University of Windsor
Scholarship at UWindsor

Electronic Theses and Dissertations

2014

Interpolation of Low Resolution Images for Improved Accuracy in Human Face Recognition

Abbas Elazhari
University of Windsor

Follow this and additional works at: http://scholar.uwindsor.ca/etd

Part of the Electrical and Computer Engineering Commons

Recommended Citation

Elazhari, Abbas, "Interpolation of Low Resolution Images for Improved Accuracy in Human Face Recognition" (2014). *Electronic Theses and Dissertations.* Paper 5059.

This online database contains the full-text of PhD dissertations and Masters' theses of University of Windsor students from 1954 forward. These documents are made available for personal study and research purposes only, in accordance with the Canadian Copyright Act and the Creative Commons license—CC BY-NC-ND (Attribution, Non-Commercial, No Derivative Works). Under this license, works must always be attributed to the copyright holder (original author), cannot be used for any commercial purposes, and may not be altered. Any other use would require the permission of the copyright holder. Students may inquire about withdrawing their dissertation and/or thesis from this database. For additional inquiries, please contact the repository administrator via email (scholarship@uwindsor.ca) or by telephone at 519-253-3000ext. 3208.

Interpolation of Low Resolution Images for Improved Accuracy in Human Face Recognition

By

Abbas Elazhari

A Thesis
Submitted to the Faculty of Graduate Studies through the
Department of Electrical and Computer Engineering
in Partial Fulfillment of the Requirements for
the Degree of Master of Applied Science at the
University of Windsor

Windsor, Ontario, Canada

2014

© 2014 Abbas Elazhari

Interpolation of Low Resolution Images for Improved Accuracy in Human Face Recognition

by

Abbas Elazhari

APPROVED BY:

A. Azab, Outside Departmental Reader,
Dep. of Industrial & Manufacturing Systems Engineering

M. Khalid, Internal Dept. Reader,
Electrical and Computer Engineering

M. Ahmadi, Advisor,
Electrical and Computer Engineering

Jan. 21st, 2014

Declaration of Co-Authorship and Previous Publications

This thesis includes two original papers that have been previously submitted/published for publication in peer reviewed conferences, as follows:

Publication title and full citation	Publication Status
A. Elazhari, M. Ahmadi. "Interpolation of Low Resolution Images for Improved Accuracy in Human Face Recognition." *IEEE International Conference on Electronics, Circuits, and Systems. ICECS, Abu Dhabi, UAE (*2013): 425-428	Published
A. Elazhari, M. Ahmadi. "Neural Network Based Human Face Recognition of Low Resolution Images."	Submitted

I certify that I have obtained a written permission from the copyright owner(s) to include the above published material(s) in my thesis. I certify that the above material describes work completed during my registration as graduate student at the University of Windsor.

I certify that, to the best of my knowledge, my thesis does not infringe upon anyone's copyright nor violate any proprietary rights and that any ideas, techniques, quotations, or any other material from the work of other people included in my thesis, published or otherwise, are fully acknowledged in accordance with the standard referencing practices. Furthermore, to the extent that I have included copyrighted material that surpasses the bounds of fair dealing within the meaning of the Canada Copyright Act, I certify that I have obtained a written permission from the copyright owner(s) to include such material(s) in my thesis and have included copies of such copyright clearances to my appendix.

I declare that this is a true copy of my thesis, including any final revisions, as approved by my thesis committee and the Graduate Studies office, and that this thesis has not been submitted for a higher degree to any other University or Institution.

I hereby declare that this thesis incorporates material that is result of joint research with Dr. Majid Ahmadi and under his supervision. I am aware of the University of Windsor Senate Policy on Authorship, and I certify that I have properly acknowledged the contribution of other researchers to my thesis, and have obtained written permission from each of the co-authors to include the above materials in my thesis.

I certify that, with the above qualification, this thesis, and the research to which it refers, is the product of my own work.

Abstract

In a wide range of face recognition applications, such as the surveillance camera in law enforcement, it is cannot provide enough resolution of face for recognition. The first part of this research demonstrates the impact of the image resolution on the performance of the face recognition system. The performance of several holistic face recognition algorithms is evaluated for low-resolution face images. For the classification, this research uses k-nearest neighbor (k-NN) and Extreme Learning Machine-based neural network (ELM). The recognition rate of these systems is a function in the image resolution. In the second part of this research, nearest neighbor, bilinear, and bicubic interpolation techniques are applies as a preprocessing step to increase the resolution of the input image to obtain better results. The results show that increasing the image resolution using the mentioned interpolation methods improves the performance of the recognition systems considerably.

To my Family: parents, sisters, and brothers

for their love, encouragement, and support.

To the memory of my brother, Othman, May Allah bless his soul.

Acknowledgements

Fist, and foremost, I would like to give my sincere thanks to my advisor Dr. Majid Ahmadi for his help, guidance, and support throughout this thesis and for his faith in my abilities. His constant interest helped at every stage to enable this thesis to be completed. Also, I would like to thank my committee, Dr. Ahmad Azab and Dr. Mohamed Khalid for the important feedback that helped to enhance the quality of this work.

I would like to thank my colleague, Ashirbani Saha, for her help.

I am always thankful to my family and friends for their love and support.

Table of Contents

Declaration of Co-Authoeship and Previous Publications iii

Abstract v

Acknowledgements vii

List of Tables xi

List of Figures xii

List of Abbreviations xv

1. Introduction 1

1.1. Introduction 1
1.2. Typical Face Recognition System 2
1.2.1. The Acquisition Module 2
1.2.2. The image preparation 3
1.2.3. Image pre-processing 4
1.2.4. The feature extraction module 5
1.2.5. Classification 6
1.3. Face Recognition Applications 6
1.4. Problem of Face Recognition from Low Resolution Image 7
1.5. Previous Works 8
1.5.1. Face recognition from Low Resolution Images 9
1.5.2. Super-Resolution 9
1.6. Objective and Purpose of this Research 12

2. Preprocessing: Interpolation 14

2.1. Introduction 14
2.2. Problem statement 16
2.3. Nearest neighbor(Zero-order) interpolation 17
2.4. Bilinear Interpolation 18

2.5. Bicubic interpolation 19
2.6. Comparison 23
  2.6.1.The frequency response 23
  2.6.2.The Accuracy 24
  2.6.3The Computational cost 25
2.7. Chapter Summary 26

3. Feature extraction 27
3.1 Introduction 27
3.2. Local feature extraction 28
3.3. Holistic Feature Extraction 29
  3.3.1.Principal Component Analysis 31
    3.3.1.1. Calculating the Eigenfaces 32
    3.3.1.2.Limitations and Difficulties 35
  3.3.2. Discrete wavelet Transform 36
  3.3.3. PCA on DWT for face recognition 40
  3.3.4. Local Binary Pattern 42
    3.3.4.1. Parameters of the LBP method 44
    3.3.4.2 Template Matching 45
  3.3.5. Block-based Discrete Cosine Transform 45
3.4. Chapter Summary 50

4. Classification 51
4.1. Introduction 51
4.2 *k*-Nearest Neighbors 53
4.3. Single-hidden Layer FeedforwardNeural Networks (SLFNs) 54
  4.3.1. Single-hidden Layer FeedforwardNeural Networks Model 54
  4.3.2.ELM Based SLFNs Model 56
  4.3.3.Conventional Gradient-based Solution of SLFNs 57
  4.3.4. Extreme Learning Machine (ELM) 59
    4.3.4.1. The Procedures of the Extreme Learning Machine 59
4.4. Chapter Summary 61

5. Experimental Evaluation.. 62
5.1. Introduction.. 62
5.2. Experimental Setup.. 63
5.2.1. Database 64
5.2.2. Low Resolution Operator.. 65
5.2.3. Classification.. 65
5.2.3.1. k-Nearest Neighbor 65
5.2.3.2.Neural Network 66
5.2.3.3. Training-to-Testing-Samples Ratio 67
5.3. Experiments and Discussion.. 69
5.3.1. Recognition of Down-sampled Images without Interpolation........... 69
5.3.1.1. PCA Based DWT Recognition System...................................... 71
5.3.1.2. Local Binary Pattern (LBP) Face Recognition Scheme............ 72
5.3.1.3. Block-Based Discrete Cosine Transform Face Recognition Scheme 73
5.3.1.4. Results and Discussion... 74
5.3.2. Utilizing Interpolation for Recognizing Low Resolution Images...... 75
5.3.2.1. Using k-NN Classifier.. 76
5.3.2.2.Using ELM Neural Network Classifier...................................... 80
5.4. Chapter Summary 81

6. Conclusion 83
6.1. Conclusion 83
6.2. Future Work.. 85

References.. 86

Vita Auctoris... 93

List of Tables

Table 1.1: Face recognition applications: categories and examples [7]. 7

Table 5.1: Specifications of ORL database.. 64

Table 5.2: Varying the number of hidden neurons of ELM neural network: Average recognition rate (%), training time and test time using different numbers of neurons 67

Table 5.3: Recognition rate of BBDCT scheme for different image blocks and image resolutions.. 74

List of Figures

Figure 1.1: Structure of the typical face recognition. 3

Figure 1.2: Histogram equalization. a) Input image with bad illumination. b) The image after histogram equalization. c) and d) are the histogram of images (a) and (b) respectively..4

Figure 1.3: Capturing a face from a surveillance video. a) Surveillance video, b) Facial region [16]. 8

Figure 1.4: Learning-based super resolution: Learn a relationship between LR and HR gallery images in the training phase [16]................................ 11

Figure 1.5: Reconstruction-based Super Resolution [11] 12

Figure 2.1: The proposed face recognition system 15

Figure 2.2: Image interpolation concept. 16

Figure 2.3: Nearest neighbor function. 17

Figure 2.4: Nearest neighbor interpolation. (a) Kernel. (b) Fourier transform. 18

Figure 2.5: Linear interpolation. (a) the kernel function (b) the frequency response. 19

Figure 2.6: Cubic-spline interpolation. (a) Kernel function. (b) The frequency response. 22

Figure 2.7: Bicubic interpolation process. 23

Figure 2.8: Amplitude of the interpolation kernel functions in the frequency domain [29]. 24

Figure 2.9: Interpolation error. (a) the original image. (b) the difference image between (a) and the reconstructed image using nearest neighbor. (c) using bilinear interpolation. (d) using bicubic. 25

Figure 2.10: Reconstructing HR image 68×56 from LR image 34×28 using interpolation. (a) the original image of 34×28 pixels.(b) HR image using NN interpolation. (c) HR image using bilinear. (d) HR image using bicubic. 26

Figure 3.1: A grey scale image and its matrix representation in the image processing and pattern recognition. 28

Figure 3.2: Local feature extraction: a) the input image. b) extracting local (geometric) features. c) Facial vector for classification. 29

Figure 3.3: Holistic feature extraction: a) the input image. b) extracting holistic (global) features. c) Facial vector for classification 30

Figure 3.4: Transforming data from the ordinary space into the principal component space. 33

Figure 3.5: Samples of eigenface images: The mean image of facial images at the top left corner, followed by the top eleven eigenfaces. 34

Figure 3.6: Recognition rate Vs. the number of principal components on ORL database. 35

Figure 3.7: Two dimensional discrete wavelet transform (DWT) for an image: The low pass filter H(.) and the high pass filter G(.) are applied along the rows initially followed by downsampling by a factor of two. This is followed by filtering using H(.) and G(.) 38

Figure 3.8: DWT decomposition of an image at level 2. Details images hold some features in three directions: horizontal, in the left bottom image, diagonal in the right bottom image, and vertical at the top right image. The approximation image LL is at the top left corner. 38

Figure 3.9: Block diagram of wavelet based PCA recognition system. 41

Figure 3.10: The basic LBP operator. 43

Figure 3.11: An Example: the greyscale facial image and the corresponding LBP image. 43

Figure 3.12: Extracting a histogram feature vector in LBP. 44

Figure 3.13: LBP operator. On the left: the basic operator uses only the primary 7 neighbors. On the right: The circular (8,1) and (12,1.5) neighborhoods. 44

Figure 3.14: DCT transform of a facial image. 47

Figure 3.15: Face and its DCT transformation (Low, Mid. and High) regions. 47

Figure 3.16: The block diagram of BB DCT feature extraction method. 48

Figure 3.17: Raster scan (Zig-zag) pattern for choosing DCT coefficients of an image of 8×7. 49

Figure 4.1: The typical face recognition block diagram showing the classification stage. 52

Figure 4.2: Model of an artificial neuron. 55

Figure 4.3: Single layer neural network. 56

Figure 5.1: Structure of face recognition system configurations implemented in the experiments. 63

Figure 5.2: Samples from ORL database. 64

Figure 5.3: Low resolution face image formation model. 65

Figure 5.4: Recognition rate for varied training : testing-samples ratio using k-NN and ELM classifiers. 68

Figure 5.5: Samples for downsampled face image: 70

Figure 5.6: Face recognition block diagram for resolution variation. 70

Figure 5.7: Block diagram of wavelet based PCA recognition system. 72

Figure 5.8: Block diagram of the face recognition system using BB DCT. 73

Figure 5.9: The recognition rate Vs. image resolution for the three feature extraction schemes, PCA based DWT, LBP, and BB DCT. 75

Figure 5.10: Proposed face recognition system: Interpolation for low resolution face recognition. 76

Figure 5.11: RR for reconstructed image from LR of 8×7 using bicubic interpolation for the three feature extraction based recognition schemes. 77

Figure 5.12: Recognition rate for reconstructed image from LR of 8×7 using bilinear interpolation for the three feature extraction based recognition schemes. 78

Figure 5.13: Recognition rate for reconstructed image from LR of 8×7 using nearest neighbor interpolation for the three feature extraction based recognition schemes. 78

Figure 5.14: Comparison between nearest neighbor and bicubic interpolation. a) Original Reconstructing 112×96 image from 14×12 using bicubic and nearest neighbor. 79

Figure 5.15: Block diagram of face recognition using BBDCTand ELM. 80

Figure 5.16: Recognition rate for reconstructed image from 8×7 using the three interpolation techniques. This face recognition scheme uses BBDCTfeature extraction and ELM for classification. 81

List of Abbreviations

FR	Face Recognition (Human Face Recognition)
CCD	Charge Couple Device
CMOS	Complementary Metal-oxide Semiconductor
PCA	Principal Components Analysis
LR	Low Resolution
HR	High Resolution
SR	Super Resolution
LDA	Linear Discriminative Analysis
NN	Nearest Neighbor
2D	Two Dimensional
1D	One Dimensional
LBP	Local Binary Pattern
BBDCT	Block-Based Discrete Cosine Transform
DWT	Discrete Wavelet Transform
k-NN	k-Nearest Neighbors
ELM	Extreme Learning Machine
SLFNs	Single-hidden Layer Feedforward Neural Networks
ANN	Artificial Neural Networks
RR	Recognition Rate

Chapter 1

Introduction

1.1. Introduction

Human face presents a unique feature of human being, and recognizing a person by this feature seems to be an easy task for the human brain. Automatically recognizing human faces, on the other hand, has become an important issue in the last two decades for numerous real life applications. Face recognition (FR) is the process of automatically identifying or verifying a person from his facial image. The most common way to do that is by comparing a given image of an unknown individual called ’probe image’ with a large set of images ’gallery ‘of suspected people, and match it to the most similar image from the gallery. Human face recognition constitutes a very active area of research recently due to two main reasons. The first reason is the wide range of commercial and law enforcement applications, and the second reasons is that the problem of machine recognition of human faces has been attracting researches from different disciplines such as, image processing, neural network, computer vision, pattern recognition, and psychology [1]. Although there are many reliable methods of biometric human identification such as, retinal or iris scans, and fingerprint analysis, these systems need cooperation between the person and the identification system for reliable decisions. Also, the database of this type of recognition systems is difficult to effort. Therefore, face recognition is thought to have great potential to make the automatic recognition of human more convenient. A comprehensive and critical survey of face recognition system can be found in [1], [2], [3].

In real life application where the images might be taken in uncontrolled environment, machine face recognition encounters great difficulties. Many FR challenges have been addressed, such as, poor illumination, pose variation, and sever expression variation. However, in many real-world applications, such as, video surveillance cameras, where it is often difficult to obtain good quality recordings of observed human faces, face recognition is still challenging task. One important factor evaluating the quality of the recordings is the resolution of the images which is usually much lower than the resolution that is needed for the typical face recognition. Therefore, addressing the performance of FR systems when image resolution decreases, as well as attempting to increase the resolution of low-resolution (LR) images is becoming a very crucial demand. This research aims to analyze the performance of different face recognition schemes against the resolution variation, and it attempts to improve the quality of the input images such that better performance can be obtained.

1.2. Typical Face Recognition System

Many face recognition algorithms have proposed, and each has a different approach. However, they all have a similar flowchart as shown in figure 2.1. The next section describes the stage of the recognition system, and examples the processes that are needed in each stage.

1.2.1. The Acquisition Module

The goal of this step is to have a digital image. It can be an individual image or a frame of a video recording. Nowadays, the majority of the images are digital; otherwise, an analogue-to-digital converter is required to provide a suitable image format for the next step where the candidate features are extracted. The most common devices for making 2D digital images are the Charge Couple Device (CCD), and Complementary Metal-oxide Semiconductor (CMOS). The two imaging chip types are equally reliable in most of commercial and scientific applications. In some situations, such as, night or in dark space, Infrared camera is needed and stereo camera that uses two cameras to provide 3D image is required for 3D based face recognition [4].

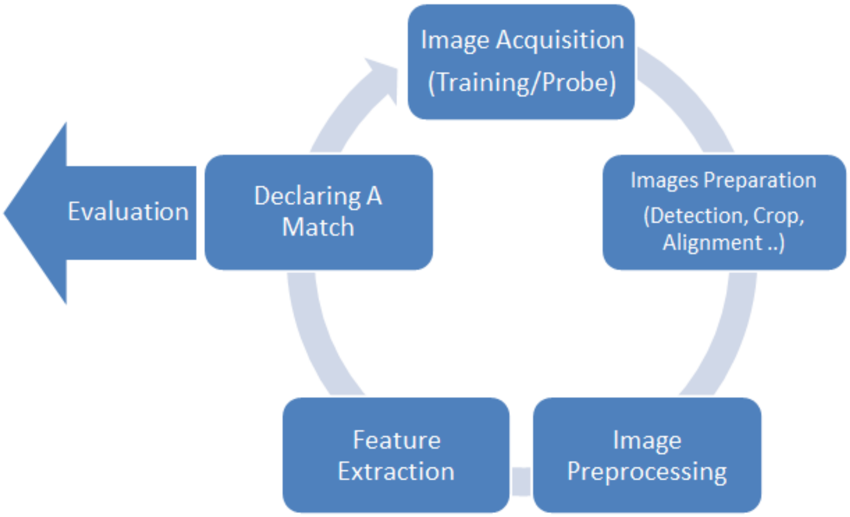


Figure 1.1: Structure of the typical face recognition.

1.2.2. The image preparation

This stage includes some processes that are applied to extract or isolate the facial region of interest from the input image. These processes include face detection, face alignment, and face region cropping. If the input image is of low quality where it becomes difficult to localize the face region, then this stage comes after the preprocessing stage where the input image quality is enhanced. Indeed, there are many methods for face detection [5]. One of the most common methods is to locate the facial region based on the color of human skin. This approach is based on the fact that the human skin color has different lighting intensity (luminance), yet it has the same Chroma. Because of this fact, other components in the image, which do not belong to the skin, can be effectively removed. Another technique uses mathematical and statistical models such as, Principal Components Analysis (PCA) to locate the face in the image. This method create manifold for the images that contain faces in a set of training images. Other methods for face detection can be found in [5], [6].Face Detection, however, is not in the scope of this research. The output image from this stage in many cases includes the facial region and parts of the background which is not needed for face recognition rather it affect the performance negatively; thus facial region needs to be cropped in order to have a more discriminative power. Subsequently, alignment of facial images is crucial in order to obtain an acceptable recognition rate. Alignment of images is required to minimize the variations in face images that are not directly related to the actual differences in faces. Therefore, the classifier would be able to compare features from the testing image with the corresponding features from the gallery images. That is, to obtain more discriminant features, multiple linear transformations can be learned from corresponding local models which are structured by aligning the gallery images with the pre-partitioned reference set.

1.2.3. Image pre-processing

In this stage, certain processes are applied in order to enhance the quality of the image. One can summarize the most common pre-processing steps as follows.

- **Histogram equalization**. It is needed to enhance image quality and to improve face recognition performance for an image that has high brightness or darkness when images are taken in unconstraint environment. Histogram equalization improves the recognition results by flattening the histogram of pixel intensities of the images as shown in figure 1.2.

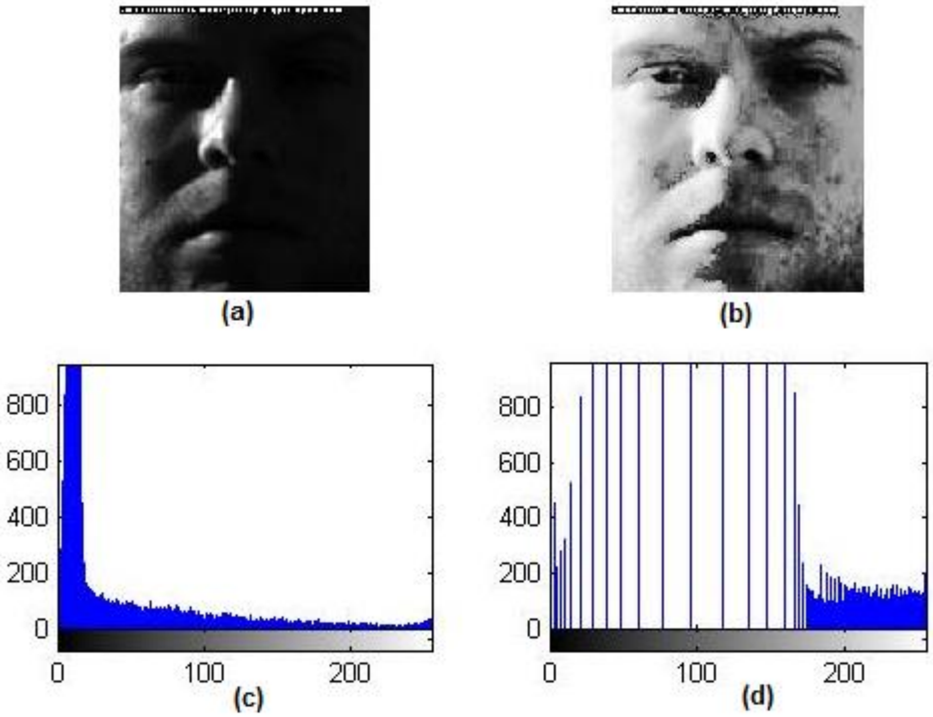


Figure 1.2: Histogram equalization. a) Input image with bad illumination. b) The image after histogram equalization. c) and d) are the histogram of images (a) and (b) respectively.

- **Image size normalization**. In some face recognition applications where the face images are treated as a whole, the input image needs to be adjusted to a default size at which the face recognition system obtains the best result.
- **Median filtering.** Acquired images sometimes have some noise either from the camera or from the atmosphere. In this case, median filter is applied to clean the image and keeps the sufficient details.
- **High-pass filtering**. This kind of preprocessing is applied for recognition systems where feature extractors are based on the local features and contours. This kind of preprocessing is usually needed when the input images are in a good quality.
- **Translational and rotational normalizations**. In order to provide high accuracy in the standard face recognition system, face images supposed to be frontal views of faces; however, in many real-life scenarios, the subject is not cooperative with the acquisition system. In this case, preprocessing stage is needed to determine and normalize the rotation, or to shift the face position to retrieve the frontal view.
- **Illumination normalization**. The pixel value reflects the brightness on the unit area at that location on the facial region. Indeed, holistic feature extraction-based face recognition schemes accuracy is influenced immediately by the pixel values. In such face recognition configurations, accuracy of recognition can be degraded when face images are taken under different illuminations which in turn, change significantly the entries of the face image array that represents the face in the system. Therefore, illumination normalization is crucial in order to impose uniform lightning on the image.

1.2.4. The feature extraction module

After processing the face image to obtain a good quality and position, it is presented to the next stage where the candidate features are extracted. There are several feature extraction algorithms available, and choosing the appropriate one is an open issue. There are two main types of feature extractors: geometric or local, and holistic or global feature extractor. The first one extracts geometrical and structural facial features such as, the shapes of the eyes, nose, mouth, chin, and the distances between them. The second type extracts pixel-based features from the whole image. This process will be discussed in details in chapter three.

1.2.5. Classification

After extracting the most discriminative features from the facial image, a feature vector is represented, and is fed to the classification process. In this classification module, the facial representation is matched and compared to all face image representations stored in the face library (database) in the training phase. The unknown image is matched to the closest image in the face space, and its class is declared decision. The training set is a set of face images that is used in training phase of the face recognition process. This process is needed for adjusting (optimize) parameters of the feature extractor and the classifier. Using a small number of training samples can cause inaccurate classification. On the other hand, using a very large database could cause overfilling if the samples of a certain class are very similar. Also, it increases the time consumption in most of the recognition system. Some face recognition configurations don’t need training phase. After classifying the input "unknown" image, it is added to its class samples in the library for later comparisons.

The classification process will be discussed in chapter four.

1.3. Face Recognition Applications

As the old people saying `Necessity is the mother of invention', the main reason face recognition has attracted so much research attentions recently is its potential in so many authorities and commercial applications. This is due the fact that smart systems and devices are becoming ever closer to people’s daily lives, and companies over the globe are racing to be at the top of this kind of technologies to gain more costumers’ interest. Although some of these techniques are not publicly available for proprietary reasons, many others have been incorporated into commercial systems. A comprehensive review of the newly enlarged application scenarios of face recognition can be found in [7]. In this book, T. Huang, et al., have listed ten of the representative commercial face recognition companies, their techniques for face detection, the face features they extract, and face similarity comparison methods. Furthermore, they grouped the scenarios where face recognition is applied into ten categories, and they have given a detailed explanation about each category. We summarized the most important categories and examples of each category in the table 1.1.

Category	Example
Access control	facility access, smart kiosk and ATM, computer access, online examinations access, online database access
Security	Park monitoring, neighborhood watching, terrorist warning
Law enforcement	Suspect tracking and investigation, crime stopping and suspect alert, identifying cheats and casino undesirables, criminal face retrieval and recognition, welfare fraud
Surveillance system	Audience scanning, Terrorist warning, CCTV control, detecting VIP
Smart cards	user authentication, Stored value security
Face ID	Immigration, national ID, Driver license, voting registration, passport
Multimedia	Face-based search, face-based video segmentation and summarization, event detection
Face databases	Face indexing and retrieval, automatic face labeling, face classification
Human-computer interaction	Proactive computing, interactive gaming

Table 1.1: Face recognition applications: categories and examples [7].

