Improvement of Real Time Human Face Recognition

تحسين تميز الوجوه خلال الوقت الحقيقي

By

Muna Suliman
200420114

Supervised by
Dr. Muzhir Shaban

A Thesis submitted in partial fulfillment of the requirements for the degree of Master of Science in Computer Science

Department of Computer Science
Graduate College of Computing Studies

Amman Arab University for Graduate Studies
2009

Authorization

I, Muna Suliman the undersigned, herby authorize Amman Arab University for Graduate Studies to provide libraries, organizations, institution, and individuals with copies of my thesis when requested.

Name: Muna Suliman

Signature……………………………..

Date: 25 / 11 / 2009

This Dissertation titled " Improvement of Real Time Human Face Recognition "
Has been defended and Approved on 7/11/2009

Examining Committee	Title	Signature
Dr.Msbah Guma'a Aqel	Chair	M. Aqel
Prof.Dr. Alaa Al-Hamamy	Member	Alahamami
Dr.Muzhir Shaban	Member and supervisor	

Dedications

To Dr Muzhir Shaban, Dr Ala'a Hamamneh for their support and encouragement. To my husband, my parents, my family for their love and support. And to everyone who supported me during my study.

Muna Suliman

Acknowledgments

I would like to thank all those who helped me immeasurably during the time I spent working on this thesis. First I would like to thank Dr Muzhir Shaban, my thesis supervisor. Special thanks to Dr Ala'a Hamamneh the dean of the colleague. Finally I would like to thank my husband my parents, and my family for their invaluable love and support.

My greatest debt also goes to the Computer Department staff and my friends at the University.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

IMPROVEMENT OF REAL TIME HUMAN FACE RECOGNITION

Abstract

Face recognition is becoming an increasingly important area with advances in technology and computer vision. There are many applications that can benefit from face recognition including surveillance systems, human tracking, and police screening. Researchers have proposed many algorithms for face recognition; these algorithms differ in recognition rate.

This thesis, presents a study of the recognition rate of face recognition algorithms. This work concentrates on the following algorithms: Principal Component Analysis, Two Dimensional Principal Component Analysis in Column Direction, Two Dimensional Principal Component Analysis in Row Direction and Two Dimensional Two Directional Principal Component Analysis. All these algorithms are implemented into two stages: training stage and recognition stage. We present a comparison between the four algorithms with respect to recognition rate.

An important optimization in the training and recognition stages is presented in this work. In case of optimization, a proposal algorithm is implemented based on Discrete Wavelet Transform, the proposed algorithm minimize the images size. Then these minimized images are entered to a Principal Component Analysis algorithm. A complexity reduction is achieved by optimizing the number of comparison operations needed. This optimization does not increase the recognition rate only, but it is also reduces the execution time, by decreasing the number of comparison operations needed.

This work shows the effect of the optimizations on the recognition rate via applying face recognition algorithms. These algorithms are implemented using MATLAB programming language. The testing occurs then using Olivetti Research Laboratory database. The obtained results show that an improvement of 4 to 5 % is achieved by introducing Discrete Wavelet Transform.

Arabic Summary

الملخص

مع تطور التكنولوجيا والرؤيا الحاسوبية، أصبح التعرف على الوجوه عن طريق الحاسوب مجالا مهما. يوجد العديد من التطبيقات على التعرف على الوجوه من ضمنها أنظمة المراقبة، تتبع الأشخاص و شاشات الشرطة. أوجد الباحثون العديد من الخوارزميات للتعرف على الوجوه، هذه الخوارزميات تختلف في الأداء معدل التعرف على الصور.

في هذه الرسالة نقدم دراسة عن أداء و معدل التعرف على الوجوه. سنركز على الخوارزميات التالية : تحليل المكونات الأساسية، تحليل المكونات الأساسية بالبعد الثنائي بالاتجاه العامودي، تحليل المكونات الأساسية بالبعد الثنائي بالاتجاه الخطي، تحليل المكونات الأساسية بالبعد الثنائي بالاتجاهين. جميع هذه الخوارزميات تقسم إلى مرحلتين: مرحلة التدريب ومرحلة التعرف. نقدم مقارنة بين الخوارزميات الأربعة بالنسبة إلى معدل التعرف على الوجوه.

بالاعتماد على الدراسة التي تم عرضها، قمنا بتقديم تحسين في المرحلتين الأساسيتين للتعرف على الوجوه. في هذا التحسين استخدمنا فرضية تحويل الموجة المتقطعة للتقليل من حجم الصور قبل تطبيق خوارزميات تحليل المكونات الأساسية. باستخدام هذه الفرضية تم تقليل التعقيدات في الخوارزميات بالاضافه إلى زيادة في معدل التعرف على الوجوه، وتقليل الوقت التنفيذي للخوارزميات.

أظهرنا تأثير التحسين على معدل التعرف على الوجوه بعد استخدام الفرضية، وذلك عن طريق تطبيق الخوارزميات على لغة برمجية الماتلاب، زمن ثم فحص هذه الخوارزميات باستخدام قاعدة بيانات(ORL). حيث أظهرت النتائج تحسينا مقداره ما بين ٤ إلى ٥ % عندما استخدمنا تحويل الموجة المتقطعة قبل تطبيق الخوارزميات.

Chapter ١
Introduction

1.1. Overview

Face recognition has many important practical applications, like surveillance and access control, military and medical. It is concerned with the problem of correctly identifying face images and assigning them to persons in a database. The new technique is compared with well-established face recognition algorithms, namely Eigenfaces [14].

Face recognition is an essential part of many applications, including, surveillance, access control and personal identification, and forensic and law enforcement applications. To implement a face recognition algorithm, firstly a database of face images for a group of people must be prepared. Then, an unknown face image is given, then the question: "which person in our database does this image belong to?" is answered Many algorithms and techniques have been proposed for solving such a problem. The results are compared with the well established algorithms of Eigenfaces [2] the results showed improved recognition accuracy over this method.

The objectives of these algorithms are to compute and explain the recognition rate. This will allow us to compare algorithms that perform same task and choose the best of them according to available resources.

The rest of this chapter is organized as follows: Section2 presents the motivation for the proposed work. Section 3 presents problem statement in detail. Section 4 presents the contribution of the proposed work. Finally, Section 5 presents the thesis organization.

1.2 Motivations

In order to have a functionality of face-recognition system, algorithms must be able to detect the face; that is a given photograph, it must be able to find the corresponding face in the stored database. Technically, this is easier and more reliable than identifying a particular person. The growth of the Internet began appearing in face detection systems field a few years ago.

In the last few years, face recognition has become more important and has been employed in different applications such as surveillance, human tracking, and police screening. To facilitate the use of face recognition application, the recognition rate must be linked to the performance issues [15].

1.3 Statement of the problem

In face recognition systems there are two major areas; the first one is called face detection and the second one is called face recognition; in face detection the algorithms focus on finding the face in an image, but in face recognition the algorithms focus on recognizing the face images, which are stored in a database.

Recently, Face recognition has become a more important area; as a result many algorithms have been developed to implement face recognition. Researchers have spent more effort on enhancing the performance of these algorithms in terms of recognition rate.

An algorithm is proposed to reduce the execution time of the face recognition algorithms and improve the face recognition rate. The execution rate of an algorithm depends mainly on the number of operations needed to recognize the indicated face image.

1.4 Contribution

This thesis deals with four algorithms of face recognition: Principle Component Analysis, Two Dimensional Principle Component Analysis in Column direction, Two Dimensional Principle Component Analysis in Row Direction and Two Dimensional Two Directional Principal Analysis.

This thesis has the following objectives:

- The first objective deals with the discussion and comparison of the recognition rates between the four indicated algorithms.
- The second objective deals with evaluating the recognition rates of the optimized algorithms and to show the effect of the optimization on the recognition rate.

1.5 Thesis Organization

The organization of the thesis is constructed as follows:

Chapter two presents a general review, definition and types of face recognition, and typical applications of face recognition, with focus on the following face recognition algorithms: Principal Component Analysis, Two Dimensional Principal Component Analysis in Column Direction, Two Dimensional Principal Component Analysis in Row Direction and Two Dimensional Two Directional Principal Component Analysis

Chapter Three presents a comparison between the four algorithms in terms of recognition rate before applying the optimization technique. Then another comparison is presented after the optimization, also a study of the effect of these optimizations on the recognition rate is implemented lastly.

Finally Chapter four presents a summary and conclusion of the thesis, and then some directions for future work are mentioned.

Chapter ٢
Literature Review

2.1. Introduction

Face recognition is becoming an increasingly important area with advances in technology and computer vision. There are many applications that can benefit from face recognition including surveillance systems, human tracking, and police screening. Researchers have proposed many algorithms for face recognition.

Many face recognition algorithms have been reported in the literature [1, 2]. Researchers have focused on improving face recognition algorithm by concentrating on ways to improve the recognition rate. This has resulted in many algorithms that can achieve the recognition task, but recognition time might be different. The execution time of an algorithm depends on major aspects like the number of comparison operations.

2.2 Related Work

Many previous researchers have developed algorithms for face recognition. This section presents some of these studies which are related to the objective of this thesis:

Comparative Study between different Eigenspace-based Approaches for Face Recognition [2]**:**

Different eigenspace-based approaches have been proposed for the recognition of faces. They differ mostly in the kind of projection method that been used and in the similarity matching criterion employed. This study considers theoretical aspects as well as simulations performed using a face database with a few number of classes [2].

One of the most successful approaches used in face recognition are the eigenspace-based methods, which are mostly derived from the Eigenface-algorithm. These methods project the input faces onto a dimensional reduced space where the recognition is carried out, performing a holistic analysis of the faces.

Different eigenspace-based approaches have been proposed. They differ mostly in the kind of projection/decomposition method being used and in the similarity matching criterion employed. The aim of this paper is to present a comparative study between some of these different approaches. The comparison considers the use of three different projection methods (Principal Component Analysis PCA, Fisher Linear Discriminate FLD and Evolutionary Pursuit EP) [30].

Face representation using Independent Component Analysis [34, 31]**:**

In this approach, the problem of face recognition using independent component analysis (ICA) is presented. Two issues on face representation using ICA is presented in this research [34]:

First, as the Independent Components (ICs) are independent but not orthogonal, images outside a training set cannot be projected into these basis functions directly. This paper proposes a least-squares solution method using Householder Transformation to find a new representation [31].

Second, it demonstrates that not all ICs are useful for recognition. Along this direction, in this approach a new design of an IC selection algorithm is presented to find a subset of ICs for recognition. Three public available databases, namely, MIT AI Laboratory, Yale University and Olivetti Research Laboratory are selected to evaluate the performance and the results are encouraging [31].

Face Recognition Using Multi-feature and Radial [32]**:**

In this paper, a face recognition algorithm using multi feature and Radial basis Function Network (RBFN) is proposed. The algorithm consists of three steps. In the first step, a coarse classification is performed using Fourier frequency spectrum feature, and only the first k gallery images with minimum Euclidean distance to the probe image are retained.

In the second step, the Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA) features of frequency spectrum are extracted, which will be taken as the input of the RBFN in the third step [32].

In the last step, the classification is carried out by using RBFN. The proposed approach has been tested on ORL face database and Shimon database. The experimental results have demonstrated that the performance of this algorithm is much superior to the other algorithms on the same database [32].

Analysis and Comparison of Eigenspace-based Face Recognition Approaches [33]**:**

Different eigenspace-based approaches have been proposed for the recognition of faces. They differ mostly in the kind of projection method that have been used and in the similarity matching criterion employed. The aim of this paper is to present a comparative study between some of these different approaches [33].

This study considers theoretical aspects as well as simulations performed using a face database with a few number of classes (Yale) and a face database with a large number of classes (FERET) [33].

Independent Component Analysis Algorithms and Applications [34]**:**

A fundamental problem in neural network research, as well as in many other disciplines, is finding a suitable representation of multivariate data, i.e. random vectors. For reasons of computational and conceptual simplicity, the representation is often sought as a linear transformation of the original data. In other words, each component of the representation is a linear combination of the original variables.

Well-known linear transformation methods include principal component analysis, factor analysis, and projection pursuit. Independent component analysis (ICA) is a recently developed method in which the goal is to find a linear representation of no Gaussian data so that the components are statistically independent, or as independent as possible.

Such a representation seems to capture the essential structure of the data in many applications, including feature extraction and signal separation. This paper presents the basic theory and applications of ICA [34].

Neural-Network-Based Gender Classification Using Genetic Search for Eigen-Feature Selection [35]**:**

This paper considers the problem of gender classification from frontal facial images using feature selection and neural networks. Feature selection is an important issue in gender classification which can be done by removing features that do not encode important gender information from the image representation of faces, the error rate can be reduced significantly [35].

Automatic feature subset selection distinguishes the proposed method from previous gender classification approaches. First, Principal Component Analysis (PCA) is used to represent each image as a feature vector (i.e., eigen-features) in a low-dimensional space, spanned by the eigenvectors of the covariance matrix of the training images (i.e., coefficients of the linear expansion).

A Genetic Algorithm (GA) is then used to select a subset of features from the low-dimensional representation by removing certain eigenvectors that do not seem to encode important information about gender (e.g., eigenvectors encoding information about classes). Finally, a Neural Network (NN) is trained to perform gender classification using the selected eigen-feature subset [35].

Experimental results demonstrate a significant improvement in error rate reduction. Using a subset of eigen-features containing only 18% of the features in the complete set, the average NN classification error goes down to 11.3% from an average error rate of 17.7% [35].

2.3.Summary

Recently human face recognition has been an active topic in the field of object recognition. Researchers have developed many algorithms, these algorithms may give the same result but they may differ in the time needed to complete the execution, or they may differ in the memory space required by each algorithm.

Chapter ٣
Face recognition

3.1 Introduction

In general, face recognition is a process where the input is an image of an unknown person and the output is a report of identification of that person from a stored database of known persons. This process involves feature extraction from the input face image and comparison of the extracted features with a database of stored features. In this chapter a review of face recognition is presented [3].

