} \alpha_i - \frac{1}{2} \sum_{i=1}^{l} \sum_{j=1}^{l} \alpha_i\ \alpha_j\ y_i\ y_j\ {x_i}^T\ x_j \\ &\text{subject to} && \textstyle\sum_{i=1}^{l} \alpha_i\ y_i = 0,\ \ \alpha_i \geq 0\ \ \forall i \end{aligned} \tag{2.41}$$

Linearly non-separable data and kernel case

In many practical problems data is not linearly separable, so the formalism need to be changed to account for that. Hence, in order to relax those constraints, positive slack variables $\{\zeta_i, i = 1, \cdots, l, \ \ \zeta_i \geq 0\}$ were introduced, into the original constraints [216] along with an additional penalty value C for the points that cross the boundaries to consider the misclassification errors:

$$\begin{array}{ll} \text{minimize} & \frac{1}{2}\,\|w\| + C\sum_{i=1}^{l}\zeta_i \\ \text{subject to} & y_i(w^T\ x_i - b) \geq 1 - \zeta_i,\ \epsilon_i \geq 0 \end{array} \tag{2.42}$$

To get rid of w, b, ζ, the previous technique (that was used in linear case) have been implemented, the dual form of the the Lagrangian is built as:

$$\begin{array}{ll} \text{maximize} & L_D = \sum_{i=1}^{l}\alpha_i - \frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l}\alpha_i\ \alpha_j\ y_i\ y_j\ {x_i}^T\ x_j \\ \text{subject to} & \sum_{i=1}^{l}\alpha_i\ y_i = 0,\ \ 0 \leq \alpha_i \leq C\ \forall i \end{array} \tag{2.43}$$

Kernel trick

The initial SVMs algorithm that suggested by Vapnik [216] was a linear classifier. Boser [16] proposed a good scheme to create nonlinear classifiers by the kernel trick. The Kernel function $K(x_i, x_j) = \phi(x_i)^T\phi(x_j)$ is a map that transforms data from the original space in which the classes may not be linearly separable, to a higher dimensional space in which they are. The discriminant function, with this function, can be written as, $g(x_i) = w^T\phi(x_i) + b$ and the corresponding dual form comes of replacing $x_i x_j$ in program 2.43 by $K(x_i, x_j)$ [19]:

$$\begin{array}{ll} \text{maximize} & L_D = \sum_{i=1}^{l} \alpha_i - \frac{1}{2} \sum_{i=1}^{l} \sum_{j=1}^{l} \alpha_i \, \alpha_j \, y_i \, y_j \, \phi(x_i)^T \, \phi(x_j) \\ \text{subject to} & \sum_{i=1}^{l} \alpha_i \, y_i = 0, \quad 0 \le \alpha_i \le C \; \forall i \end{array} \tag{2.44}$$

There are many kernel functions to build SVM-based classifier. However, the most commonly used kernel functions are:

- linear: $K(x_i, x_j) = x_i^T x_j$.
- polynomial: $K(x_i, x_j) = (\gamma x_i^T x_j + r)^d, \gamma > 0$.
- radial basis function(RBF): $K(x_i, x_j) = exp(-\gamma \left\| x_i - x_j \right\|^2), \gamma > 0$.
- sigmoid: $K(x_i, x_j) = tanh(\gamma x_i^T x_j + r)$. Here γ, r, and d are kernel parameters.

The scheme that is used for selecting kernel parameters will be discussed in Section 6.2. SVMs for Multi-classes classification problem can be found in [73].

2.8.4 Sparse Representation

A Sparse Representation-based classification (SRC) can often achieve high performance for face recognition as reported in [226]. The problem of image classification based on SRC can be formulated as follows [226]:

Let us assume that we have k classes training samples, a basic problem in pattern recognition is to correctly determine the class which a new test sample belong to. We arrange the n_i training samples from the ith class as columns of a matrix:

$$A_i = [v_{i,1}, v_{i,2}, \ldots, v_{i,ni}] \in R^{m \times n_i}. \tag{2.45}$$

A dictionary matrix A is built by concatenating $A_i, i = 1, 2, .., k$ as follows:

$$A = [A_1, A_2, \ldots A_k] \in R^{m \times k \cdot n_i}. \tag{2.46}$$

Under the assumption of linear representation, a test pattern will approximately lie on the linear subspace spanned by training samples so it can be represented as a combination of all n training samples ($n = k \cdot n_i$):

$$y = Ax \in R^m \tag{2.47}$$

where x is an unknown coefficient vector. It is noted that only those entries of x that are non-zero correspond to the class of y. This is the motivation to seek the sparsest solution to $y = Ax$, solving the following optimization problem:

$$(l^0) : \hat{x_0} = \text{ arg min} \left\|x\right\|_0 \text{ subject to } Ax = y \tag{2.48}$$

where $\left\|\cdot\right\|_0$ denotes the l^0-norm, which counts the number of the nonzero entries in a vector. It can be shown that if the solution x_0 is sparse enough, the solution of l^0-minimization problem is equal to the solution of following l^1-minimization problem [226]:

$$(l^1) : \hat{x_1} = \text{ arg min} \left\|x\right\|_1 \text{ subject to } Ax = y \tag{2.49}$$

With the solution $\hat{x_1}$ to Equation 2.49, we can compute the residual between a given probe image and each gallery image as follows:

$$r_i = \left\| y - \sum_{j=1}^{k} x_{1_{i,j}} \, v_{i,j} \right\|_2 \tag{2.50}$$

The identity of the given probe image is then determined as the one with smallest residual.

2.9 Information Fusion Techniques

Different information sources when used for the same recognition task may often lead to different errors. Therefore the fusion of complementary sources of biometric information has the potential to produce more reliable results and to reduce recognition error rates. Basically, most biometric-based authentication systems can be divided into four main components:

- Sensor unit: for capturing the biometric data;
- Feature extraction unit: the acquired data passes through this procedure in order to extract a discriminative representation of it.
- Matching unit: the generated feature vectors are compared to those stored in the template and the matching scores are produced from this stage.
- Decision unit: to determine, based matching score, if the claimed identity is genuine (accept) or it is imposter (decline).

Therefore, the fusion-based systems can occur, as illustrated in Figure 2.18 at four potential levels (or stages): sensor level, feature level, matching score level and decision level. In other words, it can be performed before implementing the classification (pre-mapping fusion) or after implementing it (post-mapping fusion). Pre-mapping techniques include fusion at the sensor and the feature levels while post-mapping techniques include fusion at the match score and decision levels.

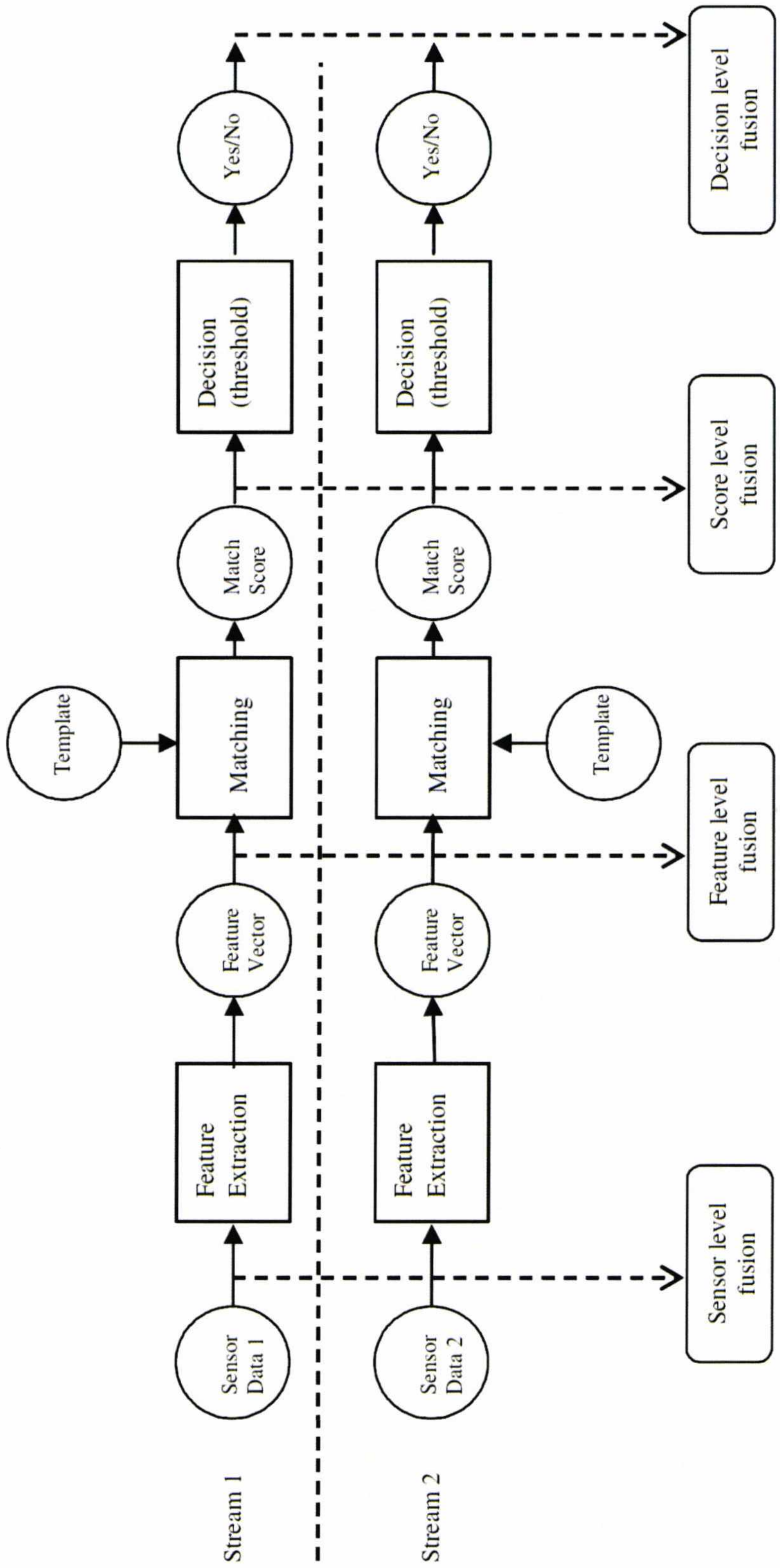


Figure 2.18: Different levels of information fusion

2.9.1 Sensor-level fusion

Sensor-level fusion can be performed either by:

(a) consolidation of raw data acquired from sensing the same biometric trait with multiple compatible sensors. For example, constructing a panoramic face using snapshots of five standard cameras that simultaneously capture multiple views of a subjects face[229] or building a 3D model of the face by combining multiple 2D face images obtained from different viewpoints [131]. Wang *et al.*[218] used two sensors to respectively capture palmprint and palm vein images. However, they suggested for the future work, to obtain palmprint and palm vein almost simultaneously using single camera by either switching the filters, or obtain fully registered images by using one camera which has good sensitivity in the visible and near infrared spectrum.

(b) integration of multiple instances (snapshots) of the same biometric trait obtained using a single sensor. A good example is *Image mosaicking* which is constructing a more complete fingerprint image or face template using multiple impressions [174] or 2D face snapshots [196] with the same camera or sensor.

Although sensor level fusion is expected to improve the recognition accuracy, it may not be applicable if the data instances are incompatible or if the correspondences between points in the raw data are not known in advance [180]. An interesting and simple combination technique between two modalities is found in [26], where the normalized, masked ear and face images are concatenated to form a combined face-plus ear image. It was found that the multimodal recognition using both the ear and face results in significant improvement over either individual biometric, for example, 90.9% percent in the analogous experiment.

2.9.2 Feature-level fusion

Feature level fusion refers to combining different feature vectors that are extracted from different multimodalities or from the same modality using different feature extraction techniques. This combination strategy is usually done by concatenating two or more feature vectors to form a single feature set.

It is stated in [178] that such a fusion type is practically difficult to achieve for a number of reasons. For example, the dimensionality of the resulting feature vector, which can lead to the curse of dimensionality problem. Although, this is a general problem in most pattern recognition applications, it is more severe in biometric systems because of cost, time and effort involved in collecting large amounts of biometric data and increasing computational load.

Hence, merging extracted features into one single feature vector usually involves applying appropriate dimensionality reduction or feature selection methods. It is also necessary to separately normalize the different feature sets before concatenating them because this step prevents certain features from dominating distance calculations merely because they have large numerical values [50].

Tan and Triggs [204] combined two of the most successful feature extraction approaches, Gabor wavelets and LBP for face recognition and they showed that the fused feature gives considerably better performance than when used separately. Examples of feature level fusion schemes extracted from different multimodalities, in the literature, can be found in Chibelushi *et al.* [31] (voice and lip shape), Son and Lee [199](face and iris) and Ross and Govindarajan[179] (face and hand geometry).

2.9.3 Score-level fusion

Each classifier, using the same biometric trait or different ones, provides a matching score indicating the proximity of the feature vector with the reference vector. At this level, it is possible to combine these scores to assert the veracity of the claimed identity. This kind of fusion, also referred as *opinion level fusion*, currently appears to be the most useful fusion level because of its good performance and simplicity.

Basically, it is divided into two groups: combination and classification. In the former approach, the matching scores are gathered using fixed rule (e.g. Maximum rule, Minimum rule, Sum rule, Product rule, Mean rule) to produce one score, which is used to make the final decision. Kittler *et al.* [103] investigated a number of different fusion methods including product, sum, min and max rules, finding that the sum rule outperformed others.

In the latter approach, the matching scores are considered as input features for a two-class pattern recognition problem and classifiers such as Neural Networks [221], SVM [183] and Decision Tree [177] can be used to arrive at the final decision. Ross and Jain [177] have shown that the combination approach performs better than some classification methods such as linear discriminant analysis and decision tree.

Prior to score fusion take place, normalization must be carried out in order to transform the scores into a common domain. According to the literature, there are various well-known score-normalization techniques (i.e. *Min-Max, Z-Score, Tanh, Median-MAD, Double-sigmoid*). Min-Max and Z-Score are chosen because they have shown to be amongst the widely used and most effective methods for this purpose

[85]. Min-max Normalization is calculated as:

$$x = \frac{n - min}{max - min} \tag{2.51}$$

where, x is the normalized score, n is the raw score, and max and min functions specify the maximum and minimum end-points of the score range respectively. Z-Score Normalization (ZS) converts the scores to a distribution with the mean of 0 and standard deviation of 1. It is given by the following form:

$$x = \frac{n - \mu}{\sigma} \tag{2.52}$$

where, n is any raw score, and μ and σ are the arithmetic mean and standard deviation of the given data.

2.9.4 Decision-level fusion

The output (decision), in a verification system, is an Accept or a Reject while in an identification system, the classifier outputs a list of possible classes with rankings for each subject in order to identify an individual.

In this fusion approach, also denoted as *abstract level fusion*, decisions that are produced from multiple classifiers are are combined via techniques such as decision level include AND/OR rules, majority voting [110], weighted voting based on Dempster-Shafer theory [228], etc.

Fusion at the decision level is considered to be the least powerful, on the basis that decisions have less information content compared to earlier levels of fusion [178]. However, Kumar *et al.*[109] showed that fusion at decision level outperformed fusion at feature level for multimodal system based on fusion of hand geometry and palmprint.

On other hand, the ranks in identification systems, could be consolidated using techniques such as: the highest rank, the Borda count,and logistic regression approaches [71], etc. This technique is called *rank-level decision.*

2.10 Conclusion

The chapter has been structured in two blocks. In the first, the different applications of texture analysis were discussed and works on the topic of texture analysis approaches and the techniques that have been used for feature selection were reviewed. In the second block, the classification approaches that have been used in this thesis were illustrated in details and the different schemes for fusing information were described.

Chapter 3

Face Recognition: Review

3.1 Introduction

Depending on the application, a face recognition system can be working either in *identification* or *verification* mode. Face verification involves a $1:1$ matching (comparisons) as the system needs to confirm or reject the claimed identity associated with the input face, whereas face *closed-set* identification is a $1:N$- comparisons problem. The input to the system is an unknown face, and the system reports back the determined identity from a database of known individuals. Phillips *et al* in the Face Recognition Vendor Test (FRVT) 2002 [165] define another scenario referred to as the *watch-list* or *open-set* identification, where the test individual in this mode may or may not be in the entire database. The image is recognized if a close enough match is on the stored watch list. All the identification experiments in this thesis are closed-set.

Facial recognition systems are usually consist of four stages, (1) detection of the face (in a complex background) and localization of its exact position, (2) extraction of facial features such as eyes, nose, etc, followed by (3) normalization to align the face with the stored face images, and (4) face classification or matching. Figure 3.1 shows the architecture of two face recognition system modes. The following sections describe each of these main sections in some detail and present a review of literature.

3.2 Challenges of Face Recognition

Over the past two decades, major advances have occurred in the area of face recognition, with many systems capable of achieving recognition rates greater than 90% [166]. However, real-world scenarios (such as those encountered in uncontrolled environments) remain a challenge [1],this comes from an inadequacy in the following two directions:

1. **Face acquisition process**: The performance of face recognition systems is significantly affected by facial appearance-based factors, some of them intrinsic (facial expression, age) and others are extrinsic (pose changes, partial occlusions and uncontrolled environmental conditions (illumination variations)). These variations are further increased by changes in the camera parameters, such as aperture, sensor spectral response and lense aberrations. Figure 3.2 illustrates some of these challenges.

   Some recent efforts have focused on using video input, and different features (e.g., skin texture) to overcome the performance drawbacks in 2D still face recognition [1]. Others tried to investigate the use of novel sensors, such as infrared or 3D images, which can help overcome limitations due to viewpoint and lighting variations [55]. An interesting comparative study between the

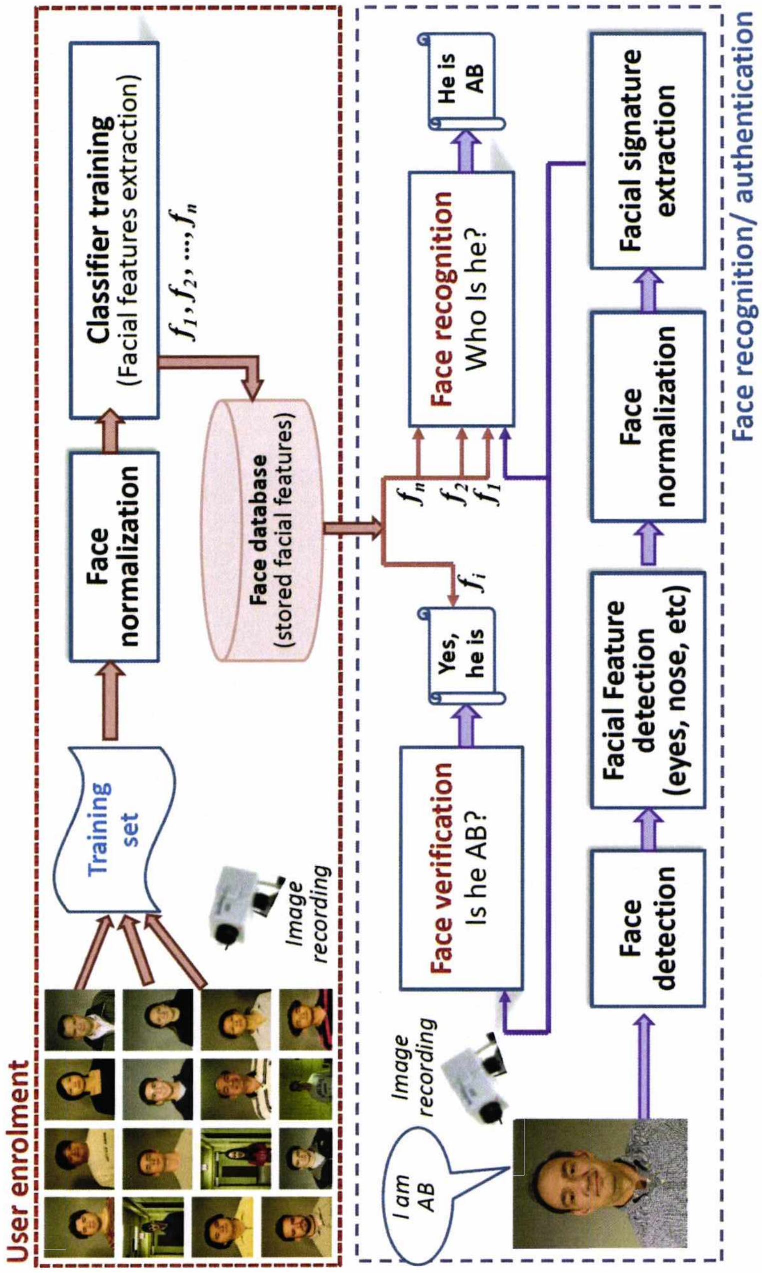
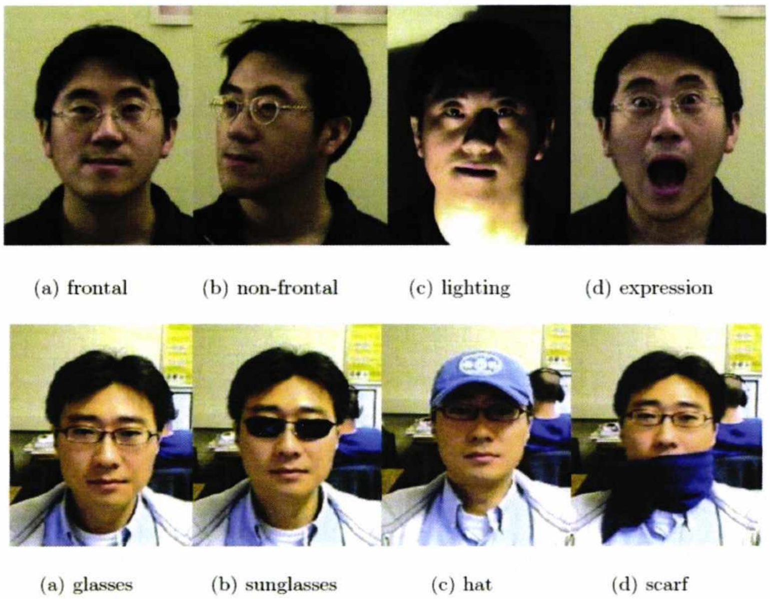

User enrolment
Training set
Face normalization
Classifier training (Facial features extraction)
$f_1, f_2, ..., f_n$
Face database (stored facial features)
Image recording
I am AB
Image recording
Face verification Is he AB?
Yes, he is
f_i
f_n
f_2
f_1
Face recognition Who Is he?
He is AB
Face detection
Facial Feature detection (eyes, nose, etc)
Face normalization
Facial signature extraction
Face recognition/ authentication

Figure 3.1: Block diagram of enrollment, face identification and face verification

performance of visible and infrared techniques can be found in [29].

2. **Pattern classification process**: Many face recognition systems suffer from the so-called small sample size problem (SSS) which exists if the number of available training samples per individual is much smaller than the dimensionality of the sample space. Therefore, the system cannot build reliable models of each person to recognise the face identity from a probe image.



(a) frontal (b) non-frontal (c) lighting (d) expression

(a) glasses (b) sunglasses (c) hat (d) scarf

Figure 3.2: Examples of different challenges (adapted from [156]

3.3 Face Detection

The first problem that needs to be addressed in any face processing system is *face detection*. This is concerned with identifying and locating human faces in an image regardless of (i) their position (ii) scale (iii) presence or absence of glasses and beards

(iv) facial expressions (v) occlusion between faces or if faces are partially occluded by other objects (vi) image illumination (vii) image orientation. *Face localization* is a face detection problem with the assumption that an input image contains only one face. A comprehensive survey of face detection approaches can be found in Yang *et al* [232] and Hjelmas *et al* [70].

Face detection plays a critical role for the success of any face processing system such as face recognition, facial expression recognition, facial feature extraction, biometric systems, face tracking, gender classification, and attentive user interfaces. Additionally, many algorithms used for face detection can be extended to detect other objects like pedestrians, cars, and signs, etc.

The following subsections give a brief review of face detection techniques in gray images and the Viola-Jones algorithm as the most popular face detector in the gray facial images. Then, two famous techniques for detecting faces in colour images are presented. They are skin-based detection and illumination-based detection. Finally, recent advances of face detection approaches are briefly discussed.

3.3.1 Face Detection in Gray-Scale Images

Face detection in gray facial images, according to [232], can be classified into four categories:

- **Knowledge-based methods**: In this top-down approach, face is represented using a set of human-coded rules that describe the facial features and their relationships. For example, the centre part of face has uniform intensity values, the difference between the average intensity values of the centre part and the upper part is significant and the face often appears with two eyes that are symmetric to each other, a nose and a mouth. Yang and Huang [230] presented a hierarchical knowledge-based method to detect faces.

- **Feature-based methods**: In contrast to the previous approach, numerous methods have been proposed to first detect invariant structural features and then to verify the presence of a face. This bottom-up approach can be performed based on several cues such as edges of facial features (eyes, nose, mouth, eyebrows, etc) [236], face texture [38], shape information [8], etc.

- **Template Matching Methods**: A standard face pattern is hand-coded (not learned) by (a) predefined template that based on edges or regions such as the shape template [35] or (b) deformable template that is based on facial contours like Active Shape Model (ASM) [114]. In these methods, the correlation values of an input image with the standard patterns are computed to locate faces.

- **Appearance-Based Methods**: Contrasted to the template matching methods where models are predefined by experts, appearance-based models are learned from positive and negative examples of faces. In general, these methods rely on techniques from machine learning and statistical analysis to find relevant characteristics in face and non-face images. Different classifiers have been proposed for the training stage, in appearance-based methods, such as Neural network [182], Eigenface [213], Support Vector Machine (SVM) [154], Adaboost [52], etc.

3.3.2 The Viola-Jones Detector

Viola and Jones detector also known as "Haar-Cascade" [217] has been shown to be a highly effective face detector that is used in a large number of real-time face detection applications. There are three main characteristics of this method to obtain an efficient performance and good accuracy:

Haar-like features

The features that Viola and Jones used are based on Haar wavelets. Haar wavelets are single cycles of the square waves (one low interval and one high interval). A square wave, in two dimensions, is a pair of adjacent rectangles - one dark and one light. Figure 3.3 shows four different simple feature types that were used in the standard Viola-Jones detector.

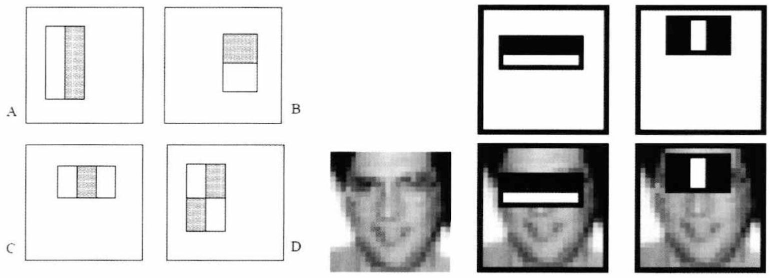


Figure 3.3: Haar-like features, the pixelsum of the white areas are subtracted from the pixelsum of the black areas [217]

Integral images

The integral image is also known as "Summed-up table". It is a two dimensional lookup tables in the form of a matrix with the same size of the original image. The integral value for each pixel is the sum of all the pixels above it and to its left. This allows to calculate the rectangle sum at any position or scale using only four lookups:

$$sum = I(4) + I(1) - I(2) - I(3) \tag{3.1}$$

where A,B,C,D belong to the integral image I, as illustrated in Figure 3.4. These sums will be compared to thresholds calculated during training stage. Based on those feature templates within a window of 24 × 24 pixels features are detected at all positions and sizes which can lead to a very large number of features and associated numbers.

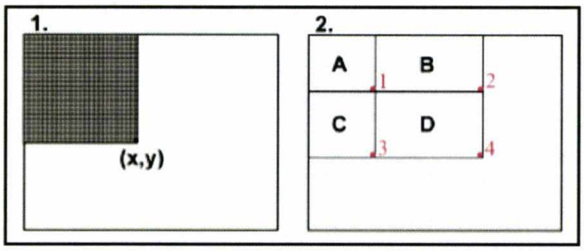


Figure 3.4: The Integral Images trick [217]

Adaboost

The Adaboost (*adaptive boosting*) is a machine-learning algorithm that aims to select few best features that fit the positive examples from the very large number of integral image features [52]. A cascade architecture is adopted to build a strong classifier from a sequence of weak classifiers (Figure 3.5).

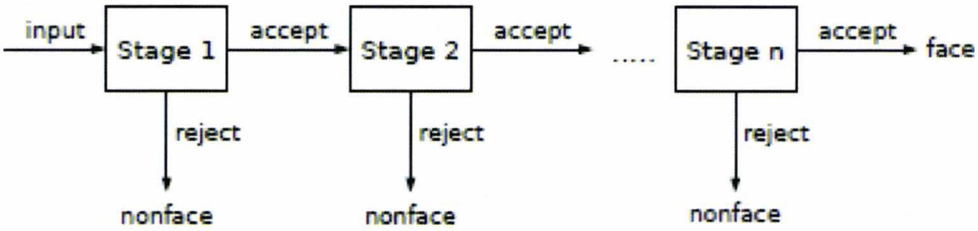


Figure 3.5: Cascade architecture (adapted from [217])

Recent Advances

Due to its robustness, Viola-Jones face detector has become dominant in many real world applications, hand detection [104], eye detection [134], license plate detection and even for ecological applications such as wild life surveillance [20].

It is also used for eye localization to localize the face in the first step. For instance, in [51], three approaches for eye localization were applied on a facial region that was obtained through Viola-Jone face detector and then these different approaches were compared. Recently, Asteriadis et al.[9] proposed a scheme that first implements Viola and Jones detector to locate the face and utilizes distance vector fields (DVFs) to locate facial features.

Despite the excellent performance of such a system, the training time is rather lengthy (may take days in training) so recent work has focused on the improvement of the boosting architecture. These include the enhancement of AdaBoost training [122], the use of new features to represent faces in different poses [45], [75] and optimal selection of features by exploiting their statistics before the training stage [227], [18].

3.3.3 Skin-Based Face Detection

In many applications from face detection to hand tracking, human skin colour detection has been used and it has proven to be an effective feature. Different colour paces have been utilized to label pixels as skin such as RGB [171], HSV (or HSI) [197], YCrCb [23], etc. Several methods have been suggested to build a skin colour model such as parametric or nonparametric methods [21], [87], [37].

