Effect of Noise, Blur and Motion on Global Appearance Face Recognition based Methods performance

Dalila CHERIFI
Institute of Electrical and Electronics Engineering
University of BOUMERDÈS
ALGERIA

Nadjet RADJI
Institute of Electrical and Electronics Engineering
University of BOUMERDÈS, ALGERIA

Amine NAIT-ALI
LiSSi, EA 3956, Paris-Est CRETEIL (UPEC), CRETEIL, FRANCE

ABSTRACT

In this work, an objective comparison between some common global appearance face recognition based methods (PCA, FLD, SVD, DCT, DWT and WPD) has been carried out when considering some natural effects that may decrease the performances. In particular, effects such as blur, motion, noise and their combination are taken into account. To evaluate the performances, FEI database containing images corresponding to 200 individuals are used.

For each individual, 14 positions have been considered. The quality of face reconnaissance is measured using the well-known Equal Error Rate (EER) criteria. Interesting results are obtained highlighting the superiority, in some specific contexts, of some of the evaluated methods.

Keywords

Face recognition, PCA, SVD, DCT, DWT, WPD, motion, blur and noise.

1. INTRODUCTION

Many face recognition methods have been proposed. Basically, they techniques can be classified into three categories [1]: Holistic or Global-Appearance-based methods [2], Local-feature-based methods or local appearance-based methods, and hybrid methods [3].

In Global-Appearance-based methods, the whole image is used as a raw input to the learning process. Examples of these techniques are Principal Component Analysis (PCA), Discrete Cosine Transforms (DCT), and Linear Discriminate Analysis (LDA). The local appearance-based methods can be divided into two groups: The ones that require the use of specific regions located on a face such as eyes, nose and mouth, as well as their relationships with each other [4] and the ones that simply partition the input face image into blocks without considering any specific regions [5]. In hybrid methods, associate feature of the holistic and local techniques. Modular Eigen faces [6] Hybrid Local Feature Analysis (LFA) [7] and Component-based 3D Models [8] are examples of these methods. The most successful and well-studied techniques to face recognition are the appearance-based methods. In this work, several appearance-based methods are presented namely: PCA, FLD, SVD, and DWT. In section 2, some common face recognition methods are presented. In section 3, the condition of experiments is described, and results are provided and discussed. Finally, a conclusion of this work is given in section 4.

2. COMMON FACE RECOGNITION METHODS

2.1. Principal Component Analysis (PCA) approach

Principal Component Analysis (PCA) is one of the most popular appearance-based methods used mainly for dimensionality reduction in compression and recognition problems [9]. PCA is known as Eigenspace projection which is based on linearly projecting the image space to a low dimensional feature space [10]. PCA based approaches typically include two phases: **training** and **classification** [11], [12] which respectively include the following steps.

A. Training phase

- A training set **S** of M face images (I_i; i = 1… M) is first acquired.
- Each face image I_i is expressed as a vector Γ_i(i=1...M) by concatenating each row (or column) into a long thin vector.
- The average face vector of the set is:

$$\Psi = \frac{1}{M}\sum_{i=1}^{M}\Gamma_i \quad (1)$$

- The images are mean centered by subtracting the mean image from each image vector.

$$\left[\Phi_i = \Gamma_i - \Psi\right]_{N^2 X 1} \quad i = 1, 2, \dots M (2)$$

- Build the mean- subtract face matrix A

$$A = \left[\Phi_1 \quad \Phi_2 \quad \Phi_3 \quad \cdots \quad \Phi_M\right]_{N^2 x\, M} \quad (3)$$

- Get the eigenvalues and eigenvectors of covariance matrix (C) which is given as:

$$\left[C = \frac{1}{M}\sum_{n=1}^{M}(\Phi_n\Phi_n^T) = AA^T\right]_{N^2 x\, N^2} \quad (4)$$

The covariance matrix C is an N^2 x N^2 real symmetric matrix, and determining the N^2 eigenvectors and eigenvalues is an intractable task for typical image sizes. We need a computationally feasible method to find these eigenvectors.

Consider the eigenvectors $\mathbf{V_i}$ of $\mathbf{A^T A}$ such that

$$A^T AV_l = \mu_l V_l \quad (5)$$

Premultiplying both sides by A, yields

$$AA^T AV_l = \mu_l AV_l \quad (6)$$

$$C\, AV_l = \mu_l AV_l \quad (7)$$

$$C\, U_l = \mu_l U_l \quad (8)$$

Where we note that AV_l are the eigenvectors of $C = AA^T$. Thus AA^T and $A^T A$ have the same eigenvalues and their eigenvectors are related as follows:

$$U_l = AV_l \quad (9)$$

Following the analysis, we construct the M x M matrix L

$$L = A^T A \quad (10)$$

- Get the Eigenfaces

The M eigenvectors, V_l of L determine linear combinations of the M training set face images to form the eigenfaces U_l

$$U_l = \sum_{k=1}^{M} V_{lk} \Phi_k \qquad l = 1 \ldots M \qquad (11)$$

- Compute for each face the projection onto the face space:

All the face images are projected into the subspace spanned by the computed M eigenvectors. The projection weights are found using the following formula:

$$W_{lk} = U_k^T (\Gamma_l - \Psi)$$
$$W_{lk} = U_k^T \Phi_l \quad (12)$$
$$\text{for } k, i = 1, 2, \ldots M$$

These weights form the feature vectors

$$\Omega_i^T = [W_{i1} W_{i2} \ldots W_{iM}] \quad (13)$$
$$\text{for } i = 1, 2, \ldots M$$

- Compute the threshold θ that defines the maximum allowable distance from any face class

$$\theta = \frac{1}{2} \max(\|\Omega_i - \Omega_j\|) \quad i, j = 1 \ldots M \qquad (14)$$

Where $\|\Omega_i - \Omega_j\|$ is the Euclidian distance between weight vector i and weight vector j

B. Classification phase

The test face image to be recognized I_{test} is expressed as a vector Γ_{test} by concatenating each row (or column) into a long thin vector.

