STARS

Electronic Theses and Dissertations, 2004-2019

2007

Novel Facial Image Recognition Techniques Employing Principal Component Analysis

Moataz Mahmoud Abdelwahab
University of Central Florida

NOVEL FACIAL IMAGE RECOGNITION TECHNIQUES EMPLOYING
PRINCIPAL COMPONENT ANALYSIS

By

MOATAZ M. ABDELWAHAB
B. S. Alexandria University, 1999
M. S. Alexandria University, 2001

A dissertation submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy
in the School of Electrical Engineering and Computer Science
in the College of Engineering and Computer Science
at the University of Central Florida
Orlando, Florida

Fall Term
2007

Major Advisor: Wasfy B. Mikhael

ABSTRACT

Recently, pattern recognition/classification has received considerable attention in diverse engineering fields such as biomedical imaging, speaker identification, fingerprint recognition, and face recognition, etc.

This study contributes novel techniques for facial image recognition based on the Two dimensional principal component analysis in the transform domain. These algorithms reduce the storage requirements by an order of magnitude and the computational complexity by a factor of 2 while maintaining the excellent recognition accuracy of the recently reported methods.

The proposed recognition systems employ different structures, multicriteria and multitransform. In addition, principal component analysis in the transform domain in conjunction with vector quantization is developed which result in further improvement in the recognition accuracy and dimensionality reduction.

Experimental results confirm the excellent properties of the proposed algorithms.

ACKNOWLEDGMENTS

I would like to express my deepest gratitude to Dr. Wasfy Mikhael for his continuous support and patience. His help and guidance throughout my studies at UCF are invaluable. I will always be indebted to him for he has taught me a lot. His experience was really of great benefit to me. His advice in academia and otherwise have undoubtedly left a mark on me.

My deepest thanks also go to members of my committees: Dr. I. Batarseh , Dr. F. Moslehy, Dr. A. Berg, and Dr. T. Yang for their encouragement and patience during my studies.

Last but not least, I would like to thank my parents. They inspired me, and simply put: without them I would not be the person I am today.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

2DPCA : Two Dimensional Principal Component Analysis

2D-2DPCA : Two Directional Two Dimensional Principal Component Analysis

2D-DCT: Two Dimensional Discrete Cosine Transform

DCT: Discrete Cosine Transform

DT: Decision Tree

ICA : Independent Component Analysis

LVQ : Linear Vector Quantization

MCMT: Multi Criteria Multi Transform

MSE: Minimum Square Error

M-TD2DPCA : Modified Transform Domain 2DPCA

N: Number of Training images

NN: Neural Network

PCA : Principal Component Analysis

SD-TD2DPCA : Spatial Domain- Transform Domain 2DPCA

SVD: Singular Value Decomposition

TD2DPCA : Transform Domain 2DPCA

TD/2D2DPCA : Transform Domain Two Directional 2DPCA

TD2DPCA/VQ : Transform Domain 2DPCA in conjunction with Vector Quantization

VQ: Vector Quantization

WT: Wavelet Transform

CHAPTER 1 INTRODUCTION

Pattern recognition and classification has become one of the important areas in research, such as biomedical imaging, speaker identification, fingerprint and image recognition.

Recognition accuracy, storage requirements, and computational complexity are considered the most important factors in designing any pattern recognition system. Within the last several years, numerous algorithms have been proposed to solve such problems. One of the most successful techniques is the principal component analysis.

Principal component analysis (PCA), or Karhunen- Loeve expansion, technique is used for feature extraction from data. It is widely used in the areas of pattern recognition. Sirovich and Kirby [1], [2] first used PCA to represent pictures of human faces. In 1991 Turk and Pentland [3] presented the well-known Eigenfaces method for face recognition. Since then, PCA has been widely investigated and has become one of the most successful approaches in face recognition [4], [5], [6], [7].

Penev and Sirovich [8] discussed the problem of dimensionality of the "face space" when eigenfaces are used for representation.

Zhao and Yang [9] tried to account for the arbitrary effects of illumination in PCA-based vision systems by generating an analytically closed form formula of the covariance matrix for the case with a special lighting condition and then generalizing to an arbitrary illumination via an illumination equation.

However, Wiskott et al. [10] pointed out that PCA could not capture even the simplest invariance unless this information is explicitly provided in the training data. They proposed a technique known as elastic bunch graph matching to overcome the weaknesses of PCA.

Recently, independent component analysis (ICA) and kernel principal component analysis (Kernel PCA) were presented. Bartlett et al. [11] and Draper et al. [12] ,[13] found that using ICA for face representation was better than PCA when cosines were used as the similarity measure rather using the Euclidean distance. Yang [14] used Kernel PCA for face feature extraction and recognition and showed that the Kernel Eigenfaces method outperforms the classical Eigenfaces method. However, ICA and Kernel PCA are both computationally more expensive than PCA. The experimental results in [14] showed the ratio of the computation time required by ICA, Kernel PCA, and PCA is, on average, 8.7: 3.2: 1.0. [15].

In the PCA technique, the 2D face image matrices are concatenated into 1D image vectors. The resulting image vectors representing the training images usually lead to a high dimensional image vector space. Evaluating the covariance matrix in such high dimensional space is usually difficult, especially in the presence of small number of training images, which is typically in practice. Fortunately, the eigenvectors (eigenfaces) can be calculated efficiently using the SVD techniques [1], [2] . Since the eigenvectors are statistically determined by the covariance matrix, using SVD does not really present the actual eigenvectors. Recently a two-dimensional principal

component analysis (2DPCA) technique [15], was developed for extracting image features. The 2DPCA is based on 2D matrices rather than 1D vectors. The covariance matrix is calculated directly using the original matrices representing the training images. In contrast to the covariance matrix of PCA, the size of the image covariance matrix using 2DPCA is much smaller. As a result, the covariance matrix is easier to evaluate accurately, so less time is required to determine the corresponding eigenvectors. In addition, the recognition accuracy is higher than that obtained using the PCA method. However the storage requirements are not as good as the PCA.

In this research, novel techniques based on the 2DPCA in the transform domain that overcomes this drawback and improves the computational time while maintaining or improving the recognition accuracy advantage of the 2DPCA method are investigated and developed. These improvements were achieved by using other classification tools such as Vector quantization (VQ) in conjunction with the PCA methods.

The rest of this dissertation is organized as follows.

In chapter two, a brief description of the 2DPCA analysis is introduced. In addition, a two directional 2DPCA technique is presented. The training and testing algorithms are explained for these two methods.

 In chapter three, a practical, facial-recognition, transform-domain, two-dimensional, principal component analysis technique (TD2DPCA) [16],[17],[18] is presented. The

TD2DPCA reduces the storage and computational requirements to train the system by approximately a factor of ten and two respectively, while retaining the high recognition accuracy, relative to the state of the art, 2DPCA method. The TD2DPCA's excellent properties are confirmed experimentally for noise free and noisy images.

In Chapter four a Transform Domain Two-Directional Two-Dimensional Principal Component Analysis (TD/2D2DPCA) algorithm [19] applied to facial recognition is presented. This algorithm has attractive properties with respect to storage and computational requirements, while maintaining the high recognition accuracy achieved before. In addition, a Modified -TD2DPCA (M-TD2DPCA) method [20], presenting a new way of calculating the covariance matrix of the training images is proposed. This approach reduced the computational complexity required to calculate the covariance matrix while maintaining good accuracy.

Chapter five presents techniques which combine VQ with transform domain principal component representation in the training mode. This results in drastically increasing the speed of recognition in the testing mode [21],[22].

In Chapter six, a parallel structure recognition system is introduced [23],[24]. The system employs different structures, multicriteria and multitransform techniques. In addition, principal component analysis in the transform domain in conjunction with vector quantization is developed which result in further improvement in the recognition accuracy and dimensionality reduction.

Experimental results are given which confirm the excellent properties of the proposed

algorithms.

Finally, In chapter Seven future work is presented.

CHAPTER 2 TWO DIMENSIONAL PRINCIPAL COMPONENT ANAYLSIS (2DPCA) AND TWO-DIRECTIONAL 2DPCA (2D-2DPCA) ALGORITHMS

2.1 The 2DPCA algorithm

Recently Yang et al [15] presented the 2DPCA method for face recognition where the covariance matrix S for N training images, A_i, of dimensions *mxn* (where *i=1 to N*) is formed in 2D rather than converting each image into a one dimensional vector of size *mxn* as in [1]. This The n x n S matrix is computed as follows,

$$S = \frac{1}{N} \sum_{i=1}^{N} (A_i - \overline{A})^T (A_i - \overline{A})$$

(2.1)

where $\overline{A}$ is the mean matrix, of all the N training images.

A set of the k largest eigenvectors of the covariance matrix, $V = [V_1, V_2 .. V_k]$ of size n is obtained, so that the projection of the training images on V gives the best scatter. V is used for feature extraction for every training image A_i. The projected feature vectors $Y_1, Y_2, ... Y_k$, where

$$Y_{j,i} = A_i V_j \qquad j = 1,2,....k \quad , i=1...N \qquad (2.2)$$

are used to form a feature matrix B_i of size *mxk* for each training image A_i, where

$$B_i = [Y_{1,i}, Y_{2,i}, ... Y_{k,i}] \qquad i = 1,2,..N \qquad (2.3)$$

The tested image is projected on V, and the obtained feature matrix B_t is compared with those of the training images.

The Euclidean distances between the feature matrix of the tested image and the feature matrices of the training images are computed. The minimum distance indicates the image to be recognized.

2.2 A brief describtion of 2D-2DPCA method

The 2DPCA method [15] forms the covariance matrix, S_r , for N training images, A_i of dimensions *mxn* (where *i=1 to N*) in 2D rather than converting each image into a one dimensional vector of size *mxn* as in [3].

 The n x n S_r matrix is computed from

$$S_r = \frac{1}{N} \sum_{i=1}^{N} (A_i - \overline{A})^T (A_i - \overline{A})$$
(2.4)

where $\overline{A}$ is the mean matrix, of all the N training images.

Recently, it was shown [25] that the 2DPCA method works in the row direction of the images to form the covariance matrix, S_r . And an alternative 2DPCA method that

works in the column direction was introduced, where the covariance matrix, Sc, is calculated as follows

$$S_c = \frac{1}{N} \sum_{i=1}^{N} (A_i - \overline{A})(A_i - \overline{A})^T \qquad (2.5)$$

In the 2D2DPCA algorithm , two sets, V_r and V_c , of the k largest eigenvectors , of sizes n and m respectively, for both covariance matrices, S_r and S_c , are obtained. Where ,

$$V_c = [Vr_1, Vr_2 .. Vr_k] \quad and \quad V_c = [Vc_1, Vc_2 .. Vc_k]$$

The projection of the training images on V_r and V_c gives the best scatter. V_r and V_c are used for feature extraction for every training image A_i .The projected feature vectors $Y_{1,i}, Y_{2,i}, ... Y_{k,I}$, where

$$Y_{j,i} = V_c^T A_i V_r \qquad j = 1,2,....k \quad , i=1...N \qquad (2.6)$$

are used to form a feature matrix B_i of size kxk for each training image A_i. Where

$$B_i = [Y_{1,i}, Y_{2,i}, ... Y_{k,i}] \qquad i = 1,2,..N \qquad (2.7)$$

The tested image is projected on V_r and V_c , and the obtained feature matrix B_t is compared with those of the training images.

The distance measures, such as the Euclidean distances, between the feature matrix of the tested image and the feature matrices of the training images are computed. The minimum distance indicates the image to be recognized.

CHAPTER 3   TRANSFORM DOMAIN 2DPCA (TD2DPCA) ALGORITHM

3.1 Introduction

Recognition accuracy, storage requirements, and computational complexity are the most important performance parameters in pattern recognition and classification. Several methods have been proposed in this area. Algorithms based on one dimensional Principal Component Analysis (PCA) or Karhunen-loeve expansions [26-60] have been reported. Existing two dimensional PCA (2DPCA) spatial domain algorithms [15] has better recognition accuracy and faster implementation relative to the PCA methods. This is achieved at the expense of higher storage requirements.

In this contribution a practical, facial-recognition, transform-domain, two-dimensional, principal component analysis technique (TD2DPCA) is presented. The proposed approach is useful for large databases encountered in several areas such as security and multimedia applications. The TD2DPCA reduces the storage and computational requirements to train the system by approximately a factor of ten and two respectively, while retaining the high recognition accuracy, relative to the state of the art, 2DPCA method. In addition the recognition speed in the testing mode is reduced by approximately a factor of two. The TD2DPCA's excellent properties are confirmed experimentally for noise free and noisy images [16],[17],[18].

In section 3.2, a spatial domain-transform domain 2DPCA (SD-TD2DPCA) algorithm is presented. In the SD-TD2DPCA method the 2DPCA analysis is performed on the training images in the spatial domain. After Computing the covariance matrix of the training images in the spatial domain, the matrix is transformed to the transform domain. This is in contrast to the TD2DPCA method, introduced in section 3.3, where all the calculations are performed in the transform domain. Experimental results confirm the excellent characteristics of the proposed algorithm. Conclusions are presented in section 3.4.

3.2 The SD-TD/2DPCA algorithm

The proposed algorithm represents the signals and their covariance matrix in the transform domain. This result in considerable reduction in the coefficients required to represent the signals. Consequently the computational and storage requirements, are greatly simplified as will be shown later. The proposed Spatial Domian-Transform Domain Two Dimensional Principal Component Analysis (SD-TD/2DPCA) algorithm is described below.

3.2.1 Training mode

In the training mode the features of the data base are extracted and stored as described by steps 1 through 7, figure 3-1, 3-2.

Step 1: The covariance matrix S for the N training images is calculated using (3.1).

$$S = \frac{1}{N} \sum_{i=1}^{N} (A_i - \overline{A})^T (A_i - \overline{A}) \qquad (3.1)$$

Where A_i ($i = 1\ to\ N$) is the matrix representing the training image i in the spatial domain and $\overline{A} = \frac{1}{N} \sum_{i=1}^{N} A_i$ is the average image for the N training images.

Step 2: An appropriate transform {Tr}, that compresses the signal energy in few components, for simplicity assumed to be in the upper left corner of the transformed matrix, is applied to *S,* which yields T.

$$T = Tr\{S\} \qquad\qquad (3.2)$$

Step 3: The significant coefficients of *T* are contained in a submatrix, *S',* (upper left part of *T*) of dimension *n' x n'* . Figure 3-3 shows an example of the ratio of energy in *S'* to the energy in *T*, as a function of *n'*. *S'* is used to replace *S* in our algorithm.

Step 4: A set of *k'* eigenvectors , $V = [\ V_1,\ V_2\ ...\ V_k]$, of size *n'* corresponding to the largest *k'* eigenvalues is obtained for *S'*. Since the dimensions of *S'* is much smaller than S, *k'* is smaller than *k*.

Step 5: The same transform is applied to each image A_i of the *N* training images, yielding T_i' (*i=1 to N*).

$$T_i' = Tr\{A_i\} \qquad\qquad (3.3)$$

Step 6: The submatrix A_i' from T_i', containing most of the energy is retained (upper left part of T_i'). This submatrix is used to represent the training image. Dimensions of A_i' is *m' x n'* where $n' \leq m'$. Figure 3-4. shows an example for the ratio of energy in A_i' to the energy in T_i', as a function of *n'* for three image samples .

<u>Step 7:</u> The feature matrices of the training images B_i are calculated as follows

$$Y_{j,I} = A_i' V_j \qquad j = 1,2,....k' \ and \ i = 1,2,..N \qquad (3.4)$$

$$B_i = [Y_{1,i}, Y_{2,i}, ... Y_{k',i}] \qquad (3.5)$$

Now the feature matrix representing the training image has dimensions $(n' \ x \ k')$ where $m' \le n'$, n' is much smaller than $n \ and \ m$, and $k' < k$.

3.2.2 Testing mode

In the testing mode a facial image A_t is presented to the system to be identified. The steps are as follows

<u>Step 1</u> The same transform used in the training mode is applied to A_t which yield T_t'.

$$T_t' = Tr\{A_t\} \qquad (3.6)$$

<u>Step 2</u> The sub matrix A_t' $(m' \ x \ n')$ containing most of the energy is obtained from T_t'.

Step 3 The feature matrix B_t for the testing image is calculated as follows

$$Y_{j,t} = A_t' \ V_j \qquad\qquad j = 1,2,....k' \qquad\qquad (3.7)$$

$$B_t = [Y_{1,t} \ , Y_{2,t} \ ,...Y_{k',t} \ ] \qquad\qquad (3.8)$$

Step 4 The Euclidean distance between the feature matrix of the testing image B_t and the feature matrices of the training images B_i ($i=1$ to N) are computed. i corresponding to the minimum distance, i_{min}, is used to identify t.

Training Mode

Step 1

Input N images
A_i (i =1 to N).
A_i is mxn pixels

Compute the covariance matrix S of the input images

$$S = \frac{1}{N} \sum_{i=1}^{N} (A_i - \overline{A})^T (A_i - \overline{A})$$

Where $\quad \overline{A} = \frac{1}{N} \sum_{i=1}^{N} A_i$

Step 2

Compute **Tr{ S}**
Tr denotes appropriate
transform, such as Discrete
Cosine Transform DCT.

Step 3

Select S'

Where S' is the submatrix of
Tr{S}containing most of the energy.
S' is n'xn' pixels

Step 4

A set of k eigenvectors $V' = [V'_1, V'_2 ..V'_{k'}]$ of size n' corresponding to the largest k' eigenvalues is obtained for S'.

Step 5

Compute Tr{ A_i}
For each training image A_i
compute T'_i
Where $\quad T'_i = $ Tr{ A_i }

Figure 3-1. SD-TD/2DPCA Training mode flow-chart

(Continued next page)

Training mode (continued)

Step 6

> **Select** $A_i^{'}$
>
> where $A_i^{'}$ is the submatrix of T'_i containing most of the energy .
> $A_i^{'}$ is n'xn' pixels.