This chapter is organized as follows: Section 2 presents the applications of face recognition, section 3 discusses the general structure for face recognition, and section 4 presents the most successful face recognition algorithms. Finally section 5 presents the summary of this Chapter.

3.2 Face Recognition Applications

In the last few years the need for security systems in many applications is becoming an important area of research. Face recognition system is one of the most security systems used nowadays, this technology gives high performance in recognizing people, and also this technology does not require any interaction with the person being recognized [15].

Face recognition is becoming more popular with the fast growth in communication technology. It has many applications in access control and information security; for example building access, internet access, medical records in hospitals and bank database security. And one of the most important applications for face recognition is the surveillance system; it can be used for crowd surveillance and bank surveillance. Also face recognition is used widely in cards applications; for example credit cards, driver's license cards, passports and in national ID cards [15].

3.3 Structure of General Face Recognition System

Face recognition algorithms are implemented in many forms to perform high efficiency of recognition. Face Recognition algorithms are affected by many factors such as recognition rate, execution time and recognition accuracy. In general, face recognition algorithms have two major stages; the training stage and the recognition stage.

The first stage is the training stage where a database of images for known people is used, at least one image per known person. In this stage the features for each known face image are extracted and stored in a database.

In the second stage which is the recognition stage, each new unknown face image is analyzed to obtain its features, and then a comparison between its features and the stored features from the training stage is performed to identify the unknown face image. Many algorithms for face recognition have both a training stage and a recognition stage.

3.3.1 Training Stage

There are three main steps in the training stage: first calculating a projection matrix from the trained images, then extracting features from the images and finally storing these features to be used in the recognition stage.

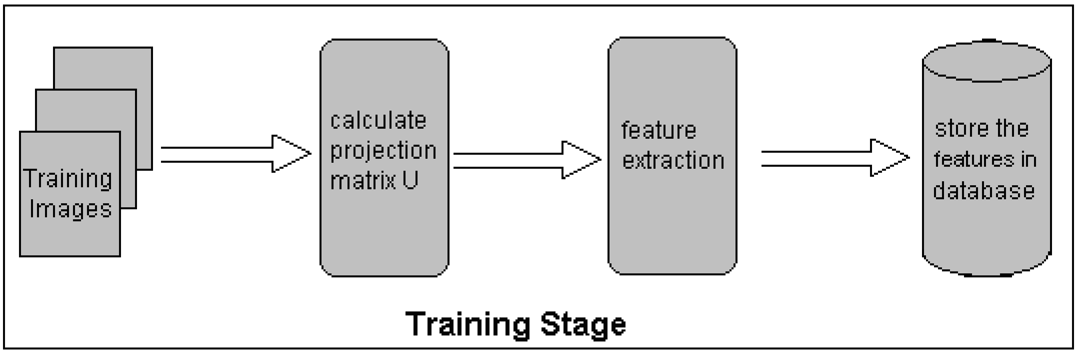


Figure 3.1: Training Stage of a general face recognition block diagram.

Firstly, the projection matrix must be calculated, in which face recognition algorithm uses the training images to calculate the projection matrix. For example in the Principal Component Analysis (PCA) the projection matrix consists of eigenvectors of the covariance matrix, where the covariance matrix is derived from the trained images [2, 16].

The second step deals with the extraction of the features from images. In PCA each image is multiplied by the projection matrix to formulate the eigenfaces. The eigenfaces represent the principal component of the images, or in other words it represents the features of the image. Finally the last step in the training stage is to store the features that are extracted from the trained images [3, 16].

3.3.2 Recognition Stage

This stage has two main steps: extracting the features for the unknown image and comparing it with the stored features. Figure 3.2 shows the steps of the recognition stage to recognize an unknown face image.

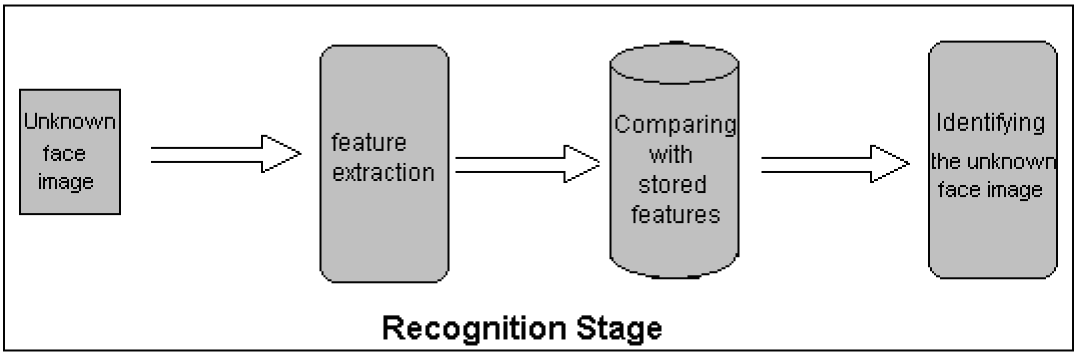


Figure 3.2: Recognition stage of a general face recognition block diagram.

The calculated projection matrix in the training stage is used to extract the features of the unknown image. So in order to extract the features from the unknown image applying PCA algorithm, multiply the unknown elements of images by the projection matrix which is derived from the training stage.

When the features of the unknown image are extracted, a comparison process is performed between these extracted features and the stored features from the training stage. There are many methods to do such a comparison, such as the Euclidean distance measure. The distance between the features of the unknown image and the stored features is computed, and then the minimum distance corresponding to the closest face features is selected as the matched face [16].

3.4 Face Recognition Algorithms

The most successful face recognition algorithms are based on eigenface technique. In this technique the features of the images are represented using eigenface images. The section presents an overview of the most popular and successful face recognition algorithms based on eigenfaces.

3.4.1 Face Recognition Using Principal Components Analysis (PCA)

Similarly to other algorithms of face recognition, PCA has two stages: training stage and recognition rate stage [16].

Training Stage

In this stage the features of the trained images are extracted and stored in a database, as illustrated in Figure 3.3 where **A** represents the matrix which contains all images, **C** represents the covariance matrix, and **V** represents the eigenvector matrix.

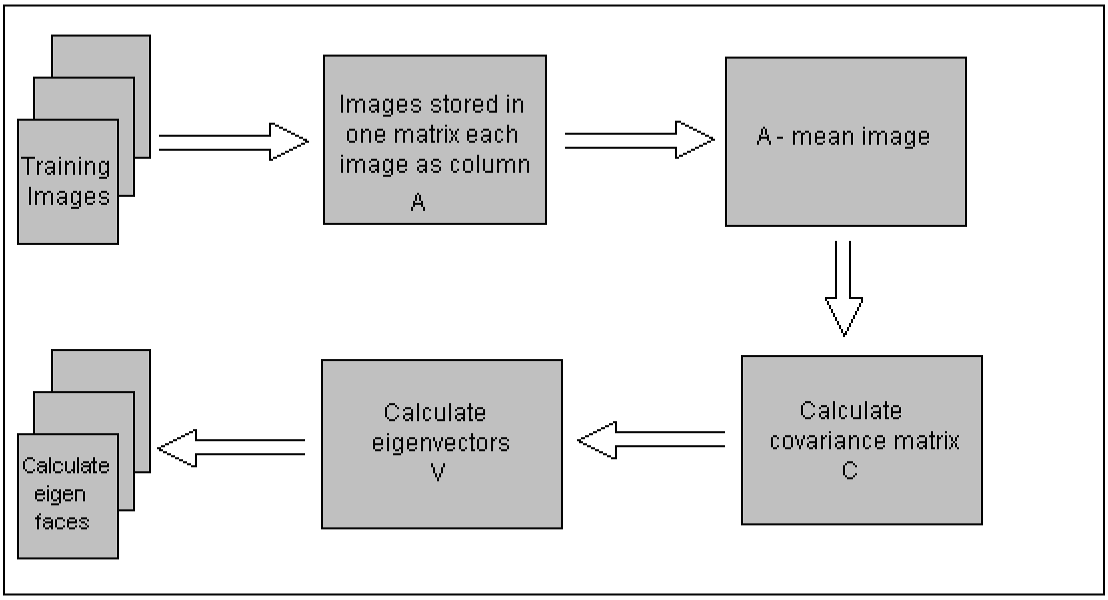


Figure 3.3: Training Stage of PCA.

The output of the training stage is a set of images called eigenfaces; these images represent the features of the trained images. To calculate the eigenfaces, first the covariance matrix must be calculated, and then the eigenvectors of that covariance matrix are calculated. After that the eigenfaces are calculated by multiplying the original trained images by the eigenvector matrix. The main steps used to calculate the training images are shown below [15,16]:

- Step One: Store Images as Vectors

Images can be represented as a matrix, but the image can be restored in a vector form instead of a matrix form, by storing the rows of pixels one after another. For example an image that is (5 × 5) pixels can be rearranged into one vector of size (1 × 25) or (25 × 1), see Figure 3.4.

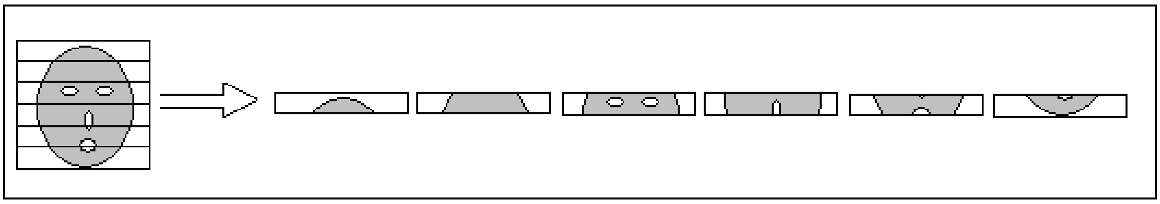


Figure 3.4: Face image viewed as a vector.

- Step Two: Calculate the difference of means

Suppose that you have *P* face images each of size (*m* × *n*) in the training set $\{X_1, X_2, X_3, \ldots\ldots, X_P\}$. Firstly, calculate the mean face image Ψ as below:

$$\Psi = 1/P \sum_{n=1}^{P} X_n \qquad \text{where } n = 1,2,\ldots..,p \qquad (3.1)$$

Then calculate the difference between each of the training images and the mean face as below:

$$a_i = X_i - \Psi \qquad \text{where } i = 1 \text{ to } P \qquad (3.2)$$

The difference images are stored into one matrix of dimension ($m \times n \times P$), where each image is stored as a vector in column wise in this matrix $\mathbf{A} = [a_1 a_2 a_3 \ldots\ldots a_P]$. Where **A** represents the whole matrix, and $a_1 a_2 a_3 \ldots\ldots a_P$ represents the vectors inside **A**.

- Step Three: Calculate the Covariance Matrix *C*

The covariance matrix C_{ov} is calculated from the training set as shown in following equations:

$$C_{ov} = AA^T \qquad (3.3)$$

where A^T is the transpose of A.

- Step Four: Calculate the Eigenvector Matrix *V*

To compute the eigenfaces, the eigenvectors of the covariance matrix are needed. The eigenvectors can be calculated using simple code in matlab. The eigenvectors can be calculated as shown in the below equation:

$$C(AV) = \lambda(AV) \qquad (3.4)$$

Where **C** is the covariance matrix, A is the matrix which contains the images, **V** represents the eigenvectors matrix, and **λ** represents the eigenvalues.

- Step Five: Feature Extraction

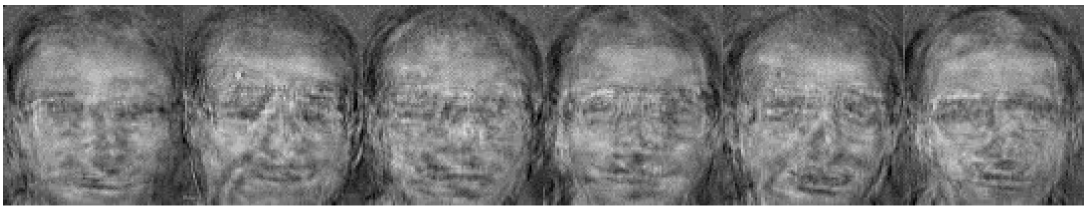
The features of the images are represented as eigenfaces $\{U_1, U_2, U_3, \ldots\ldots, U_P\}$, these eigenfaces are constructed using the eigenvectors of the *C* matrix as shown in the following Equation: where U represents the eigenfaces

, V is the eigenvectors matrix and $a_1 a_2 a_3 ....a_P$ are the training images. Figure 3.5 represents the images stored in **A** matrix, and Figure 3.6 represents the eigenfaces stored in **U** matrix [15] [16].

$$U_i = \sum_{k=1}^{P} \mathbf{V}_{ik}\, \mathbf{a}_k \qquad i = 1,2,...., P \tag{3.5}$$

Figure 3.5: Samples of ORL images.

This thesis uses the (ORL Olivetti Research Laboratory Database of Faces), which contains a set of face images taken between April 1992 and April 1994 at the lab. The database was used in the context of a face recognition project. This database of images is widely used in the face recognition research area [6].



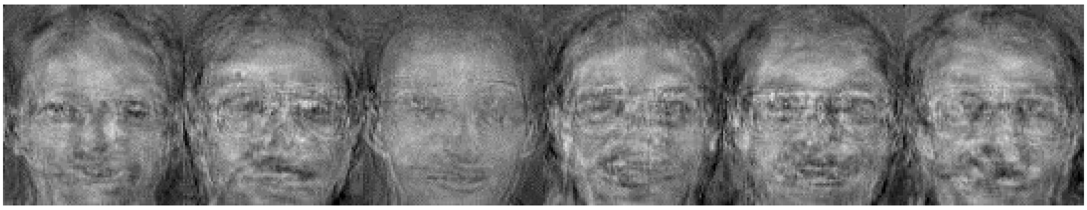


Figure 3.6: Eigenfaces for the ORL images.