- Subtract the average face from the test image

$$\Phi_{test} = \Gamma_{test} - \Psi \qquad (15)$$

- Compute its projection onto the face space

The weight of the test image is determined as:

$$W_k = U_k^T (\Gamma_{test} - \Psi) \quad \text{for } k = 1 \ldots M \quad (16)$$

The weights form a feature vector as following,

$$\Omega_{test}^T = [W_1 W_2 \ldots W_M] \qquad (17)$$

- Computing the distance between the test face and all known faces as:

$$\varepsilon_k(\Omega_{test}, \Omega_k) = \|\Omega_{test} - \Omega_k\| \quad (18)$$

Where

ε_k is Euclidean Distance or L_2 distance

Ω_{test} is the feature vector of the test face

Ω_k is a vector describing the k^{th} face class a

- Classification of the input face:

Classification is performed by comparing the feature vectors of the face library members with the feature vector of the input face image. This comparison is based on the Euclidean distance between the two members to be smaller than a user defined threshold θ

- If $min\{\varepsilon_k\} \geq \theta(k = 1, \ldots M)$ it is an unknown face
- If $min\{\varepsilon_k\} \leq \theta$ it is a known face

2.2.Face Recognition using Fisher's Linear Discriminant (FLD)

PCA is a classical technique for signal representation and fisher's linear discrimanant (FLD) is a classical technique for pattern recognition [13]. The basic steps involved in Fisherfaces method are as follows [14]:

Step1: We need training set composed of a relatively large group of subjects with diverse facial characteristics. The database should contain several examples of face images for each subject in the training set and at least one example in the test set.

Step2: For each image starting with the two-dimensional *NxN* array of intensity values *I*(*x*, *y*), we construct the lexicographic vector expansion $\Gamma \in R^{NXN}$.

Step3: Define all instances of the same person's face as being in one class and the faces of different subjectsas being in different classes for all subjects in the training set.

Step 4: In order to overcome the problem of a singular S_w,we first use PCA to reduce the dimension of the feature space to *M* - c, and then apply the standard FLD to reduce the dimension to *c* - 1.

Step 5: Calculate the mean μ_j of each class as:

$$\mu_j = \frac{1}{n_j} \sum_{i=1}^{n_j} \Omega_{PCA_i^j} \qquad j = 1, \ldots, c \, (19)$$

Where

μ_j is an $(M - c)$ column vector.

n_j is the number of samples in class j.

$\Omega_{PCA_i^j}$ is the i^{th} PCA feature vector of class j.

Ω_{PCA} is $(M - c)XM$ matrix.

Step 5: Calculate the mean of all classes μ as

$$\mu = \frac{1}{M} \sum_{j=1}^{c} \Omega_{PCA_i^j} \qquad i = 1, \ldots, n_j \, (20)$$

Where

μ is $(M - c)$ column vector.

M is the total number of images.

Step 6: Subtract the mean of each class from the PCA projected images.

$$\Phi_i^j = \Omega_{PCA_i^j} - \mu_j \quad i = 1, \ldots, n_j \quad j = 1, \ldots, c \quad (21)$$

Where

Φ_i^j are $(M-c)$ column vectors.

Step 7: Build scatter matrices for each class S_j,

$$S_j = \sum_{i=1}^{n_j} (\Phi_i^j)(\Phi_i^j)^T \qquad j = 1, \ldots, c \qquad (22)$$

$$S_j = \sum_{i=1}^{n_j} (\Omega_{PCA_i^j} - \mu_j)(\Omega_{PCA_i^j} - \mu_j)^T \quad (23)$$

Where
S_j is an $(M-c)x(M-c)$ matrix.

Step 8: Evaluate the within-class scatter matrix S_w as follows*:*

$$S_w = \sum_{j=1}^{c}\sum_{i=1}^{n_j} (\Omega_{PCA_i^j} - \mu_j)(\Omega_{PCA_i^j} - \mu_j)^T \quad (24)$$

$$S_w = \sum_{j=1}^{c} S_j = S_1 + S_2 +, \ldots, S_c \quad (25)$$

Where S_w is an $(M-c)x(M-c)$ matrix.

Step 9: Calculate the between-class scatter matrix S_b as follows:

$$S_b = \sum_{j=1}^{c} n_j(\mu_j - \mu)(\mu_j - \mu)^T \quad (26)$$

Where S_b is $(M-c)x(M-c)$ matrix.

Step 10: Calculate the projection matrix eigenvectors.
The optimal projection W_{opt} is chosen as the matrix with orthonormal columns which maximizes the ratio of the determinant of the between-class scatter matrix of the projected samples to the determinant of the within-class scatter matrix of the projected samples, i.e.,

$$W_{opt} = argmax\left|\frac{W^T S_b W}{W^T S_W W}\right| \quad (27)$$

This ratio is maximum when the column vectors of the projection matrix W are the eigenvectors of ${S_W}^{-1}S_b$ i.e.

$$W = eig(S_W^{-1}S_b) \qquad (28)$$

Columns of W are eigenvectors satisfying

$$S_b W_i = \lambda_i S_w W_i \qquad i = 1, 2, \ldots, m \quad (29)$$

Where m=c-1
The generalized eigenvalues are the roots of the characteristic polynomial:

1.1.1.1 $|S_b - \lambda_i S_w$

generalized eigenvectors W_i are obtained by solving the equality:

$$(S_b - \lambda_i S_w)W_i = 0 \qquad (31)$$

The eigenvectors of ${S_W}^{-1}S_b$ are also referred to as the Fisherfaces.

$$W_{FLD_i}{}^T = [W_{i1} W_{i2} \ldots W_{i(c-1)}]$$
$$for\ i = 1, 2, \ldots M - c \quad (32)$$

Step 11: Project faces onto the Fisher Linear Discriminant space.

$$\Omega_{FLD} = W_{FLD}{}^T \Omega_{PCA} \quad (33)$$

Where
Ω_{FLD} is the Fisher $(c-1)xM$ projection matrix.
Ω_{PCA} is the $(M-c)xM$ PCA projection matrix.
W_{FLD} is the $(M-c)x(c-1)$ Fisherfaces matrix.