1.4. Problem of Face Recognition from Low Resolution Image

The performance of a face recognition mainly depends on a variety of factors that might influence the quality of the face representation, such as, illumination, facial pose, aging, face expression, hair, facial wear, and facial image scaling. Based on these challenges, one may classify the face recognition into two categories depending on the cooperation of the user. The first category is the user cooperative. In this case, the user is willing to present him/herself to the camera or the computer in many applications, such as, e-passport, and physical access control. The result is controlled-environment face recognition where the face is presented in a proper way, for example, frontal image in a high resolution, controlled lighting, normal face expression. The non-cooperative user face recognition is most commonly seen in surveillance applications as shown in figure 1.3. In such applications, the user is unaware of being recognized by the system. Consequently, uncontrolled-environment recognition is applied. This type of face recognition application suffers from the challenges mentioned above including the fact that the objective face is located at a distance from the camera, which produces the face

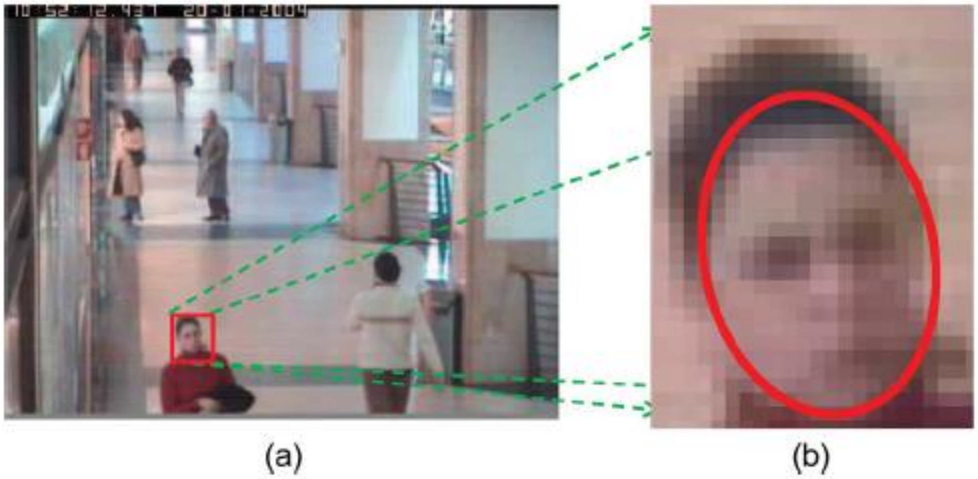


Figure 1.3: Capturing a face from a surveillance video. a) Surveillance video, b) Facial region [16].

image to be in a low resolution. There are other applications in between the two categories mentioned above where the user presents him/herself to the camera but it still does not provides the appropriate image to the classification system. This image might have a good lighting and normal face expression, but it mostly suffers from low quality when the user stands far away from the camera.

Face recognition systems can obtain promising results when using high- resolution (HR) frontal images, but face recognition at a distance, such as, in in surveillance situation, is still challenging. The face regions of images acquired in such situations are usually small and have low quality. Therefore, addressing this problem is very crucial. There have been a few studies and reports addressing the face recognition of low resolution images.

1.5. Previous Works

This section presents the literatures of the first part of our work in which the influence of the image resolution is demonstrated. Also, it presents the literatures of the second part of our work which is the techniques that proposed to improve the quality of the low resolution image for face recognition. The most competitive technique for increasing the image resolution beside the image interpolation is the Super Resolution (SR). The principles and the difficulties of this method are presented in a followed section.

1.5.1. Face recognition from Low Resolution Images

Currently typical face recognition systems usually require face images with more than 50 pixels between the eyes. Due to the high demand for recognizing human faces from low resolution images, many works have been done to address face recognition from smaller images. Experimental studies [8] showed that minimum face image resolution between 32×32 and 64×64 is required for existing algorithms. Boom et al in [9] use PCA and LDA based face recognition and claim that their approach can give good results on 19×17 pixels. Wang, et. al, [10] the relation between face recognition rate and face image resolution using two global face recognition algorithms(PCA and LDA) on the AR-face database where a threshold resolution of 64× 48 is observed. Also, the work in [11] demonstrates that the LBP features are discriminative and robust over a range of facial image and it outperforms Wavelet transformer at resolutions down to 14×19 pixels. In [12] it is claimed that PCA/LDA-based system are not very sensitive to resolution and still give good results at resolutions as low as 32×32 pixels. They also claimed that the number of PCA components depends much more on the dataset that is used than on the resolution. Lastly, Baker, et. al., in [13] claimed that the resolution used for performing face recognition resolution should be less than the resolution that is required to localize the facial landmarks.

We notice that these works do not give enough insight about the impact of the image resolution on different face recognition systems. A few feature extraction methods are presented in this literatures and the impact of the resolution variation on this methods is demonstrated separately. Therefore, part of our work demonstrates in an informative way the performance of different and popular face recognition systems against the resolution variation of the training and testing images. The experiments are designed in a way for comparison results and remarks to be obtained.

1.5.2. Super-Resolution

Super Resolution (SR) refers to the process of creating a high spatial resolution image than what is afforded by the acquisition system through preprocessing step using one or more training samples from low or high resolution observations of the same class. Therefore, SR techniques aim to upsample the image, by that increasing the spatial frequency can be obtained. As well, they work on removing the aliasing and blurring degradations that arise during the image capture. In general, there are two categories of super-resolution techniques: learning-based and reconstruction-based super resolution.

The methods in the first category learn the mapping between LR images and the corresponding HR images using a certain model as shown in figure 1.4. As a result, the reconstructed SR images are close to the HR images that belong to the same subject and far away from others. There are many learning-based face SR algorithms methods proposed in the last decade [14], [15]. The SR method of Zou and Yuen in [16] introduces a data constraint which helps clustering the constructed SR images with the corresponding HR images where identity information constraint about the subject is also used to improve the recognition accuracy. In [17], [18] Biswas, et al., proposed an approach using multidimensional scaling to improve the matching performance of LR images from HR gallery. Their method finds a transformation matrix so that the distance between transformed features of LR images is as close as possible to the corresponding HR images. Moreover, class information is also used to make sure that the distance is small between data from the same class. As well, another learning-based method is proposed in [19]. This method utilizes kernel PCA to build a prior model for frontal face images. It obtains higher-order correlations present in face images by nonlinearly mapping the face images to a higher-dimensional feature space and performing PCA on the feature space. Single-Image Super-resolution is also presented in some works. In [20], [21] given a single low-resolution input image and a database of several high-resolution images, they obtain a high-resolution output using contourlet transform. This method requires fewer training examples than other learning-based super-resolution methods. But it demands a set of HR training images. The reconstruction-based super resolution methods, however, compares LR and HR facial images without explicitly constructing SR images. They rather find a coherent feature space where the correlation of LR and HR is maximum, and then they compute the mapping from LR to HR in this feature space [22].

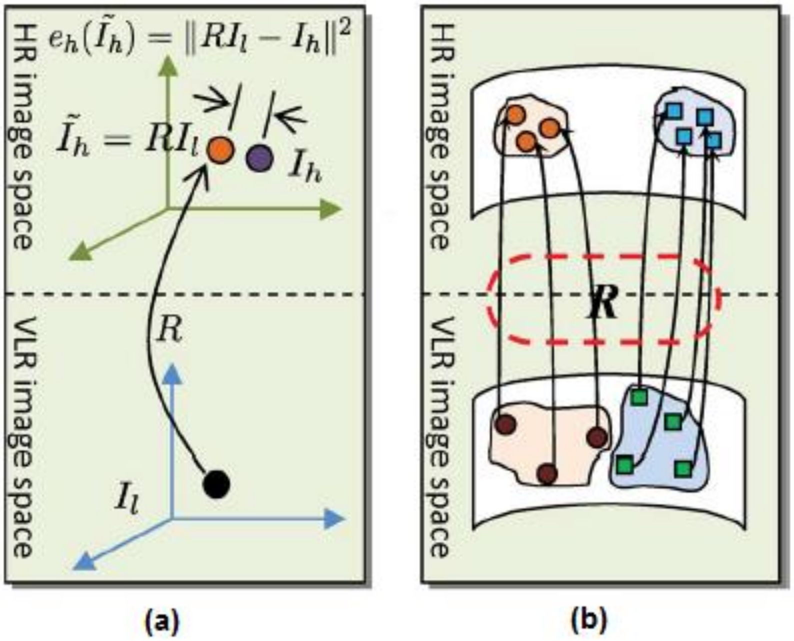


Figure 1.4: Learning-based super resolution: Learn a relationship between LR and HR gallery images in the training phase [16].

Another type of reconstruction-based SR is applied for multi-frame of a subject is available. For a set of LR frames of a scene, this method attempts to interleave these images using a certain type of interpolation or regression method to produce a HR image. Figure 1.5 shows illustrates multi-frame interpolation concept. This method, however, demands a set of LR images of a scene, event or a person that have a sub-pixel movement for better results [23]. This kind of data is not affordable in most of the real-life face recognition scenarios.

From the above discussion, we conclude the main limitations and difficulties of super resolution for face recognition as follows. Most of SR methods, specially the learning-based ones, demand a set of HR images in order to improve the special resolution of a LR probe image. Moreover, those methods assume that this set of images should be taken in a controlled environment, which is not the case in many scenarios of where face recognition is needed. Another difficulty related to the learning-based SR is that a large database of the subjects is needed for estimating the relationship between the HR and the LR sets. In reconstruction-based SR, especially when it performs interleaving for estimating new pixel values, artifacts in the super resolution image is highly likely to occur. These artifacts might be accepted for commercial applications, but for face recognition it cause noise in the face manifold, which impact the recognition accuracy negatively. Besides, when multi-frame SR is applied as in figure 1.5, sub-pixel shift in the LR training images should be obtained. This kind of images can be afforded from video frames that are taken in without disturbance. All those restrictions make the reliability of the super resolution techniques for many applications questionable. The face databases are often taken in a controlled environment. The question we ask is that if one encounters a low-quality input image, can it be enhanced using a low quality database images.

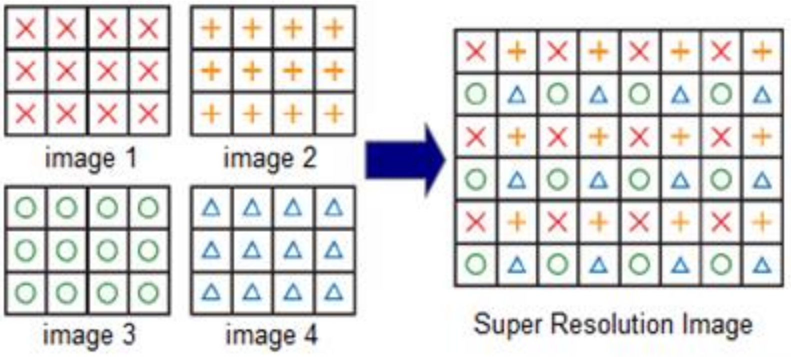


Figure 1.5: Reconstruction-based Super Resolution [11]

Image interpolation, on the other hand, is still competitive technique to increase the special resolution of LR images. The advantages of this technique are that it can be applied on a single image without using gallery set of LR or HR images. Also, it is extremely fast, and it does not influenced by the size of the face database. As well, it produces images with no artifacts, so a stable performance can be obtained. In this work provide a comprehensive study of recognizing faces from low resolution images, and we show to what interpolation can improve the recognition performance of low resolution input images.

1.6. Objective and Purpose of this Research

The objective of this research effort is to improve the performance of the face recognition system when the input image is of low resolution. In this technique, images in both training and testing are considered to be of low-resolution. In our proposed human face recognition system the LR images are enhanced by applying interpolation technique as a preprocessing step before the feature extraction stage. Different Interpolation techniques, namely, nearest neighbor (NN), bilinear and bicubic are used, and their performance is compared with each other. For feature extraction, three methods are used, namely, Principal Component Analysis, Discrete Wavelet Transform, and Block-Based discrete Cosine Transform. For classification, we use two effective classifiers: k-nearest neighbor and Extreme Learning Machine-based neural networks. This goal is accomplished in two experiments as follows:

1. In the first experiment, the impact of the image resolution on the accuracy of recognition different recognition systems is demonstrated. The database images are down-sampled from the original size down to very low resolution, and the recognition rate is reported at each resolution.
2. In the second part, interpolation techniques are applied to increase the resolution of downsampled version of our database. The results will be shown using all the feature extractors, and the classifiers will be used with the appropriate features. Improvement in the recognition rate is expected as higher resolution images are reconstructed.

Chapter 2

Preprocessing: Interpolation

2.1. Introduction

The typical face recognition system consists of three main stages: image preparation, feature extraction, and classification. In fact, the real-life applications of face recognition are more challenging since the face images can be degraded due to many factors. In this case, the need for preprocessing step is crucial. The face recognition system proposed in this thesis seeks to classify images in low quality in term of pixel resolution. Therefore, interpolation is used as a preprocessing step to compensate the lack in the image pixels, so better face representation can be fed to the feature extractor and then to the classifier. This process takes place before the feature extraction in the recognition system, as shown in figure 2.1.Some comprehensive surveys of existing interpolation methods can be found in [24], [25], [26].

The simplest interpolation method is nearest neighbor since no complex mathematical computation is needed. Bilinear interpolation uses a linear approximator to estimate an unknown value between two known values. Besides that, there are many functions that are proposed to achieve 2D nonlinear interpolation [25], such as, sinc function and Lagrange interpolation. However, these approaches have several limitations especially when it takes into consideration the fact that any given image is accounted as a non-stationary signal. The sinc function suffers from slow rate decay, and it is not designed for non-stationary signals, which make it impractical for many applications. The polynomial approaches including Langrage interpolation, on the other land, do not guarantee that the sequence of the interpolated function will converge uniformly to the original desired function. Also, even if the sequence of the interpolations may converge to the original function, using the derivatives of the interpolated signal can be extremely inaccurate. The third limitation of the polynomial interpolators is that they are considered as a global operator. That is, when the function to be interpolated varies rapidly in some parts of the area of interest, the effect of this on the polynomial interpolants would be spread everywhere [27]. The above considerations lead us to use piesewise interpolator with a limited degree. The category of piesewise polynomials is called spline. Spline functions can alleviate the above mentioned difficulties and it can minimize the least square error of the desired function as will be shown later in this chapter.

In this work, the B-spline function is used for representing the basis kernel of our bicubic interpolation. Many literatures [24], [25], [27], [30] have proven that this kernel is superior for interpolating any given non-stationary signal locally and without artifacts.

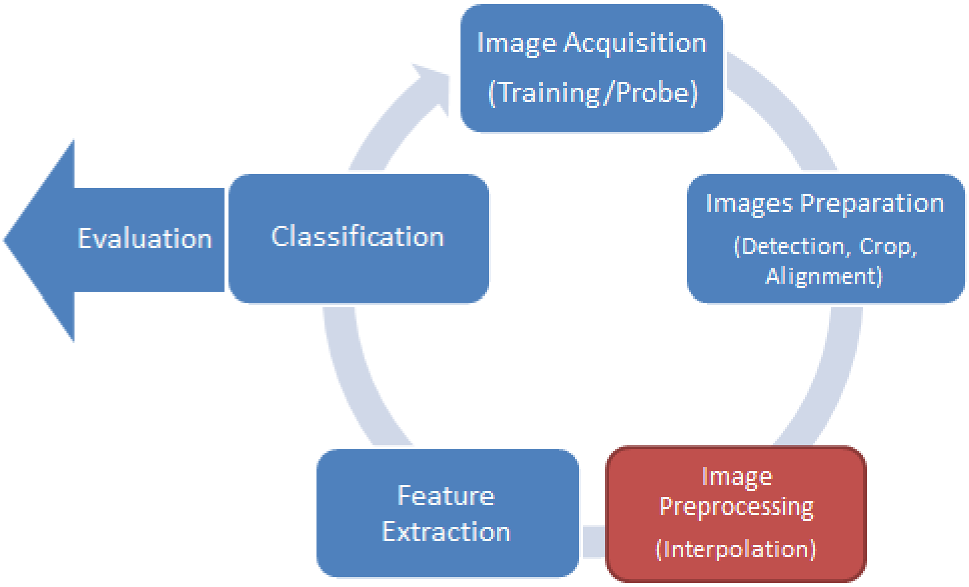


Figure 2.1: The proposed face recognition system

2.2. Problem statement

Any given digital image $g\mathrm{d}(n,m)$ can be considered as an equally spaced 2D sampled signal. This 2D signal corresponds to a higher-rate sampled signal $g\mathrm{a}(x, y)$, which refers the ideal representation of the continuous scene. The image pixels are assumed to be ideal samples taken on a rectilinear grid, *i.e.*, $g\mathrm{d}[n,m] = g\mathrm{a}(n\Delta \mathrm{X},m\Delta \mathrm{Y})$, where $\Delta_x, \Delta_y$are the sample lengths in the two coordinates.The goal is to estimate values of the image $g\mathrm{a}(x, y)$ at arbitrary (x, y) locations that are not existed in the input image, as shown in figure 2.2. That is, we rather resample the input digital image so the output image should provide more information about the facial contents, which helps the classifier make a decision more precisely. Since the input signal of the interpolator is a digital image, and we need to obtain another digital image in higher resolution, this kind of interpolation application can be considered as resampling or up-sampling. The interpolation for resampling is the process of estimating the intermediate values of a sampled event of higher frequency usually from sampled signal of lower frequency. Two-dimensional interpolation is easily accomplished by performing 1D interpolation in each orientation.

In this chapter, the three interpolation schemes of different degrees, namely, nearest neighbor, bilinear, and bicubic are introduced, and they are discussed in terms of the mathematical derivation and the relative performance.

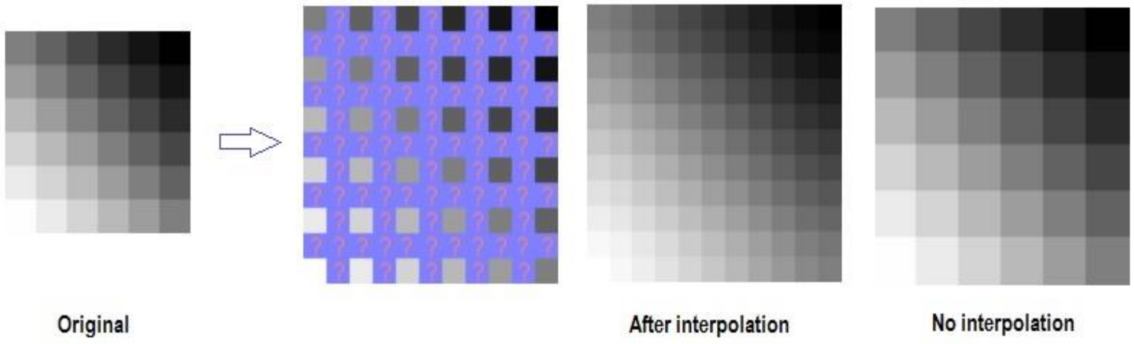


Figure 2.2: Image interpolation concept.

2.3. Nearest neighbor (Zero-order) interpolation

The nearest neighbor (NN) is the simplest method of interpolation, and it can be given the following definition: for interpolation at a given location, use the value at the nearest sample of that location. A mathematical description [28] of 1D interpolation can be given as:

$$\hat{g}(x) = g\left[round\left(\frac{x}{\Delta_x}\right)\right] = g_d[n]|_{n=round(\frac{x}{\Delta_x})} \tag{2.1}$$

Which is considered as an integer shift invariant. The kernel of 1D NN interpolation can be determined as:

$$h(x) = h\left[round\left(\frac{x}{\Delta_x}\right)\right] = \begin{cases} 1, & -1/2 \leq x/\Delta_x < 1/2 \\ 0, & \text{otherwise.} \end{cases} \tag{2.2}$$

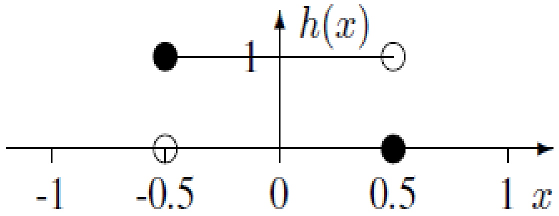


Figure 2.3: Nearest neighbor interpolation function.

Thus, 2D interpolation, using the NN interpolator with the same “rounding up” convention as in 1D, can be written as:

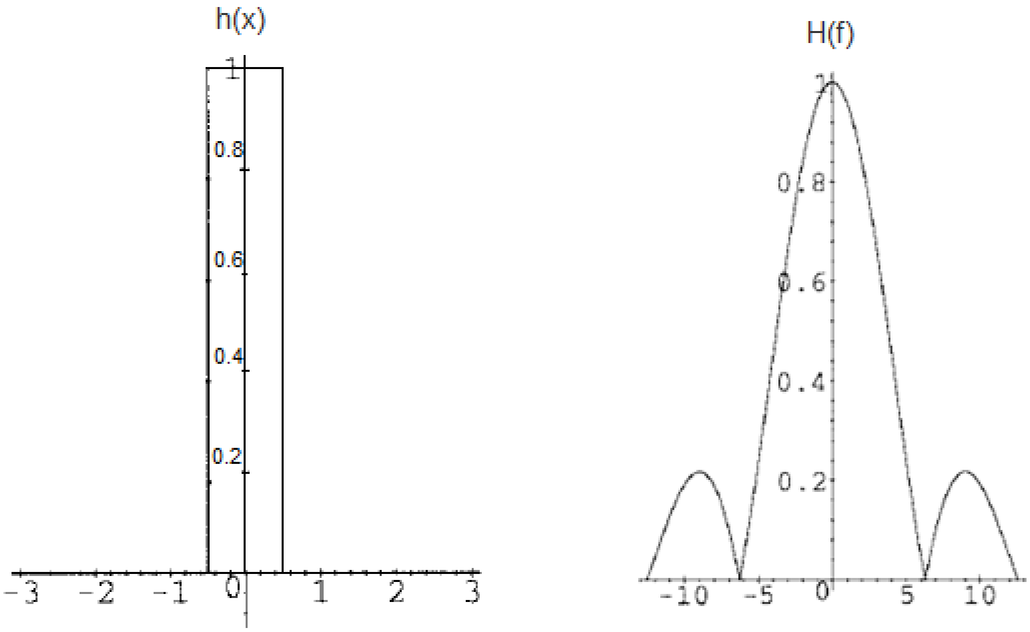
$$g(x,y) = g\left[round\left(\frac{x}{\Delta_x}\right), round\left(\frac{y}{\Delta_y}\right)\right] \tag{2.3}$$

Where x and y are the location where the interpolation is needed, and Δ_x, Δ_y are the sample width (the pixel wide) in the horizontal and the vertical directions, respectively. It is clear that the nearest neighbor is a DC-constant interpolator. Figure 2.4shows the kernel function and the Fourier spectrum of the nearest neighbor kernel [27]. From the figure, this kind of kernel can pass components of relatively high frequency. Therefore, applying nearest neighbor method for image interpolation will cause strong aliasing at the output image [25]. More details about the performance of the NN interpolation will be shown in section 2.6.

2.4. Bilinear Interpolation

Bilinear interpolation is the result of applying two 1D linear interpolations for a two-dimensional signal, and is considered as the second fastest interpolation method. In this method, for a given two known points, an in-between unknown point can be estimated by projection from the location of that unknown on the line that connects those two neighbors. The corresponding interpolation kernel can be determined as a triangle function:

$$h(x) = \begin{cases} 1 - |x|, & 0 \leq x < 1 \\ 0, & \text{elsewhere.} \end{cases} \tag{2.4}$$



(a) (b)

Figure 2.4: Nearest neighbor interpolation. (a) Kernel. (b) Fourier transform.

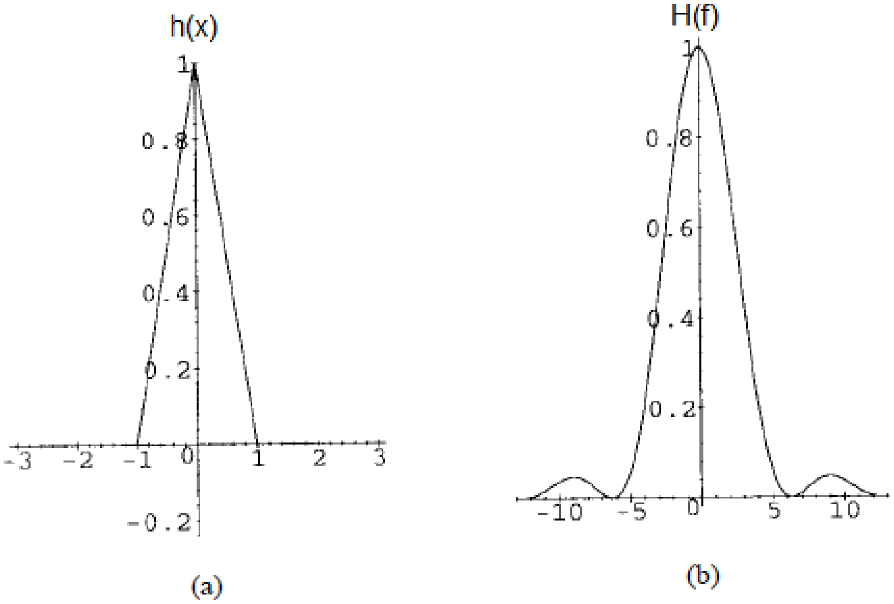


Figure 2.5: Linear interpolation. (a) the kernel function (b) the frequency response.

Therefore, the linear interpolator has a kernel of the first order. This kernel represents a low-pass filter H(f) in the frequency domain as shown in figure 2.5. We notice that the sidelobes in the stopband in figure 2.5.b are below 10%, which is low compared to the NN interpolation, but is still considerable. Therefore, the linear interpolation attenuates both of the high-frequency components and the aliasing of the data of frequencies beyond the cutoff point into the low frequencies [29], [30].

2.5. Bicubic interpolation

Cubic interpolation uses a kernel that designed to maximize the accuracy in a relatively limited level of computational effort. This section shows how to derive the B-spline kernel for cubic interpolation.