Step 7

> **Compute B_i**
>
> Where B'_i *is* the feature matrix for the training image A_i .
>
> $$Y'_{j,i} = A_i' V'_j \qquad\qquad j = 1,2,....k'$$
>
> $$B'_i = [Y'_{1,i}, Y'_{2,i}...Y'_{k',i}]$$
>
> Store B'_i corresponding to each training image A_i
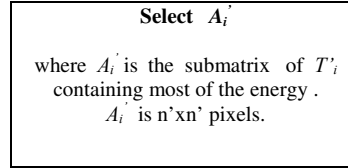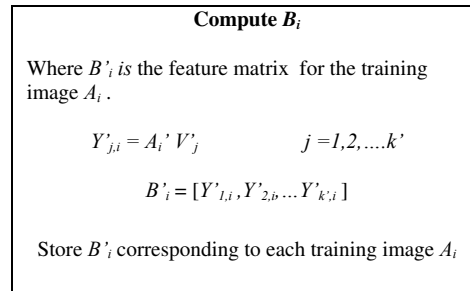
Figure 3-1.Cont. SD-TD/2DPCA Training mode flow-chart

<u>Testing Mode</u>

Step 1

Step 2

Step 3

Step 4
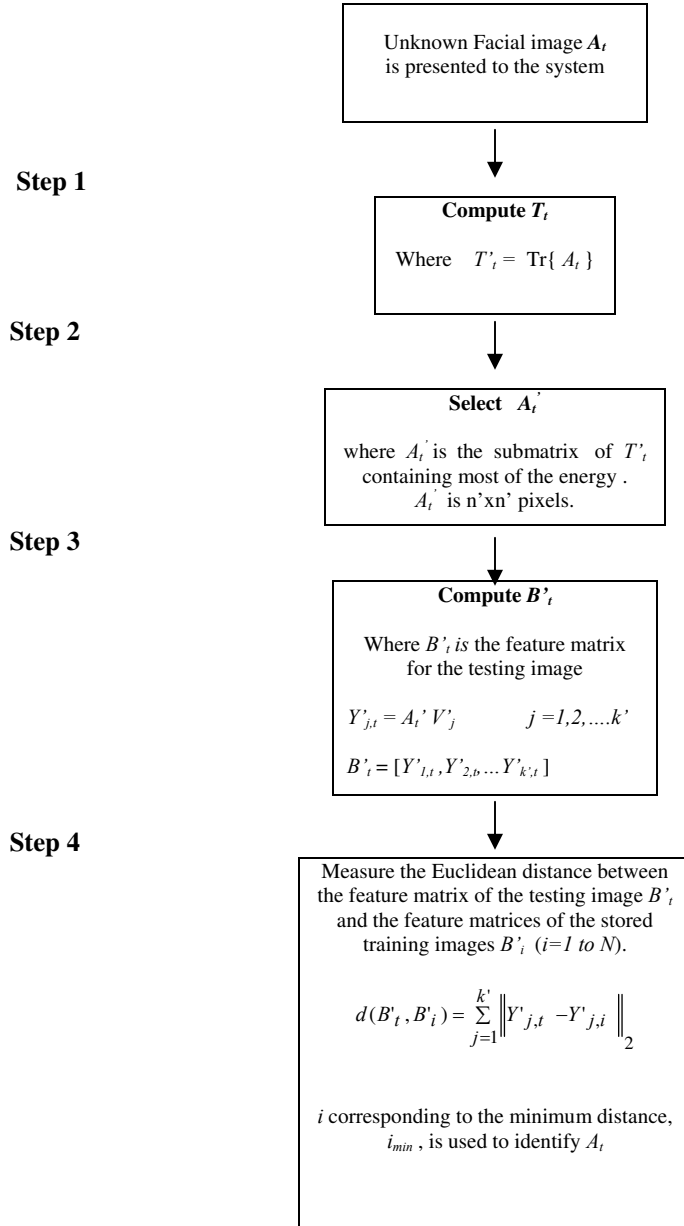
Figure 3-2

Figure 3-2: SD-TD/2DPCA Testing mode flow-chart.
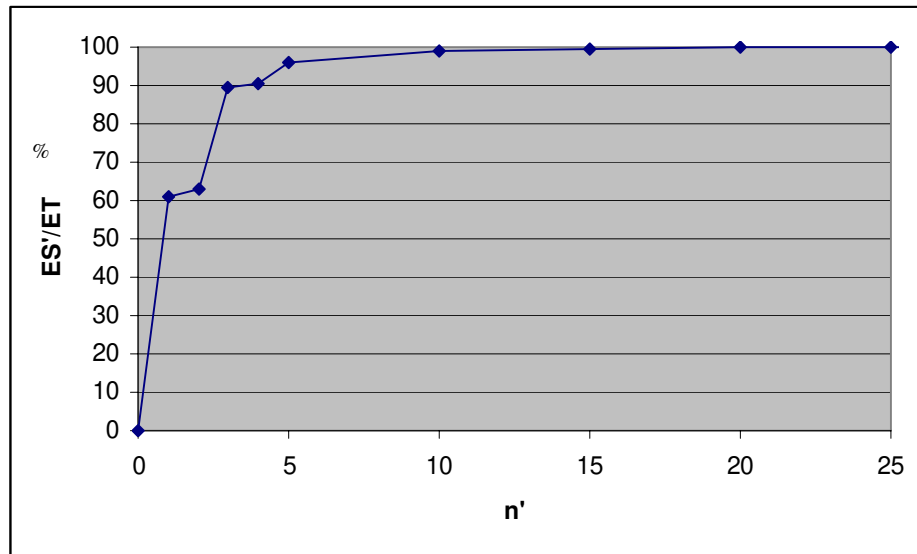
Figure 3-3 The ratio of energy in the TD2DPCA covariance matrix S' (Es') to the

energy in the covariance matrix of 2DPCA (ET) as a function of number of rows

and columns of S' (n') .
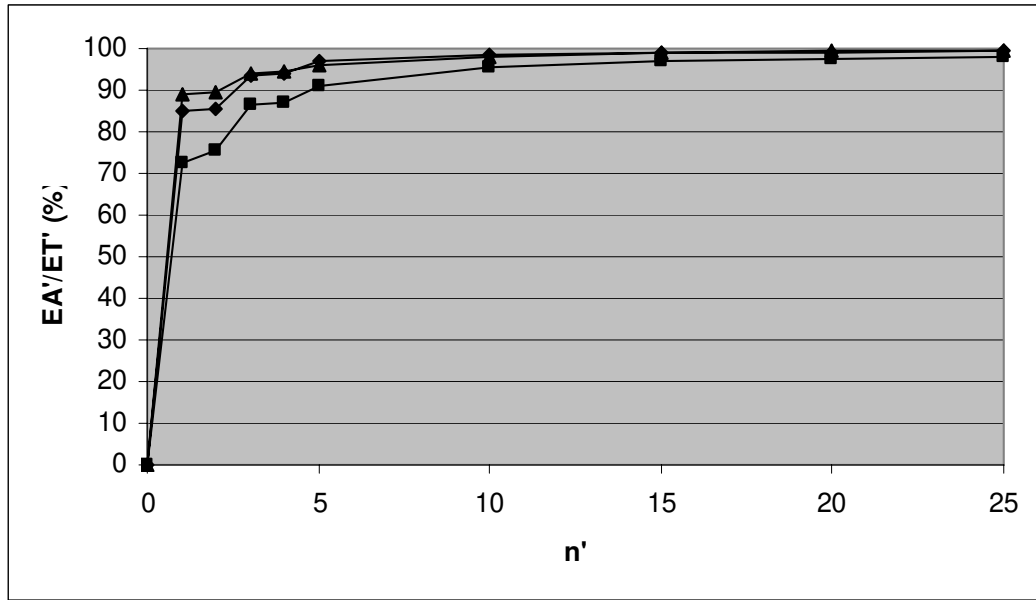
Figure 3-4 : The ratio of energy in Ai' (EA') to the energy in Ti' (ET') , as a

function   of n' for three image samples , where Ti' = DCT2(Ai) , and Ai' is a

truncated version of dimension m'xn' , obtained from Ti' .

3.2.3 Application of the proposed algorithm to face recognition, employing DCT.

The proposed algorithm, using two dimensional discrete cosine transform, was applied to the ORL database [61] , the Yale database [62] and a subset of the UMIST database [63].   The ORL database consists of 400 images of 40 individuals (10 images each), where pose and facial expressions are varying, figure 3-5.

The Yale database consists of 165 images of 15 individuals (11 images each) where illumination and face expression are varying, figure 3-6.

The subset used for the UMIST database consists of 200 images of 20 individuals where pose is varying figure 3-7.

Results are compared with those obtained using the 2DPCA and PCA methods.
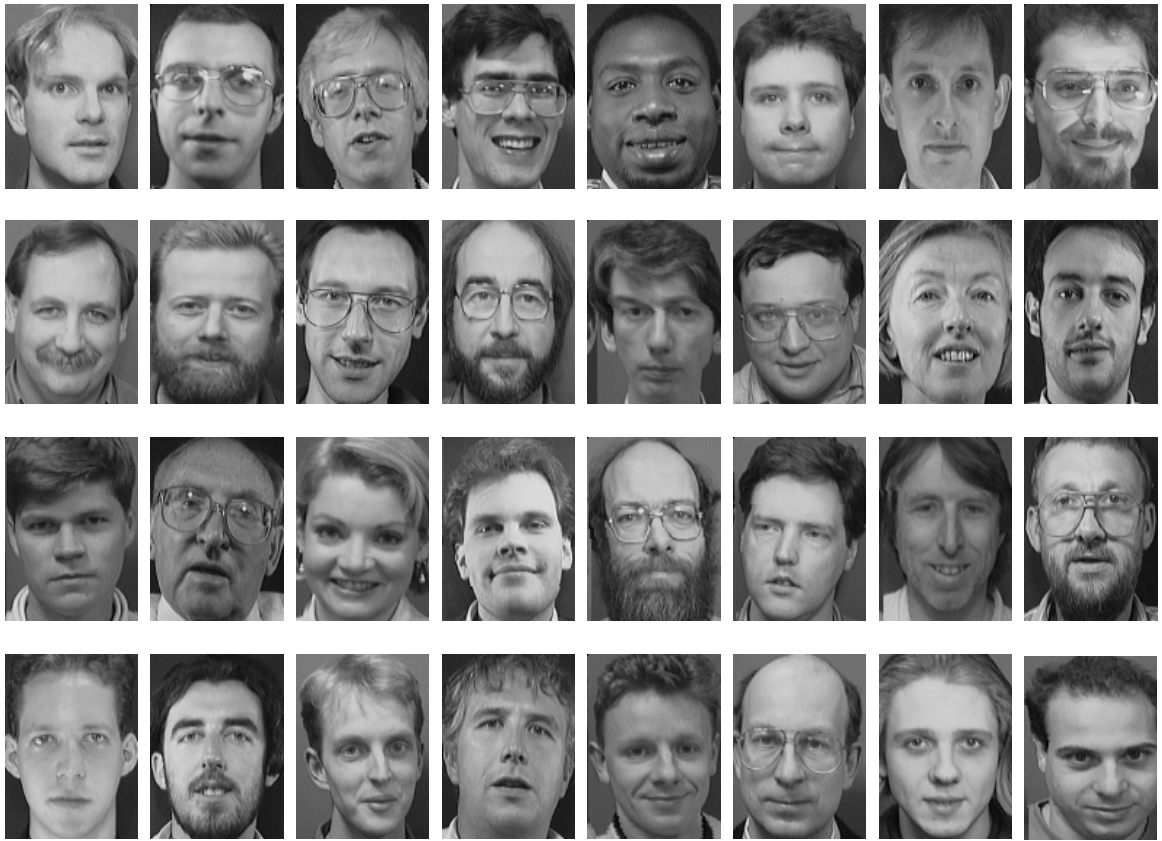
Figure 3-5: Sample of 32 individuals in the ORL database

Figure 3-6: Eleven individuals in the Yale Database
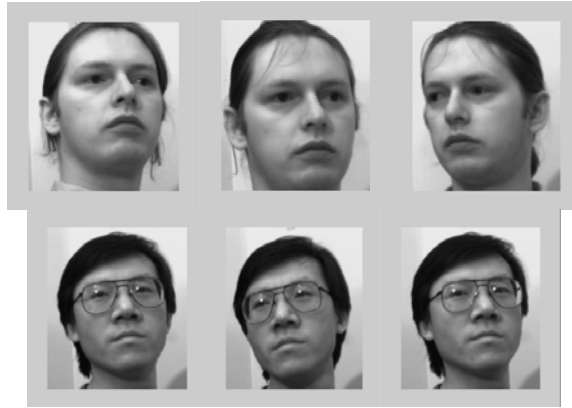
Figure 3-7: Three samples for two individuals in the UMIST database .

3.2.3.1 Experimental results using the ORL database

Two experiments have been applied to the ORL data base, where all the images are grayscale with 112 x 92 pixels each.

In the first experiment, 40 images of 40 different individuals are used for training and the remaining 360 images are used for testing .

The dimensions of the covariance matrix S for the 40 training images is 92x92. A two-dimensional DCT [64] is applied to the covariance matrix S which yields T. S' is obtained for $n'=20$. The 5 largest eigenvectors of S' corresponding to the 5 largest eigenvalues are obtained, i.e, k' is chosen to be 5 (for the 2DPCA method $k = 10$ is used for the best recognition accuracy). T_i' (i =1 to 40) are obtained. Then A_i' of dimensions 20x20   (i =1 to 40) are determined,

 i.e , $m'xn'$= 20x20 in our experiment.

The feature matrices for all the training images are obtained. The procedure in section 3.2.2 is followed for the 360 testing images.

Table 3-1 gives the recognition accuracy for the proposed technique as well as 2DPCA and PCA methods.

In the second experiment 5 images per class are used for training and the remaining 200 images are used for testing. The Dimensions of S' and A_i' are the same as in the first experiment. k' is chosen equal to 5. For the 2DPCA method, k equals 10 is used for the best recognition accuracy. Results using the proposed algorithm, 2DPCA, and PCA techniques are listed in Table 3-1,

Table 3-1 shows that the proposed algorithm yields similar recognition accuracy as the 2DPCA method.

Table 3-2 illustrates the computational complexity, in terms of the number of multiplications [64], and the storage requirements, in terms of the dimensions of the feature matrix. It is seen that, for the TD/2DPCA, the amount of storage is drastically reduced (by approximately 90%), while the computational complexity is lower, compared with one of the best available algorithm, 2DPCA. This is accomplished while maintaining the same level of recognition accuracy. It can be easily shown that the excellent properties of the new technique are maintained for the facial databases in section 3.2.3.2, and 3.2.3.3 and others.

3.2.3.2 Results for Yale database

In this experiment the dimensions of the images used are 243x320. Five images per class are used for training and the remaining images are used for testing. The Dimensions of S' is ($50x50$), and the dimension of A_i' is ($50x50$). k' is chosen equal to 5. For the 2DPCA method, k equals 20 is used for the best recognition accuracy. Results are listed in Table 3-3, Where it shows that the proposed algorithm gives similar recognition accuracy as the 2DPCA method with a feature matrix per image much more reduced in size (approximately 95%).

The computation requirements in terms of the number of multiplications during the training and the testing modes are significantly reduced.

3.2.3.3   Results for UMIST database

In this experiment each image is cropped and scaled to 185x160. Three images per class are used for training and the remaining images are used for testing. The Dimensions of S' is ($40x40$), and the dimension of A_i' is ($40x40$). k' is chosen equal to 5. For the 2DPCA method, k equals 15 is used for the best recognition accuracy. Results are listed in table IV. where it confirms that the proposed algorithm gives similar recognition accuracy as the 2DPCA method with a reduced feature matrix per image and lower computation requirements.
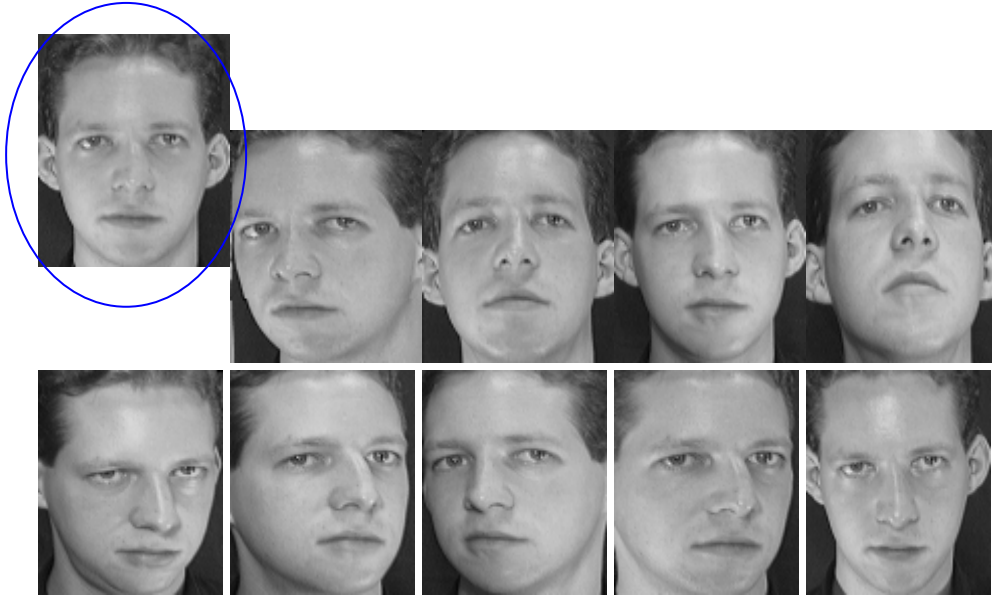
Figure 3-8 : Training with one image per individual in the ORL database and testing

with the remaining images

Figure 3-9 : Training with five images per individual in the ORL database and testing

with the remaining images

Figure 3-10 : Training with five images per individual in the Yale database and testing with the remaining images

Table 3-1 : Recognition accuracy for experiments, I and II, on the ORL database

using SD-TD/2DPCA, 2DPCA and PCA methods

Method	Recognition accuracy for experiment I	Recognition accuracy for experiment II
SD-TD/2DPCA	73.61 %	92.0 %
2DPCA	72.77 %	91.0 %
PCA	62.77 %	83.5 %

Table 3-2 : Dimensions of the feature matrix and number of multiplications required
for N training images in the ORL database, for experiment I, and II.