The parameters in a unimodal Gaussian distribution are often estimated using maximum likelihood [21], [100], [233]. The colour histogram for the skin of people with different ethnic backgrounds does not form a unimodal distribution, therefore, this multimodal distribution is represented by mixtures of Gaussians. The parameters in a mixture of Gaussians are usually estimated using an EM algorithm [87],

In contrast to the parametric methods mentioned above, Crowley *et al* [37] built a histogram $h(r, g)$ of (r, g) values in normalized RGB colour space. According to this algorithm, a pixel is classified to belong to skin colour, if $h(r, g) \geq \tau$, where τ is a threshold selected empirically from the histogram of samples. Jones and Rehg [93] conducted a large-scale experiment analyzing 1 billion labeled skin tone pixels which were collected in normalized RGB colour space. They found that the performance of the histogram model for skin detection outperformed a Gaussian mixture model in accuracy and in computational cost.

3.3.4 Illumination-Based Face Detection

Hsu, Mottaleb and Jain [74] proposed a face detection algorithm for colour images. In this approach, face localization was performed using a lighting-compensated skin detection model in YCbCr colour space based on Terrillon *et al* [206] comparison of nine colour spaces. A lighting compensation method was first introduced because different lighting conditions causes the appearance of the skin-tone to change. A nonlinearly transformation was performed on the YCbCr colour space to make the skin cluster luma-independent because the skin-tone colour depends on luminance. A parametric ellipse in the nonlinearly transformed Cb-Cr colour subspace is used as a model of skin distribution.

This algorithm then exploits facial features such as eyes, mouth and head contours for verifying each face candidate. According to this method, two separate eye maps, one from the chrominance components of the image and the other from luminance component, are built. The eye map from the chroma is based on the observation that high C_b and low C_r values are found around the eyes. It is constructed by:

$$EyeMapC = \frac{1}{3}\left\{(C_b)^2 + (\tilde{C}_r)^2 + (C_r/C_b)\right\} \tag{3.2}$$

where C_b, C_r and $\tilde{C}_r$ are the normalized blue, red and negative of red chroma components respectively. These values were normalized to the range $[0, 255]$. The eye map from luma is based on the observation that eye regions contain both dark and bright pixels, so brighter and darker pixels in this component around eye can be emphasized by implementing gray-scale morphological operators (e.g. erosion and dilation). They use gray-scale dilation and erosion with a hemispheric structuring element to build the eye map from the luma as follows:

$$EyeMapL = \frac{Y(x,y) \oplus g_\sigma(x,y)}{Y(x,y) \ominus g_\sigma(x,y) + 1} \tag{3.3}$$

where the gray-scale dilation $\oplus$ and erosion $\ominus$ operations on a function $f : F \subset \Re^2 \rightarrow$

$\Re$ using a structuring function $g_\sigma : G \subset \Re^2 \rightarrow \Re$ are defined in [80]. The eye map from the chroma is enhanced by histogram equalization and then combined with eye map from the luma by an AND (multiplication) operation, i.e.

$$EyeMap = (EyeMapC)AND(EyeMapL) \tag{3.4}$$

The resulting eye map is then dilated and normalized to brighten both eyes and suppress other facial areas as shown in Figure 3.6

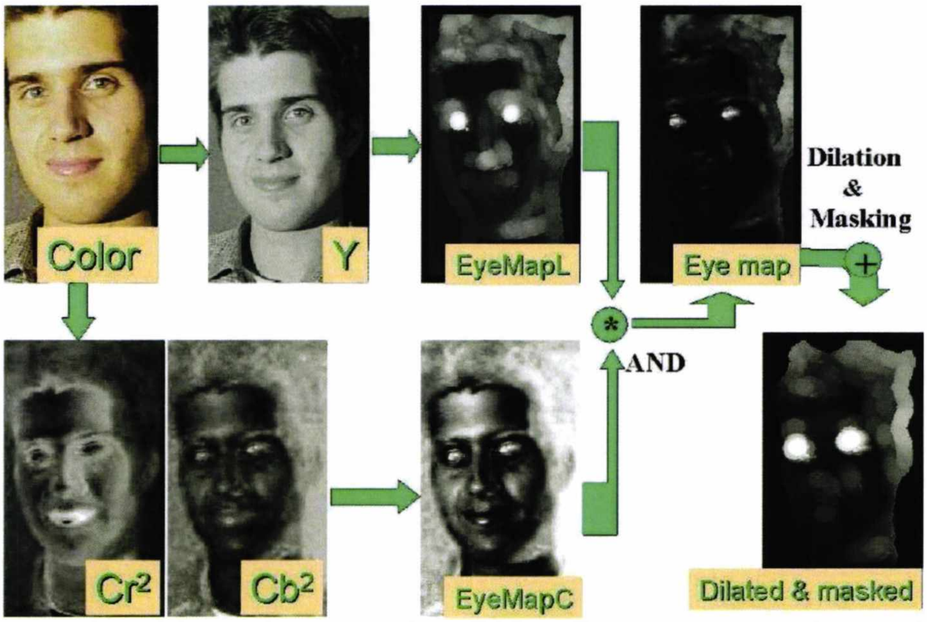


Figure 3.6: Eye Map Construction adapted from Hsu et al.[74]

The mouth map, according to this method, is based on the observation that mouth region contains stronger red component and weaker blue component as compared to other facial regions, which means in YCbCr colour space that C_r is greater than C_b for mouth region. The mouth also has a relatively low response in the C_r/C_b feature, but it has a high response in C_r^2. Therefore its map (Figure 3.7) is constructed as follows:

$$
\begin{aligned}
MouthMap &= C_r^2 \cdot (C_r^2 - \eta \cdot C_r/C_b)^2 \\
\eta &= 0.95 \cdot \frac{\frac{1}{n}\sum_{(x,y)\in FG} C_r(x,y)^2}{\frac{1}{n}\sum_{(x,y)\in FG} \cdot C_r(x,y)/C_b(x,y)}
\end{aligned}
\tag{3.5}
$$

Then they form an *eye-mouth triangle* for all possible combination of two eye candidates and one mouth candidate. Every eye-mouth triangle is verified by checking (i) luma variations of eye and mouth blobs; (ii) geometry and orientation constraints of the triangles; and (iii) the presence of a face boundary around the triangles. A face score is computed for each verified eye-mouth triangle based on its eye/mouth maps, ellipse vote and face orientation and the face with highest triangle score that exceeds a threshold is retained.

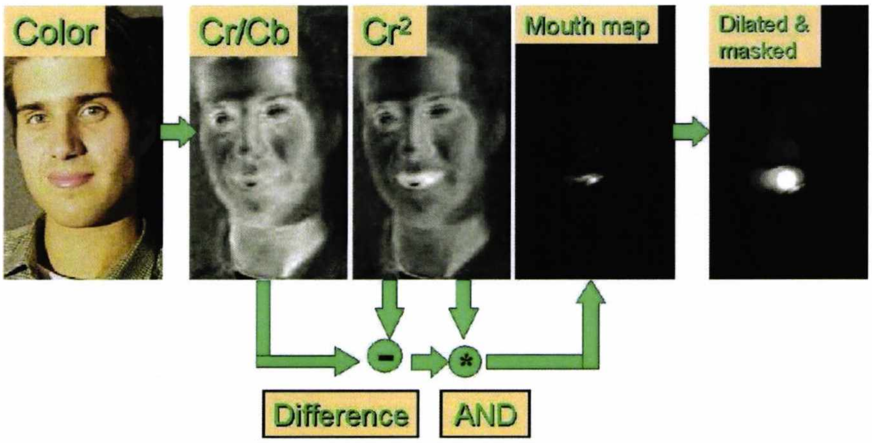


Figure 3.7: Map-based Mouth Construction adapted from Hsu et- al.[74]

3.4 Face Normalization

After a face has been detected in an image by a face detector, there are sometimes pre-processing steps which may be implemented to improve the performance of the facial recognition systems.

- **Geometric Normalization** is one of the key steps in most face recognition systems. Especially for systems that are based on the frontal views of faces, it may be desirable to normalize the shifts and rotations of facial images in the head position so that image comparison and feature extraction may be conducted on corresponding areas of facial images.

  Formally, let $(x_r,\ y_r)$ and $(x_l,\ y_l)$ be the locations of the centres of right and left eyes respectively, the amount of horizontal rotation of the face, is then determined using Equation 3.6:

  $$\theta = \tan^{-1}\left(\frac{y_r - y_l}{x_r - x_l}\right) \tag{3.6}$$

  In order to align the eyes, the whole image is then rotated by the angle θ. The direction of the rotation (clock-wise or counter clock-wise) is determined by the polarity of difference of vertical distances $(y_r - y_l)$ between the right and left eye centres. The new coordinates of every point in the face and, therefore, the new locations of eyes and mouth centres are given by Equation 3.7:

  $$\begin{bmatrix} \acute{x} \\ \acute{y} \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \tag{3.7}$$

- **Illumination normalization**: Face images taken under different illuminations can reduce recognition performance. The illumination normalization technique aims to eliminate these effects among different images. It is generally performed after the geometric normalization. Figure 3.8 shows example of these techniques: Multi Scale Retinex (MSR) [91], the wavelet-based normalization technique (WA) [47], Istoropic-diffusion based normalization (IS) [69] and the adaptive non-local means based normalization technique (ANL) [201].

- **Image size normalization**: The acquired image size is changed to a default image size on which the face recognition system operates.

- **Histogram equalization**: If the images are too bright or too dark this step is done in order to enhance image quality because it modifies the contrast range of the image and therefore some important facial features become more apparent.

3.5 Face Recognition Approaches

Still Face recognition techniques can be categorized into two main classes [14]: global-based (Holistic) approach and Local-based (Analytic) approach. Some recent methods try to exploit the advantages of both approaches at the same time so they are classified as hybrid. A comprehensive survey of 2D face recognition approaches is found in [245] and [1].

In Holistic approach, a single feature vector that encodes the whole face image is used as the input of a classifier. This technique works well for classifying frontal views of faces. However, its performance can be highly affected by face occlusion such as wearing sunglasses or masks and image variation due to pose and or illumination

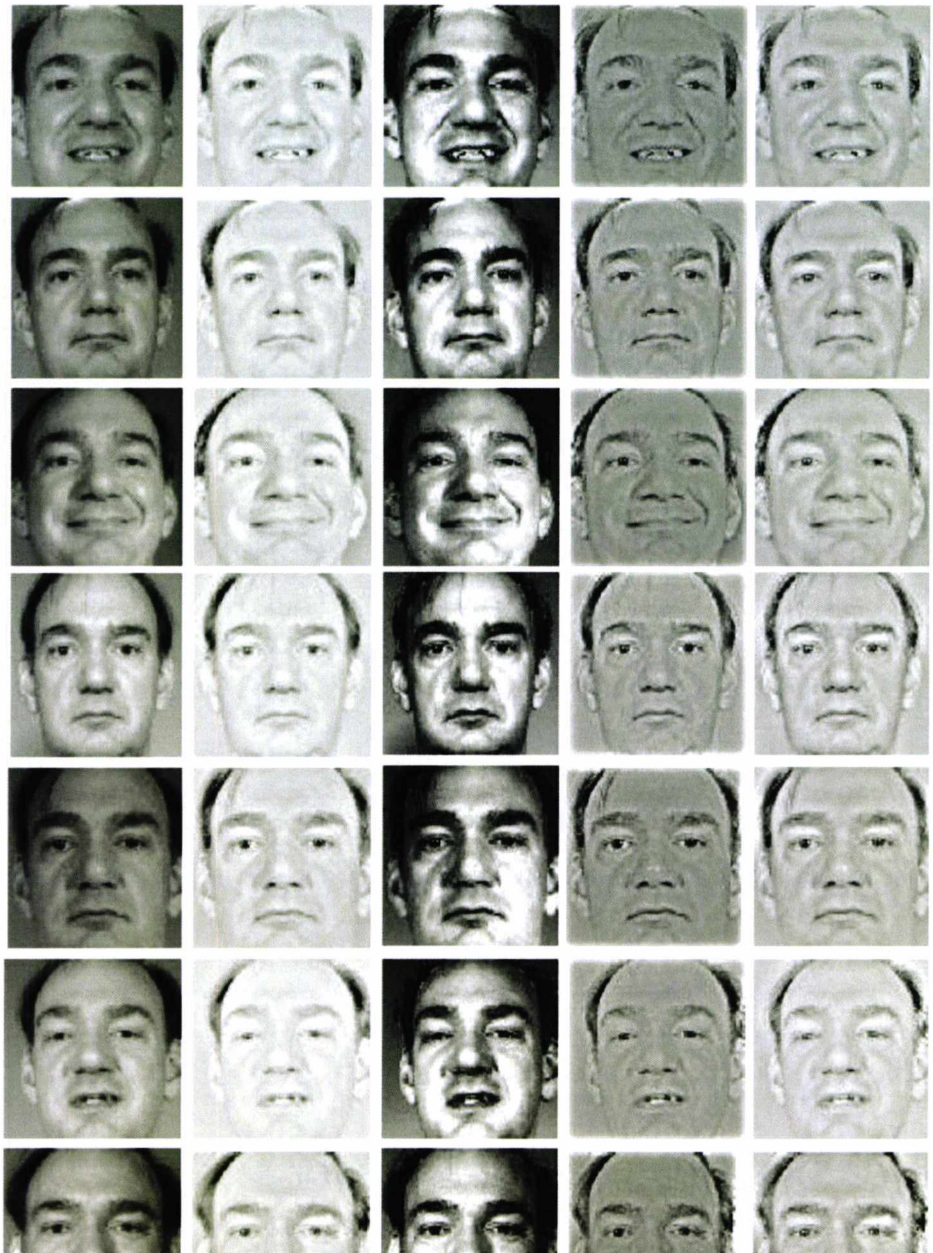


Figure 3.8: Different techniques for Illumination normalization; (a) first column: Original image (b) second column: MSR [91](c) third column: WA [47](d) forth column: IS [69](e) fifth column: ANL [201]

change [101]. To avoid this problem face alignment can be implemented before the classification stage.

In contrast with the previous approach, classifying local facial components can limit the influence of the facial acquisition conditions to those within the small component regions. To discriminate between different persons using such systems, facial features such as mouth, nose and eyes have been used for face recognition [224]. Heisele [14] compared local and global approaches and concluded that "the component system outperformed the global systems for recognition rates larger than 60%".

LBP and Gabor wavelets are considered as two of the most popular and successful local face representations, in recent years. Gabor features extract shape information over a broader range of scales. In contrast, LBP is a good choice for coding fine details of facial appearance. Both representations are computationally efficient, rich in information and their complementary nature makes them a good choice for fusion. A simple fusion technique is to separately extract Gabor and LBP features in parallel, and then, combine them on feature level, matching score level, or decision level [205]. Tan and Triggs [205] combined these two feature sets and projected them to PCA space. They concluded that the combination method has only around 2/3 of the errors of either feature set alone.

Another way of fusion is the serial technique, which consists in applying multiscale Gabor filtering prior to LBP feature extraction. Zhang *et al* [242] proposed the extraction of LBP features from images obtained by filtering a facial image with 40 Gabor filters of different scales and orientations, and only the magnitude value of the result was taken into their account. A downside of the method which is called Local Gabor Binary Pattern Histogram Sequence (LGBPHS), is the high dimensionality of the representation of the extracted features. Zhang *et al* [238] argued that phase parts of Gabor wavelet are also useful for face recognition. Thus they presented the Histogram of Gabor Phase Patterns (HGPP) method which encoded the Gabor phases through LBP and forming local feature histograms.

Zou *et al* [246] compared PCA, Gabor wavelets and LBP for FERET and AR databases. The comparison of the three local feature representations was conducted on four 37×37 windows centred at four facial landmarks (two eyes, nose, and mouth). They concluded that Local Binary Pattern is a good local feature, but it is inadequate for non-monotonic illumination changes, which often appear in facial regions such as the nose. The Gabor jet is their choice for local feature representation because of its robustness to illumination variations. Their main drawback was that they still detected facial features manually.

3.5.1 Holistic Approaches

In the Holistic approach, a single feature vector that encodes the whole face image is used as the input to recognition system so it is provided in high-dimensional form. A Dimensionality Reduction (DR) is the common way to attempt to resolve this problem. Principal Component Analysis (PCA) [213] and Linear Discriminative Analysis (LDA) [12] have been the two most popular for this purpose and two state of art FR methods, *Eigenfaces* and *Fisherfaces*, built on the two techniques, respectively have been proved to be efficient and very successful.

However, PCA and LDA consider only the global scatter (Euclidean structure) of training samples and fail to discover the underlying structure, if face images lie on a nonlinear submanifold hidden in the image space. To overcome these drawbacks, the manifold learning methods were proposed by assuming that the data lie on a low dimensional manifold of the high dimensional space [181]. Locality Preserving Projection (LPP) is one representative of these methods. A face subspace which is obtained by LPP is characterized by a set of feature images, called *Laplacianfaces* [67].

It is generally believed that LDA-based algorithms are superior to PCA based ones. However, many LDA based algorithms suffer from SSS problem. The traditional solution to the SSS problem is to first utilize PCA as a pre-processing step and then LDA is performed in the lower dimensional PCA subspace. More effective solutions, called Direct LDA (D-LDA) algorithms, have been proposed [28].

Although the so called Direct LDA (D-LDA) method [28] provides an effective solution to this problem, it is still a linear method. Thus it could fail to capture the important information that may be contained in higher order relationships among the image pixels of a face pattern. As a result, there has been interest in developing low dimensional representations through kernel based techniques for face recognition, for instance, Kernel PCA (KPCA) [186] and Kernel LDA (GDA) [10]. These methods can discover and model the nonlinear structure of the face images.

The enhanced Kernel D-LDA (KDDA) [133] is a new kernel discriminant analysis algorithm that built on D-LDA and GDA so it generalizes the strengths of the recently presented D-LDA and the kernel techniques while at the same time overcomes many of their limitations. Additionally, its computational complexity is significantly reduced compared to the other two popular kernel methods, GDA and KPCA.

3.5.2 Gabor-based methods for face recognition

There are two possible strategies for extracting Gabor-based features [191]: either using features from the whole filtered image (holistic methods), or extracting the features from selected points (nodes) in faces and using them for recognition (analytic methods).

Holistic methods suffer from a dimensionality problem (feature vectors are too long) so this problem has been tackled by the use of dimension reduction or down-sampling techniques [130]. Analytic methods mainly vary in the way they select the nodes, which can be divided into: graph-matching based methods such as Dynamic Link Architecture (DLA) [112], Elastic Bunch Graph Matching (EBGM) [224] and non-graph matching based methods, for instance, detecting feature points manually [172] or by applying a ridges and valleys detector to a face image [89]. The recent improvement of the analytical methods includes the optimal selection of Gabor parameters via boosting techniques [234].

3.5.3 LBP-based methods for face recognition

Ahonen et al [3] proposed the basic methodology for LBP based face description. It divides the facial image into local regions and LBP texture descriptors are extracted from each region independently. The descriptors are then concatenated to form a global representation of the face. Extensive experiments in [3] showed the superiority of the proposed scheme over all considered methods (PCA, Bayesian Intra/extrapersonal Classifier and Elastic Bunch Graph Matching) on all four probe sets of the FERET database.

Most of the existing works adopt the aforementioned technique to extract LBP features for facial representation. To address the issue of selecting LBP settings (such as the size and the location of local regions, the number of neighboring pixels, the most discriminative bins of an LBP histogram etc.), boosting learning is commonly used. For instance, by shifting and scaling a subwindow over face image, in [239], many more subregions are obtained and boost learning is adopted to select the most discriminative subregions in terms of LBP histograms. Compared with the approach in [3], the boosting LBP-based method achieves a slightly better recognition accuracy.

The authors of [244], employed Laplacian PCA (LPCA) for LBP feature selection and pointed out the superiority of LPCA over PCA and KPCA for feature selection. Another approach for deriving compact and discriminative LBP-based feature vectors consist of applying Dimensionality reduction methods (or Subspace learning). For example, a linear discriminant analysis (LDA) is exploited to project high-dimensional multiscale LBP features into a discriminant space [24]. Yang and Wang [231] introduced Hamming LBP for face recognition. The experimental results on the FRGC dataset revealed that the Hamming LBP outperforms the original LBP, especially when variations of illumination and facial expression exist.

Liao and Chung [126] investigated the impact of using an ellipse neighborhood for LBP-based feature. The combination of this LBP variant, an elliptical binary pattern (EBP), with a local gradient measure provided improved results in face recognition experiments compared to the ordinary LBP. Another extension of LBP operator is the multiscale block local binary pattern (MB-LBP) which has gained popularity especially in facial image analysis. Li et al. [127] utilized MB-LBP operator for face recognition.

Tan and Triggs [205] developed a very effective preprocessing chain for normalizing illumination variations in face images. Their method consists of gamma correction, Difference of Gaussian (DoG) filtering, masking (optional) and equalization of variation. In addition, a three-level operator called Local Ternary Patterns (LTP) has been proposed and employed for face recognition.

3.6 Partial Face Recognition

There have been a considerable number of research studies addressing the face recognition problem from full frontal/profile facial images. However, there have been only a few studies reported in the literature for Partial Face Recognition (PFR) scenarios. Sato [185] showed that certain facial sub-images (such as eye, nose, and ear) could be used for recognition. It has been shown in [62] that there is no significant difference between the half (right or left) and full face images for recognition.

A new framework that utilizes the eye region and the bottom face region for access control is presented in [150]. More recently, an alignment free approach [125], is presented to deal with the PFR problem. That approach adopts a variable-size description which represents each face with a set of keypoints descriptors. Then, a probe face image is sparsely represented by a dictionary of gallery descriptors.

3.7 Skin Information for Face Recognition

Some researchers have focused their work on studying only *skin marks*. For instance, Pierrard *et al* [168] presented a method to extract irregularities in facial skin, in particular nevi (moles, birthmarks) using normalized cross correlation (NCC) matching and a morphable model. Lee *et al* [118] proposed a Content-Based Image Retrieval

(CBIR) for matching and retrieving tattoo images which are considered as a soft biometric [86]. Firstly , features of tattoos were extracted based on SIFT and then their system computed the similarity between the query tattoo image and tattoos in the criminal database.

In [86], an Active Appearance Model (AAM) was used to locate and segment facial organs (e.g., eyes, nose, and mouth). Then, Laplacian-of-Gaussian (LoG) and morphological operators were used to detect facial marks. Experimental results based on FERET and Mugshot databases show that the use of facial marks improves the rank-one identification accuracy of a state of the-art face recognition system. A combined algorithm consists of a traditional Eigenfaces matcher with a skin mark matcher has been proposed in [243]. The resulting combined face matcher delivers better performance than either one of the single matchers. The AR Face Database was used in the experiments.

Lin *et al* [128] used both skin marks and skin texture as additional means of distinctiveness to improve the performance of face recognition system. Each face image is factorized into four layers: global appearance, facial organs, skin texture, and facial irregular details. In the third layer, a new skin texture representation has been established based on texton-distribution which comprises three stages: Filtering, Dictionary Building, and Discriminant Learning. In the fourth layer, SIFT operator has been used to extract facial irregularities. However, the accuracy achieved by the skin layer alone was limited. The experiments are conducted on XM2VTS [142] databases.

3.8 Databases of Facial Images

There are several face images databases which have been made available by the research community. Each of these databases is designed to address specific factors covering a wide range of scenarios. Table 3.1 reviews the most popular 2D face images databases.

Table 3.1: Different examples of recent databases of facial images

Database	Year	Resolution	Color
PUT	2009	2048 × 1536	Colour
HRDB	2006	1024 × 768	Colour
FRVT/FRGC	2005	2272 × 1704	Colour
Cas-Peal	2003	360 × 480	Colour
ND HID	2003	1600 × 1200	Colour
Banca	2002-2003	720 × 576	Colour
U of Texas	2002	720 × 480	Gray
KFDB	2002	640 × 480	Colour
Equinox	2002	240 × 320	Gray
CMU-Hyper	2002	640 × 480	Gray
XM2VTS	2002	720 × 576	Colour
CMU PIE	2000	640 × 486	Colour
AR	2000	768 × 576	Colour
FERET	1993,1996,2000	256 × 384	Gray
Yale (B)	1993,1996	640 × 480	Gray

3.8.1 XM2VTS

The XM2VTS database[142] is a multi-modal database consisting of face images, video sequences and speech recordings taken of 295 people of both sexes and different ethnic origins with 4 face images for each person. This data set contains 1180 facial images with size of 720 × 576 pixels. The distance between the two eyes ranges from 84 up to 122 pixels. Over a period of five months, recordings were acquired during four sessions under controlled conditions (uniform illumination, blue background).

Since the data acquisition was carried out over a long period of time, significant variability of appearance of people is presented in the recordings e.g. changes of hair style, facial hair, shape and presence or absence of glasses. About 42.97% (507 images) of the dataset face images contain spectacles which represents another challenge in this dataset because glasses have a strong influence on skin appearance in an image. In addition, areas of skin in some images are affected by facial expressions, shading from other body areas or they are likely to be artificially changed by makeup or other factors.

3.8.2 FRGC

The Face Recognition Grand Challenge (FRGC ver 2.0) [164] is a large database containing high resolution 2D images (2,272 by 1,704 pixels) with controlled or uncontrolled lighting and 3D meshes with shape and texture information. In total 50,000 recordings of 625 subjects is contained in this database.

The FRGC data for the experiments is divided into training and testing sets. The data in the training set contains 12,776 2D still images from 222 subjects where 6,389 images are collected in a controlled environment and the others are acquired in an

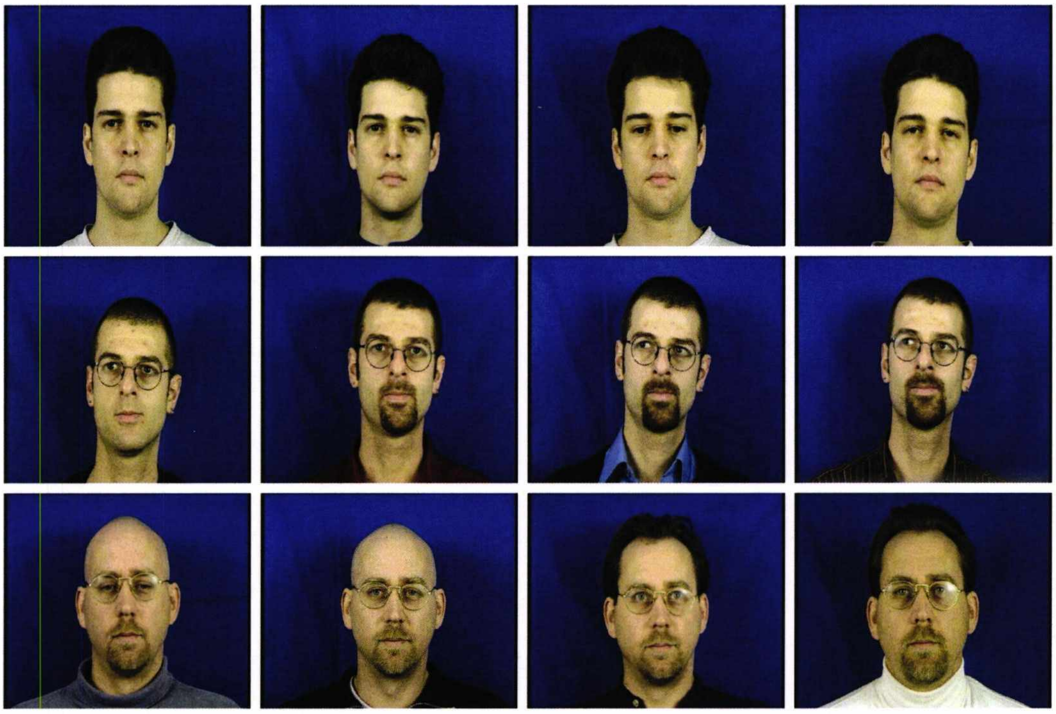


Figure 3.9: Examples of XM2VTS database images

uncontrolled environment. The controlled face images taken in a studio setting are taken in the full frontal pose with two facial expressions under two lighting conditions. The uncontrolled frontal images are captured with two facial expressions in varying illumination, such as hallways or outdoors.

There are six experiments, the two most popular for 2D still face recognition problem are Experiment 1 and Experiment 4. Experiment 1 is designed to measure the recognition performance on controlled still face images from frontal images. In this experiment, 16,028 images from 466 subjects under the controlled environment are used to establish $16,028 \times 16,028$ similarity confusion matrix. Experiment 4 is designed to measure the performance of automatic face recognition on controlled versus uncontrolled frontal face still images. In this experiment, the query set contains of 8,014 uncontrolled still images and the target set consists of 16,028 controlled still images. Thus, the dimension of the similarity confusion matrix is $16,028 \times 8,014$. Figure 3.10 shows the example images of this database.

3.9 Performance Evaluation

During the past decade, many commercial face recognition systems have emerged for various application and different testing protocols have been designed such as, the FERET evaluations [163] and series of FRVT vendor tests 2006 [166] and 2002 [165] which play an important role in the standardization of the testing protocol.