- Step Six: Calculate the Weights

For each training image calculate the vector of weights $\Omega_i^{\mathrm{T}} = [w_1, w_2, w_3, ..., w_{\mathrm{P}}]$ where *i = 1,2,….. P* and w_p is the weight of the P^{th} eigenface.

$$w_k = u_k^T a_k \qquad k = 1,...,P \tag{3.6}$$

For each different person who has more than one image in the training set, calculate the average weight Ω. After that you will have $\Omega_1 \ldots \Omega_k$ corresponding to the maximum number of different known persons [15, 16].

Recognition Stage

The input of the recognition stage is a new image, and we want to know to whom this image belongs. This stage involves two steps: feature extraction for the new image and comparison of these features with the stored features.

- Step One: Feature Extraction

To recognize an unknown face image X_{new}, the first step is to calculate its weights feature vector $\Omega_{\mathrm{new}}^{\mathrm{T}} = [w_1, w_2, w_3, ..., w_{\mathrm{P}}]$. This can be done using the eigenfaces constructed in the training stage according to the following equation.

$$w_k = u_k^T (X_{new} - \Psi) \qquad k=1,...,P \tag{3.7}$$

-Step Two: Comparison with Stored Features

To find which of the stored faces is the closest one to the input image, the Euclidean distance measure is used to compare Ω_{new} of the new unknown image with each Ω_k of the stored faces, see Equation 3.8. The minimum distance corresponds to the nearest face.

$$E_k = \| \ \Omega_{\mathrm{new}} - \Omega_K \ \| \qquad (3.8)$$

3.4.2 Face Recognition Using Two-Dimensional Principal Component Analysis in Row Direction (2DPCA_R)

The standard one dimensional Principal Component Analysis has been extended to two dimensional Principal Component Analysis. This algorithm differs from the standard one in the way it deals with the training images; while the standard PCA deals with images as one dimensional vector the 2DPCA deals with images as two dimensional matrices [17].

The size of the covariance matrix of this algorithm is less than that of the PCA covariance matrix, because the size of the image covariance matrix is equal to the width of images which is the number of pixels per row [18].

Training Stage

- Step One: Calculate the difference of means

Suppose that you have *P* face images of size (*m* × *n*) in the training set $\{X_1, X_2, X_3, \ldots\ldots, X_P\}$. As in PCA it is needed to calculate the mean face image Ψ.

$$\Psi = 1/P\sum_{n=1}^{P} \mathrm{X}_{\mathrm{n}} \qquad (3.9)$$

Then, the difference between each of the training images and the mean face is calculated as below:

$$F_i = X_i - \Psi \qquad i = 1 \text{ to } P \qquad (3.10)$$

- Step Two: Calculate the Column Covariance Matrix C_{ov_col}

The column covariance matrix $\boldsymbol{C_{ov\_col}}$ is calculated as shown below:

$$C_{ov\_col} = 1/P\sum_{i=1}^{P}(X_i - \Psi)^T(X_i - \Psi) \qquad (3.11)$$

- Step Three: Calculate the Projection Matrix *V*

The best projection matrix *V* is composed of the eigenvectors for the column covariance matrix.

- Step four: Feature Extraction

The projection matrix *V* is used to extract the features from the trained images, as shown in Equation 3.12. For each trained image F_i the feature matrix Y_i is calculated. For each different person who has more than one image in the training set the average *Y* is calculated so that there will be *K* different features that correspond to *K* different persons [18].

$$Y_i = F_i V \qquad (3.12)$$

Recognition Stage

As in PCA the recognition stage involves two steps: feature extraction and the comparison with stored features.

- Step One: Feature Extraction

To recognize an unknown face image X_{new}, the first step is to calculate its feature matrix using the projection matrix V which has been calculated in the training stage [17].

$$Y_{new} = (X_{new} - \Psi)V \tag{3.13}$$

-Step Two: Comparison with Stored Features

To find the closest face to the new unknown image the Euclidean distance measure is used to compare Y_{new} with each Y of the predefined features.

$$E_i = \| Y_{new} - Y_i \| \qquad i = 1,2 ,\ldots,K \tag{3.14}$$

3.4.3 Face Recognition Using Two-Dimensional Principal Component Analysis in Column Direction (2DPCA_C)

The two dimensional principal component analysis in column direction 2DPCA_C is similar to the 2DPCA_R. It extracts the features from training images using a projection matrix Z. This projection matrix consists of the eigenvectors for the row covariance matrix C_{ov_row}. The size of the covariance matrix of this algorithm is equal to the number of rows per image [17, 18].

Training Stage

The main difference between the 2DPCA_C and 2DPCA_R is the projection matrix. The best projection matrix for 2DPCA_C is the eigenvectors matrix derived from the row covariance matrix.

- Step One: Calculate the difference of means

$$\Psi = 1/P\sum_{n=1}^{P} X_n \quad (3.15)$$

$$F_i = X_i - \Psi \qquad i = 1 \text{ to } P \quad (3.16)$$

- Step Two: Calculate the Row Covariance Matrix C_r

The row covariance matrix C_{ov_row} is calculated from the trained images as shown in the following:

$$C_{ov\_row} = 1/P\sum_{i=1}^{P}(X_i - \Psi)\ (X_i - \Psi)^T \quad (3.17)$$

- Step Three: Calculate the Projection Matrix *Z*

The projection matrix is composed of the eigenvectors for row covariance matrix, the size of the covariance matrix is ($m \times m$).

- Step Four: Feature Extraction

In this algorithm the projection matrix *Z* is used to extract the features from the training images, as shown in Equation 3.18. Calculate the feature matrix Fi For each trained image , for each different person who has more than one image in the training set calculate the average *B* so that there will be *K* different feature images that correspond to *K* different persons [16, 17].

$$B_i = Z^{T} F_i \tag{3.18}$$

Recognition Stage

Similarly to algorithms the 2DPCA_R in the recognition stage involves two steps: feature extraction and the comparison with the stored features.

- Step One: Feature Extraction

To recognize an unknown face image X_{new}, the first step is to calculate its feature matrix using the projection matrix Z.

$$B_{new} = Z^{T} (X_{new} - \Psi) \tag{3.19}$$

-Step Two: Comparison with Stored Features

To find the closest face feature image to the new unknown image the Euclidean distance measure is used to compare B_{new} with each B of the predefined features.

$$E_i = \| \; B_{new} - B_i \; \| \qquad i = 1,2 ,\ldots..,K \tag{3.20}$$

Where K is the number of different feature images derived from the training images, correspond to K different persons [16, 18].

3.4.4 Face Recognition Using Two-Dimensional Two-Directional Principal Component Analysis $(2D)^2$PCA

The two dimensional two directional principal component analysis $(2D)^2$PCA was developed by considering the row and column directions at the same time. 2DPCA_R works only in the row direction of images and uses a projection matrix V that reflects the information between rows of images. While 2DPCA_C works only in the column direction of images and uses a projection matrix Z that reflects information between columns of images. This stage represents the main stages in $(2D)^2$PCA [15, 18].

Training Stage

In this stage the $(2D)^2PCA$ extracts the features from the training images using two projection matrices V and Z as shown in figure 3.7. where Cr is the row covariance matrix, Z is the eigenvectors for the row covariance matrix, Cc is the column covariance matrix, and V is the eigenvectors for the column covariance matrix [18].

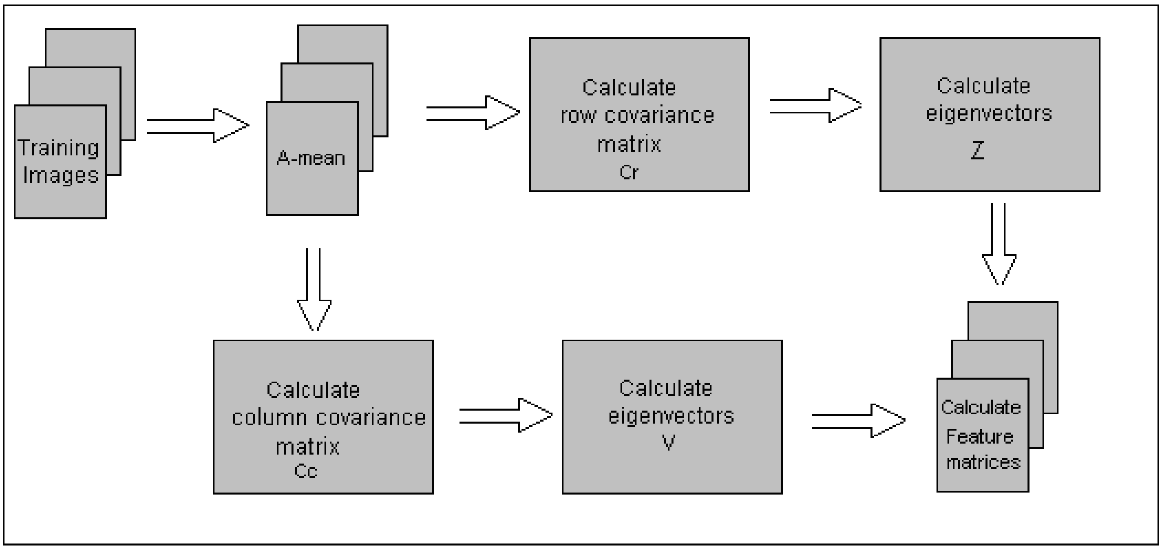


Figure 3.7: Training stage in $(2D)^2PCA$ algorithm.

- Step One: Calculate the difference of means

Suppose that you have *P* face images of size (*m* × *n*) in the training set $\{X_1, X_2, X_3, \ldots\ldots, X_P\}$. As in 2DPCA_R and 2DPCA_C , the mean face image is calculated from the trained face images. Then subtract that mean image from all training images [17, 18].

- Step Two: Calculate Row and Column Covariance Matrices

The row covariance matrix C_{ov_row} is calculated from the trained images as shown in Equation 3.21. The column covariance matrix C_{ov_col} is calculated from the training images as shown in Equation 3.22.

$$C_{ov\_row} = 1/P\sum_{i=1}^{P}(\mathrm{X_i}-\Psi)\ (\mathrm{X_i}-\Psi)^{\mathrm{T}} \quad (3.21)$$

$$C_{ov\_col} = 1/P\sum_{i=1}^{P}(\mathrm{X_i}-\Psi)^{\mathrm{T}}(\mathrm{X_i}-\Psi) \quad (3.22)$$

- Step Three: Calculate the Projection Matrices *Z* and *V*

The projection matrices are composed of the eigenvectors for row and the column covariance matrices, the size of the *Z* projection matrix is ($m \times m$), and the size of the *V* projection matrix is ($n \times n$).

- Step Four: Feature Extraction

The $(2D)^2$PCA use the projection matrices *V* and *Z* to extract features from the trained images as shown in Equation 3.23.

$$U_i = Z^T \mathrm{F_i} V \quad (3.23)$$

Recognition Stage

The $(2D)^2$PCA involves two steps: feature extraction and the comparison with the stored features.

- Step One: Feature Extraction

To recognize an unknown face image X_{new}, the first step is to calculate its feature matrix using the projection matrix *Z*.

$$U_{new} = Z^T(\mathrm{X_{new}}-\Psi)V \quad (3.24)$$

-Step Two: Comparison with Stored Features

The Euclidean distance between U_{new} and each *U* of the predefined classes is computed:

$$E_i = \| U_{new} - U_i \| \qquad i = 1,2 ,\ldots,K \qquad (3.25)$$

Where K is the number of different feature images derived in the trained images, correspond to K different persons. $(2D)^2$PCA achieves the same or even higher recognition accuracy than 2DPCA [17, 18].

3.5 The Proposed Solution

The main problem of the face recognition algorithms is that it requires huge number of comparisons between the unknown image and the stored images, and this requires time.

The proposed solution for this problem is to use the main components of the images to be compared, this will reduce the amount of comparisons needed, and this will decrease the execution time. The proposed solution is to apply The Discrete Wavelet Transforms (DWT) on the stored images before applying the PCA algorithms, and applying the DWT on each incoming unknown image. Using the DWT will decompose the images into four parts. One of them contains the most important components of the original image; like the lines and the shapes and the location of the mouth and ears.

After applying the DWT the comparison process will take place during the recognition stage using the small images. When the algorithm uses the small images it will reduce the number of comparison operations needed, as a result the execution time will decrease.

Discrete Wavelet Transforms DWT

The Discrete Wavelet Transforms (DWT) technique minimizes the images size. DWT is used to transform the original image into four sub images each one containing some information of the original image. The sub image of our interest is the LL sub image which is identical to the original image [20, 21].

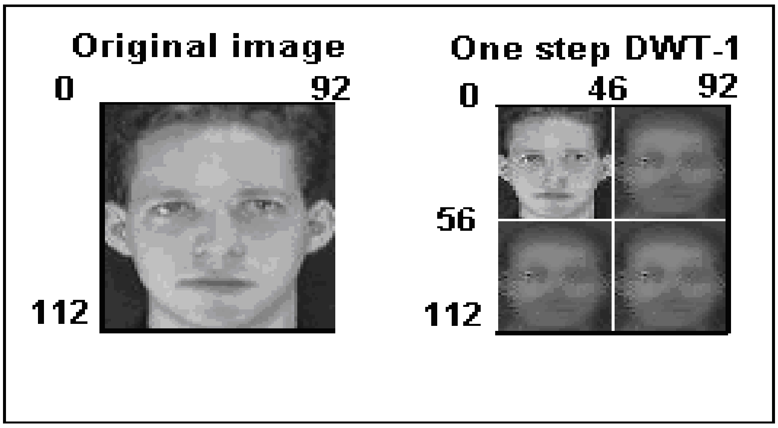


Figure 3.8: Decomposing an image using DWT (one step).

Second level DWT is implemented to achieve more reduction in image size, i.e. more reduction in comparison operations required. In addition this improvement in recognition rate causes some reduction in image resolution as shown in Figure 3.9.