The general interpolation form can be given as [29]:

$$g(x) = \sum_k c_k u(\frac{x-x_k}{h}) \tag{2.5}$$

Where h represents the sampling increment, the x_k's are the interpolation nodes, u is the interpolation kernel, and g is the interpolation function. c_k's parameters that depend upon the pixel values in the image window. In cubic interpolation, a signal subinterval of four piecewise is needed at every interpolation location. That is, for a given subintervals (- 2, -

l), (- 1, *0)*, (0, l), and (1, 2), the interpolation kernel outside the interval (- 2**,** 2) is zero; thus, the number of data samples (pixels) used to obtain the interpolation function in (2.5) is four. The interpolation kernel must be symmetric, which means that u must have the form:

$$u(s) = \begin{cases} A_1|s|^3 + B_1|s|^2 + C_1|s| + D_1 & 0 < |s| < 1 \\ A_2|s|^3 + B_2|s|^2 + C_2|s| + D_2 & 1 < |s| < 2 \\ 0 & 2 < |s| \end{cases} \tag{2.6}$$

This kernel assumes that the value $u(0) = 1$, and $u(n) = 0$ for n is any nonzero integer value. This condition, fortunately, imposes considerable reduction in the computation of this process. Considering the sampling increment h, which is, in our case of digital images, refers to the pixel size, and the difference between the interpolation nodes $x_j$and x_k can be represented by $(j-k)h$, then (2.5) can be given as:

$$g(x_j) = \sum_k c_k\, u(j-k) \tag{2.7}$$

$u(j-k)$ is zero unless when j=k, then (2.7) equals toc_j. That yields to $c_j = f(x_j)$. Which means that $c_k{}'$s in (2.5) can be replaced by the pixel values. This remark, indeed, provides an important computational improvement for this interpolation process since there is no need for computing those parameters. Now, from (2.6), there are eight unknowns that need to be calculated. Using the condition that the interpolation kernel has a continuous first derivative, a set of equations can be derived as [29]:

$$\begin{gathered} 1 = u(0) = D_1 \\ 0 = u(1^-) = A_1 + B_1 + C_1 + D_1 \\ 0 = u(1^+) = A_2 + B_2 + C_2 + D_2 \\ 0 = u(2^-) = 8A_2 + 4B_2 + 2C_2 + D_2 \end{gathered} \tag{2.8}$$

From the fact that the variable u is continues from the nodes 0,1 and 2, other three equations can be provided as,

$$\begin{gathered} -C_1 = u'(0^-) = u'(0^+) = C_1 \\ 3A_1 + 2B_1 + C_1 = u'(1^-) = u'(1^+) = 3A_2 + 2B_2 + C_2 \\ 12A_2 + 4B_2 + C_2 = u'(2^-) = u'(2^+) = 0. \end{gathered} \tag{2.9}$$

The above seven equations are not enough to obtain the eight unknown coefficients in the cubic kernel u, and one more condition is needed to obtain a unique solution. According to Key in [29], If interpolation function g(x) in (2.7) agrees with the first three terms of the Tylor series for f , where f is the image intensity representation, that provides the final condition of the interpolation kernel: $A_2 = -\frac{1}{2}$. This condition is the only choice to provide third-order precision. Using this constraint, the cubic convolution interpolation kernel will take the following form:

$$u(s) = \begin{cases} \frac{3}{2}|s|^3 - \frac{5}{2}|s|^2 + 1 & 0 < |s| < 1 \\ -\frac{1}{2}|s|^3 + \frac{5}{2}|s|^2 - 4|s| + 2 & 1 < |s| < 2 \\ 0 & 2 < |s| \end{cases} \tag{2.10}$$

To determine $g$for all x in the interval [a, b], the values of c_k's for k = - 1 , 0, 1, , N+1 are needed. For k = 0, 1,2,..,N,c_k=$f(x_k)$, which is the intensity pixel value at the location x_k. According to Key in [29], for k = -1 and for k = N+1, the values assigned to $c_k$are:

$$c_{-1} = f(x_2) - 3f(x_1) + 3f(x_0) \tag{2.11}$$

$$c_{N+1} = 3f(x_N) - 3f(x_{N-1}) + f(x_{N-2}) \tag{2.12}$$

Therefore, the cubic interpolation function when $x_k < x < x_{k+1}$ can now be written as:

$$g(x) = \frac{c_{k-1}(-s^3+2s^2-s)}{2} + \frac{c_k(3s^3-5s^2+2)}{2} + \frac{c_{k+1}(-3s^3+4s^2+s)}{2} + \frac{c_{k+2}(s^3-s^2)}{2} \tag{2.13}$$

The spacial kernel function and the frequency spectrum of B-spline cubic interpolation is shown in figure 2.6[29]. The two-dimensional cubic interpolation function (bicubic) is a separable extension of the 1D interpolation function. It can be accomplished by performing one-dimensional interpolation in each dimension and applying another interpolation in the orthogonal direction that passes through the interpolated point across the four 1D interpolations. This can be given mathematically as:

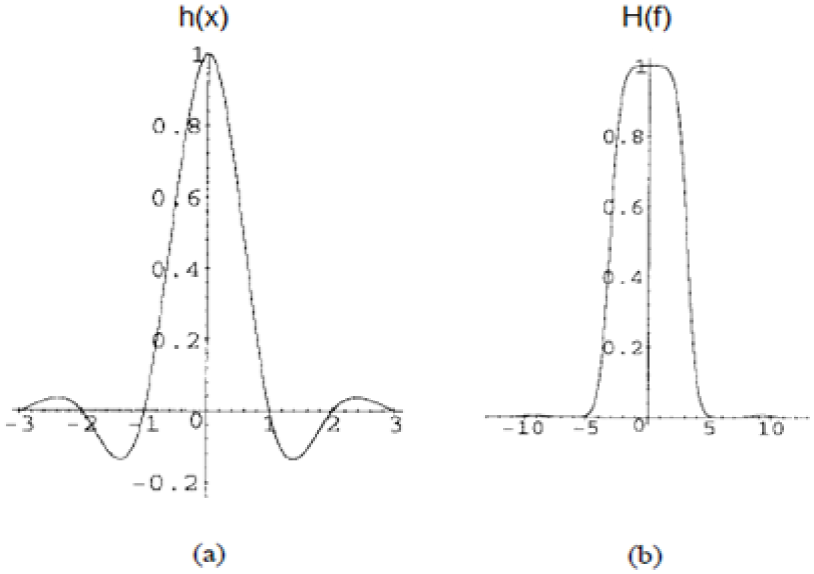


Figure 2.6: Cubic-spline interpolation. (a) Kernel function. (b) The frequency response.

$$g(x,y) = \sum_{l=-1}^{2}\sum_{m=-1}^{2} c_{j+l,k+m} u(\frac{x - x_{j+l}}{h_x}) u\left(\frac{y - y_{k+m}}{h_y}\right) \qquad (2.14)$$

Where u is the interpolation kernel of (2.4) and h_x and h_y are the x and y coordinate sampling increments. Within the kernel, the c_{jk}'s are given by $c_{jk} = f(x_j, y_k)$ which is the intensity pixel values of the image.Figure 2.7 illustrates how to estimate an intensity value at a location (x,y) within a 4×4 window using bicubic interpolation.

Remark: Generally speaking, large kernel sizes were found to be superior to smaller interpolation masks. Although modern computers are able to process a huge amount of data in real-time, fast methods might be required for online interpolation of image sequences or films [31]. In this work, however, we use interpolation schemes in order to increase the resolution of the image that located at a distance from the camera for some applications of recognizing identities in low enforcement. Such applications are usually processed offline. Thus, small difference regarding the time consumption can be ignored.

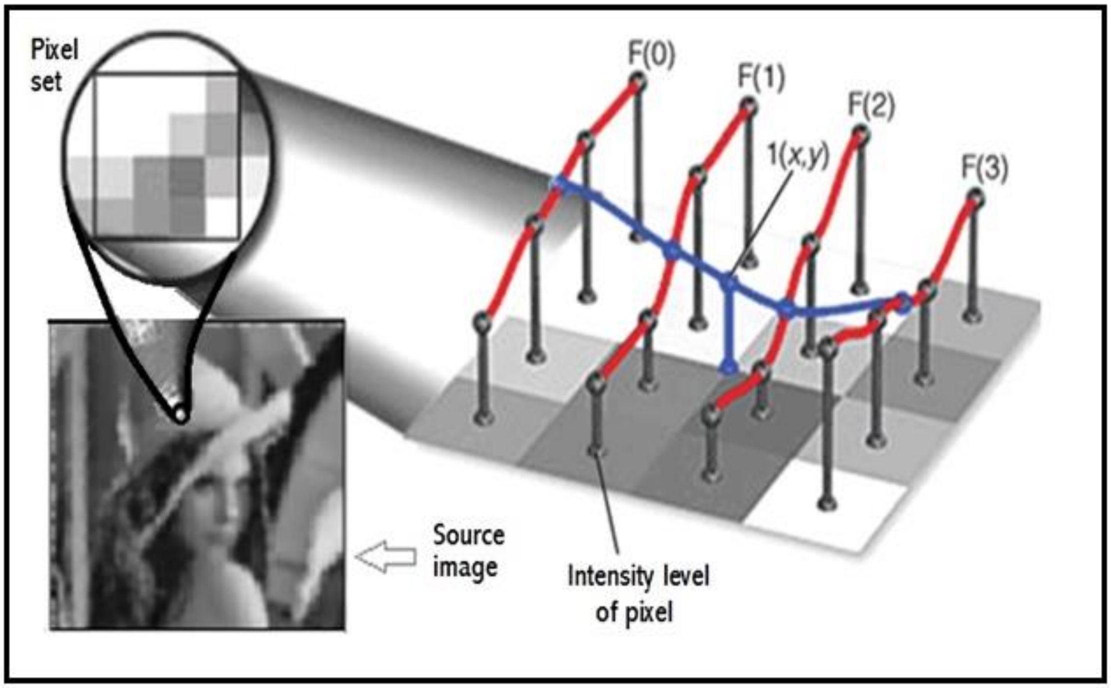


Figure 2.7: Bicubic interpolation process.

2.6. Comparison

In this section, the accuracy, the spectral properties, and the cost of the three convolution interpolation algorithms will be compared with each other.

2.6.1. The frequency response

The amplitude spectra of the nearest-neighbor, linear interpolation, and cubic convolution interpolation kernels are shown in figure 2.8[29]. For frequencies from 0 to $4\pi/h$, the frequency response of the ideal interpolation kernel is a unit step function which has the value of one for frequencies between $-\pi/h$ and $+\pi/h$, and zero elsewhere. That means that this kernel passes every frequency component of that limited band function without change providing higher details in a small scale. On the other hand, deviations from the ideal response in the interval from 0 to $+\pi/h$ cause a loss of the high frequency information. In image data, the loss of high frequency information causes the image to appear blurred. From the figure we notice that cubic and linear interpolation provide the best approximation of the ideal spectrum, yet they still lose high frequency information which causes the image to appear blurred. In contrast of that, deviations from the ideal spectrum beyond the shaded area causes aliasing, which is provided by the nearest neighbor method [25], [29], [30].

2.6.2. The Accuracy

To address the accuracy of the three techniques, we compare the reconstructed image using each scheme with the original image. In order to do that, the “error” images are obtained by subtracting the reconstructed images from the original image, and taking absolute values, as shown in figure 2.9. The original 112×92image is downsampled by factor of 6 to produce a 19×16 image. The downsampled image is then reconstructed (upsampled) using the three interpolation schemes to the same size as the original image. When nearest neighbor interpolation is used, the error image in figure 8(b) and the error image of bilinear and bicubic are in figure 8(c,d) respectively. The absolute black intensity in the error image corresponds to zero error, on the contrast of the Maximum intensity (white), which corresponds to the maximum absolute error. The most noticeable observation is that the nearest neighbor error image contains more white elements than in the other methods, which means that it provides the highest error. As well, bicubic provides darker error image than bilinear although it is not easily noticeable. We can calculate the error more precisely by summing all the greyscale pixel values of the error images. The sum of the cubic error image is 45681, and in bilinear is 48811, and 55405 in the nearest neighbor. This means that the error of bilinear is higher than bicubic, and the maximum error is obtained by nearest neighbor.

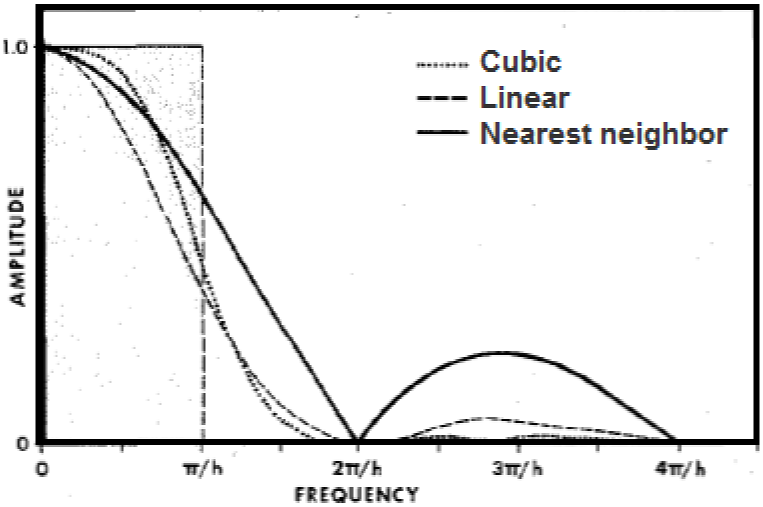


Figure 2.8: Amplitude of the interpolation kernel functions in the frequency domain [29].

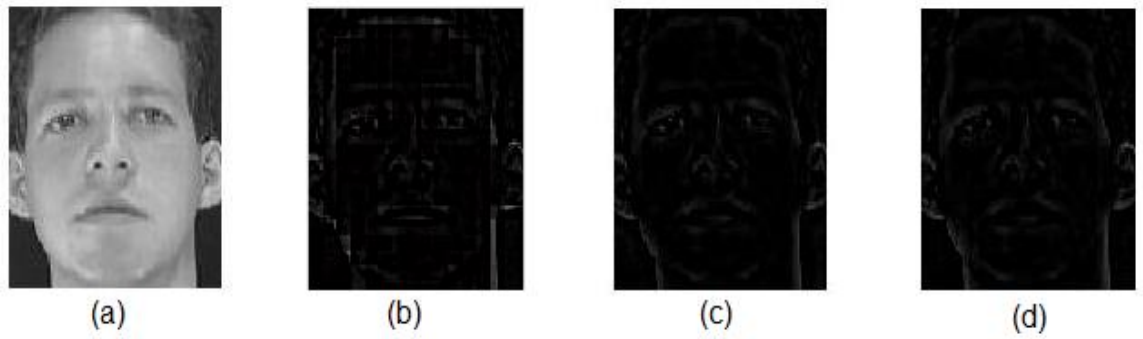


Figure 2.9: Interpolation error. (a) the original image. (b) the difference image between (a) and the reconstructed image using nearest neighbor. (c) using bilinear interpolation. (d) using bicubic.

2.6.3 The Computational cost

Another important consideration when comparing numerical procedures is their computational efficiency. Among the three schemes demonstrated in this work, the nearest-neighbor algorithm is the simplest, since no arithmetic operations are necessary. The interpolated sample point is directly replaced by the value of the nearest pixel from the original image. The second in simplicity is bilinear interpolation; it needs two linear interpolations to be achieved, which involves two sample points for each new interpolated point. In order to interpolate one point, two additions and one multiplication are needed. Bicubic interpolation, however, is achieved by five basic cubic interpolations, which needs four sample points for each interpolation point. Nine additions and eight multiplications are performed for each interpolated point [31], [32], [33].

An example in figure 2.10 shows the performance of the three interpolation schemes. From the figure, we notice that nearest neighbor suffers from high aliasing since no change is added to the interpolated image rather than magnifying the existed pixel of the original image. When bilinear and bicubic are used (Fig. 1.10 c, d respectively), however, the reconstructed images showmore details although it suffers from some blur. We also notice that the image of bicubic is slightly better than the image of bilinear.

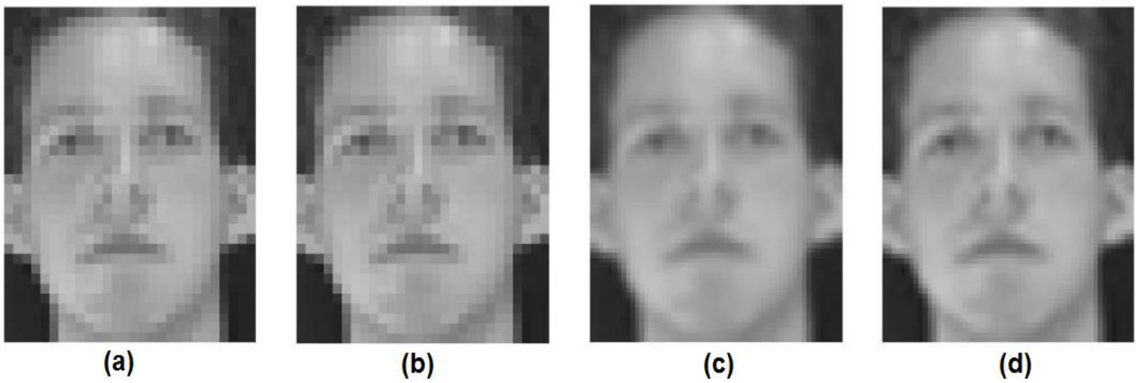


Figure 2.10: Reconstructing HR image 68×56 from LR image 34×28 using interpolation. (a) the original image of 34×28 pixels.(b) HR image using NN interpolation. (c) HR image using bilinear. (d) HR image using bicubic.

2.7. Chapter Summary

In this chapter presents the preprocessing step in the proposed face recognition system. In this step, different interpolation techniques are presented in order to increase the resolution of low resolution input images. Three methods are used for this purpose, namely, nearest neighbor, bilinear, and bicubic interpolations. These methods are presented in terms of mathematical computation and performance. For cubic interpolation, B-spline kernel is used since it shows superiority for reconstructing images from low resolutions with no artifacts. Generally, bicubic performs better than bilinear, but with more computational cost. Nearest neighbor, however, is the cheapest, yet it produces a sever aliasing in the interpolated image.

Chapter 3

Feature extraction

3.1 Introduction

The aim of this chapter is to describe one of the most important stages in the face recognition system. In this stage we transfer the ordinary representation of the facial image from a large matrix that contains the pixel values of the images as entries to a smaller vector of applicable features. This process is called feature extraction or feature selection stage. Feature extraction is crucial for designing a face recognition system with high recognition rate and low computational complexity. This is because the facial image that obtained from the acquisition system contains high number of features (pixels) even if it has a low resolution. For example, a small image of 20×20 pixels is basically represented to the face recognition system as vector of 400 components (features), which is still considered as a high dimensional vector in the face space. An example of 10×8 image is shown in figure 3.1. A simple method for classification is to use Euclidean distance to measure the correlation between face images, which can be achieved by considering the N×N image as a vector in $\Re^{N^2}$ space. Unfortunately, direct calculation of this distance measure is rather expensive. Moreover, if we discard the computational cost, this method might be used for detecting a facial region in the image. However, its accuracy for face recognition is questionable. Indeed, using all the facial features in concert not only increases the complexity of the system, but it also affects the recognition process negatively.

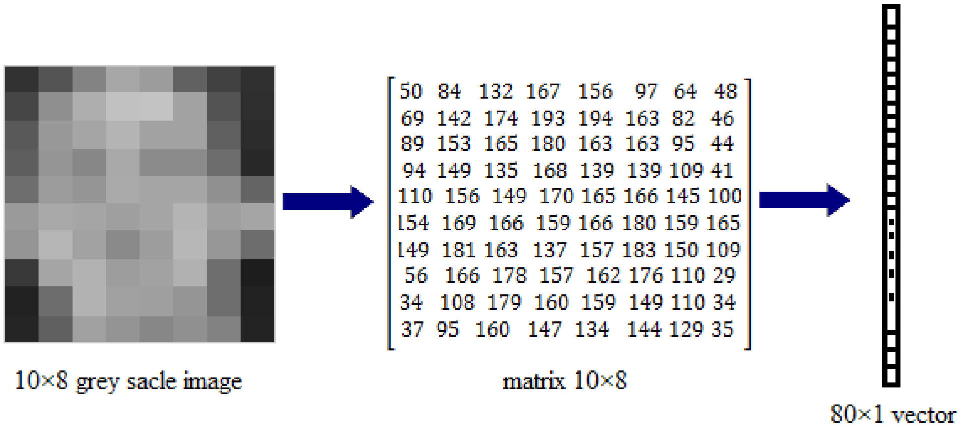


Figure 3.1: A grey scale image and its matrix representation in the image processing and pattern recognition.

Unfortunately, not all feature extractors are efficient for recognition of human faces. One of the unwanted types of features is the ‘mutual information’, which has the same value in different classes. Therefore, the idea of the feature extraction process is to represent the facial image in a proper form before is sent to the classifier. This representation should hold robust features that make the face as discriminative as possible.

Feature extraction methods can be categorized, according to the type of the selected features, to constituent-based or local feature based, and face-based, which also might refer to as, global feature-based or holistic method.

3.2. Local feature extraction

In local feature-based methods [34], [35], the shape, contour, and the dimensions of facial local features, such as, eyes, nose, mouth, as well as, the relationship between these features are used to represent the facial image as shown in figure 3.2. The main advantage of these types of features is that they are robust against the change in illumination and lighting [36]. The effectiveness of this method relies mainly on the accuracy of detecting and localizing these features on the facial region. That means, inaccurate localization of those features might cause a huge error in the classification process. This issue, unfortunately, poses a significant challenge in real life applications

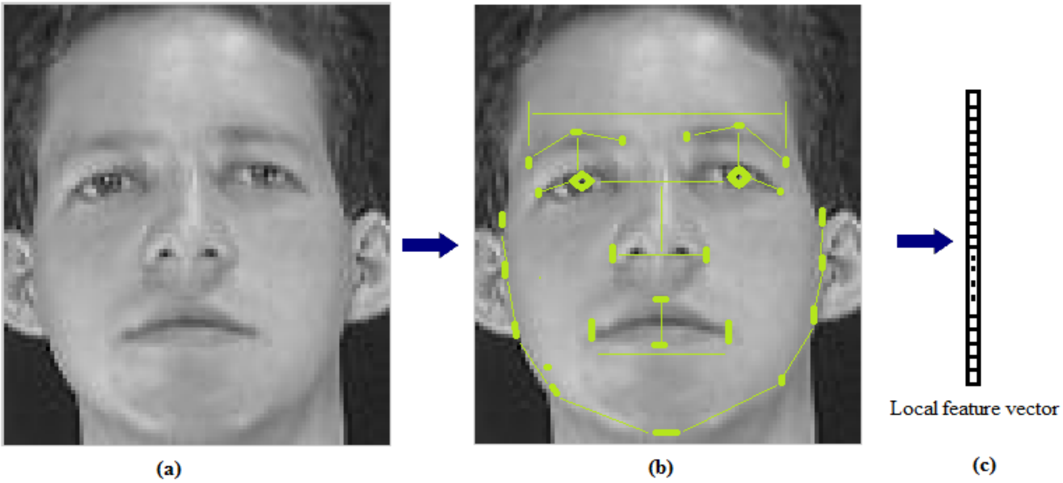


Figure 3.2: Local feature extraction: a) the input image. b) extracting local (geometric) features. c) Facial vector for classification.

where the facial images come with different types of degradations, which make this method uncommon and less reliable.

Therefore, in this thesis, we use some effective holistic feature (face-based) extraction method in which, features are extracted from the facial region in a pixel based regardless of their location on the face image.

3.3. Holistic Feature Extraction

The face-based (holistic) approach [37, 38, 39, 40], attempts to define the facial image as a whole and does not consider the local features. It treats the image as a two-dimension pattern of intensity using underlying statistical regularities. In other words, this type of methods takes into consideration every single pixel in the image regardless of its location on the facial image. These features show more robustness and reliability against many face recognition challenges, such as, resolution degradation, scale and pose variation [36].

There are many holistic descriptors, and choosing the appropriate one for a certain application is an open issue. Many factors should be taken into consideration when choosing the face descriptor. Ideally, the face descriptor should be efficient in terms of providing high intra-class variance (high discriminability), robust against illumination

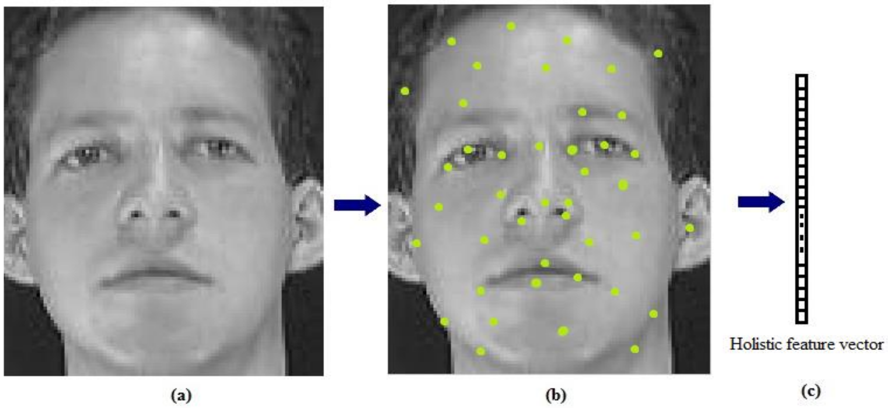


Figure 3.3: Holistic feature extraction: a) the input image. b) extracting holistic (global) features. c) Facial vector for classification

alternating, subjects aging and other factors. As well, the face descriptor should be easy to compute, i.e., with less computational cost and time consumption [41].

Based on the above remarks, some efficient holistic descriptors are adopted for face representation in this thesis. As well, since we are dealing with images of different resolutions, some factors related to these methods should be adjusted as the image size gets smaller. One of the first and most common face descriptors is the Principal Component Analysis (PCA) or eigenface method [37]. This method chooses some features according to their variances over the training image samples. The second method uses wavelet transform [42-49] to select some components within a specific band of frequency as candidate features for face description. This method needs to be combined with dimensionality reduction, such as, PCA, for better performance. The Local Binary Pattern (LBP) operator [50],is one of the best texture descriptors, which is also used for feature extraction. It converts the input image from the intensity scale into a new binary representation. Then, the histogram of the binary image is used for face representation. The last holistic method we use in our work is adapted Discrete Cosine Transform, called: Block-Based Discrete Cosine Transform [51]. The facial image in this method is divided into blocks, taking into consideration the size of the image when we choose the number of image blocks. The feature vector is then constructed using selected DCT components from each block.

The next section provides more details and analysis about these methods.

3.3.1.Principal Component Analysis

Turkand Pentland [37] developed a popular face recognition system using Principal Component Analysis (PCA).This work was inspired by Sirovich and Kirby [52] when they used Karhunen-Loeve [53] transform to represent human faces. In this method, faces are represented by a linear combination of weighted eigenvector, called “eigenfaces”. Therefore, PCA can be considered as an information theory approach used for reducing the dimensionality of the feature space. This is achieved by transforming a number of possibly correlated variables into a smaller number of uncorrelated variables called principal components. Geometrically, PCA spans a new feature space with much lower dimensionality using a number of training samples as shown in figure 3.4. The dimensions of the new space are rotated dimensions of the ordinary space to where data have high variances. Consequently, the facial images are represented in a common space with less dimensionality and more discriminability.

In mathematical terms, this method finds a set of eigenvectors of the covariance matrix of the training images. The eigenvectors’ characters vary according to their corresponding eigenvalues, so that the maximum eigenvectors are those correspond to the maximum eigenvalues of the covariance matrix. Therefore, the associated eigenvalues allow us to rank the eigenvectors according to their usefulness in characterizing the variance among the images. Projecting the face image onto those eigenvectors results in ghost-like images, and we call these images eigenfaces. Figure 3.5 shows the top eleven eigenfaces extracted from an input image. That means, that any face image can be approximated by a linear combination of all of its eigenfaces, and that every eigenface contributes to the reconstruction of the original image. When the eigenvalues of the covariance matrix are ordered from top to bottom, they drop drastically in an exponential way. This observation represents the basic idea of using the eigenfaces for dimensionality reduction. That is, the original face image can be approximated by combining the a few eigenfaces that correspond to the top eigenvalues. Those eigenfaces are used to represent the facial image in the recognition system. Therefore, PCA approach can be summarized in the following steps:

- Acquiring a training set of the candidate facial images.

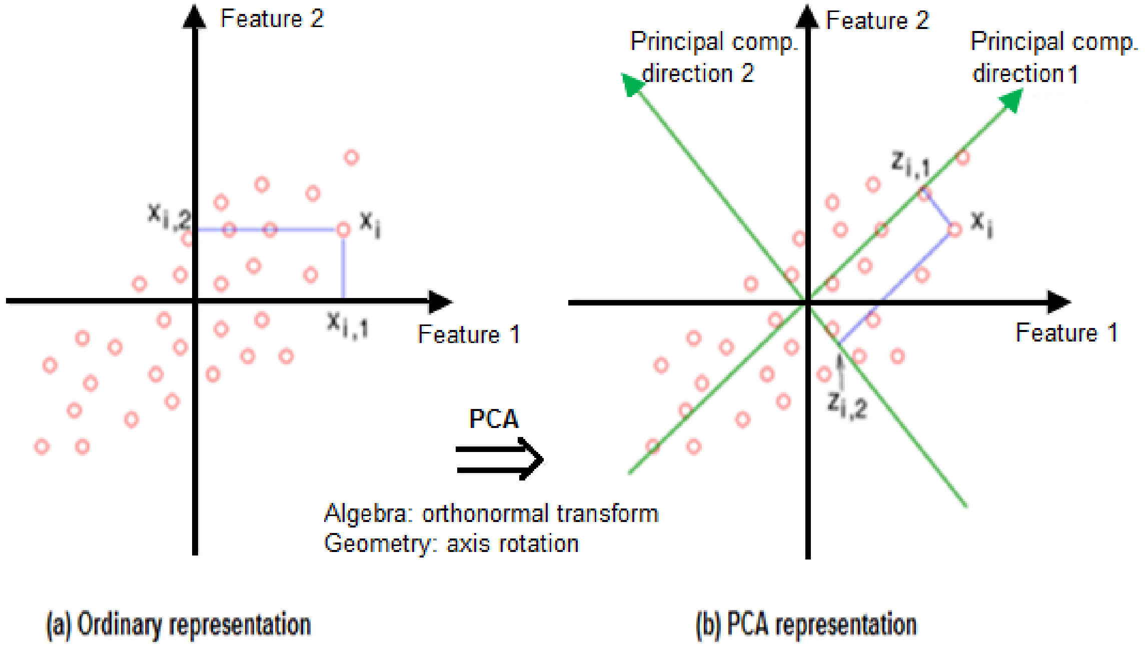
- Eigenfaces of the training set are calculated. Only M eigenfaces that corresponding to the highest eigenvalues are chosen. Those eigenfaces are named principal components, which represent the coordinates of the new face space. Eigenfaces can be recomputed as the face library is updated by new samples.
- Calculating the new coordinates of each training image in the new M-dimensional space by projecting that image onto the selected eigenfaces.