This section will summarise the current understanding by the biometrics community of the best scientific practices for conducting technical performance evaluating. This section is divided into two parts. In the first part, the criteria to evaluate the performance of a biometric verification system is presented. In the second part, the

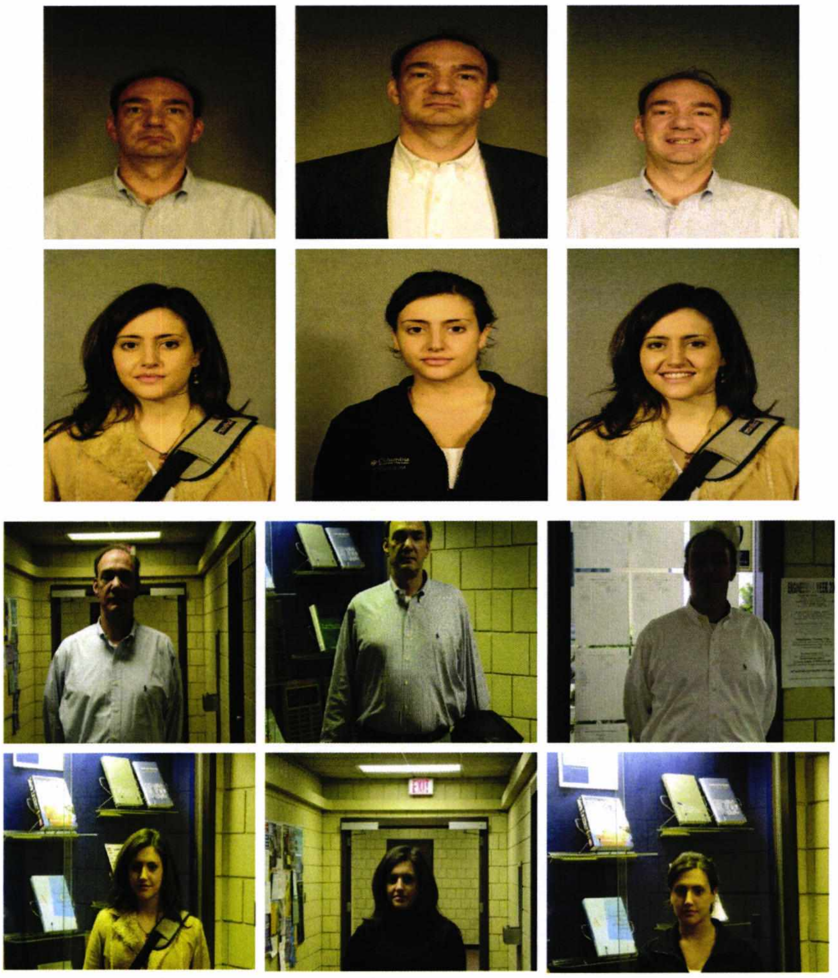


Figure 3.10: Examples of FRGC 2.0 database images (controlled and uncontrolled images)

evaluation of the performance of a biometric identification system is discussed.

3.9.1 Face Verification System

The system of face verification has two different types of error, False Acceptance (FA) and False Rejection (FR). FA is mistaking biometric measurement from two individuals to be from the same person while FR is mistaking two biometric measurement from the same individual to be from two different individuals. Therefore, the performance is evaluated according to False Accept Rate or False Match Rate (FAR or FMR) and False Rejection Rate or False Non-Match Rate (FRR or FNMR):

$$\mathrm{FAR} = \frac{\text{Number of FAs}}{\text{Number of imposter accesses}} \tag{3.8}$$

$$\mathrm{FRR} = \frac{\text{Number of FRs}}{\text{Number of total true client accesses}} \tag{3.9}$$

These two rates are a function of the threshold(t).i.e.for a given threshold (t), there will be two corresponding values FAR(t) and FRR(t). Three criteria (two curves and one value) have been resulted from this pair[138] and can be used to evaluate the performance of a verification system:

Receiver Operating Characteristic (ROC) curve

An ROC curve plots the false acceptance rate (FAR) on the x-axis, against 1 - false rejection rate (FRR) on the y-axis (Figure 3.11(a)). This curve is threshold independent, allowing performance comparison of different systems under similar conditions.

Detection Error Trade-off (DET) curve

The DET curve is often preferred to the ROC curve, in the case of biometric systems. The DET curve plots FAR on the x-axis against FRR on the y-axis using normal deviate scale (Figure 3.11(b)).

Equal Error Rate (EER)

The equal error rate is computed as the point where $FAR(t) = FRR(t)$ (Figure 3.12). Practically, the score distributions are not continuous and a crossover point might not exist. In this case (Figure 3.12 (b),(c)), this value is computed as follows [140]:

$$EER = \begin{cases} \frac{FAR(t_1)+FRR(t_1)}{2} & \text{if } FAR(t_1) - FRR(t_1) \leq FRR(t_2) - FAR(t_2) \\ \frac{FAR(t_2)+FRR(t_2)}{2} & \text{otherwise} \end{cases} \tag{3.10}$$

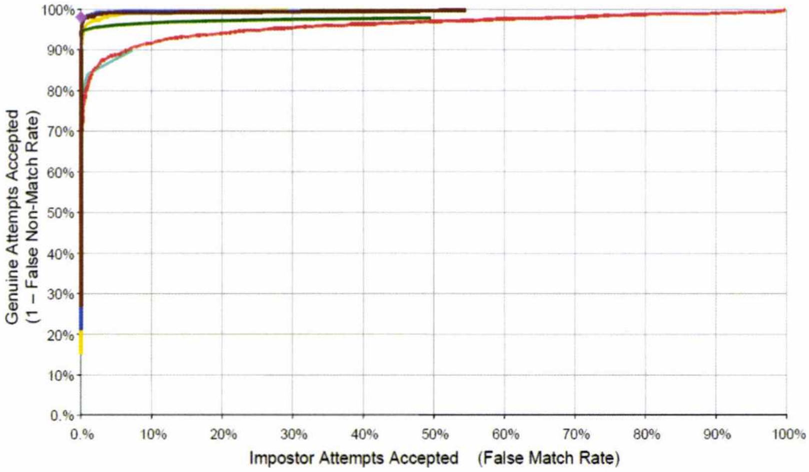
where $t_1 = \max_{t \in S} \{t | FRR(t) \leq FAR(t)\}, \quad t_2 = \min_{t \in S} \{t | FRR(t) \geq FAR(t)\}$ and S is the set of thresholds used to calculate the score distributions.

3.9.2 Face identification

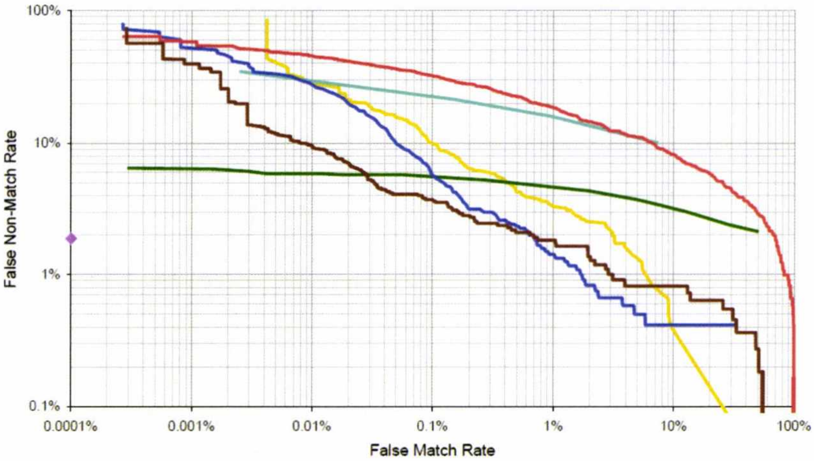
Two criteria for evaluating face recognition systems will be presented:

k-fold cross-validation

Cross-validation method allows using the whole data set for training and testing. In k-fold cross-validation procedure, the relevant dataset is divided randomly into k subsets of approximately equal size. Subsequently one subset is tested using the

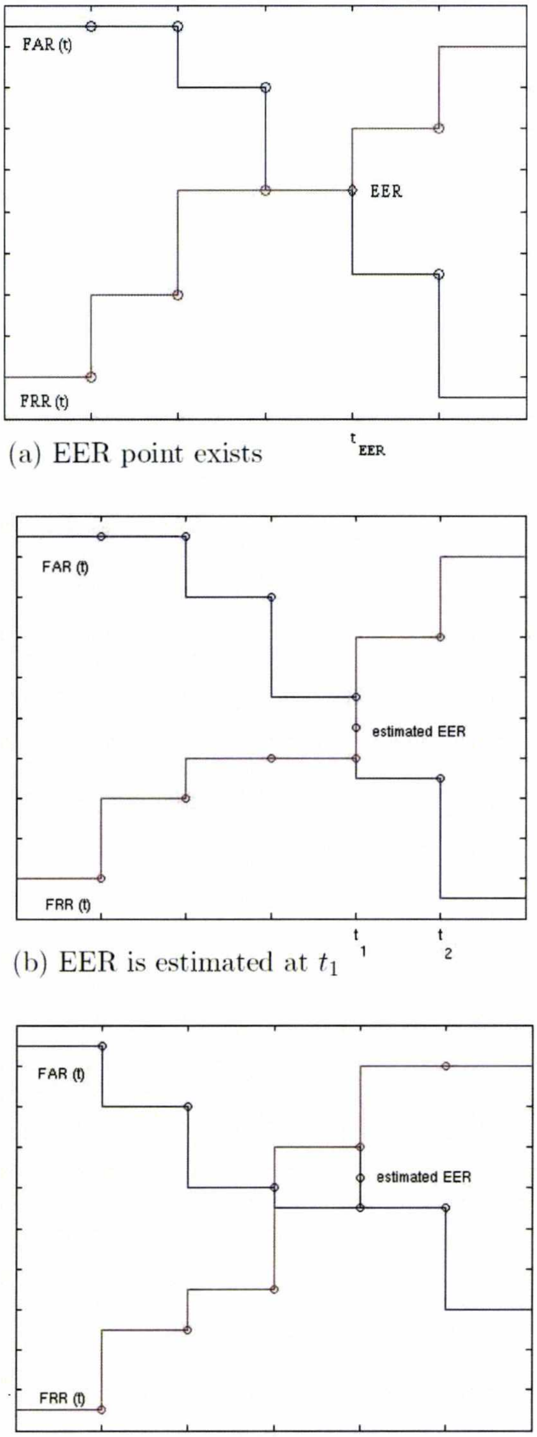


(a)



(b)

Figure 3.11: (a)Example ROC curves, (b)Example DET curves [138]



(a) EER point exists

(b) EER is estimated at t_1

(c) EER is estimated at t_2

Figure 3.12: Estimation EER in DET curves [140]

classifier trained on the remaining $k-1$. Then the average accuracy across all k trials is estimated as the mean accuracy rate.

Rank-one identification

In the testing protocol, the data is divided into two sets of images: the training set and the test set. The performance of identification model is usually evaluated by matching a set of test face images with those in the database. The resulted matches are sorted according to distance measures and the smallest distance obtained is considered to be the top match. Rank-one identification is the percentage of all the correct matches among all comparisons.

Cumulative match score

The main question in the identification model is not always regarding the top match correct, but is the correct in the top k matches?. This allows to know how many images have to be examined to get desired level of performance. The performance is reported as a cumulative match score, which can be plotted on graph. The horizontal axis of the graph is the rank and the vertical axis is the percentage of correct match.

3.10 Conclusion

The chapter covered different topics that related to face recognition. For instance, challenges that met the current face recognition systems were discussed. A comprehensive account of the face detection approaches used in current face recognition systems were presented. Face normalization was briefly discussed. A brief but comprehensive literature review of the current state of the art face recognition approaches was presented. Specifically we have made distinction in the holistic and local approaches for facial features extraction and differentiate them. Different algorithms for partial face recognition were briefly discussed and a survey of skin-based methods for face recognition was presented. Finally, the different ways for evaluating the performance of face recognition system were discussed and the facial databases used along this thesis were described.

Chapter 4

Automatic Extraction of Facial Skin Regions

4.1 Introduction

This chapter presents an automatic method for segmenting four facial regions that will be used for the face recognition techniques reported in later chapters. Automatic extraction of the facial skin regions requires automatic detection of facial features such as eye centres and mouth centre.

Facial features detection is an essential step for the initialization of many face processing techniques like face recognition, facial expression recognition, face pose estimation methods and face tracking. Many face recognition techniques presented in the literature do not clearly state the assumptions made in the system and assume perfect localization by relying upon manual annotations of the facial feature positions.

This chapter is organized as follows: Section 4.2 presents an overview of the existing eye localization methods. Then, Section 4.3 and Section 4.4 respectively present novel schemes, that resulted from this research project, for eyes and mouth localization. The mouth localization algorithm is a modified version of [74] and the eye localization algorithm is a hybrid method which combines geometric and colour information to detect the eyes. Section 4.5 shows some examples of eye and mouth localization results and it also presents a comparison between the results of the proposed algorithm and other published results. Finally, the automatic segmentation of four facial regions: forehead, two cheeks and chin are described in Section 4.6. Section 4.7 concludes this chapter.

4.2 Eye Localization

There are two ways to find eyes in an image, Eye detection and Eye localization. Eye localization when we rely on the bounding box that supplied by a face detector. Eye detection when we search for them in the entire image (no prior face detection) [203].

Each of those approaches has its own advantages or disadvantages. For the first case, by knowing the region of the face, we can locate eyes easier and faster. The rate of false detection decreases; but, the face detection plays a very crucial role. If face detector fails for any reason, then eye detector fails as well. In the second approach, we search for eyes in the whole image without considering a face location. The percentage of true-positive detections with this method can be higher than for the first method, however, the speed is slower, and the false-positive rate is expected to increase.

A great research effort has been devoted to eye detection and localization. This is due to different reasons, among which:

1. It can be considered that both the location of eyes and the inter-ocular distance

between them are relatively constant on the face for most people so the size, the location and the image-plane rotation of face in the image can be normalized by knowing only the position of both eyes [76].

2. It has been shown that eye localization has a considerable impact on face recognition accuracy [176]. Further, the accuracy of some face recognition approaches (such as PCA or LDA) has been shown to decrease with poor localization [219].

3. In addition, the position of other facial landmarks can be estimated using the eye position [115].

4. Eye movements may reveal "interest", or "attention" of a person for human computer interactions as they often reflect a person's emotions [124].

Despite the considerable amount of research effort which has been spent, the problem of automatic eye detection is still far from being fully solved [22]. Two types of images have been used for this problem, namely infrared (IR) and visible-light images. Methods that use IR images require a special technology and equipment that is not always available and have more false detections within an outdoor environment. Methods for visible light images can be broadly classified into three categories: template-based methods [77], appearance-based methods [90] and feature-based methods [61].

In template-based algorithms, a generic eye model is firstly designed based on eye shapes. Cross-correlation is then used for searching eyes in the face images. These methods usually have good accuracy, however, they are normally time-consuming. Feature-based algorithms exploit the characteristics of the eyes such as edge, intensity of iris and the colour distributions of the sclera and the flesh to distinguish the eyes from other objects. While these methods are usually efficient, they lack accuracy when the input image is with low contrast and they fail in case of partially occlusion or rotation of head in depth.

Finally appearance-based methods aim to localize eyes based on their photometric appearance. These algorithms usually need to collect a large amount of training data, representing the eyes of different subjects, under different illumination conditions, and under different face orientations. Once features are extracted, supervised classification algorithms such as a neural network or the support vector machine are used to differentiate between eyes and non-eyes.

4.3 The Proposed Eye Localization algorithm

In the illumination-based method that proposed in [74], a skin-based face detector was first implemented to detect the face before localizing facial features. Later eye/mouth maps were used to localize the candidates facial features and further rules depending on the shape and geometry of the resulted points were then tested in order to achieve to the right locations of eyes and mouth. Shafi and Chung [190] suggested a hybrid method that uses the same technique. Their method is tested on PICS [167] facial images database. The resulted percentage accuracy equals to 93.75%.

Like other eye localization methods that in [134], [51], and [9], Viola-Jones detector is used in the first step in our proposed algorithm to localize the face. Then it estimates the eye windows in the face and then colour-based maps are applied only on the eye windows.

Geometric techniques for determining eyes regions in the facial images are discussed in Section 4.3.1 and the proposed colour-based algorithm for eyes localization is explained in Section 4.3.2.

4.3.1 The Geometric localization of eyes regions

Face shape variability is highly limited by both genetic and biological constraints, and is characterized by a high degree of (approximate) symmetry and (approximate) invariants of face length scales and ratios.

Therefore, it is possible to build model that represents the entire database and extract eye region from the detected faces simply based on geometrical characteristics of faces. The proposed eye regions will be the input for the proposed eye localization algorithm described later in Section 4.3.2.

Let $b = [x, y, r]$ be the vector containing the main parameters of the square resulted from face detector, namely the top left corner of cropping image (x, y) and the length of square width r respectively. Based on the anthropometric model of the human face and depending on initial experiments on XM2VTS, we suggest that the vectors which represent the corresponding right and left eye regions to be b_r and b_l respectively, where:

$$\begin{aligned} br &= [x + 0.20r, y + 0.25r, 0.22r], \\ bl &= [x + 0.58r, y + 0.25r, 0.22r] \end{aligned} \tag{4.1}$$

i.e. $(x + 0.20r, y + 0.25r)$ and $(x + 0.58r, y + 0.25r)$ will be the coordinates of the top left corner of the cropping right eye region and left eye region respectively, $0.22r$ is the width length of the squares that represent these two regions. This works for frontal face images in XM2VTS only. More details are shown in Figure 4.1.

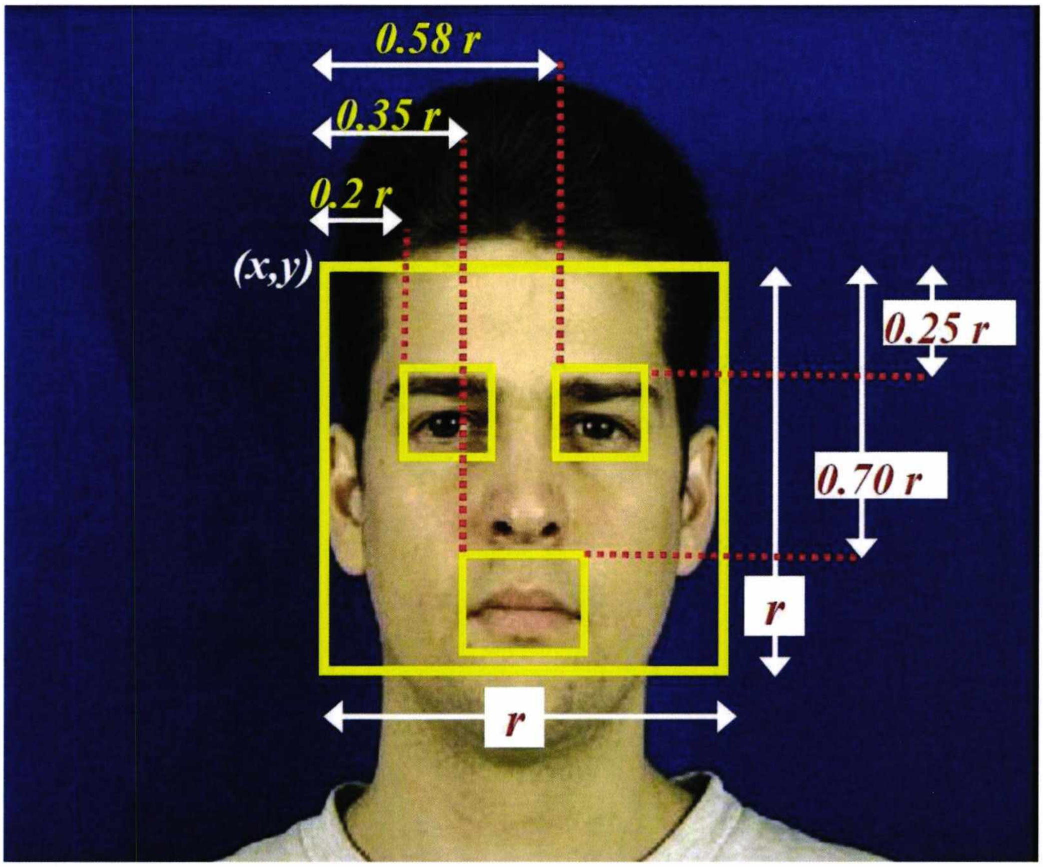


Figure 4.1: Geometric relationships between the locations of eyes and mouth regions

4.3.2 Novel Technique for Localizing Eyes in Colour Images

Figure 4.2 illustrates the steps involved for eye localization in the proposed colour-based algorithm. The proposed algorithm which will be applied on the eye regions that first have been located using the geometric technique that was mentioned in Section (4.3.1), can be summarized in the following steps:

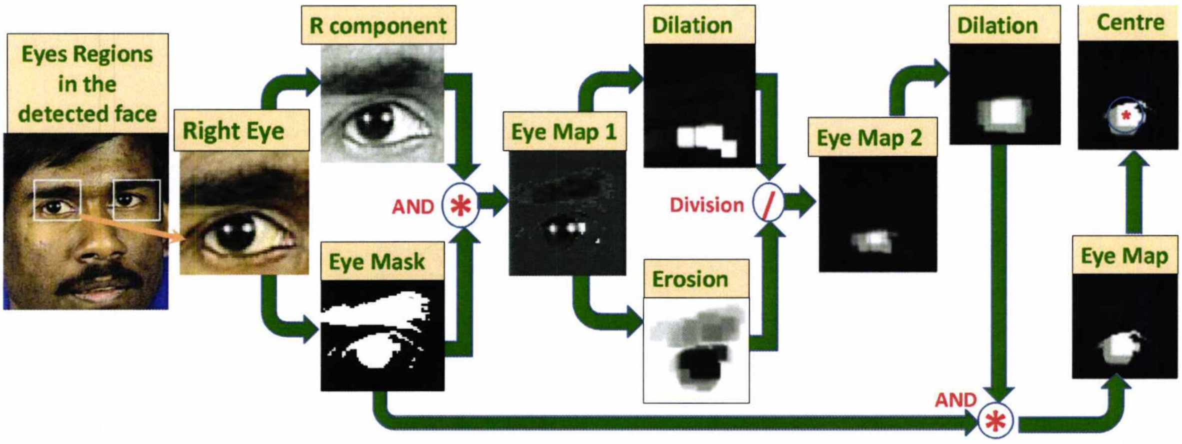


Figure 4.2: The proposed hybrid algorithm for eye localization

- **Step1, Construct** $EyeMap1$: Since the area that covers eye and eyebrow is the complement of skin in the eye region, we implement an RGB skin detection algorithm [106] to build mask of them, $EyeMask$.

  The $EyeMask$ is then combined with the R component (first component) of the eye image by the multiplication operation i.e.

$$EyeMap1 = (EyeMask) * (R) \tag{4.2}$$

Like in [74], we fill the background of the $EyeMap1$ with the mean value of R

in skin region in order to smooth the noisy boundary of detected skin areas.

- **Step2, Construct** $EyeMap2$: We use greyscale dilation $\oplus$ and erosion $\otimes$ with structuring element g to construct the $EyeMap2$ from $EyeMap1$, as described in the following equation:

$$EyeMap2 = (R(x,y) \oplus g(x,y))/(R(x,y) \otimes g(x,y) + 1) \tag{4.3}$$

- **Step3, Construct** $EyeMap$: The $EyeMap2$ is dilated, and then combined with $EyeMask$ by the multiplication operation to construct $EyeMap$.
- **Step 4: Iris centre**: The resulting $EyeMap$ is dilated, and normalized to brighten the iris and suppress other areas. The blob centre represents the iris centre. The performance of this algorithm will be discussed in Section 4.5.

4.4 The Proposed Mouth Localization algorithm

Based on initial experiments on XM2VTS database, we suggest that the vectors which represent the mouth region to be:

$$bm = [x + 0.20r, y + 0.25r, 0.22r] \tag{4.4}$$

where (x, y) and r are parameters that were defined in Section 4.3.1.

A map-based scheme, as illustrated in Section 3.3.4, for locating the mouth was presented in [74]. An analysis of the chrominance components in $YCbCr$ colour space indicates that the mouth contains a stronger red component and weaker blue component than other facial regions. Hence, the chrominance component C_r is greater than C_b in the mouth region. The mouth also has a relatively low response in the C_r/C_b feature, but it has a high response in C_r^2. Therefore, the mouth map is constructed as follows:

$$\begin{aligned} MouthMap &= C_r^2 \cdot \ (C_r^2 - \eta \cdot C_r/C_b)^2, \\ \eta &= 0.95 \cdot \frac{\frac{1}{n}\sum_{(x,y)\in FG} C_r(x,y)^2}{\frac{1}{n}\sum_{(x,y)\in FG} C_r(x,y)/C_b(x,y)} \end{aligned} \tag{4.5}$$

where both C_r^2 and C_r/C_b are normalized to the range [0; 255], and n is the number of pixels within the face mask, FG. The parameter η is estimated as a ratio of the average C_r^2 to the average Cr/Cb.

Since usually multiple blobs are produced by this scheme, we add , as illustrated in Figure 4.3, further geometric restrictions in the mouth region to localize the mouth centre.

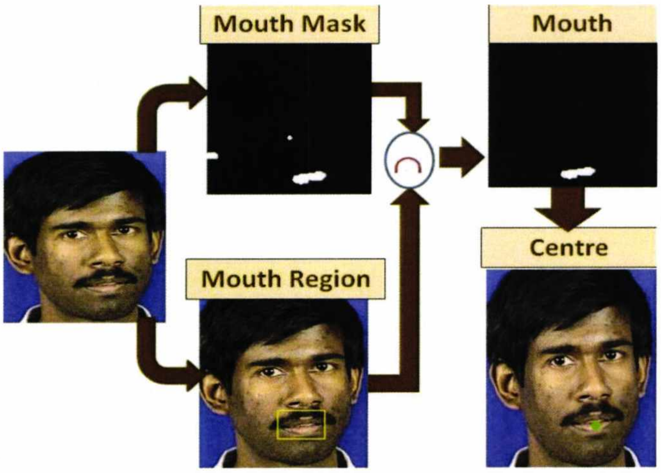


Figure 4.3: The proposed Map-based algorithm for localizing mouth centre

4.5 Localizations Results on XM2VTS

Table 4.1 lists the total number of the correctly detected centres of right/left eyes and mouth centres. Eyes localizations results are discussed in Section 4.5.1 and results of mouth localization are discussed in Section 4.5.2. Section 4.5.3 presents a comparison between the results of the proposed method and other published results.

Table 4.1: Localizations results

	Correct Localization %
Right Eye centre Localization	98.90
Left Eye centre Localization	98.47
Mouth centre Localization	99.58

4.5.1 Eye Localization results

Precise eye center detection is a challenging task. An eye can be open or close, and when open the iris may be pointing at any direction. Another reason behind the difficulty is the high variability of eyes. They may, for ethnicity and race reasons, have very distinct shape, pupil size, and colors. Wearing glasses is another factor which has a strong influence on how eyes are seen in an image because 507 individuals (42.97%) wear glasses in this dataset.

In addition, sometimes there are external drawbacks like cast shadows, and lighting. We consider the detection to be correct if the estimated position of eye centre is located anywhere in the iris. Figure 4.4 shows examples of correct localizations and

Figure 4.5 shows incorrect localization of eyes using the proposed algorithm.

4.5.2 Mouth Localization results

We consider the mouth centre detection to be correct if the estimated position locates in any place on the lips. The proposed mouth detection method produces good results even in challenging cases such as individuals with dense beards, red skin, lighting variation, different head positions and wide open mouths showing teeth. Only five images in the XM2VTS database had lips wrongly detected. These are shown in the last row in Figure 4.6.

4.5.3 Comparison with other Published Results

The method has been compared with other published results that were tested on the same database for the eye center localization task. Unfortunately, no mouth centre detection method tested on the XM2VTS database has been found.

In order to evaluate the performance of the proposed algorithm, the criterion of [88] is used. The criterion is a relative error measure based on the distances between the corrected and the estimated eye positions. It is defined as:

$$d_{eye} = \frac{max(d_r, d_l)}{||C_l - C_r||} \tag{4.6}$$

where C_r and C_l are the correct eye centre positions and d_r, d_l distances between the detected eye centres and the correct ones, $||.||$ indicates the distance between two points [88].