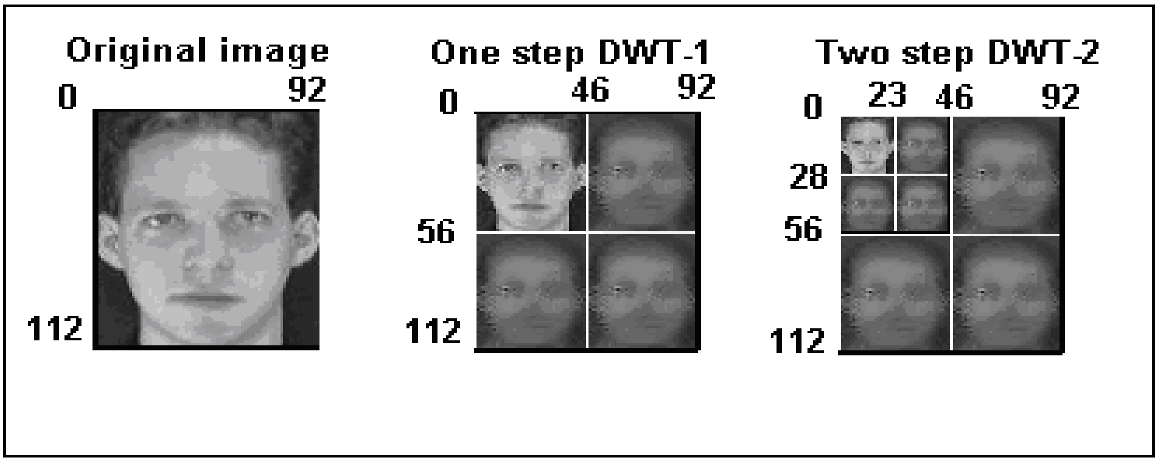


Figure 3.9: Decomposing an image using DWT (two steps).

An increasing number of Digital Signal Processing (DSP) applications uses the discrete wavelet transform (DWT), so the need for highly efficient DWT software is apparent. [36]

The transformation of a signal does not change the information content present in the signal; the transformation is just another form of representing the signal. The Wavelet Transform provides a time-frequency representation of the signal. It was developed from the Short Time Fourier Transform (STFT), which can also be used to analyze non-stationary signals. The Wavelet Transform uses multi-resolution technique while STFT gives a constant resolution at all frequencies

A wave is a periodic function of time or space, while the wavelets are waves which have their energy concentrated in time or space. The Fourier Transform uses waves to analyze signals; the Wavelet Transform uses wavelets to analyze signals.

To analyze the signal using the wavelet analysis; the signal is multiplied with a wavelet function, this process is very similar to the STFT analysis where the signal to be analyzed is multiplied with a window function, and then the transform is computed for each segment generated. In addition the Wavelet Transform function changes with each spectral component.

The Continuous Wavelet Transform (CWT) is another type of wavelets where the wavelet functions used in the transformation are derived from the mother wavelet through translation or shifting, and scaling or compression.

All the basis functions are designed based on the mother wavelet. The translation depends on a parameter relating to the location of the wavelet function as it is shifted through the signal. This parameter corresponds to the time information in the Wavelet Transform.

Another important parameter is the scaling parameter and it is defined as |1/frequency| and corresponds to frequency information. Scaling either expands or compresses a signal. Large scales means low frequencies; expand the signal and provide detailed information hidden in the signal, while small scales means high frequencies; compress the signal and provide global information about the signal.

In addition the Wavelet Transform can perform the convolution operation of the signal. This analysis becomes very useful in most practical applications; high frequencies which have low scales do not stay for a long duration, while low frequencies which have high scales usually last for the entire duration of the signal.

3.6 Summary

A general face recognition system has two major stages: the trained stage and the recognition stage. In the training stage the features of the training images are extracted using a projection matrix. The features are stored in order to be used then in the recognition stage. In the recognition stage, the features of the unknown image are extracted and compared to the stored features. The least difference corresponds to the nearest face image.

An overview of four algorithms is presented: Principal Component Analysis, Two Dimensional Principal Component Analysis in Column Direction, Two Dimensional Principal Component Analysis in Row Direction and the Two Dimensional Two Directional Principal Component Analysis.

Chapter 4
Proposed Work

4.1 Introduction

This chapter presents the proposed work and how to implement the PCA algorithms discussed in chapter two. Also it presents the proposed optimization which will be applied to the algorithms. The results of, the optimized algorithm are presented in chapter 5.

4.2 The Proposed Optimization Technique

In the recognition stage when a new face image passes to the system, the face recognition algorithm extracts the features of this new image, then it compares the extracted features with the features of the database which has been created in the training stage. This takes a lot of execution time due to the big number of operations.

The proposed technique based on discreet wavelet transforms (DWT) algorithm is used to minimize images size, then PCA algorithm is applied for face recognition. DWT is used to transform the original image into four sub images each one containing some of the information of the original image. The sub image of our interest is the LL sub image which is identical to the original image.

When the DWT is implemented, the images will be decomposed into four parts, the most important components of the original image are stored in one part. When this part is used for comparing images it will give the main differences between the original images, this will increase the recognition rate even thought it uses images of smaller size.

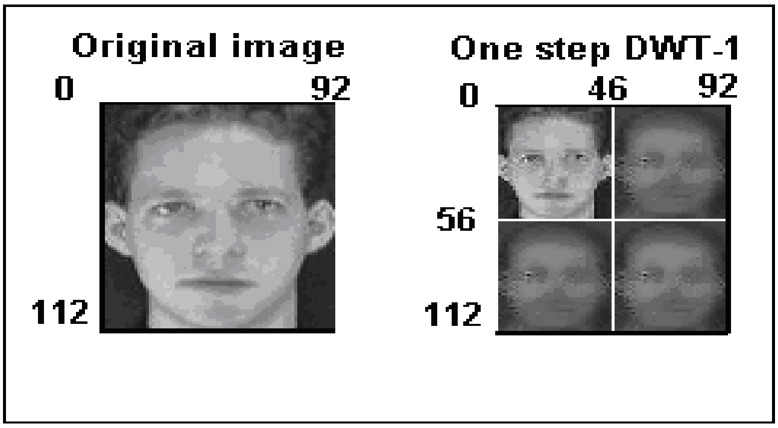


Figure 4.1: Decomposing an image using DWT (one step).

Second level DWT is implemented to achieve more reduction in image size, i.e. more reduction in comparison operations required. In addition this improvement in recognition rate causes some reduction in image resolution as shown in Figure 4.2.

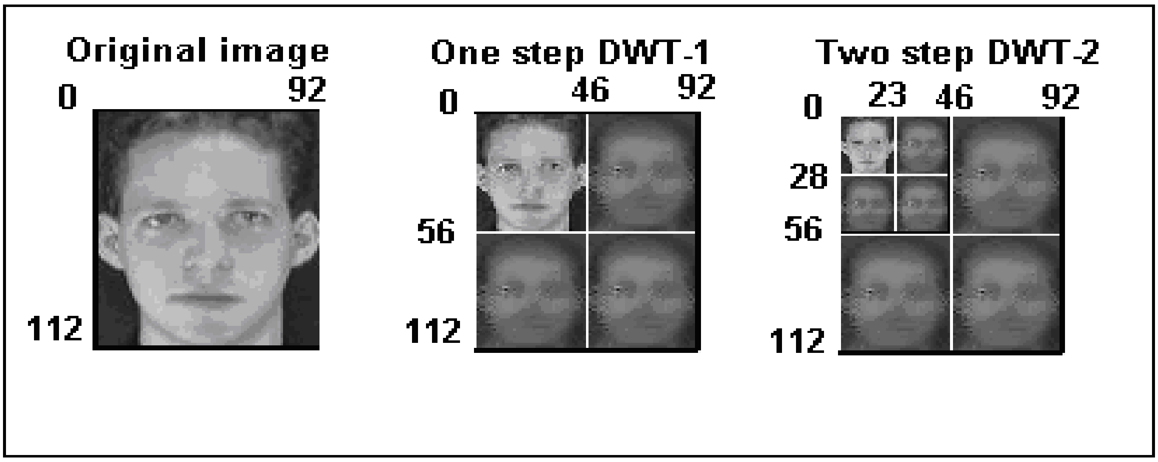


Figure 4.2: Decomposing an image using DWT (two steps).

The next section presents the results of the new technique after implementing the DWT in two steps on the original images, and shows the effect of the new optimization on the recognition rate and also on the execution time.

For each new image after extracting the features of the image the extracted features are with compared the features stored from the training stage, there will be two possibilities: the extracted features match the stored images or not, in this case the features of the new image are added to the database, See Figure 4.3.

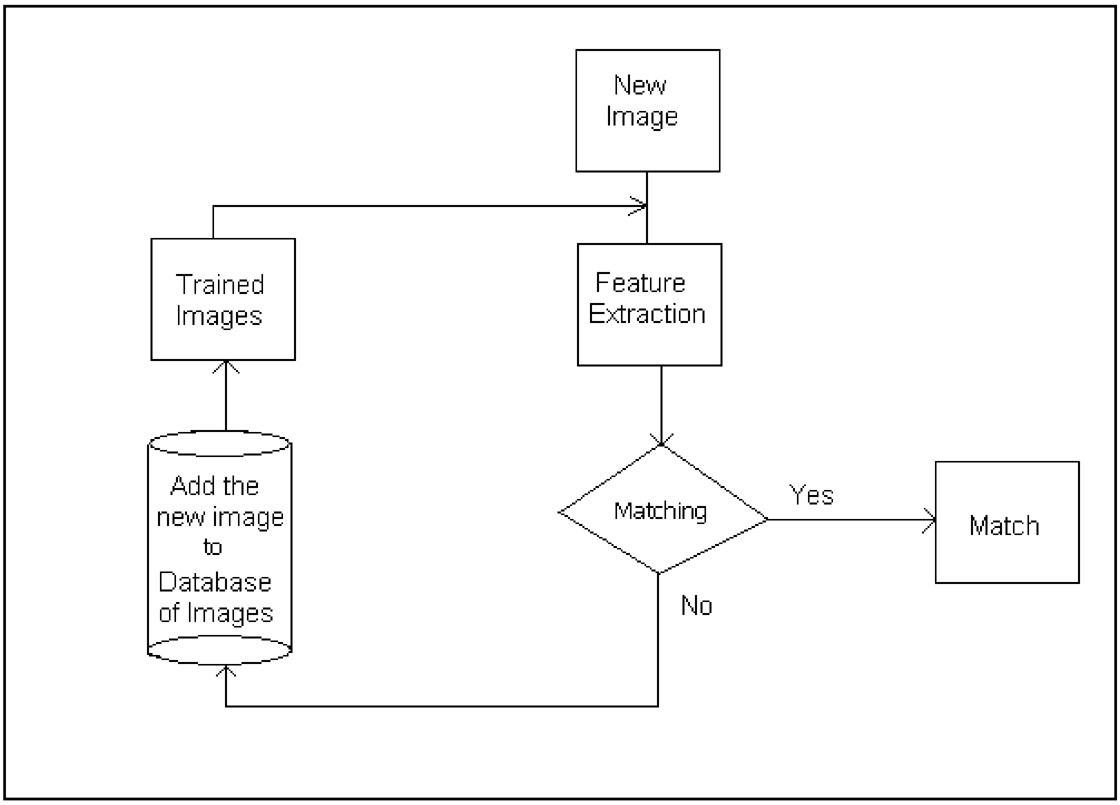


Figure 4.3: The diagram of the face recognition algorithm.

4.3 Implementation for PCA Algorithms

The principal component algorithm is implemented via MATLAB to demonstrate the recognition rate results.

PCA is used for face recognition via Euclidean distance, in which images are tested and recognized. The Euclidean distance is measured as below.

$$d(x, y) = \|x - y\| = \sqrt{\sum_{i=1}^{k}(x_i - y_i)^2} \qquad (4.1)$$

Where X and Y are the images compared with each other, and K is the number of elements “pixels” per image.

The previous four algorithms are tested via ORL database, which is extremely used for face recognition research.

PCA implementation:

For the standard PCA algorithm it will be implemented without applying the DWT at first, the results are shown in Chapter 5. Then the DWT will take place in the implementation in the two stages of the PCA algorithm; the training stage and the recognition stage.

The first implementation for the standard PCA will be without using the DWT, in this implementation the size of images in the training stage and in the recognition stage will be 92 x 112 pixels.

The second implementation for the standard PCA will take place using the first level of the DWT, where the images will be decomposed into four parts, and the PCA will deal with the LL (Low pas filter) part, where the LL is the part which was filtered using low pas filter, the low pass filter keeps only the main components of the original image. This implementation using the first level of the DWT makes the images of size 46 x 56 pixels. The PCA will deal with images of this size instead of the original size 92 x 112 pixels.

The third implementation will take place using the second level of the DWT, where the images from the first level will be decomposed for the second time; this will make the images size 23 x 28 pixels. After the second level of the DWT the PCA will deal with smaller images, this will make the comparison process faster, see Figure 4.4.