Step 12: This stepconcerns the test image classification.
The test image is first projected onto PCA subspace for dimension reduction and then projected on the FLD subspace for feature extraction. The test image projections onto the PCA subspace, follows similar steps in Eigenfaces method, that is

$$W_k = U_k^T \Phi_{test} = U_k^T(\Gamma_{test} - \Psi)$$
$$for\ k = 1 \ldots M - c \qquad (34)$$

$$(\Omega_{PCA})_{test}{}^T = [W_1 W_2 \ldots W_{M-c}] \qquad (35)$$

Where
Γ_{test} is the test image concatenated vector.
Ψ is the original set mean image.
U_k^T is the kth covariance matrix eigenvectors.
$(\Omega_{PCA})_{test}$ is the test image PCA feature vector.

Projection of the test image onto FLD subspace:

$$(\Omega_{FLD})_{test} = W_{FLD}{}^T(\Omega_{PCA})_{test} \quad (36)$$

$(\Omega_{FLD})_{test}$ is the FLD feature vector of the test image.
The test image is identified using a similarity measure. The face with the minimum distance with the test face image is labeled with the identity of that image. The minimum distance is evaluated using the Euclidian distance method as expressed earlier in Eigenfaces method.

2.3.Face recognition Singular Value Decomposition (SVD)

Most conventional techniques of image analysis assume images as elements of a vector space. Later, there has been a steady growth of literature which regards images as matrices, [15], [16]. Singular Value Decomposition (SVD) is among the methods which consider an image as a matrix. Several successful methods have been proposed using holistic features of the faces and one popular category of these techniques is based on singular value decomposition (SVD) for face recognition [17], [18].
SVD theory states that any matrix A of size MxN can be factorize into two orthogonal matrices (U,V) and a diagonal matrix $\sum_A$ that is

$$A = U\textstyle\sum_A V^T \quad (37)$$

Where
U is an $M \times M$ orthogonal matrix.
V is an $N \times N$ orthogonal matrix.
$\sum_A$ is an $M \times N$ diagonal matrix

The diagonal element of $\sum_A$ are the singular values (SV) which are arranged in a decreasing order.
The main steps involved in an SVD-based face recognition are:

Step1: Training set of M face images (A_i; i = 1… M) is first acquired.

Step2: The training images are decomposed using eq. (37). Suppose $\boldsymbol{A}_i$ is matrix of a face image. If $\boldsymbol{A}_i$ is the face image matrix, the feature vector $\boldsymbol{Y}_i^{SVD} \in \boldsymbol{R}^P$ is obtained by arranging its singular values:

$$Y_i^{SVD} = [\sigma_1, \sigma_2, \sigma_3, \dots, \sigma_P]^T \quad (38)$$

Step3: A test face image is also decomposed using eq. (37) and its feature vector $\boldsymbol{Y}_{test}^{SVD} \in \boldsymbol{R}^P$ is formed by arranging its singular values:

$$Y_{test}^{SVD} = [\sigma_1, \sigma_2, \sigma_3, \dots, \sigma_P]^T \quad (39)$$

Step 4: Compute the Euclidean distance between feature vector of the test face imageand feature vectors of training face images.

$$d_i = \left\| Y_{test}^{SVD} - Y_i^{SVD} \right\| \quad \text{for i = 1 ... M} \quad (40)$$

Step 5: The face with has the minimum distance with the test face image is labeled with the identity of that image. If the minimum d_i is less than a predefined threshold **θ** the face image is classified as "known", otherwise the face is classified as an “unknown face”.

2.4. Face recognition using Discrete Cosine Transform (DCT)

Discrete Cosine Transform (DCT) is a popular technique in imaging and video compression, which was first applied in image compression [19], by transforming signals in the spatial representation into a frequency representation.
DCT is an invertible linear transform that can express a finite sequence of data points in terms of a sum of cosine functions oscillating at different frequencies. Face recognition algorithms incorporating DCT are available in [15, 16, 20].
One dimensional discrete cosine transform is useful in processing one-dimensional signals such as speech waveforms. For analysis of two-dimensional (2D) signals such as images, a 2-D form of the DCT is required.
Application of the formal definition for the DCT to a one-dimensional sequence of length *N* is as follows:

$$C(u) = \alpha(u) \sum_{x=0}^{N-1} I(x) \cos\left[\frac{\pi(2x+1)u}{2N}\right]$$
$$u = 0, 1, \dots, N-1 \quad (41)$$

$$\alpha(u) = \begin{cases} \sqrt{\frac{1}{N}} & for\ u = 0 \\ \sqrt{\frac{2}{N}} & for\ u = 1 \dots, N-1 \end{cases} \quad (42)$$

However, the two-dimensional DCT applied for an $N \times N$ sample is takes the form:

$$C(u,v) = \alpha(u)\alpha(v) \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} I(x,y)\beta(u,v) \quad (43)$$

Where
$$u, v = 1, 2, \dots, N-1$$
$$\beta(u,v) = \cos\left[\frac{\pi(2y+1)v}{2N}\right] \cos\left[\frac{\pi(2x+1)u}{2N}\right] \quad (44)$$

Face recognition using DCT involves two fundamental steps training and classification.