This method can be used for classifying new images as follows: after forming the new face space and projecting the training samples onto that space, a new face image can be classified as following:

- Projecting the query face image onto the same eigenface space that calculated in the training stage, so we can obtain the weights (coordinates) of that image in the new space.
- Checking if the input image is a facial image at all if it locates close enough to the images (points) in the space.
- If the system verifies it is a face, a similarity measure can be used to classify the facial patterns whether it belongs to the face library or, in further applications, decide whom in the database this image belongs to.

3.3.1.1. Calculating the Eigenfaces

A face image $I(x,y)$ of $\mathrm{n} \times \mathrm{m}$, where n and m are the numbers of pixels in a column and a row respectively. This image can be considered as a vector in an $\mathrm{n} \times \mathrm{m}$ dimensional space $\Re^{n \times m}$. Our aim is to represent this image in a space with much lower dimensionality. For a given face dataset Γ of N images, which is a set of cropped and randomly selected images, first, we need to calculate the covariance matrix of the training images. In order to do that, the average matrix Ψ of those images need to be calculated, and then to be subtracted from the original faces Γ_i .In biometric and imaging applications, images are formed by row concatenation of the image data so that the length of the image vector becomes the product of the width and the height of an image.



(a) Ordinary representation

(b) PCA representation

Figure 3.4: Transforming data from the ordinary space into the principal component space.

$$\Psi = \frac{1}{N}\sum_{i=1}^{N} \Gamma_i \tag{3.1}$$

$$\Phi_i = \Gamma_i - \Psi \tag{3.2}$$

Now, covariance matrix C of the training images Γ_i can be calculated as:

$$C = \frac{1}{N}\sum_{i=1}^{N} \Phi_i {\Phi_i}^T \tag{3.3}$$

The covariance matrix C has an $n \times m$ eigenvalues and eigenfaces. For example, if the input image is 92× 112, then the number of eigenvalues and eigenfaces are 10,304 each. The process of dimensionality reduction starts here when only selected eigenfaces are chosen to represent the image. However, the problem of computation is not solved yet, since calculating the eigenvalues and eigenvectors for a matrix of 10,304×10,304 is still an exhaustive process. Turk and Pentland [54] resolved this problem, stating that in linear algebra, it is known that an $M\times N$ matrix, where M>N, can only have N-1 non-zero eigenvalues. Thus, covariance matrix can be represented as:

$$C = \frac{1}{N}\sum_{i=1}^{N} {\Phi_i}^T \Phi_i \tag{3.4}$$

Where C has a size of $N\times N$ instead of $(n \times m)^2$, where $N << n \times m$. That means that the amount of calculations to be performed is reduced from the number of the pixels in the input image $(n \times m)^2$ to the number of images in the training set *N*. Moreover, only a subset of M eigenfaces that correspond the largest eigenvalues is used. The higher the eigenvalue, the more discriminative the corresponding eigenvector is. On the other hand, eigenfaces with low eigenvalues explain only a very small part of the face features; therefore, they can be omitted. Figure 3.5 shows the mean image at the top left corner followed by the top eleven eigenfaces that have the highest eigenvalues.

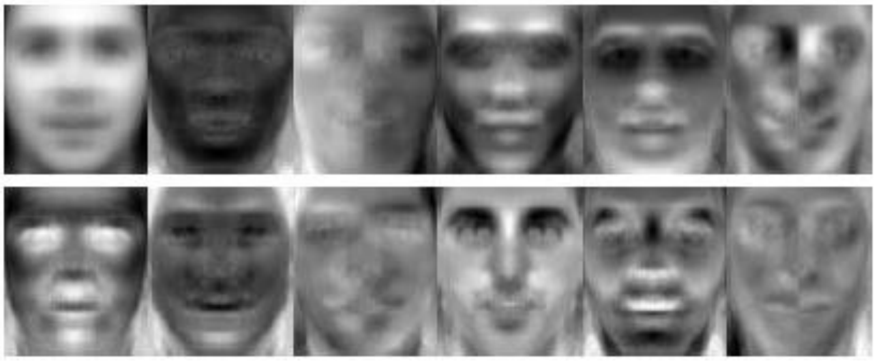


Figure 3.5: Samples of eigenface images: The mean image of facial images at the top left corner, followed by the top eleven eigenfaces.

Choosing the number of the principal components (eigenfaces) to represent the facial image is very important factor in face recognition. The number of eigerifaces to be used is chosen heuristically based on the recognition rate we obtain. In order to do that on the ORL face database, which will be described in chapter 5, we conduct an experiment as a part of this research using ORL database. The results are depicted in figure 3.6, which is similar to the results obtained in the literatures using other databases. From the graph we notice that the number of eigenfaces to be used is around 100. The recognition rate incre-

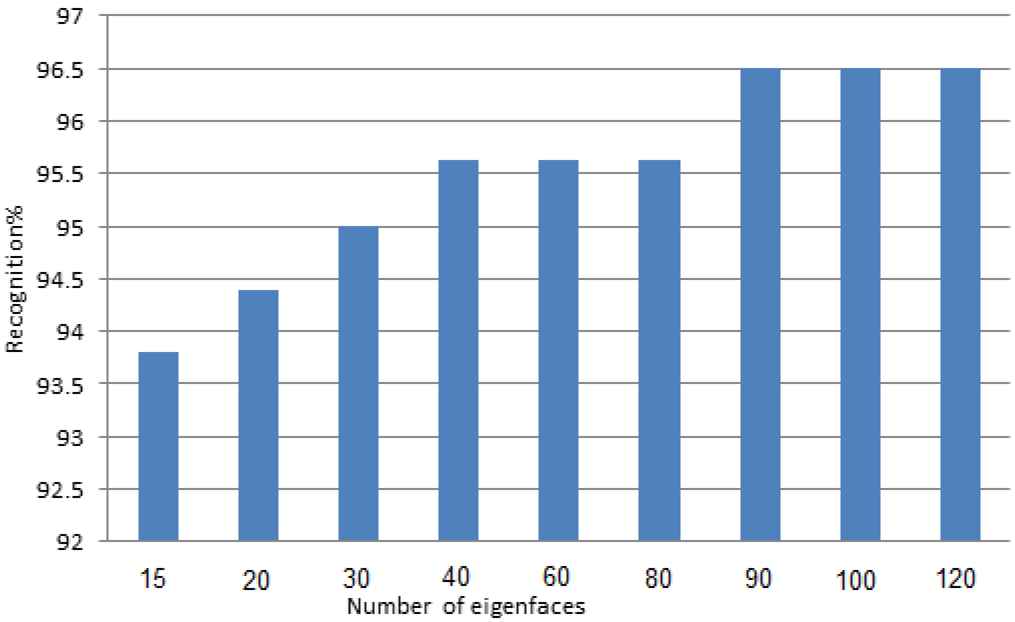


Figure 3.6: Recognition rate Vs. the number of principal components on ORL database.

ases as the number of eigenfaces increases until it reach 90, where the performance of the system remains constant. It is important to notice that the recognition rate is not significantly sensitive to the number of eigenfaces since the recognition rate increases from 93% to 96.5% when the eigenfaces are increased from 15 to 90.

3.3.1.2. Limitations and Difficulties

Although PCA has been adopted in many face recognition schemes over the last two decades, it suffers from some considerable limitations and difficulties [55]. One of the major difficulties is that PCA demands an accurate registration for the facial images before eigenfaces are calculated. Video frames of sub-pixel level of movement can guarantee such an accurate registration. Another issue is that eigenfaces should be explicit and relevant. This condition demands using a large number of faces which are cropped and aligned in the same way so that principal components of different faces line up with each other. Practically, this is quite a challenging task since the contours and yaws of human face cannot be controlled, and some deviations must exist. Furthermore, the eigenfaces effectiveness might be limited for some desired use. Eigenface method is mainly designed to identify subjects rather than human faces. It demands face images to be frontal and non-rotated. Although eigenfaces can be developed to be more generalized by using both frontal and tilted faces for training, this might affect the registration accuracy significantly. Over this work, however, all face images are in frontal view so the only challenge we are tackling is the lack of the image details when it comes in very low resolution. Another limitation of this algorithm is the ability of generalization across environments with different lighting. The basic PCA, as a holistic method, deals immediately and only with the pixels values, which make it sensitive to any change in lighting. The amount and the angle of the lighting can over brighten some parts of the face, such as the cheekbones and the overhead, or it can create some local shadow on the eyes or beside the nose. Consequently, the pixel values vary drastically. This means that to obtain better results from eigenface algorithm, face images should be taken in a controlled environment in terms of lighting. Another problem the eigenface approach suffers from is its lack of discriminatory power. Moghaddam et al. [56] have demonstrated the impact of each single eigenface on the output of eigenface algorithm. When they plotted the projection of the face image onto the largest three eigen coefficients of each class, they found that the projections overlap each other. Such a result stands to reason since eigenfaces are calculated based of the variation of the face data, rather than eliminating the redundant information. This means that PCA does not work robustly when it stands alone for feature extraction and it should be configured with another feature extraction method. PCA can enhance the performance of another feature extraction method so it chooses the only output features of that method which have the highest variation. In this thesis, PCA will be configured with wavelet transform to extract more discriminative features.

3.3.2. Discrete wavelet Transform

The wavelet transform is a powerful mathematical tool used to analyze the image as a non-stationary signal. It provides a multi-resolution analysis of the image in the form of coefficient matrices with spatial and frequency decompositions at the same time. The use of multi-resolution decomposition has been pointed out in many psychovisual articles where evidences were offered that the human visual system processes the images in a multiscale process [44]. Lately, wavelet analysis has been used in many pattern recognition researches and applications [42-49]. One of the important advantages of wavelet transform is its flexibility in use since there are many bases existed, and the more suitable one can be chosen for a given application. However, this option raises a question about which wavelet form should be chosen for a specific application or task, since only empirical remarks demonstrate the choice of the wavelet form. Another characteristic of wavelet is that the computational complexity of this method is linear with the number of the output coefficients (O(N)) which is less cost than other transformations. Therefore, wavelet is recommended also from a hardware design prospective. Thus, in the case of real time face recognition application, where the computational complexity is highly considered, it is possible to embed a part of the process in hardware [52], [51].

The continuous wavelet transform of 1-D signal $f\,(x)$ is defined as:

$$(W_a f) = \int f(x)\mathcal{T}_{a,b}(x)dx \tag{3.5}$$

Where,

$$\mathcal{T}_{a,b}(x) = \frac{1}{\sqrt{a}}\,\mathcal{T}\left(\frac{x-b}{a}\right) \tag{3.6}$$

$\mathcal{T}$ is the mother wavelet function which has to be localized zero-mean function. For digital signals, equation 3.5 can be discretized by constraining a and b to be a discrete($a = 2^n, b \in Z$). In this case, the transform is called Discrete Wavelet Transform (DWT). DWT can also be extended to a 2D transformation to be applicable for images by applying separate filter banks in different orientations. This process leads to decomposing the image into sub-images at different levels repeatedly on low frequency output. Figure 3.7 shows the block diagram of 2-level decomposition of DWT. This can be done by passing the image through a series of filter bank stages. The filtered outputs are then down sampled by a factor of two in the horizontal and vertical orientations. The typical wavelet transformer uses low-band pass filters and high-band pass filters. At one level of decomposition, the input image is decomposed into four subbands LL, LH, HL, and HH subbands. The output of the LL subband is a half size of the original image and it contains the low frequency components of the image. HL, LH and HH contain high frequency components of the image that correspond to vertical, horizontal and diagonal subbands respectively. The LL subband contains the low frequency components of the

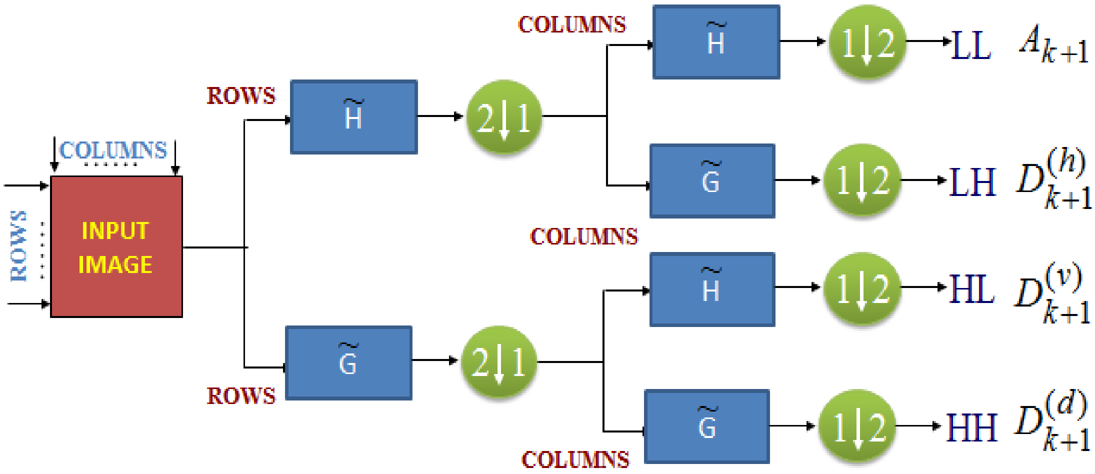


Figure 3.7: Two dimensional discrete wavelet transform (DWT) for an image: The low pass filter H(.) and the high pass filter G(.) are applied along the rows initially followed by downsampling by a factor of two. This is followed by filtering using H(.) and G(.)

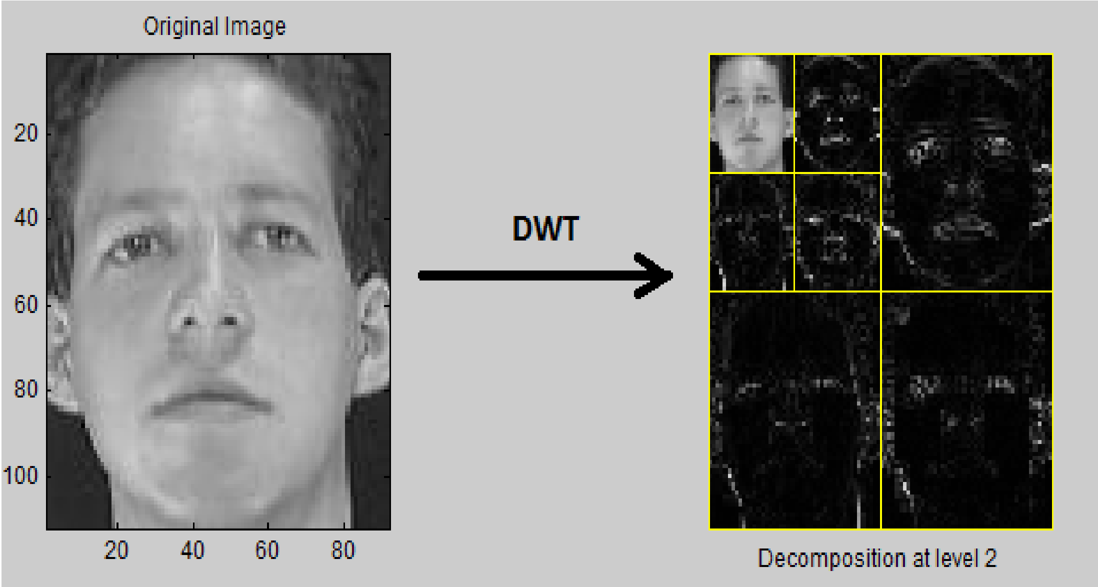


Figure 3.8: DWT decomposition of an image at level 2. Details images hold some features in three directions: horizontal, in the left bottom image, diagonal in the right bottom image, and vertical at the top right image. The approximation image LL is at the top left corner.

original image. Figure 3.8 shows an example of 2-level discrete wavelet decomposition of an image from ORL database of size 92×112. The approximate image LL, shown in the greyscale in quarter the size of the original image, is the output of two low-pass filters passed over the original image in horizontal and vertical directions. This output image has the most wanted component in face recognition applications. An *n*-level wavelet decomposition is calculated as [44]:

$$\begin{aligned} A_n &= [H_x*[H_y * A_{n-1}]_{\downarrow 2,1}]_{\downarrow 1,2} \\ D_{n1} &= [H_x*[G_y * A_{n-1}]_{\downarrow 2,1}]_{\downarrow 1,2} \\ D_{n2} &= [G_x*[H_y * A_{n-1}]_{\downarrow 2,1}]_{\downarrow 1,2} \\ D_{n3} &= [G_x*[G_y * A_{n-1}]_{\downarrow 2,1}]_{\downarrow 1,2} \end{aligned} \qquad (3.7)$$

where * denotes the convolution operator, ↓2, 1(↓1, 2) is subsampling along the rows (columns) and $A_0 = I(x, y)$ is the original image. *An* is the approximation image of A_0 after n decompositions, and is obtained by low pass filtering in column and row directions. The details images *Dni* are obtained by bandpass filtering in a specific direction (*i*=1, 2, 3 for vertical, horizontal and diagonal directions, respectively) and they contain directional details information at scale *n*.

Indeed, there are some functions that can be used as a mother wavelet, such as, Haar, Daubechies, Coiflet, Symlet, Biorthogonal and Reverse Biorthogonal wavelet function. In this work, those functions provide similar results, and no significant change in performance is noticed whena wavelet function is replaced. We choose Haar function for all of the experiments due to its reliability.

In the next section, we configure a system by which, more robust features are extracted from the wavelet images. Eigenfaces are calculated for the approximate wavelet images in order to extract the features that have the highest variance. Therefore, since we are dealing with images in low resolution, the number of wavelet decomposition levels should be taken into consideration, so that the output image should hold sufficient contents to apply PCA.

3.3.3. PCA on DWT for face recognition

PCA approach for face recognition encounters two main issues as mentioned in section 3.2.2. The first one is the computation finding the eigenfaces. PCA has a computational complexity of O(N^2), where N is the number of the training images. Another limitation of PCA is that the only criterion it uses to select features is the variance of the feature. This criterion is not enough to because the pixel values might be changed due to many factors, such as, the lighting or the image noise. On the other hand, the output of the wavelet transform is an approximate image, which contains the low frequency components of the original image. The whole approximate image cannot be fed to the classifier because of two reasons. First, using the whole approximate image can increase the computational cost and time consumption of the system. Second, the components of the approximate image can be redundant although they are more beneficial for classification than those in the original image. Therefore, more discriminative features need to be extracted from the wavelet image. It has been proven that extracting eigen features from the wavelet images can provide better recognition performance and can decrease the complexity of the system [48].

In this section, wavelet transform based PCA is used in order to cope the limitations of the PCA and wavelet transform. The advantage of this configuration is that a smaller image, which is the wavelet image, is applied to the PCA. This can decrease the complexity of the PCA, and make the selected eigenfaces more robust. As well, PCA can eliminate the information redundancy that wavelet images suffers from, and to make the classification process faster than using the whole wavelet image. Figure 3.9shows the block diagram of the PCA based wavelet method for face recognition. From the figure, this method consists basically of two phases: the training and the testing or recognition phase. In the training stage, images from different classes are represented and they are named reference images. Those representations are then stored in the feature library.

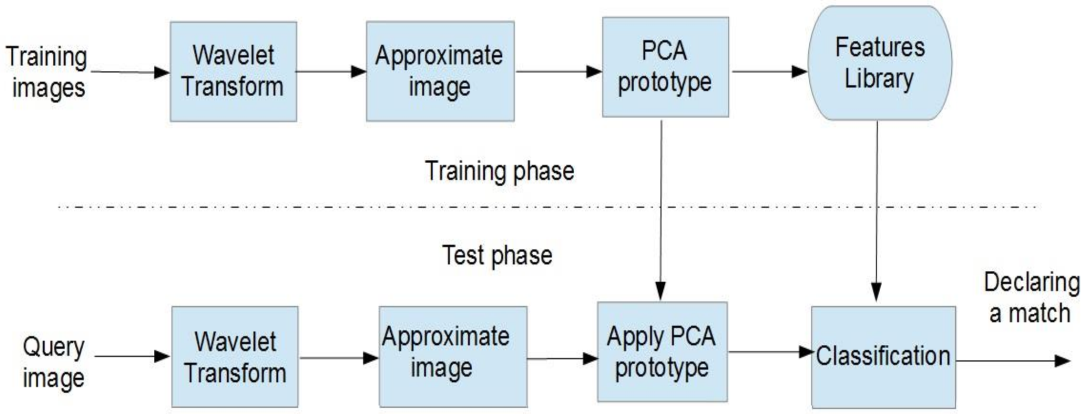


Figure 3.9: Block diagram of wavelet based PCA recognition system.

Therefore, any given inquiry image is represented according to the representation prototype that is calculated in the training phase, and then to be matched with the image representations stored in the library using a certain classifier or a similarity measurement. In more details, training stage can be accomplished in three steps: First, an input image is passed through wavelet transformer where it is decomposed into smaller subband images: LH, HL, HH, LL as in equation 3.7. At 1-level wavelet decomposition, the output sub-image is in one quarter size of the original image. For example, if the size of the input image is 128×128, then the subband images are of size 64×64. For 2-leveldecomposition, the approximation image of the first decomposition is passed through the same filters to provide an output subband image of size 32×32, and so on. The number of decomposition levels is chosen depending on the size of the original image and the size of the output subband image so it holds enough number of low frequency components that are sufficient for classification. As the input image size gets lower, less number of decompositions is needed. The Second step is that after applying wavelet transform, the output approximation image applied to PCA, and sufficient number of principal components is extracted. Then, the output feature vector is stored in the feature library to form the face space. For any query image in test phase, the same number of wavelet decompositions is accomplished, and the output approximation image is then projected onto the PCA prototype. At this point, both the gallery face images and the test image are represented in a common space using a few but robust features. Lastly, images are matched using a certain classifier or a similarity measure.

3.3.4. Local Binary Pattern

Local Binary Pattern (LBP) is one of the best texture image descriptors that has been used in many applications [50], [58]. Recently, it has been adopted as a descriptor of face images in many face recognition schemes [41,59,60]. The idea of using such an approach for face description is motivated by the fact that the human face can be seen as a composition of micro-patterns which are well described using LBP. Our hypothesis is that the low resolution facial images and the interpolated images, where the local features are no longer clear, can be seen as texture images. Therefore, LBP as a texture descriptor can be used for representing those images for classification.

For a given facial image, the LBP operator assigns a label for every pixel (x_c, y_c) of a greyscale valueg_c. This is done by thresholding that pixel with its 3×3-neighborhood, which are defined by the texture $T = t(g_0, g_1, g_2, \dots, g_7)$, where g_i is their greyscale values. The result of thresholding is considered as a binary number. The new T can be written as a binary set [59]:

$$T \approx t(s(g_0 - g_c), \dots, s(g_7 - g_0)) \tag{3.8}$$

Where $s(x)$ is given by:

$$s(x) = \begin{cases} 1 & if \quad x > 0, \\ 0 & if \quad x \leq 0. \end{cases} \tag{3.9}$$

After that, one can calculate the LBP at the given pixel location by sorting the binary result of comparison in this way:

$$LBP(x_c, y_c) = \sum_{i=1}^{7} s(g_i - g_c) 2^i \tag{3.10}$$

Figure 3.10 depicts the process of calculating the LBP for a pixel. After LBP operator is applied for all image pixels, a new image with sharper details is provided as in figure 3.11.This method exploits the histogram of the LBP map to represent the image for recognition. The *i*-th bin of the histogram is defined as:

$$h_i = \sum_{x,y} I\{fl(x, y) = i\}, i = 0, \dots, n - 1 \tag{3.11}$$

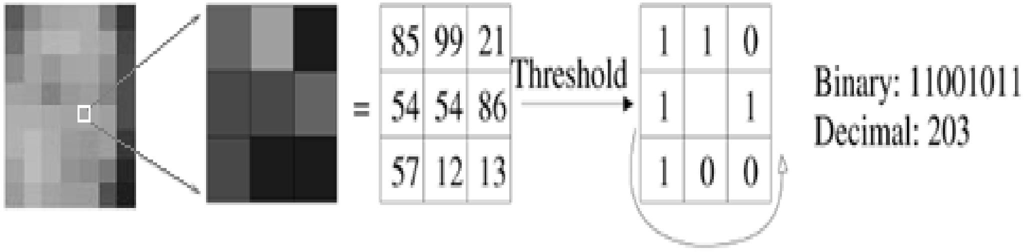


Figure 3.10: The basic LBP operator.

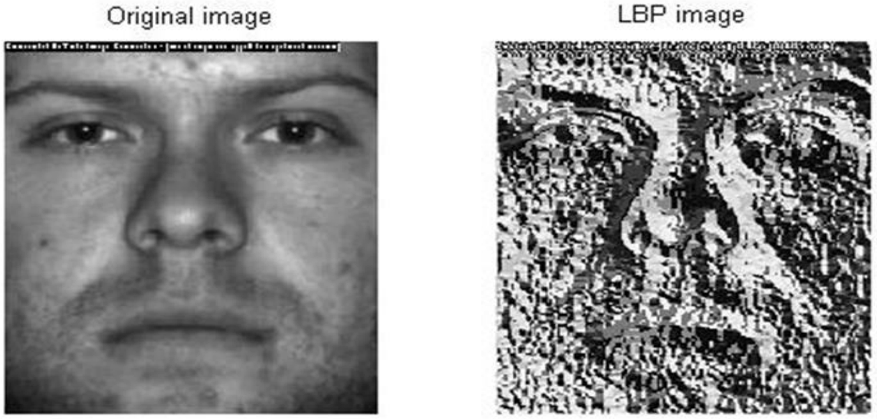


Figure 3.11: An Example: the greyscale facial image and the corresponding LBP image.

Where n is the number of the histogram bins, which is defined according to the definition of the greyscale depth of the image (in this work n = 256 = 2^8). The function I is given by:

$$I(A) = \begin{cases} 1, & A = true \\ 0 & A = false. \end{cases} \tag{3.12}$$

To form the image feature vector, the histogram of the LBP is facilitated in such a way that retains information about spatial relations. In order to do that, LBP image is divided into non-overlapping regions, and the histogram is computed for each region $H = (h_0, h_1, \dots, h_{n-1})$. To describe the whole facial image, the LBP map is presented by concatenate the histogram of each region, so that $H = (H_1, H_2, \dots, H_m)$, where m is the number of regions. Figure 3.12 illustrates the LBP algorithm for face representation.