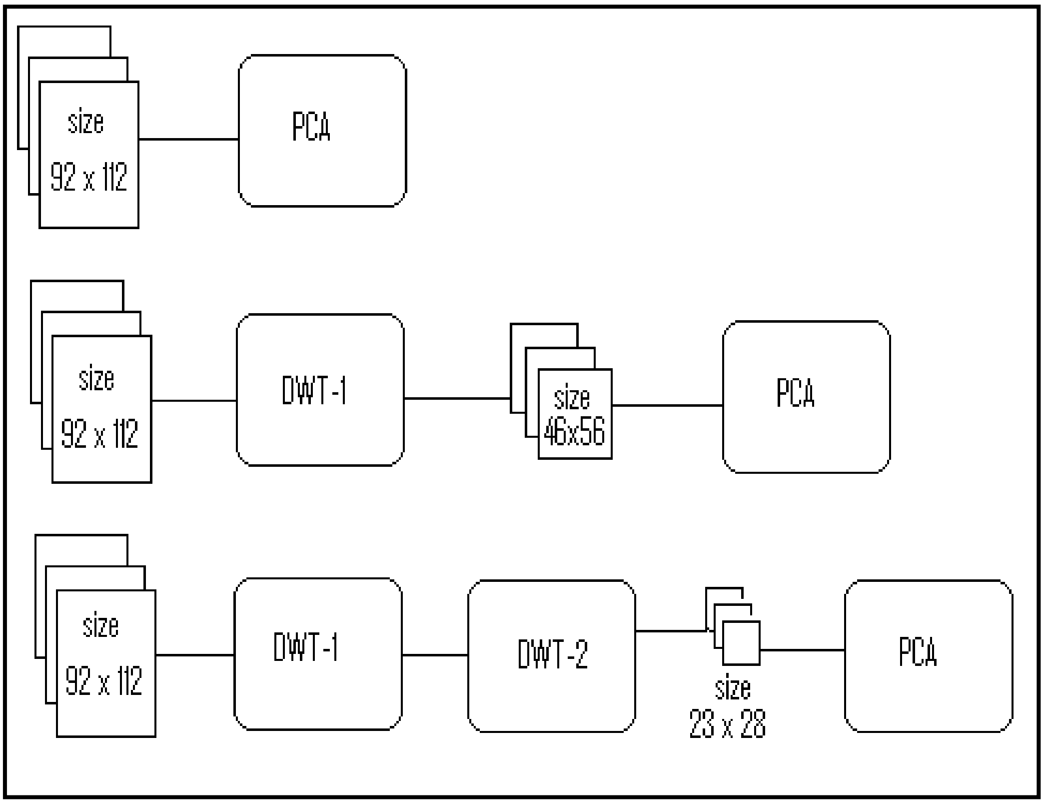


Figure 4.4: The using of DWT with PCA.

After implementing the PCA with the DWT the results of the recognition rate are obtained. A comparison between the results is presented in Chapter 5 showing the effect of using the DWT on the recognition rate.

2DPCA_R implementation:

The same steps will be followed with the 2DPCA_R algorithm; it will be implemented without applying the DWT at first, then the DWT will take place in the implementation.

The first implementation is without using the DWT, and the size of images will be 92 x 112 pixels. The second implementation will be using the first level of the DWT, where the images will become of size 46 x 56 pixels. The third implementation will take place using the second level of the DWT, and the images will become of size 23 x 28 pixels. See Figure 4.5.

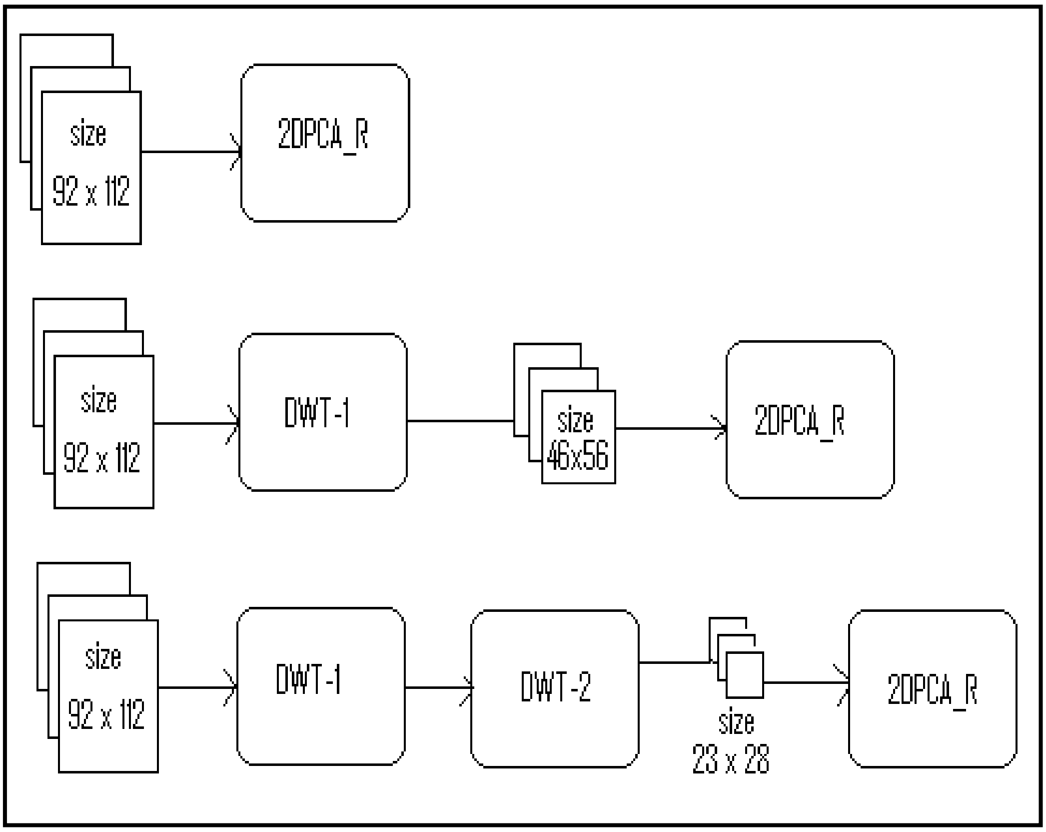


Figure 4.5: The using of DWT with 2DPCA_R.

The result are obtained and presented in Chapter 5, also a comparison between the results before and after using the DWT is presented.

2DPCA_C implementation:

The same procedure will be followed with the 2DPCA_C algorithm; first it will be implemented without using the DWT, where the size of images will be 92 x 112 pixels. Then the DWT will take place in the second implementation, where the images will become of size 46 x 56 pixels. The third implementation use the second level of the DWT, and the images will become of size 23 x 28 pixels. The result obtained and presented in Chapter 5. See Figure 4.6.

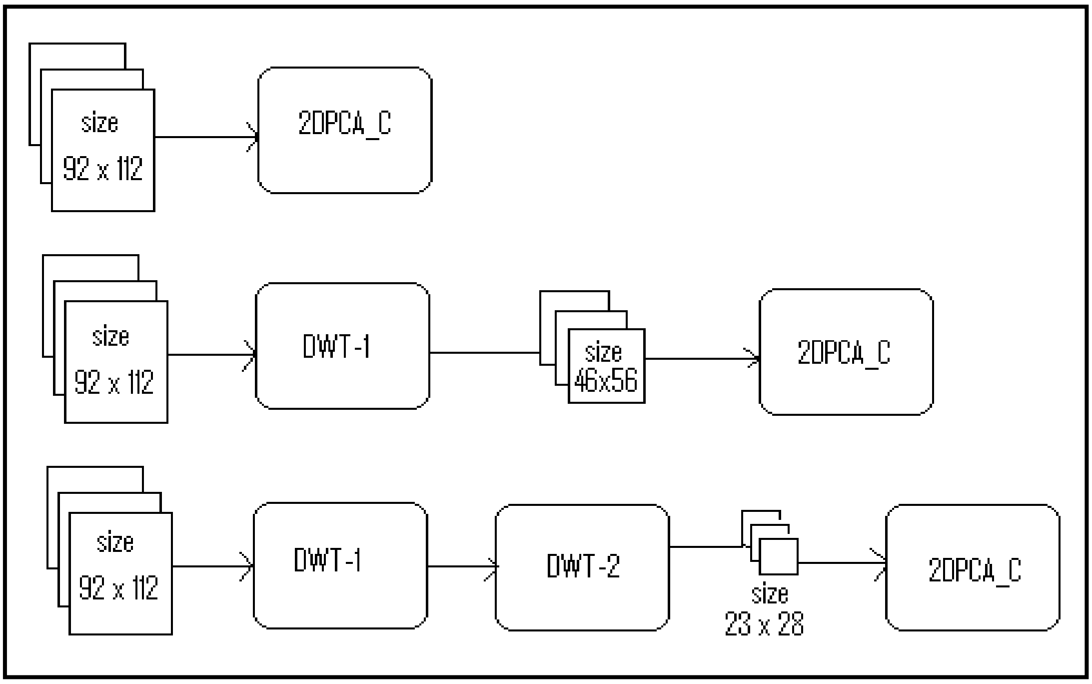


Figure 4.6: The using of DWT with 2DPCA_C.

(2D)²PCA implementation:

For the last algorithm, the same steps will be followed. It will be implemented without applying the DWT at first. The second implementation will be using the first level of the DWT, and the third implementation will take place using the second level of the DWT, The results obtained are presented in Chapter 5.

4.4 Summary

In this chapter the proposed optimization is presented, where the DWT is used to minimize the images size, which will reduce the number of comparisons needed to recognize a new image. This reduction in the number of comparisons needed lead to reduction in the execution time.

Chapter 5
Results and Discussion

5.1 Introduction

This chapter presents a comparison between the PCA algorithms discussed in chapter two; it presents a comparison in terms of recognition rate and Signal to Noise Ratio SNR. The optimized algorithm is presented here to show its effects on the recognition rate and performance.

In addition this chapter demonstrates the comparison between these algorithms and the optimization process to obtain an efficient one. The four PCA algorithms are tested via ORL database

5.2 The Results

This section presents the results of the recognition rate for the principal component analysis algorithms using DWT, The idea of using DWT is to decompose the images which are stored in the database so that it keeps all the features of the images in small matrices. DWT is applied first, and then PCA algorithm is implemented to recognize images.

5.2.1 The Results before Applying the Optimization

Table 5.1 presents the recognition rate for the standard principal component analysis algorithm, it can be obviously seen that the recognition rate depends on the number of images per person used in the training stage. When using more images per person in the training stage better recognition rate is achieved.

Table 5.1: Recognition rate of PCA.

Image per person	recognition rate
1	60%
2	71%
3	78%
4	84%
5	88%
AVG	76.2%

The Euclidean method shows that the recognition rate is within the range 60% to 88% as shown in Table 5.1 and Figure 5.1. Note that the testing is done on the ORL database, which contains 400 images for 40 different persons (10 images for each person). For example to calculate the percentage of the recognition rate using PCA and using one image per each different person in the training stage, 360 images remain for testing, take these images and test all of them, then divide the number of recognized images by 360 to obtain the recognition rate.

For two images per person in the training stage, 320 images remain for testing, to calculate the recognition rate, you have to test the 320 images then divide the total number of the recognized images by 320.

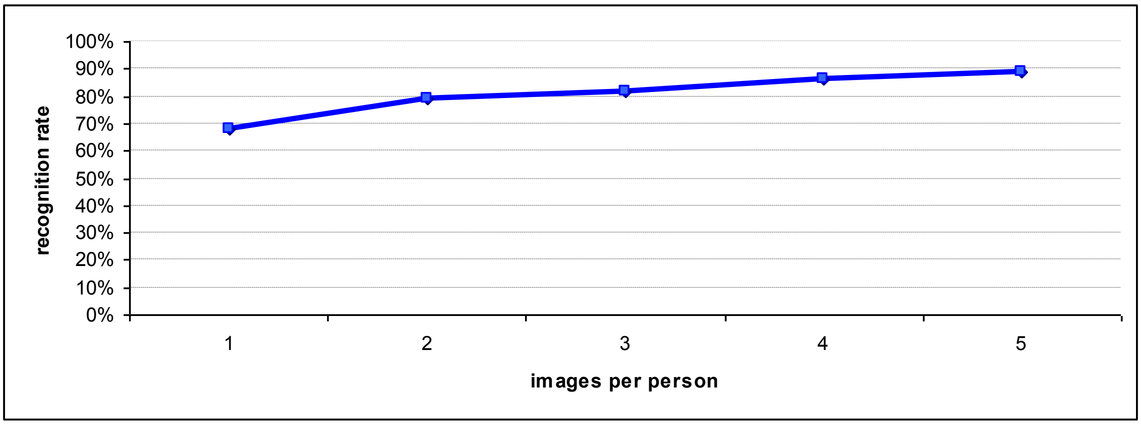


Figure 5.1: Recognition rate of PCA.

The results obtained from 2DPCA_C show that the recognition rate depends on the number of images per person used in the training stage like the standard PCA. The results of 2DPCA_C using the ORL database are listed in Table 5.2, and illustrated in Figure 5.2

Table 5.2: Recognition rate of 2DPCA_C.

Image per person	recognition rate
1	67%
2	75%
3	82%
4	87%
5	91%
AVG	80.4%

The results indicated that there is an improvement of recognition rate by 4.2% compared with the standard PCA algorithm.

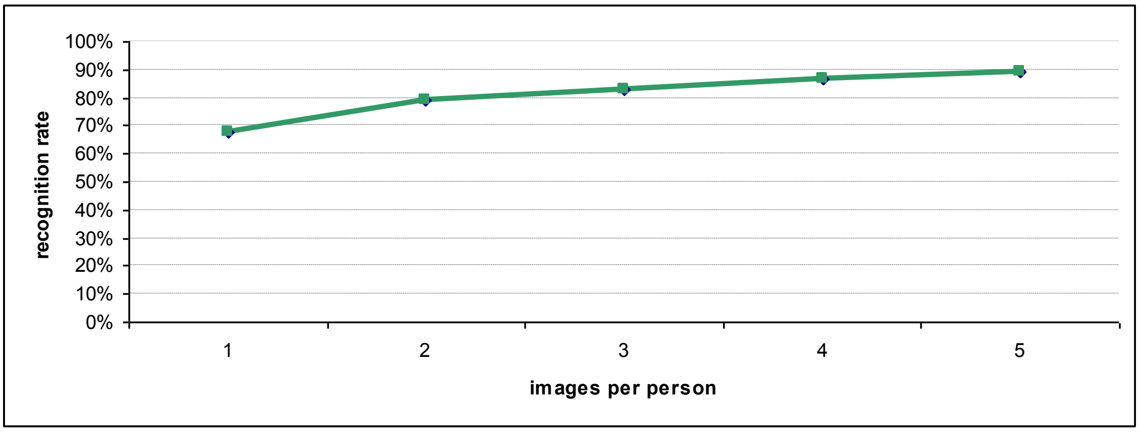


Figure 5.2: Recognition rate of 2D PCA _C.