	SD-TD/2DPCA	2DPCA
Dimensions of feature matrix per image	(20x5)	(112x10)
Storage requirements for N images	(20x5) x N	(112x10) x N
# of multiplications required for training	47104+57344xN	103040xN
# of multiplications required for testing	57344	103040

Table 3-3 : Recognition accuracy, storage requirements and computational complexity for the experiment on the Yale database.

	SD-TD/2DPCA	2DPCA
Recognition accuracy	78.8 %	77.7 %
Dimensions of feature matrix per image	(50 x 5)	(243 x 20)
Storage requirements for N images	(50x5) x N	(243x20) x N
# of multiplications required for training	248832+262144 x N	1555200 x N
# of multiplications required for testing	262144	1555200

Table 3-4 : Recognition accuracy and dimensions of feature matrix per image for the experiment on the subset of the UMIST database.

	SD-TD/2DPCA	2DPCA
Recognition accuracy	83.57 %	80 %
Dimensions of feature matrix per image	(40 x 5)	(185 x 15)

3.3   TD2DPCA recognition and classification algorithm:

A novel facial recognition two-dimensional principal component analysis technique in the transform domain (TD2DPCA) which lends itself to practical applications is presented. The facial images are transformed using an appropriate domain that compacts the image energy. Then, the dominant principal components of the compacted images are obtained to represent the compacted images. This is shown to simultaneously achieve two desirable objectives, namely, the reduction of storage and computational requirements, and the improved accuracy due to the use of the principal component representation of the transformed images. The TD2DPCA is implemented using DCT and applied to several databases of facial images. It is shown that the new technique retains the high accuracy of recently proposed methods [15], namely 2DPCA, while reducing the storage requirements and the computational complexity by approximately 90 % and 50 % repectively.

The proposed algorithm is described below.

3.3.1   Training mode:

In the training mode, the features of the facial images in the data base are extracted and stored as described by steps 1 through 5.

Step 1: A suitable transform (Tr) is applied to each spatial training image A_i (i=1 to N), yielding T_i (i=1 to N), thus

$$T_i = T\{A_i - \bar{A}\} \qquad\qquad (3.8)$$

Where $\bar{A}$ is the mean image, of all the N training images and A_i is of dimensions *mxn*.

Step 2: The covariance matrix *S'* for the N training images is calculated as follows

$$S = \frac{1}{N}\sum_{i=1}^{N}(T_i)^T(T_i) \qquad\qquad (3.9)$$

Where $(T_i)^T$ is the transpose of (T_i) .

The transform is chosen such that most of the energy in S is concentrated in a much smaller submatrix , S ', (upper left corner of S) of dimensions n' x n' , where n' is much smaller than m and n. Then, S' is used to replace S in our algorithm.

Step 3: The set of dominant k' eigenvectors, $V = [V_1 , V_2 \dots V_{k'}]$, corresponding to the largest k' eigenvalues of S' is obtained. V_j is an n'x1 vector, j =1 to k'.

Step 5: The feature matrix, B_i , corresponding to A_i', in the dominant principal components space V , is calculated from

$$B_i = [Y_{1,i}, Y_{2,i}, \dots Y_{k',i}] \qquad\qquad (3.10)$$

where

$$Y_{j,i} = A_i' \, V_j \qquad j = 1,2,\dots k' \text{ and } i = 1,2,..N \qquad (3.11)$$

The B_i matrices, of dimensions *n'xk'*, are stored.

3.3.2 Testing mode:

In the testing mode, a facial image A_t is presented to the system to be identified. This is accomplished as follows.

Step 6: The same transform used in the training mode is applied to A_t, as described before, which yields the sub matrix A_t'

Step 7: The feature matrix, B_t, for the unknown image, is calculated as given in Step 5 in the training mode, which yields

$$B_t = [Y_{1,t}, Y_{2,t}, \dots Y_{k',t}] \tag{3.13}$$

Where

$$Y_{j,t} = A_t' V_j \qquad j = 1, 2, \dots k' \tag{3.14}$$

Step 9: The Euclidean distance $d(B_t, B_i)$ between the feature matrix B_t and the feature matrix B_i, for all i, is computed from

$$d(B_t, B_i) = \sum_{j=1}^{k} \left\| Y_{j,t} - Y_{j,i} \right\|_2 \tag{3.15}$$

Where $\left\| Y_{j,t} - Y_{j,i} \right\|_2$ denotes the distance between the two vectors $Y_{j,t}$ and $Y_{j,i}$ expressed as the sum of the squares of the differences of the corresponding elements in the two vectors.

i which corresponds to the minimum distance, i_{min}, is used to identify t.

3.3.3 Experimental Results:

Sample results employing the proposed technique to the facial recognition of some of the existing databases, ORL and Yale [61, 62], are given for illustration. The ORL images are greyscale with 112 x 92 pixels each. The Yale images are greyscale with 243x320 pixels each.

3.3.3.1 Experimental Results Employing the ORL database

Two experiments, I and II, are conducted using the ORL database. In experiment I, 40 images of 40 different individuals are used for training and the remaining 360 images are used for testing. In experiment II, 5 images per individual are used for training and the remaining 200 images are used for testing. The proposed TD2DPCA, and two existing methods, namely, 2DPCA and PCA are used. The Results are given for comparison in Tables 3-5 and 3-6.

3.3.3.2 Experimental Results Employing the Yale database

In experiment III, our technique as well as the existing 2DPCA approach is applied to the Yale database. Five images per class are used for training and the remaining images are used for testing. The Results are summarized in Table 3-7.

Tables 3-5 and 3-7 show that, the new technique achieves the same recognition accuracy as the spatial 2DPCA.

Also, from Tables 3-6 and 3-7, the computational requirements employing the TD2DPCA, described by the number of multiplications, are reduced by a factor of at least 2 relative to the 2DPCA method. In addition, the storage requirements are reduced by a factor 10 for the data used. Considerable reduction in computational and storage requirements employing TD2DPCA was consistently obtained for other databases, namely the FERET, without any loss of recognition accuracy.

3.3.3.3 Experimental Results on the ORL database in the presence of noise

In this experiment the TD2DPCA algorithm is tested in the presence of noise, namely the salt and pepper noise and the white Gaussian noise.

Salt and pepper noise is a form of noise typically seen on images. It represents itself as randomly occurring white and black pixels. Usual and effective noise reduction method for this type of noise involves the usage of median filter.

Gaussian noise is noise that has a probability density function (abbreviated pdf) of the normal distribution (also known as Gaussian distribution). In other words, the values that the noise can take on are Gaussian distributed. It is most commonly used as additive white noise to yield additive white Gaussian noise (AWGN).

Tables 3-5and 3-7 show that the proposed TD2DPCA algorithm yields good recognition accuracy compared to the 2DPCA method. This recognition accuracy is maintained with up to 40% salt and pepper noise added to the tested images , and up to 80 gray level white Gaussian additive noise added to the tested images figure 3-10.

Table 3-5 :   Recognition accuracy results employing TD2DPCA, 2DPCA and PCA
methods for , Experiments I and II, on the ORL database.

Method	Recognition accuracy for experiment I	Recognition accuracy for experiment II
TD2DPCA	73.61 %	92.0 %
2DPCA	72.77 %	91.0 %
PCA	62.8 %	83.5 %

Table 3-6 : Dimensions of feature matrix and number of multiplications required for

N training images for, ORL database, experiments I and II.

	TD2DPCA	2DPCA
Dimensions of feature matrix per image	(20x5)	(112x10)
Storage requirements for N images	(20x5)xN	(112x10)xN
# of multiplications in the training mode	(57344xN)*	103040xN
# of multiplications in the testing mode	57344	103040

* Approximate number of multiplications required to compute the T_i's and the B_i's .

Table 3-7 : Recognition accuracy, storage requirements and computational
complexity for experiment III, on the Yale database.

	TD2DPCA	2DPCA
Recognition accuracy	78.8 %	77.7 %
Dimensions of feature matrix per image	(50x5)	(243x20)
Storage requirements for N images	(50x5)xN	(243x20)xN
# of multiplications in the training mode	(262144xN)*	1555200XN
# of multiplications in the testing mode	262144	1555200

* Approximate number of multiplications required to compute the T_i's and the B_i's .

Figure 3-11 : Thirty-two noisy facial images

3.4 Conclusions:

A 2DPCA technique, in the transform domain, is presented for classification and recognition. It possesses attractive properties, namely, reduced storage requirements and computational complexity while yielding high accuracy. The application of the proposed method to the important problem of facial recognition is given. The fast two dimensional DCT is employed to implement the algorithm. Results for the ORL, Yale, and UMIST databases are given which confirm the excellent properties of the proposed approach.

It is worthwhile to note that the TD2DPCA approach is applicable to the classification and recognition of other types of signals.

CHAPTER 4   TRANSFORM DOMAIN TWO DIRECTIONAL TWO DIMENSIONAL 2DPCA (TD/2D2DPCA) AND MODIFIED TD2DPCA (M-TD2DPCA) ALGORITHMS

4.1 Introduction

In 1991 Turk and Pentland [3] developed the Eigenfaces method based on the principal component analysis (PCA) or Karhunen-loeve expansion [1,2] for face recognition. The main idea of PCA is to find the vectors that best account for the distribution of face images within the entire image space. This technique yielded good accuracy despite variations in the pose, illumination and face expressions. In 2004 Yang et al [15] proposed a two dimensional PCA technique (2DPCA) that processes images in 2D rather than in 1D as the eigenfaces (PCA) method [3]. The 2DPCA technique is working in the row direction of images. The 2DPCA technique has shown a higher recognition accuracy with a faster computational speed [15]. However, the storage requirements are more than that required by the PCA method.

Recently the Two-Directional Two-Dimensional PCA (2D2DPCA) method was introduced, that simultaneously considers the row and column directions of the image matrix [25]. This method, compared to the 2DPCA, reduced the storage requirements while maintaining the same recognition accuracy [19].

In chapter three, we introduced the TD2DPCA algorithm that reduced the storage requirements by almost 90 percent and reduced the computational speed by a factor of two, compared to the 2DPCA method, while maintaining the same high accuracy.

In this Chapter a Transform Domain Two-Directional Two-Dimensional Principal Component Analysis (TD/2D2DPCA) algorithm applied to facial recognition is presented. This algorithm has attractive properties with respect to storage and computational requirements, while maintaining the high recognition accuracy achieved before. The storage requirements are reduced by more than 95 percent compared to the spatial 2DPCA method and 75 percent compared to the TD2DPCA algorithm. The computational speed, compared to the spatial 2D2DPCA method, is reduced to a great deal. Experimental results obtained by applying the new algorithm to the ORL and Yale databases confirm these excellent characteristics.

Section 4.2 presents the proposed algorithm. and discusses the results obtained by testing the new algorithm on the ORL and Yale databases.

In Section 4.3 we present the Modified- TD2DPCA (M-TD2DPCA) algorithm [20]. The M-TD2DPCA method presents a new way of calculating the covariance matrix of the training images. This approach reduced the computational complexity required to calculate the covariance matrix while maintaining good accuracy. Experimental results confirm this. Section 4.4 presents the conclusions.

4.2   The TD/2D2DPCA algorithm

The TD/2D2DPCA algorithm deals with the coefficients representing the images in the transform domain, in both the row and column directions (the TD2DPCA algorithm was working on the row direction only).   This approach maintains the relation between these coefficients, which yields a better representation of the images and their covariance matrix, where energy is compacted in as small number of coefficients. This reflects into a considerable reduction in the coefficients required to represent the images (feature matrices). Consequently, the computational and storage requirements are further simplified, compared with the excellent TD2DPCA.   The algorithm is described in the following sections.

4.2.1   Training mode

In the training mode, the features of the data base are extracted and stored as described by steps 1 through 5.

<u>Step 1:</u> The suitable transform (*Tr*), such as DCT, is applied to each *m x n* image A_i of the *N* training images, yielding T_i (*i=1 to N*).

$$T_i = Tr\{A_i - \bar{A}\} \qquad\qquad (4.1)$$

Where $\bar{A}$ is the mean matrix, of all the N training images.

<u>Step 2:</u> The covariance matrices, S'_r and S'_c ,for the N training images are calculated as follows.

$$S_r = \frac{1}{N} \sum_{i=1}^{N} (T_i)^T (T_i)$$ (4.2)

$$S_c = \frac{1}{N} \sum_{i=1}^{N} (T_i) (T_i)^T$$ (4.3)

The transform is chosen such that most of the energy in S_r and S_c is concentrated in a much smaller submatrices , S'_r and S'_c , (upper left corner of S_r and S_c) of dimensions n' x n' .

<u>Step 3:</u> V_r *and* V_c are obtained using the sets of k' eigenvectors ,of size n' , corresponding to the largest k' eigenvalues for S'_r and S'_c , respectively.

Step 4: The feature matrices of the training images B_i are calculated

$$B_i = [Y_{1,i}, Y_{2,i}, \ldots Y_{k',i}] \hspace{3cm} (4.4)$$

where

$$Y_{j,i} = V_c^T A_i' V_r \hspace{1cm} j = 1,2,\ldots k' \hspace{0.3cm} and \hspace{0.3cm} i = 1,2,\ldots N \hspace{1cm} (4.5)$$

The B_i matrices are stored .

It is worthwhile to note that the feature matrix representing the training image has dimensions much lower than those obtained using the spatial 2D2DPCA method (*n'<<n , and now k' is smaller*).

4.2.2 Testing mode

In the testing mode a facial image A_t is presented to the system to be identified. The following steps are followed

<u>Step 1</u> The same transform used in the training mode is applied to A_t.

<u>Step 2</u> The sub matrix A_t' containing the significant coefficients is obtained (dimension $n' x n'$)

<u>Step 3</u> The feature matrix B_t for the testing image is calculated

$$B_t = [Y_{1,t}, Y_{2,t}, ... Y_{k',t}] \qquad\qquad (4.6)$$

where

$$Y_{j,t} = V_c^T T_t' V_r \qquad\qquad j = 1,2,....k' \qquad (4.7)$$

<u>Step 4</u> Distance measures, such as the Euclidean distances, between the feature matrix of the testing image and the feature matrices of the training images are measured . The minimum distance represents the image to be identified.

4.2.3   Experimental Results and Analysis

The TD/2D2DPCA algorithm was tested using the ORL and Yale datasets [8,9]. Results are compared with the TD2DPCA and 2DPCA methods.

Two experiments, I and II, have been applied to the ORL dataset, where all the images are grayscale with 112 x 92 pixels each.

In experiment I, 40 images of 40 different individuals are used for training, and the remaining 360 images are used for testing. A two-dimensional discrete cosine transform (DCT) is applied to the N training images. The dimensions that give comparable results with the 2DPCA method, of A'_i and the covariance matrices, S'_r and S'_c, are $20x20$. The 5 largest eigenvectors, of S_r and S_c corresponding to the 5 largest eigenvalues are obtained. In our approach k' of only 5 was needed relative to $k = 10$ in other approaches.

The feature matrices for all the training images are obtained and stored using (eq. 4.4) and (eq. 4.5).

 The procedure in section 4.2.2 is followed for the 360 testing images. Results are listed in Tables 4-1 and 4-2.

In experiment II, on the ORL database 5 images per individual are used for training, and the remaining 200 images are used for testing. The Dimensions of A'_i and , S'_r and S'_c, are the same as in the first experiment.

Results are listed in Tables 4-1 and 4-2.

In the experiment applied to the Yale database the dimensions of the images used are 243x320. Five images per individual are used for training and the remaining images are used for . For satisfying accuracy, the dimensions of S_r and Sc are (*50x50*), and the dimensions of T_i' is (*50x50*). k' is chosen equal to 5. For the 2DPCA method, k equals 20 is used for the best recognition accuracy. Results are listed in Tables 4-3.

Tables 4-1 and 4-2 show that the proposed algorithm yields good recognition accuracy compared to the TD2DPCA and 2DPCA methods.

Table 4-2, 4-3 illustrates the storage requirements, in terms of the dimensions of the feature matrix. It is seen that, for the TD2D2DPCA, the amount of storage is reduced by approximately 95 %, compared to the 2DPCA method and 75% compared to the TD2DPCA algorithm. Also it is worthwhile to note that the computational requirements in the training and testing modes compared to number of multiplications are reduced to a great deal.

Table 4-1 : Recognition accuracy for experiment I and II on ORL dataset
using,TD/2D2DPCA,TD2DPCA, 2D2DPCA and 2DPCA

Method	Recognition accuracy for experiment I	Recognition accuracy for experiment II
TD/2D2DPCA	73.80 %	92.20 %
TD/2DPCA	73.61 %	91.94 %
2D2DPCA	73 %	90.5 %
2DPCA	72.77 %	91.00 %

Table 4-2: Dimensions of feature matrix and number of computations required for the

training and testing modes on ORL dataset, for experiments I, II.

	TD2DPCA	2DPCA	TD/2D2DPCA
Dimensions of feature matrix per image	(20x5)	(112x10)	(5x5)
Storage requirements for N images	(20x5)xN	(112x10)xN	(5x5)xN
# of multiplications required for the training mode	(57344xN)*	103040xN	(57844xN)*
# of multiplications required for the testing mode	57344 **	103040	57844 **

* Approximate number of multiplications required (including those needed to
compute the transform of the N images)
** Including number of multiplications required to compute the transform of the
tested image

Table 4-3 : Recognition accuracy for experiment on Yale dataset employing,TD/2D2DPCA, TD2DPCA and 2DPCA

	TD2DPCA	2DPCA	TD/2D2DPCA
Recognition accuracy	78.8 %	77.7 %	78.8 %
Dimensions of feature matrix per image	(50 x 5)	(243 x 20)	(5 x 5)
Storage requirements for N images	(50x5)xN	(243x20)xN	(5x5)xN
# of multiplications for training mode	(262144xN)*	1555200XN	(263394xN)*
#of multiplications for testing mode	262144 **	1555200	263394 **

* Approximate number of multiplications required (including those needed to compute the transform of the N images)
** Including number of multiplications required to compute the transform of the tested image

4.3 M-TD2DPCA

4.3.1 Introduction

A modified transform-Domain Two dimensional Principal Component Analysis M-TD2DPCA algorithm is presented. The proposed algorithm maintains the improved performance of the TD2DPCA technique while considerably reducing the computational requirements in the training mode. This has been confirmed experimentally.