If the relative error is less than threshold T, the detection is considered to be

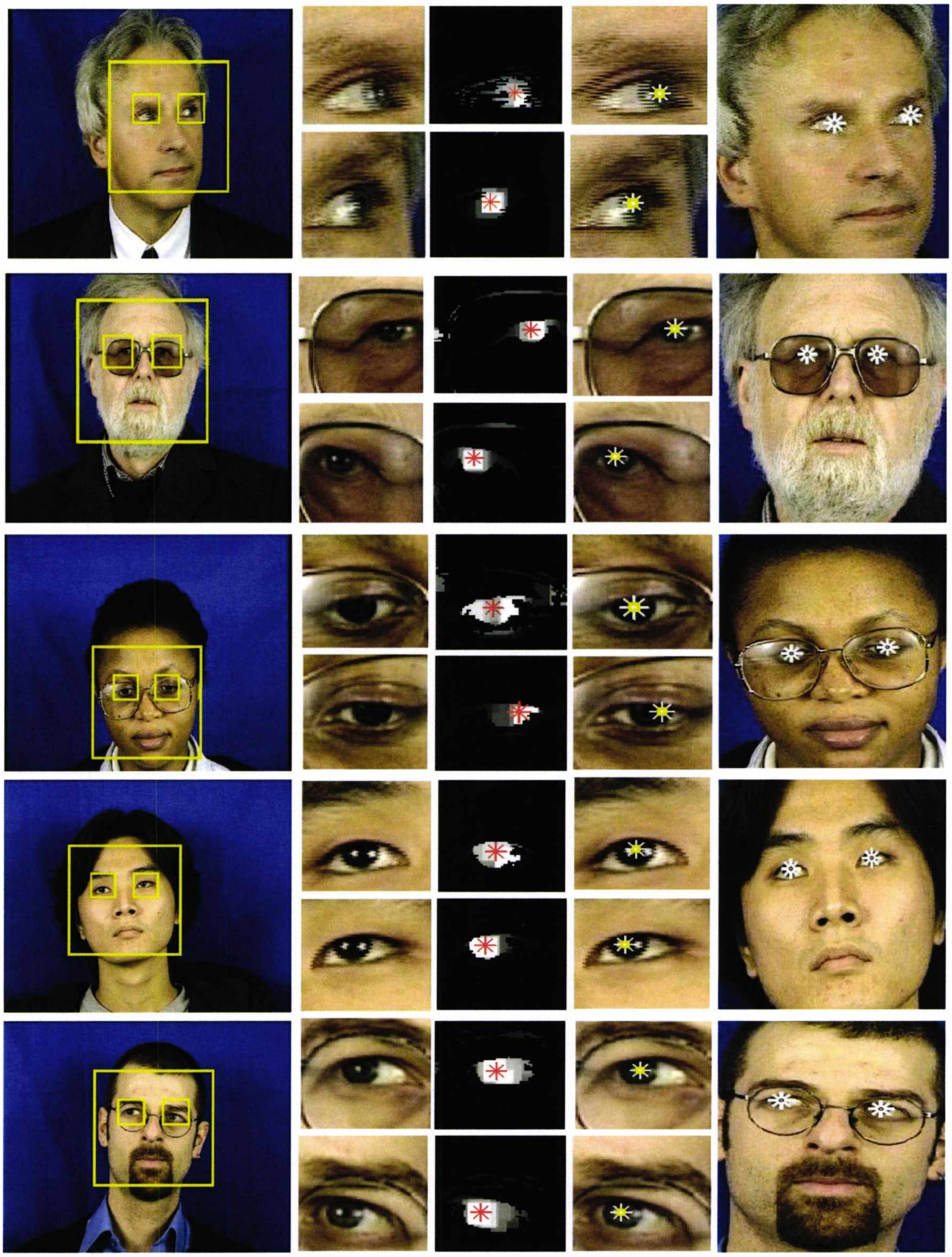


Figure 4.4: Examples of correct localization using the proposed algorithm

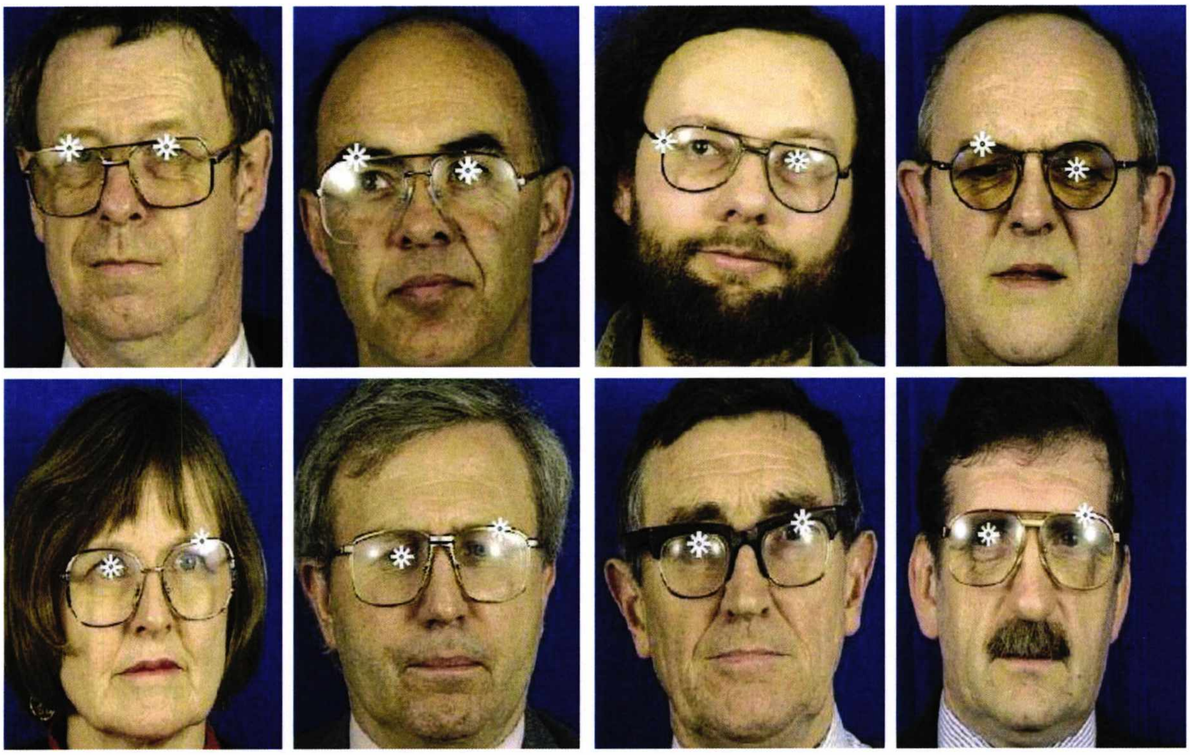


Figure 4.5: Examples of incorrect localization using the proposed algorithm

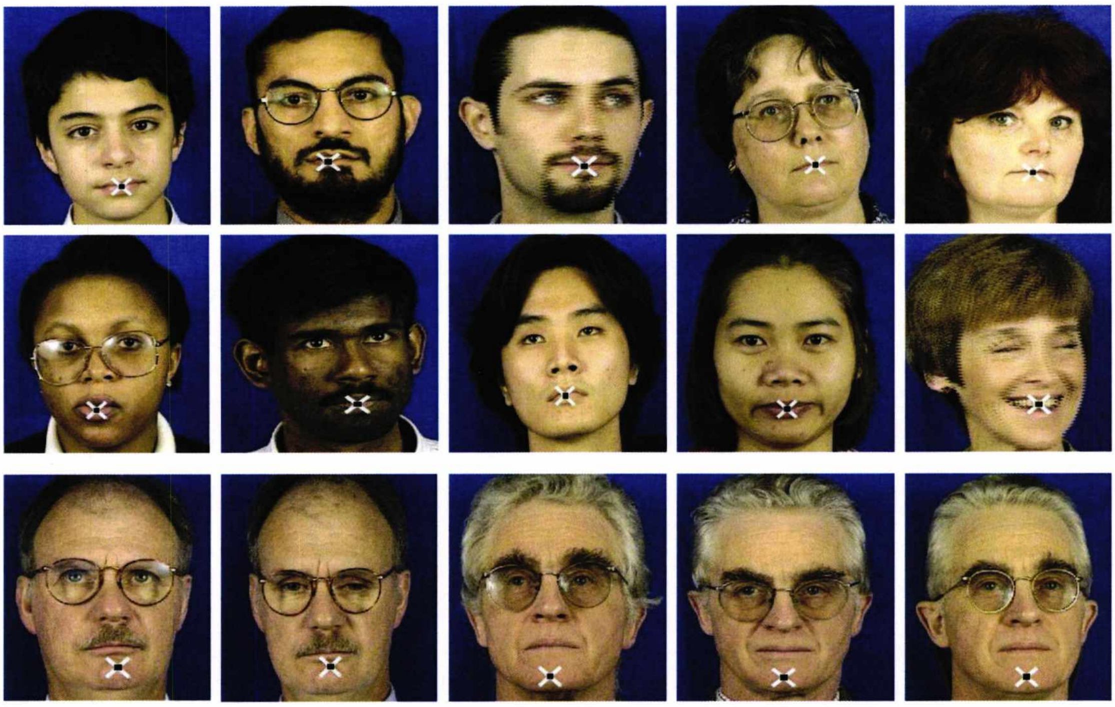


Figure 4.6: Examples of correct and incorrect detection using the proposed algorithm

correct. As explained in [88], $d_{eye} = 0.25$ means the maximum one of two distances roughly equals half an eye width. Table 4.2 lists the performance of the proposed method and other methods for detecting eye centre point for $T = 0.1$ and $T = 0.25$. The experimental results show that the proposed method is significantly superior to other methods for smaller value of the threshold. Figure 4.7 depicts the success rates for various values of the threshold using the proposed algorithm.

Table 4.2: Comparison of the proposed method performance with other published methods

Database	$T = 0.1$	$T = 0.25$
Method in [88]	93.00%	98.40%
Method in [36]	93.00%	99.00%
Method in [66]	-	97.90%
The proposed method	**96.95**%	98.00%

4.6 Segmentation of Facial Skin Regions

Face shape variability is highly limited by both genetic and biological constraints, and is characterized by a high degree of (approximate) symmetry and (approximate) invariants of face length scales and ratios. Therefore, we refer to the anthropometric model of the human face in order to extract facial skin regions correctly.

Once the positions (x_r, y_r) and (x_l, y_l) of the right and left eye respectively have

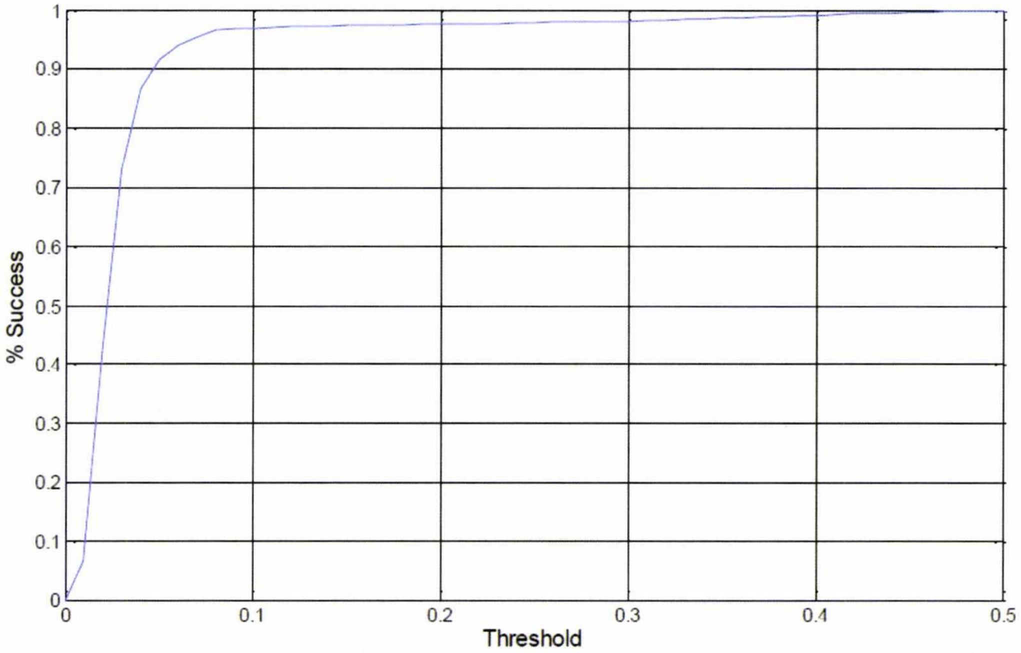


Figure 4.7: Eye centre localization for various threshold T

been determined, the distance between them is calculated as follows:

$$d = \sqrt{(x_r - x_l)^2 + (y_r - y_l)^2} \tag{4.7}$$

This distance serves as the principal parameter for measuring the centre location and size of the other facial feature regions. For instance, face is represented in [192] by an ellipse that is placed at $0.4d$ below the midpoint of the two eyes. The length of the major axis is given as $3d$ and of the minor axis is given as $2d$. Figure 4.8 shows an example of geometric face model.

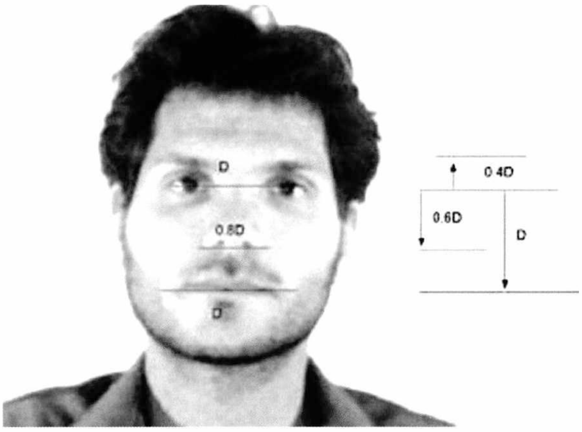


Figure 4.8: The geometric face model that was used in [192]

Figure 4.9 shows the distances (anthropometric measurements) $D1$ to $D4$ which are used for building human face model. Their descriptions as follows:

D1: the distance between eye centers.

D2: the distance between right/left eye center and right/left eyebrow center.

D3: the distance between midpoint of eye centers and nose tip.

D4: the distance between midpoint of eye centers and mouth center.

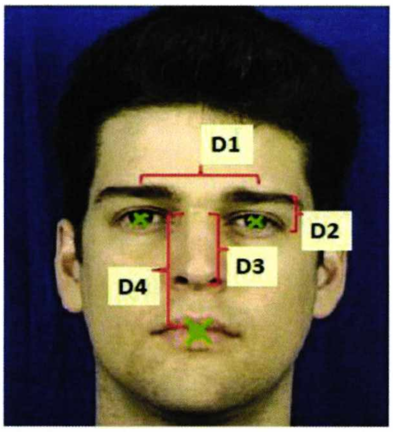


Figure 4.9: Distances that are used in [192], [198] for building the anthropometric model of human face

The facial proportions of two different geometric models ([192], [198]) are listed in Table 4.3.

Table 4.3: Two examples of the anthropometric of the model human face

	[198]	[192]
D2/D1	0.33	0.4
D3/D1	0.6	0.6
D4/D1	1.1	1

The rectangular areas for confining the facial skin regions are also approximated using distance between two eyes as the measurement criteria. We will focus on four prime regions of interest corresponding to forehead, right cheek, left cheek and chin because they present regions of the face that are clearly visible in a frontal-view face image. These areas are kept large enough to cover each of the facial skin regions completely. The proposed dimensions of forehead, right/left cheek and chin regions are: $d \times 0.5d$, $0.5d \times 0.5d$, $0.5d \times 0.5d$, $0.5d \times 0.5d$ respectively.

Figure 4.10 shows these windows and their cropping points are given by:

$A = (x_c - 0.5d,\ y_c - 0.45d)$,

$B = (x_r - 0.44d,\ \frac{y_r + y_m}{2} - 0.06d)$,

$C = (x_l - 0.06d,\ \frac{y_l + y_m}{2} - 0.06d)$ and

$D = (x_m - \frac{d}{4},\ y_m + 0.2d)$,

where (x_c, y_c), (x_r, y_r), (x_l, y_l) and (x_m, y_m) represent the coordinates, after geometric normalization, of midpoint of eye centers, right eye center, left eye center and mouth center respectively.

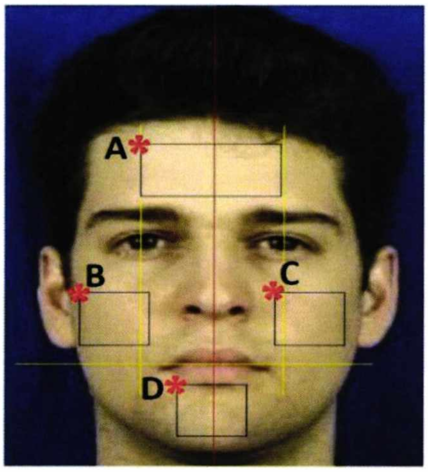


Figure 4.10: The proposed locations of the four regions of interest: Forehead, Right Cheek, Left Cheek and Chin

4.7 Conclusion

An automatic method for localizing eyes in colour images was presented in this chapter. This method has been compared with other published results. Experimental results on the XM2VTS database shows that the proposed method is significantly superior to some of other methods for smaller values of the error rate(threshold). Based on the position of eye and mouth centres, four locations of four skin regions in the face image have been suggested. These regions will be used in the next chapters.

Chapter 5

Facial Skin Texture as a Source of Biometric Information

5.1 Introduction

This chapter focuses on the extraction of usable forehead regions from facial images and their initial pre-processing to provide useful information for the task of biometric person identification/verification.

The proposed algorithm automatically segments the forehead region based on the locations of eyes and divides it into non-overlapping patches. Features were firstly extracted from these patches and then concatenated to form a feature vector that represents the forehead region. $k-NN$ was used in the classification process. The features were extracted separately using Gabor filter and LBP operator and their classification performance were compared. The experiments of the proposed method were carried out using the XM2VTS database.

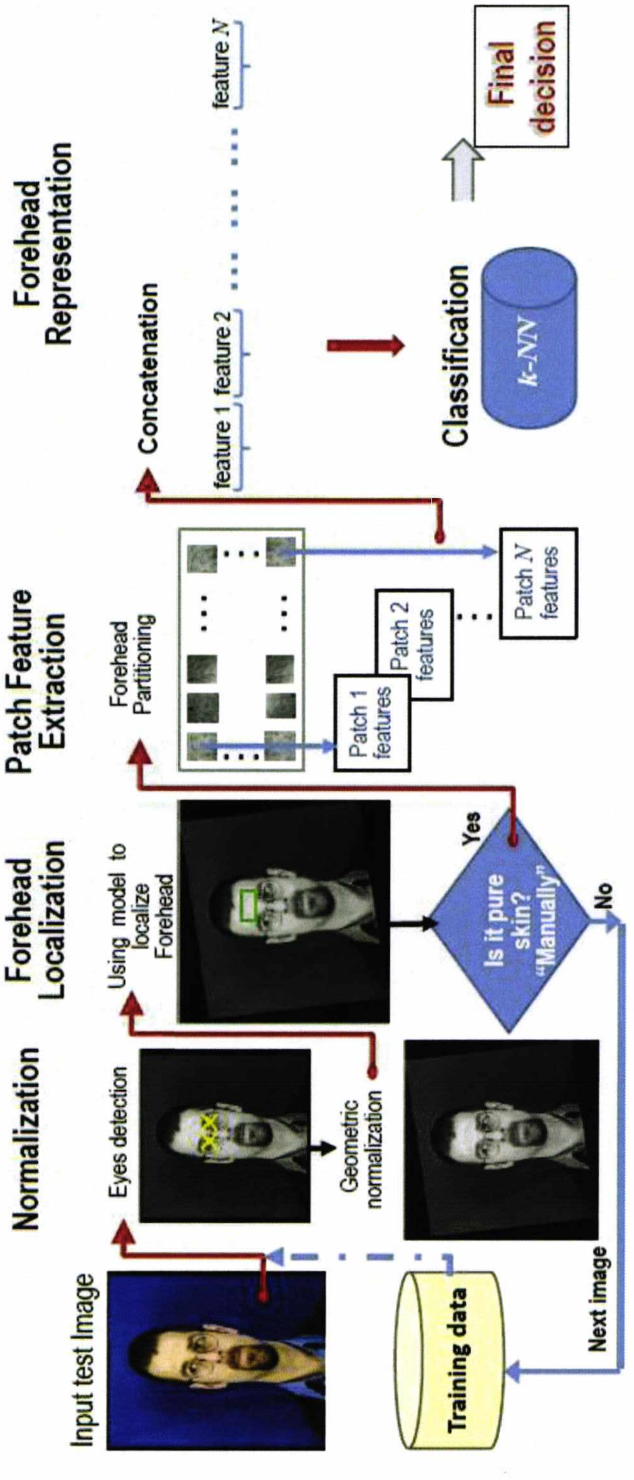


Figure 5.1: System block diagram

The chapter is organized as follows. Section 5.2 describes the the proposed algorithm. Section 5.3 provides the experimental results and performance comparisons. Finally, conclusions are given in Section 5.4

5.2 Overview of the Proposed Algorithm

The overall block-diagram of the proposed system is shown in Figure 5.1. This section will describe the various stages of the proposed algorithm.

5.2.1 Face Normalization

The first step in the processing chain is to extract a normalized forehead region from face images. Since it is assumed that all the subjects are roughly placed at the same distance from the camera, no scale adjustments are necessary. Only rotation of the face, as illustrated in Section 3.3, and then cropping of the forehead region are performed. Figure 5.2 shows the face before and after applying face normalization.

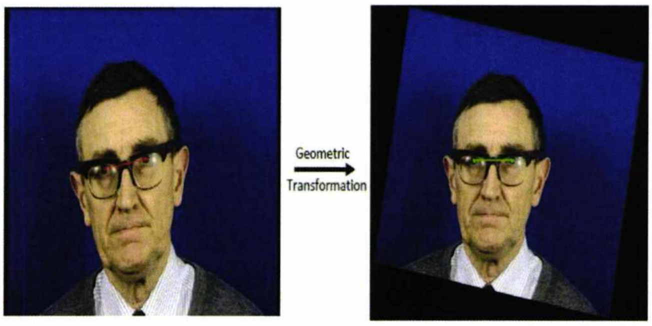


Figure 5.2: Face normalization

5.2.2 Forehead localization

Based on the geometric model of the face in [192], the vertical distance between eyes and eyebrows may be set to $0.4d$ where d is the interocular distance. Therefore, the proposed forehead windows are rectangles of size $d \times 0.5d$ above the eyebrows region as shown in Figure 5.3.

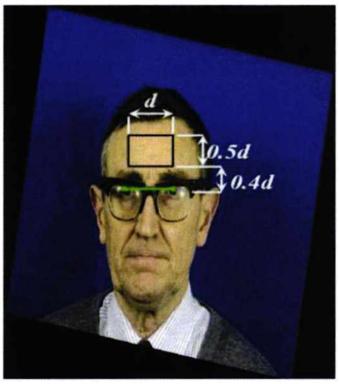


Figure 5.3: Forehead region localization

5.2.3 Forehead Partitioning

In the proposed algorithm, the forehead image is treated a collection of smaller sub-images, referred to here as patches. For the purpose of feature extraction it is necessary to establish the optimum patch size for texture analysis. This could be the whole of the extracted forehead. Alternatively a number of smaller patches can be chosen inside the forehead region for extracting features which can then be concatenated to form longer feature vectors.

Subdividing the pure skin forehead region into non-overlapping patches (of equal sizes) is explored with varying partitionings : 1×2, 2×2, 2×4, 2×8, 4×4 and

4 × 8 as shown in Figure 5.4. The 1 × 1 subdivision indicates the whole forehead being used for analysis without any partitioning.

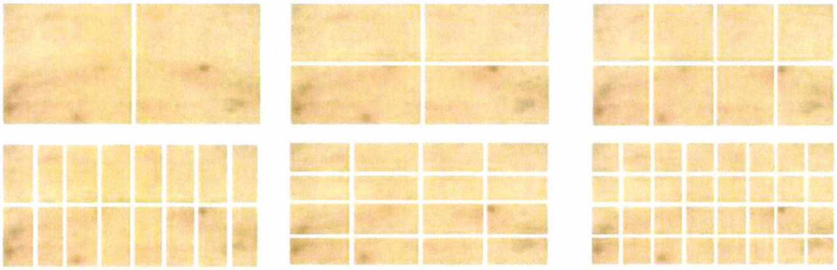


Figure 5.4: Partitioning pure skin forehead region

5.2.4 Forehead Representation and Classification

Each forehead region is decomposed into equal non-overlapping square patches and each patch is then analyzed to extract separately the Gabor and LBP-based features that will be described in the following subsections. Finally, the feature vectors extracted from patches are concatenated to form a *forehead feature vector.*

Forehead Representation using Gabor-based feature

To obtain Gabor-based features of a patch image I, it is convolved with a bank of scale and orientation selective Gabor filters:

$$r_{uv}(x, y;\ f_u, \theta_v) = \psi(x, y;\ f_u, \theta_v) * I(x, y) \tag{5.1}$$

where u,v indicate the filter scale and orientation respectively.

Responses of the Gabor filters r_{uv} in Equation (5.1) are computed for several frequencies f_u and orientations θ_v. The values of the frequency f_u and the orientation θ_v were discussed in Section (2.4.3). All means μ_{uv} and standard deviations σ_{uv} of the magnitude $|r_{uv}|$ are computed as follows:

$$\mu_{uv} = \sum_x \sum_y |r_{uv}|, \text{ and } \sigma_{uv} = \sqrt{\sum_x \sum_y (|r_{uv}| - \mu_{uv})^2} \tag{5.2}$$

A feature vector $\overline{\mathbf{F}}$ is finally constructed using the collections of means and standard derivations. This is denoted as:

$$\overline{\mathbf{F}} = [\mu_{00}\, \sigma_{00}\, \mu_{01} \ldots \mu_{m-1n-1}\, \sigma_{m-1n-1}] \tag{5.3}$$

where m is the total number of scales and n is the total number of orientations to be used in the filter bank.

We report, in Section 5.3, on an investigation to establish the optimal parameters used for the class of images that are to be classified with these filters in our work. It is observed that the patches that are extracted from different forehead images are not of the same size because they are depending on the inter-distance of the two eyes which is not fixed for all database images.

However, the Gabor-based features were not affected by this variation because they consist of means and standard deviations of the modulus-of-convolution images obtained by convolving the patch image with a bank of Gabor filters of different scales and orientations. Therefore, the dimension of Gabor-based features depends only on the number of scales and number of orientations which are chosen after some tests.

As a result, every forehead image can be represented using Gabor wavelet filter by feature vectors that have $2lmn$ elements - where m is the total number of scales and n is the total number of orientations in the filter bank and l is the total number of patches.

Forehead Representation using LBP-based feature

The dimension of the standard LBP feature vector that is derived from each patch is fixed at 256 elements irrespective of the patch size. Thus, every forehead is represented by $256l$ elements using LBP(8,1) or LBP(8,2) operators, where l indicates the number of patches in the forehead. The length of the LBP-based feature is reduced by using uniform and rotation invariant operators, so each patch image is represented using LBP(8,1,$u2$) and LBP(8,1,ri) operators by feature vectors that have 59 and 36 elements respectively instead of 256. Thus, every forehead in this case is represented by a vector with dimensions of $59l$ and $36l$ respectively.

Classification

A k-Nearest-Neighbour ($k - NN$) classifier with $k = 1$ is used as a common platform to compare the different feature extraction methods.

5.3 Experimental Results

Matlab is used as our coding platform. We will discuss the results of the proposed algorithm which is tested on the XM2VTS database in two different scenarios: Identification scenario and verification scenario.

5.3.1 Experiment 1: Identification Scenario

This experiment has been conducted on a subset of the XM2VTS database to evaluate the performance of the Gabor-based features for skin texture classification. The data used consists of pure skin forehead region as labeled **manually**. The number of these images in this group is 336 pure skin foreheads which came from 84 different persons. Only three partitionings for the forehead region were evaluated. These were: 1×2, 2×4 and 4×8 with patch size of $0.5d \times 0.5d$, $0.25d \times 0.25d$ and of $0.125d \times 0.125d$ respectively. Table 5.1 lists the recognition accuracy achieved for different parameter settings.

The results indicate that the partitioning with larger analysis windows significantly improve by increasing the number of orientations/scales. For example, when the number of scales/orientations has changed from 4/4 to 12/12 respectively, the classification accuracy improves only from 60.12% to 64.29% for the 4×8 partitioning but from 58.33% to 67.86% for the 2×4 partitioning and from 35.12% to 56.55% for 1×2.

The best recognition rate of 68.16% has been obtained at the least orientation value of 8 and least scale value of 16. Additionally, it is noted that increasing the number of scales and orientations of the Gabor filter bank will increase the complexity of the system and the computational time, and will not always result in an increase in the classification accuracy.

5.3.2 Experiment 2: Verification Scenario

The aim for conducting this experiment is threefold: (i) to determine the appropriate number of scales and orientations that can be used for Gabor features. (ii) to

Table 5.1: Classification accuracy using Gabor features

Orientation	Patch	Scales				
		4	6	8	12	16
4	1 × 2	35.12%	37.20%	38.99%	44.05%	45.54%
	2 × 4	58.33%	62.80%	65.18%	65.77%	66.96%
	4 × 8	60.12%	62.20%	61.91%	62.50%	62.20%
6	1 × 2	36.61%	42.26%	43.75%	49.11%	50.59%
	2 × 4	61.01%	63.09%	65.48%	66.07%	67.56%
	4 × 8	60.12%	62.80%	63.39%	63.39%	63.39%
8	1 × 2	42.26%	45.83%	48.81%	53.57%	54.76%
	2 × 4	61.61%	65.48%	66.96%	67.86%	***68.16***%
	4 × 8	60.42%	63.09%	63.99%	63.99%	63.69%
10	1 × 2	41.67%	48.51%	51.49%	55.06%	56.84%
	2 × 4	62.50%	65.48%	66.67%	67.86%	67.86%
	4 × 8	61.31%	62.20%	63.10%	64.29%	63.99%
12	1 × 2	44.34%	49.40%	53.57%	56.55%	57.14%
	2 × 4	62.50%	65.18%	67.56%	67.86%	66.96%
	4 × 8	61.91%	63.09%	63.69%	64.29%	63.99%
16	1 × 2	46.73%	49.40%	55.06%	57.44%	58.04%
	2 × 4	62.50%	65.18%	67.56%	67.26%	67.86%
	4 × 8	61.91%	63.39%	63.99%	64.29%	63.99%

investigate the different LBP operators (standard, Uniform '$u2$', Rotation invariant 'ri') using different radius and sample points, and (iii) to decide the patch size for achieving the best performance.

The experiment has been conducted on the whole XM2VTS database (1180): 460 forehead images that came from 115 different persons have been 'manually' determined as faces with pure skin forehead regions. Half of the entire dataset is used as the training set and the remaining represents the test set. Table 5.2 lists the Equal Error Rates (EER) obtained for both Gabor and LBP approaches for varied patches size.

This table suggests that the achievable performance with Gabor features is better than with LBP features, showing that the best EER of 0.065 for Gabor features has been obtained using 6 scales, 6 orientations with a 4 × 4 partitioning. If no patch partitioning is used, the resulting EER is significantly higher at 0.213.

It also shows that the performance of the standard LBP approach using 8 sample points with radius equal to 2 outperforms other LBP operators. If no patch partitioning is used an EER of 0.165 is achieved with this method. Features using the forehead partitioning of 2 × 4 and 4 × 4, achieved the best results for LBP and Gabor approaches respectively for this dataset. It is clear that the performance of LBP approach does not significantly improve using rotation invariant features because all images were normalized in the pre-processing stage.

Their DET curves are presented in Figure 5.5. Since only one partitioning scheme should be used, the error that is produced from Gabor and LBP-based features at partitioning 2 × 4 is less that the error that is produced from them at 4 × 4 so this experiment suggest that the 2 × 4 partitioning is an appropriate choice for the partitioning scheme. The best EER of 0.066 for Gabor features with a 2 × 4 partitioning has been obtained using 8 scales and 12 orientations.