A. Training stage

The DCT transform is used for facial features extraction. Face images having high correlation and redundant information cause computational burden in terms of processing speed and memory utilization. Therefore, the 2D blocked-DCT segments an image non-overlapping blocks and applies the DCT to each block, which results in low frequency and high frequency subbands. Most of the visually significant signal energy lies at low-frequency sub-band which contains the most important visual parts of the image. These coefficients can be used as a type of signature that is useful for face recognition [21].
The training face images are analyzed on block by block basis. Given an image block I (x, y), where $x, y = 0, 1, \dots, N-1$, we decompose it in terms of orthogonal 2D DCT basis functions. The result is a $N \times N$ matrix $C\ (u, v)$ containing DCT coefficients. The first element, value of C (0, 0) is the average of all the samples in the input image and is referred to as Direct Current (DC) component. The remaining elements in C (*u*, *v*) each indicate the amplitude corresponding to the frequency component of I(x, y), and are defined as Alternate Current (AC) coefficients. It is well-known that the DC coefficient is only dependent to the brightness of the image. Consequently, it becomes DC-free (i.e., zero mean) and invariant against uniform brightness change by simply removing the DC coefficient [22],[23].
The DCT coefficients with large magnitude are mainly located in the upper-left corner of the DCT matrix. Accordingly, we scan the DCT coefficient matrix in a zig-zag manner starting from the upper-left corner (reflecting the amount of information stored) and subsequently convert it to a one-dimensional (1-D) vector [23],[24] as given in Figure.4. For block located at (b, a) the DCT feature vector is composed of:

$$x = \left[C_0^{(a,b)} C_1^{(a,b)} \dots\dots\dots\dots\dots\dots\dots\dots C_{M-1}^{(a,b)}\right]^T \quad (45)$$

Where $C_n^{(a,b)}$ denotes the n^{th} DCT coefficient and M is the number of retained coefficients. The DCT coefficients obtained from each block are concatenated to construct the final feature vector for each image in the training set.

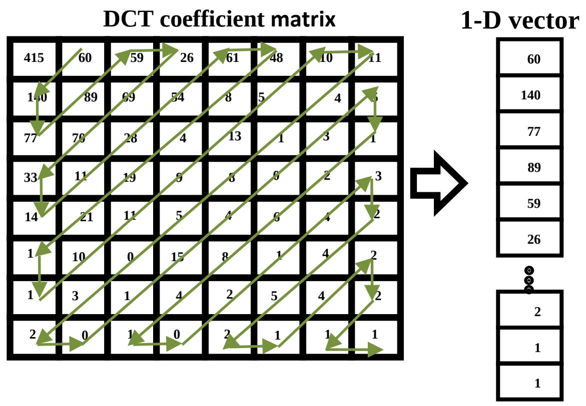

Fig1: Scheme of converting 2D DCT coefficients to 1-D vector

B. Recognition Stage

The test face image is decomposed using DCT to form its feature vector following the same steps done with the training set. Then, its distances to all stored feature vectors are calculated. The face which has the minimum distance with the test face image is labeled with the identity of that image.

2.5. Face recognition using Discrete Wavelet Transform (DWT)

In recent years, wavelet transformation is being increasingly used in signal processing, image processing, computer vision and pattern recognition etc.

2-D wavelet decomposition is carried out by applying a 1-D transform to the row of the original image data and the columns of the row-transformed data respectively. The image can be decomposed into four sub-images.

Sub-image LL denotes low frequency component of original face image and it is smoothing similar image of original image. In sub-image LH, sharp changes in the horizontal direction, i.e., vertical edges. However, in the sub-image HL, sharp changes in the vertical direction, i.e., horizontal edges. In the sub-image HH, sharp changes in non-horizontal, non-vertical directions, i.e., inclined edges. Figure.2 is 2-D wavelet decomposition of a face image. First, we decompose the original face images (as shown in Figure 2.(a)) into four sub-images via one level wavelet decomposition as shown in Figure 2.(b). Then we decompose sub-image LL again, and we obtained 2-level decomposition image of original image as shown in Figure 2.(c). We can decompose sub-image LLL again, and we obtain 3-level decomposition image of original image as shown in Figure 2. (d). Following the process, we can make multiplayer wavelet decomposition.

While in the wavelet decomposition only LL is further decomposed, in the wavelet packet decomposition all LL, LH, HL and HH are further decomposed [25]. Figure.3.(a) shows the wavelet packet decomposition at level 1 of the original image given in Figure.2.(a) and Figure.3.(b) shows the wavelet packet decomposition at level 1.

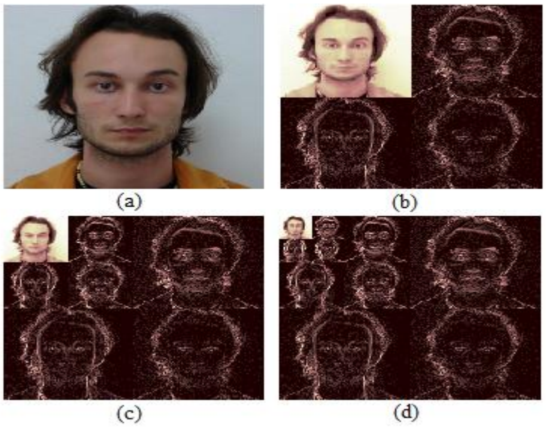

Fig2: (a) original image, Wavelet Decomposition
(b) at levels 1 (c) at levels 2, and (d) at level 3

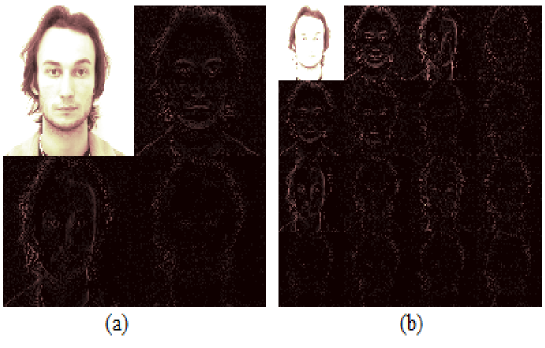

Fig3: Wavelet Packet Decomposition
(a) at levels 1 (b) at level 2

Face recognition system based on DWT consists of two processes, training and recognition processes. In the training process; we adopt db4 wavelet to decompose images of training set. After appropriate wavelet decomposition, we obtain the sub-image LL which is low dimension image. And we can obtain the wavelet feature vector. In recognition process, for an unknown image, we find wavelet feature vector from its sub-image LL after appropriate wavelet decomposition. According to the Nearest Neighbor Classifier, we can recognize the unknown face image.