An alternative formulation of the images autocorrelation matrix is introduced which further reduces the computational requirements to obtain the images autocorrelation matrix by a factor of N, where N is the number of images in the database, relative to the TD2DPCA.

4.3.2   The Proposed algorithm

The algorithm is described in the following section.

4.3.2.1 Training mode

In this mode, the system is trained with a set of N facial images . The features of the data base are extracted and stored as described by steps 1 through 5.

<u>Step 1:</u> The suitable transform (Tr) is applied to each $m \: x \: n$ image A_i of the N training images, yielding T_i ($i=1 \: to \: N$).

<u>Step 2:</u> The transform is chosen such that the significant coefficients of T_i are contained in a submatrix, A_i', (upper left part of T_i) of dimension $n' \: x \: n'$. Thus A_i' is used to replace A_i in our algorithm.

<u>Step 3:</u> The covariance matrix S' for the N training images is calculated as follows.

$$S = \frac{1}{N} \sum_{i=1}^{N} (G_i)^T (G_i) \qquad\qquad (4.8)$$

Where $G_i = \frac{1}{N} \sum_{i=1}^{N} (T_i) \qquad\qquad (4.9)$

The transform is chosen such that most of the energy in S is concentrated in a much smaller submatrix , S ', (upper left corner of S) of dimensions n' x n' , where n' is much smaller than m and n. Then, S' is used to replace S in our algorithm.

<u>Step 4:</u> A set of k' eigenvectors, V_j $(j=1$ to $k')$ corresponding to the largest k' eigenvalues of S' is obtained. V_j is an n' x 1 vector.

<u>Step 5:</u> The feature matrix, B_i (i=1 to N), for each training image is calculated.

$$B_i = [Y_{1,i}, Y_{2,i}, ... Y_{k',i}] \qquad\qquad (4.10)$$

where

$$Y_{j,i} = T_i' V_j \quad j = 1,2,....k' \ and \ i = 1,2,..N \qquad\qquad (4.11)$$

The B_i matrices (i=1 to N) are stored .

4.3.2.2 Testing mode

In the testing mode a facial image A_t is presented to the system to be identified. The following steps are followed

Step 1 The same transform used in the training mode is applied to A_t which yield T_t.

Step 2 The sub matrix T_t' containing the significant coefficients is obtained (dimension n' x n')

Step 3 The feature matrix B_t for the testing image is calculated

$$B_t = [Y_{1,t}, Y_{2,t}, ... Y_{k,t}]$$ (4.12)

where

$$Y_{j,t} = T_t' V_j \qquad j = 1,2,....k$$ (4.13)

Step 4 Distance measures, such as the Euclidean distances, between the feature matrix of the testing image and the feature matrices of the training images are measured. The stored image that produces the minimum distance   represents the image to be identified.

4.3.3 Experimental Results and Analysis

The proposed algorithm was applied to the ORL and Yale databases. Results are compared with the TD2DPCA and 2DPCA methods.

Two experiments have been applied to the ORL database, where all the images are grayscale with 112 x 92 pixels each.

In the first experiment, 40 images of 40 different individuals are used for training and the remaining 360 images are used for testing. A two-dimensional discrete cosine transform (2D-DCT) is applied to the N training images. The dimensions of T'_i and the covariance matrix S' are *20x20*. The 5 eigenvectors of S' corresponding to the 5 largest eigenvalues are obtained. In our approach k' of only 5 was needed relative to k = *10* in 2DPCA method.

The feature matrices for all the training images are obtained using (4.10) and (4.11).

 The procedure in section 4.3.2.2 is followed for the 360 testing images. Results are listed in Tables 4-4 and 4-5.

In the second experiment on the ORL database 5 images per class are used for training, and the remaining 200 images are used for testing. The Dimensions of T'_i and S are the same as in the first experiment. Results are listed in Tables 4-4 and 4-5.

In the experiment applied to the Yale database the dimensions of the images used are 243x320. Five images per class are used for training and the remaining images are used for testing. The Dimensions of S is (*50x50*), and the dimension of T_i' is (*50x50*). K' is chosen equal to 5. For the 2DPCA method, k equals 20 is used for the best recognition accuracy. Results are listed in Table 4-6.

Tables 4-4 and 4-6 show that the proposed algorithm maintains the good recognition accuracy of the TD2DPCA and 2DPCA methods.

Tables 4-5, 4-6 illustrate the storage requirements, in terms of the dimensions of the feature matrix. It is seen that, for the Proposed M-TD2DPCA and TD2DPCA, the amount of storage is drastically reduced (by approximately 90%), compared with, 2DPCA . Also it is worthwhile to note that the computational requirements for the covariance matrix in the training mode employing the proposed algorithm is reduced by a factor of N compared to the TD2DPCA algorithm. This reduction is particularly important when the number of images N in the databases is large. This is frequently encountered in practice.

Table 4-4 : Recognition accuracy for experiment I and II on ORL database using,

M-TD2DPCA, TD2DPCA, and 2DPCA methods

Method	Recognition accuracy for experiment I	Recognition accuracy for experiment II
M-TD2DPCA	73.05 %	91.68 %
TD2DPCA	73.61 %	92.0 %
2DPCA	72.77 %	91.0 %

Table 4-5 : Dimensions of feature matrix and number of computations required for
the training and testing modes on ORL database, for experiments I , II.

	M-TD2DPCA	TD2DPCA	2DPCA
Dimensions of feature matrix per image	(20x5)	(20x5)	(112x10)
Storage requirements for N images	(20x5)xN	(20x5)xN	(112x10)xN
# of multiplicat<u>n</u> required for the training mode	(57344xN)** + (103040) * **[independent of N]**	(57344xN)**+ (103040**xN**)*	947968xN +103040xN
# of multiplicat<u>n</u> required for the testing mode	57344	57344	103040

* Approximate number of multiplications required to compute the covariance matrix
** Approximate number of multiplications required to compute the transform of the N images

Table 4-6 : Recognition accuracy, dimensions of feature matrix and number of

computations required for the training and testing modes for experiment on

Yale database employing, M-TD2DPCA, TD2DPCA, and 2DPCA methods

	M-TD2DPCA	TD2DPCA	2DPCA
Recognition accuracy	78.8 %	78.8 %	77.7 %
Dimensions of feature matrix per image	(50 x 5)	(50 x 5)	(243 x 20)
Storage requirements for N images	(50x5)xN	(50x5)xN	(243x20)xN
# of multiplicatn required for the training mode	(262144xN)**+ (1555200)* [independent of N]	(262144xN)**+ (1555200**XN**) *	18895680xN+ 1555200XN
# of multiplicatn required for the testing mode	262144	262144	1555200

*Approximate number of multiplications required to compute the covariance matrix
** Approximate number of multiplications required to compute the transform of the N images

4.4 Conclusions

In this contribution a TD/2D2DPCA algorithm is presented for facial recognition. It is shown that the new technique retains the high recognition accuracy of the 2DPCA and TD2DPCA methods while reducing the storage requirements by 95 percent compared to the 2DPCA and 75 percent compared to TD2DPCA. It is worthwhile to note that the computational speed has been reduced by a great deal relative to 2DPCA algorithm. Experimental results confirm these excellent characteristics.

In addition a modified transform domain two dimensional principal component analysis (M-TD2DPCA) algorithm is described and applied to facial recognition. The proposed technique, while maintaining the excellent characteristics of the recently reported TD2DPCA approach , it requires much fewer computations to obtain the images autocorrelation. matrix. Sample results and performance comparison with existing techniques are given which confirm the improved performance of the M-TD2DPCA.

CHAPTER 5 TRANSFORM DOMAIN TWO DIMENSIONAL PRINCIPAL COMPONENT ANALYSIS IN CONJUNCTION WITH VECTOR QUANTIZATION (TD2DPCA/VQ)

5.1 Introduction

In this chapter we present an algorithm that uses the TD2DPCA analysis in conjunction with vector quantization (TD2DPCA/VQ). This method benefits from both the TD2DPCA analysis and vector quantization. TD2DPCA analysis results in considerable reduction in the coefficients required to represent the images. Consequently, the computational and storage requirements are greatly simplified. Vector quantization is an efficient way to group vectors representing different signals.

A technique is developed which combines VQ [73] with transform domain principal component representation in the training mode. This results in drastically increasing the speed of recognition in the testing mode. The algorithm is described in the following sections.

In section 5.1 we present a brief description of VQ . In section 5.2 a tree structure TD2DPCA/VQ-1 algorithm is presented. Experiment results confirm the excellent properties of the proposed techniques. Section 5.3 presents a TD2DPCA/VQ-2 algorithm [22] used when more than one image per individual is employed to train the system . Conclusions are discussed in section 5.4.

5.1.1 Classification Decision Tree

Decision trees are considered one of the most popular classification approaches due to their accuracy and simplified computational properties [84, 89, 92]. Moreover, they are fast in training [65]. They are capable of performing non-linear classification [74] and they do not rely on statistical distribution. This has yielded successful applications in many fields such as remote sensing data [83].

The tree is composed of a root node, intermediate nodes and terminal nodes. The data set is classified at each node according to the decision framework defined by the tree [23]. It starts with a coarse classification, and then followed by a fine classification where finally each group contains only one signal.

Classification decision trees have the advantages of employing more than one feature. Each feature provides partial information about the signal. The combination of such features can be used to obtain accurate recognition decision [91]. There are more than one decision tree that can be used for a given example. But the smaller the decision tree, the better it becomes [90].

A large number of methods have been proposed in the literature for the design of the classification tree. Classification and Regression Trees (CART) is one of the approaches that have achieved high popularity [72]. It was developed during the years 1973 through 1984 [4]. It has the advantage of constructing classification regions with sharp corners. However, it is computationally expensive [76]. In this approach,

splitting continues until terminal nodes are reached. Then, a pruning criterion is used to sequentially remove splits [74]. Pruning can be implemented by using different data than those used in training. The main advantages of pruning is reducing the size of the decision tree [93] and hence reducing the classification error [72] and avoiding both overfitting and underfitting.

Most of the pruning methods proposed in the literature are based on removing some of the nodes of the tree. Kijsirikul et al. [77] have introduced a pruning method which employs neural networks, trained by backpropagation algorithm, to give weights to nodes according to their significance instead of completely removing them.

5.1.2 Vector quantization

Vector quantization is a powerful technique for data compression. Recently, it has been used to simplify image processing tasks such as halftoning, edge detection [58], image recognition [77] and enhancement classification.

Vector Quantization and Classification can be combined because both techniques can be designed and implemented using methods from statistical clustering and classification trees [82]. They can be implemented with a tree structure that greatly reduces the encoding complexity [90]. It has been shown that if an optimal vector quantizer is obtained, under certain design constraints and for a given performance objective, no other coding system can achieve a better performance. This approach

70

has several advantages in coding and in reducing the computation in speech recognition [38].

One of the most widely used algorithms is the Lloyd algorithm. It improves a codebook by alternately optimizing the encoder for the decoder and the decoder for the encoder [72].

Linear Vector Quantization (LVQ) has been used to classify the various kinds of signals. The reasons to use the LVQ are that it can process the unsupervised classification and treat many input data with small computational burden [71]. In other words, it can treat high dimensional input and has a simple learning structure.

A LVQ is composed of two layers; a competitive layer that learns the feature space topology and the linear layer that transforms classes into target classes. It can be used as a method for training competitive layers of the unsupervised neural network model developed by Kohonen, called Self-Organizing Map (SOM), in a supervised manner. It also has the advantage of increasing the classification accuracy of the SOM network [76].

5.1.3   The K-MEANS clustering algorithm

The K-MEANS algorithm is one of the classification techniques that have been introduced in the literature. It is partially supervised because the number of clusters is predefined.

In order to clusters M feature vectors into G clusters, assume a data set of M vectors, v_i, $i=1, 2\ ...M$ of dimensionality 1 X N.

Algorithm

1) Select G such that $G < M$ (5.1)

2) Define the clusters centers c_g, g = 1,2…G (5.2)

3) Associate each of vectors v_i to the closest center according to a distance measure. There are several distance measures defined in the literature. Euclidean distance is often used because of its simplicity.

4) The Euclidean distance D between two vectors $v_1 = \{v_{1,1}, v_{1,2}... v_{1,N})$ and $v_2 = \{v_{2,1}, v_{2,2}... v_{2,N})$ is defined as

$$D = \frac{1}{N} \sum_{i=1}^{N} \left(v_{1,i} - v_{2,i} \right)^2 \qquad (5.3)$$

5) The new cluster center c_g is the average of all vectors that belongs to this cluster.

$$c_g^{new} = \frac{1}{N_g} \sum_{i=1}^{N_g} v_i \qquad (5.4)$$

where N_g is the numbers of vectors belonging to g^{th} cluster.

6) The algorithm is repeated until the change in centers is not significant.

5.2 TD2DPCA/VQ-1 tree structure classifier : at each tree node, the subset of subjects are divided into two groups

5.2.1 Training Mode

In the training mode, the features of the data base are extracted, stored, and grouped as described by steps 1 through 7.

<u>Step 1:</u> The suitable transform (Tr) is applied to each m x n image A_i of the N training images, yielding T_i ($i=1$ to N).

$$T_i = Tr\{A_i - \bar{A}\} \qquad (5.5)$$

Where $\bar{A}$ is the mean matrix, of all the N training images.

<u>Step 2:</u> The transform is chosen such that the significant coefficients of T_i are contained in a submatrix, T_i', (upper left part of T_i) of dimension n' x n'. Thus T_i' is used to replace A_i in our algorithm.

<u>Step 3:</u> The covariance matrix S for the N training images is calculated using (5.6).

$$S = \frac{1}{N} \sum_{i=1}^{N} (T_i')^T (T_i') \qquad (5.6)$$

Step 4: A set of k eigenvectors, $V = [V_1, V_2 ... V_k]$ of size n' corresponding to the largest k eigenvalues is obtained for S.

Step 5: The feature matrices of the training images B_i are calculated in (5.8) and (5.7),

$$Y_{j,i} = T_i{}'V_j \quad j = 1,2,....k \ \ and \ \ i = 1,2,..N \qquad (5.7)$$

$$B_i = [Y_{1,i}, Y_{2,i}, ... Y_{k,i}] \qquad (5.8)$$

It is worthwhile to note that the feature matrix representing the training image has dimensions much lower than those obtained using the spatial 2DPCA method ($n' << n$, and now $k' << k$).

Step 6: Vector quantization is employed to group the feature vectors, $Y_{j,i}$, ($i = 1$ to N), representing the training images, Where a tree of VQ codebooks, using $Y_{j,i}$, are constructed as shown in figure 4.1

Step 7: The vectors representing the centroids of all groups are stored.

Input $Y_{1,i}$ $(i=1 \text{ to } N)$

Codebook 1

$G_{1,1}$ $G_{2,1}$

$G_{1,2}$ $G_{2,2}$

Process is continued until $G_{1,1}$ & $G_{2,1}$ contains two images

$G_{d,l}$: d represents group number, l represents stage index
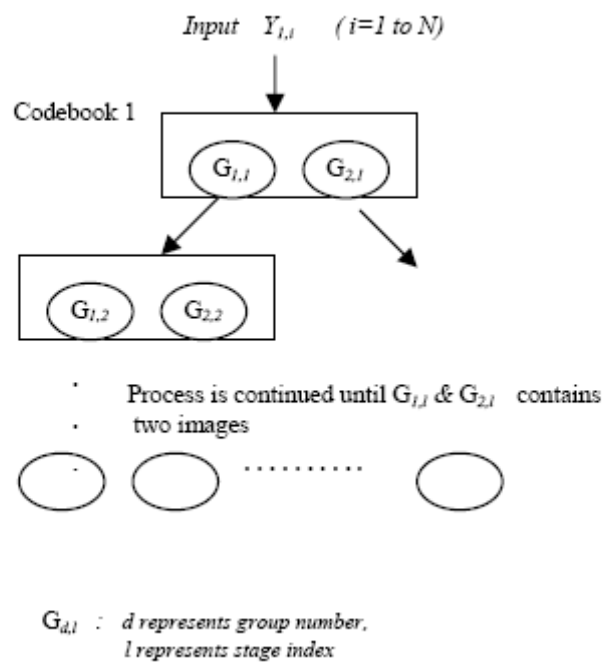
Figure 5-1 : A tree of VQ codebooks employing $Y_{1,i}$

5.2.2   Testing Mode

In the testing mode a facial image A_t is presented to the system to be identified. The following steps are followed

Step 1 The same transform used in the training mode is applied to A_t which yield T_t.

Step 2 The submatrix T_t' containing the significant coefficients is obtained (dimension $n' \, x \, n'$)

Step 3 The feature matrix B_t for the testing image is calculated from

$$Y_{j,t} = T_t' \, V_j \qquad\qquad j = 1,2,....k' \qquad (5.9)$$

$$B_t = [Y_{1,t}, Y_{2,t}, ... Y_{k,t}] \qquad\qquad (5.10)$$

Step 4 Distance measures, such as the Euclidean distances, between the feature vectors of the testing image $Y_{j,t}$ and the centroids, are computed. The group corresponding to the minimum distance is determined. The tested image is assigned to that group.

5.2.3 Experimental Results And Analysis

The proposed algorithm was applied to the ORL database . Results are compared with those obtained using TD2PCA without employing VQ and existing techniques, namely, the 2DPCA, and PCA .

Two experiments have been applied to the ORL database, where all the images are grayscale with 112 x 92 pixels each.