Table 5.3 shows that the result of 2DPCA_R algorithm is better than that of 2DPCA_C algorithm. The recognition rate of 2DPCA_R is improved by 0.8% of 2DPCA_C algorithm. These results are also shown in Figure 5.3.

Table 5.3: Recognition rate of 2DPCA_R.

Image per person	recognition rate
1	68%
2	76%
3	82%
4	88%
5	92%
AVG	81.2%

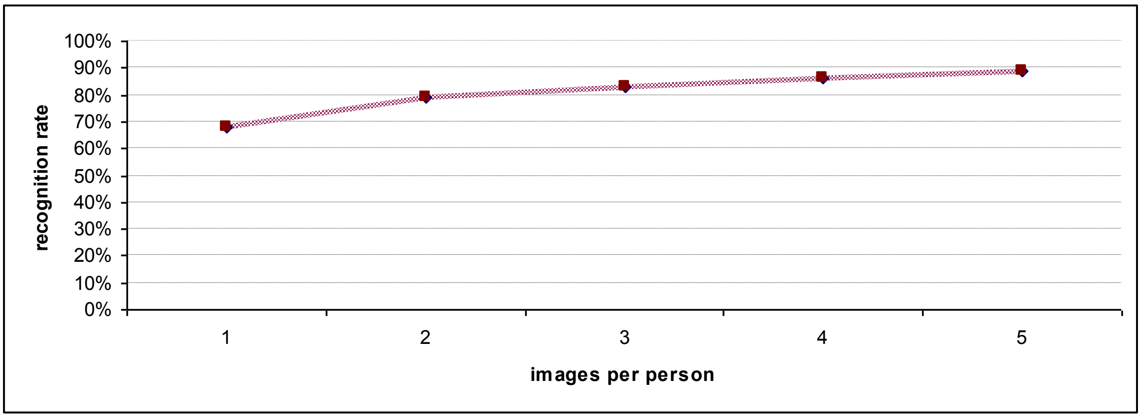


Figure 5.3: Recognition rate of 2D PCA _R.

The two dimensional two directional principal component analysis $(2D)^2$PCA algorithm is the most complicated , in which two projection matrices are used. Table 5.4 shows that for small number of images in the training set, the Euclidean method gives lower recognition rate, but if more than three images are used per person, it gives better recognition rate.

Table 5.4: Recognition rate of $(2D)^2$PCA.

Image per person	recognition rate
1	69%
2	77%
3	83%
4	89%
5	93%
AVG	82.2%

Also it can be seen that the AVG recognition rate using this algorithm is about 82.2% this means it is improved by 1% more than 2D PCA _R algorithm (see Figure5.4).

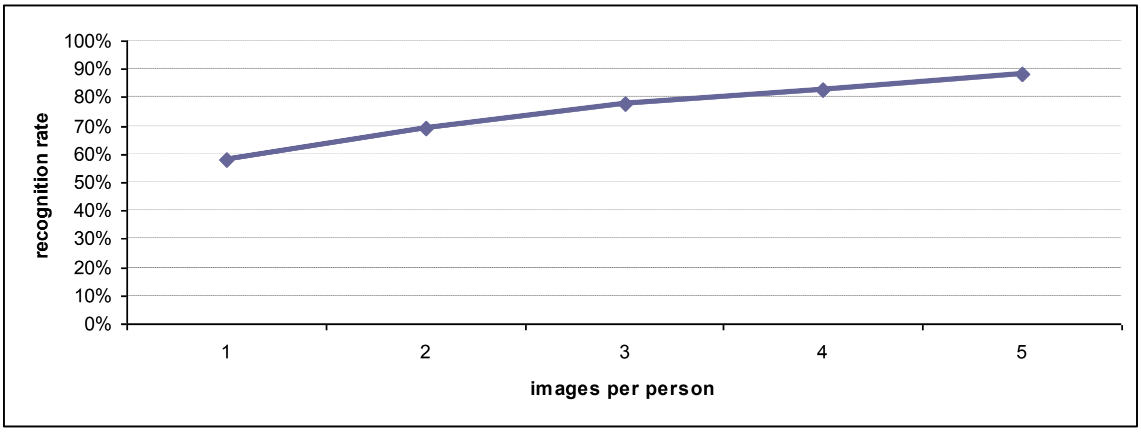


Figure 5.4: Recognition rate of $(2D)^2$PCA.

5.2.2 The Results after Applying the Optimization

Table 5.5 presents the recognition rate for the PCA algorithm before and after applying DWT, applying DWT indicates that the recognition rate increases and also the size of the feature matrices is reduced. DWT can be implemented via different types of filters such as “harr”, “db1”, “db2”...

Table 5.5: Recognition rate of PCA before and after using DWT with “haar” filter.

Image per person in training stage	Original recognition rate	Recognition rate using DWT-1	Recognition rate using DWT-2
1	60%	62%	66%
2	71%	73%	76%
3	78%	79%	82%
4	84%	85%	87%
5	88%	89%	91%
AVG	76.2%	77.6%	80.4%

Table 5.6 shows that the recognition rate is increased by 3.6% using ‘db2’ filter, but the increase of the recognition rate reaches 4.2% using ‘harr’ filter.

Table 5.6: Recognition rate of PCA before and after using DWT with “db2” filter.

Image per person in training stage	Original recognition rate	Recognition rate using DWT-1	Recognition rate using DWT-2
1	60%	62%	65%
2	71%	72%	75%
3	78%	79%	82%
4	84%	85%	86%
5	88%	89%	91%
AVG	76.2%	77.4%	79.8%

After testing many filters, it is obviously seen that the “haar” filter gives the best recognition rate. Depending on this fact the “haar” filter is used in to obtain results for the rest of this thesis.

The same effect happened when using the two dimensional principal component analysis in column direction 2DPCA_C, where using the DWT increased the recognition rate and reduced the number of comparisons needed by reducing the images size. Table 5.7 represents the results of the recognition rate of the original 2DPCA_C and the optimized one. These results are demonstrated in Figure 5.5.

Table 5.7: Recognition rate of 2DPCA_C before & after using DWT with “harr” filter.

Image per person in training stage	Original recognition rate	Recognition rate using DWT-1	Recognition rate using DWT-2
1	67%	69%	75%
2	75%	77%	80%
3	82%	84%	85%
4	87%	88%	90%
5	91%	92%	94%
AVG	80.4%	82%	84.8%

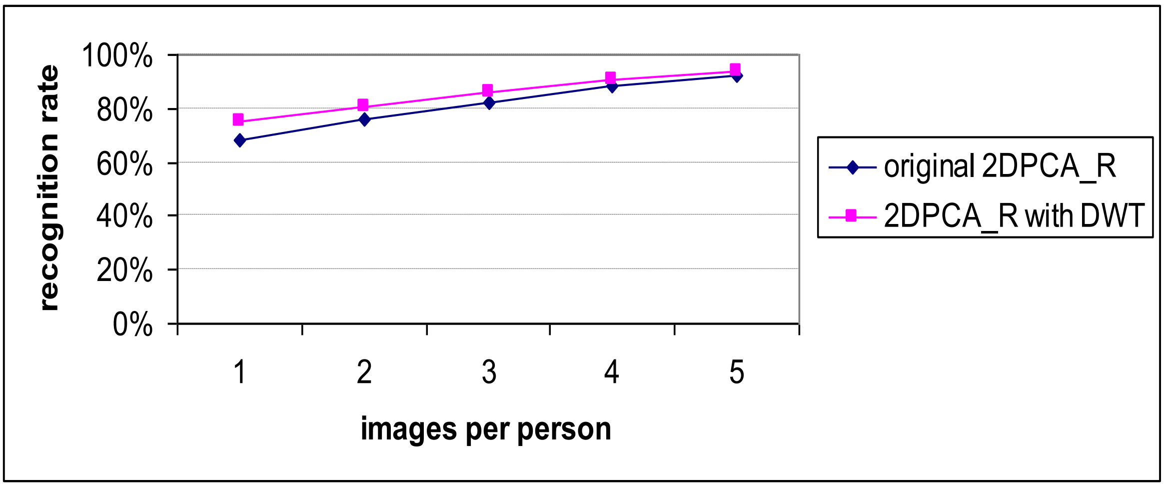


Figure 5.5: Comparison for recognition rate of 2DPCA_C before & after optimization.

2DPCA_R extracts features in a similar form of 2DPCA_C, so these results are approximately similar as shown in Table 5.8 and Figure 5.6.

Table 5.8: Recognition rate of 2DPCA_R before & after using DWT with "harr" filter.

Image per person in training stage	Original recognition rate	Recognition rate using DWT-1	Recognition rate using DWT-2
1	68%	70%	75%
2	76%	77%	81%
3	82%	83%	86%
4	88%	89%	91%
5	92%	92%	94%
AVG	81.2%	82.2%	85.4%

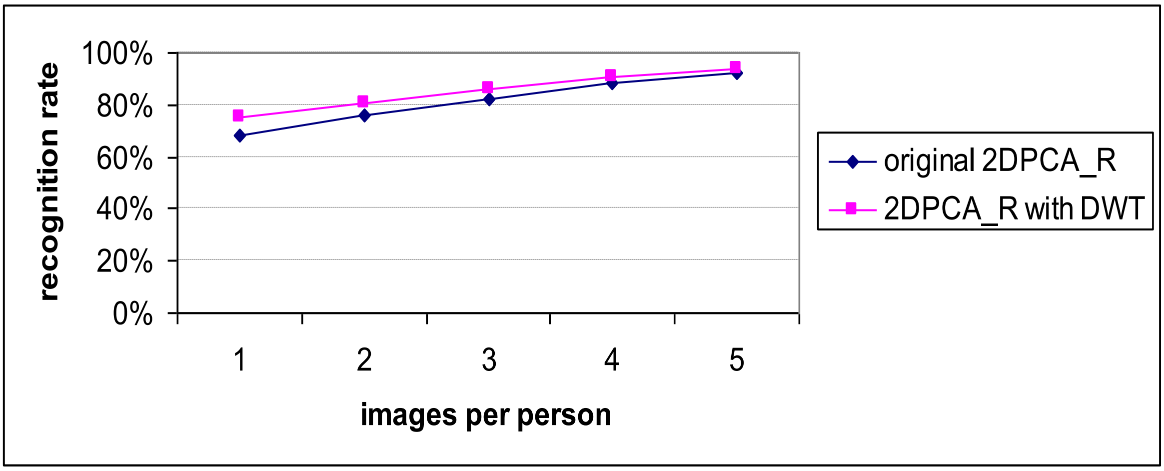


Figure 5.6: Comparison for recognition rate of 2DPCA_R before & after optimization.

The last algorithm is the two dimensional two directional principal component analysis $(2D)^2PCA$, this algorithm gives almost the best results for recognition rate, because this algorithm uses two projection matrices to extract the features from the images. As you can see from Table 5.9 the recognition rate of this algorithm after using DWT is better than the standard $(2D)^2PCA$, in which the recognition rate is improved by 1.8% in the first level and 4.6% in the second level, see figure 5.7.

Table 5.9: Recognition rate of $(2D)^2PCA$ before and after using DWT with "harr" filter.

Image per person in training stage	Original recognition rate	Recognition rate using DWT-1	Recognition rate using DWT-2
1	69%	72%	77%
2	77%	79%	82%
3	83%	85%	87%
4	89%	90%	93%
5	93%	94%	95%
AVG	82.2%	84%	86.8%

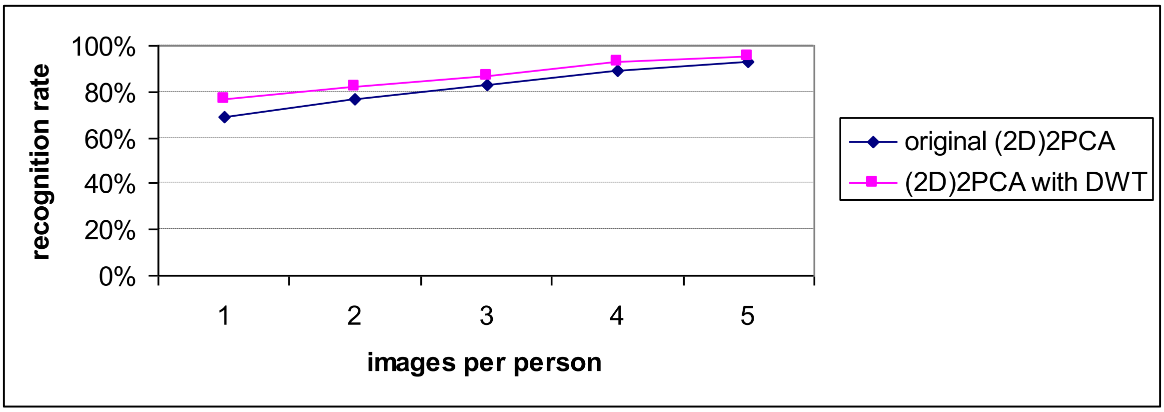


Figure 5.7: Comparison for recognition rate of $(2D)^2PCA$ before and after optimization.

Figure 5.8 shows the four PCA algorithms together in terms of recognition rate, it can be seen that the PCA in one dimension gives the lowest recognition rate, while the 2DPCA_R and 2DPCA_C gives almost the same results, and the best recognition rate refers to the $(2D)^2PCA$.