2.6. Face recognition using Wavelet Packet Decomposition (WPD)

Steps involved in face recognition system using WPD are as follow:

Step1: Decompose the original images into 16 parts using the Wavelet Packet Decomposition at level 2.

Step2: form the feature vectors of the training images by fusing only the low frequency subimages

Step3: To classify the test image, its feature is formed the same as it is done with the training images.Then compute the Euclidean distance between the test image and all the training images. The face that best match the test image is that which has the minimum distance.

3. RESULTS AND DISCUSSION

In the experimentation, the FEI face database is used. It contains a set of face images taken between June 2005 and March 2006 at the Artificial Intelligence Laboratory of FEI in São Bernardo do Campo, São Paulo, Brazil. There are 14 images for each of 200 individuals, a total of 2800 images. Figure.4 shows some examples of image variations from the FEI face database.

Fig4: Some examples of image variations from the FEI face database.

FEI database is divided on two parts. First part represented authorized persons and has 10 persons, from which random 5 images were used for training (total 50 images), other 5 for testing (total 50 images). Second part represented unauthorized persons. It has 10 persons and 5 images per person (total 100 images) only for testing purposes. Thus system has 50 images for training, and 100 for testing (50 authorized and 50 unauthorized).

3.1. Adding some effects

In this work, it is wanted to evaluate the consequence of adding blur, noise, motion and combination of these effects on the performance of face recognition using different approaches namely PCA, FLD, SVD, DCT, DWT and WPD.

Figure 5 illustrates these effects which are obtained as follows:

Blur: To introduce blur effect on an image, an average filter is applied three times with a window of size 12*x*12 (figure 5 (b)).

Motion: The motion is obtained using the Matlab software function simulating this effect. This behaves as a filter, when convolved with an input image produces the effect of a linear motion of a camera by *len* pixels with an angle *theta* degrees in a counterclockwise direction. The effect produced when assuming *len*=30 and *theta*=45 is illustrated in figure.5 (c).

Noise:Two types of noise are used in this simulation: the *Saltand Pepper* type noise with a noise density d=0.04(Figure 5 (d)) and Gaussian noise with mean m=0, variance v=0.01 (figure 5 (e)) and mean m=0, variance v=0.04 (figure 5 (f)).

Furthermore, the following combinations of these effects are performed.

Blur and motion (BM): This is obtained by applying successively blur and motion effects (figure 5 (g)).

Noise and Motion (NM):This is obtained by applying respectively noise and motion effects (figure 5 (h)).

Noise and Blur (NB): this effect, illustrated in figure 5 (i) is produced when applying successively noise and blur to the original image.

Blur, Motion and Noise combination (BMN): This effect is obtained by applying successively the Blur, Motion and Noise effects (figure 5 (j)).

(a) No effect **(b)**Blur effect (c) Motion effect

(d) Salt&pepper Noise **(e)** Gaussian Noise m=0,v=0.01 **(f)** Gaussian Noise m=0, v=0.04

(g) Blur & Motion **(h)**Noise& Motion **(i)** Noise&Blur

(j) Blur & Motion & Noise

Fig5:Adding effects

3.2. Evaluation basis

In order to evaluate the performance of the methods when adding the different effects, we have used following factors: The false acceptance rate, or FAR, is the measure of the likelihood that the biometric security system will incorrectly accept an access attempt by an unauthorized user. The false rejection rate, or FRR, is the measure of the likelihood that the biometric security system will incorrectly reject an access attempt by an authorized user. The values of FAR and FRR are computed using:

$$FAR = \frac{n_{ac}}{n_u} \quad FRR = \frac{n_{re}}{n_a}$$

where n_a is the number of access attempts by an authorized user, n_u is the number of access attempts by an unauthorized user, n_{ac} is the number of acceptances for unauthorized users and n_{re} is the number of rejections for unauthorized users. In our experiment, $n_a = 50$ and $n_u = 50$.

The Receiver Operating Curve (ROC) plots FAR versus FRR. The Equal Error Rate (EER) is defined as the point where the value of FAR equals the value of FRR.

In these experiments, we have evaluated the methods (PCA, FLD, SVD, DCT, DWT and WPD) using the EER criteria.

In this work, the methods are first compared without adding any effect and then, when adding Blur, Motion, Noise(salt & pepper noise, Gaussian). Further, combinations of these effects (Blur and Motion, Noise and Motion, Noise and Blur and Blur and Motion and Noise) have been performed and analyzed.

3.3. Results

The obtained experiments results are summarized in Tables1-10. The ROC curves of the methods with no effect and different effects are shown in figures 6-15.

3.5.1. Without any effect

In the first experiment, we have compared between the methods (PCA, FLD, SVD, DCT, DWT and WPD) without adding any effect using the EER. We have found that DWT gives the lowest EER of 0.12.

Table 1.EER of the methods without any effect

Method	EER
PCA	0.20
FLD	0.32
SVD	0.30
DCT	0.23
DWT	0.12
WPD	0.16

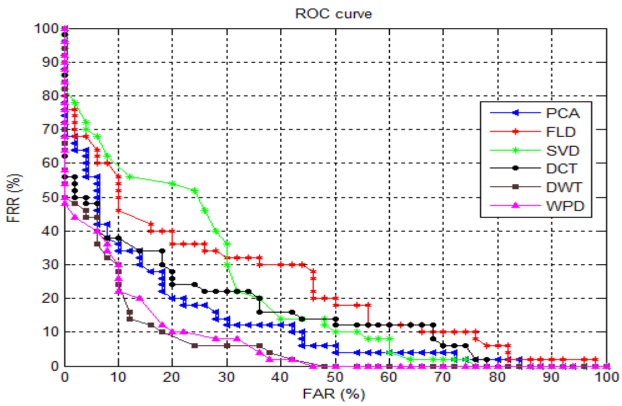

Fig6: ROC of the methods without any effect

3.5.2. Blur effect

In the second experiment, we have compared between the methods (PCA, FLD, SVD, DCT, DWT and WPD) with Blur effect using the EER. We have found that PCA gives the lowest EER of 0.12.