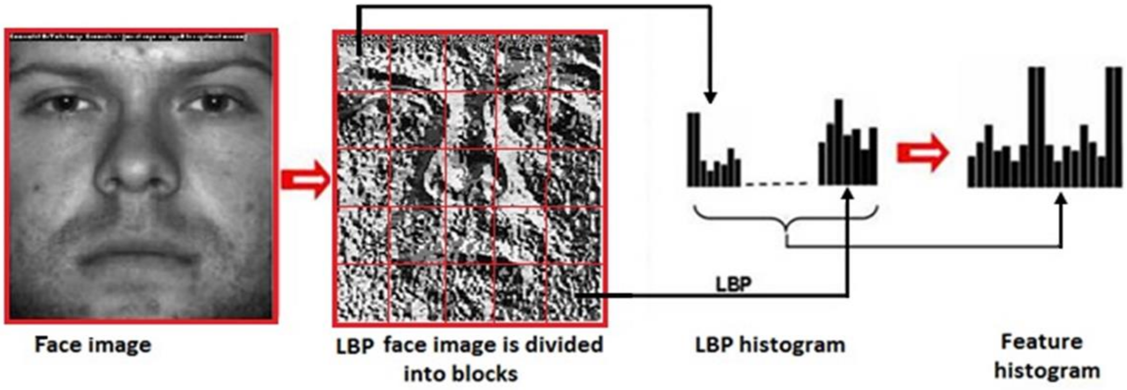


Figure 3.12: Extracting a histogram feature vector in LBP.

3.3.4.1. Parameters of the LBP method

The basic LBP has some limitations representing the image robustly for recognition. Therefore, some extensions and adjustments are needed. The first issue with the basic LBP operator is that the small number of neighbourhoods is not sufficient to create a dominant with large scale structure [60]. To overcome this limitation, more neighbors can be involved within the 3×3 region using circular, and bilinear interpolation is used to estimate the values of those neighbors on the circle. We can change the number of neighbors and the length of the radius between the central pixel and the circle for better representation as shown in figure 3.13. Also, using uniform patterns can reduce the com-

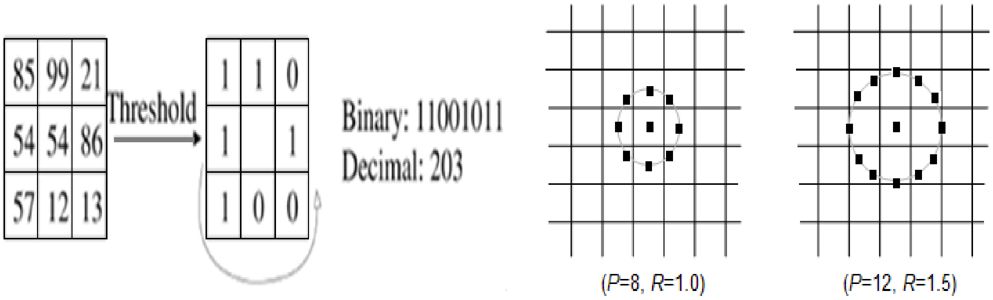


Figure 3.13: LBP operator. On the left: the basic operator uses only the primary 7 neighbors. On the right: The circular (8,1) and (12,1.5) neighborhoods.

putation of the LBP. Uniform patterns are the ones that have at most two bitwise transitions from 0 to 1 or vice versa. For example, 00001000, 001100000 and 11000001 are uniform patterns. Using these patterns can present around 70% of the whole patterns if (16,2) pattern is used, and 90% of the all LBP patterns if (8,1) pattern is used [58].

Therefore, the LBP operator can be written in the subscript $LBP_{P,R}^{u2}$, where u_2 refers to uniform patterns, and P,R, represent the number of neighbors and the length of the radius respectively. In this work, we adapt the patterns $LBP_{8,1}^{u2}$ as it better represents the face images according to the previous works.

Remark: After calculating the LBP image, this image is divided into a number of regions as mentioned above. We choose a number of regions so that useful image contents vary gracefully within this region, and this number can be estimated heuristically. In our experiments, as the image size gets smaller, the number of regions is reduced to meet the above criterion.

3.3.4.2 Template Matching

Face recognition based on LBP does not need optimization in the training stage. After calculating the histogram feature vector for all candidate images, a similarity measure is applied to declare the closest image as a match. Chi square distance is adopted as a dissimilarity measure for histogram. Chi distance X^2 is given by:

$$X^2(S,M) = \sum_{i,j} \frac{\left(S_{i,j} - M_{i,j}\right)^2}{S_{i,j} + M_{i,j}} \tag{3.13}$$

Where S and M are two LBP histograms; $S_{i,j}$ is the histogram of region *j* in image *i*.

3.3.5. Block-based Discrete Cosine Transform

DCT transform is used for feature extraction in image processing applications. It has been used either as holistic appearance-based or local appearance-based feature extraction scheme in many face recognition [61-65]. DCT expresses data as a sum of cosine functions for reduced size of data. The 2D-DCT of an image can be given by [51]:

$$F(u,v) = \alpha(u)\alpha(v) \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} cos\left[\frac{\pi u}{2N}(2x+1)\right] cos\left[\frac{\pi u}{2M}(2y+1)\right] f(x,y) \quad (3.14)$$

$$\alpha(u)\alpha(v) = \begin{cases} \sqrt{\frac{1}{N}} & for\, u,v \neq 0 \\ \sqrt{\frac{2}{N}} & for\, u,v = 0 \end{cases} \quad (3.15)$$

Where f(x, y) is the intensity of the pixel at the location (x,y), *u* varies from 0 to M-1, and *v* varies from 0 to N-1, where M×N is the size of the image. The DCT map of any given image is similar to one in figure 3.14.This transformation can be utilized to extract important features by retaining information from different frequency bands. Whereas, this local appearance based face description is a generic local method, and does not demand detection of any local landmarks, such as, eyes, nose, mouth, or any distinguishing features as in other common local feature-based approaches for face representation [66]. DCT map, as shown in figure 3.15, contains information of three frequency bands, namely, low frequency, middle frequency and high frequency. Low frequency coefficients, which concentrated at the top right corner of the DCT map, are related to illumination variation and smooth regions. The middle frequency coefficients come secondly at the middle of the map to represent the basic structure of the image. The last region contains the high frequency coefficients which represent noise and detailed information, such as, edges. Each band of the image contains important information and can be utilized according to the application and the process.

In this thesis, we choose face database that does not suffers from illumination variation, and the only challenge we create is the resolution degradation; therefore, a small number of the low frequency components of the DCT, including the DC component, are used to represent the facial image for classification. In order to that, and because of the variety of the local facial features over the face texture, we adopt recent works [51], [66] that tends to use block-based DCT (BBDCT).

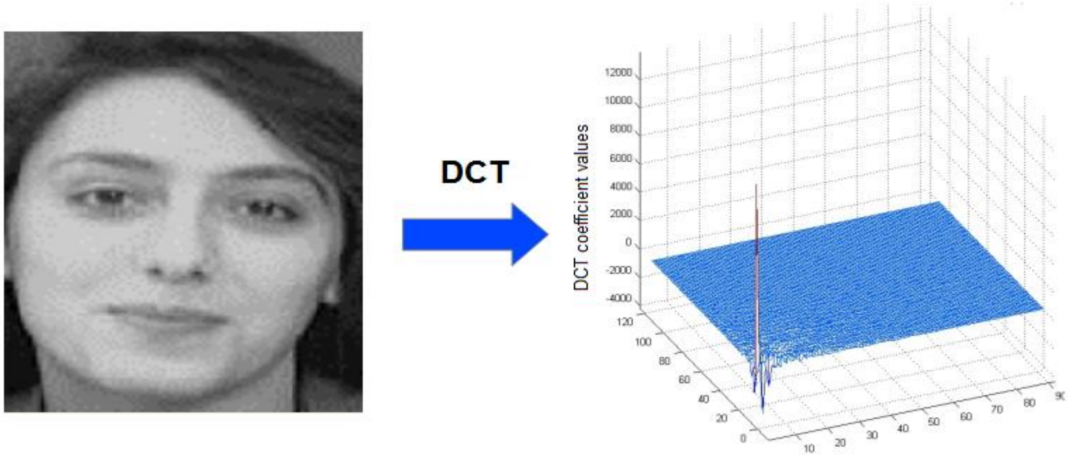
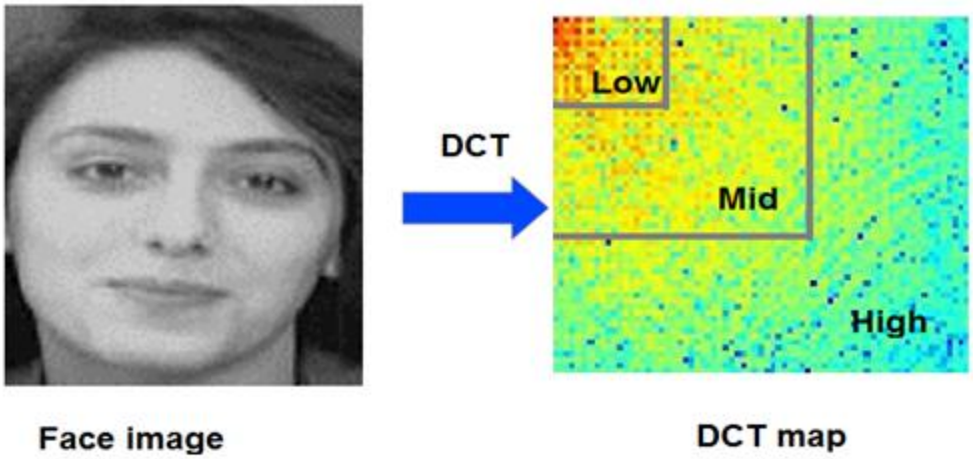


Figure 3.14: DCT transform of a facial image.



Face image

DCT map

Figure 3.15: Face and its DCT transformation (Low, Mid. and High) regions.

This approach divides the image into blocks of uniform size, and isolates the most relevant features of each block. In [66], face images are divided into a certain number blocks, and each block is then represented by its diagonal DCT coefficients. Selecting the diagonal coefficients from Block based DCT not only retains different band information, but also it reduces the computation of the algorithm.

In this work, however, we adopt the algorithm proposed in [51] as it provides better results. However, this algorithm cannot handle images with very low resolutions, such as, 8×7, since the block in this case does not contain enough components to apply DCT.

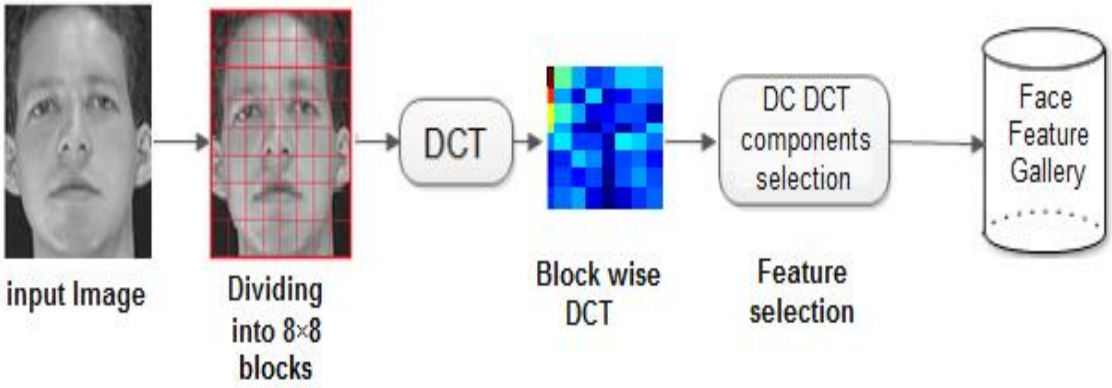
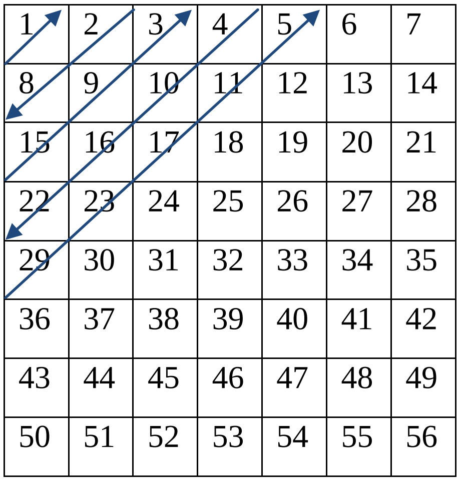


Figure 3.16: The block diagram of BB DCT feature extraction method.

Therefore, some modifications are needed. The block diagram of the Block-Based DCT for feature extraction is showed in figure 3.16. From the figure, the facial image is, first, divided into 8×8 blocks. If the dimensions of the image are not integral multiples of 8, the extra rows or columns are replaced by zeroes. For the face database that is used in this work, the experiments that are conducted in chapter five proved that if the image size is not very small, 8×8 blocks provides the best recognition performance. This conclusion is also given in [51], [66]. 8×8 blocks can provide an adequate block size that holds enough information, and image details can gracefully vary from block to another within this division [51]. As the size of the image decrease, the number of blocks needs to be reduced properly so that each block can hold an adequate part of the facial information. The top left element of a DCT map, as shown in figure 3.15, is the DC component, which represents the average intensity of the image. The rest of the DCT map contains AC coefficients that correspond to the high frequency components of the image. It is proven that the high frequency components are not required in face recognition and eliminating those components also causes the image to be robust to the scale variations which is required in FR systems. Also, it is observed that the DC coefficient carries 95% of the total energy that the image holds. Experimental proof of these conclusions is shown in [51].

Remark: Since we are seeking to extract the low frequency components, which play a key role in FR, and because the images in our experiments have relatively low resolutions, some adjustments are needed for better recognition performance as follows: the number of blocks varies from 8×8 down to one block for the whole image when the image has very low resolutions, so we assure that each block contains enough contents. This is done as following:

- When the image has a resolution around 19×17 or higher, it is divided into 8×8 blocks regardless of the size of each block.
- When the image size goes down in range between 12×10 up to 15×14 approximately, facial images are divided into 6×6 blocks.
- When image size is less than 12×10, only one block is used and DCT is applied for the whole image. In this case, *Raster scan* is used to order the wanted DCT components. It starting from the upper-left corner, and subsequently we form a 1D feature vector. In an image with pixel values in the range of [0,255] (grayscale image), the DCT values range from minimum -255×8×8 to maximum +255×8×8, and they change from low to high variation in a zig-zag pattern (raster scan order). This pattern is the basis for selecting the number of features in the selected blocks. Thus, low frequency components in the triangle at the top left corner of DCT map are chosen. It has been proven in [67], [68] that the best number of DCT coefficients can be extracted from the triangle for ORL database is from 20-25 components; however, this is the case when the image are in the original size 112×92. In our case, the images are 8×7; therefore, a smaller number of components are needed.



1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31	32	33	34	35
36	37	38	39	40	41	42
43	44	45	46	47	48	49
50	51	52	53	54	55	56

Figure 3.17: Raster scan (Zig-zag) pattern for choosing DCT coefficients of an image of 8×7.

From our experiments, the triangle, with equal sides of 5 pixels, is chosen as it provides better results. The 15 coefficients inside the triangle are organized in *Zigzag* order as shown in the figure 3.17.

3.4. Chapter Summary

This chapter presents the feature extraction methods that are used in our face recognition system. Because of the small images we use in our experiments, Holistic (global) feature methods are preferred over the local feature methods. Local feature methods use the dimensions and the measurements of the local features, such as, eyes, nose, mouth, and cheek bones for representing the facial image for classification. Holistic feature methods, however, extracts some features from the image using underlying statistical regularities. In other words, this type of methods takes into consideration every single pixel in the image regardless of its location on the facial image. Some holistic methods are adopted in this work: the principal Component Analysis (PCA) or eigenface method, Discrete Wavelet Transform (DWT), Local Binary Pattern (LBP), and Block-Based Discrete Cosine Transform (BBDCT). PCA extracts some facial features according to their variance within the training samples. The best number of these features is around 100. In this work we combine PCA with DWT so limited principal components are extracted from the wavelet images for better representation. LBP coverts the greyscale image into a new image using a binary code. The histogram of LBP image is used for image representation. In BBDCT, the image is divided into blocks and DCT is applied to each block, and the DC component is extracted from each block. If the image is small, fewer blocks are needed, so each block should have sufficient image contents to apply the DCT.

Chapter4

Classification

4.1. Introduction

The classification process takes place after the feature extraction stage where the facial images are represented in feature vectors. Classification refers to the process in which a given face image is assigned to a specific class. Figure 4.1 shows the classification stage in the typical face recognition system. The main objective of the classification stage is to measure the similarity between the feature vector (either geometric based or holistic based) of the probe image and other feature vectors of the database images using a certain technique called classifier. Therefore, in order to recognize a given probe image, a set of images for the candidate objects (database) need to be introduced to the system; this set is named training set or gallery set.

In machine learning, classification can be categorized into two types of tasks: supervised and unsupervised classification. In the first type, the class (label) of the input samples (gallery) is predefined and the decision is given to the class of the most similar sample to the probe image vector in the face space. Unsupervised classification, however, is more challenging where the class of the input samples is unknown. In this case, the classes of the input samples are assigned using a learning method called clustering. In the proposed face recognition system, we are achieving supervised classification since the classes of our images are known.

Face recognition systems can also be categorized, according to the purposes of the classification, to face identification and face authentication or verification. In face identi-

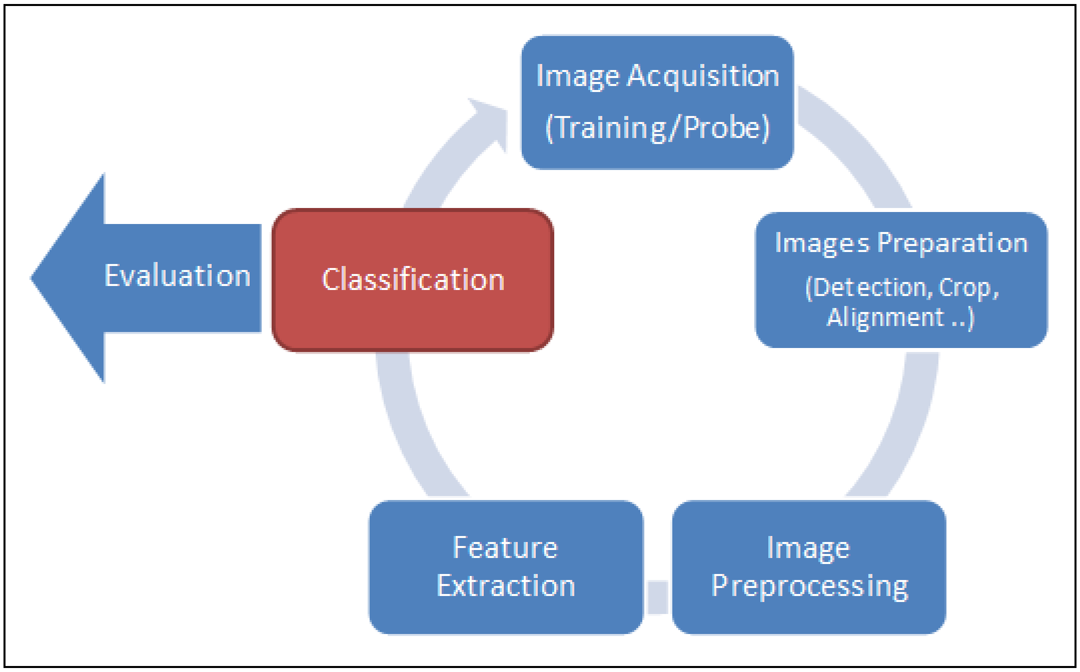


Figure 4.1: The typical face recognition block diagram showing the classification stage.

fication, the system performs a one-to-many matching that decides which class (person) among the classes (people) stored in the database the probe image belongs to. On the other hand, one-to-one matching is achieved in face authentication in order to determine if the input image (query image) matches a specific identity (person) in the database as claimed by the user of the system. Indeed, face authentication is used in many applications, such as security entrances where only specific people are allowed to enter. Face identification, however, is used for investigative purposes, including identifying a person of interest from a suspected criminal list. Another application can be seen in media and newspaper reporting where an image for a specific person needs to be retrieved from a large multimedia database [69]. This thesis seeks to identify a set of test images from our database.

There are many classifiers designed for pattern recognition and they are distinctive according to their computational cost and accuracy. Among many classifiers proposed in machine learning, *k*-NN method is widely used as an effective nonparametric approach of classification. It uses Euclidean distance to measure the similarity between the objects. This method is adopted for classification and for estimating the match for query images.

Neural Network (NN) is one of the most common machine learning algorithms that are designed for classification. Neural network, as a parametric algorithm, may be claimed to be notorious mainly because of its computational cost and non-guaranteed convergence.

In this work, however, we use an elegant fast version of neural network: Extreme Learning Machine (ELM) that will be introduced for learning Single-hidden Layer Neural Networks (SLNNs). Both k-NN and ELM methods will be used in our experiments in chapter six, and comparison results will be demonstrated. The following content introduces these classifiers.

4.2 *k*-Nearest Neighbors

The k-nearest neighbor (k-NN) algorithm is a simple but reliable machine learning algorithms for classification. For a given set of facial feature point distances of a probe image (unknown person) in the face space, nearest neighbor classification rule is used to identify the label (class) of this person. The rule of this method is that the class of the probe image is assigned by a majority vote of its k closest neighbors, where k is a small integer number. Euclidean distance is usually used for distance measurement. For two feature vectors, X_1, X_2 of N components, Euclidean distance can be measured directly as:

$$d = \sum_{i=1}^{N} (X_{1i} - X_{2i})(X_{1i} - X_{2i})^T \tag{4.1}$$

Where N is the length of the feature vector X. Theoretically, using a larger database usually allows us to use a larger value of k, which in turn, can make our classifier insensitive to noise (noise of the class distribution). Generally, when the system is tested on a small database, the smaller the value of k the better recognition rate can be obtained [70].

4.3. Single-hidden Layer Feedforward Neural Networks (SLFNs)

Artificial Neural Networks (ANN) is a popular machine learning approach that used in many pattern recognition applications. It determines a linear or non-linear functional relationship between dependent and independent variables by learning a mapping function from the training samples. ANN is a biologically inspired model consisting of interconnected units called neurons. The neurons are the basic units of the network. One neuron processes multiple inputs and produces an output which is a result of a function of a linear combination of the weighted inputs. This function is called the activation function and it is usually nonlinear. Figure 4.2 shows the basic structure of the neuron. When an input vector is applied to the network, it produces an actual output which should be different from the target output. In supervised learning, the aim of learning the network is to minimize this difference. This kind of ANN is called Feedforward neural network. Feedforward neural networks have been widely used in many fields of pattern recognition due to their ability to approximate complex nonlinear mappings between two high dimensional data, as well as its ability to provide a model to represent complicated natural or artificial phenomena of large classes.

Generally, the most difficult and challenging part using neural networks, is the training phase. Training process is needed in order to estimate the values of the network parameters that provide the best mapping between the input vector and the desired output. Back propagation (BP) is one of the most common learning algorithms for the ANN. However, it suffers from some limitations including a slow learning process [71]. In this chapter we present an efficient and fast algorithm called Extreme Learning Machine (ELM) proposed by Huang et al., (2004) for training Single-hidden Layer Feedforward Neural Networks (SFNNs). This algorithm will be utilized to estimate our neural network parameters fast and elegantly.

4.3.1. Single-hidden Layer Feedforward Neural Networks Model

SLFNs was presented as the simplest neural network architecture, and it has been preferred over multi-hidden layer neural network in terms of the computational cost and complexity. SLFNs basically consists of three layers as shown in figure 4.2. The input layer, which represents the entrance port of the network and has a number of nodes equal

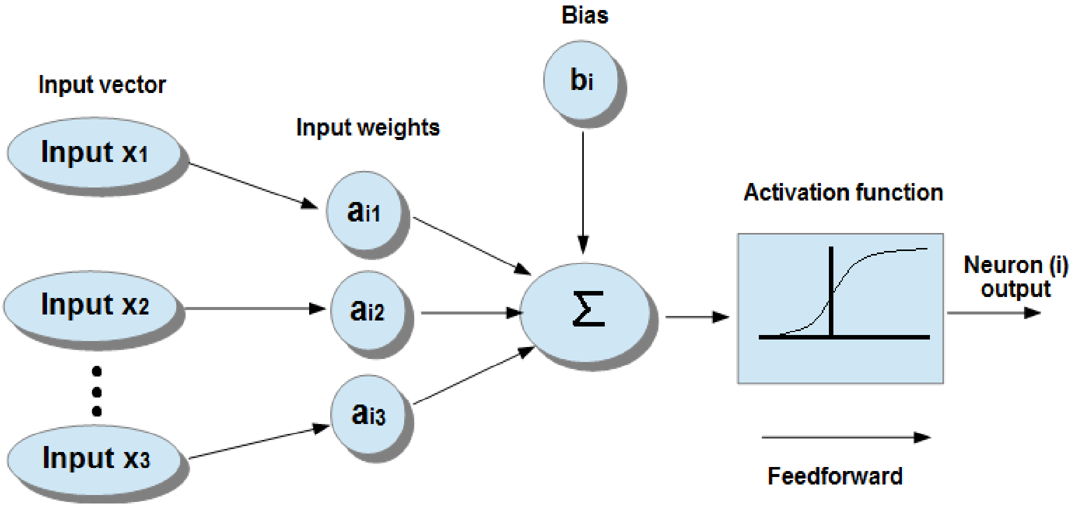


Figure 4.2: Model of an artificial neuron.

to the number of the input vector components. The input layer is connected by a set of weights (input weights) to the hidden layer which, in turn, has L number of hidden neurons. The last layer is the output layer which has a number of neurons (nodes) equals to the number of the classes. The output layer is connected to the hidden layer bya set of weights called output weights. Thus, the parameters of this structure can be summarized as following:

- Input vector x = $(x_1, x_2, \ldots, x_n)$
- Input weights = $(a_1, a_2, \ldots, a_n)^T$ for each hidden node.
- Hidden layer bias = $(b_1, b_2, \ldots, b_L)^T$
- Output weights = $(\beta_1, \beta_2 \ldots, \beta_L)^T$ for each output node.
- Output vector y = $(o_1, \ldots, o_N)$

The input vectors are nothing but the normalized output of the feature extractor that we adopted in chapter three. The output vector represents all candidate decisions (classes). In ELM approach, the only parameters we need to optimize in this architecture is the input weights a_i for each hidden neuron, the output weights $\beta_i$for each output node, and the bias (threshold) for each hidden layer,b_i.

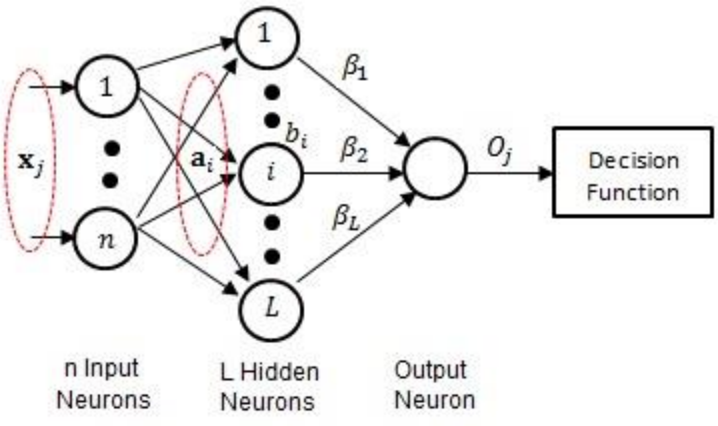


Figure 4.3: Single layer neural network:

with n-dimensional input vector x $= [x_1, \ldots, x_n]^T$ and output vectory $= [y_1, \ldots, y_N]^T$ corresponding to N classes. The input vector is augmented by the hidden layer neurons biases $b = [b_1, \ldots, b_L]^T$. The connections between the input vector and the hidden nodes are weighted by the input weights $a_i = [a_{i1}, a_{i2}, \ldots, a_{in}]^T$ linked to each hidden node. β_i is the output weights linking the hidden layer to the output layer.