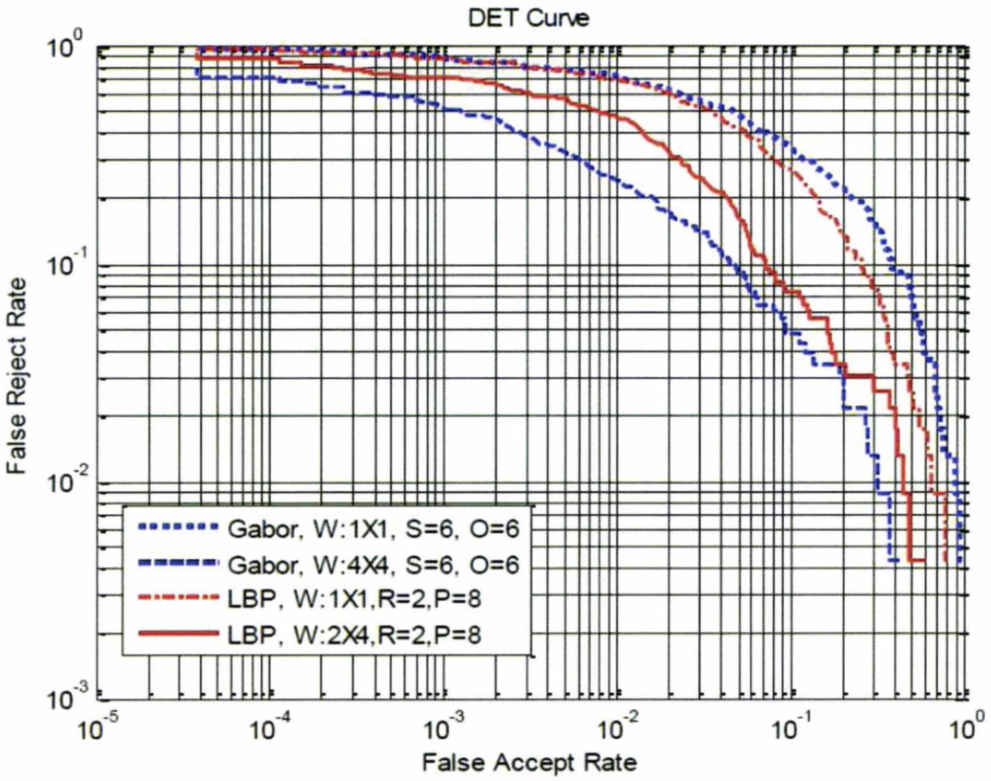


Figure 5.5: DET curves for Gabor and LBP approaches; W indicates partitioning scheme, Gabor parameters are: S: No of Scales and O: No of Orientations, LBP parameters are: R: Radius and P: Sample Points

Table 5.2: EER using Gabor features and different LBP operators

Approach		Partitioning						
		1 × 1	1 × 2	2 × 2	2 × 4	2 × 8	4 × 4	4 × 8
Gabor	Scale=4/Ori=6	2.13e-01	1.83e-01	1.48e-01	1.30e-01	1.13e-01	1.00e-01	1.01e-01
	4/8	2.14e-01	1.70e-01	1.38e-01	1.13e-01	1.09e-01	9.13e-02	9.90e-02
	4/12	2.01e-01	1.52e-01	1.13e-01	1.01e-01	1.08e-01	7.87e-02	9.97e-02
	6/6	2.13e-01	1.75e-01	1.27e-01	1.09e-01	8.37e-02	6.52e-02	***6.52e-02***
	6/8	2.04e-01	1.57e-01	1.14e-01	9.57e-02	8.26e-02	6.52e-02	6.96e-02
	6/12	1.83e-01	1.26e-01	1.00e-01	8.27e-02	8.26e-02	6.52e-02	7.02e-02
	8/6	1.96e-01	1.53e-01	1.01e-01	8.26e-02	7.39e-02	7.44e-02	8.79e-02
	8/8	1.79e-01	1.35e-01	9.57e-02	7.49e-02	6.86e-02	7.01e-02	9.57e-02
	8/12	1.73e-01	1.26e-01	9.06e-02	6.59e-02	7.36e-02	6.62e-02	9.57e-02
LBP	1,8	1.61e-01	1.44e-01	1.09e-01	9.57e-02	1.00e-01	9.57e-02	1.23e-01
	2,8	1.65e-01	1.38e-01	1.13e-01	***8.26e-02***	1.10e-01	1.17e-01	1.87e-01
	$1,8^{u2}$	1.69e-01	1.55e-01	1.35e-01	1.05e-01	1.13e-01	1.09e-01	1.17e-01
	$1,8^{ri}$	2.09e-01	2.13e-01	2.00e-01	1.91e-01	2.31e-01	2.13e-01	2.40e-01

5.4 Conclusion

We have presented a novel technique for extracting biometric information from the forehead region. Pure skin forehead region are manually labeled and the performance of Gabor-based and LBP-based features in classifying forehead regions that contain only pure skin are then compared. The effects of changing the parameters of the Gabor filter-bank and LBP approaches are investigated. The results indicate that the forehead region alone provides useful biometric information for person recognition.

Chapter 6

Explicit Integration of Identity Information from Skin Regions to Improve Face Recognition

6.1 Introduction

Most of the global-based face recognition systems use a single feature vector for encoding the whole face image so resizing the facial images into low resolution images is often first step [121]. As expected, micro-structures of facial skin texture may be lost and subtle details are not explicitly modeled because only the global appearance of the faces are utilized.

Therefore, the main objective of this chapter is to combine the conventional face recognition systems with skin texture features (from the region which has the most useful information), to compensate for the information which has been lost in the dimensionality reduction stage. Thus we have to compare different facial regions in order to get the region which has the most important biometric information.

Biometric system using facial skin texture, that was developed in the previous chapter, has some drawbacks. For instance, facial skin regions may be obscured by hair, facial hair or spectacles. In that chapter, manual annotation was used to label the forehead region that contains only skin texture. This chapter reports on the development of a novel Skin/Non-Skin classifier based on Support Vector Machines (SVM), in order to automatically exclude non-skin areas in facial images.

The chapter consists of seven sections. The partitioning scheme is presented in Section 6.2. The SVM approach for Skin/Non-Skin patch classifier is discussed in Section 6.3. Section 6.4 presents a comparison between the performance of different facial skin texture regions. The proposed approach for the fusion of scores from skin and whole face channels of identity information is described in Section 6.5. The results obtained are reported and discussed and compared with other methods in Section 6.6. Section 6.7 provides conclusions for this chapter.

6.2 Partitioning of Facial Skin Regions

The preliminary experiment, in Section 5.3.2, suggested that the patch of size $0.25d \times 0.25d$ is a good choice for partitioning scheme, where d is the interocular distance, so the forehead will be divided into eight patches while other regions, right cheek, left cheek and chin will be divided into four patches as illustrated in Figure 6.1. This novel decomposition approach is proposed to eliminate, using SVM-based classifier, patches that may be obscured by hair, facial hair or spectacles and take into account only the pure skin patches.

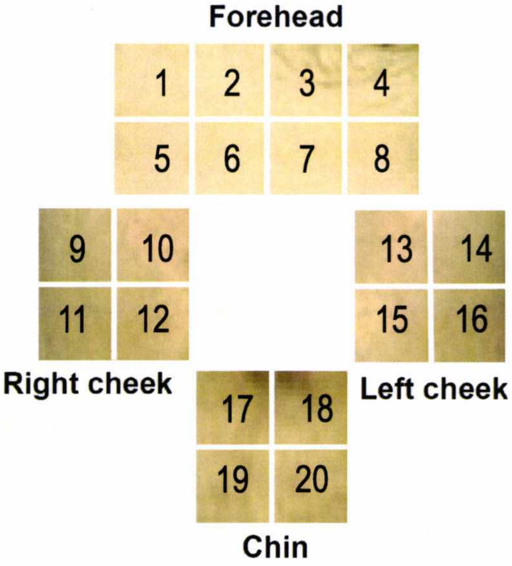


Figure 6.1: The partitioning of facial skin patches in the four regions

6.3 Skin/Non-Skin Patch Classifier based on SVM

An SVM classifier is considered as one of the most suitable approaches for two-class problem, in order to be processed by an SVM classifier, these patches must be represented as labeled feature vectors. The labels in this case would be numbers 0 and 1 that represent non-skin and skin patches respectively. The patch is labeled as non-skin if it is covered partially or completely by facial hair, hair, spectacles,etc. Otherwise it is considered as a skin patch.

Figure 6.2 illustrates the patch classification scheme which aims to automatically distinguish pure "skin" patches from "non-skin" patches. The feature vector and model building will be discussed in the following sections.

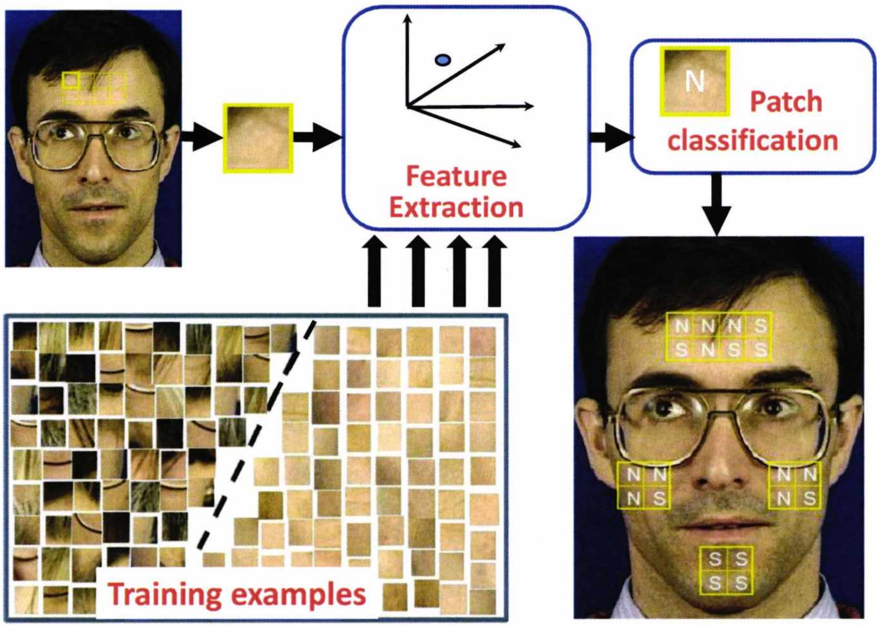


Figure 6.2: Skin/Non-Skin classifier, S: Skin patch, N: Non-Skin patch

6.3.1 Patch representation using feature-level fusion

The Gabor-based and LBP-based features that were described in Section 5.2.4, are first extracted from each patch. Twelve orientations and eight scales have been used to generate a series of Gabor responses. As a result, every patch can be represented by a Gabor-based feature vector that has 192 ($= 8 \times 12 \times 2$) elements.

The dimension of the LBP derived feature vector is also fixed at 256 elements irrespective of the patch size. A feature-level fusion algorithm is applied to combine the different sources of information.

It is necessary to separately normalize the different feature sets before concatenating them into a single vector. The normalized features used are given by: $\acute{f}_i = \frac{f_i}{\sum_{i=1}^{N} f_i}$, where f_i is the feature component before normalizing, $\acute{f}_i$ is the feature component after normalizing and N is the dimension of the feature set. Therefore, the fused feature for each patch (448 elements long) will be used to learn the texture of skin and non-skin patterns.

6.3.2 Model selection

In the case of two-class linearly separable data, as illustrated in Section 2.8.3, SVMs aims to find a maximal separating hyperplane between two classes. This "optimal separating hyperplane" is the boundary which separate the the two classes the best.

In the case of two-class nonlinearly separable data, the technique is extended to nonlinearly separable data by transforming the data to some high space, where it is linearly separable. This enables us to determine the optimal separating hyperplane in that space. This requires using a mapping kernel function. The Gaussian RBF function, which is given by Equation 6.1 was used as a mapping kernel.

$$K(x_i, x_j) = exp(-\gamma \left\| x_i - x_j \right\|^2), \gamma > 0. \tag{6.1}$$

According to [72], RBF kernel has less numerical problems as compared to the polynomial and sigmoid kernels and the linear kernel cannot separate nonlinearly separable data, whereas the RBF can so this kernel function is highly recommended in [72].

To optimize the parameters of the model, Hsu [72] has proposed a procedure to get reasonable results with LIBSVM [25]. To get appropriate generalization ability, the cross validation process was conducted to choose parameters. The procedure is given by the following steps [72] and described by algorithm 2:

1. A "grid search" space is considered of (C,γ) where $C = 2^{-5}, 2^{-3}, ..., 2^{15}$ and

$\gamma = 2^{-15}, 2^{-13}, ..., 2^{3}$.

2. Using five-cross-validation on the training set, to evaluate the validation performance for each hyper-parameter pair (C,γ) in the search space.

3. The optimal parameters pair (C,γ), the one with the best five-cross-validation accuracy, is selected.

4. The SVM model is built using the best parameters.

5. Using the trained model to test the testing set.

Algorithm 1 Grid search for determining optimum values of *gamma* and C

C: penalty for errors

γ: the kernel parameter

Accuracy: the percentage of the training data that currently classified for current γ and C using five cross-validation

$Accuracy_{best}$ represents the best *Accuracy* obtained

```
for $C \leftarrow 2^{-5}, 2^{-3}, .., 2^{15}$ do
    for $\gamma \leftarrow 2^{-15}, 2^{-13}, .., 2^{3}$ do
        Compute $Accuracy$ for current $\gamma$ and $C$ $Accuracy_{best} \leftarrow Accuracy$
        if ($Accuracy \geq Accuracy_{best}$) then
            Record $\gamma$ and $C$
        end if
    end for
end for
```

6.3.3 Experiment (1): Building the classifier

The experiment was conducted on the XM2VTS database. The data set consists of square patches from different skin regions (forehead, right region, left region and chin). Both skin and non-skin classes should be divided into training and testing sets.

Training set

The training set consists of 5900 (295×20) patches came from 295 persons. The total number of skin and non-skin patches are 4556 and 1344 patches respectively. The statistics of non-skin class are listed in Table 6.1.

Table 6.1: The statistics of non-skin class

Noise	Percentage	
Hair and facial hair	82.74%	(1112/1344 patches)
Spectacles	12.43%	(167/1344 patches)
Blue background	4.84%	(65/1344 patches)

The main question is what should be the proportion of skin and non-skin classes in the training set? Four trials (Table 6.2) will be experimented and the testing accuracy will be computed for every trial.

The next step is to decide the associated model parameters (C,γ), so we applied five-fold cross validation to the training patches set in the four trials. Then, the testing accuracy will be computed using the corresponding parameters. In the end, the model with the smallest testing error will be adopted.

Table 6.2: Four trials for determining the proportion of skin and non-skin classes

Trial	Non-skin	Skin	proportion
Trial 1	1000	1000	equal size, big size set
Trial 2	1000	500	skin class size is half non-skin class size
Trial 3	500	1000	non-skin class size is half skin class size
Trial 4	500	500	equal size, small size set

Testing set

The test set contained 344 facial patches for the non-skin class and such as the number for skin patches was chosen. Thus, there were 688 testing samples in total. In the evaluations, the testing accuracy will be used as a figure of merit. The test accuracy was defined as the total number of correctly classified patches divided by the total number of patches classified (=688). Table 6.3 lists the four trials and the results of testing phase.

Table 6.3: Four trials for building the classifier model

Trial	C	γ	Cross-Validation (%)	Testing-Accuracy(%)
Trial 1	512	2	93.45	93.17
Trial 2	128	2	93.80	86.34
Trial 3	128	8	93.26	91.28
Trial 4	512	2	93.00	88.08

Table 6.3 shows that the size of the two classes in the training set has significantly affect on the testing accuracy. It also suggests that the first model "Trial 1" should be used for building the proposed classifier because it produces the best testing accuracy

i.e. 93.17%. Figure 6.3 shows the results of the training phase for Trial 1. The circle denotes the area of optimum accuracy (optimum Cross-Validation= 93.45%) corresponding to the best value of the parameters $C = 2^9$ and $\gamma = 2^1$.

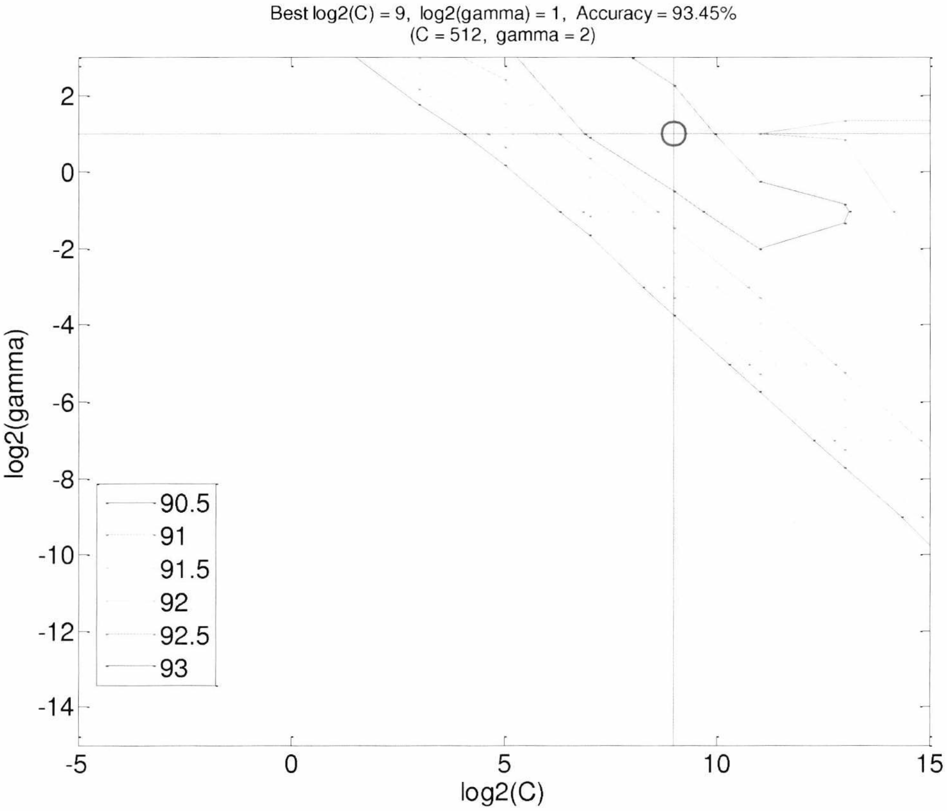


Figure 6.3: Grid search on $C = 2^{-5}, 2^{-3}, \ldots, 2^{15}$ and $\gamma = 2^{-15}, 2^{-13}, \ldots, 2^{3}$.

6.3.4 Experiment (2): Classifying the twenty patches

The aim of this experiment is to automatically classify the twenty facial patches. The model that resulted from Experiment 1 will be used to classify the patches. The data consists of 1180 facial images where every face image has twenty patches.

Figure 6.4 shows the total number of classes as determined automatically by the proposed Skin/Non-Skin Classifier which represent the twenty different patches. The individual is considered if the patch belonging to him is classified as a skin patch in all the four sessions.

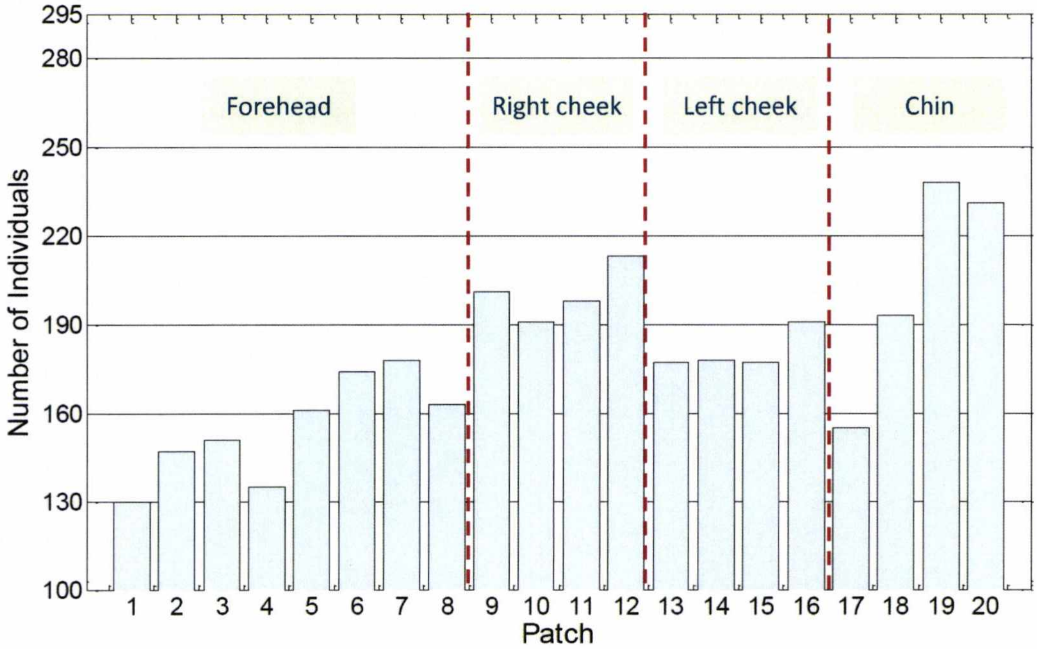


Figure 6.4: The total number of the classes of usable skin images for the twenty patches

This figure shows that the top area of forehead (1-4) has more noise than the bottom area (5-8). This noise could come from the hair line or its shadow. It can also be seen that the lower chin area (19-20) has a higher of classified skin patches. The issue with low skin patches in the upper chin area (17-18) could be due to the lower lip area (or its shadow) being present in the selected patches. Additionally, the left cheek area (13-16) has more noise than the right cheek area (9-12), this could be due to the head pose variation.

Overall, the lower chin area has the highest rate of skin patches for the images provided by the XM2VTS database [142] used. There will be, in Section 7.2, a comparison between these patches for identity recognition.

6.4 Comparison between different facial skin regions for identity recognition

The motivation of this experiment is to determine the relative biometric information content of different face regions namely forehead, right cheek, left cheek and chin in order to achieve to the region which has the most useful information.

6.4.1 Overall region Skin/Non-Skin classification

Each overall region window can be classified into two classes depending on the labels of its patches. These classes are: (i) pure-skin region if all the patches are classified as skin. (ii) non-skin region if at least one of the patches is classified as non-skin. Figure 6.5 shows the advantage of using the decomposition into eight patches for the forehead region classification rather than doing a Skin/Non-Skin classifier for the whole forehead region. It is seen that with the proposed approach forehead regions that have a small amount of non-skin content will not be classified as Pure-Skin Forehead, thereby only leaving regions purely composed of skin texture for further processing.

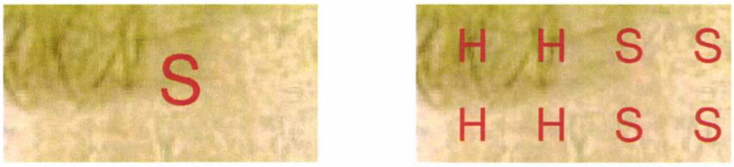


Figure 6.5: An example of the classification results of applying the Skin/Non-Skin classifier on (left) the whole forehead region and (right) eight patches separately

6.4.2 Experimental Results

The experiment was conducted on the XM2VTS database [142]. As determined by Skin/Non-Skin classifier, only 111, 160, 153 and 240 of the 295 people have images that could be used for the forehead, right cheek, left cheek and chin regions respectively. For the identification scenario, the images from the fourth session are chosen as the probe set, and the first three sessions are used as the reference gallery set.

Two levels of fusion are compared: feature-level and score-level. Each region, in feature-level fusion, is represented by concatenating the feature vectors extracted from every patch to form *a forehead feature vector*. Thus, the four regions: forehead, right cheek, left cheek and chin will be respectively represented by four feature vectors with dimensions of 8 × 448, 4 × 448, 4 × 448 and 4 × 448.

Table 6.4 shows the rank-1 identification results for each region. These results suggest that forehead region yields the highest recognition accuracy. Similar observations have been reported by Lin et al [129]. They did face recognition experiments on three facial regions: forehead, nose and left eye and concluded that forehead region produces the best recognition rate.

The rank-1 identification of chin region is in the second rank. This may be due to the fact that forehead and chin regions are less affected by expression variations.

However, the size of chin region is half of the forehead region size. This indicates to the importance of chin region skin features.

Table 6.4: Rank-1 identification accuracy (%) of the proposed system for four facial regions

	Forehead	Right Cheek	Left Cheek	Chin
Gabor	87.39	74.38	68.63	75.42
LBP	80.18	66.25	64.05	69.17
Feature-Level fusion	**90.99**	78.75	73.20	**82.50**
Score-Level fusion	**89.19**	78.13	77.78	**80.83**

For the verification scenario half of the experimental data are used for training and half of them are used for testing. Table 6.5 lists the resulting Equal Error Rates of the proposed system for each of these regions. Figure 6.6 shows the FRR against FAR of different regions. Evidently, the DET curve for the forehead region indicates the best performance in the verification scenario.

Table 6.5: Equal Error Rate of the proposed system for four facial regions in the verification scenario

	Forehead	Right Cheek	Left Cheek	Chin
Gabor	1.08*e*-01	1.16*e*-01	1.33*e*-01	1.29*e*-01
LBP	1.26*e*-01	1.10*e*-01	1.44*e*-01	1.29*e*-01
Feature-Level fusion	8.64*e*-02	9.09*e*-02	1.27*e*-01	1.15*e*-01
Score-Level fusion	**6.76** *e*-02	**7.81*e*-02**	**1.01*e*-01**	**1.06*e*-01**

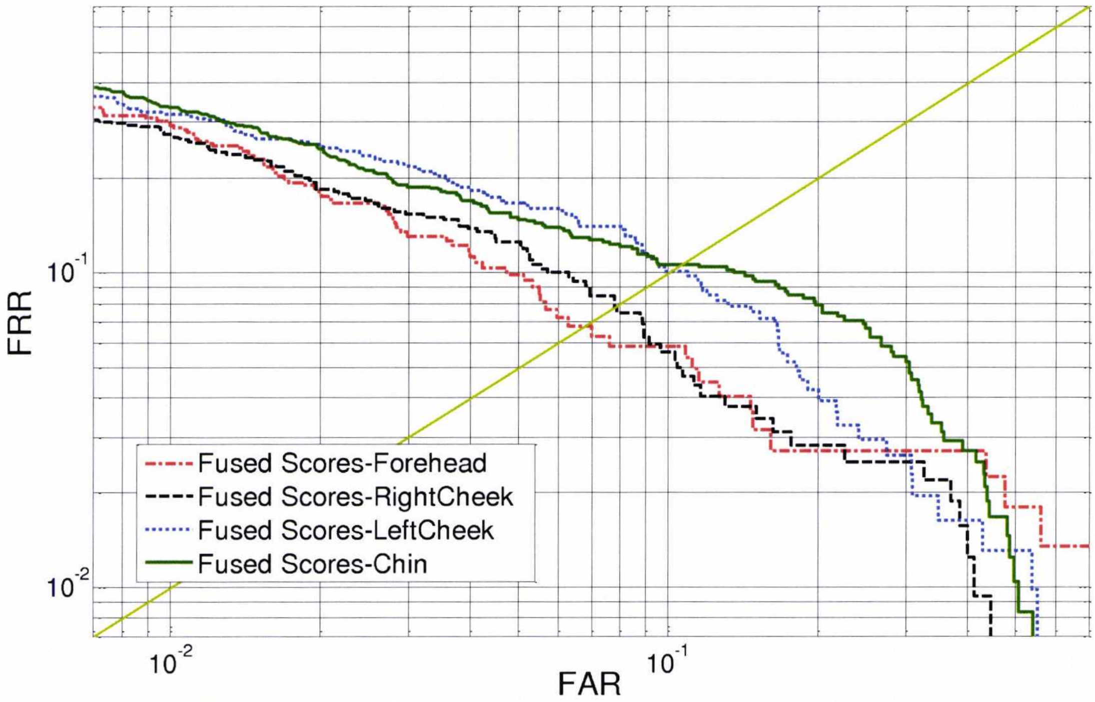


Figure 6.6: DET curves of the performance of the proposed system on four facial regions in the verification Scenario

6.5 Skin texture features for improving face recognition systems

This section investigates the possibility of exploiting facial skin texture regions to further improve the performance of face recognition systems. The forehead region is chosen as a representative of facial skin texture since it achieved the best recognition result (90.99%) as shown in Table 6.4.

The overall block-diagram of the proposed system is shown in Figure 6.7. It processes the information from the overall face and the facial skin region in two separate channels and combines the results of these independent classifications, where possible, at the final stage using score fusion techniques. This section will give a step-by-step description of the proposed algorithm.

6.5.1 Global Feature Extraction

The following pre-processing procedure was applied to all images in the database:

1. A rotation step, which, based on the manually labeled eye coordinates, aligned and normalized facial images.

2. Face detection using the *Viola-Jones* detector [217], which extracted the face regions from the facial images.

3. Cropping and down-sampling them to a fixed size (90 × 110 pixels).

Later, five different methods global methods for face recognition are implemented. They are based on the following algorithms: PCA [213], LDA [12], KPCA [186], GDA [10] and KDDA [133].

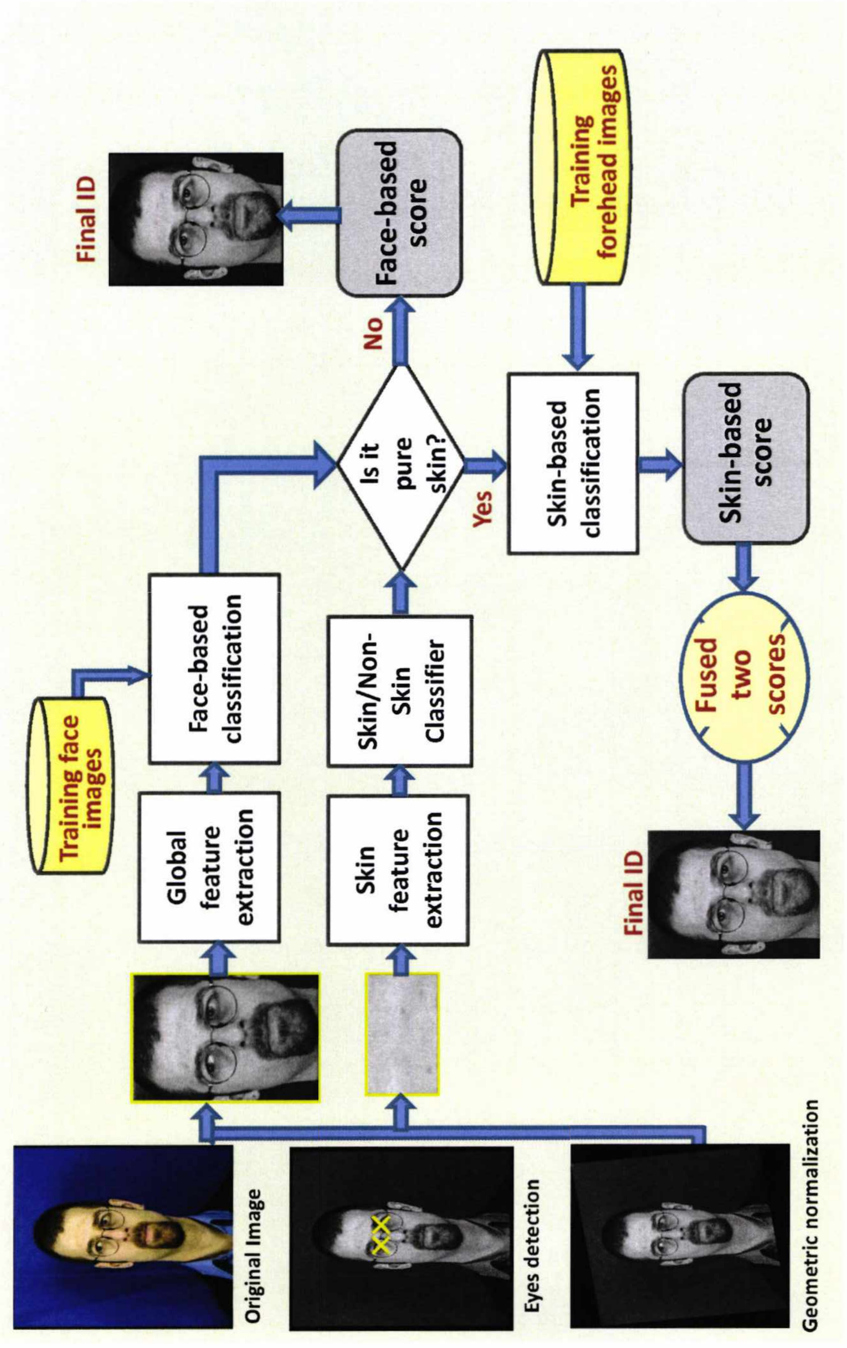


Figure 6.7: The framework for the face recognition system

6.5.2 Skin Feature Extraction

The skin channel processing on the other hand includes:

(a) skin region localization (described in 5.2.2),

(b) feature extraction of skin texture (described in 6.2.1),

(c) region classification into skin and non-skin (described in 6.4.1) and

(d) [overall] classification using a Sparse Representation Classifier (SRC).