Table 2. EER of the methods with blur effect

Method	**EER**
PCA	0.12
FLD	0.30
SVD	0.39
DCT	0.27
DWT	0.20
WPD	0.16

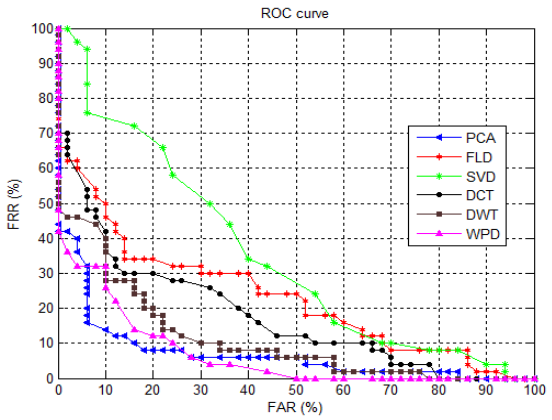

Fig7: ROC of the methods with Blur effect

3.5.3. Motion effect

In the third experiment, we have compared between the methods (PCA, FLD, SVD, DCT, DWT and WPD) with Motion effect using the EER. We have found that PCA and WPD give the lowest EER of 0.12.

Table 3. EER of the methods with motion effect

Method	**EER**
PCA	0.12
FLD	0.30
SVD	0.47
DCT	0.24
DWT	0.14
WPD	0.12

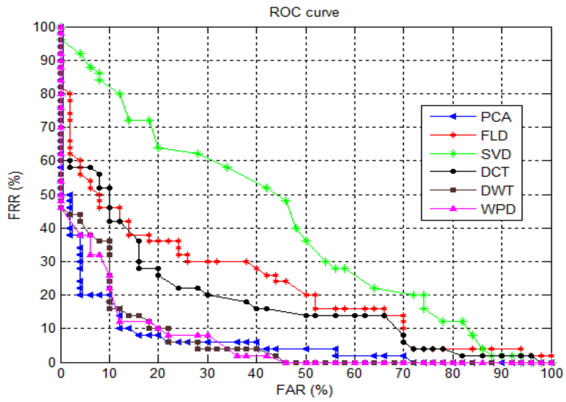

Fig8: ROC of the methods with Motion effect

3.5.4. Noise effect

In the fourth experiment, we have compared between the methods (PCA, FLD, SVD, DCT, DWT and WPD) with Noise effect using the EER.

3.5.4.1. Slat & pepper noise effect

With the salt & pepper noise we have found that PCA gives the lowest EER of 0.14.

Table 4. EER of the methods with salt & pepper noise effect

Method	**EER**
PCA	0.14
FLD	0.33
SVD	0.30
DCT	0.24
DWT	0.19
WPD	0.17

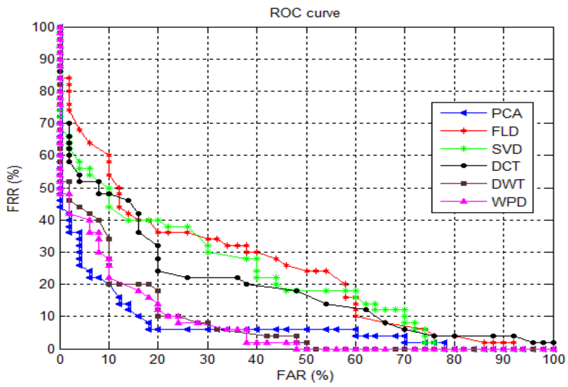

Fig9: ROC of the methods with salt & pepper noise effect

3.5.4.2. Gaussian noise effect

- *Gaussian noise (zero mean and 0.01 variance)*

We have found that PCA gives the lowest EER of 0.14.

Table 5. EER of the methods with Gaussian noise effect (zero mean and 0.01 variance)

Method	**EER**
PCA	0.14
FLD	0.32
SVD	0.37
DCT	0.25
DWT	0.19

WPD	0.16

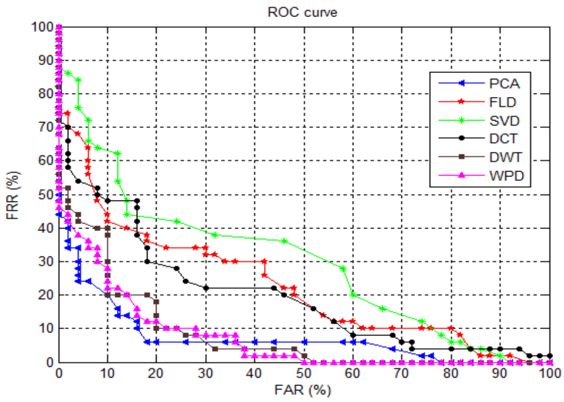

Fig10: ROC of the methods with Gaussian noise (zero mean and 0.01 variance)

- *Gaussian noise (zero mean and 0.04 variance)*
  We have found that PCA gives the lowest EER of 0.14.