In the first experiment, 40 images of 40 different individuals are used for training and the remaining 360 images are used for testing. A two-dimensional discrete cosine transform (DCT) is applied to the N training images. The dimensions of T'_i and the covariance matrix S are $20x20$. The 5 largest eigenvectors of S corresponding to the 5 largest eigenvalues are obtained. In our approach k of only 5 was needed relative to $k = 10$ in other approaches, while even achieving better recognition accuracy.

The feature matrices for all the training images are obtained using (5.7) and (5.8). A tree of VQ codebooks, using $Y_{1,i}$ $(i= 1$ to $N),$ are constructed as shown in Figure 5-1 , where $Y_{1,i}$ are used to represent the images

 The procedure in section 5.2.2 is followed for the 360 testing images. Results are listed in Tables 5-1 and 5-2.

In the second experiment 5 images per class are used for training and the remaining 200 images are used for testing. The Dimensions of T'_i and S' are the same as in the first experiment. Results are listed in Tables 5-1 and 5-2

Table 5-1 shows that the proposed algorithm yields better recognition accuracy than the TD2DPCA , and 2DPCA method.

Table 5-2 illustrates the storage requirements, in terms of the dimensions of the feature matrix. It is seen that, for the TD2DPCA/VQ and TD2DPCA, the amount of storage is drastically reduced (by approximately 90%), compared with one of the best available algorithm, 2DPCA. In addition the new technique drastically improves the recognition speed in the testing mode. It can be easily shown that in contrast with other techniques, the number of steps required to uniquely identify an unknown facial image is considerably reduced by almost 75%. Consequently, TD2DPCA/VQ lends itself to facial recognition of large databases.

Table 5-1 : Recognition accuracy for experiment I and II on ORL database using

TD2DPCA/VQ-1 ,TD2DPCA, 2DPCA and PCA methods.

Method	Recognition accuracy for experiment I	Recognition accuracy for experiment II
TD2DPCA/VQ-1	79.25 %	92.8 %
TD2DPCA	73.61 %	92.0 %
2DPCA	72.77 %	91.0 %
PCA	62.80 %	83.5 %

Table 5-2 : Dimensions of feature matrix and number of computations required for

the testing mode on ORL database, for experiments I, II.

	TD2DPCA/VQ	2DPCA
Dimensions of feature matrix per image	(20x5)	(112x10)
Storage requirements for N images	(20x5)xN	(112x10)xN
# of multiplications for the testing mode	57344	103040
# of comparisons for the testing mode	10 (experimental)	40

5.3 TD2DPCA/VQ-2 classifier

The proposed algorithm is used when more than one image is used to train the system. It realizes excellent feature reduction properties by exploiting images and covariance matrices representation in the Transform domain as well as vector quantization (VQ). Consequently, the computational and storage requirements are greatly simplified as will be shown later. The algorithm is described below.

5.3.1 Training mode

In the training mode, the features of the data base are extracted, stored, and grouped as described by steps 1 through 7.

<u>Step 1:</u> The suitable transform (Tr) is applied to each $m \; x \; n$ image A_i of the N training images, yielding T_i ($i=1 \; to \; N$).

$$T_i = Tr\{A_i - \bar{A}\} \qquad\qquad (5.7)$$

Where $\bar{A}$ is the mean matrix, of all the N training images.

Step 2: The transform is chosen such that the significant coefficients of T_i are contained in a submatrix, T_i', (upper left part of T_i) of dimension $n' \times n'$. Thus T_i' is used to replace A_i in our algorithm.

Step 3: The covariance matrix S' for the N training images is calculated using (8).

$$S = \frac{1}{N} \sum_{i=1}^{N} (T_i)^T (T_i)$$ (5.8)

S' is the submatrix containing the significant coefficients of S (upper left part of S), dimension $n' \times n'$.

Step 4: A set of k eigenvectors, $V = [V_1, V_2 ... V_k]$ of size n' corresponding to the largest k eigenvalues is obtained for S'.

Step 5: The feature matrices of the training images B_i are calculated in (9) and (10),

$Y_{j,i} = T_i'V_j \quad j = 1,2,....k \ and \ i = 1,2,..N$ (5.9)

$B_i = [Y_{1,i}, Y_{2,i}, ... Y_{k,i}]$ (5.10)

Step 6: Employing vector quantization [75], one centroid , C_i (i=1 to N), per individual is obtained by grouping the feature vectors, representing the different training images per individual (number of poses, P) into one group figure 5-2 . In this work, to illustrate our technique, only the first feature vector, $Y_{1,i}$, is used for grouping the images. In future work, the system will be implemented using more feature vectors.

Step 7: All vectors representing the centroids , C_i (i=1 to N), are stored.

5.3.2 Testing mode

In the testing mode a facial image A_t is presented to the system to be identified. The following steps are followed

Step 1 The same transform used in the training mode is applied to A_t which yield T_t.

Step 2 The sub matrix T_t' containing the significant coefficients is obtained (dimension $n' \, x \, n'$)

Step 3 The feature matrix B_t for the testing image is calculated from

$$Y_{j,t} = T_t' \, V_j \qquad\qquad j = 1,2,....k \qquad\qquad (5.11)$$

84

$$B_t = [Y_{1,t}, Y_{2,t}, \dots Y_{k,t}] \qquad\qquad (5.12)$$

<u>Step 4</u> Distance measures, such as the Euclidean distances, between the feature vectors of the testing image $Y_{j,t}$ and the centroids, are computed. The group corresponding to the minimum distance is determined. The tested image is assigned to that group.

5.4 Experimental Results and Analysis

The proposed algorithm was applied to the ORL database and the Yale database. Two experiments have been performed. In Experiment I , the ORL database is used, where all the images are grayscale with 112 x 92 pixels each. In Experiment II, the Yale database is used, where all the images are grayscale with 243 x 320 pixels each.

In experiment I, 200 images of 40 different individuals are used for training (five images per individual, P=5) and the remaining 200 images are used for testing. A two-dimensional discrete cosine transform (DCT) is applied to the N training images. The dimensions of T'_i and the covariance matrix S' are *20x20*. The 5 largest eigenvectors of S' corresponding to the 5 largest eigenvalues are obtained. The feature matrices for all the training images are obtained.

Employing vector quantization, one centroid per individual, C_i (i=1 to N), is obtained by grouping the feature vectors, representing the five training images per individual, into one group.

In experiment II, 75 images of 15 different individuals are used for training (five images per individual, P=5) and the remaining 90 images are used for testing. A two-dimensional discrete cosine transform (DCT) is applied to the N training images. The dimensions of T'_i and the covariance matrix S' are *50x50*. The 5 largest eigenvectors of S' corresponding to the 5 largest eigenvalues are obtained. The feature matrices for all the training images are obtained using (9) and (10). Again, employing vector quantization, one centroid per individual is obtained by grouping the feature vectors, representing the five training images per individual, into one group.

In experiment I, II , only the first feature vector ($Y_{I,i}$) is used for grouping the images. All vectors representing the centroids , C_i (i=1 to N), are stored.

Table 5-3, illustrates that for the TD2DPCA/VQ method the amount of storage is reduced by a factor of 5 , (corresponding to P = 5), compared with one of the best available algorithms, TD2DPCA. In addition, the new technique increases the recognition speed by a factor of 5.

Due to the considerable reduction in the number of steps required to uniquely identify an unknown facial image the new TD2DPCA/VQ-2 algorithm lends itself to facial recognition of large databases, for real time applications.
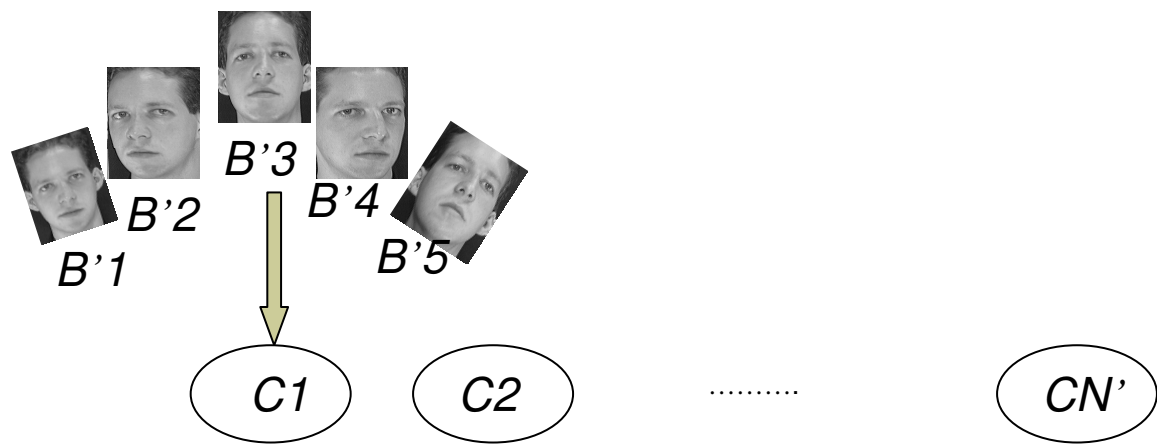
Figure 5-2 Grouping images employing TD2DPCA/VQ

Table 5-3 : Recognition accuracy, storage requirements and computational speed, for

experiments employing TD2DPCA/VQ-2.

	TD2DPCA/VQ	TD2DPCA	**Savings**
Recognition accuracy for experiment I	89.5 %	92 %	**Comparable**
Recognition accuracy for experiment II	82.2 %	78.8 %	**Comparable**
Storage requirements for experiment I	N x (n' x k) 40 x (20x5)	(N x P) x (n' x k) 200 x (20x5)	**80%**
Storage requirements for experiment II	N x (n' x k) 15 x (50x5)	(N x P) x (n' x k) 75 x (50x5)	**80%**
# of comparisons for the testing mode in experiment I	40	200	**80%**
# of comparisons for the testing mode in experiment II	15	75	**80%**

N = number of individuals
P = number of training images (poses) per individual
n' x k = dimensions of the feature matrix

5.5 Conclusion

The TD2DPCA algorithm for recognition and classification of facial images was presented in chapter three. This algorithm reduces the storage requirements by a factor of magnitude and the computational complexity by a factor of 2 while maintaining the recognition accuracy of the recently, reported spatial domain 2DPCA algorithm. The compact representation of the images employing the proposed algorithm enables the usage of other classification tools, vector quantization. This led to TD2DPCA/VQ methods. The TD2DPCA/VQ-1 classifier reduced the computational requirements in the testing mode. The TD2DPCA/VQ-2 further reduces the storage and computational requirements by a factor of P where P is the number of training images per individual when more than one image per individual are used for training.

Experimental results using the ORL, YALE databases confirm these excellent properties.

CHAPTER 6 : PARALLEL STRUCTURE RECOGNITION SYSTEM

6.1 Introduction

Recently, due to emerging critical applications such as biomedical, and security applications, the area of intelligent signal processing has been receiving considerable attention. In this contribution, we present an intelligent signal processing system applied to signal recognition and classification [66-89]. The system employs different structures, multicriteria and multitransform techniques. In addition, principal component analysis in the transform domain in conjunction with vector quantization is developed which result in further improvement in the recognition accuracy and dimensionality reduction. Experimental results are given which confirm the excellent properties of the proposed approaches.

The propose technique can be designed to have evolutionary learning by developing the features and selecting the criteria that are best suited for the recognition problem under consideration. It is conjectured that, ultimately, it will be capable of recognizing an enormously large number of patterns by virtue of the fact that it analyzes the signals in different domains and explores the distinguishing characteristics in each of these domains. Many criteria are developed from the features extracted from the projection of the original and preprocessed signals in different domains, as shown in figure 6-1.

Based on the selected set of criteria and according to the classification technique used, the signals are grouped into a particular number of groups.

Finally, each signal will be identified by a composite index according to the group numbers throughout the classification process.

This Chapter is organized as follows: Section 6.2 presents the parallel implementation grouping structure. In Section 6.3, sample results are given to demonstrate the excellent performance of the parallel implementation structure. Section 6.4 presents the conclusions.

Classification

Criteria Evaluation
in case of corrupted data

Criteria Extraction

D_1 D_2 $\cdot\ \cdot\ \cdot$ D_n

Input
Images

Voting &
Final
Decision

Output
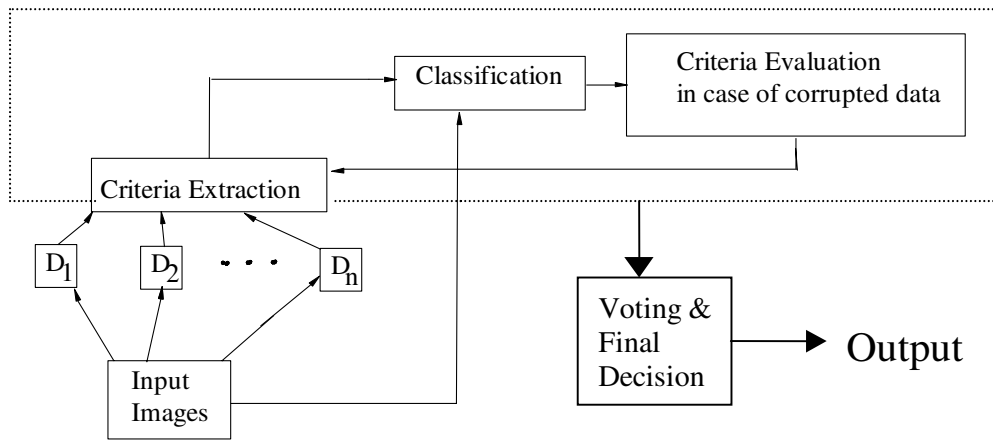
Figure 6-1   The proposed Pattern Recognition System [24]

6.2 <u>The parallel implementation structure</u>

In this implementation, shown in figure 6-2, the pattern recognizer extracts the features in parallel, from more than one transform domain. Different classification criteria in each domain can be developed using the coefficients in that particular domain such as the spectral characteristics, the energy distribution in the different transform domain regions, etc. First, a criterion, with adaptable parameters, is introduced to the TD2DPCA/VQ-2 classifier. A potentially successful criterion with its selected values of the parameters, in a particular domain, clusters the N input signals in a number of distinct non-overlapping clusters. The cluster index, according to that criterion, is denoted.

The TD2DPCA/VQ-2 Classifier learning continues, by testing all the criteria presented over the parameters range for each criterion, until a successful set of criteria is obtained. A successful parallel structure implementation recognition system should yield a unique composite index ($c_1c_2c_3.... c_D$) corresponding to each of the N input signals.

$c_1 = 1,2,... g_1$, $c_2 = 1,2,... g_2$, ... $c_D = 1,2,... g_n$

where D is the number of transform domains, and g_i is the number of groups for each TD2DPCA/VQ-2 Classifier.
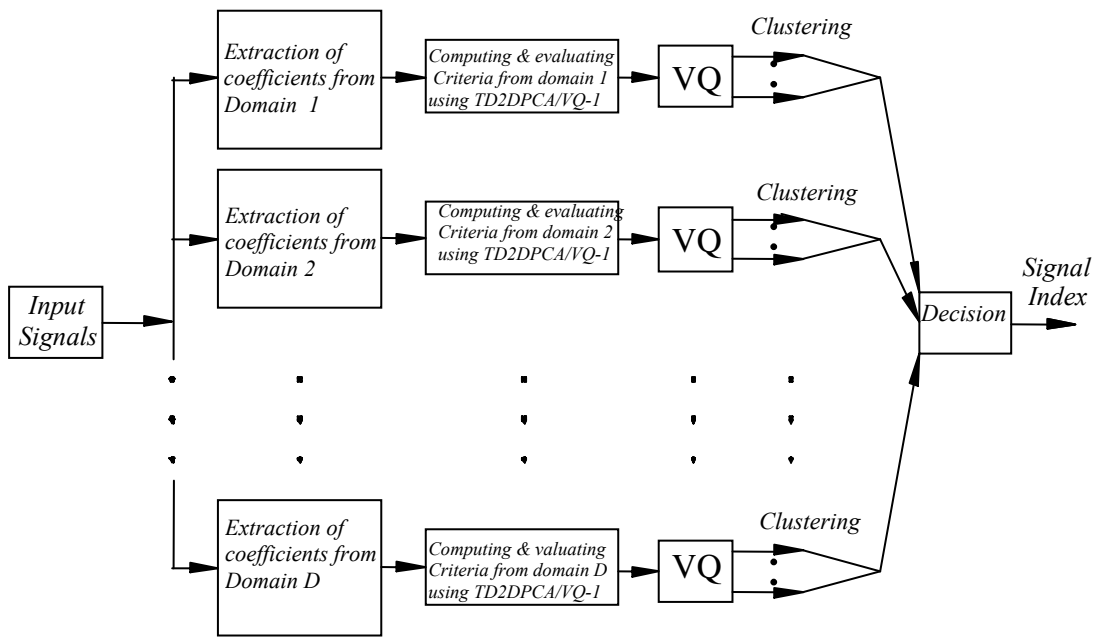
Figure 6-2 : A parallel implementation of the proposed classification technique

6.3   Experimental Results and Analysis employing the Parallel Structure System

In this experiment, three classifiers have been used to recognize these images. The three TD2DPCA/VQ-2 , described in section 5.3, classifiers have the same structure but they employ different set of criteria.

• DCT transform: Submatrix representing the training image contains 20 x 20 low-frequency components,

• Haar transform: Submatrix representing the training image contains 56 x 56 low frequency,

• Haar transform: Submatrix representing the training image contains 56 x 56 high frequency,

The proposed algorithm was applied to the ORL database and the Yale database. Two experiments have been performed.

In Experiment I, the ORL database is used, where all the images are grayscale with 112 x 92 pixels each. In Experiment II, the Yale database is used, where all the images are grayscale with 243 x 320 pixels each.

In experiment I, 200 images of 40 different individuals are used for training (five images (poses) per individual, P=5) and the remaining 200 images are used for testing.

For the first TD2DPCA/VQ-2 classifier, a two-dimensional-DCT is applied to the N training images. The dimensions of T'_i and the covariance matrix S' are *20x20*. The 5 largest eigenvectors of S' corresponding to the 5 largest eigenvalues are obtained.