4.3.2. ELM Based SLFNs Model

The structure in figure 4.3 represents the typical SLFN which can be learnt by different algorithms. In order to learn this network using ELM, this network needs to be in a specific mathematical model which is derived as follows. For N given independent samples (x_i, t_i), where $x_i = [x_{i1}, x_{i2}, \ldots, x_{in}]^T \epsilon\, R^n$ and $t_i = [t_{i1}, t_{i2}, \ldots, t_{im}]^T \epsilon\, R^m$, a standard SLFNs with L hidden neurons, as portrayed in figure 4.3, can mathematically be modeled as [70],[72]:

$$\sum_{i=1}^{L} \beta_i\, g_i(x_j) = \sum_{i=1}^{L} \beta_i\, g(w_i . x_j + b_i) = o_j\,, \quad j = 1, \ldots, N \tag{4.2}$$

Where $a_i = [a_{i1}, a_{i2}, \ldots, a_{in}]^T$ is the vector of the weights connecting the ith hidden node and the input nodes x_j, *bi* is the threshold or the bias of the ith hidden node. $a_i . x_j$ denotes the inner product of a_i and x_j. Lastly, $\beta_i = [\beta_{i1}, \beta_{i2} \ldots, \beta_{im}]^T$ represents the weight vector between the ith hidden node and the output nodes. This network, with L hidden neurons with an activation function $g(x)$, can approximate the N samples with zero error means so that $\sum_{j=1}^{L}\left\|o_j - t_j\right\| = 0$, which, in turn, means that for β_i, a_i and b_i :

$$\sum_{i=1}^{L} \beta_i \, g\left(a_i . x_j + b_i\right) = t_j, \qquad j = 1, \dots, N \tag{4.3}$$

The above N equations can be written in a compact form as:

$$H\beta = T \tag{4.4}$$

So that

$$\begin{bmatrix} G(a_1 . x_1 + b_1) & . \; . \; . & G(a_L . x_1 + b_L) \\ . & & . \\ . & \dots & . \\ G(a_1 . x_N + b_1) & \dots & G(a_L . x_N + b_L) \end{bmatrix}_{N \times L} \begin{bmatrix} \beta_1^T \\ . \\ . \\ . \\ \beta_L^T \end{bmatrix}_{L \times m} = \begin{bmatrix} t_1^T \\ . \\ . \\ . \\ t_N^T \end{bmatrix}_{N \times m} \tag{4.5}$$

In Huang et al. [71-77], H is named the *hidden layer output matrix* of the neural network. The *i*th column of H corresponds to the *i*th hidden node output with respect to inputs $x_1, x_2, \dots, x_N$. Indeed, the input weights (linking the input layer to the hidden layer) and hidden layer biases need to be adjusted in all conventional methods[71].This point is accounted as one of the major drawbacks of using Feedforward neural networks.

In order to have insight about the achievement role of the ELM, the conventional back-propagation (BP) for learning SLFNs needs to be reviewed first.

4.3.3.Conventional Gradient-based Solution of SLFNs

The gradient descent-based feedforward learning method minimizes the error function with respect to the network parameters by taking small steps in a downhill direction of the error function. The error function corresponds to the energy surface for the descent which is usually defined by the mean squared difference between desired and actual outputs of the network. In training the SLFN using gradient descent, we seek to find specific $\acute{a}_{\iota}, \acute{b}_{\iota}, \acute{\beta}(\iota = \acute{1}, \dots, L)$ such that:

$$\left\|H(a_1',\dots,a_{\tilde{N}}',b_1,\dots,b_L)\hat{\beta}-T\;\right\| = \min_{w_i,b_i,\beta}\left\|H(a_1',\dots,a_{\tilde{N}}',b_1,\dots,b_L)\hat{\beta}-T\;\right\| \qquad (4.6)$$

This is similar to the cost function minimization:

$$E=\sum_{j=1}^{N}\left(\sum_{i=1}^{L}\beta_i g(a_i.x_j+b_i)-t_j\right)^2 \qquad (4.7)$$

Thus, the aim is to search for the best H that minimizes$\|H\beta-T\|$. When gradient-based algorithm is used for minimization, vector W, the set of weights (a_i,β_i), and the biases (b_i) parameters can be estimated iteratively as:

$$W_k=W_{k-1}-\eta\frac{\partial E(W)}{\partial W} \qquad (4.8)$$

Where η is the learning rate. In back-propagation learning algorithm of feedforward neural networks, gradients are computed efficiently by back-propagation from the output to the input of the network. In particular, two advantages of back-propagation are its simplicity and its ability to be applied to nonlinear networks without a need for prior knowledge. However, this method still suffers from the following issues:

1- Selecting a proper value for the learning rate η is critical. If η is too small, the leaning process gets very slow because the steep descent often takes many steps in the same direction; when η is too large, however, the algorithm provide unstable performance where the estimated error value might fluctuate around the global minima or, in the worst case, it shows divergence to a global minima.
2- Local minima is another issue that impacts the performance of the BP learning algorithm when the learning process stops at a local minima instead of the global minima.
3- Back-propagation is a time consuming process in most of the real life tasks.
4- Over-training when using back-propagation causes the system to tend to overfitting, which, in turn, generates inferior generalization performance.

The above mentioned issues related to the gradient-based algorithms need to be resolved in an effective learning algorithm. In 2004, Huang [76] proposed a new approach that copes the previously mentioned issues when he rigorously proved that the SLFNs can be learnt without iterative methods.

4.3.4. Extreme Learning Machine (ELM)

Haung et al. uses some machine learning theorems to propose an elegant, but effective, method named Extreme Learning Machine (ELM). This method estimates the SLFNs parameters including the input weights, the hidden layer biases, and the output weights. Unlike the common understanding, he claims that those parameters may not necessarily need to be tuned. Furthermore, he rigorously proves that the input weights and hidden layer biases of SLFNs can be randomly assigned with only one condition: the activation functions of the hidden layer neurons are infinitely differentiable. After that, SLFNs can be considered as a linear system in which the output weights, which link the output nodes and the hidden neurons, can be determined by simply computing a generalized inverse of the hidden layer output matrix H. The following theorem is employed to innovate the ELM algorithm:

> ***Theorem:*** For a standard SLFN with N hidden neurons and activation function $g: R \rightarrow R$, which is infinitely differentiable for any interval, given N arbitrary distinct samples (x_i, t_i), where $\mathrm{x}_i \in R^{\mathrm{n}}$ and $t_i \in R^{\mathrm{m}}$, where n is the number of the input vector components and m is the number of the classes ,for any $a_i$and b_i randomly chosen from any intervals of R^{n} and R^{m} respectively; according to any continuous probability distribution, and with probability theorem one, the hidden layer output matrix H of the SLFN is invertible and $\|H\beta - T\| = 0$ [71].

The proof of this theorem is mentioned in details in [71],[77],[78].

4.3.4.1. The Procedures of the Extreme Learning Machine

After proving that the entries of the hidden layer output matrix H can be assigned randomly, ELM can be considered as a simple machine learning method. Training SLFN is to find a least square solution $\acute{\beta}$ of the linear system $H\beta = T$:

$$\left\|H(a_1 \ldots, a_L, b_1, \ldots, b_L)\acute{\beta} - T \right\| = \min_{\beta}\|H(a_1, \ldots, a_L, b_1, \ldots, b_L)\beta - T\| \qquad (4.9)$$

If the number L of hidden nodes is equal to the number of training samples, matrix H is square and invertible even if the input weights and the hidden biases are randomly chosen. However, in most cases, the number of hidden nodes is much less than the number of distinct training samples yields H is a nonsquare matrix. The inverse of this matrix can be computed using *Moore-Penrose generalized pseudo inverse* (Rao &Mitra, 1971). Therefore, for a training set $\aleph = \{(X_i t_i), X_i \in R^n, t_i \in R^m, i = 1, \ldots, N\}$, number of hidden neurons L, and infinitely differentiable activation function $g(x)$, ELM algorithm can be summarized in three main steps:

Step 1: Assign input weights a_i and hidden nodes biases $b_i$randomly for $i = 1, \ldots, L$.

Step 2: Calculate the hidden layer output matrix H.

Step 3: Calculate the output weight vector entries β simply by:

$$\beta = H^{\dagger} T \tag{4.10}$$

Where $T = [t_1, \ldots, t_N]^T$ is the ideal (supervised) output vector, $H^{\dagger}$ is Moore-Penrose pseudo inverse of H and can be given as:

$$H^{\dagger} = (H^{\dagger} H)^{-1} H^T \tag{4.11}$$

Moore–Penrose generalized inverse of H can be computed using different methods. The orthogonalization method and iterative method are candidate operators for achieving this inverse. Nonetheless, they are somehow notorious since they are using iteration which we want to avoid. The orthogonal project method can be used to calculate the equation (4.10). This equation requires $H^T H$ to be nonsingular which is not the case in all applications, and thus orthogonal projection approach cannot reliable. SVD method [79] can be used to compute this inverse since it does not have to worry about H being a singular matrix [71].

Remark 4.1: Once the values of the weight vectors $a_i \in \Re^L$ and the bias $b_i \in \Re^L$arerandomly assigned at the beginning of the learning algorithm, they remain fixed and the matrix His unchanged.

Remark 4.2: Functions, such as, the sigmoidal, radial basis, sine, cosine and exponential functions are infinitely differentiable and can be used as activation functions for the network. In this work, we use Sigmoid function as it is used in most of the literatures. Sigmoid function can be expressed as:

$$Sig(a, b, x) = \frac{1}{1+e^{-(a.x+b)}} \tag{4.11}$$

Where b represents the hidden bias, x is the input feature vector, and a is the input weights connected to that neuron.

4.4. Chapter Summary

This chapter presents methods that are used for classification in our face recognition system. This part of the thesis also emphasizes on the impact of the classification stage on the performance of the entire recognition process. Two types of classifiers are presented: k-nearest neighbor (k-NN) and Extreme Learning Machine (ELM). The k-NN is a nonparametric method for measuring the similarity between the objects, and mathematically is considered to be a simple method. This method however, shows reliability, and it has been adopted in many experimental evaluations in pattern recognition. On the other hand, ELM presented as an elegant algorithm for learning Single-hidden Layer Feedforward Neural Networks (SLFNs). Unlike other neural networks, the parameters of this network do not need to be adjusted. This feature, indeed, makes this classifier extremely fast and accurate. The only condition of ELM is that the activation function in the hidden neurons must be infinitely differentiable.

Chapter 5

Experimental Evaluation

5.1. Introduction

This chapter presents a comprehensive experimental evaluation of different human face recognition schemes when the facial images are in low resolution. The feature extractors discussed in chapter three, namely Principal Component Analysis based Discrete Wavelet Transform (PCA based DWT), Local Binary Patterns (LBP) and Block Based Discrete Cosine Transform (BBDCT) are used. After extracting the feature vector from the target image it is sent to classifier. For comparison purpose, we use the same classifier on the same database for all feature extraction methods. Two classifiers are used, namely k-NN and neural network. When the resolution of the input image is low, interpolation is used as a preprocessing step to improve the quality of the image for better recognition performance. As mentioned in chapter two, three interpolation techniques are used, namely nearest neighbor, bilinear and bicubic interpolant. There are two experiments designed to achieve this task. The first experiment is evaluation of the performance of each feature extraction method against the resolution decrease. This is done by calculating the recognition rate at different resolutions. The recognition rate, as will be discussed later in this chapter, is used for measuring the ability of the system to identify a given face image. In the second experiment, the input images have low resolution and the interpolation techniques are employed to increase the resolution for better recognition performance.

5.2. Experimental Setup

The goal of these experiments is, first, to compare all the feature extraction methods in term of their identification rate when the image resolution decreases. The second goal is to employ the interpolation methods to reconstruct high resolution images from low resolution images to obtain better recognition performance. The block diagram in figure 5.1 shows face recognition system, which contains the processes carried out at each stage. Assuming that the facial images in our database are cropped and aligned, the other processes are as follows. For preprocessing stage we use three types of interpolation, namely nearest neighbor, bilinear and bicubic. For feature extraction stage three methods are used, namely PCA based DWT, LPB and BBDCT. For classification, *k*-NN and neural network are employed. Lastly, identification rate is calculated to evaluate the recognition accuracy. The system is implemented using MATLAB since it is supported by numerous tools and built-in functions.

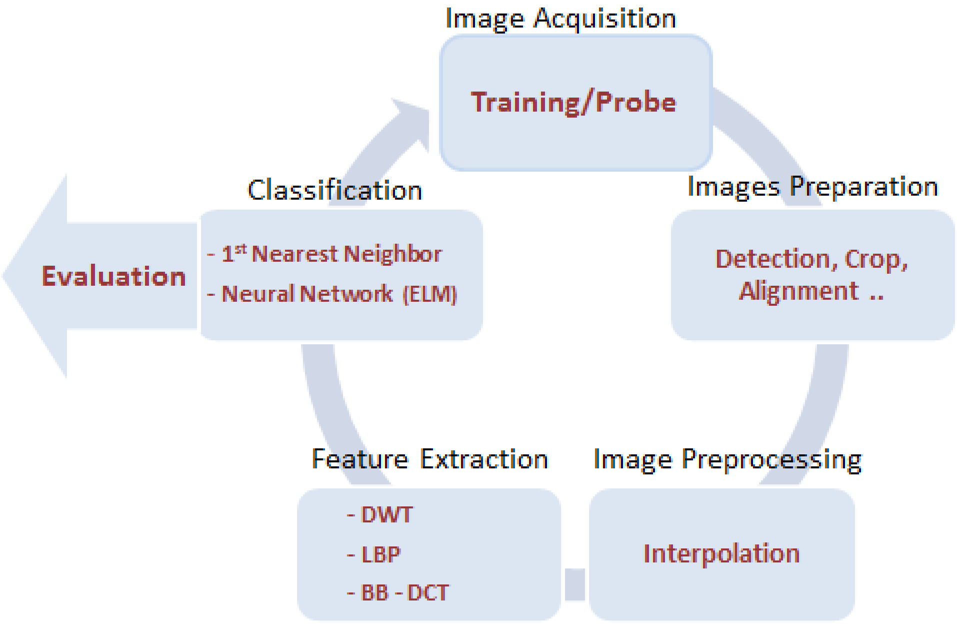


Figure 5.1: Structure of face recognition system configurations implemented in the experiments.

5.2.1. Database

To evaluate our system, ORL database (also known as AT&T)is used. It contains a set of face images taken between April 1992 and April 1994[80]. This database, as shown in figure 5.2 and table 5.1, consists of ten different images of each of forty distinct subjects. Images for some subjects were taken at different times with variation in the lighting, pose, and face expression (open / closed eyes, smiling / not smiling, gloomy, normal). All images are upright frontal position and were taken against a dark background. The images are in JPG format, and the size of each image is 92×112 pixels with 256 grey levels for representing the pixel value.

No. of classes	Images for each class	Image Resolution
40	400	92× 112

Table 5.1: Specifications of ORL database

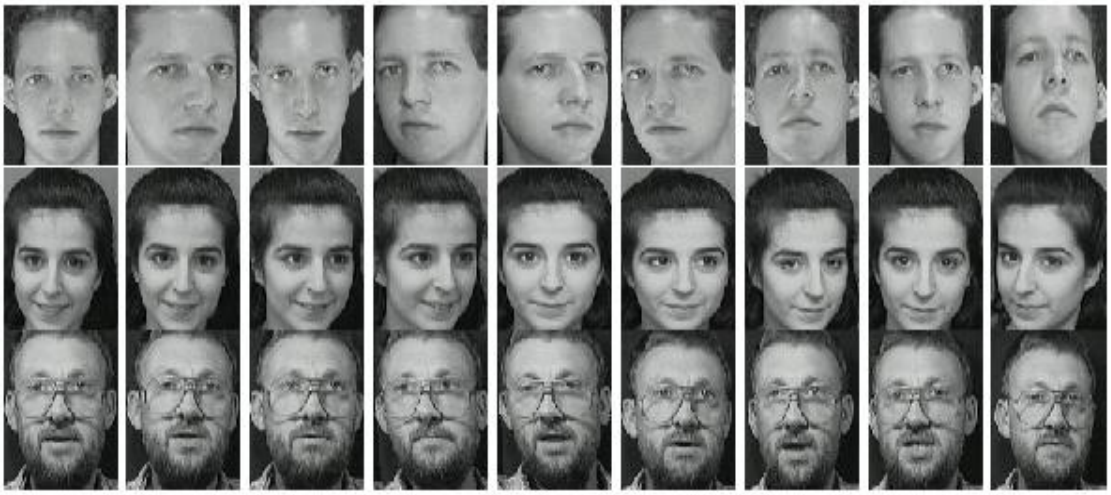


Figure5.2: Samples from ORL database.

5.2.2. Low Resolution Operator

In the experiments, we need to classify images of different resolutions. Therefore, the first step in the experiments is to generate those resolutions, because the images in our database are all in the same resolution of 92×112, This can be done using a down - sampling operator (Fig. 5.3) which takes place before the feature extraction stage. For this operator, an image in a specific resolution and the down-sampling factor are considered as inputs and the output is the image in lower resolution. For example, if the input image resolution is 112×92 and the down-sampling factor is 1/2, then the output image resolution is 56×46. The down-sampling factor is tuned in order to produce different resolutions.

High resolution face image
DOWNSAMPLING
low resolution face image

Figure 5.3: Low resolution face image formation model.

5.2.3. Classification

In these experiments we are employing two classifiers, k-NN and neural network. This section discusses the architectures and the settings of these classifiers.

5.2.3.1. k-Nearest Neighbor

The method of k-nearest neighbor (*k*-NN) is a supervised classification method. That means that to classify a given test sample, the classes of the training samples from the database must be known. Thereafter, the class of the test sample is predicted based on the majority voting of the k nearest neighbors of the test sample. Euclidean distance is used for measuring the distances between the test sample and the training samples as described in chapter four. Therefore, the only factor we need to manipulate is k. It has proven that increasing the value of k can make the classifier robust against the class noise; however, using a small k can provide better performance when a small database is used. Because we are using a relatively small database, k is set to one (k=1), by which the class of the test image is assigned to the class of its nearest neighbor image.

5.2.3.2. Neural Network

The second classification method used in our system is neural network. This network consists of three layers, the input layer, the hidden layer and the output layer. The input layer consists of n nodes, where n is equal to the number of the input vector components. The hidden layer consists of L neurons, where L can be estimated empirically [74]. The output layer consists of m nodes. Typically, one output node is used to represent one class. In our database we have forty classes; thus, there are forty output nodes in our neural network. When an input representation is applied to the network, the decision is given to the class that corresponds to the output node with the maximum value among all the outputs in the other nodes.

The neural network is a parametric classifier, and to adjust these parameters a training process is needed. As mentioned in chapter four, we use Extreme Learning Machine (ELM) for learning our network [73]. The hidden layer output matrix H is initialized by random values from -1 to 1. Unlike other neural networks, the learning phase in the ELM is not achieved iteratively. That is, all computational time of this method is consumed in computing the Moore-Penrose pseudo inverse of H [71]. Fast learning and testing speed offered by ELM allow us to repeat each classification process twenty times and the average recognition rate is reported.

The recognition rate (RR) is used in our experiments to measure the accuracy of our classification systems. This measure relates to the test's ability to identify given objects correctly. Thus, it can be defined as the percentage of the correct matched (identified) test images to the total number of test images, and it can be written as:

$$\text{Recognition Rate(RR)} = 1 - \frac{\text{error matches}}{\text{total probe images}} \times 100\ \% \qquad (5.1)$$

Indeed, it is important to estimate the number of the hidden neurons in our network. In order to do that, an experiment is carried out on the neural network recognition system where the number of the hidden neurons is changed and recognition rate is reported. For feature extraction, BBDCT is used for ORL database images of 92×112. The training and testing time is also estimated in order to demonstrate the speed of this classifier. In this experiment, a 2.80 GHz PC with 4 GB of RAM is used. The recognition rate (RR), the training and testing time for various numbers of hidden neurons are listed in the table 5.2. From the table, we notice that as the number of hidden neurons increases, the recognition rate increases and a negligible change in the rate is observed beyond 120 neurons. The best performance is obtained using 100 neurons. We also notice that ELM is extremely fast compared with other types of neural networks.

	Number of hidden neurons								
	L=10	L=20	L=30	L=50	L=60	L=80	L=100	L=120	L=190
RR %	47.88	71.67	84.38	89.17	90.83	92.08	93.98	93.96	90.63
Training Time(Sec)	0.000	0.019	0.035	0.066	0.016	0.043	0.063	0.086	0.0118
Test Time(Sec)	0.000	0.000	0.008	0.000	0.008	0.004	0.008	0.00	0.008

Table 5.2: Varying the number of hidden neurons of ELM neural network: Average recognition rate (%), training time and test time using different numbers of neurons

5.2.3.3. Training-to-Testing-Samples Ratio

The performance of the classifier is usually influenced by the number of the training and test samples. Theoretically, as the number of training samples increases, the accuracy of the classifier increases. To examine this assumption and to decide how our database will be separated into training and testing parts, an experiment is designed for our two classifiers: the *k*-NN and the ELM neural network. Our database consists of 400 images: ten images from each of the forty subjects. In this experiment, we vary the training-to-testing-samples ratio from 1:9 to 9:1, and each time we report the average recognition rate of twenty iterations. For comparison purposes, the images selected for training and testing at each division are the same for the both classifiers. Also, the images are used in the original size of 92×112 and the BBDCT is used for feature extraction where the image is divided into 8×8 blocks. The results are depicted in figure 5.4. The graph demonstrates that for the two classifiers, the recognition rate increases as the number of the training samples increases. However, the k-NN provides better recognition rates as the number of training samples decreases. Moreover, the difference between the RRs of the two classifiers increases as the number of training samples decreases. That is, when we use one sample for training and nine samples for testing, the RR of the k-NN is about 16% higher than the RR of the ELM. As the number of the training samples increases, the RRs of the two classifiers get closer till they have the same value when we use nine samples for training and one sample for testing. We conclude from this experiment that neural network demands high number of training samples for better results. Therefore, using k-NN is preferable when a small number of samples are available for training, which is the case in a small database. ELM should, in contrast, provide a promising performance for high number of training samples in a huge database.

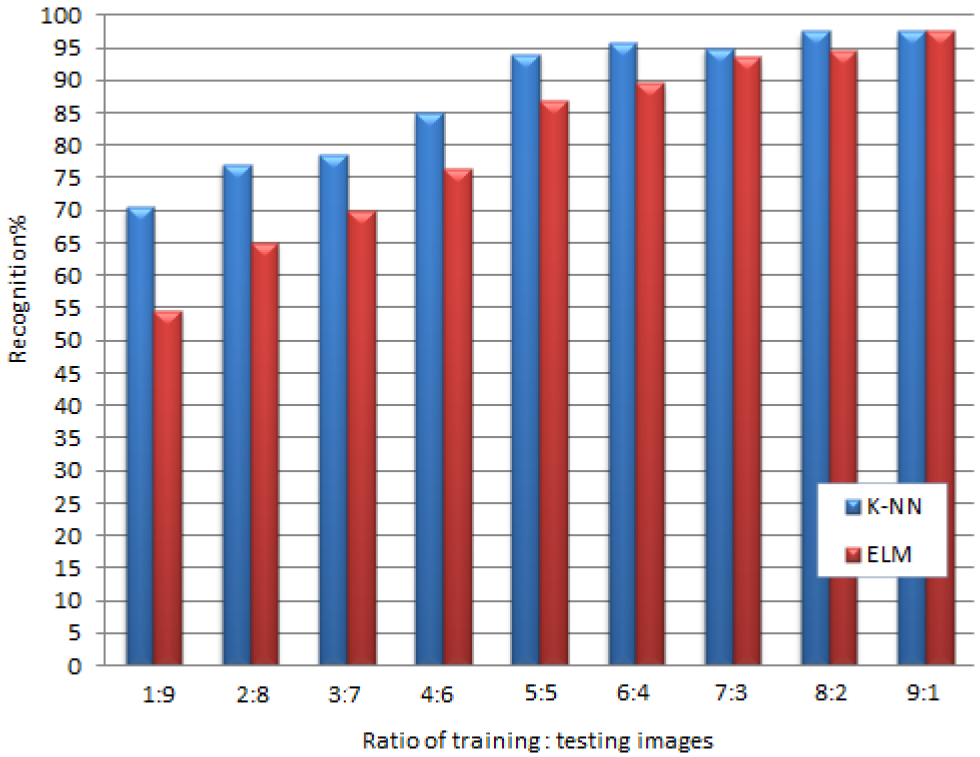


Figure 5.4: Recognition rate for varied training : testing-samples ratio using k-NN and ELM classifiers.

Based on these results, we decided to split our database into two non-overlapped sets; six images of each class are used for training and the remaining four are for testing.

5.3. Experiments and Discussion

The aim of our work is to use interpolation techniques to improve the performance recognition system on low resolution face database. Before we do that, we need to demonstrate the performance of our feature extractors against the resolution decrease which will be discussed in the next section.

5.3.1. Recognition of Down-sampled Images without Interpolation

In this section, we design a series of experiments to compare the performance of feature extraction schemes, presented in chapter three, for face recognition. Those schemes will be examined in terms of resolution variation. These experiments will provide us useful information and conceptions about which feature extraction scheme is superior over range of resolutions, from high resolutions HR down to low resolutions LR. Instead of taking face images at different distances from the camera to obtain low resolution images, we use the down-sampling operator presented in section 5.2.2. This operator is applied for all 92×112 images in our database and generates the following resolutions: 56×49, 28×23, 19×16, 14×12, 12×10, and 8×7. An example of an original size image and its down-sampled versions is shown in figure 5.5. The feature extraction methods introduced in chapter three, namely, PCA/DWT, Local Binary Patter and Block-Based Discrete Cosine Transform will be used separately for representing ORL face images of the six resolutions mentioned above. The *k*-NN classifier will be used for classification and to declare a match. Therefore, our face recognition system for these experiments will be as shown in figure 5.6.

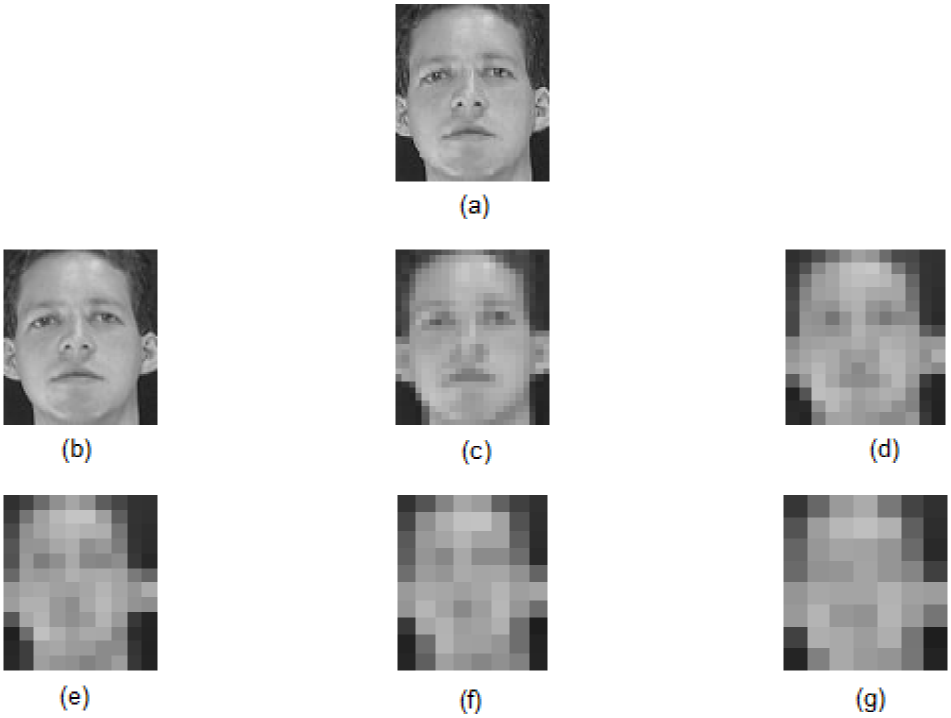


Figure 5.5: Samples for downsampled face image:

(a) the original face image 112×92. (b) downsampling the image in -a by a factor of 1.2 to the size of 94×82. (c) by factor of 4 to the size 28×25. (d) by factor of 8 to the size 14×13. (e) by factor of 10 to the size 13×10. (f) by factor of 12 to the size 10×9. (g) by factor of 14 to the size 8×7.