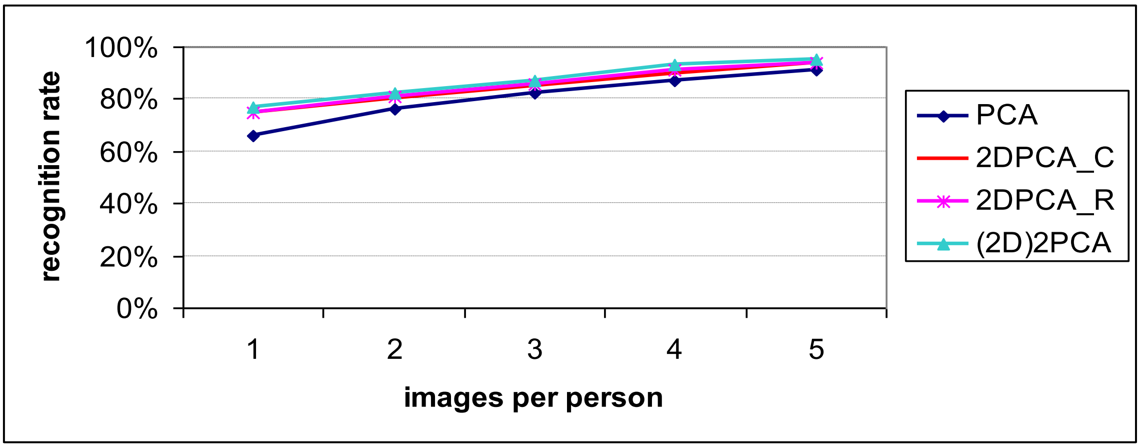


Figure 5.8: Comparison between PCA algorithms after using DWT.

The most important advantage of using DWT is the reduction of the number of comparisons needed to recognize an image, in which reduce matrices of the features are reduced . This technique puts all the features of the original image in one small matrix where its size is 1/4 of the original image size. This reduction in image size results in a reduction of the number of comparisons needed to recognize an image, and this results in a reduction in the execution time.

Table 5.10: The relation between the image size and comparisons needed.

Original image size	PCA comparisons	DWT-1 image size	DWT-1 comparisons	DWT-2 image size	DWT-2 comparisons
112 X 92	10304	SIZE / 4	2576	SIZE / 8	644

Table 5.10 shows the number of comparisons needed. As you can see, the number of comparisons needed to recognize an image decreases more and more if we apply DWT with more levels. Figure 5.9 represents the results of Table 5.10.

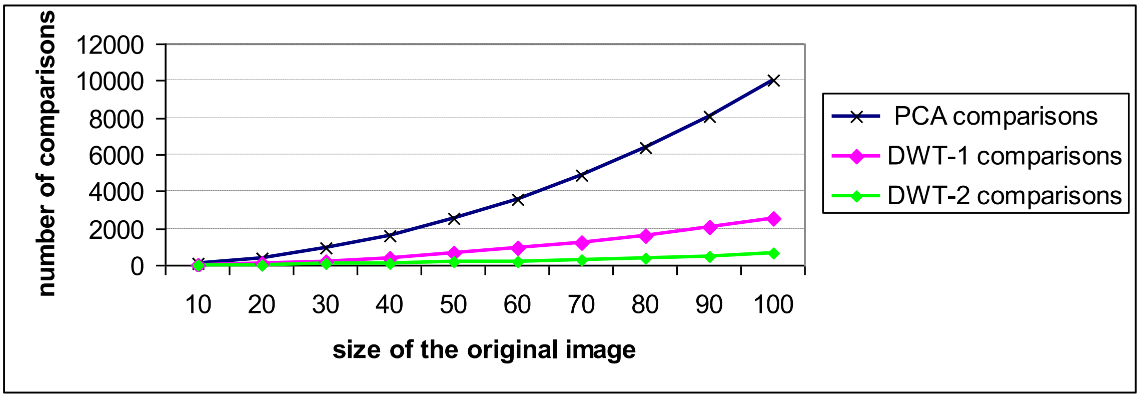


Figure 5.9: The relation between the image size and comparisons needed.

5.3 Signal-to-noise ratio (SNR)

(SNR) computes the signal-to-noise ratio between two images. This ratio is often used as a quality measurement between the original and the compressed image. The higher value of the SNR means better quality of the compressed image. The SNR can be calculated as in equation 5.1.

$$SNR = 10\log_{10}\left(\frac{R^2}{MSE}\right) \quad (5.1)$$

Where: **SNR** is Signal to Noise Ratio, **R** is the maximum value for the pixel, **MSE** is the mean square error, **a** is the original image, **aw** is the image after implementing DWT, **m** is the number of columns per image, **n** is the number of rows per image

The Mean Square Error (MSE) and the Signal to Noise Ratio (SNR) are the two measurements used to compare image compression quality. The MSE represents the cumulative squared error between the compressed and the original image. Table 5.11 contains the SNR for 40 images used in training stage using one image per person. SNR are measured after applying 1st level DWT on images.

Table 5.11: The SNR values for the PCA using DWT_1.

Image number	SNR DWT-1	Image number	SNR DWT-1	Image number	SNR DWT-1	Image number	SNR DWT-1
1	5.3081dB	11	6.7849 dB	21	7.8595 dB	31	7.1638 dB
2	6.4393 dB	12	5.258 dB	22	8.8434 dB	32	5.2652 dB
3	6.5397 dB	13	5.6689 dB	23	6.2265 dB	33	8.3833 dB
4	6.3225 dB	14	6.5049 dB	24	6.9383 dB	34	7.6169 dB
5	5.8142 dB	15	6.5916 dB	25	5.784 dB	35	6.3281 dB
6	5.4765 dB	16	6.9162 dB	26	5.9838 dB	36	5.3365 dB
7	6.379 dB	17	6.0734 dB	27	5.8651 dB	37	6.2313 dB
8	5.9618 dB	18	5.0019 dB	28	6.2402 dB	38	6.5007 dB
9	6.6961 dB	19	5.8397 dB	29	8.8675 dB	39	8.6315 dB
10	7.2276 dB	20	7.5351 dB	30	7.2216 dB	40	6.1068 dB

Table 5.11 contains SNR for 40 images used in training stage using one image per person. SNR are measured after applying 2nd level DWT on images.

Table 5.12: The SNR values for the PCA using DWT_2 .

Image Number	SNR DWT-2	Image number	SNR DWT-2	Image number	SNR DWT-2	Image number	SNR DWT-2
1	2.2945dB	11	4.7152 dB	21	8.1687 dB	31	5.7483 dB
2	4.499 dB	12	2.2516 dB	22	10.375 dB	32	2.0471 dB
3	4.9848 dB	13	2.7681 dB	23	4.0443 dB	33	7.7161 dB
4	4.5301 dB	14	3.5295 dB	24	5.4897 dB	34	5.8969 dB
5	3.1993 dB	15	5.0787 dB	25	3.4649 dB	35	3.9703 dB
6	2.3779 dB	16	4.9675 dB	26	3.5231 dB	36	2.1724 dB
7	4.0771 dB	17	3.6973 dB	27	3.1119 dB	37	3.5743 dB
8	3.062 dB	18	1.685 dB	28	3.5114 dB	38	4.5072 dB
9	4.9275 dB	19	3.0527 dB	29	9.6188 dB	39	8.1098 dB
10	4.3688 dB	20	6.2208 dB	30	6.1731 dB	40	3.7762 dB

The result of Table 5.11 and Table 5.12 are presented in Figure 5.10. This figure shows the differences of SNR between 1st level DWT and 2nd level DWT.

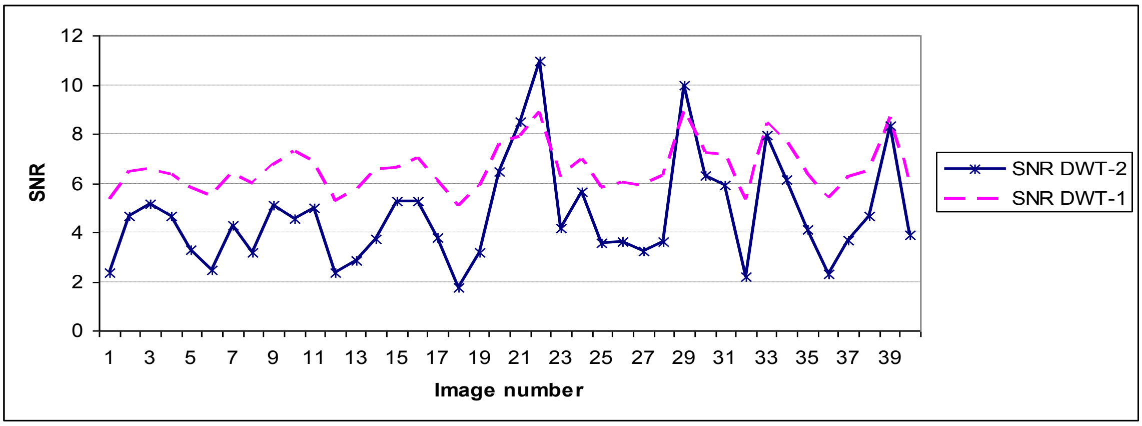


Figure 5.10: The values of the SNR for the 40 images used in the training stage.

When we use the PCA algorithm with two images per person we will have 80 SNR value, one for each image. As shown in Figure 5.11.

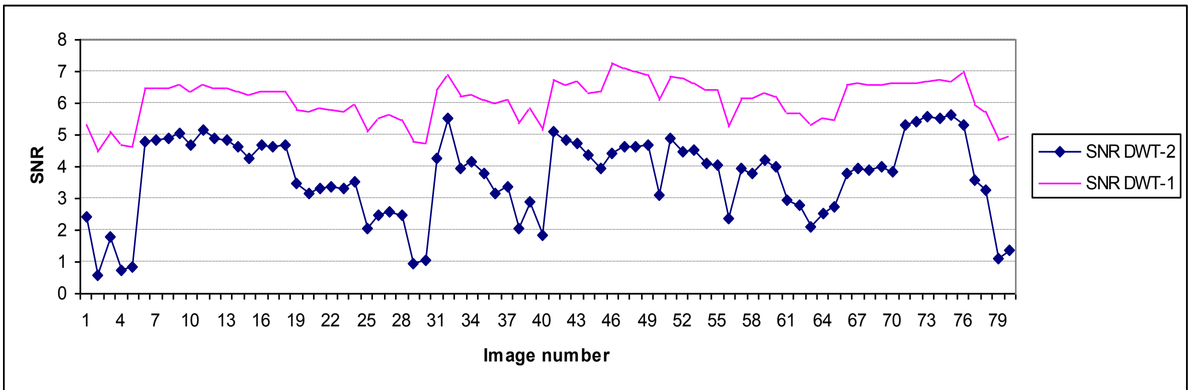


Figure 5.11: The values of the SNR for the 80 images used in the training stage.

When using the PCA algorithm with three images per person we will have 120 SNR value, one for each image. As shown in Figure 5.12.

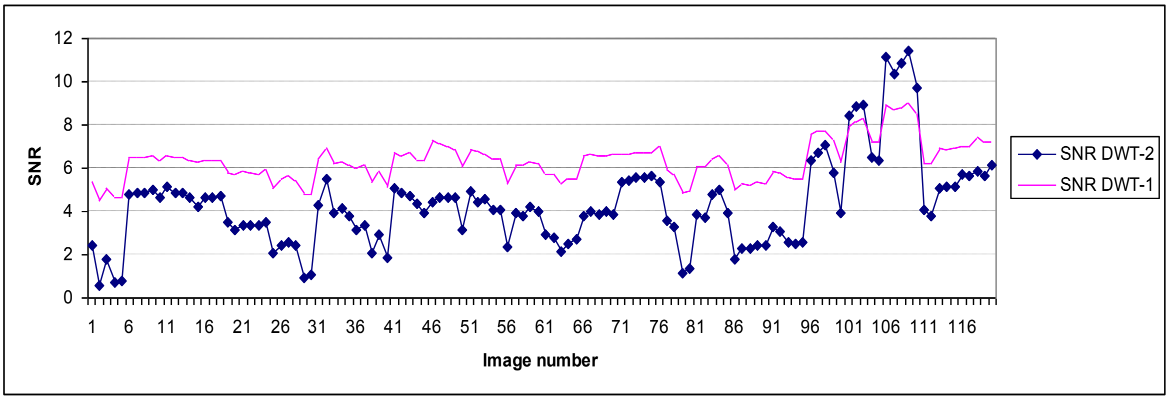


Figure 5.12: The values of the SNR for the 120 images used in the training stage.

When we use the PCA algorithm with four images per person we will have 160 SNR value, one for each image. As shown in Figure 5.13.

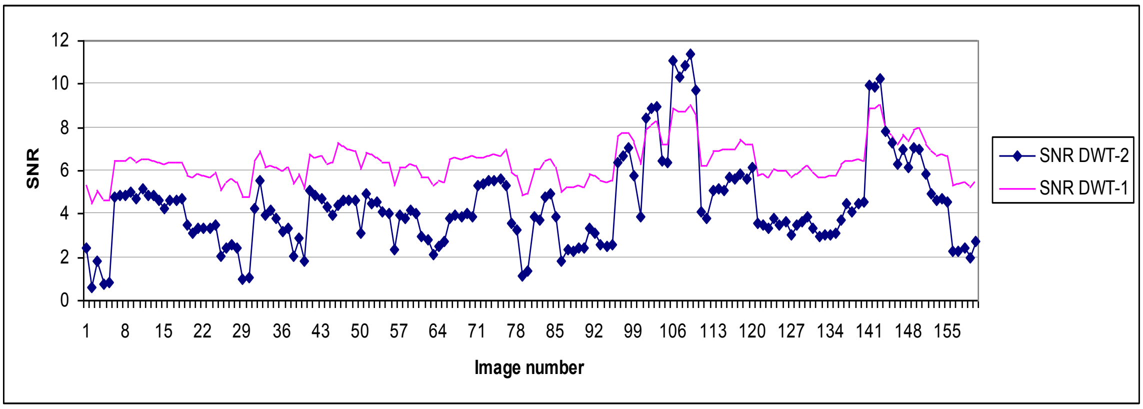


Figure 5.13: The values of the SNR for the 160 images used in the training stage.