Table 6. EER of the methods with Gaussian noise effect (Zero mean and 0.04 variance)

Method	EER
PCA	0.14
FLD	0.34
SVD	0.36
DCT	0.27
DWT	0.18
WPD	0.14

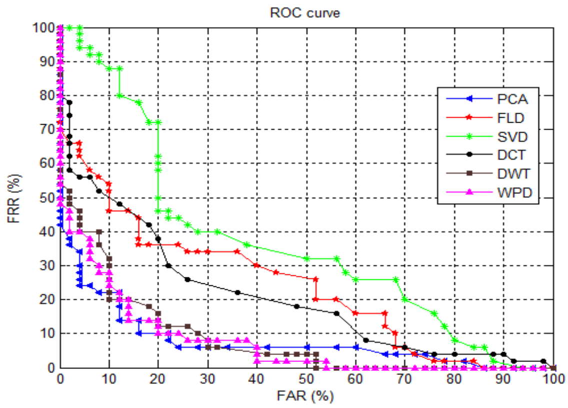

Fig11: ROC of the methods with Gaussian noise (zero mean and 0.04 variance)

3.5.5. BM effect

In the fifth experiment, we have compared between the methods (PCA, FLD, SVD, DCT, DWT and WPD) with *BM* effect using the EER. We have found that PCA gives the lowest EER of 0.12.

Table 7. EER of the methods with *BM* effect

Method	EER
PCA	0.12
FLD	0.29
SVD	0.46
DCT	0.28
DWT	0.16
WPD	0.14

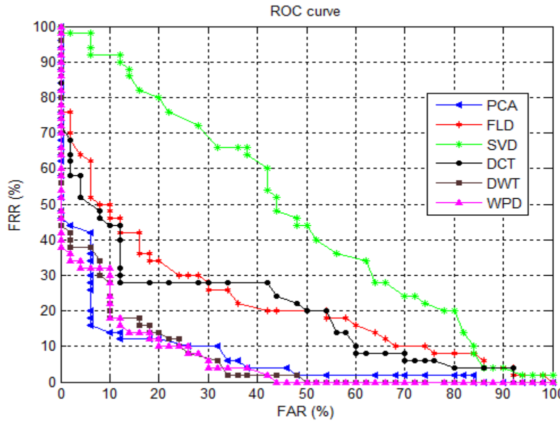

Fig12: ROC of the methods with *BM* effect

3.5.6. NM effect

In the sixth experiment, we have compared between the methods (PCA, FLD, SVD, DCT, DWT and WPD) with *NM* effect (the noise used here is salt & pepper) using the EER. We have found that PCA gives the lowest EER of 0.12.

Table 8. EER of the methods with *BM* effect

Method	EER
PCA	0.12
FLD	0.30
SVD	0.52
DCT	0.24
DWT	0.14
WPD	0.14

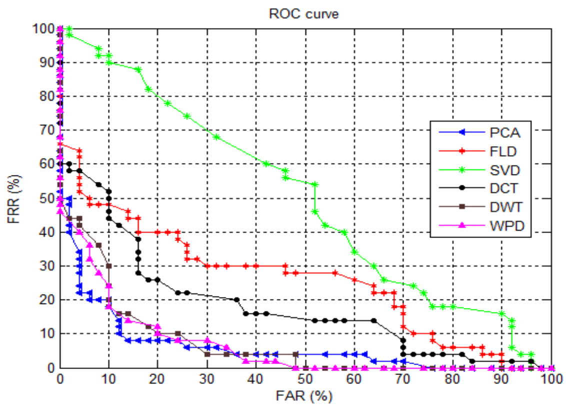

Fig13: ROC of the methods with *NM* effect

3.5.7. NB effect

In the seventh experiment, we have compared between the methods (PCA, FLD, SVD, DCT, DWT and WPD) with *NB* (the noise used here is salt & pepper) effect using the EER. We have found that PCA gives the lowest EER of 0.12.

Table 9. EER of the methods with *NB* effect

Method	EER
PCA	0.12
FLD	0.32
SVD	0.44
DCT	0.28

DWT	0.18
WPD	0.16

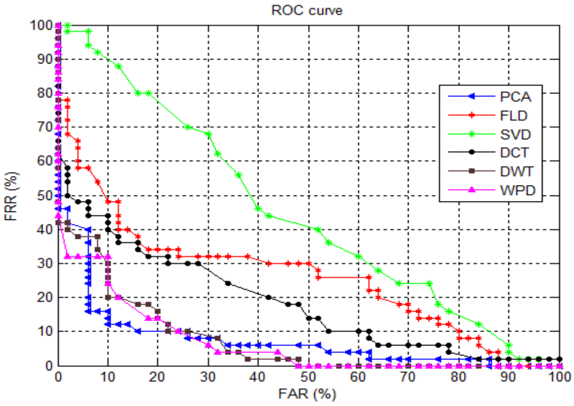

Fig14: ROC of the methods with *NB* effect

3.5.8. BMN effect

In the last experiment, we have compared between the methods (PCA, FLD, SVD, DCT, DWT and WPD) with *BMN*effect (the noise used here is salt & pepper) using the EER. We have found that PCA gives the lowest EER of 0.13.

Table 10.EER of the methods with *BMN* effect

Method	EER
PCA	0.13
FLD	0.34
SVD	0.38
DCT	0.30
DWT	0.16
WPD	0.16

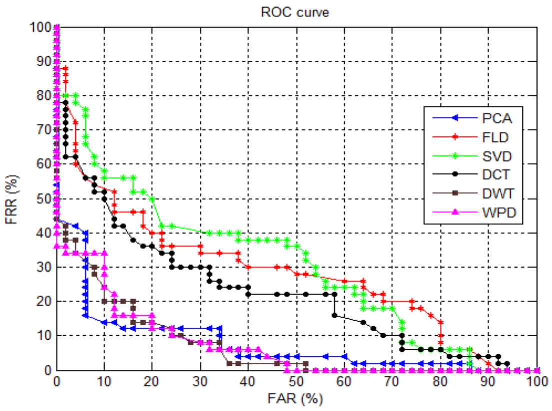

Fig15: ROC of the method with *BMN* effect

4. CONCLUSION

In this work, different global appearance face recognition based methods have been compared with and without added effects. The analysis concerns PCA, FLD, SVD, DCT, DWT and WPD methods and it is based on Equal Error Rate (EER) criteria. In the case where no effect is added, it has been observed that DWT technique provides the lowest EER. Whereas, when adding some specific effects such as Blur, Noise, MN, BM, NB, BMN, the PCA approach produces the minimum EER. On the other hand, in case of Motion effect, high performance is obtained using WPD.
Based on the multiplicity of the investigated methods and assumed effects, we believe that this work provides a tool for engineers to select the most suitable global appearance face recognition technique for the specified application.