6.5.3 Fusion Technique

The fusion of the output produced by the whole-face and forehead region classifiers are combined at score-level in two possible ways:

- Scheme 1: Z-score normalization [85] is first implemented on the scores obtained from forehead-region and whole-face classifiers. Then the fused score will be equal to the summation of both scores if forehead-based score is available, otherwise, the fused score will be equal to the whole face-based score.
- Scheme 2: The fused score will be equal to the forehead-based score if is available, otherwise, the fused score will be equal to the whole face-based score.

6.6 Experimental Results

Several experiments were conducted on the XM2VTS dataset using gray level images. For the purpose of comparison with published results, 295 × 3 images from the first three sessions are selected for the training stage. The probe set is composed of 295 images from the fourth session. As determined by the Skin/Non-Skin classifier, only

111 of the 295 individuals in the database have images that could be used in the forehead region texture channel. For the identification scenario, the images from the fourth session are chosen as the probe set, and the first three sessions are used as the reference gallery set.

6.6.1 Performance of the whole-face channel

Table 6.6 shows a performance comparison for PCA, LDA and kernel-based methods KPCA [186] and GDA [10] both employing the polynomial kernel function $k(x, y) = (x \cdot y)^d$ of degree $d = 2$. KDDA [133] is also implemented using a RBF kernel function $k(x, y) = \exp\left(-\left\|x - y\right\|^2 / \sigma^2\right)$ with variance $\sigma^2 = 1.11e7$.

We find that the choice of the distance metric is important as it affects the classification performance (e.g. cosine distance appears to work better for the measurement of the similarity and the LDA-based methods appear to work worst when used in conjunction with the *l1*-norm).

Table 6.6: Rank-1 identification of several whole-face methods using three different distance metrics

method	*l1*-norm	*l2*-norm	*Cos*
PCA	76.95	75.93	77.63
LDA	20.00	64.41	67.80
KPCA	73.90	72.88	77.29
GDA	32.20	75.93	84.75
KDDA	**80.34**	**81.69**	**91.19**

6.6.2 Performance of the pure-skin channel

Table 6.7 lists rank-1 identification rates for the subset of images with usable forehead regions of the proposed skin-based algorithm using two different classification approaches, the SRC [226] and the k-nearest neighbour with $k = 1$. We can see that the SRC outperforms the $k - NN$ classifier (for all distance metrics).

Table 6.7: Rank-1 identification accuracy for the proposed skin-based algorithm with forehead region

Classifier		Accuracy%
Nearest Neighbour	*l1*-norm	90.99
	l2-norm	85.59
	Cos	88.29
Sparse Representation		**96.40**

6.6.3 Performance of the combined channels

Figure 6.8 shows the improvement in the performance of different face recognition systems after combination with the skin information. Although the database contains only a relatively small number of images where the forehead region can provide pure-skin patches, nevertheless the use of this information improves the performance of face recognition system from 77.63% to 84.07% (PCA) , from 67.80% to 75.25% (LDA), from 77.29% to 83.73% (KPCA), from 84.75% to 88.14% (GDA) and from 91.19% to 93.22% (KDDA) using Scheme 2. The highest rank-1 identification accuracy is equal to 93.56% which is produced from score-level fusion of KDDA scores and forehead-based score using Scheme 1.

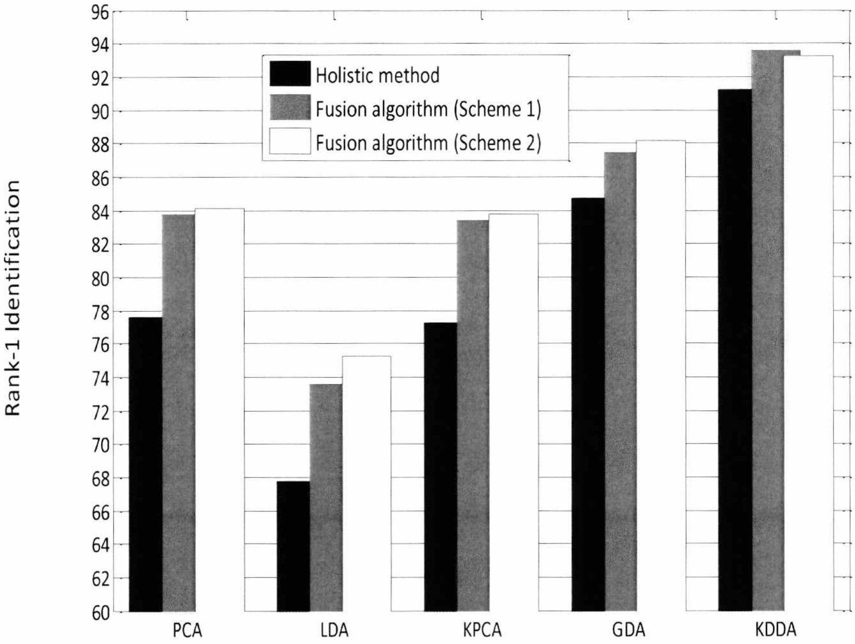


Figure 6.8: Improving holistic approaches by skin-based information.

6.6.4 Comparison with the state-of-the-art techniques

We compared the best result of the proposed fusion algorithm to several previously published results on the XM2VTS dataset based on the common evaluation protocol mentioned above. These methods included: Bayesian classification [123], Bayesian [123], Kernel LDA [123], LDE [123],Multi-class NDA [123], NNSA [123] [123], wavelet decomposition using three levels [237], direct correlation of face image intensity [220], direct correlation using local area image intensity around each of the 35 fiducial points chosen according to a face graph model [220], and Bayes [220]. It is noted, from Table 6.8, that the proposed method achieves comparable performance to the best of the published results.

Table 6.8: Comparative recognition results for the proposed algorithm and other face recognition methods

Method	Accuracy %
Bayesian [123]	88.50
Kernel LDA [123]	89.80
LDE [123]	90.20
Multi-class NDA [123]	91.50
NNSA [123] [123]	93.60
Wavelet level[2][237]	77.29
Wavelet level[3][237]	80.34
Wavelet level[4][237]	78.31
Local intensity [220]	80.30
Global intensity [220]	84.10
Bayes [220]	92.90
Proposed Algorithm	**93.56**

6.7 Conclusion

The development of a novel Skin/Non-Skin classifier was an essential step for automatically excluding non-skin areas in facial images. A comparison between different facial regions was implemented and these results suggest that the forehead region yields the highest recognition accuracy followed by the chin region. The forehead skin information was combined with the kernel discriminant analysis KDDA at score-level. The experimental results indicate that the face recognition accuracy can be improved by the explicit exploitation of identity information in pure skin regions.

Chapter 7

An Adaptive Scheme for Skin-based Face Recognition

7.1 Introduction

This chapter presents a fully automated and novel method for face recognition based on exploiting facial skin texture as a source of biometric information. The twenty extracted patches that were passed, in the previous chapter, through a SVM-based classifier will be used for identity recognition.

The chapter is organized as: A comparative study between the performance of these patches will be presented in Section 7.2 to investigate which patch has the most significant contribution to biometric recognition. Section 7.3 will describes the adaptive strategy for combining pairwise patch distances between individuals with different numbers of usable skin patches.

Experiments using the XM2VTS database will be discussed in Section 7.4 and Section 7.5 will concludes this chapter. The results show that the proposed method achieves comparable performance to the best of the published results and suggests that it may be particularly useful to combine this method with the traditional face recognition algorithms because it can automatically cope with the problem of noise in skin areas.

7.2 Recognition by Facial Skin Patches

The motivation of this experiment is to identify which face patches contain significant facial information. Therefore, face recognition experiments will be conducted using only individual facial patches one at a time. The total number of individuals as automatically determined by Skin/Non-Skin Classifier was shown in Section 6.3. Since the distance between the two eyes in the XM2VTS database ranges from 84 up to 122, the size of facial patches ranges from 21 × 21 pixels up to 30 × 30 pixels.

For the recognition scenario, the people patches from the fourth session are chosen as the probe set, and the first three sessions are used as the reference gallery set. Two levels of fusion are compared, Feature-level and score-level fusion. Table 7.1 records the rank-1 identification for the twenty different patches.

The results show that some facial patches appear to exhibit significantly higher recognition performance when compared to the other patches. For instance, these results suggest that the upper chin area (17-18) yields the highest recognition accuracy. These results can be used for weighting the proposed adaptive algorithm which will be presented later.

Table 7.1: Rank-1 identification(%) at two different levels of fusion for the twenty patch images

Patch	Feature-Level (%)	Score-Level (%)
1	16.95	20.34
2	11.86	18.31
3	17.29	21.02
4	15.25	21.02
5	19.66	25.42
6	17.29	23.05
7	17.97	24.75
8	18.31	27.80
9	23.05	26.10
10	18.64	28.14
11	20.34	25.09
12	25.42	35.59
13	20.34	26.78
14	21.70	25.42
15	21.02	29.83
16	22.03	23.05
17	27.80	38.64
18	**33.22**	**45.09**
19	27.12	31.19
20	23.73	30.85

7.3 An Adaptive Scheme for Skin-Based Face Recognition

The enrollment and authentication phases of the proposed system (Figure 7.1) contain the following steps:

(1) Face and facial features detection,

(2) Face normalization (Section 5.2.1),

(3) Localization of facial skin regions (Section 4.6),

(4) Partitioning of Facial Regions into twenty non-overlapping patches (Section 6.2),

(5) Feature extraction of these patches using Gabor and LBP approaches (Section 5.2.4) and fused feature of them (Section 6.3.1),

(6) Training a "Skin/Non-Skin" classifier in the training mode (Section 6.3.3), and

(7) Application of this classifier to classify every patch into skin or Non-skin classes (Section 6.3.4).

(8) Matching: To obtain the distance matrix, local distances are first computed between skin patches of the query face image and the corresponding skin patches of the reference templates in the database.

Then the global distance between the probe and gallery image is a weighted sum of the local distances adapted in such a way to include the contribution of each usable skin patch to the recognition stage. Finally, the person is identified as the one whose image produces the smallest global distance. More details for computing these two distances as follows.

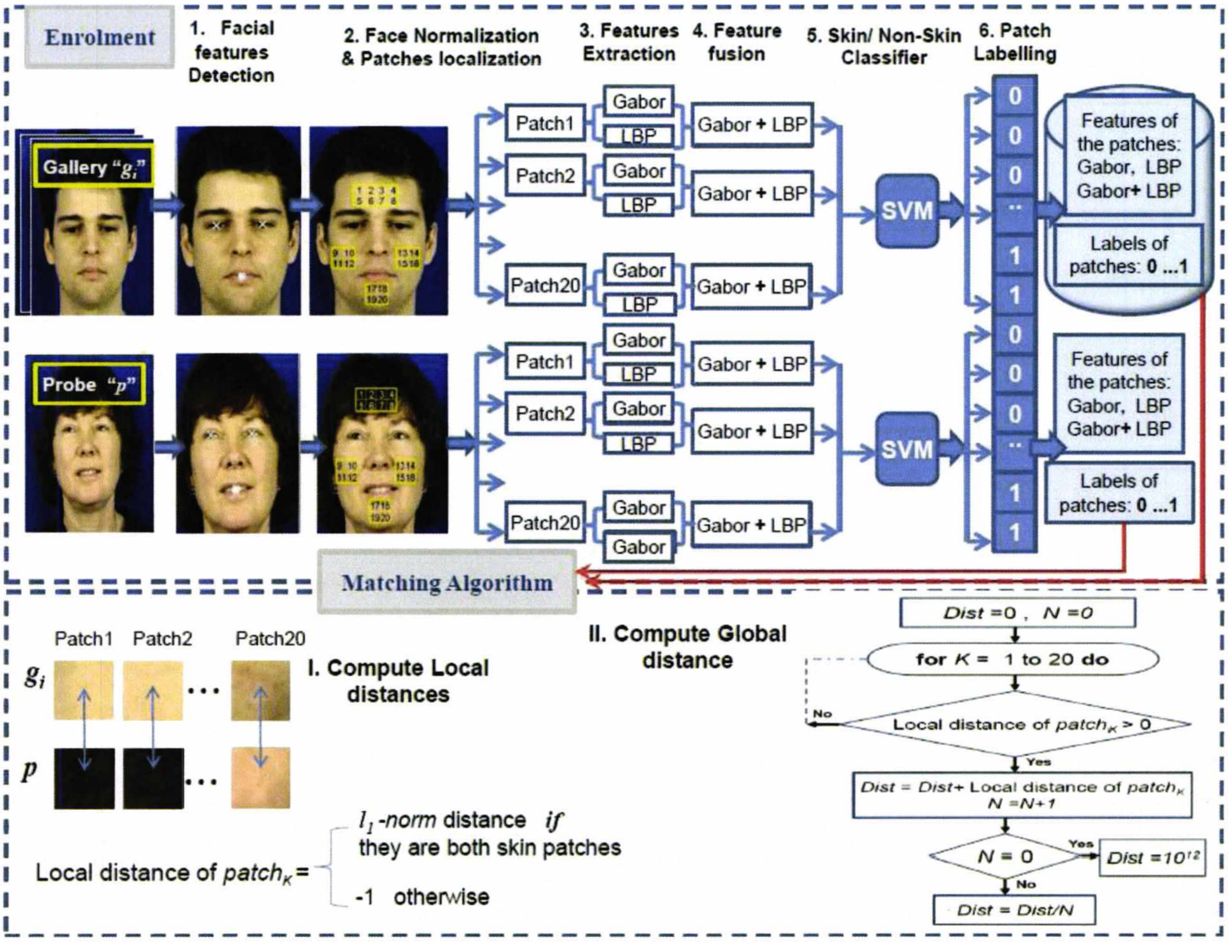

Enrolment
1. Facial features Detection
2. Face Normalization & Patches localization
3. Features Extraction
4. Feature fusion
5. Skin/ Non-Skin Classifier
6. Patch Labelling
Gallery "g_i"
Probe "p"
Patch1
Patch2
Patch20
Gabor
LBP
Gabor + LBP
SVM
0
1
..
Features of the patches: Gabor, LBP Gabor+ LBP
Labels of patches: 0 ...1
Matching Algorithm
Patch1 Patch2 Patch20
g_i
p
I. Compute Local distances
II. Compute Global distance
Local distance of $patch_K$ = l_1-norm distance if they are both skin patches; -1 otherwise
Dist =0 , N =0
for K = 1 to 20 do
Local distance of $patch_K$ > 0
No
Yes
Dist = Dist+ Local distance of $patch_K$
N =N+1
N = 0
Dist =10^{12}
Dist = Dist/N

Figure 7.1: The framework of the face recognition system

7.3.1 Computing local distances

To match a probe image p against one of the records in the database, gallery g_i, local distances should be calculated patch by patch as shown in Algorithm 2. Hence, the difference of two patches can be represented by the distance of two feature vectors, after checking (using line 2 in algorithm 2) that they are both skin patches. There are many methods to measure the distance of two vectors such as l_1 and l_2 norms.

Algorithm 2 Local Distances computation between the gallery g_i and probe p

Input:
$PatchLabel(g_i)$: the labels of the patch for ith gallery image g_i,
$Feature(g_i)$:the features set of the patch for ith gallery image g_i,
$PatchLabel(p)$: the labels of the patch for the probe image p,
$Feature(p)$: the features set of the patch for the probe image p.
Output: $d \in \Re^{1\times 20}$.

```
1: for $K = 1$ to 20 do
2:     if ($PatchLabel_K(x_i) = 1$ and $PatchLabel_K(y) = 1$) then
3:         $d_K$ = the distance between $Feature_K(x_i)$ and $Feature_K(y)$;
4:     else
5:         $d_K = -1$
6:     end if
7: end for
```

7.3.2 The global distance

The simplest way to get an overall score or distance of combination is to simply add the local distances from different patches, i.e. equal weights are assigned since we do not know the relative importance of each patch. Since it is possible for the images of the same subject to have different number of usable facial skin patches extracted, the matching of any two facial images based on facial skin patches could result in matching n patches in one image against m patches in the other, where $n \neq m$.

Therefore, for matching two images p and g_i based on their facial skin patches, it is essential to identify the set of common skin patches between them. The corresponding elements of this set are compared and the distances are combined, weighted with the reciprocal of the number of common patches N.

It is noticed that, if there is no common skin patches between the probe image p and the ith gallery image g_i, the resulted global distance $Dist(p, g_i)$ will be equal to zero (the minimum distance), so we could replace it with a very large number, say 10^{12}, to guarantee that the corresponding image would not be returned as identification result. Algorithm 3 describes this procedure.

7.4 Experimental Results

We use a four-fold cross-validation analysis for testing, thereby, the 1180 face images are partitioned into 4 sets. Each set contains one face image for each individual. For each experimental trial, one set is chosen as the probe set, and the remaining three folders are used as the reference gallery set. The pair-wise distances are then computed between the probe image and the gallery set. The mean of rank-1 identification rates of the proposed method on the four experimental trials are reported in Table 7.2.

The analysis of performances in Table 7.2 shows that the proposed algorithm using the $l_1 - norm$ distance performs significantly better than using $l_2 - norm$. In

Algorithm 3 Global Distance computation between the gallery g_i and probe p

Input: $d \in \Re^{1\times 20}$, the local distances between the ith gallery image g_i and the probe image p.

Output: $Dist \in \Re^{1\times 1}$.

Initialization: $Dist = 0$, N(Number of common patches)$= 0$.

```
for $K = 1$ to 20 do
    if $d_K > 0$ then
        $Dist = Dist + d_K$;
        $N = N + 1$;
    end if
end for
if $N \neq 0$ then
    $Dist = Dist/N$;
else
    $Dist = 10^{12}$;
end if
```

Table 7.2: Mean of rank-1 identification rates (%) of the proposed method using two different similarity measures with cross-validation

	$l_1 - norm$ (%)	$l_2 - norm$ (%)
Gabor	73.56	66.86
LBP	78.64	64.58
Feature-Level fusion	85.34	73.22
Score-Level fusion	**87.12**	80.25

addition, it shows that the different similarity measures perform differently for each feature set. For instance, Gabor feature set performs better than LBP feature set in the system which uses $l_2 - norm$ distance while LBP feature set works noticeably better than Gabor when $l_1 - norm$ distance is used in the proposed system.

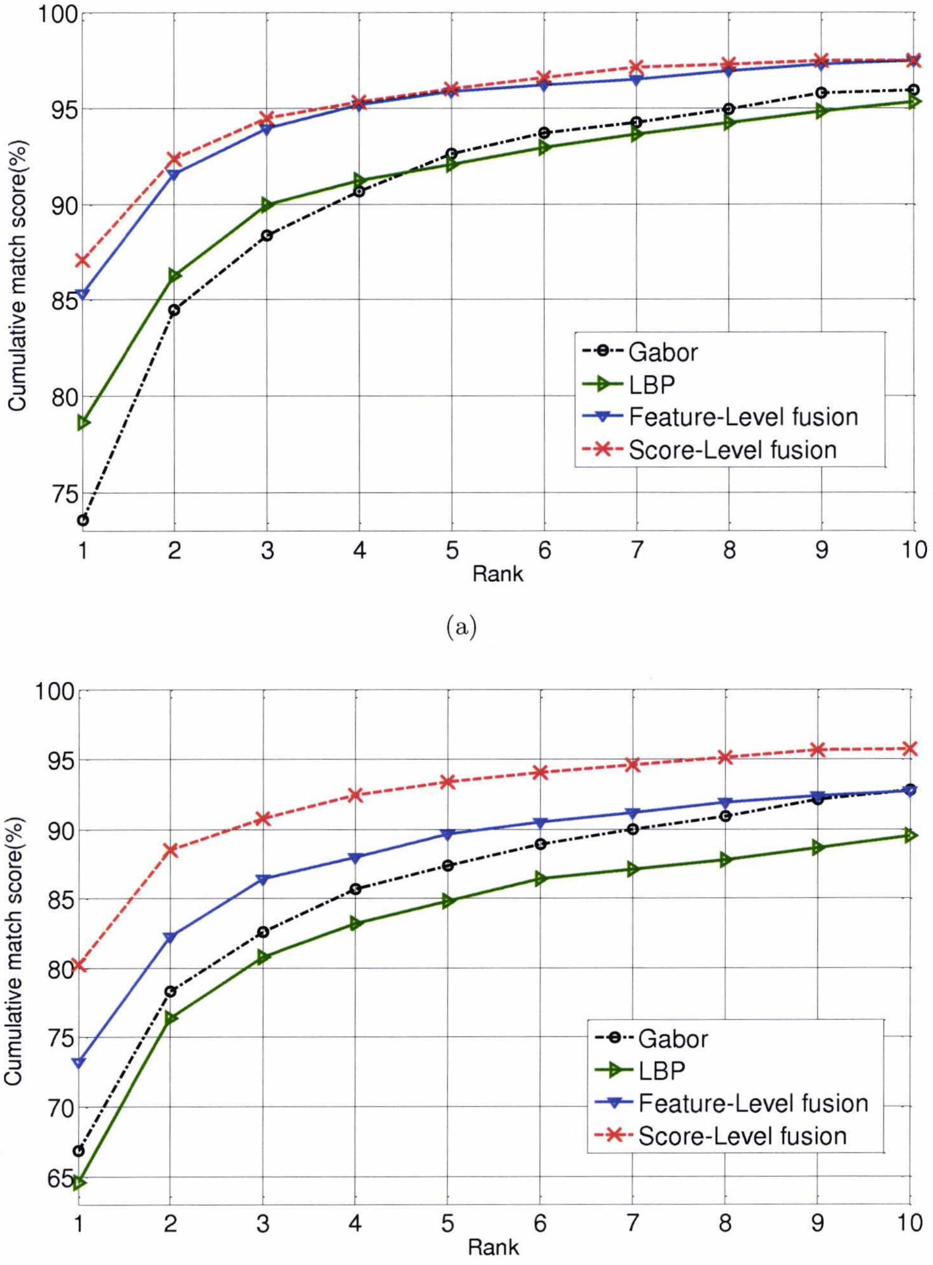
It is evident that fusing these two types of features, Gabor and LBP, at feature level gives much better performance than the individual features alone. However, fusing them at score level provides a further performance improvement. This suggests that these two feature sets do extract different but complementary information. The cumulative match score vs rank is used to show the performance of each algorithm, in Figure 7.2. Fused features at score level again exhibit obvious evidence of superiority in performance over other methods.

7.4.1 Comparison to previous work

Using the protocol mentioned in Section 7.4.1, we compared the proposed adaptive method, which is obtained by using the l_1- norm distance and fused scores of Gabor-based and LBP-based features, to several previously published results on the XM2VTS dataset.

These methods include wavelet decomposition using three levels [237], PCA [220], direct correlation of face image intensity [220], and direct correlation using local area image intensity around each of the 35 fiducial points chosen according to a face graph model [220].

Table 7.3 lists the Rank-1 identification of every partition. It is noted that the proposed new method of integrating the Gabor and the local binary pattern features achieves comparable performance. It also indicates that skin texture-based feature is



(a)

(b)

Figure 7.2: Cumulative match score curves of four different algorithms using (a) $l_1 - norm$ distance (b) $l_2 - norm$ distance

not short term biometric trait since the data acquisition was distributed over a long period of time (a period of five months) [142].

Table 7.3: Rank-1 identification(%) of the proposed method and other five methods on the XM2VTS face database using their protocol

Partition	*l*	2	3	4	**Mean**(%)
Wavelet (level[2]) [237]	77.29	81.69	82.03	83.73	81.19
Wavelet (level[3]) [237]	80.34	85.42	83.05	84.75	83.39
Wavelet (level[4]) [237]	78.31	85.42	81.69	83.39	82.20
PCA [220]	86.40	84.40	82.00	83.40	84.10
Full intensity [220]	89.20	85.10	86.10	84.10	86.10
Local intensity [220]	87.50	89.20	84.40	80.30	85.40
Proposed method	84.41	86.44	87.80	89.83	**87.12**

In order to compare our results with the texton-based method presented in [128], we used the protocol that has been used in that paper. For each person, we chose the first two partitions as the gallery set, while the other two partitions are used as the probe set. Thus, we have 590 gallery samples, and 590 probe samples. Table 7.4 shows that the proposed method significantly outperforms that in [128].

Table 7.4: Rank-1 identification(%) of the proposed system and another skin-based algorithm on the XM2VTS face database using its protocol

Approach	Accuracy(%)
Proposed method	**83.05**
Texton-based method[128]	66.90

7.5 Conclusion

A novel method for face recognition is presented in this chapter. By partitioning the face image into several patches, we apply the Skin/Non-Skin Classifier that includes the skin patches and exclude patches that may be obscured by hair, facial hair or spectacles. An adaptive method working on the usable skin patches is presented for face recognition. The experiments on the XM2VTS database have been carried out to demonstrate that this adaptive scheme is able to utilize all available skin texture information for biometric recognition. The results also indicate that facial skin texture is not short term biometric trait.

Chapter 8

Partial Face Recognition using the Forehead Region Alone

8.1 Introduction

This chapter presents a scheme for face recognition in application scenarios where only a partial image of the forehead region available for classification. Images of the forehead (above the eye-brows) are chosen for this study as they are less influenced by facial expressions. A scheme for segmenting the head region in such images is presented. The forehead region is then extracted, without the use of facial landmarks such as eye locations, and analyzed to segment and exploit available skin patches for classification.

We do not assume the presence of the eyes in the images so the challenges, as some of them are illustrated in Figure 8.1, will be (i) how to detect partial faces in images with high illumination variation (ii) how to localize the forehead without any of the usual landmarks such as eye locations (iii) how to detect pure-skin forehead regions in the presence of possible occlusions by hair, head coverings, etc. (iv) how to recognize such regions without using facial landmarks for alignment.

The system is tested on FRGC ver2.0 database and presents promising recognition results for a very challenging problem. The outline of the chapter is organized as follows: Section 8.2 presents the overview of the proposed algorithm. Sections 8.3 and 8.4 are devoted to the experimental results and conclusions respectively.

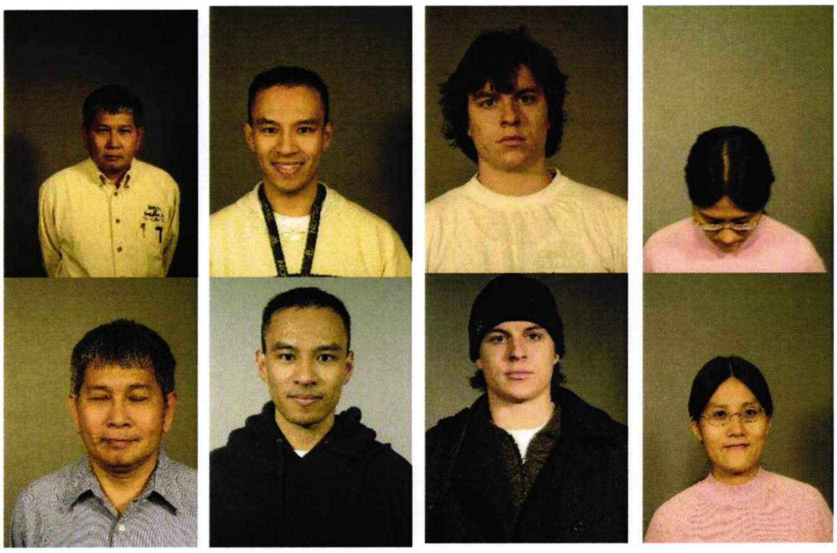


Figure 8.1: Challenging samples

8.2 Overview of the Proposed Algorithm

The overall block-diagram of the proposed system is shown in Figure 8.2. This section will give a step-by-step description of its components.