When using the PCA algorithm with five images per person we will have 200 SNR value, one for each image. As shown in Figure 5.14.

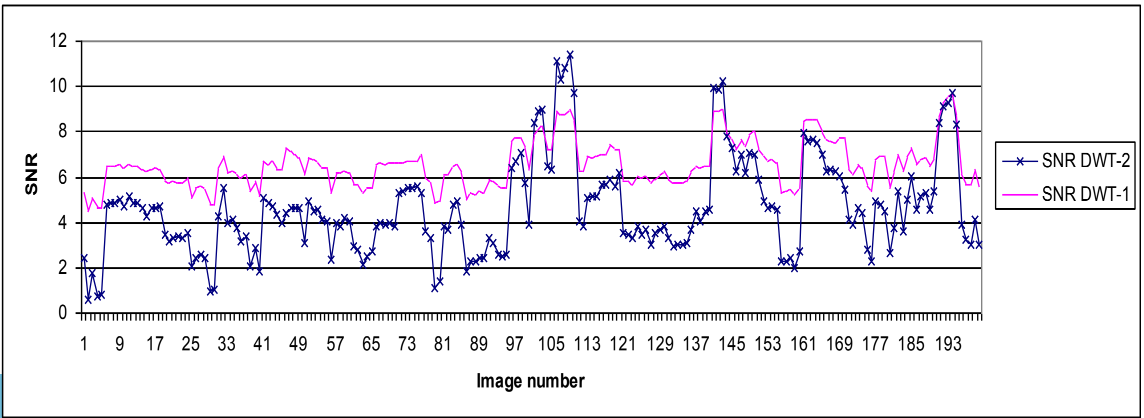


Figure 5.14: The values of the SNR for the 200 images used in the training stage.

Table 5.13 contains the average values for the SNR after using DWT-1 and DWT-2 for more than one image per person.

Table 5.13: The Average SNR values for PCA using DWT_2.

Image per person	Average SNR DWT-1	Average SNR DWT-2
1	6.5433	4.5322
2	6.5675	4.5382
3	6.5695	4.5319
4	6.5538	4.4928
5	6.5042	4.4412

5.4 Summary

This chapter shows the comparison between the implemented algorithms. According to the results, we can see that two dimensional two directional PCA is the most powerful algorithm. In addition the proposed algorithm using DWT leads to improvement of recognition rate, and leads to reduction in the execution time because it reduces the images size.

The total number of images affects the average image face, and this affects the images later when we subtract the mean image from them, we subtract the mean image then we calculate the DWT, so the total number of images affects the SNR values

Actually, using DWT does not reduce the image size, but it decomposes the image into four parts, the size of each part is equal to the quart of the original image size. According to the DWT one of these parts contains the most important components of the original image; like the lines and the shapes. We used this part instead of the original image; this will increase the recognition rate because we compare only the main components and the main differences of the original images.

Chapter 6
Conclusion and future work

6.1 Conclusion

Face recognition is considered the most effective method for personal identification. There are many applications for face recognition including surveillance systems, access control and information security.

Many algorithms have been proposed for face recognition and these algorithms differ in recognition rate. The Principal Components Algorithms are considered as the most effective algorithms in face recognition.

This thesis presents a study of the recognition rate for the PCA algorithms. Also this thesis presents an optimization on the principal components algorithms. A comparison between the recognition rate of the PCA algorithms before and after the optimization is also presented in this thesis.

Four PCA algorithms were discussed and implemented in this thesis: Principal Component Analysis PCA, Two Dimensional Principal Component Analysis in Column Direction 2DPCA_C, Two Dimensional Principal Component Analysis in Row Direction 2DPCA_R and the Two Dimensional Two Directional Principal Component Analysis $(2D)^2$PCA.

All the algorithms compared in this thesis are similar in the procedure, but they differ in the dimensions. The first algorithm deals with images as vectors, while the last algorithm deals with them as matrices.

An optimization is proposed on PCA algorithms. Then these algorithms are compared before and after applying the proposed optimization, the results were presented in terms of recognition rate for these algorithms.

The principal components algorithms in general have two major stages: the trained stage and the recognition stage. In the training stage the features of the training images are extracted using a projection matrix. The features are stored in order to be used in the recognition stage. In the recognition stage, the features of an incoming unknown image are extracted and compared with the stored features.

It was proved practically that the Two Dimensional Two Directional Principal Component Analysis $(2D)^2$PCA gives the best recognition rate, because this algorithm uses two projection matrices, one in the column direction and another in the row direction, so it compares the components of the images in both directions at the same time, and this is the reason it gives the best recognition rate.

For the second and the third algorithms which are Two Dimensional Principal Component Analysis in Column Direction 2DPCA_C and Two Dimensional Principal Component Analysis in Row Direction 2DPCA_R, they give almost the same recognition rate. But they will differ if the number of columns per images is different from the number of rows per images, see the following:

If $m > n$ then the projection matrix for the 2DPCA_C will be of size $m \times m$, and the projection matrix of the 2DPCA_R will be of size $n \times n$, in this case the projection matrix of the 2DPCA_C is larger than 2DPCA_R so it will contain more components to compare and it will give higher recognition rate.

As for the standard PCA it deals with the images as vectors and the size of the recognition rate will be equal to the number of images used. And because it deals with images in one dimension it will give the worst recognition rate, unless it uses a large number of images in the training stage, which makes the size of the projection matrix larger.

The proposed optimization on the four algorithms mentioned above, is to apply the Discreet Wavelet Transform DWT to the images before the PCA algorithms take their place.

Using DWT will reduce the images size and put all the features of the face image in one small matrix. This will reduce the execution time of the algorithm due to the reduction in the number of comparison operations required to compare the new image with the stored ones.

This thesis achieves an improvement in the recognition rate for the principal components algorithms using the DWT between 4-5%. In addition a reduction of the execution time of the implemented algorithms is achieved.

6.2 Future Work

This thesis provides a study for the recognition rate for principal components face recognition algorithms. Many directions can be taken to pursue future research in this area including:

- **A study on the recognition rate using special remarks in the faces**.

  Such a study depends on a special remark in the face like a mole or flat nose or fat lips, a proposed technique to use these remarks for classifying the database into groups can be implemented, this calcification will reduce the amount of comparisons to recognize an incoming unknown image, this reduction can be achieved if the new image is classified before the recognition stage. This will reduce the execution time also, because the new image will not be compared with the whole database, it will be compared with a specific class or group.

- **A study on the recognition rate using FFT instead of DWT**.

  Such a study can be implemented using Forward Fourier Transform, the FFT has similar characteristics to the DWT, a study using the FFT can be done, and then a comparison between the DWT and the FFT can be presented, showing the effect of each algorithm on the recognition rate and on the execution time.

- **A study on using DWT with different face recognition algorithms**.

  Such a study can be done by applying DWT on neural network face recognition algorithms, and then a comparison between the effect of the DWT on the performance of the neural network and the effect of the DWT on the performance of the PCA algorithms can be presented, a comparison study can show when to use the DWT and where.

References

[1] Xiaoguang Lu, Yunhong Wangy and Anil K. Jain ***Combining Classifiers For Face Recognition*** Department of Computer Science & Engineering, Michigan State University 2002.

[2] Pablo Navarrete and Javier Ruiz-del-Solar ***Comparative Study between different Eigenspace-based Approaches*** supported by the DID Chile Department of Electrical Engineering, Universidad Chile, CHILE 2001.

[3] Kyong Chang, Kevin W. Bowyer, and Sudeep Sarkar ***Comparison and Combination of Ear and Face Images In Appearance-Based Biometrics*** supported by the DARPA "Human ID at a Distance" program 2002.

[4] Ralph Gross, Jie Yang and Alex Waibel ***Face Recognition in a Meeting Room*** Interactive Systems Laboratories Carnegie Mellon University Pittsburgh, PA 15213, USA 2000.

[5] Kyungim Baek, Bruce A. Draper and J. Ross Beveridge ***PCA vs. ICA: A comparison on the FERET data set*** Department of Computer Science Colorado State 2001.

[6] Xiaoguang Lu and Anil K. Jain ***Resampling for Face Recognition*** Dept. of Computer Science & Engineering, Michigan State University 2002.

[7] Mauricio Orozco-Alzate **Comparison *of the nearest feature classifiers for face recognition*** Machine Vision and Applications (2006) .

[8] Jin Ok Kim1 and Jin Soo Kim2 ***Real-Time Implementation of Face Detection for a Ubiquitous Computing*** Kwangwoon University, 447-1, Wolgye-dong (2005).

[9] Department of Electronic Engineering, City University of Hong Kong, Kowloon, Hong Kong. ***Face Matching in Large Database by self-Organizing Maps*** Neural Processing Letters (2006) .

[10] Department of Mathematical and Physical Science, Japan Women's University ***Fast Face-Recognition Optical Parallel Correlator Using High Accuracy Correlation Filter*** Optical Review Vol. 12, No. 6(2005)

[11] Fengzhi Dai · Tatsuya Kodani · Yutaka Fujihara ***Research on a face recognition system by the genetic algorithm*** Artif Life Robotics (2007) .

[12] Eriko WATANABE_, Sayuri ISHIKAWA, and Kashiko KODATE ***A Highly Accurate Face Recognition System Using Filtering Correlation*** OPTICAL REVIEW Vol. 14, No. 5 (2007) .

[13] Saleh Aly · Naoyuki Tsuruta · Rin-Ichiro Taniguchi ***Face recognition under varying illumination using Mahalanobis self-organizing map*** Artif Life Robotics (2008).

[14] Mohamed Aly ***Face Recognition using SIFT Features*** CNS186 Term Project Winter 2006

[15] Thomas Fromherz, Peter Stucki and Martin Bichsel. ***A Survey of Face Recognition.*** University of Zurich (1997).

[16] M.A. Turk and A.P. Pentland. ***Face Recognition Using Eigenfaces***. IEEE Conf. on Computer Vision and Pattern Recognition 1991; 586-591.

[17] Zhou, R. Chellappa. ***Multiple-exemplar discriminant analysis for face recognition***. International Conference on Pattern Recognition 2004; 191-194.

[18] Daoqiang Zhang and Zhi-Hua Zhou. ***(2D)2PCA: 2-Directional 2-Dimensional PCA for Efficient Face Representation and Recognition***. Neurocomputing 2005; 69: 224-231.

[19] Wendy S. Yambor Bruce A. Draper J. Ross Beveridge ***Analyzing PCA-based Face Recognition Algorithms: Eigenvector Selection and Distance Measures*** Fort Collins, CO, U.S.A 80523 July 1, 2000

[20] Florian Luisiera, Thierry Blua ***Which wavelet bases are the best for image denoising*** Biomedical Imaging Group 2005

[21] Donald Walter ***Image Compression Fourier and Wavelet Analysis*** CTH & GU 2008

[22] Asim Shankar and Priyendra Deshwal ***Face Detection in images : Neural networks & Support Vector Machines*** Indian Institute of Technology, Kanpur April 2002.

[23] Juwei Lu, K.N. Plataniotis, and A.N. Venetsanopoulos ***Face Recognition Using LDA*** 2003

[24] Su Hongtao, David Dagan and Zhao Rong ***Face Recognition Using Multi-feature and Radial Basis Function Network*** Hong Kong Polytechnic University. supported by the UGC and ARC Grants 2002.

[25] Meng Joo Er, Shiqian Wu, Juwei Lu and Hock Lye ***Face Recognition With Radial Basis Function (RBF) Neural Networks*** IEEE NEURAL NETWORKS, VOL. 13, NO. 3, MAY 2002.

[26] Zehang Sun, Xiaojing Yuan, and George Sushil ***Neural-Network-Based Gender Classification Using Genetic Search for Eigen-Feature Selection*** University of Nevada 2001.

[27] Javad Haddadnia , KarimFaez and Majid Ahmadi ***N-Feature Neural Network Human Face Recognition*** Electrical Engineering Department, Amirkabir University of Technology, Tehran, Iran 2002.

[28] Kyungim Baek, Bruce A. Draper and J. Ross Beveridge ***PCA vs. ICA: A comparison on the FERET data set*** Department of Computer Science Colorado State 2001.

[29] Xiaoguang Lu and Anil K. Jain ***Resampling for Face Recognition*** Dept. of Computer Science & Engineering, Michigan State University 2002.

[30] Pablo Navarrete and Javier Ruiz-del-Solar ***Comparative Study between different Eigenspace-based Approaches for Face Recognition*** Department of Electrical Engineering, Universidad de Chile, CHILE 2001

[31] Pong C. Yuena; , ***Representation Using Independent Component analysis*** Department of Computer Science, Hong Kong Baptist University, 224 Waterloo Road, Kowloon, Hong Kong 2002

[32] Su Hongtao(a,b), David Dagan Feng(a,c), Zhao Rong-chun(b) ***Face Recognition Using Multi-feature and Radial*** Dept. of Computer Science and Engineering, Northwestern Polytechnical University.china 2003

[33] Pablo Navarrete and Javier Ruiz-del-Solar ***Analysis and Comparison of Eigenspace-based Face Recognition*** Approaches Department of Electrical Engineering, Universidad de Chile, CHILE 2002

[34] Aapo Hyvärinen and Erkki Oja ***Independent Component Analysis:Algorithms and Applications*** Neural Networks Research Centre Helsinki University of Technology. 2000

[35] Zehang Sun, Xiaojing Yuan, and George Bebis Sushil J. Louis ***Neural-Network-Based Gender Classification Using Genetic Search for Eigen-Feature Selection*** Department of Computer Science 2004

[36] Aca Gaˇci´c, Markus P¨uschel, and Jos´e M. F. Moura ***Automatically Generated High-Performance Code For Discrete Wavelet Transforms*** Department of Electrical and Computer Engineering Carnegie Mellon University, 2004