The feature matrices for all the training images are obtained.

Employing vector quantization, one centroid per individual, C_i (i=1 to N), is obtained by grouping the feature vectors, representing the five training images per individual, into one group.

For the second ,TD2DPCA/VQ-2 ,classifier, the Haar transform is applied to the N training images. The Submatrix containing 56 x 56 low frequency, representing the training image is retained. The dimensions of T'_i and the covariance matrix S' are *56x56*. The 10 largest eigenvectors of S' corresponding to the 10 largest eigenvalues are obtained.

The feature matrices for all the training images are obtained.

Employing vector quantization, one centroid per individual, C_i (i=1 to N), is obtained by grouping the feature vectors, representing the five training images per individual, into one group.

For the third , TD2DPCA/VQ-2 , classifier, the Haar transform is applied to the N training images. The Submatrix containing 56 x 56 high frequency, representing the training image is retained. The dimensions of T'_i and the covariance matrix S' are

56x56. The 10 largest eigenvectors of *S'* corresponding to the 10 largest eigenvalues are obtained.

The feature matrices for all the training images are obtained.

Employing vector quantization, one centroid per individual, C_i (i=1 to N), is obtained by grouping the feature vectors, representing the five training images per individual, into one group.

In experiment II, 75 images of 15 different individuals are used for training (five images per individual, P=5) and the remaining 90 images are used for testing.

For the first classifier a two-dimensional discrete cosine transform (DCT) is applied to the *N* training images. The dimensions of T'_i and the covariance matrix *S'* are *50x50*. The 5 largest eigenvectors of *S'* corresponding to the 5 largest eigenvalues are obtained.

Again, employing vector quantization, one centroid per individual is obtained by grouping the feature vectors, representing the five training images per individual, into one group.

For the second TD2DPCA/VQ-2 classifier, the Haar transform is applied to the *N* training images. The Submatrix containing 56 x 56 low frequency, representing the training image is retained. The dimensions of T'_i and the covariance matrix *S'* are *56x56*. The 10 largest eigenvectors of *S'* corresponding to the 10 largest eigenvalues are obtained.

The feature matrices for all the training images are obtained.

Employing vector quantization, one centroid per individual, C_i (i=1 to N),is obtained by grouping the feature vectors, representing the five training images per individual, into one group.

For the third TD2DPCA/VQ-2 classifier, the Haar transform is applied to the N training images. The Submatrix containing 56 x 56 high frequency, representing the training image is retained. The dimensions of T'_i and the covariance matrix S' are *56x56*. The 10 largest eigenvectors of S' corresponding to the 10 largest eigenvalues are obtained.

In experiment I and II, only the first feature vector $(Y_{1,i,})$ is used for grouping the images. All vectors representing the centroids for the three different classifiers, C_i (i=1 to N), are stored.

Experimental results are compared with results obtained by TD2DPCA/VQ-2, TD2DPCA, and 2DPCA methods.

As shown in Table 6-1, the recognition accuracy has improved by 1.5 % from the previous methods.

The storage requirements are still less than that that required by TD2DPCA, and 2DPCA methods. This is achieved at the expense of more computational complexity.

Table 6-1 : Recognition accuracy for experiment I and II using the parallel structure

TD2DPCA/VQ-2 classifiers, TD2DPCA/VQ-2, TD2DPCA, and 2DPCA.

Method	Recognition accuracy for experiment I	Recognition accuracy for experiment II
Parallel Structure implementation	93.5%	83.5%
TD2DPCA/VQ-2	92.0 %	82.2%
TD2DPCA	91.0 %	78.8%
2DPCA	83.5 %	77.7%

Table 6-2 : Storage requirements for experiment I and II using the parallel structure

TD2DPCA/VQ-2 classifiers, TD2DPCA/VQ-2, TD2DPCA, and 2DPCA.

Method	Comparison of the the storage requirements for experiment I	Comparison of the storage requirements for experiment II
Parallel Structure implementation	3N/5	3N/5
TD2DPCA/VQ-2	N/5	N/5
TD2DPCA	N	N
2DPCA	N	N

Table 6-3 : Table 6-4Number of comparisons required per image in the testing mode, for experiment I and II using the parallel structure TD2DPCA/VQ-2 classifiers, TD2DPCA/VQ-2, TD2DPCA, and 2DPCA.

Method	# of comparisons per image required in the testing mode for experiment I	# of comparisons per image required in the testing mode for experiment II
Parallel Structure implementation	3	3
TD2DPCA/VQ-2	1	1
TD2DPCA	5	5
2DPCA	5	5

6.4 <u>Conclusions</u>

In this contribution, a powerful Intelligent Signal Processing system applied to recognition and classification of signals is presented. This includes the different aspects of the recognition system: multicriteria, multitransform, principal component analysis and Vector Quantization. Sample results are given which confirm the excellent performance of the techniques presented in terms of recognition accuracy, speed, and storage requirements.

CHAPTER 7 CONCLUSIONS AND FUTURE WORK

7.1 Conclusions

In this contribution, 2DPCA analysis, in the transform domain, is presented for classification and recognition of facial images. Several algorithms based on the TD2DPCA are presented. Theses algorithms possess attractive properties, namely, reduced storage requirements and computational complexity while yielding high recognition accuracy.

Experimental results on the ORL, Yale, and UMIST databases are given which confirm the excellent properties of the proposed approaches.

It is worthwhile to note that the TD2DPCA approach is applicable to the classification and recognition of other types of signals.

The TD2DPCA algorithm for recognition and classification of facial images was presented in chapter three. This algorithm reduces the storage requirements by a factor of magnitude and the computational complexity by a factor of 2 while maintaining the recognition accuracy of the recently, reported spatial domain 2DPCA algorithm

In chapter four a TD/2D2DPCA algorithm is presented for facial recognition. It is shown that the new technique retains the high recognition accuracy of the 2DPCA and TD2DPCA methods while reducing the storage requirements by 95 percent compared to the 2DPCA and 25 percent compared to TD2DPCA. It is worthwhile to

note that the computational speed has been reduced greatly relative to 2DPCA algorithm. Experimental results confirm these excellent characteristics.

In addition a modified transform domain two dimensional principal component analysis (M-TD2DPCA) algorithm is described and applied to facial recognition. The proposed technique, while maintaining the excellent characteristics of the recently reported TD2DPCA approach , it requires much fewer computations to obtain the images autocorrelation matrix. Sample results and performance comparison with existing techniques are given which confirm the improved performance of the M-TD2DPCA.

The compact representation of the images employing the TD2DPCA algorithm enables the usage of other classification tools, such as vector quantization. This led to TD2DPCA/VQ method, as shown in chapter five, which further reduces the storage and computational requirements by a factor of P where P is the number of training images per individual when more than one image per individual are used for training. Experimental results using the ORL, YALE databases confirm these excellent properties.

In chapter six, a powerful Intelligent Signal Processing system applied to recognition and classification of signals is presented. This includes the different aspects of the recognition system: multicriteria, multitransform, principal component analysis and Vector Quantization. Sample results are given which confirm the excellent

performance of the techniques presented in terms of recognition accuracy, speed, and storage requirements.

7.2 Future work

Although different types of criteria and classification methods have been examined, throughout this dissertation, still, more work must be done to examine more criteria and more classification techniques to enhance the performance of the suggested pattern recognition systems for all types of problems.

In addition, a self-designing cascaded implementation, shown in Fig. 7-1, needs to be examined. When the classification process is completed, each signal should be represented by a unique composite index, corresponding to the signal path through the decision tree, from the input to one of the terminal nodes of the tree. Classification techniques such as vector quantization or neural networks could be used in conjunction with TD2DPCA method [90-107].

Input signals

Clustering

1 · · · k · · · g_1

Clustering   Clustering   Clustering

1 · · · g_{21} · · · 1 · · · g_{2k} · · · 1 · · · g_{2g1}

clustering clustering   clustering clustering   clustering clustering

clustering   clustering   clustering   clustering

1 · · · r1 · 1 · · · rk · · 1 · · · rj · · 1 · · · rg(r-1)
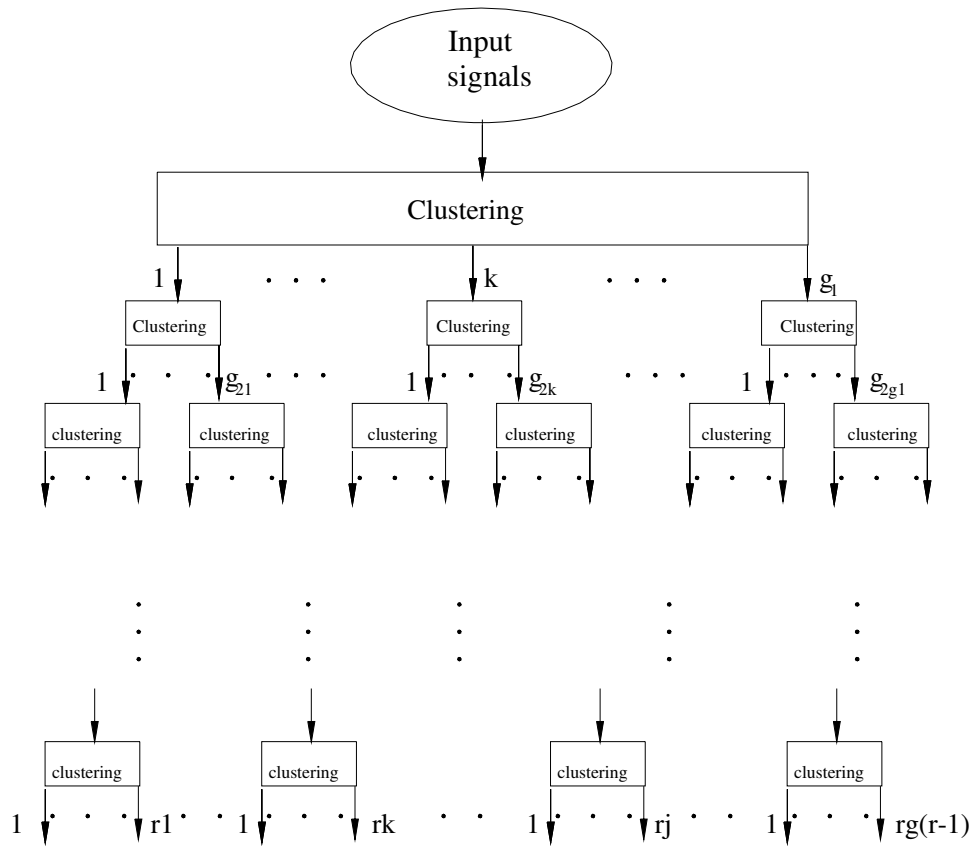
Figure 7-1: Self-designing cascaded implementation