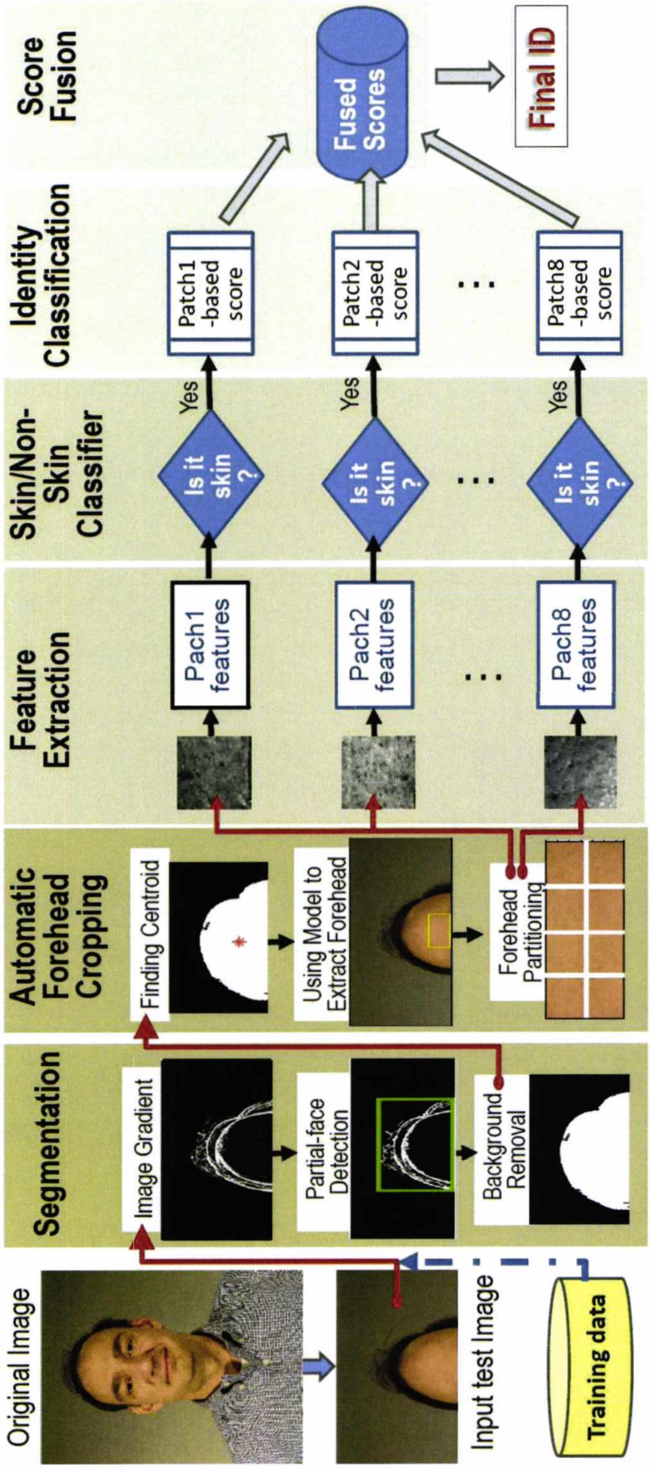


Figure 8.2: Overall system architecture

8.2.1 Partial-Face Segmentation

We assume that the input to the system are images that contain only the upper head region (above the eyes). Therefore, we implement the following steps for subtracting the background:

- Step 1: Image is first processed to obtain a gradient image (which is less sensitive to illumination variations [56]) and thresholded to form a binary image.
- Step 2: The resulting blob (non-zero pixels) is bounded in a rectangle but it still contains some background pixels as well as pixels from the skin and non-skin (hair) regions of the face.
- Step 3: A skin detection algorithm [106] is used to identify pixels belonging to the skin region in the bounding rectangle.
- Step 4: A k-means clustering algorithm is then used on all the non-skin pixels within the rectangle to classify them further into background and hair classes.
- Step 5: All the background pixels are removed and the partial-face image is segmented from the background to obtain a mask of the face region available for further processing.

8.2.2 Automatic Cropping of Forehead Region

In a conventional full face recognition scenario, a forehead window can be easily determined using the eye centers (Section 5.2.2). For the problem posed in this chapter, there is no such landmark present so the localization of the forehead window becomes the principal challenge facing this approach. To solve this problem, the following steps are implemented:

1. An automated algorithm was developed that exploits head shape cues for localizing a virtual landmark ; the segmented region centroid. This point is the centroid of the binary mask mentioned in Step 5, in Section 8.2.1.

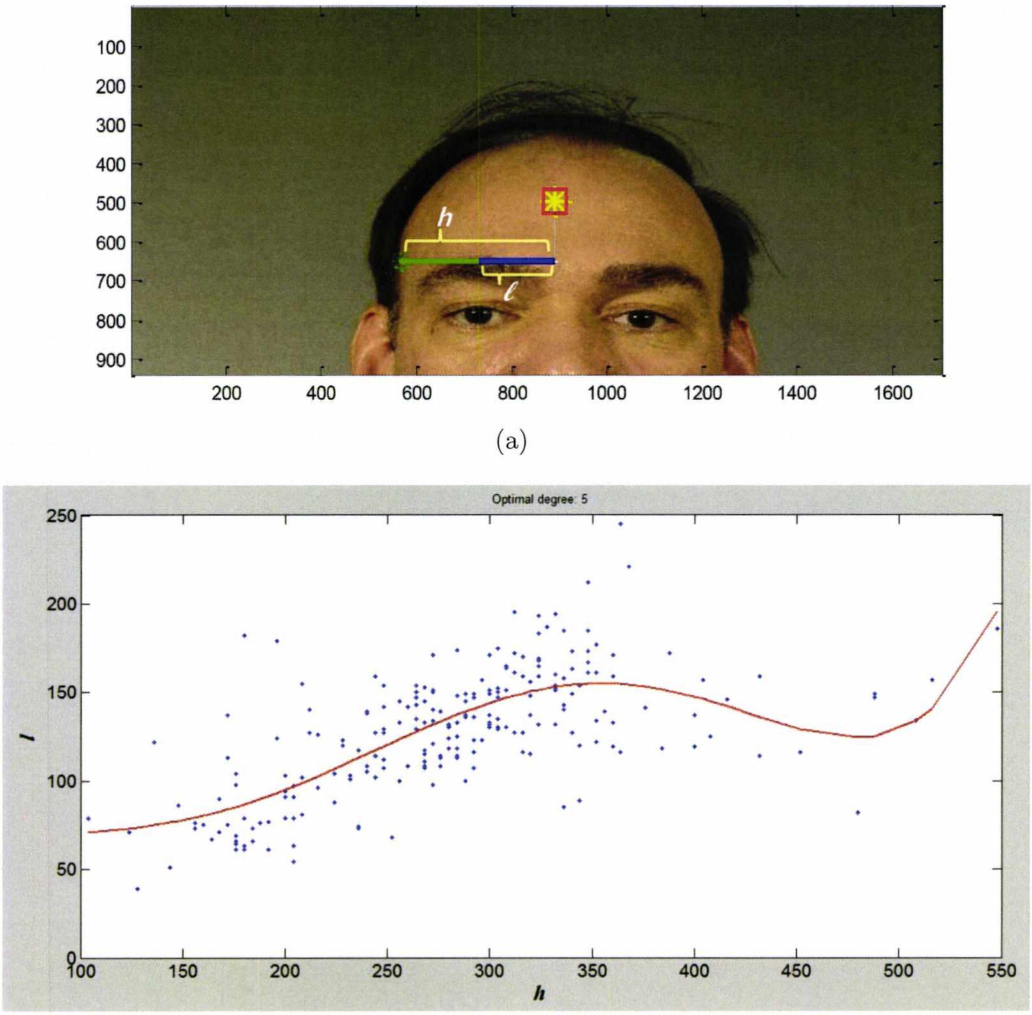
2. The forehead window was chosen to be a rectangle of size of $2l \times l$, where l indicates the estimated horizontal distance between the right eye center and the centroid. This choice was based on results in Section 5.2.2.

3. Since the coordinate of right eye center is unknown, the value of l is inferred from the model that represents the distribution of l and h, where h is the horizontal distance from the centroid to the last skin-pixel in the right side of the face. These two distances are illustrated in Figure 8.3 (a).

4. A polynomial model is chosen to perform a parametric regression analysis on the training data. This training data contains image where the eyes are included. The polynomial degree is determined by minimizing Akaike's information criterion (AIC) [5]. The AIC is given by:

$$\text{AIC} = 2k - 2\ln(L) \tag{8.1}$$

where k is the number of training data, and L is the maximum likelihood of the estimated model. A model with minimum AIC value is chosen as the best model to fit the data (Figure 8.3(b)).



(a)

(b)

Figure 8.3: The proposed forehead model (a) The two distances l and h (b) The distribution of these distances

8.2.3 Feature Extraction

After the forehead windows are cropped, they are resized to 80×160 pixels. Then they are divided into eight non-overlapping patches with equal size. Extensive experiments have been conducted to determine the optimal patch size, number of orientations and scales. We have found out that for this database, the optimum number of orientations and scales for Gabor features are 16 and 10 respectively and the best patch size was found to be of size 40×40 pixels.

The texture features are extracted from every patch in order to implement two classification stages; (i) Skin/Non-Skin classification and (ii) Identity classification. Each patch, in both stages, is represented by a combined feature vector of Gabor and LBP-based feature sets that are separately computed.

The LBP operator [152] is first used for extracting features from the input patch. This vector $\overline{F_L}$ is combined with another set of features extracted using Gabor filters.

The Gabor feature vector used is different for the Skin/Non-Skin classifier and the identification classifier as shown below. An input patch I is transformed to multiple Gabor images, $r_{o,s}$, by convolving it with Gabor filters $\psi_{o,s}$ that are illustrated in Section 5.2.4.

$$r_{o,s}(x, y;\ o, s) = I(x, y) * \psi_{o,s}(x, y) \tag{8.2}$$

where o and s denote the orientation and scale of the Gabor filters respectively, and, $*$ denotes the convolution operator.

i. Features for Skin/Non-Skin classification All means, μ_{os}, and standard deviations, σ_{os}, of the magnitude $|r_{o,s}|$ are computed as follows:

$$\mu_{os} = \sum_x \sum_y |r_{o,s}|, \text{ and } \sigma_{os} = \sqrt{\sum_x \sum_y (|r_{o,s}| - \mu_{os})^2} \tag{8.3}$$

A feature vector $\overline{F_s}$ is constructed using the collections of means and standard derivations where m, n are the total number of scales and orientations respectively. It is denoted as:

$$\overline{F_s} = [\mu_{00}\, \sigma_{00}\, \mu_{01} \ldots\, \mu_{m-1n-1}\, \sigma_{m-1n-1}] \tag{8.4}$$

ii. Features for Identity classification For an efficient texture characterization and local representation of the input patch, the Entropy is added as another feature to the previous statistical measures. This is defined as:

$$e_{os} = -\sum(p\, log_2(p)) \tag{8.5}$$

where p is the probability of each intensity level in the image. A modified feature vector $\overline{F_I}$, is used for classification, as follows:

$$\overline{F_I} = [\mu_{00}\, \sigma_{00}\, e_{00}\, \mu_{01} \ldots\, \mu_{m-1n-1}\, \sigma_{m-1n-1}\, e_{m-1n-1}]\,. \tag{8.6}$$

iii. Feature-Level fusion Finally, a feature-level fusion algorithm is applied to combine the different sources of information together.i.e

$$\begin{aligned} F_s &= [\overline{F_L}, \overline{F_s}]; \\ F_I &= [\overline{F_L}, \overline{F_I}]; \end{aligned} \tag{8.7}$$

It is necessary to separately normalize the different feature sets before concatenating them into a single vector. The normalized features used are given by:

$$\acute{f_i} = \frac{f_i}{\sum_{i=1}^{N} f_i}, \tag{8.8}$$

where f_i is the feature component before normalizing, $\acute{f_i}$ is the feature component after normalizing and N is the dimension of feature set. This procedure leads to a feature vector size of $576(256 + 2(16 \times 10))$ for the Skin/Non-Skin classifier and a feature vector size of $736(256 + 3(16 \times 10))$ for the identification classifier.

8.2.4 Skin/Non-Skin Classifier

This classifier aims to distinguish pure "Skin" patches from "Non-Skin" patches because only such patches that contain skin texture alone will be used for identity classification. A patch is considered to be non-skin if it is covered partially or completely by hair, otherwise it is labeled as a skin patch. The fused features vector, as explained in Section 8.2.3, is used in the training stage to learn the texture of skin and non-skin patterns. In our experiments, we use the LIBSVM software library [25] for building SVM classifiers and we deploy the Gaussian RBF as the mapping kernel.

8.2.5 Overall Classification and Score Fusion

The k Nearest Neighbour classifier $k-NN(k=1)$ is implemented in this study. The $l1$-norm is used as the distance metric since it is observed that it provides better results compared to the $l2$-norm and the cosine angle (Section 6.5.2).

The output scores from the eight classifiers are fused using the mean-rule in order to make the final decision. If a patch is covered by hair it will not be passed to the classifier and an output score of zero will be recorded for that patch (see Figure 8.2).

8.3 Experimental Results

To evaluate the proposed algorithm, we generated a large database of partial faces from the Face Recognition Grand Challenge Ver2.0 (FRGCv2.0) database [164] which consists of 16,028 frontal face images of 466 subjects. Figure 8.1 shows examples from this database showing the significant variations in scale, lighting, partial occlusion and pose. Two interesting examples of the output of the proposed technique for forehead localization are shown in Figure 8.4.

250 subjects were randomly selected and one image from every subject was chosen to build the model mentioned in Section 8.2. Next, half of the remaining images were chosen for testing and the other half were used for training. It was found that the rank-1 identification of the proposed partial face recognition algorithm is equal to 54.10%. This result is encouraging given the amount of variability in the face images, and indicates that facial skin texture may be a source of useful biometric information, even when no landmarks are available.

8.4 Conclusion

We have proposed a novel alignment free approach to deal with the problem of recognizing a face from its partial image. The system is tested on FRGC ver2.0 database and presents promising recognition results for a very challenging problem.

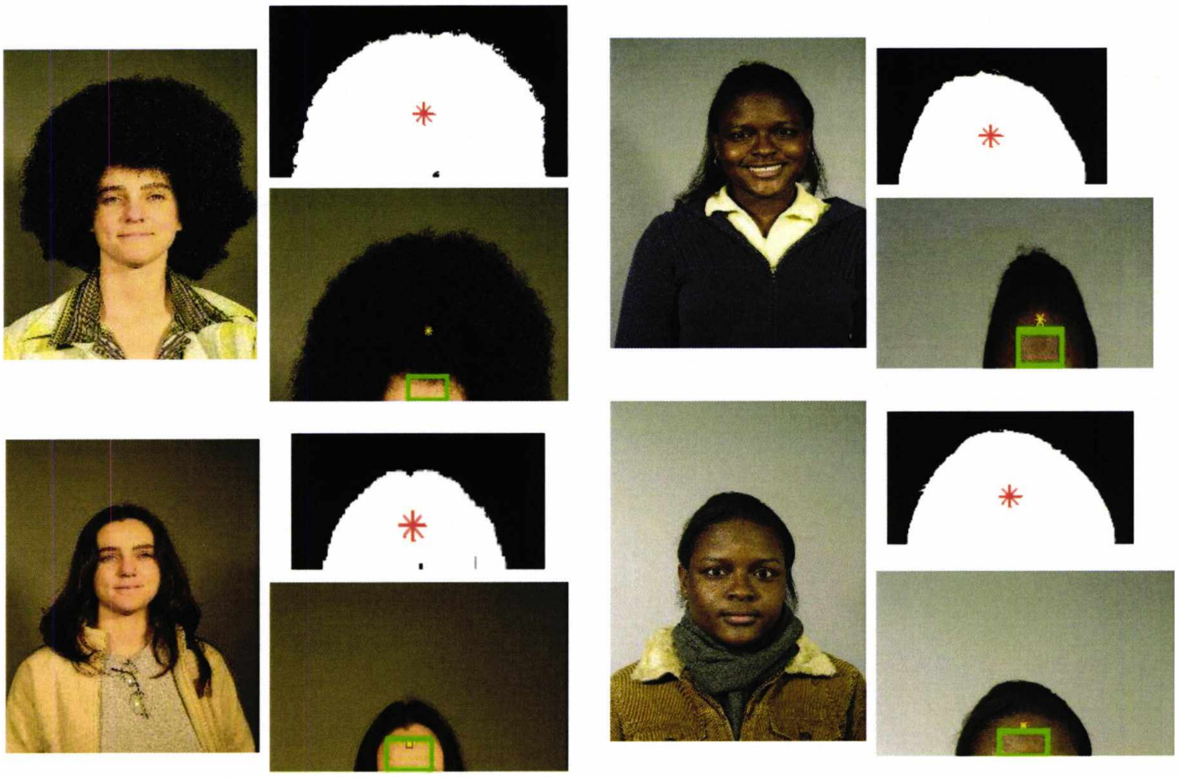


Figure 8.4: Two examples of cropping forehead from the head centroid

Chapter 9

Conclusion and Future work

This section will give conclusions and suggest future research directions in the following two sections, respectively.

9.1 Conclusions

The work presented in this thesis focused on the topic of using skin texture feature for face recognition particularly when the face images are high-dimensional.

The theoretical background is covered in Chapter 2 and Chapter 3 of this thesis. Chapter 2 presents a thorough review of the topic of texture analysis, in particular the numerous applications of texture analysis and those techniques which are used for extracting texture features. It then gives an overview of a number of commonly used classification approaches, with particular attention given to their use in the field of texture classification. Topics covered in Chapter 3 include the challenges of face recognition, the components of a face recognition system and the measures commonly used for assessing the performance of face recognition systems.

Questions Q1 and Q2 in Section 1.2 are answered in Chapter 4. It describes an automatic method for segmenting the facial skin areas. It also shows that the automatic extraction of the facial skin regions requires a precise localization of eye centres and mouth centre. Thus, a new scheme for the precise localization of the eye centres in 2D still images is presented in this Chapter. It is one important subject of this thesis because eye localization (detection) is the fundamental step for the initialization of other methods, mostly face recognition techniques.

The topic of extracting the texture features from the facial skin areas (Q3, Q4, Q5, Q6) is covered in Chapter 5. Pure-skin forehead region in the face images are manually labeled and Gabor and LBP approaches are then used to extract skin texture features. The experiments that are conducted on XM2VTS database indicate that the forehead region alone provides useful biometric information for person recognition and it is not short term biometric trait. The effects of changing the parameters of the Gabor filter-bank and LBP approaches are also investigated and tuned to optimize the recognition accuracy.

To develop an automatic skin-based face recognition system, we need to build Skin/Non-Skin classification technique. Chapter 6 illustrates this classifier which is able to automatically handle the noise in skin areas such as hair covering forehead region, facial hair covering chin region or sometimes dark sun glass covering parts of cheeks. We also in Chapter 6 compare the discriminative capabilities of different facial regions such as forehead, chin, right cheek and left cheek. Since the results suggest that the forehead region yields the highest recognition accuracy, its information was combined with the kernel discriminant analysis KDDA at score-level. The empirical results indicate that the face recognition accuracy can be improved by the explicit exploitation of identity information in pure skin regions. The results of the fused system are comparable to or better than those obtained by the state-of-the-art methods.

An adaptive skin-based method for face recognition is proposed in Chapter 7. The first step is to automatically extract several patches from face images. Then patches that may be obscured by hair, facial hair or spectacles are excluded by using Skin/Non-Skin Classifier. Finally, An adaptive method working on the usable skin patches is applied for face recognition. The experiments on the XM2VTS database have been carried out to demonstrate that this adaptive scheme is able to utilize all available skin texture information for building face recognition system.

Finally, Chapter 8 is devoted to a novel alignment free approach to deal with the problem of recognizing a face from its partial image. The system is tested on FRGC ver2.0 database and presents promising recognition results for a very challenging problem.

9.2 Future Directions

The following issues are suggested for future work:

1. **Skin areas segmentation:** The automatic segmentation of skin areas from facial images is an application for which texture analysis has been shown to be effective. One possible avenue for further research is the use of recent approaches of texture segmentation such as Graph-Cut.

2. **Photometric normalization:** Chapter 8 has presented a novel alignment free approach to deal with the problem of recognizing a face from its partial image.

The proposed algorithm has been evaluated using FRGC ver2 database which has a significant variation in illumination. Thus, it is possible to investigate the photometric rectification algorithms to handle more extreme illumination variations that are exist in this dataset.

3. **Feature extraction:** The system that tested in Chapter 8 on FRGC ver2.0 database presents promising recognition results for a very challenging problem. By extracting other rotation-invariant texture features, it may be possible to further improve this representation, leading to lower overall classification error rates.

4. **Classification:** Exploration of using different types of classification techniques such as Neural Network or Bayesian classifier may lead to improved performance.

5. **Resolution:** An investigation into the effects of images resolution on the overall performance is another avenue for possible future research.

6. **Enhancing face recognition systems:** The field of face recognition is an active and challenging research field. Chapter 6 showed that the global face recognition system could be improved by skin texture features. There exists much potential to improve the results presented in this thesis by fusing the proposed adaptive algorithm (that is in Chapter 7) to several kinds of global face recognition systems.

References

[1] A.F. Abate, M.Nappi, D.Riccio, and G.Sabatino. 2D and 3D face recognition: A survey. *Pattern Recognition Lett*, 28(14):1885–1906, 2007.

[2] F. Ade. Characterization of textures by eigenfilters. *Signal Processing*, 5(5):451 – 457, 1983.

[3] T. Ahonen, A.Hadid, and M.Pietikäinen. Face recognition with local binary patterns. In *European conference on Computer vision, ECCV*, pages 469–481, 2004.

[4] T. Ahonen, A. Hadid, and M. Pietikäinen. Face description with local binary patterns: Application to face recognition. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 28(12):2037–2041, 2006.

[5] H. Akaike. A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19:716–723, 1974.

[6] H. Almuallim and T.Dietterich. Learning with many irrelevant features. In *In Proceedings of the Ninth National Conference on Artificial Intelligence*, pages 547–552, 1991.

[7] A. Amet, A. Ertzn, and A. Eril. An efficient method for texture defect detection: sub-band domain co-occurrence matrices. *Image Vision Comput.*, 18:543–553, 2000.

[8] Y. Amit, D.Geman, and B.Jedynak. Efficient focusing and face detection. *Face Recognition: From Theory to Applications, H. Wechsler, P.J. Phillips, V. Bruce, F. Fogelman-Soulie, and T.S. Huang*, 163:124–156, 1998.

[9] S. Asteriadis, N.Nikolaidis, and I.Pitas. Facial feature detection using distance vector fields. *Pattern Recognition*, 42(7):1388–1398, 2009.

[10] G. Baudat and F.Anouar. Generalized discriminant analysis using a kernel approach. *Neural Computation*, 12(10):2385–2404, 2000.

[11] H. Bay, A.Ess, T.Tuytelaars, and L.Van Gool. Speeded-up robust features (surf). *Comput. Vis. Image Underst.*, 110(3):346–359, 2008.

[12] P. Belhumeur, João P. Hespanha, and D. Kriegman. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 19(7):711–720, 1997.

[13] Y. Ben Salem and S. Nasri. Automatic recognition of woven fabrics based on texture and using svm. *Signal, Image and Video Processing*, 4(4):429–434, 2010.

[14] B.Heisele, P.Ho, and T.Poggio. Face recognition with support vector machines: global versus component-based approach. In *International conference on Computer vision, ICCV*, volume 2, pages 688–694, 2001.

[15] F. Bianconi, A.Fernández, E.González, and F.Ribas. Texture classification through combination of sequential colour texture classifiers. In *CIARP*, pages 231–240, 2007.

[16] B.E. Boser, I. Guyon, and V. Vapnik. A training algorithm for optimal margin classifiers. *Proceedings of COLT*, pages 144–152, 1992.

[17] A.C. Bovic, M. Clark, , and W.S. Geisler. Multichannel texture analysis using localized spatial filters. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 12(1):55–73, 1990.

[18] S. Brubaker, J.Wu, J.Sun, M.Mullin, and J.Rehg. On the design of cascades of boosted ensembles for face detection. *International conference on Computer vision, ICCV*, 77(1-3):65–86, 2008.

[19] C. Burges. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2:121–167, 1998.

[20] T. Burghardt, B.Thomas, P.Barham, and J.Calic. Automated visual recognition of individual african penguins. In *International Penguin Conference, Ushuaia, Tierra del Fuego, Argentina*, 2004.

[21] J. Cai, A. Goshtasby, and C. Yu. Detecting human faces in color images. *International Workshop Multimedia Database Management Systems*, 0:124–131, 1998.

[22] P. Campadelli, R.Lanzaotti, and Gi.Lipori. Eye localization: a survey. *in the book of The Fundamentals of Verbal and Non-verbal Communication and the Biometrical Issue, NATO Science Series*, 2007.

[23] D. Chai and K.N. Ngan. Locating facial region of a head-and-shoulders color image. In *International Conference on Face & Gesture Recognition, FG*, pages 124–129, 1998.

[24] C. Chan, J.Kittler, and K.Messer. Multi-scale lbp histograms for face recognition. In *Int.conf Advances in Biometrics ICB, Korea*, pages 809–818, 2007.

[25] C. Chang and C.Lin. LIBSVM: A library for support vector machines. *ACM Trans on Intelligent Systems and Technology*, 2:1–27, 2011. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

[26] K. Chang, K.Bowyer, S.Sarkar, and B.Victor. Comparison and combination of ear and face images in appearance-based biometrics. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 25:1160–1165, 2003.

[27] R. Chellappa and A. Jain. *Markov Random Fields: Theory and Applications.* Academic Press, 1993.

[28] L. Chen. A new LDA-based face recognition system which can solve the small sample size problem. *Pattern Recognition*, 33(10):1713–1726, 2000.

[29] X. Chen, P.J. Flynn, and K.W. Bowyer. Pca-based face recognition in infrared imagery: Baseline and comparative studies. In *International Workshop on Analysis and Modeling of Faces and Gestures (AMFG), France*, pages 127–134, 2003.

[30] Y. Chen and E. Dougherty. Gray-scale morphological granulometric texture classification. *Optical Engineering*, 33(8):2713–2722, 1994.

[31] C. Chibelushi, J. Mason, and F. Deravi. Feature-level data fusion for bimodal person recognition. In *International Conference on Image Processing and Its Applications*, volume 1, pages 399 -403, 1997.

[32] M. Clerc and S.Mallat. The texture gradient equation for recovering shape from texture. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 24(4):536–549, 2002.

[33] J. Coggins and A. Jain. A spatial filtering approach to texture analysis. *Pattern Recogn. Lett.*, 3(3):195–203, 1985.

[34] T. Cover and P. Hart. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1):21–27, 1967.

[35] I. Craw, D.Tock, and A.Bennett. Finding face features. In *European conference on Computer vision, ECCV*, pages 92–96, 1992.

[36] D. Cristinacce, T.Cootes, and I.Scott. A multi-stage approach to facial feature detection. In *British Machine Vision Association, BMVC*, pages 231–240, 2004.

[37] J. Crowley and F.Berard. Multi-modal tracking of faces for video communications. In *Proceedings of Computer Vision and Pattern Recognition, CVPR*, pages 640–645, 1997.

[38] Y. Dai and Y.Nakano. Face-texture model based on sgld and its application in face detection in a color scene. *Pattern Recognition*, 29(6):1007–1017, 1996.

[39] B.V. Dasarathy. *Nearest neighbor (NN) norms: NN pattern classification techniques.* IEEE Computer Society Press, 1991.

[40] M. Dash and H. Liu. Feature Selection for Classification. *Intelligent Data Analysis*, 1:131–156, 1997.

[41] J. Daugman. Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters. *J. Opt. Soc. Am. A*, 2(7):1160–1169, 1985.

[42] J. Daugman. Complete discrete 2-d gabor transforms by neural networks for image analysis and compression. *IEEE Trans.Acoustics, Speech and Signal Processing*, 36(7):1169–1179, 1988.

[43] C. Ding and I.Dubchak. Multi-class protein fold recognition using svms and neural networks. *Bioinformatics*, 17:349–358, 2001.

[44] J. Doak. An Evaluation of Feature Selection Methods and Their Application to Computer Security. Technical report, University of California, Department of Computer Science, 1992.

[45] P. Dollár, Z.Tu, H.Tao, and S.Belongie. Feature mining for image classification. In *Proceedings of Computer Vision and Pattern Recognition, CVPR*, 2007.

[46] A. Drimbarean and P.Whelan. Experiments in colour texture analysis. *Pattern Recogn. Lett.*, 22:1161–1167, 2001.

[47] S. Du and R.Ward. Wavelet-based illumination normalization for face recognition. In *IEEE Trans. Image Process(2)*, pages 954–957, 2005.

[48] R. Dubes and A. Jain. Random field models in image analysis. *Journal of Applied Statistics*, 16:131–164, 1989.

[49] M. Dubuisson-Jolly and A.Gupta. Color and texture fusion: application to aerial image segmentation and gis updating. *Image Vision Comput.*, 18(10):823–832, 2000.

[50] R.O. Duda, P.E. Hart, and D.G. Stork. *Pattern classification.* Wiley, 2001.

[51] M. Everingham and A.Zisserman. Regression and classification approaches to eye localization in face images. In *International Conference on Face & Gesture Recognition, FG*, pages 441–448, 2006.

[52] Y. Freund and R.Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. In *EuroCOLT*, pages 23–37, 1995.

[53] K.S. Fu. *Syntactic pattern recognition and applications.* Prentice-Hall advanced reference series: Computer science. 1982.

[54] D. Gabor. Theory of communications. *Journal of International Electrical Engineers*, 93:429–457, 1946.

[55] A. Georghiades, P.Belhumeur, and D.Kriegman. From few to many: Illumination cone models for face recognition under variable lighting and pose. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 23(6):643–660, 2001.

[56] R. Gonzales and P.Wintz. *Digital image processing (2nd ed.).* Addison-Wesley Longman Publishing Co., Inc., 1987.

[57] R. Gonzalez and R.E. Woods. *Digital Image Processing.* Prentice-Hall, Inc., 2006.

[58] C. Gotlieb and H.Kreyszig. Texture descriptors based on co-occurrence matrices. *Computer Vision, Graphics, and Image Processing*, 51(1):70 – 86, 1990.

[59] G. A. Granlund. In search of a general picture processing operator. *Computer Graphics and Image Processing*, 8:155–173, 1978.

[60] A. Grossmann and J. Morlet. Decomposition of hardy functions into square integrable wavelets of constant shape. *SIAM Journal on Mathematical Analysis*, 15(4):723–736, 1984.

[61] Y. Guan. Robust eye detection from facial image based on multicue facial information. In *IEEE Inte Conf on Control and Automation*, pages 1775–1778, 2007.

[62] S. Gutta, V. Philomin, and M. Trajkovic. An investigation into the use of partial-faces for face recognition. *Int Conf on Automatic Face and Gesture Recognition, FG*, pages 33–38, 2002.

[63] A. Hafiane, G.Seetharaman, and B.Zavidovique. Median binary pattern for texture classification. *Int.Conf.Image Analysis and Recognition*, pages 387–398, 2007.