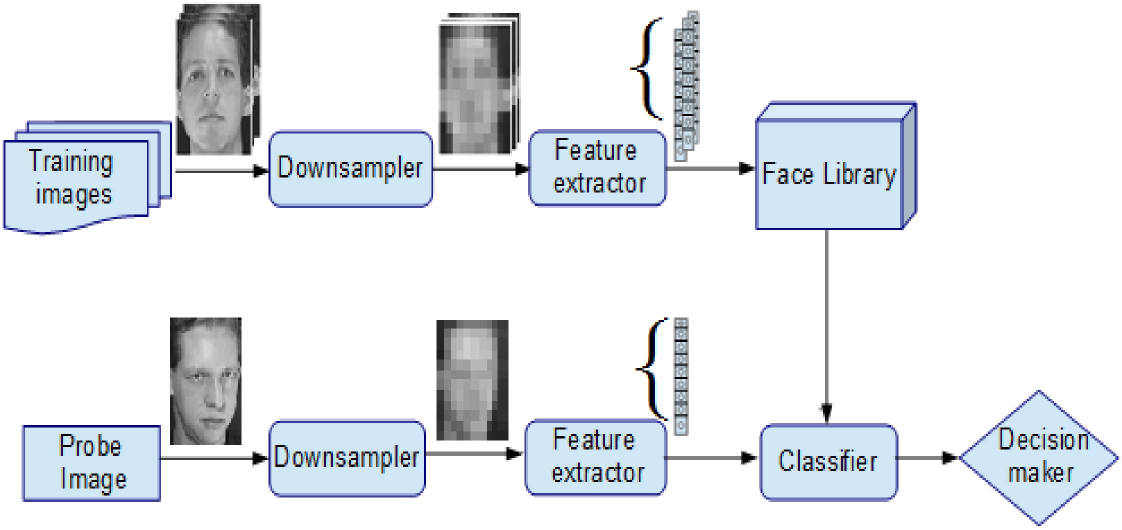


Figure 5.6: Face recognition block diagram for resolution variation.

5.3.1.1. PCA Based DWT Recognition System

Principal Component Analysis (PCA) is one of the most common feature extraction approaches in face recognition. Detailed information about the concept and the algorithm procedures of PCA is mentioned in chapter three. It is also mentioned that this method encounters some issues. The main issue is the high computational complexity when computing the eigenvalues of the covariance matrix in the training phase; this issue shows up when a large database or images of high resolution are used. Another limitation of PCA is that it extracts features from the image considering only the data variance regardless of any other discriminative aspect of the data. To cope these issues, images need to be processed before they are applied to the PCA. In this experiment, Discrete Wavelet Transform (DWT) is applied to decompose the input images before they are fed to the PCA. When applying 2D DWT, the image is decomposed into four sub-bands LL, LH, HL, and HH sub-bands respectively. The band LL represents the approximate image which holds the low frequency components of the original image. These components, indeed, are the most wanted in face recognition task. Besides, the dimensions of the approximate image are half the of the original image dimensions, which leads to less computational complexity. For example, if the size of the input image is 128×128, then the LL approximate image is 64×64. Figure 5.6 shows the block diagram this method for face recognition. From the figure, this system basically consists of two phases; the training and the testing (recognition) phase. In the training phase, images from all candidate classes are represented in the feature extraction stage. In this stage, images are decomposed using DWT at a certain number of levels. The output approximate images are then fed to the PCA. Those principal components based representations are then stored in the feature library. Thereafter, when unknown image is introduced to the recognition stage, 2D DWT and PCA are applied to transform the unknown image into the representational basis identified in the training stage. After that, the matching is done using our classifier. It is worth mention that the number of wavelet decomposition levels is chosen depending on the size of the original image so we assure that the output approximate image holds sufficient discriminative components. That is, the smaller the

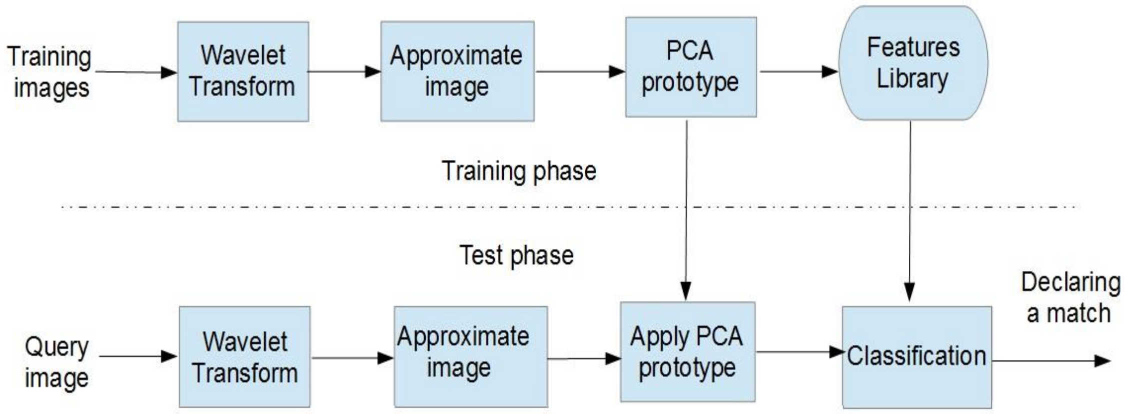


Figure 5.7: Block diagram of wavelet based PCA recognition system.

input image, fewer number of decompositions is needed. When the input image is in very low resolution, no decomposition is needed. The experiment is repeated seven times and each time we use database of the following resolutions: 112×92, 56×49, 28×23, 19×16, 14×12, 12×10, and 8×7. The result of the recognition rate versus the image resolution is depicted in figure 5.9.

5.3.1.2. Local Binary Pattern (LBP) Face Recognition Scheme

The second face recognition system we examine in this work is the Local Binary Pattern (LBP) based face recognition scheme, as described in chapter three. To assess this approach, the same database will be used with the same training-to-testing-samples ratio for comparison purpose. There are some settings that influence the performance of this method significantly. This includes the LBP operator and the number of regions that we need to divide our images into. The$LBP_{8,1}^{u2}$ operator is chosen in this experiment which refers to using only the uniform patterns in the face texture and using (8,1) neighborhood for calculating the binary pattern. For measuring the distances between the histogram representation, Chi distance is used. The number of regions that the facial images should be divided by is an important factor. Since the size of our face images varies from 112×92 to 8×7, a small number of regions are needed. For resolutions from 8×7 to 19×16, images are divided into 2×2, while larger images are divided by 5×5 regions. The graph of the recognition rate versus the image resolution is also shown in figure 5.9.

5.3.1.3. Block-Based Discrete Cosine Transform Face Recognition Scheme

Block-Based Discrete Cosine Transform (BBDCT), for feature extraction, is presented in chapter three. In this section, The BBDCT algorithm for face recognition is tested using the same database with the same settings for comparison purpose. The typical BBDCT based identification system is shown in figure 5.8 where the *k*-NN classifier is used for identifying the test images. Indeed, there are some remarks regarding this algorithm when the images resolutions is changed (refer to chapter three, section 3.6). When we take those settings into consideration while using BBDCT for image representation, the results in table 5.3 are obtained.

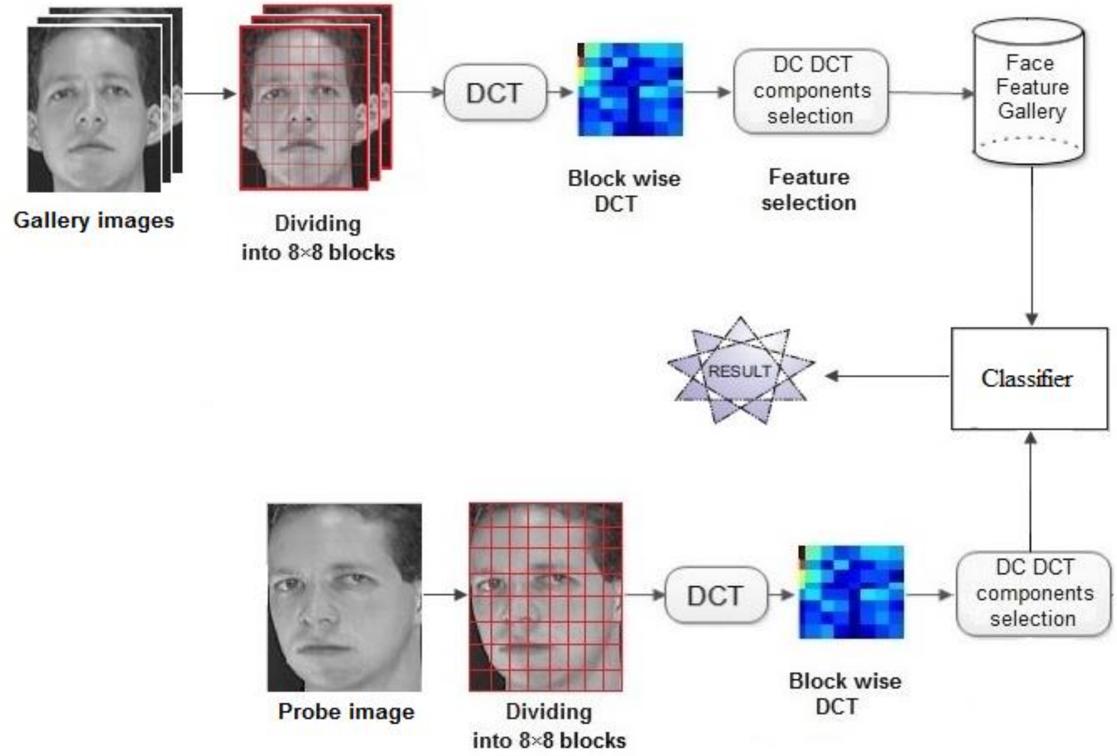


Figure 5.8: Block diagram of the face recognition system using BB DCT.

Image size Blocks No.	8×7	12×10	14×12	19×17	28×25	56×49	112×92
One block	92.5	93.5	92.5	93	91.88	91.88	91.88
2×2 blocks	65.63	68.13	88.0	59.37	61.25	62.5	62
3×3 blocks	81.25	91.8	83.1	88.75	89.38	88.13	70.3
4×4 blocks	88.25	92.3	92.5	92	90.63	91.38	92.5
5×5 blocks	\	93.75	92.5	93.13	92.5	93.13	92.5
6×6 blocks	\	94.3	92.5	92	93.13	95	94.3
7×7 blocks	\	\	\	93.13	95	95	95
8×8 blocks	\	\	\	93.75	95.5	95	95.63

Table 5.3: Recognition rate of BBDCT scheme for different image blocks and image resolutions

5.3.1.4. Results and Discussion

The recognition rates, reported against the resolution variation for the three feature extractors, are shown in figure 5.9. From the graph, we notice that the recognition rates for the schemes are directly proportional to the image resolution but in different ways. For LBP-based face recognition system, the recognition rate (RR) decreases gradually as the image resolution of the training and testing sets decreases. The decrease in the RR starts from 96.3% at 112×92 and ends down to 65.5% at 8×7. When DWT is used for face representation, however, the RR drops drastically at a threshold resolution around 12×10 pixels while for resolutions from 112×92 down to 14×12, the recognition rate decreases by only 5%.On the other hand, the BBDCT based recognition system shows more robustness against resolution. The RR of this scheme decreases slightly by 5% as the image resolution decreases from 112×92 to 8×7.

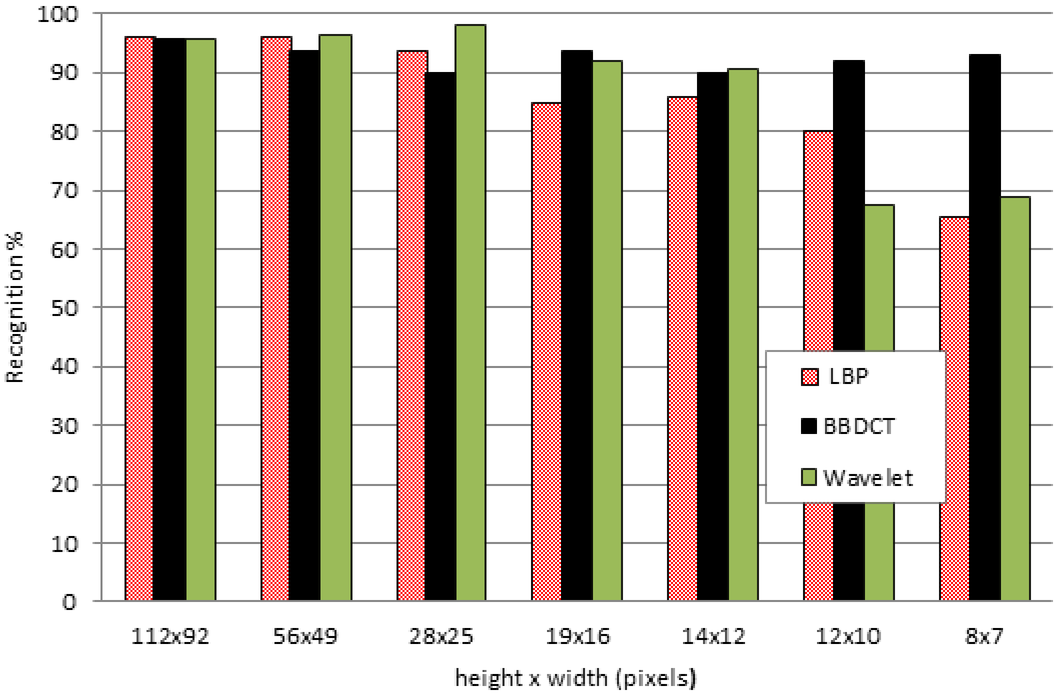


Figure 5.9: The recognition rate Vs. image resolution for the three feature extraction schemes, PCA based DWT, LBP, and BB DCT.

5.3.2. Utilizing Interpolation for Recognizing Low Resolution Images

In this part of our experiments, we simulate the real-life scenario when the face images for face recognition, either the gallery or the probe images, are in low resolution. Image interpolation techniques, namely, nearest neighbor, bilinear and bicubic are employed as a preprocessing stage to reconstruct high resolution (HR) images from low resolution (LR) images for better identification performance. Unlike other techniques used for improving the performance of low resolution face recognition system, interpolation can be done on a single frame and does not need high resolution gallery for training; as well, image interpolation is a fast technique and can be done on a small database. The typical recognition system that achieves our proposed method is shown in figure 5.10. In this experiment, the two classifiers are used: k-NN and ELM.

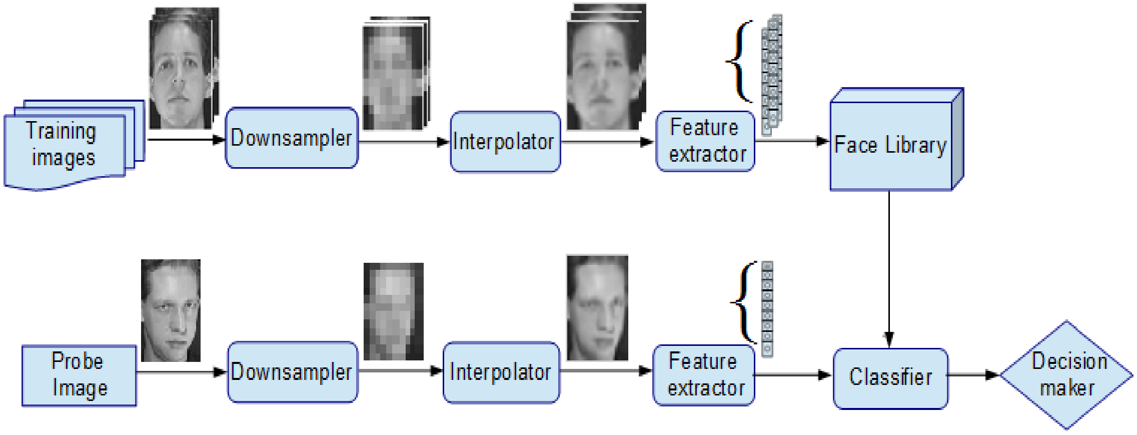


Figure 5.10: Proposed face recognition system: Interpolation for low resolution face recognition.

5.3.2.1. Using k-NN Classifier

In order to examine the performance of each interpolation method, we apply the reconstructed images after interpolation to our face recognition systems. Three face recognition systems are designed for comparison purpose, namely, DWT, LBP and BBDCT based recognition systems. The *k*-NN method (refer section *5.2.3.1*) is used for classification.

When bicubic interpolation is used, the recognition rate (RR) versus the image resolution is plotted in figure 5.11. For LBP scheme, the RR increases gradually for resolutions higher than 10×9 and retrieves the RR of the original HR images at 94×82. For low resolution, LBP shows, relatively, low performance since the LBP image is smaller than the corresponding grey-scale image. This is because the edge rows and columns in the grey scale input image have no representation in the LBP image. For example, for an input grey-scale image of 10×9, the corresponding LBP image is 8×7, a change which causes damage in the image representation. On the other hand, when DWT is used for representation, the recognition system shows a threshold resolution at about 10×9 where the RR jumps by 20%. For resolution 13×10 and above, the RR remains stable. BBDCT scheme, however, enjoys a stable performance versus the resolution variation for resolutions from 8×7 to 94×82. It is worth mentioning here that the number of blocks in this method is changeable according to the size of the image as shown in the previous experiment.

Bilinear interpolation is also used to increase the resolution of 8×7 images and the RR at different image sizes is depicted in figure 5.12. We observe that the results are very similar to those we obtained using bicubic, although, there is a difference in the computational cost between bilinear and bicubic interpolations, as mentioned in chapter two. Therefore, if hardware with limited specifications is used to implement our system, priority can be given to bilinear interpolation over bicubic to increase the resolution of LR images at least when these face recognition schemes are used.

The last interpolation method used in this experiment is nearest neighbor method. The graph in figure 5.13 depicts the performance of the face identification schemes when nearest neighbor is used for reconstructing HR images from 8×7 images. We observed that DWT and BBDCT schemes performed similarly to bicubic and bilinear, however, LBP provides an unfortunate performance. The RR of LBP scheme increases relatively slowly as the resolution increases from 8×7 to 14×13 and then it drops drastically down to 5.5% at 94×82.

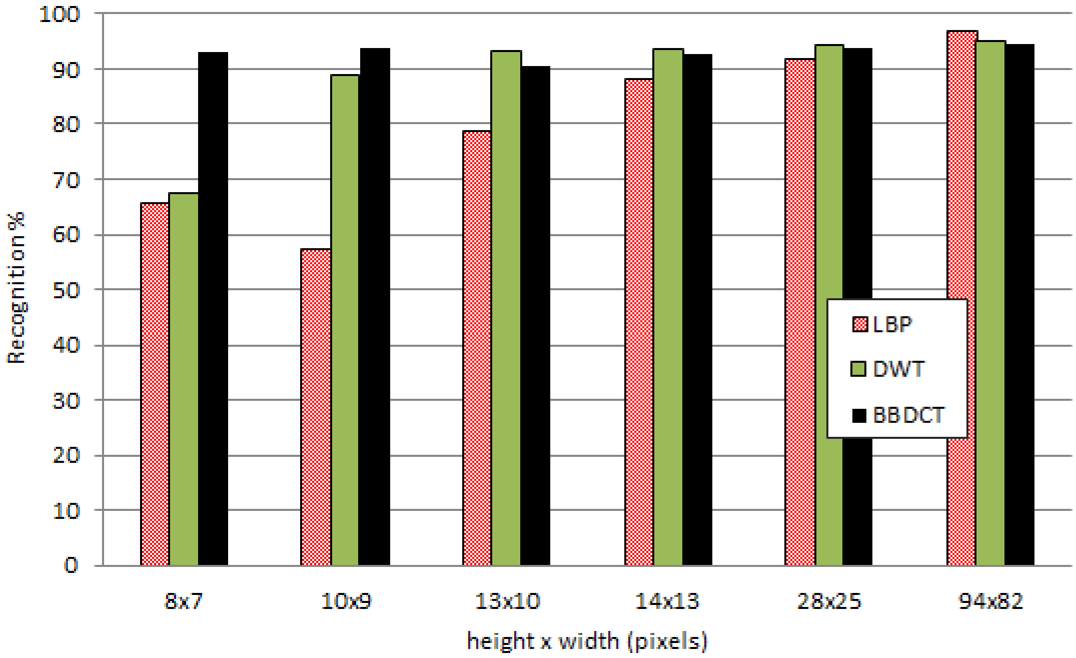


Figure 5.11: RR for reconstructed image from LR of 8×7 using bicubic interpolation for the three feature extraction based recognition schemes.

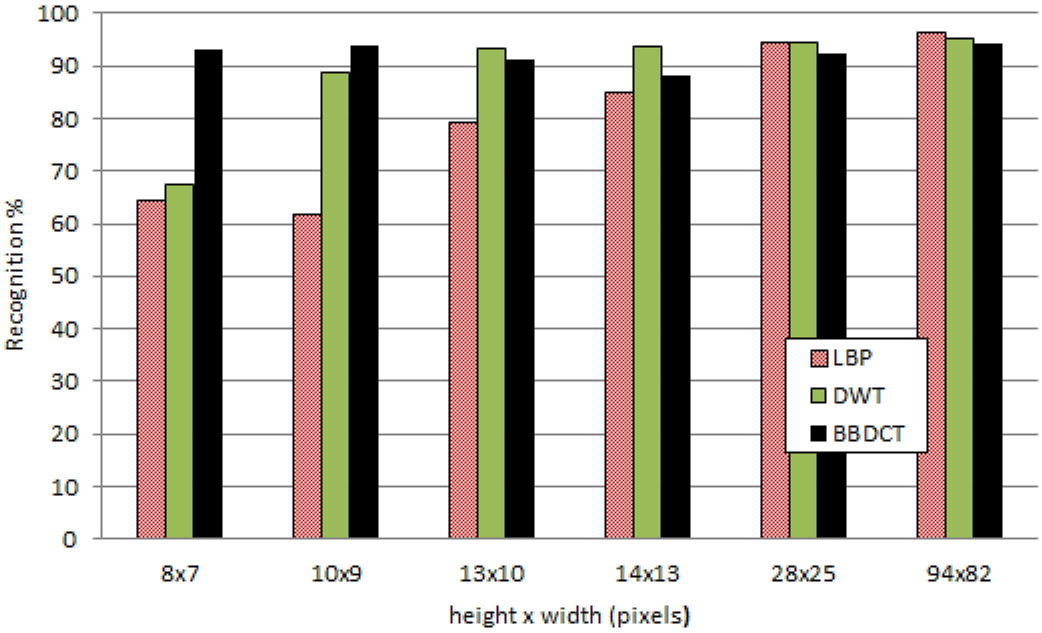


Figure5.12: Recognition rate for reconstructed image from LR of 8×7 using bilinear interpolation for the three feature extraction based recognition schemes.

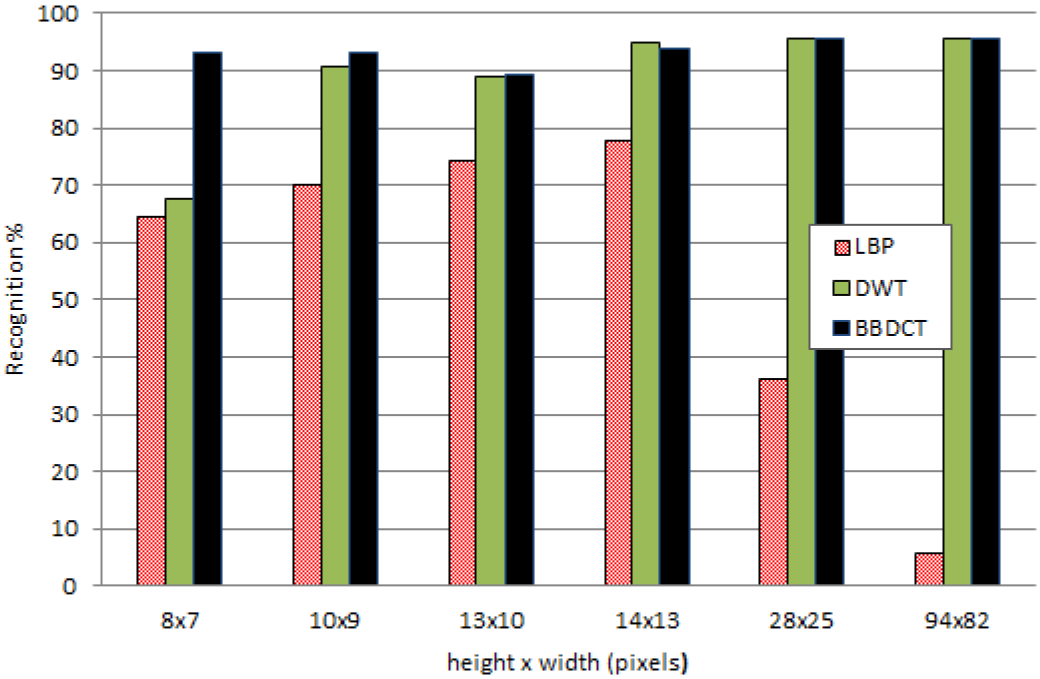


Figure 5.13: Recognition rate for reconstructed image from LR of 8×7 using nearest neighbor interpolation for the three feature extraction based recognition schemes.

The reason for this issue is that the nearest neighbor interpolation algorithm selects the value of the nearest neighbor without any change, which yields a piecewise-constant interpolant. This leads to a repetition in the pixel values of original LR image which in turn, implies information redundancy in the reconstructed image. Because LBP is a histogram based image descriptor, it does not reduce the dimensionality of the input image nor can it be configured with any statistical dimensionality reduction method. The issue of information redundancy in DWT is resolved using PCA which eliminates the redundant data, and provides better performance as the image resolution increases.

For more illustration of information redundancy, an example is shown in figure 5.14.A low resolution image of 14×12 is reconstructed by factor of eight using bicubic and nearest neighbor interpolations. It is clear from the figure that bicubic interpolator produces a blurred image, yet it contains new grey-scale levels. The interpolated image, using the nearest neighbor, however, contains the same grey scale levels as in the original LR image. This is because the new pixel values in this method are just a repetition of existed pixels in the original image.