APPENDIX

An simplified example of the SD-TD2DPCA Algorithm using Matlab

SD-TD2DPCA Algorithm

```
x1=imread('C:\ image\image1\s1\1.bmp');
x2=imread('C:\ image\image1\s2\1.bmp');
x3=imread('C:\ image\image1\s3\1.bmp');
x4=imread('C:\ image\image1\s4\1.bmp');
x5=imread('C:\ image\image1\s5\1.bmp');
x6=imread('C:\ image\image1\s6\1.bmp');
x7=imread('C:\image\image1\s7\1.bmp');
x8=imread('C:\image\image1\s8\1.bmp');
x9=imread('C:\image\image1\s9\1.bmp');
x10=imread('C:\image\image1\s10\1.bmp');
x11=imread('C:\ image\image1\s11\1.bmp');
x12=imread('C:\ image\image1\s12\1.bmp');
x13=imread('C:\ image\image1\s13\1.bmp');
x14=imread('C:\ image\image1\s14\1.bmp');
x15=imread('C:\ image\image1\s15\1.bmp');
x16=imread('C:\ image\image1\s16\1.bmp');
x17=imread('C: \image\image1\s17\1.bmp');
x18=imread('C:\ image\image1\s18\1.bmp');
x19=imread('C: \image\image1\s19\1.bmp');
x20=imread('C:\ image\image1\s20\1.bmp');
x21=imread('C:\ image\image1\s21\1.bmp');
x22=imread('C:\ image\image1\s22\1.bmp');
x23=imread('C:\ image\image1\s23\1.bmp');
x24=imread('C:\ image\image1\s24\1.bmp');
x25=imread('C: \image\image1\s25\1.bmp');
x26=imread('C: \image\image1\s26\1.bmp');
x27=imread('C:\ image\image1\s27\1.bmp');
x28=imread('C:\ image\image1\s28\1.bmp');
x29=imread('C:\ image\image1\s29\1.bmp');
x30=imread('C:\ image\image1\s30\1.bmp');
x31=imread('C:\ image\image1\s31\1.bmp');
x32=imread('C:\ image\image1\s32\1.bmp');
x33=imread('C:\ image\image1\s33\1.bmp');
x34=imread('C:\ image\image1\s34\1.bmp');
x35=imread('C:\ image\image1\s35\1.bmp');
x36=imread('C:\ image\image1\s36\1.bmp');
x37=imread('C:\ image\image1\s37\1.bmp');
x38=imread('C: \image\image1\s38\1.bmp');


x1=double(x1);
x2=double(x2);
```

```
x3=double(x3);
x4=double(x4);
x5=double(x5);
x6=double(x6);
x7=double(x7);
x8=double(x8);
x9=double(x9);
x10=double(x10);
x11=double(x11);
x12=double(x12);
x13=double(x13);
x14=double(x14);
x15=double(x15);
x16=double(x16);
x17=double(x17);
x18=double(x18);
x19=double(x19);
x20=double(x20);
x21=double(x21);
x22=double(x22);
x23=double(x23);
x24=double(x24);
x25=double(x25);
x26=double(x26);
x27=double(x27);
x28=double(x28);
x29=double(x29);
x30=double(x30);
x31=double(x31);
x32=double(x32);
x33=double(x33);
x34=double(x34);
x35=double(x35);
x36=double(x36);
x37=double(x37);
x38=double(x38);


I1=x1;
I2=x2;
I3=x3;
I4=x4;
I5=x5;
I6=x6;
I7=x7;
I8=x8;
```

```
I9=x9;
I10=x10;
I11=x11;
I12=x12;
I13=x13;
I14=x14;
I15=x15;
I16=x16;
I17=x17;
I18=x18;
I19=x19;
I20=x20;
I21=x21;
I22=x22;
I23=x23;
I24=x24;
I25=x25;
I26=x26;
I27=x27;
I28=x28;
I29=x29;
I30=x30;
I31=x31;
I32=x32;
I33=x33;
I34=x34;
I35=x35;
I36=x36;
I37=x37;
I38=x38;


Xav=(x1+x2+x3+x4+x5+x6+x7+x8+x9+x10+x11+x12+x13+x14+x15+x16+x17+x1
8+x19+x20+x21+x22+x23+x24+x25+x26+x27+x28+x29+x30+x31+x32+x33+x34+
x35+x36+x37+x38)/38;
x1=x1-Xav;
x2=x2-Xav;
x3=x3-Xav;
x4=x4-Xav;
x5=x5-Xav;
x6=x6-Xav;
x7=x7-Xav;
x8=x8-Xav;
x9=x9-Xav;
x10=x10-Xav;
x11=x11-Xav;
```

```
x12=x12-Xav;
x13=x13-Xav;
x14=x14-Xav;
x15=x15-Xav;
x16=x16-Xav;
x17=x17-Xav;
x18=x18-Xav;
x19=x19-Xav;
x20=x20-Xav;
x21=x21-Xav;
x22=x22-Xav;
x23=x23-Xav;
x24=x24-Xav;
x25=x25-Xav;
x26=x26-Xav;
x27=x27-Xav;
x28=x28-Xav;
x29=x29-Xav;
x30=x30-Xav;
x31=x31-Xav;
x32=x32-Xav;
x33=x33-Xav;
x34=x34-Xav;
x35=x35-Xav;
x36=x36-Xav;
x37=x37-Xav;
x38=x38-Xav;

XT1=x1'*x1;
XT2=x2'*x2;
XT3=x3'*x3;
XT4=x4'*x4;
XT5=x5'*x5;
XT6=x6'*x6;
XT7=x7'*x7;
XT8=x8'*x8;
XT9=x9'*x9;
XT10=x10'*x10;
XT11=x11'*x11;
XT12=x12'*x12;
XT13=x13'*x13;
XT14=x14'*x14;
XT15=x15'*x15;
XT16=x16'*x16;
XT17=x17'*x17;
XT18=x18'*x18;
```

```
XT19=x19'*x19;
XT20=x20'*x20;
XT21=x21'*x21;
XT22=x22'*x22;
XT23=x23'*x23;
XT24=x24'*x24;
XT25=x25'*x25;
XT26=x26'*x26;
XT27=x27'*x27;
XT28=x28'*x28;
XT29=x29'*x29;
XT30=x30'*x30;
XT31=x31'*x31;
XT32=x32'*x32;
XT33=x33'*x33;
XT34=x34'*x34;
XT35=x35'*x35;
XT36=x36'*x36;
XT37=x37'*x37;
XT38=x38'*x38;


XT=(XT1+XT2+XT3+XT4+XT5+XT6+XT7+XT8+XT9+XT10+XT11+XT12+XT1
3+XT14+XT15+XT16+XT17+XT18+XT19+XT20+XT21+XT22+XT23+XT24+XT
25+XT26+XT27+XT28+XT29+XT30+XT31+XT32+XT33+XT34+XT35+XT36+X
T37+XT38)/38;

DXT=dct2(XT);
figure
imshow(log(abs(DXT)),[]), colormap(jet(64)), colorbar
RXT=DXT(1:20,1:20);

[V,D] = eig(RXT);

T1=dct2(I1);
T2=dct2(I2);
T3=dct2(I3);
T4=dct2(I4);
T5=dct2(I5);
T6=dct2(I6);
T7=dct2(I7);
T8=dct2(I8);
T9=dct2(I9);
T10=dct2(I10);
T11=dct2(I11);
T12=dct2(I12);
```

```
T13=dct2(I13);
T14=dct2(I14);
T15=dct2(I15);
T16=dct2(I16);
T17=dct2(I17);
T18=dct2(I18);
T19=dct2(I19);
T20=dct2(I20);
T21=dct2(I21);
T22=dct2(I22);
T23=dct2(I23);
T24=dct2(I24);
T25=dct2(I25);
T26=dct2(I26);
T27=dct2(I27);
T28=dct2(I28);
T29=dct2(I29);
T30=dct2(I30);
T31=dct2(I31);
T32=dct2(I32);
T33=dct2(I33);
T34=dct2(I34);
T35=dct2(I35);
T36=dct2(I36);
T37=dct2(I37);
T38=dct2(I38);

%figure
%imshow(log(abs(T38)),[]), colormap(jet(64)), colorbar
R1=T1(1:20,1:20);
R2=T2(1:20,1:20);
R3=T3(1:20,1:20);
R4=T4(1:20,1:20);
R5=T5(1:20,1:20);
R6=T6(1:20,1:20);
R7=T7(1:20,1:20);
R8=T8(1:20,1:20);
R9=T9(1:20,1:20);
R10=T10(1:20,1:20);
R11=T11(1:20,1:20);
R12=T12(1:20,1:20);
R13=T13(1:20,1:20);
R14=T14(1:20,1:20);
R15=T15(1:20,1:20);
R16=T16(1:20,1:20);
R17=T17(1:20,1:20);
```

```
R18=T18(1:20,1:20);
R19=T19(1:20,1:20);
R20=T20(1:20,1:20);
R21=T21(1:20,1:20);
R22=T22(1:20,1:20);
R23=T23(1:20,1:20);
R24=T24(1:20,1:20);
R25=T25(1:20,1:20);
R26=T26(1:20,1:20);
R27=T27(1:20,1:20);
R28=T28(1:20,1:20);
R29=T29(1:20,1:20);
R30=T30(1:20,1:20);
R31=T31(1:20,1:20);
R32=T32(1:20,1:20);
R33=T33(1:20,1:20);
R34=T34(1:20,1:20);
R35=T35(1:20,1:20);
R36=T36(1:20,1:20);
R37=T37(1:20,1:20);
R38=T38(1:20,1:20);


V1=R1*V(:,1:5);
V2=R2*V(:,1:5);
V3=R3*V(:,1:5);
V4=R4*V(:,1:5);
V5=R5*V(:,1:5);
V6=R6*V(:,1:5);
V7=R7*V(:,1:5);
V8=R8*V(:,1:5);
V9=R9*V(:,1:5);
V10=R10*V(:,1:5);
V11=R11*V(:,1:5);
V12=R12*V(:,1:5);
V13=R13*V(:,1:5);
V14=R14*V(:,1:5);
V15=R15*V(:,1:5);
V16=R16*V(:,1:5);
V17=R17*V(:,1:5);
V18=R18*V(:,1:5);
V19=R19*V(:,1:5);
V20=R20*V(:,1:5);
V21=R21*V(:,1:5);
V22=R22*V(:,1:5);
```

```
V23=R23*V(:,1:5);
V24=R24*V(:,1:5);
V25=R25*V(:,1:5);
V26=R26*V(:,1:5);
V27=R27*V(:,1:5);
V28=R28*V(:,1:5);
V29=R29*V(:,1:5);
V30=R30*V(:,1:5);
V31=R31*V(:,1:5);
V32=R32*V(:,1:5);
V33=R33*V(:,1:5);
V34=R34*V(:,1:5);
V35=R35*V(:,1:5);
V36=R36*V(:,1:5);
V37=R37*V(:,1:5);
V38=R38*V(:,1:5);


xt=imread('C:\manal\image\image1\s22\2.bmp');


figure;
imshow(xt);
xt=double(xt);
IT=xt;
TT=dct2(IT);
RT=TT(1:20,1:20);
VT=RT*V(:,1:5);


D1=V1-VT;
D2=V2-VT;
D3=V3-VT;
D4=V4-VT;
D5=V5-VT;
D6=V6-VT;
D7=V7-VT;
D8=V8-VT;
D9=V9-VT;
D10=V10-VT;
D11=V11-VT;
D12=V12-VT;
D13=V13-VT;
```

```
D14=V14-VT;
D15=V15-VT;
D16=V16-VT;
D17=V17-VT;
D18=V18-VT;
D19=V19-VT;
D20=V20-VT;
D21=V21-VT;
D22=V22-VT;
D23=V23-VT;
D24=V24-VT;
D25=V25-VT;
D26=V26-VT;
D27=V27-VT;
D28=V28-VT;
D29=V29-VT;
D30=V30-VT;
D31=V31-VT;
D32=V32-VT;
D33=V33-VT;
D34=V34-VT;
D35=V35-VT;
D36=V36-VT;
D37=V37-VT;
D38=V38-VT;


N11=norm(D1(:,1))
N12=norm(D1(:,2))
N13=norm(D1(:,3))
N14=norm(D1(:,4))
N15=norm(D1(:,5))
NT1=N11+N12+N13+N14+N15

N21=norm(D2(:,1))
N22=norm(D2(:,2))
N23=norm(D2(:,3))
N24=norm(D2(:,4))
N25=norm(D2(:,5))
NT2=N21+N22+N23+N24+N25


N31=norm(D3(:,1))
N32=norm(D3(:,2))
N33=norm(D3(:,3))
N34=norm(D3(:,4))
```

```
N35=norm(D3(:,5))
NT3=N31+N32+N33+N34+N35


N41=norm(D4(:,1))
N42=norm(D4(:,2))
N43=norm(D4(:,3))
N44=norm(D4(:,4))
N45=norm(D4(:,5))
NT4=N41+N42+N43+N44+N45


N51=norm(D5(:,1))
N52=norm(D5(:,2))
N53=norm(D5(:,3))
N54=norm(D5(:,4))
N55=norm(D5(:,5))
NT5=N51+N52+N53+N54+N55


N61=norm(D6(:,1))
N62=norm(D6(:,2))
N63=norm(D6(:,3))
N64=norm(D6(:,4))
N65=norm(D6(:,5))
NT6=N61+N62+N63+N64+N65


N71=norm(D7(:,1))
N72=norm(D7(:,2))
N73=norm(D7(:,3))
N74=norm(D7(:,4))
N75=norm(D7(:,5))
NT7=N71+N72+N73+N74+N75


N81=norm(D8(:,1))
N82=norm(D8(:,2))
N83=norm(D8(:,3))
N84=norm(D8(:,4))
N85=norm(D8(:,5))
NT8=N81+N82+N83+N84+N85


N91=norm(D9(:,1))
N92=norm(D9(:,2))
N93=norm(D9(:,3))
N94=norm(D9(:,4))
N95=norm(D9(:,5))
NT9=N91+N92+N93+N94+N95


N101=norm(D10(:,1))
```

```
N102=norm(D10(:,2))
N103=norm(D10(:,3))
N104=norm(D10(:,4))
N105=norm(D10(:,5))
NT10=N101+N102+N103+N104+N105

N111=norm(D11(:,1))
N112=norm(D11(:,2))
N113=norm(D11(:,3))
N114=norm(D11(:,4))
N115=norm(D11(:,5))
NT11=N111+N112+N113+N114+N115

N121=norm(D12(:,1))
N122=norm(D12(:,2))
N123=norm(D12(:,3))
N124=norm(D12(:,4))
N125=norm(D12(:,5))
NT12=N121+N122+N123+N124+N125


N131=norm(D13(:,1))
N132=norm(D13(:,2))
N133=norm(D13(:,3))
N134=norm(D13(:,4))
N135=norm(D13(:,5))
NT13=N131+N132+N133+N134+N135

N141=norm(D14(:,1))
N142=norm(D14(:,2))
N143=norm(D14(:,3))
N144=norm(D14(:,4))
N145=norm(D14(:,5))
NT14=N141+N142+N143+N144+N145

N151=norm(D15(:,1))
N152=norm(D15(:,2))
N153=norm(D15(:,3))
N154=norm(D15(:,4))
N155=norm(D15(:,5))
NT15=N151+N152+N153+N154+N155

N161=norm(D16(:,1))
N162=norm(D16(:,2))
N163=norm(D16(:,3))
N164=norm(D16(:,4))
```

```
N165=norm(D16(:,5))
NT16=N161+N161+N161+N161+N161

N171=norm(D17(:,1))
N172=norm(D17(:,2))
N173=norm(D17(:,3))
N174=norm(D17(:,4))
N175=norm(D17(:,5))
NT17=N171+N172+N173+N174+N175

N181=norm(D18(:,1))
N182=norm(D18(:,2))
N183=norm(D18(:,3))
N184=norm(D18(:,4))
N185=norm(D18(:,5))
NT18=N181+N182+N183+N184+N185

N191=norm(D19(:,1))
N192=norm(D19(:,2))
N193=norm(D19(:,3))
N194=norm(D19(:,4))
N195=norm(D19(:,5))
NT19=N191+N192+N193+N194+N195

N201=norm(D20(:,1))
N202=norm(D20(:,2))
N203=norm(D20(:,3))
N204=norm(D20(:,4))
N205=norm(D20(:,5))
NT20=N201+N202+N203+N204+N205

N211=norm(D21(:,1))
N212=norm(D21(:,2))
N213=norm(D21(:,3))
N214=norm(D21(:,4))
N215=norm(D21(:,5))
NT21=N211+N212+N213+N214+N215

N221=norm(D22(:,1))
N222=norm(D22(:,2))
N223=norm(D22(:,3))
N224=norm(D22(:,4))
N225=norm(D22(:,5))
NT22=N221+N222+N223+N224+N225
```

```
N231=norm(D23(:,1))
N232=norm(D23(:,2))
N233=norm(D23(:,3))
N234=norm(D23(:,4))
N235=norm(D23(:,5))
NT23=N231+N232+N233+N234+N235


N241=norm(D24(:,1))
N242=norm(D24(:,2))
N243=norm(D24(:,3))
N244=norm(D24(:,4))
N245=norm(D24(:,5))
NT24=N241+N242+N243+N244+N245


N251=norm(D25(:,1))
N252=norm(D25(:,2))
N253=norm(D25(:,3))
N254=norm(D25(:,4))
N255=norm(D25(:,5))
NT25=N251+N252+N253+N254+N255


N261=norm(D26(:,1))
N262=norm(D26(:,2))
N263=norm(D26(:,3))
N264=norm(D26(:,4))
N265=norm(D26(:,5))
NT26=N261+N262+N263+N264+N265


N271=norm(D27(:,1))
N272=norm(D27(:,2))
N273=norm(D27(:,3))
N274=norm(D27(:,4))
N275=norm(D27(:,5))
NT27=N271+N272+N273+N274+N275


N281=norm(D28(:,1))
N282=norm(D28(:,2))
N283=norm(D28(:,3))
N284=norm(D28(:,4))
N285=norm(D28(:,5))
NT28=N281+N282+N283+N284+N285


N291=norm(D29(:,1))
N292=norm(D29(:,2))
N293=norm(D29(:,3))
N294=norm(D29(:,4))
```

```
N295=norm(D29(:,5))
NT29=N291+N292+N293+N294+N295

N301=norm(D30(:,1))
N302=norm(D30(:,2))
N303=norm(D30(:,3))
N304=norm(D30(:,4))
N305=norm(D30(:,5))
NT30=N301+N302+N303+N304+N305

N311=norm(D31(:,1))
N312=norm(D31(:,2))
N313=norm(D31(:,3))
N314=norm(D31(:,4))
N315=norm(D31(:,5))
NT31=N311+N312+N313+N314+N315

N321=norm(D32(:,1))
N322=norm(D32(:,2))
N323=norm(D32(:,3))
N324=norm(D32(:,4))
N325=norm(D32(:,5))
NT32=N321+N322+N323+N324+N325


N331=norm(D33(:,1))
N332=norm(D33(:,2))
N333=norm(D33(:,3))
N334=norm(D33(:,4))
N335=norm(D33(:,5))
NT33=N331+N332+N333+N334+N335

N341=norm(D34(:,1))
N342=norm(D34(:,2))
N343=norm(D34(:,3))
N344=norm(D34(:,4))
N345=norm(D34(:,5))
NT34=N341+N342+N343+N344+N345

N351=norm(D35(:,1))
N352=norm(D35(:,2))
N353=norm(D35(:,3))
N354=norm(D35(:,4))
N355=norm(D35(:,5))
NT35=N351+N352+N353+N354+N355
```

```
N361=norm(D36(:,1))
N362=norm(D36(:,2))
N363=norm(D36(:,3))
N364=norm(D36(:,4))
N365=norm(D36(:,5))
NT36=N361+N362+N363+N364+N365

N371=norm(D37(:,1))
N372=norm(D37(:,2))
N373=norm(D37(:,3))
N374=norm(D37(:,4))
N375=norm(D37(:,5))
NT37=N371+N372+N373+N374+N375

N381=norm(D38(:,1))
N382=norm(D38(:,2))
N383=norm(D38(:,3))
N384=norm(D38(:,4))
N385=norm(D38(:,5))
NT38=N381+N382+N383+N384+N385




R=[NT1,NT2,NT3,NT4,NT5,NT6,NT7,NT8,NT9,NT10,NT11,NT12,NT13,NT14,N
T15,NT16,NT17,NT18,NT19,NT20,NT21,NT22,NT23,NT24,NT25,NT26,NT27,NT
28,NT29,NT30,NT31,NT32,NT33,NT34,NT35,NT36,NT37,NT38];
[S,H] = min(R)
```

LIST OF REFERENCES

References

[1] L. Sirovich and M. Kirby, "Low-Dimensional Procedure for Characterization of Human Faces," J. Optical Soc. Am., vol. 4, pp. 519-524, 1987.

[2] M. Kirby and L. Sirovich, "Application of the KL Procedure for the Characterization of Human Faces," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 12, no. 1, pp. 103-108, Jan. 1990.

[3] M. Turk and A. Pentland, "Eigenfaces for Recognition," J. Cognitive Neuroscience, vol. 3, no. 1, pp. 71-86, 1991.

[4] A. Pentland, "Looking at People: Sensing for Ubiquitous and Wearable Computing," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 22,no. 1, pp. 107-119, Jan. 2000.

[5] M.A. Grudin, "On Internal Representations in Face Recognition Systems," Pattern Recognition, vol. 33, no. 7, pp. 1161-1177, 2000.

[6] G.W. Cottrell and M.K. Fleming, "Face Recognition Using Unsupervised Feature Extraction," Proc. Int'l Neural Network Conf., pp. 322-325, 1990.

[7] D. Valentin, H. Abdi, A.J. O'Toole, and G.W. Cottrell, "Connectionist Models of Face Processing: a Survey," Pattern Recognition, vol. 27, no. 9, pp. 1209-1230, 1994.

[8] P.S. Penev and L. Sirovich, "The Global Dimensionality of Face Space," Proc. Fourth IEEE Int'l Conf. Automatic Face and Gesture Recognition, pp. 264- 270, 2000.

[9] L. Zhao and Y. Yang, "Theoretical Analysis of Illumination in PCA-Based Vision Systems," Pattern Recognition, vol. 32, no. 4, pp. 547-564, 1999.

[10] L. Wiskott, J.M. Fellous, N. Kru¨ger, and C. von der Malsburg, "Face Recognition by Elastic Bunch Graph Matching," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 19, no. 7, pp. 775-779, July 1997.

[11] M.S. Bartlett, J.R. Movellan, and T.J. Sejnowski, "Face Recognition by Independent Component Analysis," IEEE Trans. Neural Networks, vol. 13, no. 6, pp. 1450-1464, 2002.

[12] B.A. Draper, K. Baek, M.S. Bartlett, J.R. Beveridge, "Recognizing Faces with PCA and ICA," Computer Vision and Image Understanding: special issue on face recognition, in press.

[13] P.C. Yuen and J.H. Lai, "Face Representation Using Independent Component Analysis," Pattern Recognition, vol. 35, no. 6, pp. 1247-1257, 2002.

[14] M.H. Yang, "Kernel Eigenfaces vs. Kernel Fisherfaces: Face Recognition Using Kernel Methods," Proc. Fifth IEEE Int'l Conf. Automatic Face and Gesture Recognition (RGR'02), pp. 215-220, May 2002.

[15] Yang J. , Zhang, D., Frangi, A.F. "Two-Dimensional PCA: A New Approach to Appearance-Based Face Representation and Recognition", IEEE Transaction on Pattern Analysis and Machine Intelligence,Vol.26, No(1), January 2004, pp.131-137.

[16] **Moataz M. Abdelwahab**, and Wasfy B. Mikhael, "An Efficient Frequency Domain Algorithm For Face Recognition Based On Two-Dimensional Principal Component Analysis," 4th WSEAS International Conference on Electronics, Control, and Signal Processing (ICECS'05), pp.240-244, Miami, Florida, USA, November 17-19, 2005.

[17] Wasfy B. Mikhael, and **Moataz M. Abdelwahab**, "A Recognition and Classification Algorithm with Improved Accuracy, Storage, and Computational Properties with Application to Face Recognition," WSEAS Transactions on Signal Processing, Vol.1, Issue 3, pp. 439-445, December 2005.

[18] **Moataz M. Abdelwahab**, and Wasfy B. Mikhael, "Recognition of Noisy Facial Images Employing Transform –Domain Two-Dimensional Principal Component Analysis," The 49th IEEE International Midwest Symposium on Circuits and Systems (MWSCAS'06), San Juan, Puerto Rico, August 2006. **(2nd Best Student paper award (302 papers))**

[19] **Moataz M. Abdelwahab**, and Wasfy B. Mikhael, "A Novel Facial Recognition Technique Employing Karhunen-Loeve Transform," The 12th International Conference on Information Systems Analysis and Synthesis, (ISAS'06), Vol.5, pp.217-220, Orlando, Florida, USA, July 16-19, 2006.