[64] R. Haralick. Statistical and structural approaches to texture. *Proceedings of the IEEE*, 67:786–804, 1979.

[65] R. Haralick, K. Shanmugam, and I. Dinstein. Textural Features for Image Classification. *IEEE Transactions on Systems, Man and Cybernetics*, 3(6):610–621, 1973.

[66] M. Hassaballah, T. Kanazawa, and S. Ido. Efficient eye detection method based on grey intensity variance and independent components analysis. *Computer Vision, IET*, 4(4):261 –271, 2010.

[67] X. He, S.Yan, Y.Hu, P. Niyogi, and H.Zhang. Face recognition using laplacianfaces. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 27(3):328 –340, 2005.

[68] G. Healey and L. Wang. Illumination-invariant recognition of texture in color images. *J. Opt. Soc. Am. A*, 12(9):1877–1883, 1995.

[69] G. Heusch, F. Cardinaux, and S. Marcel. Lighting normalization algorithms for face verification. Technical report, IDIAP, 2005. IDIAP-com 05-03.

[70] E. Hjelmås and B.Low. Face detection: A survey. *Computer Vision and Image Understanding*, 83(3):236–274, 2001.

[71] T. Ho, J.J. Hull, and S.N. Srihari. Decision combination in multiple classifier systems. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 16:66 –75, 1994.

[72] C. Hsu, C.Chang, and C.Lin. A Practical Guide to Support Vector Classification, 2000.

[73] C. Hsu and C.Lin. A comparison of methods for multiclass support vector machines. *IEEE Trans. Neural Networks*, 13:415–425, 2002.

[74] R. Hsu, M.Abdel-Mottaleb, and A.Jain. Face detection in color images. In *IEEE Trans on Image Processing*, pages 1046–1049, 2001.

[75] C. Huang, H.Ai, Y.Li, and S.Lao. High-performance rotation invariant multi-view face detection. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 29(4):671–686, 2007.

[76] J. Huang and H.Wechsler. Visual routines for eye location using learning and evolution. *IEEE Trans. Evolutionary Computation*, 4(1):73–82, 2000.

[77] W. Huang and R.Mariani. Face detection and precise eyes location. In *International Conference on Pattern Recognition, ICPR*, pages 4722–4727, 2000.

[78] T. Igarashi, K. Nishino, and S. Nayar. The Appearance of Human Skin. Technical report, Department of Computer Science, Columbia University, 2005. CUCS-024-05.

[79] H. Igehy and L.Pereira. Image replacement through texture synthesis. In *ICIP*, volume 3, pages 186–189, 1997.

[80] P. Jackway and M.Deriche. Scale-space properties of the multiscale morphological dilation-erosion. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 18(1):38–51, 1996.

[81] A. Jain and B. Chandrasekaran. Dimensionality and sample size considerations. In P.R. Krishnaiah and L.N. Kanal, editors, *Pattern Recognition in Practice*, pages 835–855. 1982.

[82] A. Jain and D.Zongker. Feature selection: Evaluation, application, and small sample performance. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 19:153–158, 1997.

[83] A. Jain and F.Farrokhina. Unsupervised texture segmentation using gabor filters. *Pattern Recognition*, 24(12):1167–1186, 1991.

[84] A. Jain and G. Healey. A multiscale representation including opponent color features for texture recognition. *IEEE Trans. Image Process*, 7(1):124–128, 1998.

[85] A. Jain, K.Nandakumar, and A.Ross. Score normalization in multimodal biometric systems. *Pattern Recognition*, 38(12):2270–2285, 2005.

[86] A. Jain and U.Park. Facial marks: Soft biometric for face recognition. In *IEEE Trans. Image Processing*, pages 37–40, 2009.

[87] T.S. Jebara and A.Pentland. Parametrized structure from motion for 3d adaptive feedback tracking of faces. In *Proceedings of Computer Vision and Pattern Recognition, CVPR*, pages 144–150, 1997.

[88] O. Jesorsky, K.Kirchberg, and R.Frischholz. Robust face detection using the hausdorff distance. In *Int.con.on Audio and Video-based Biometric Person Authentication*, pages 90–95. Springer, 2001.

[89] D. Jimenez and J.Alba-Castr. Shape-driven gabor jets for face description and authentication. *IEEE Trans. Information Forensics and Security*, 2:769 –780, 2007.

[90] L. Jin, X-H.Yuan, S.Satoh, J.Li, and L.Xia. A hybrid classifier for precise and robust eye detection. In *International Conference on Pattern Recognition, ICPR*, pages 731–735, 2006.

[91] D.J. Jobson, Z. Rahman, and G.A. Woodell. A multiscale retinex for bridging the gap between color images and the human observation of scenes. *IEEE Trans. Image Process*, 6(7):965 –976, 1997.

[92] I. Jolliffe. *Principal Component Analysis.* Springer-Verlag, 1986.

[93] M. Jones and J.Rehg. Statistical color models with application to skin detection. In *Proceedings of Computer Vision and Pattern Recognition, CVPR*, volume 1, pages 247–280, 1999.

[94] B. Julesz. Visual pattern discrimination. *IRE Trans.Information Theory*, 8(2):84 –92, 1962.

[95] H. Kaizer. A quantification of textures on aerial photographs. Technical report, Boston University Research Lab., 1955.

[96] J. Kamarainen, V.Kyrki, and Heikki Kälviäinen. Invariance properties of gabor filter-based features-overview and applications. *IEEE Trans. Image Process*, 15(5):1088–1099, 2006.

[97] J.-K. Kamarainen. *Feature Extraction Using Gabor Filters.* PhD thesis, Lappeenranta University of Technology, 2003.

[98] A. Karasaridis and E. Simoncelli. A filter design technique for steerable pyramid image transforms. *IEEE Int.Conf.on Acoustics, Speech, and Signal Processing*, 4:2387–2390, 1996.

[99] M. Kasari, J.Parkkinen, T.Jaaskelainen, and R. Lenz. Multi-spectral texture segmentation based on the spectral cooccurrence matrix. *Pattern Anal. Appl.*, pages 275–284, 1999.

[100] S-H. Kim, N-K.Kim, S.Ahn, and H-G Kim. Object oriented face detection using range and color information. Int Conf on Face & Gesture Recognition, FG, pages 76–81, 1998.

[101] T. Kim, H.Kim, W.Hwang, S.Kee, and J.Lee. Component-based lda face descriptor for image retrieval. In *BMVC, Cardiff, UK*, 2002.

[102] S. Kirkpatrick, C. Gelatt, and M. Vecchi. Optimization by Simulated Annealing. *Science, Number 4598, 13 May 1983*, 220, 4598:671–680, 1983.

[103] J. Kittler, M.Hatef, R.Duin, and J.Matas. On combining classifiers. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 20:226–239, 1998.

[104] M. Kölsch and M.Turk. Analysis of rotational robustness of hand detection with a viola-jones detector. In *ICPR (3)*, pages 107–110, 2004.

[105] D. Korycinski, M.Crawford, J. W. Barnes, and J. Ghosh. Adaptive feature selection for hyperspectral data analysis using a binary hierarchical classifier and tabu search. In *Int Symposium on Geoscience and Remote Sensing*, volume 1, pages 297 – 299, 2003.

[106] J. Kovac, P.Peer, and F.Solina. Human skin color clustering for face detection. In *The IEEE Region 8, Computer as a Tool, EUROCON 2003*, volume 2, pages 144–148, 2003.

[107] M. Kudo and J.Sklansky. Comparison of algorithms that select features for pattern classifiers. *Pattern Recognition*, 33(1):25–41, 2000.

[108] S. Kukkonen, H. Kälviäinen, and J. Parkkinen. Color features for quality control in ceramic tile industry. *Optical Engineering*, 40(2):170–177, 2001.

[109] A. Kumar, D.Wong, H.Shen, and A.Jain. Personal authentication using hand images. *Pattern Recogn. Lett.*, 27(13):1478–1486, 2006.

[110] L. Kuncheva. That elusive diversity in classifier ensembles. In *IbPRIA*, pages 1126–1138, 2003.

[111] V.V Kyrki and J. Kamarainen. Simple gabor feature space for invariant object recognition. *Pattern Recogn. Lett.*, 25:311–318, 2004.

[112] M. Lades, J.Vorbruggen, J. Buhmann, J. Lange, C.Malsburg, R.Wurtz, and W. Konen. Distortion invariant object recognition in the dynamic link architecture. *IEEE Trans.Computers*, 42:300–310, 1993.

[113] P. Langley. Selection of relevant features in machine learning. In *The AAAI Fall symposium on relevance*, pages 140–144. AAAI Press, 1994.

[114] A. Lanitis, C.Taylor, and T.Cootes. Automatic face identification system using flexible appearance models. *Image Vision Comput.*, 13(5):393–401, 1995.

[115] R. Lanzarotti. *Facial feature detection and description.* PhD thesis, LAIV Laboratory, Universit degli Studi di Milano, 2003.

[116] K. Laws. Texture energy measures. *Proc. of Image Understanding Workshop*, pages 47–51, 1979.

[117] K.I Laws. *Textured Image Segmentation.* PhD thesis, University of Southern California, 1980.

[118] J-E. Lee, A.Jain, and R.Jin. Scars, marks and tattoos (SMT). In *Biometrics Symposium, BYSYM '08*, pages 1–8, 2008.

[119] L. Lepistö, I.Kunttu, J.Autio, and A. Visa. Rock image classification using non-homogeous textures and spectral imaging. In *WSCG*, 2003.

[120] S. Li, X. Fu, and B. Yang. Nonsubsampled contourlet transform for texture classifications using support vector machines. In *IEEE Int.Conf. Networking, Sensing and Control,ICNSC*, pages 1654 –1657, 2008.

[121] S. Li and A. Jain. *Encyclopedia of Biometrics*. Springer US, 2009.

[122] S. Li and Z.Zhang. Floatboost learning and statistical face detection. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 26:1112–1123, 2004.

[123] Z. Li, D.Lin, and X.Tang. Nonparametric discriminant analysis for face recognition. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 31(4):755–761, 2009.

[124] Z. Li and X.Mao. Emotional eye movement markup language for virtual agents. In *Int.Conf.Autonomous Agents and Multiagent Systems (AAMAS 2010)*, pages 1533–1534, 2010.

[125] S. Liao and A.Jain. Partial face recognition: An alignment free approach. *Int. Joint Conf on Biometrics (IJCB)*, pages 1 – 8, 2011.

[126] S. Liao and A. Chung. Face recognition by using elongated local binary patterns with average maximum distance gradient magnitude. Asian conference on Computer vision, ACCV, pages 672–679, 2007.

[127] S. Liao, X. Zhu, Z. Lei, L. Zhang, and Z. LiS. Learning multi-scale block local binary patterns for face recognition. In *Int.Conf. Biometrics, ICB*, pages 828–837, 2007.

[128] D. Lin and X.Tang. Recognize High Resolution Faces: From Macrocosm to Microcosm. In *Proceedings of Computer Vision and Pattern Recognition, CVPR*, pages 1355–1362, 2006.

[129] W. Lin, M.Chen, K.Widder, Y.Hu, and N.Boston. Fusion of multiple facial regions for expression-invariant face recognition. In *IEEE 9th Workshop. Multimedia Signal Processing, MMSP*, pages 426 –429, 2007.

[130] C. Liu and H. Wechsler. Gabor feature based classification using the enhanced fisher linear discriminant model for face recognition. *IEEE Trans. Image Process*, 11(4):467 –476, 2002.

[131] X. Liu and T. Chen. Geometry-assisted statistical modeling for face mosaicing. In *IEEE Trans. Image Process*, volume 2, pages 883–886, 2003.

[132] D.G. Lowe. Object recognition from local scale-invariant features. In *International conference on Computer vision, ICCV*, pages 1150–1157, 1999.

[133] J. Lu, K.Plataniotis, and A.Venetsanopoulos. Face recognition using kernel direct discriminant analysis algorithms. *IEEE Trans.Neural Networks*, 14(1):117–126, 2003.

[134] Y. Ma, X.Ding, Z.Wang, and N.Wang. Robust precise eye location under probabilistic framework. In *FGR*, pages 339–344, 2004.

[135] T. Mäenpää and M. Pietikäinen. Classification with color and texture: jointly or separately? *Pattern Recognition*, 37(8):1629–1640, 2004.

[136] S. Mallat. A theory for multiresolution signal decomposition: the wavelet representation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 11:674–693, 1989.

[137] B.S Manjunath and W.Y Ma. Texture features for browsing and retrieval of image data. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 18(8):837–842, 1996.

[138] A. Mansfield, J.Wayman, Authorised Dr, D.Rayner, and J. Wayman. Best practices in testing and reporting performance, version 2.01. Technical report, 2002.

[139] J. Mao and A.Jain. Texture classification and segmentation using multiresolution simultaneous autoregressive models. *Pattern Recogn.*, 25(2):173–188, 1992.

[140] A. Mayoue. Biosecure tool - performance evaluation of a biometric verification system. Technical report, 2007.

[141] K Messer and J.Kittler. A region based image database system using colour and texture. *Pattern Recogn. Lett.*, 20(11-13):1323–1330, 1999.

[142] K. Messer, J. Matas, J. Kittler, J. Luettin, and G. Maitre. Xm2vtsdb: The extended m2vts database. In *Int Conf.Audio and Video-based Biometric Person Authentication*, 1999.

[143] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 27(10):1615–1630, 2005.

[144] M. Mirmehdi, X.Xie, and J.Suri. *Handbook of Texture Analysis.* Imperial College Press, 2009.

[145] B. Moghaddam and M.Yang. Learning gender with support faces. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 24(5):707–711, 2002.

[146] A. Muhamad and F.Deravi. Neural networks for the classification of image texture. *Engineering Applications of Artificial Intelligence*, 7(4):381 – 393, 1994.

[147] I.T. Nabney. *NETLAB: Algorithms for Pattern Recognition.* Springer, 2004.

[148] L. Nanni, A.Lumini, and S.Brahnam. Local binary patterns variants as texture descriptors for medical image analysis. *Artificial Intelligence in Medicine*, 49(2):117 – 125, 2010.

[149] P. Narendra and K.Fukunaga. A branch and bound algorithm for feature subset selection. *IEEE Trans. Comput.*, 26(9):917–922, 1977.

[150] H. Neo, C.Teo, and A.Teoh. Development of partial face recognition framework. *Int. Conf on Computer Graphics, Imaging and Visualization (CGIV)*, pages 142–146, 2010.

[151] T. Ojala, M.Pietikäinen, and D.Harwood. A comparative study of texture measures with classification based on featured distributions. *Pattern Recognition*, 29(1):51–59, 1996.

[152] T. Ojala, M. Pietikainen, and T. Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 24(7):971 –987, 2002.

[153] T. Ojala, K. Valkealahti, E. Oja, and M. Pietikäinen. Texture discrimination with multidimensional distributions of signed gray-level differences. *Pattern Recognition*, 34(3):727–739, 2001.

[154] E. Osuna, R.Freund, and F.Girosit. Training support vector machines: an application to face detection. In *computer vision and pattern recognition, CVPR*, pages 130 –136, 1997.

[155] C. Palm. Color texture classification by integrative co-occurrence matrices. *Pattern Recognition*, 37(5):965–976, 2004.

[156] U. Park. *Face Recognition: Face in Video, Age Invariance, and Facial Marks.* PhD thesis, Michigan state University, 2009.

[157] J.R. Parker. *Algorithms for Image Processing and Computer Vision.* Wiley, 1996.

[158] G. Paschos and F.Valavanis. A color texture based visual monitoring system for automated surveillance. *Trans. Sys. Man Cyber*, 29(2):298–307, 1999.

[159] T. Pavlidis and Y. Liow. Integrating region growing and edge detection. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 12(3):225–233, 1990.

[160] A. Pentland. Fractal-based description of natural scenes. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 6(6):661 –674, 1984.

[161] M. Petrou and P.García-Sevilla. *Image processing - dealing with texture.* Wiley, 2006.

[162] P. Phillips. Support vector machines applied to face recognition. In *Advances in Neural Information Processing Systems*, pages 803–809. MIT Press, 1999.

[163] P. Phillips, H.Moon, S.Rizvi, and P.Rauss. The feret evaluation methodology for face-recognition algorithms. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 22(10):1090–1104, 2000.

[164] P. Phillips, P.Flynn, T.Scruggs, K.Bowyer, J.Chang, K.Hoffman, J.Marques, J.Min, and W.Worek. Overview of the face recognition grand challenge. In *Proceedings of Computer Vision and Pattern Recognition, CVPR*, volume 1, pages 947–954, 2005.

[165] P. Phillips, P.Grother, R.Micheals, D.M. Blackburn, E.Tabassi, and M.Bone. Face recognition vendor test 2002. In *IEEE Int. Workshop. Analysis and Modeling of Faces and Gestures (AMFG), Nice, France*, 2003.

[166] P. Phillips, W. Todd, A.O'Toole, P.Flynn, K.Bowyer, C.Schott, and M.Sharpe. Frvt 2006 and ice 2006 large-scale experimental results. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 32(5):831–846, 2010.

[167] PICS. Psychological image collection at stirling pics image database. In *University of Stirling Psychology Department*, 2003.

[168] J-S. Pierrard and T.Vetter. Skin detail analysis for face recognition. In *Proceedings of Computer Vision and Pattern Recognition, CVPR*, pages 1–8, 2007.

[169] M. Pietikäinen, A.Hadid, G.Zhao, and T.Ahonen. *Computer Vision Using Local Binary Patterns.* Springer-Verlag London Limited, 2011.

[170] M. Pontil and A.Verri. Support vector machines for 3d object recognition. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 20:637–646, 1998.

[171] R. Qian, M.Sezan, and K.Matthews. A robust real-time face tracking algorithm. In *IEEE Tran.Image Processing*, volume 1, pages 131–135, 1998.

[172] J. Qin and Z.He. A svm face recognition method based on gabor-featured key points. In *Int. conf. Machine Learning and Cybernetics*, volume 8, pages 5144 –5149, 2005.

[173] T. Randen and J. Håkon Husøy. Filtering for texture classification: A comparative study. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 21(4):291–310, 1999.

[174] N.K. Ratha, J.H. Connell, and R.M. Bolle. Image mosaicing for rolled fingerprint construction. In *International Conference on Pattern Recognition, ICPR*, volume 2, pages 1651 –1653 vol.2, 1998.

[175] T.R. Reed and H. Wechsler. Segmentation of textured images and gestalt organization using spatial/spatial-frequency representations. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 12(1):1 –12, 1990.

[176] Y. Rodriguez, F.Cardinaux, S.Bengio, and J Mariéthoz. Measuring the performance of face localization systems. *Image Vision Comput.*, 24(8):882–893, 2006.

[177] A. Ross and A.Jain. Information fusion in biometrics. *Pattern Recogn. Lett.*, 24:2115–2125, 2003.

[178] A. Ross and A.Jain. Multimodal biometrics: an overview. In *European Signal Processing Conference*, pages 1221–1224, 2004.

[179] A. Ross and R. Govindarajan. Feature Level Fusion Using Hand and Face Biometrics. In *Proc. SPIE*, pages 196–204, 2005.

[180] A. Ross, K.Nandakumar, and A.Jain. *Handbook of Multibiometrics*. Springer, 2006.

[181] S. Roweis and L.Saul. Nonlinear dimensionality reduction by locally linear embedding. *SCIENCE*, 290:2323–2326, 2000.

[182] H. Rowley, S.Baluja, and T.Kanade. Neural network-based face detection. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 20(1):23–38, 1998.

[183] C. Sanderson and K.Paliwal. Information Fusion and Person Verification using Speech and Face information. Technical Report Idiap-RR-33-2002, 2002.

[184] S. Santini and R. Jain. Similarity measures. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 21(9):871–883, 1999.

[185] K. Sato, S. Shah, and J. Aggarwal. Partial face recognition using radial basis function networks. *Int Conf on Automatic Face and Gesture Recognition*, 99:288–293, 1998.

[186] B. Schölkopf, A.Smola, and K.Müller. Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, 10(5):1299–1319, 1998.

[187] M. Schwarz, W.Cowan, and J.Beatty. An experimental comparison of rgb, yiq, lab, hsv, and opponent color models. *ACM Trans. Graph.*, 6(2):123–158, 1987.

[188] L. Semler and L.Dettori. Curvelet-based texture classification of tissues in computed tomography. In *IEEE Trans. Image Process*, pages 2165–2168, 2006.

[189] J.P. Serra. *Image analysis and mathematical morphology*, volume 1. Academic Press, 1982.

[190] M. Shafi and P.Chung. A hybrid method for eyes detection in facial images. *Int.Journal of Electrical, Computer, and Systems Engineering*, pages 231–236, 2009.

[191] L. Shen and L.Bai. A review on Gabor wavelets for face recognition. *Pattern Anal. Appl.*, 9:273–292, 2006.

[192] F. Shih and C.Chuang. Automatic extraction of head and face boundaries and facial features. *Inf. Sci.*, 158:117–130, 2004.

[193] W. Siedlecki and S. Sklansky. A note on genetic algorithms for large-scale feature selection. *Pattern Recogn. Lett.*, 10(5):335 – 347, 1989.

[194] W. J.Sklansky Siedleckiand. On automatic feature selection. *PRAI*, 2(2):197–220, 1988.

[195] E. Simoncelli and J.Portilla. Texture characterization via joint statistics of wavelet coefficient magnitudes. In *IEEE Trans. Image Process*, pages 62–66, 1998.

[196] R. Singh, M. Vatsa, A. Ross, and A Noore. Performance enhancement of 2d face recognition via mosaicing. In *AutoID*, pages 63–68, 2005.

[197] K. Sobottka and I.Pitas. Face localization and feature extraction based on shape and color information. In *IEEE Trans.Image Processing*, pages 483–486, 1996.

[198] A. Sohail and P.Bhattacharya. *Detection of facial feature points using anthropometric face model.*, pages 189–200. 2008.

[199] B. Son and Y.Lee. Biometric authentication system using reduced joint feature vector of iris and face. In *International Conference on Audio and Video-based Biometric Person Authentication, AVBPA*, pages 513–522, 2005.

[200] K. Song, J.Kittler, and M.Petrou. Defect detection in random colour textures. *Image Vision Comput.*, 14(9):667–683, 1996.

[201] V. Struc and N.Pavesic. Illumination invariant face recognition by non-local smoothing. In *Int.Conf.Biometric ID Management and Multimodal Communication, Joint COST 2101 and 2102*, pages 1–8, 2009.

[202] P. Suen and G.Healey. Modeling and classifying color textures using random fields in a random environment. *Pattern Recognition*, 32(6):1009–1017, 1999.

[203] X. Tan, F.Song, Z-H.Zhou, and S.Chen. Enhanced pictorial structures for precise eye localization under incontrolled conditions. In *Proceedings of Computer Vision and Pattern Recognition, CVPR*, pages 1621–1628, 2009.

[204] X. Tan and B. Triggs. Fusing gabor and lbp feature sets for kernel-based face recognition. In *Int.Workshop Analysis and Modeling of Faces and Gestures*, pages 235–249, 2007.

[205] X. Tan and B. Triggs. Enhanced local texture feature sets for face recognition under difficult lighting conditions. *IEEE Trans. Image Process*, 19(6):1635 – 1650, 2010.

[206] J. Terrillon, H.Fukamachi, S.Akamatsu, and M.Shirazi. Comparative performance of different skin chrominance models and chrominance spaces for the automatic detection of human faces in color images. In *Int Conf on Automatic Face and Gesture Recognition*, pages 54–61, 2000.

[207] B. Thai and G. Healey. Modeling and classifying symmetries using a multiscale opponent color representation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 20(11):1224 –1235, 1998.

[208] J. Tou, Y.Tay, and P.Lau. A comparative study for texture classification techniques on wood species recognition problem. In *Int.Con.Natural Computation ICNC(5)*, pages 8–12, 2009.

[209] D. Tsai. Optimal gabor filter design for texture segmentation using stochastic optimization. *Image Vision Comput*, 19:299–316, 2001.

[210] F. Tsalakanidou. *The Handbook of Pattern Recognition and Computer Vision*. Informatics and Telematics Institute, PowerPoint presentation.

[211] M. Tuceryan. Moment based texture segmentation. In *International Conference on Pattern Recognition, ICPR*, pages 45 –48, 1992.

[212] M. Tuceryan and A. Jain. Texture analysis. *Handbook of Pattern Recognition and Computer Vision (2nd Edition), by C.H. Chen, L.F.Pau, and P.S.P.Wang (eds.)*, pages 207–248, 1998.

[213] M. Turk and A.Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991.

[214] M. Unser. Local linear transforms for texture analysis. In *International Conference on Pattern Recognition, ICPR*, volume II, pages 1206–1208, 1984.

[215] M. Unser. Local linear transforms for texture measurements. *Signal Process.*, 11(1):61–79, 1986.

[216] V. Vapnik. *The Nature of Statistical Learning Theory.* Springer, 1995.

[217] P. Viola and M.Jones. Rapid object detection using a boosted cascade of simple features. In *Proceedings of Computer Vision and Pattern Recognition, CVPR (1)*, pages 511–518, 2001.

[218] J. Wang, W.Yau, A.Suwandy, and E.Sung. Person recognition by fusing palmprint and palm vein images based on laplacianpalm representatio. *Pattern Recognition*, 41(5):1514 – 1527, 2008.

[219] P. Wang, M.Green, Q.Ji, and J.Wayman. Automatic eye detection and its validation. *Proceedings of Computer Vision and Pattern Recognition, CVPR Workshop*, page 164, 2005.

[220] X. Wang and X.Tang. Bayesian face recognition using gabor features. In *Proceedings of the 2003 ACM SIGMM workshop on Biometrics methods and applications*, WBMA '03, pages 70–73, 2003.

[221] Y. Wang, T. Tan, and A. Jain. Combining face and iris biometrics for identity verification. In *International Conference on Audio and Video-based Biometric Person, AVBPA*, pages 805–813, 2003.

[222] M. Welling. Support vector machines. Technical report, Department of Computer Science, University of Toronto.

[223] J. Weszka, C. Dyer, and A. Rosenfeld. A comparative study of texture measures for terrain classification. *Trans.Systems, Man and Cybernetics*, pages 269 –285, 1976.

[224] L. Wiskott, J.Fellous, N.Krüger, and C.Malsburg. Face recognition by elastic bunch graph matching. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 19(7):775–779, 1997.

[225] G. Wouwer, P. Scheunders, S. Livens, and D. Van Dyck. Wavelet correlation signatures for color texture characterization. *Pattern Recognition*, 32:443–451, 1999.

[226] J. Wright, A.Yang, A.Ganesh, S.Sastry, and Y.Ma. Robust face recognition via sparse representation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 31(2):210–227, 2009.

[227] J. Wu, S.Brubaker, M.Mullin, and J.Rehg. Fast asymmetric learning for cascade face detection. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 30(3):369–382, 2008.

[228] L. Xu, A.Krzyzak, and C.Y.Suen. Methods of combining multiple classifiers and their applications to handwriting recognition. *Trans.Systems, Man, and Cybernetics*, 22:418–435, 1992.

[229] F. Yang, M.Paindavoine, H.Abdi, and D.Arnoult. Fast image mosaicing for panoramic face recognition. *Journal of Multimedia*, pages 14–20, 2006.

[230] G. Yang and T.Huang. Human face detection in a complex background. *Pattern Recognition*, 27(1):53–63, 1994.

[231] H. Yang and Y.Wang. A lbp-based face recognition method with hamming distance constraint. In *Int.Conf.on Image and Graphics*, pages 645 –649, 2007.

[232] M. Yang, D.Kriegman, and N.Ahuja. Detecting Faces in Images: A Survey. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 24:34–58, 2002.

[233] M. Yang and N.Ahuja. Detecting human faces in color images. In *IEEE Trans on Image Processing*, pages 127–130, 1998.

[234] P. Yang, S.Shan, W.Gao, S.Z. Li, and D.Zhang. Face recognition using adaboosted gabor features. In *IEEE International Conference on Automatic Face and Gesture Recognition*, pages 356–361, 2004.

[235] C. Yao and S.Chen. Retrieval of translated, rotated and scaled color textures. *Pattern Recognition*, 36(4):913–929, 2003.

[236] K. Yow and R.Cipolla. Feature-based human face detection. *Image Vision Comput.*, 15(9):713–735, 1997.

[237] B. Zhang, H.Zhang, and S.Ge. Face recognition by applying wavelet subband representation and kernel associative memory. *IEEE Transactions on Neural Networks*, 15(1):166–177, 2004.

[238] B. Zhang, S.Shan, X.Chen, and W.Gao. Histogram of gabor phase patterns (hgpp): A novel object representation approach for face recognition. *IEEE Trans. Image Process*, 16(1):57–68, 2007.

[239] G. Zhang, X.Huang, S.Li, Y.Wang, and X.Wu. Boosting local binary pattern (lbp)-based face recognition. In *SINOBIOMETRICS, China*, volume 3338, pages 179–186, 2004.

[240] J. Zhang and T.Tan. Brief review of invariant texture analysis methods. *Pattern Recognition*, 35:735–747, 2002.

[241] L. Zhang, R.Chu, S.Xiang, Sh.Liao, and St.Li. Face detection based on multiblock lbp representation. In *Int.Conf. Advances in Biometrics,ICB, Korea*, pages 11–18, 2007.

[242] W. Zhang, S.Shan, W.Gao, X.Chen, and H.Zhang. Local gabor binary pattern histogram sequence (lgbphs): a novel non-statistical model for face representation and recognition. In *International conference on Computer vision, ICCV*, volume 1, pages 786–791, 2005.

[243] Z. Zhang, S.Tulyakov, and V.Govindaraju. Combining facial skin mark and eigenfaces for face recognition. In *Int.Conf. Biometrics, ICB*, pages 424–433, 2009.

[244] D. Zhao, Z.Lin, and X.ang. Laplacian pca and its applications. In *ICCV, Rio de Janeiro, Brazil*, pages 1–8, 2007.

[245] W-Y. Zhao, R.Chellappa, P.Phillips, and A.Rosenfeld. Face recognition: A literature survey. *ACM Comput. Surv.*, 35(4):399–458, 2003.

[246] Q. Zou and G.Nagy. A comparative study of local matching approach for face recognition. *IEEE Trans. Image Process*, 16:2617–2628, 2007.