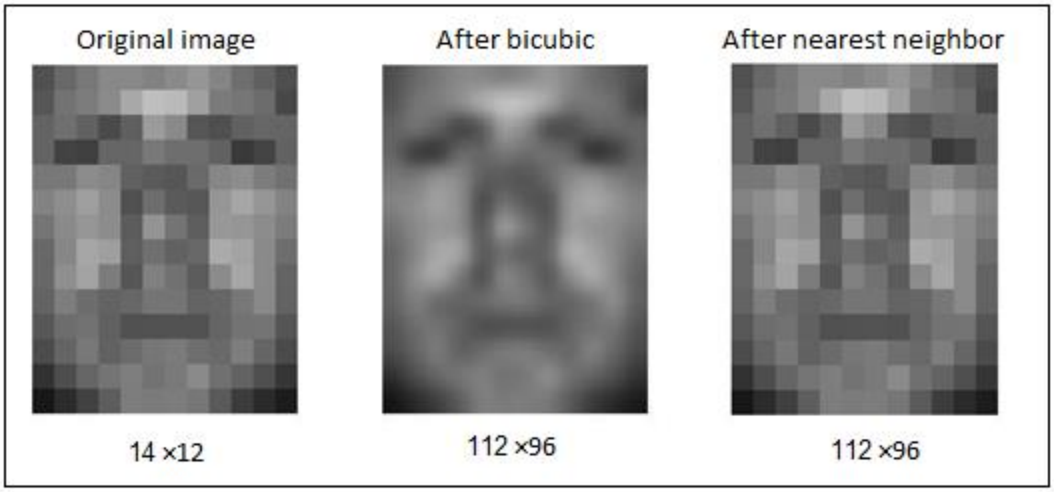


Figure 5.14: Comparison between nearest neighbor and bicubic interpolation. a) Original Reconstructing 112×96 image from 14×12 using bicubic and nearest neighbor.

5.3.2.2. Using ELM Neural Network Classifier

In this experiment, we examine the Neural Network (refer to section5.2.3.2) instead of *k*-NN, with the aim of identifying low-resolution face images. In order to achieve this, face recognition algorithm based on BBDCT and Extreme Learning Machine (ELM) neural network is implemented. The block diagram of this system is shown in figure 5.15. We use the table 5.3 to choose the number of blocks for the BBDCT at each image size. The steps of this experiment are briefed as following: first, the training and the test images are down-sampled to 8×7. Second, we start upsizing the images using our interpolation methods. At each size, we extract the candidate DCT components and apply them to our neural network. The results are depicted in figure 5.16. From the figure, we notice that the BBDCT feature extractor provides relatively high performance at low resolutions, and it shows insensitivity against resolution variation; though, interpolation techniques success to improve the recognition rate by 2-3.5%. As well, we notice that there is no significant difference between the *k*-NN and ELM methods in terms of the identification rate.

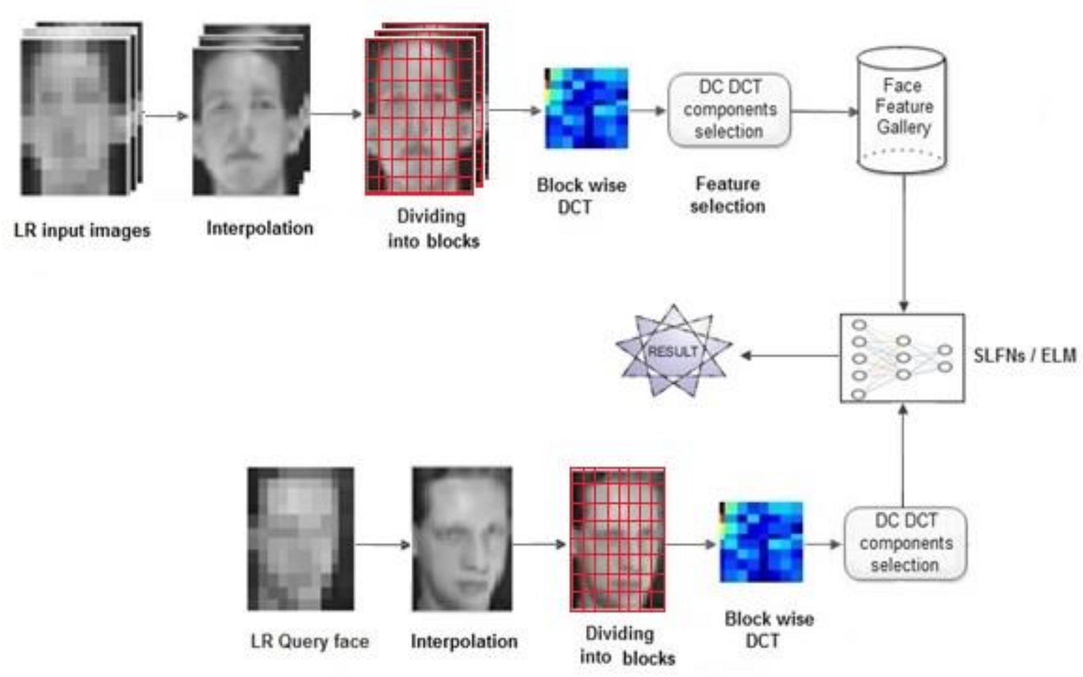


Figure 5.15: Block diagram of face recognition using BBDCT and ELM.

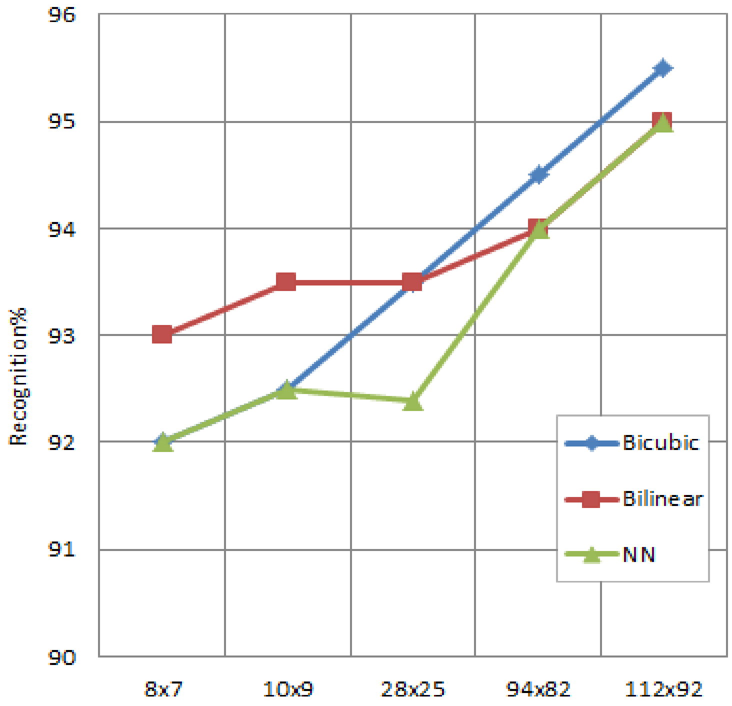


Figure 5.16: Recognition rate for reconstructed image from 8×7 using the three interpolation techniques. This face recognition scheme uses BBDCT feature extraction and ELM for classification.

5.4. Chapter Summary

Experiments in recognizing faces from low-resolution images using interpolation techniques are conducted in this chapter. Interpolation techniques presented in chapter two, namely nearest neighbor, bilinear, and bicubic are used as preprocessing step to increase the resolution of low-resolution images. The ORL database is used to evaluate the performance of the presented face recognition systems. For classification, k-NN and Extreme Learning Machine ELM neural network methods, presented in chapter four, are employed. The first experiment is conducted on the three feature extraction methods presented in chapter three, namely, PCA based DWT, LBP and BBDCT. In this experiment, the performance of these feature extraction methods for face recognition against the resolution decrease is demonstrated. The recognition rate RR, reported for each extractor, decreases as the image resolution decrease. The performance of BBDCT, however, showed more insensitivity for the resolution reduction when a specific number of blocks are chosen for each range of resolution. In the second experiment, low resolution images are interpolated in a preprocessing stage before the candidate features are selected to represent the image. The same three feature extractors mentioned above are used in this experiment. The database images are first, down-sampled to create a low resolution version of the database. Thereafter, the three interpolation techniques are used to increase the resolution of the LR images. The interpolation processes success to improve the recognition rate of the low resolution images and the performance of the original images is retrieved. The only exception is that nearest neighbor interpolation failed to improve the RR for LBP representation. Experimental results also showed that ELM is a fast classification method yet it requires a large number of training samples, compared to the *k*-NN method, to obtain higher accuracy.

Chapter 6

Conclusion

6.1. Conclusion

The research work conducted in this thesis addresses the issue of recognizing faces from low resolution images. Interpolation techniques are used as a preprocessing step to improve the recognition rate of those images. This research covers two aspects of the problem of face recognition from low resolution images as follows.

First, the performance of several face recognition algorithms, namely PCA based DWT, LBP, and Black-Based DCT, is evaluated for low-resolution face images. The experiments are conducted on ORL database. The results show that DWT based PCA system performance has a threshold resolution at about 12×10 where recognition rate drops significantly for resolution below this threshold, and no significant decrease in the recognition rate for resolutions higher than the threshold. LBP shows more sensitivity for the image resolution since the recognition rate decreases gradually as the image resolution decreases. BBDCT, however, enjoys a stable and robust performance against resolution decrease. It is worth to mention that in order to obtain such a performance, the number of the image blocks should be varied according to the image size. That is, for high resolution images, 8×8 blocks are used, and it is decreased as the image size decreases to be one block at 8×7 pixels. The same consideration should be taken into account when LBP is used. The image regions are changes from 5×5 down to 2×2 as the resolution decreases. For DWT method, the number of wavelet decompositions is limited and it decrease as the image resolution decreases to ensure that a sufficient image components are remained to apply the PCA.

Second, interpolation techniques are applied to improve the recognition accuracy of a low resolution database of 8×7. Bicubic, bilinear and nearest neighbor interpolations are used for increasing the image resolution and significant improvement in the recognition rate is obtained. Bicubic interpolation uses a nonlinear kernel to estimate new pixel values while bilinear interpolation estimates a new pixel value within a pixel window linearly. Nearest neighbor, however, uses a constant-piecewise interpolant, which duplicates the original image contents. Bilinear and bicubic Interpolation techniques succeed to retrieve the recognition rate of the HR images. The LBP system responds relatively slowly to the increase in the resolution, and the original recognition rate is retrieved at 94×82. When PCA/DWT system is used the recognition rate improves significantly at low resolution of 10×9, and the original image performance is retrieved at 14×13. BBDCT however, shows a high and a stable performance all over the resolution interval, and slight improvement is reported. . Nearest neighbor interpolation also improves the recognition rate for BBDCT and DWT systems significantly, however, in LBP system it fails to retrieve the accuracy of the high resolution images, and the recognition rate drops drastically as the resolution increases. This is because LBP works alone without dimensionality reduction method, i.e. the redundancy occur in the nearest neighbour interpolated images cannot be removed in this scheme.

Bilinear and bicubic have quite similar performances for these recognition systems although of the fact that bicubic is computationally more expensive, which gives priority for bilinear in applications that are implemented with limited hardware specifications.

Lastly, the k-nearest neighbour (k-NN) and Extreme Learning Machine-based Single-hidden Layer Feedforward Neural Networks (ELM/SLFNs) are used for classification. The results mentioned above are obtained when k-NN is used. When ELM is used also the performance of BBDCT system is improved, and succeeds to retrieve the recognition rate of the original high resolution images. This research also points out that the ELM is a fast classification methodbut it is more influenced by the number of the training samples than k-NN, which means that itrequires a large number of training samples to obtain higher accuracy. Therefore, using k-NN is preferable when a small number of samples are available for training, while ELM should give promising results for large databases.

6.2. Future Work

This research addresses a challenge of having images in low resolution in the human face recognition. This problem occurs in an essential real-life application where the object locates at a distance from the camera. Therefore, an appropriate database is chosen for this work so the challenge it has is the low resolution. The uncontrolled environment scenario indeed, implies other challenges such as, the bad illumination. Future work can address this issue besides the low resolution using an appropriate preprocessing step.

References

[1] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld. "Face recognition: A literature survey." *Acm Computing Surveys (CSUR)* 35, no. 4 (2003): 399-458.

[2] S. Akamatsu. "Computer recognition of human face -A survey." *Systems and Computers in Japan* 30, no. 10 (1999): 76-89.

[3] Abate, A. F., M.Nappi, D.Riccio, and G.Sabatino. "2D and 3D face recognition: A survey." *Pattern Recognition Letters* 28, no. 14 (2007): 1885-1906.

[4] D. Litwiller. "CCD vs. CMOS." *Photonics Spectra*, *35*(1), (2001): 154-158.

[5] Hjelmås, E., and B. K. Low. "Face detection: A survey." *Computer vision and image understanding* 83, no. 3 (2001): 236-274.

[6] P. Viola, M. J. Jones, and D. Snow. "Detecting pedestrians using patterns of motion and appearance." *International Journal of Computer Vision* 63, no. 2 (2005): 153-161.

[7] T. Huang, Z.Xiong, and Z. Zhang. "Face recognition applications." In *Handbook of Face Recognition*, Springer London, (2011):617-638

[8] Y. M. Lui, D. Bolme, B. A. Draper, J. R. Beveridge, G. Givens, and P.J. Phillips, "A meta-analysis of face recognition covariates," in *Proc.Int. Conf. Biometrics: Theory, Appl. Syst.*, (2009): 1–8.

[9] Boom, B. J., G. M. Beumer, L. J. Spreeuwers, and R.N.Veldhuis. "The effect of image resolution on the performance of a face recognition system." In *Control, Automation, Robotics and Vision, 2006. ICARCV'06. 9th International Conference on*, IEEE,(2006):1-6.

[10] J. Wang, C. Zhang, and H.Y. Shum, "Face image resolution versus face recognition preformance based on two global methods," *Proceedings of Asia Conference on Computer Vision*, 2004.

[11] C. Shan, S. Gong, and P. W. McOwan. "Robust facial expression recognition using local binary patterns." In *Image Processing, 2005. ICIP 2005. IEEE International Conference on*. Vol. 2, IEEE, (2005): II-370

[12] J. Wang, Z. Changshui, and H. Y. Shum. "Face image resolution versus face recognition performance based on two global methods." In *Asia Conference on Computer Vision*. 2004.

[13] S. Baker and T. Kanade, “Hallucinating faces,” in *Fourth InternationalConference on Automatic Face and Gesture Recognition*, (March 2000): 83–89.

[14] S. Baker and T. Kanade, “Limits on super-resolution and how to breakthem,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 9, (Sep. 2002):1167–1183

[15] Y. Peng, B.Gökberk, L. J. Spreeuwers, and R. N. J. Veldhuis. "An Evaluation of Super-Resolution for Face Recognition." (2012).

[16] W.Zou, W.W., and P. C. Yuen. "Very low resolution face recognition problem." *Image Processing, IEEE Transactions on* 21, no. 1 (2012): 327-340.

[17] S. Biswas, K. W. Bowyer, and P. J. Flynn. “Multidimensional scaling for matching low-resolution facial images.” In *IEEE 4th International Conference on Biometrics:Theory, Applications and Systems(BTAS), 2010 Fourth IEEE International Confer-ence on,* IEEE, (2010): 1-6.

[18] S. Biswas, K. W. Bowyer, and P. J. Flynn. "Multidimensional scaling for matching low-resolution face images." *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 34, no. 10 (2012): 2019-2030.

[19] A. Chakrabarti, A. N. Rajagopalan, and R. Chellappa, “Super-resolutionof face images using kernel PCA-based prior,” *IEEE Trans. Multimedia*,vol. 9, no. 4, (Jun. 2007): 888–892.

[20] Jiji, C. V., and S. Chaudhuri. "Single-frame image super-resolution through contourlet learning." *EURASIP Journal on Applied Signal Processing* 2006, (2006): 235-235.

[21] X. Wang and X. Tang, "Hallucinating face by eigentransformation," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews,* vol. 35, no. 3, (Aug. 2005): 425-434.

[22] Y. Peng, B.Gökberk, L. J. Spreeuwers, and R. N. J. Veldhuis. "An Evaluation of Super-Resolution for Face Recognition." (2012).

[23] Park, S. C., M. K. Park, and M. G. Kang. "Super-resolution image reconstruction: a technical overview." *Signal Processing Magazine, IEEE* 20, no. 3 (2003): 21-36.

[24] T.M.Lehmann, C. Gonner, and K. Spitzer. "Addendum: B-spline interpolation in medical image processing." *Medical Imaging, IEEE Transactions on* 20, no. 7 (2001): 660-665.

[25] T. M. Lehmann, C. Gonner, and K. Spitzer. "Survey: Interpolation methods in medical image processing." *Medical Imaging, IEEE Transactions on* 18, no. 11 (1999): 1049-1075.

[26] S. W. Rowland, "Computer implementation of image reconstruction formulas," in *Image Reconstruction from Projections: Implementationand Applications,* G. T. Herman Ed. Berlin, Germany: Springer- Verlag, (1979): 9–70.

[27] J. D. Faires and R. L. Burden, Numerical Methods. Boston, MA: PWS, 1993.

[28] J. Fessler. 2D Interpolation. February 18, 2013. Retrieved from: http://web.eecs.umich.edu/~fessler/course/556/l/n-07-interp.pdf.

[29] K. Robert. "Cubic convolution interpolation for digital image processing." *Acoustics, Speech and Signal Processing, IEEE Transactions on* 29, no. 6 (1981): 1153-1160.

[30] J. A. Parker, R. V. Kenyon, and D. E. Troxel, "Comparison of interpolating methods for image resampling," *IEEE Trans. Med. Imag.*, vol. MI-2. (1983): 31–39.

[31] Y. Yui. "Digital image processing by hardware using cubic convolution interpolation." U.S. Patent 4,578,812. March 25, 1986.

[32] H. Hou, and H. Andrews. "Cubic splines for image interpolation and digital filtering." *Acoustics, Speech and Signal Processing, IEEE Transactions on* 26, no. 6 (1978): 508-517.

[33] Han, J. K., and S. U.Baek. "Parametric cubic convolution scaler for enlargement and reduction of image." *Consumer Electronics, IEEE transactions on* 46, no. 2 (2000): 247-256.

[34] R. Chellappa, C. L. Wilson and S. Sirohey. Human and machine recognition of faces: a survey, *Proceedings of the IEEE,* Vol. 83, No. 5, (1995):705-740.

[35] G. Chow and X. Li, Towards a system for automatic facial feature detection, *PatternRecognition*, Vol. 26, No. 12, (1993): 1739-1755.

[36] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld. "Face recognition: A literature survey." *Acm Computing Surveys (CSUR)* 35, no. 4 (2003): 399-458.

[37] M. Turk and A. Pentland, Eigenfaces for recognition, *J. Cognitive Neuroscience*, Vol. 3,(1991):71-86.

[38] D. L. Swets and J. J. Weng, Using discriminant eigenfeatures for image retrieval, *IEEE Trans. PAMI.*, Vol. 18, No. 8, (1996):831-836.

[39] D. Valentin, H. Abdi, A. J. O'Toole and G. W. Cottrell, Connectionist models of faceprocessing: A Survey, *Pattern Recognition*, Vol. 27, (1994):1209-1230.

[40] M. V. Wickerhauser, "Large-rank approximate component analysis with wavelets for signal feature discrimination and the inversion of complicated maps.", *J. Chemical Information and Computer Sciences*, Vol. 34, No. 5, (1994):1036-1046.

[41] Ahonen, T., Hadid, A., &Pietikainen, M. "Face description with local binary patterns: Application to face recognition". *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, *28*(12), (2006): 2037-2041.

[42] J. T.Chien and C. C. Wu. "Discriminant Waveletfaces and Nearest Feature Classifier for Face Recognition". IEEE Tran. on Pattern Analysis and Machine Intelligence, 24(12), (Dec. 2002):1644-1649.

[43] Hazim Kemal Ekenel and Bulent Sankur. Multiresolution Face Recognition.Image and Vision Computing, 23, (2005):469-477.

[44] C. Garcia, G. Zikos, and G. Tziritas. Wavelet Packet Analysis for Face Recognition.Image and Vision Computing, 18:289-297, 2000.

[45] M. Feretti, D. Rizzo, Wavelet transform architectures: a system level review, Proceedings of the Ninth International Conference ICIAP'97, 2, Florence, Italy, (1997): 77–84.

[46] B. L. Zhang, H. Zhang, and S. S.Ge. "Face Recognition by Applying Wavelet Subband Representation and Kernel Associative Memory."*IEEE Tran. on Neural Networks, 15(1),*(Jan. 2004):166-177.

[47] Ramesha, K., and K. B. Raja. "Face Recognition System Using Discrete Wavelet Transform and Fast PCA." *Information Technology and Mobile Communication*. Springer Berlin Heidelberg, (2011): 13-18.

[48] Yuela, P. C., D. Q. Dai, and G. C. Feng. "Wavelet-based PCA for human face recognition." In *Image Analysis and Interpretation, 1998 IEEE Southwest Symposium on*,IEEE, (1998):223-228.

[49] M. Feretti, D. Rizzo, Wavelet transform architectures: a system level review, Proceedings of the Ninth International Conference ICIAP'97, 2, Florence, Italy, (1997): 77–84.

[50] T. Ojala, M. Pietik¨ainen, and D. Harwood, “A comparative study of texture measures with classification based on feature distributions,” *Pattern Recognition*, vol. 29, no. 1, (1996):51–59.

[51] V. Govindarajan, and S. K. VVS. "Face Recognition using Block-Based DCT Feature Extraction." *Journal of Advanced Computer Science & Technology* 1.4 (2012): 266-283.

[52] L. Sirovich, and M. Kirby. "Low-dimensional procedure for the characterization of human faces." *JOSA A* 4, no. 3 (1987): 519-524.

[53] K. Fukunaga, *Introduction to Statistical Pattern Recognition* (Academic, New York, 1990).

[54] M. A. Turk, and A. P. Pentland. "Face recognition using eigenfaces." In *Computer Vision and Pattern Recognition, 1991. Proceedings CVPR'91., IEEE Computer Society Conference on*, IEEE, (1991): 586-591.

[55] S. Ruwan, B. Arachchige. Face Recognition in Low Resolution Video Sequences using Super Resolution. MSc. thesis, Rochester Institute of Technology. Rochester, NY .(August 2008).

[56] B. Moghaddam, W. Wahid and A. pentland. Beyond eigenfaces: Probabilistic matching for face recognition, *Proceeding of face and gesture recognition*, (1998): 30 –35.

[57] Yuela, P. C., Dai, D. Q., & Feng, G. C. “Wavelet-based PCA for human face recognition”. In *Image Analysis and Interpretation, 1998 IEEE Southwest Symposium on*, IEEE,(1998): 223-228.

[58] T. Ojala, M. Pietikainen, and D. Harwood. A comparative study of texture measures with classification based on featured distributions. Pattern Recognition, 29(1), (1996):51-59.

[59] Liu, Weifeng, Yanjiang Wang, and Shujuan Li. "LBP Feature Extraction for Facial Expression Recognition." *Journal of Information & Computational Science* 8.2 (2011): 412-421.

[60] C. Shan, S. Gong, & P. W. McOwan. "Robust facial expression recognition using local binary patterns." In *Image Processing, 2005. ICIP 2005. IEEE International Conference on*, vol. 2, IEEE, (2005): II-370.

[61] C. Sanderson and K. K. Paliwal, “Features for Robust Face-Based Identity Verification”, *Journal of SignalProcessing*, 83, (2003): 931-940.

[62] C. Podilchuk, X. Zhang. “Face Recognition using DCT-Based Feature Vectors”.*In Proceedings of theAcoustics, Speech, and Signal Processing, ICASSP’* U.S. Patent 5,802,208, issued September 1, 1998.

[63] W.Chen, M. J.Er, S. Wu, "Illumination compensation and normalization for robust face recognition using discrete cosine transform in logarithm domain." *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on* 36, no. 2 (2006): 458-466.

[64] J. Jiang, G. Feng, “Robustness Analysis on Facial Image Description in DCT Domain”, *Electronics Letters* 43 (24) (2007): 1354-1356.

[65] M.J. Er, W. Chen, S. Wu., “High Speed Face Recognition Based on Discrete Cosine Transform and RBF Neural Networks”, *IEEE Transactions on Neural Networks* 16 (3) (2005): 679-691.

[66] M.Vikas, et al. "DCT-based Reduced Face for face Recognition." *International Journal of Information Technology and Knowledge Management* 5 (2012): 97-100.

[67] Z. Pan, A. G. Rust, and H.Bolouri. "Image redundancy reduction for neural network classification using discrete cosine transforms." In *Neural Networks, 2000. IJCNN 2000, Proceedings of the IEEE-INNS-ENNS International Joint Conference on*, vol. 3, IEEE, (2000): 149-154.

[68] M. J. Er, W. Chen, and S. Wu. "High-speed face recognition based on discrete cosine transform and RBF neural networks." *Neural Networks, IEEE Transactions on* 16, no. 3 (2005): 679-691.

[69] C. I. Podilchuk, and X. Zhang. "Face recognition using DCT-based feature vectors." U.S. Patent 5,802,208, issued September 1, 1998.

[70] P. Parveen, and B.Thuraisingham. "Face recognition using multiple classifiers." *InTools with Artificial Intelligence, 2006.ICTAI'06. 18th IEEE International Confer-ence on*, IEEE, (2006): 179-186.

.

[71] G.B.Huang, Q. Y., and C. K. Siew. "Extreme learning machine: theory and applications." *Neurocomputing* 70, no. 1 (2006): 489-501.

[72] G.B.Huang, D. H. Wang, and Y. Lan. "Extreme learning machines: a survey." *International Journal of Machine Learning and Cybernetics* 2, no. 2 (2011): 107-122.

[73] G. B. Huang, Q.Y. Zhu, and C. K. Siew. "Extreme learning machine: a new learning scheme of feedforward neural networks." In *Neural Networks, 2004. Proceedings. 2004 IEEE International Joint Conference on*, vol. 2, IEEE, (2004): 985-990.

[74] G. B. Huang, Learning capability and storage capacity of twohidden- layer feedforward networks, IEEE Trans. Neural Networks 14 (2) (2003): 274–281.

[75] G. B. Huang, H.A. Babri, Upper bounds on the number of hidden neurons in feedforward networks with arbitrary bounded nonlinear activation functions, IEEE Trans. Neural Networks 9 (1) (1998): 224–229.

[76] G. B. Huang, Q. Y. Zhu, and C. K. Siew. "Extreme learning machine: a new learning scheme of feedforward neural networks." In *Neural Networks, 2004. Proceedings. 2004 IEEE International Joint Conference on*, vol. 2. IEEE, (2004): 985-990.

[77] G. B. Huang.“Learning capability and storage capacity of two-hidden- layer feedforward networks.” , IEEE Trans. Neural Networks 14 (2) (2003): 274–281.

[78] S. Tamura, M. Tateishi, Capabilities of a four-layered feedforward neural network: four layers versus three, IEEE Trans. Neural Networks 8 (2) (1997) 251–255.

[79] J.M. Ortega, *Matrix Theory*, Plenum Press, New York, London, 1987.

[80] ORL Database: AT&T Laboratories, Cambridge, UK, ORL Databaseof Faces (now AT&T The Database of Faces). Available online: http://www.cl.cam.ac.uk/Research/DTG/attarchive/pub/data/att_faces.zip

Vita Auctoris

NAME: Abbas Elazhari

PLACE OF BIRTH: Libya

YEAR OF BIRTH: 1985

EDUCATION:

2011-2013

Masters of Applied Science

Electrical and Computer Engineering

University of Windsor, Windsor, Ontario, Canada

2002-2007

Bachelors of Engineering

Electronics and Computer Engineering

Aljabal Algharbi University, Gharian\Libya