[20] **Moataz M. Abdelwahab**, and Wasfy B. Mikhael, "Computationally Efficient Transform–Domain Two- Dimensional Principal Component Analysis Algorithm for Facial Recognition," The 2nd International Computer Engineering Conference (ICENCO'06), Cairo, Egypt, December 2006. **(Best paper award (Signal processing track)).**

[21] **Moataz M. Abdelwahab**, and Wasfy B. Mikhael, "A New Fast Facial Recognition Algorithm Applicable to Large Databases," The 4th International IEEE-NEWCAS Conference, pp. 193-196**,** June 18-21, 2006, Gatineau, Canada.

[22] **Moataz M. Abdelwahab**, and Wasfy B. Mikhael, "Efficient Feature Representation Employing PCA and VQ in the Transform Domain for Facial Recognition," The 50th IEEE International Midwest Symposium on Circuits and Systems (MWSCAS'07), Montreal, Canada, August 2007.

[23] Wasfy B. Mikhael, **Moataz M. Abdelwahab**, and Manal M. Abdelwahab, "Intelligent Signal Processing Applied to Recognition and Classification of One and Multidimensional Signals," 10th WSEAS International Conference on Circuits, pp.
292-296, July 2006, Vouliagmeni, Athens, Greece.

[24] **Moataz M. Abdelwahab**, Manal M. Abdelwahab, and Wasfy B. Mikhael, "Several Structures and Techniques for the Accurate Classification and Recognition of Signals," WSEAS Transactions on Signal Processing, Vol. 2, Issue 7, pp. 970-976, July 2006.

[26] R. Chellappa, C. Wilson, and S. Sirohey, "Human and Machine Recognition of Faces: A Survey," *Proc. IEEE*, vol. 83, no. 5, pp. 705-740, 1995.

[27] A. Samal and P. Iyengar, "Automatic Recognition and Analysis of Human Faces and Facial Expressions: A Survey," *Pattern Recognition*, vol. 25, pp. 65-77, 1992.

[28] A. Shashua, "Geometry and Photometry in 3D Visual Recognition," PhD thesis, Massachusetts Institute of Technology, 1992.

[29] R. Duda and P. Hart, *Pattern Classification and Scene Analysis*. New York: Wiley, 1973.

[30] R.A. Fisher, "The Use of Multiple Measures in Taxonomic Problems," *Ann. Eugenics*, vol. 7, pp. 179-188, 1936.

[31] Y. Moses, Y. Adini, and S. Ullman, "Face Recognition: The Problem of Compensating for Changes in Illumination Direction," *European Conf. Computer Vision*, 1994, pp. 286-296.

[32] Y. Cheng, K. Liu, J. Yang, Y. Zhuang, and N. Gu, "Human Face Recognition Method Based on the Statistical Model of Small Sample Size," *SPIE Proc. Intelligent Robots and Computer Vision X: Algorithms and Technology*, 1991, pp. 85-95.

[33] S. Baker and S.K. Nayar, "Pattern Rejection," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 1996, pp. 544-549.

[34] Y. Cui, D. Swets, and J. Weng, "Learning-Based Hand Sign Recognition Using SHOSLIF-M," *Int'l Conf. on Computer Vision*, 1995, pp. 631-636.

[35] P. Hallinan, "A Deformable Model for Face Recognition Under Arbitrary Lighting Conditions," PhD thesis, Harvard Univ., 1995.

[36] A. Pentland, B. Moghaddam, and Starner, "View-Based and Modular Eigenspaces for Face Recognition," *Proc. IEEE Conf.Computer Vision and Pattern Recognition*, 1994, pp. 84-91.

[37] H. Murase and S. Nayar, "Visual Learning and Recognition of 3-D Objects from Appearance," *Int'l J. Computer Vision*, vol. 14, pp. 5-24, 1995.

[38] D. Beymer, "Face Recognition Under Varying Pose," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 1994, pp. 756-761.

[39] A. Gee and R. Cipolla, "Determining the Gaze of Faces in Images," *Image and Vision Computing*, vol. 12, pp. 639-648, 1994.

[40] A. Lanitis, C.J. Taylor, and T.F. Cootes, "A Unified Approach to Coding and Interpreting Face Images," *Int'l Conf. Computer Vision*, 1995, pp. 368-373.

[41] Q. Chen, H. Wu, and M. Yachida, "Face Detection by Fuzzy Pattern Matching," *Int'l Conf. Computer Vision*, 1995, pp. 591-596.

[42] I. Craw, D. Tock, and A. Bennet, "Finding Face Features," *Proc. European Conf. Computer Vision*, 1992, pp. 92-96.

[43] T. Leung, M. Burl, and P. Perona, "Finding Faces in Cluttered Scenes Using Labeled Random Graph Matching," *Int'l Conf. Computer Vision*, 1995, pp. 637-644.

[44] K. Matsuno, C.W. Lee, S. Kimura, and S. Tsuji, "Automatic Recognition of Human Facial Expressions," *Int'l Conf. Computer Vision*, 1995, pp. 352-359.

[45] B. Moghaddam and A. Pentland, "Probabilistic Visual Learning for Object Detection," *Int'l Conf. Computer Vision*, 1995, pp. 786-793.

[46] R. Brunelli and T. Poggio, "Face Recognition: Features vs. Templates," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 15, no. 10, pp. 1,042-1,053, Oct. 1993.

[47] J.M. Gilbert and W. Yang, "A Real-Time Face Recognition System Using Custom VLSI Hardware," *Proc. IEEE Workshop on Computer Architectures for Machine Perception*, 1993, pp. 58-66.

[48] B.K.P. Horn, *Computer Vision*. Cambridge, Mass.: MIT Press, 1986.

[49] W.M. Silver, *Determining Shape and Reflectance Using Multiple Images*, PhD thesis, Massachusetts Institute of Technology, 1980.

[50] R.J. Woodham, "Analysing Images of Curved Surfaces," *Artificial Intelligence*, vol. 17, pp. 117-140, 1981.

[51] S. Nayar and H. Murase, "Dimensionality of Illumination in Appearance Matching," *IEEE Conf. Robotics and Automation*, 1996.

[52] P.N. Belhumeur and D.J. Kriegman, "What is the Set of Images of an Object under all Possible Lighting Conditions?," *IEEE Proc. Conf. Computer Vision and Pattern Recognition*, 1996.

[53] P. Hallinan, "A Low-Dimensional Representation of Human Faces for Arbitrary Lighting Conditions," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 1994, pp. 995-999.

[54] K. Liu et al., "Algebraic Feature Extraction for Image Recognition Based on an Optimal Discriminant Criterion," Pattern Recognition, vol. 26, no. 6, pp. 903-911, 1993.

[55] P.N. Belhumeur, J.P. Hespanha, and D.J. Kriengman, "Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 19, no. 7, pp. 711-720, July 1997.

[56] A.M. Martinez and R. Benavente, "The AR Face Database," CVC Technical Report, no. 24, June 1998.

[57] A.M. Martinez and R. Benavente, "The AR Face Database," http://rvl1.ecn.purdue.edu/~aleix/aleix_face_DB.html, 2003.

[58] J. Yang, J.Y. Yang, "From Image Vector to Matrix: A Straightforward Image Projection Technique—IMPCA vs. PCA," Pattern Recognition, vol. 35, no. 9, pp. 1997-1999, 2002.

[59] W. Yambor, B. Draper, and R. Beveridge, "Analyzing PCA-Based Face Recognition Algorithms: Ei-Genvector Selection and Distance Measures," Empirical Evaluation Methods in Computer Vision, H. Christensen and J. Phillips, eds., Singapore: World Scientific Press, 2002.

[60] J.R. Beveridge, K. She, B. Draper, and G.H. Givens, "Parametric and Nonparametric Methods for the Statistical Evaluation of Human ID Algorithms," Proc. Third Workshop Empirical Evaluation of Computer Vision Systems, Dec. 2001.

[61] http://www.orl.co.uk/facedatabase.html.

[62] http://cvc.yale.edu/projects/yalefaces /yaleface /yalefaces.html.

[63] http://images.ee.umist.ac.uk/danny /database.html

[64] P. Duhamel and C. Guillemot, "Polynomial transform computation of 2-D DCT", Proc. ICASSP'90, Apr.1990, pp1515-1518.

[65] Chang, T., and Kuo, C.C.J., "Texture Analysis and Classification with Tree-Structured Wavelet Transform", IEEE Transactions on Image Processing, Vol.2, No. 4, October 1993, pp. 429-441.

[66] Abdelwahab, M. M., and Mikhael, W. B., "Neural Network Pattern Recognition Employing Multicriteria Extracted From Signal Projections in Multiple Transform Domains", Proceedings of International Symposium on Intelligent Multimedia, Video and Speech Processing, Hong Kong, 2001, pp. 40-43.

[67] Abdelwahab, M. M., and Mikhael, W. B., "Multistage Classification/Recognition Employing Vector Quantization Coding and Criteria Extracted from Nonorthogonal and Preprocessed Signals Representation", Applied Optics, Vol. 43, No.2, January 2004, pp.416-424

[68] Allali, M., and DeBrunner, V., "Using Wavelet-Based Indices For Detecting Abrupt Changes in Signals", Conference Record of the Thirty-Fifth Asilomar Conference on Signals, Systems and Computers, Vol. 1, Nov. 2001, pp.726-729.

[69] Chapa, J.O., and Rao, R. M., " Algorithms For Designing Wavelets To Match A Specified Signal", IEEE Transactions on Signal Processing, Vol. 48, No. 12, December 2000, pp.3395-3406.

[70] Berg, A. P., and Mikhael, W. B., "A Survey of Mixed Transform Techniques for Speech And Image Coding", Proceedings of the 1999 IEEE International Symposium on Circuits and Systems, ISCAS '99, Vol.4, 30 May-2 June 1999, pp.106-109.

[71] Bishof, H., Schneider, W., and Pinz, A. J., "Multispectral classification of LANDSATimages using neural networks", IEEE Trans. on Geoscience and Remote Sensing, May 1992, Vol. 30, pp. 482-490.

[72] Mikhael, W. B., and Abdelwahab, M. M., "Multi Criteria Multi Transform Neural Network", Circuits, Systems and Signal Processing, Vol. 21 No.5 pp. 451-460 Sep.-Oct. 2002.

[73] Mikhael, W. B., and Abdelwahab, M. M., "A Novel Intelligent Classifier/Recognizer Employing Vector Quantization Coding of Non Orthogonal Signal and Preprocessed Signal Representations", ICNNSP'03 Conference, Nanjng china, Dec. 14-17 2003,

[74] Mikhael, W.B., Abdelwahab, M. M., and Krishnan, V. "Self Designing Intelligent Signal Processing System Capable of Evolutionary learning for classification/Recognition of 204 one and multidimensional Signal", UCF Patent Disclosure August 24, 2001, Provisional obtained.

[75] Mikhael, W.B., Krishnan, V., and Abdelwahab, M. M., Projection of Sampled Signals in Different Transform Domains With Useful Applications, The Fourth International Conference on Sampling Theory and Its Applications, Orlando, FL, May 2001. (Invited Paper and Plenary Talk).

[76] Mikhael, W. B., and Ramaswamy, A., "Resolving Images in Multiple Transform Domains with Applications", Digital Signal Processing, Vol.5, 1995, pp. 81-90.

[77] Mikhael. W. B., and Ramaswamy, A., "An Efficient Representation of Nonstationary Signals Using Mixed-Transforms with Application to Speech", IEEE Transactions on Circuit and Systems-II: Analog and Digital Signal Processing, Vol. 42, No. 6, June 1995, pp393-401.

[78] Patel, J. N., Khokhar, A. A., and Jamieson, L. H., "Scalability of 2-D Wavelet Transform Algorithms: Analytical and Experimental Results on MPPs", IEEE Transactions on Signal Processing, Vol. 48, No.12, December 2000, pp.3407-3419.

[79] Rice, B. F., "Introduction to Automatic Signal Classification", International conference on signal processing applications and technology 1997.

[80] Sandberg, I. W., "General Structures for Classification", IEEE Transaction on Circuits and Systems-1: Fundamental Theory and application, May 1994, Vol. 41, No. 5, pp. 372- 376.

[81] Sarlashkar, M. N., Bodruzzaman, M., and Malkani, M.J., "Feature Extraction Using Wavelet Transform For Neural Network Based Image Classification", Proceedings of the Thirtieth Southeastern Symposium on System Theory, 8-10 March,1998, pp.412-416.

[82] Senior, A., " A Combination Fingerprint Classifier", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 23, No.10, October 2001, pp.1165-1174.

[83] Simard, M., Saatchi, S. S., and Grandi, G. D., "The Use of Decision Tree and Multiscale Texture for Classification of JERS-1 SAR Data over Tropical Forest", IEEE Transactions on Geoscience and Remote Sensing, Vol. 38, No.5, September 2000, pp.2310-2321.

[84] Srivastava, A., Han, E. and Kumar, V., "Parallel Formulations of Decision-Tree Classification Algorithms", Proceedings of International Conference on Parallel Processing, 1998, pp237-244.

[85] Tilki, J. F., and Beex, A. A., "Image Data Compression Using Multiple Bases Representation", Proceedings of the 26th Southeastern Symposium on System Theory, 1194, pp.457-461.

[86] Tran, T. D., "The BinDCT: Fast Multiplierless Approximation of the DCT", IEEE Signal Processing Letters, Vol.7, No.6, June 2000, pp.141-144.

[87] Yu, X., Dent, D., and Osborn, C., "Classification of Ballistocardiography Using Wavelet Transform and Neural Networks", 18th Annual International Conference of IEEE Engineering in Medicine Biology Society, Amsterdam 1996, pp.937-938.

[88] Zhang, G. P., "Neural Networks for Classification: A Survey", IEEE Transactions on Systems, Man, and Cybernetics-Part C: Applications and Reviews, Vol. 30, No.4, November 2000, pp.451-462.

[89] Zhang, P., Chen, L., and Kot, A. C., "A Novel Hybrid Classifier for Recognition of Handwritten Numerals", IEEE International Conference on Systems, Man and Cybernetics, 2000, Vol.4, pp.2709-2714.

[90] Fogel, D. B., "An Information Criterion for Optimal Neural Network Selection", IEEE Transaction on Neural Network, Vol. 2, No.5, September 1991, pp.490-497. Fontenla- Romero, O., Alonso-Betanzos, A., and Guijarro-Berdinas, B., "Adaptive Pattern Recognition in the Analysis of Cardiotocographic Records", IEEE Transactions on Neural Networks, Vol.12, No.5, September 2001, pp.1188-1195.

[91] Gao, Q., Förster, P., Möbus, K. R., and Moschytz, G. S., "Fingerprint Recognition Using CNNS: Fingerprint Preprocessing", The 2001 IEEE International Symposium on Circuits and Systems, 2001. ISCAS 2001, Vol.2, 6-9 May 2001, pp. 433-436.

[92] Gelfand, S. B., Ravishankar, C. S., and. Delp, E. J., "An Iterative Growing and Pruning Algorithm for Classification Tree Design", IEEE Transaction on Pattern analysis and Machine Intelligence, Vol. 13, No.2, February 1991, pp.163-174.

[93] Giacinto, G., Roli, F., Fumera, G., "Unsupervised Learning of Neural Network Ensembles for Images Classification", IEEE International Joint Conference on Neural Networks, 2000, Vol. 3, pp.155-159.

[94] Giusti, N., Masulli, F., Sperduti, A., "Theoretical and Experimental Analysis of a Two- Stage System for Classification", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 24, No. 7, July 2002, pp. 893-904.

[95] Go, J., Han, G., Kim, H., and Lee, C., "Multigradient: A New Neural Network Learning Algorithm For Pattern Classification", IEEE Transactions on Geoscience and Remote Sensing, Vol. 39, No.5, May 2001, pp.986-993.

[96] Gray, R. M., Oehler, K.L., Perlmutte, K.O., Ohlsen, R.A., "Combining Tree-Structured Vector Quantization with Classification and Regression Trees", Conference Record of The Twenty-Seventh Asilomar Conference on Signals, Systems and Computers, Nov. 1993, pp.1494-1498.

[97] Haddadnia, J., Faez, K., and Ahmadi, M., "A Neural Based Human Face Recognition System Using An Efficient Feature Extraction Method With Pseudo Zernike Moment", Journal Of Circuits, System and Computers, 2002, Vol. 11, No. 3, pp.283-304.

[98] Hansen, L., K., and Salamon, P., "Neural Network Ensembles", IEEE Transactions on Pattern Analysis and Machine Intelligence, October 1990, Vol. 12, No.10, pp.993-1001.

[99] Hara, Y., Atkins, R.G., Yueh, S.H., Shin, R.T., and Kong, J.A., "Application of Neural Networks to Radar Image Classification", IEEE Transactions on Geoscience and Remote Sensing, January 1994, Vol. 32, No. 1, pp. 100-109.

[100] Hatzichristos, G., and Fargues, M. P., "A Hierarchical Approach To The Classification of Digital Modulation Types in Multipath Environments", 35th Asilomar conference on Signals, Systems and Computers, 2001, vol.2, pp.1494-1498.

[101]Haykin, S., and Deng, C., "Classification of Radar Clutter Using Neural Networks", IEEE Transactions on Neural Networks, Vol. 2, No. 6, November 1991, pp.589-600.

[102] Jain, A. K., Prabhakar, S. and Hong, L. "A Mulichannel Approach to Fingerprint Classification", IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol. 21, No.4, April 1999, pp.348-359.

[103] Jaisimha, M. Y., Goldschneider, J. R., Mohr, A. E., Riskin, E. A., Haralick, R. M., "On Vector Quantization for Fast Facet Edge Detection", IEEE International Conference on Acoustics, Speech, and Signal Processing, 1994. ICASSP-94, 19-22 Apr 1994, Vol.5, pp. V/37-V/40.

[104] Kijsirikul, B., Chongkasemwongse, K., "Decision Tree Pruning Using Backpropagation [105] Neural Networks", Proceedings of International Joint Conference on Neural Networks, IJCNN '01, 2001, Vol.3, pp.1876-1880.

[106] Leondes, C. T.,"Image Processing and Pattern Recognition", (Academic Press, 1998).

[107] Lu, Y.," Knowledge Integrations in a Multiple Classifier System", Applied Intelligence, April 1996, Vol.6, no.2, pp.75-86.