Finally, the robustness of these approaches will be evaluated in a future work by considering local appearance face recognition techniques.

5. REFERENCES

[1] Zhao W. and Chellappa R. 2006. Face processing advanced modeling and methods.Elsevier.

[2] HeiseleB., Ho P., WuJ., and PoggioT. 2003. Face recognition: component-based versus global approaches, Comput. Vision Image Understand. Vol. 91, No. 1-2, pp. 6–21.

[3] KimC., Oh J. and ChoiC.H. Combined Subspace Method Using Global and Local Features for Face Recognition. Proceedings of International JointConference on Neural Networks, Montreal, Canada, July 31 - August 4, 2005.

[4] Ullman S.,Vidal-NaquetM. and SaliE. 2002.Visual features of intermediate complexity and their use in classification.Nature neuroscience. Vol. 5, No. 7, pp. 682-687.

[5] Hazim K. and Rainer S.2005. Local Appearance based Face Recognition Using Discrete Cosine Transform. Proceedings of the 13th European Signal Processing Conference (EUSIPCO), Antalya, Turkey.

[6] PentlandA.,MoghaddamB. and Starner T. 1994. View-Based and Modular Eigenspaces for Face Recognition. Proc. IEEE Computer Society Conf. Computer Vision and Pattern Recognition, pp. 84-91.

[7] PenevP. S. and AtickJ. J. 1996. Local feature analysis: A general statistical theory for object representation. Network: Computation in Neural Systems, vol. 7, No. 3, pp. 477-500.

[8] Bronstein A.M.,BronsteinM.M., KimmelR. 2003. Expression-invariant 3D face recognition, in: International Conference on Audio-and Video-Based Person Authentication (AVBPA 2003), LNCS, vol. 2688, pp. 62–70.

[9] Toygar Ö.Acan A. 2003. Face recognition using PCA, LDA and ICA approaches on colored images, journal of electrical & electronics engineering,Istanbul university, vol. 3, No.1 ,pp 735-743.

[10] Te-Hsiu Sun, Mingchih Chen, Shuchuan Lo, Fang-ChihTien. 2007. Face recognition using 2D and disparity eigenface, Expert Systems with Applications: An International Journal, v.33 n.2, p.265-273.

[11] TurkM. and PentlandA. 1991. Face recognition using eigenfaces, Proc. IEEE Conf.on Computer Vision and Pattern Recognition, p. 586-59.

[12] TurkM. and PentlandA. 1991.Eigenfaces for recognition,Journal of Cognitive Neuroscience, vol.3, No.1.

[13] LiuC. and WechslerH. Face recognition using Evolutionary pursuit. Appears in the fifth European conference on computer vision, ECCV'98, University of Freiburg, Germany, June 2-6 1998.

[14] EtemadK..andChellappa R. 1997. Discriminant analysis for recognition of human face images, J. Opt. Soc. Am. A, vol. 14, No. 8, pp. 1724-1733.

[15] HafedZ. M. and Levine M. D. 2001. Face recognition using the discrete cosine transform. International Journal of ComputerVision, vol. 43, No. 3, pp. 167-188,

[16] Ye. J. 2004.Generalized low rank approximations of matrices. in the Proceedings of the Twenty-First International Conference on Machine Learning (Outstanding Student Paper), Banff, Alberta, Canada, pp. 887-894.

[17] Thaahirah S., Rasied,Khalifa M.O. and KamarudinY.B. 2005. Human face recognition based on singular valued decomposition andneural network. Proceeding of the GVIP 05Conference, Dec. 19-21, CICC, Cairo, Egypt, pp. 1-6.

[18] Wang Y. H.,TanT. N. and ZhuY. Face Verification Based on Singular Value Decomposition and Radial Basis Function Neural Network.National Laboratory of PatternRecognition (NLPR), Institute of Automation, Chinese Academy of Sciences.

[19] Tjahyadi R., Liu W. andVenkatesh S., 2004. Application of the DCT Energy Histogram for Face Recognition, Proc. of the 2nd International Conference on Information Technology for Applications, ICITA 2004. pp. 305-310.

[20] PanZ. and BolouriH.1999. High Speed Face Recognition Based on Discrete Cosine Transforms and Neural Networks.Technical report, Univ. of Hertfordshire, September 27, 1999.

[21] Jawad N., Syed K. A., and Farrukh N. 2008. Pose invariant Face Recognition using Hybrid DWT-DCT Frequency Features with Support Vector Machines, in Proc. of the 4th International Conference on Information Technology and Multimedia at UNITEN (ICIMu), Nov. 18-19, 2008, Bandar BaruBangi, Selangor, Malaysia, pp. 99–104.

[22] Chen,Y.,Jiang,S. and Abraham, A. 2005.Face Recognition Using DCT and Hybrid flexible Neural Tree. In Proc. of the International Conference on Neural Networks and Brain. pp. 1459-1463,

[23] Choi,J., Chung,Y., Kim,K. and Yoo, J. 2006.Face Recognition Using Energy Probability in DCT Domain; IEEE. pp. 1549-1552. ICME.

[24] Hassan M.,OsmanI. and YahiaM.2007. Walsh-Hadamard Transform for Facial Feature Extraction in Face Recognition.International Journal of Computer and Information Science and Engineering, Vol.1, No. 3, pp 167- 171.

[25] Chao-Chun L., Dao-Qing, D. and Hong, Y. 2007. Local Discriminant Wavelet Packet Coordinates for Face Recognition.Journal of Machine Learning Research, v8,1165-1195.