Facial Emotion Feature Extraction Based Eigenface for 3D Video

Salwa A. Al-agha[1], Hilal H. Saleh[2], Rana F. Ghani[3]
Department of Computer Science, University of Technology
Baghdad, Iraq

[1]*salwa_alagha2009@yahoo.com*, [2]*hhsrq888@yahoo.com*, [3]*ranafghany@yahoo.com*

ABSTRACT

Recent psychological research has shown that facial expressions are the most expressive way in which humans display emotion. Facial expressions are widely used in the behavioral interpretation of emotions, cognitive science, and social interactions. Therefore, automated and real-time facial expression recognition would be useful in many applications, such as human computer interfaces, virtual reality, video-conferencing, customer satisfaction studies, etc. This paper presents a proposed technique for facial expression extraction, which based on the Appearance Features Technique - Principle Component Analysis (PCA), which depending on extract features (largest Eigenvalues and Eigenvectors).
Experimental results show the quick technique for feature extraction of 3D video frames, which takes 5.1 seconds in the process of feature extraction.

KEYWORDS: PCA, feature extraction, eigenface, facial emotion expression, 3D video

1. INTRODUCTION

Emotion recognition from video is becoming an essential approach in modern human computer interaction (HCI) systems [1], and face recognition has received significant attention in the past decades due to its potential applications in biometrics, information security, law enforcement, etc. [2]. In recent years there has been a growing interest in developing all aspect of human computer interaction (HCI) systems. A challenging aspect of future HCI is to give the computer more human like capabilities, such as emotion recognition [1].

Many researchers have worked on geometric-based as well as holistic-based face recognition approaches. In case of geometric-based approaches, the local features such as mouth, eyes, eyebrows, and nose are initially extracted from face images. These local statistics and locations are used for classification [3]. The most successful geometric-based methods are elastic bunch graph matching (EBGM) and Active Appearance graph Models (AAM) [4].

Most approaches to automatic facial expression analysis attempt to recognize a small set of prototypic emotional facial expressions, i.e., fear, sadness, disgust, anger, surprise, and happiness [5], [6]. This practice may follow from the work of Darwin [7], and more recently Ekman [8], who suggested that basic emotions have corresponding prototypic expressions. From several methods for recognition of facial gestures, the facial action coding system (FACS) [9] is the best known and most commonly used in psychological research [10]. The changes in the facial expression are described with FACS in terms of 44 different action units (AUs), each of which is anatomically related to the contraction of either a specific facial muscle or a set of facial muscles. Along with the definition of various AUs, FACS also provides the rules for AU detection in a face image [11].

In contrast to geometric-based methods, holistic-based approaches extract the holistic features of the whole face region. They consider all the pixels of a face image of size n × m and represented by a vector in an n × m-dimensional space. However, these vector spaces are too large, which increases the computational complexity in face recognition system. A common way to overcome this problem is the dimensionality reduction techniques [12]. The most popular dimensionality reduction techniques are principal component analysis (PCA), which is a statistical approach used to express the faces in a subset of their eigenvector [13].

The scheme is based on an information theory approach that decomposes face images into a small set of characteristic feature images called 'Eigenfaces', which are actually the principal components of the initial training set of face images [14].

In this paper, Eigenface method has been presented to design and extract an Eigenvector based facial expression feature extraction system from the 3D video.

Eigenvector based features are extracted from the sequences of video frames, in training phase, a set of 65 3D images (database) for each basic expression (presented 455 3D training images) is processed and eigenvectors specific to the expressions are stored. In the testing phase, the eigenvectors of the testing 3D video frames are computed.

Recently, a publicly available 3D facial expression database, called BOSPHOPUS DATABASE, was collected by Arman Savran, Boğazicin University, was used in this paper [15].

2. EIGENVALUES AND EIGENVECTORS APPROACH

Eigenvectors of a linear operator are non-zero vectors which, when operated results in a scalar multiple of them. The scalar is then called the eigenvalue (λ) associated with the eigenvector (X). Eigenvector is a vector that is scaled by a linear transformation. It is a property of a matrix. When a matrix acts on it, only the vector magnitude is changed not the direction.

$$AX = \lambda X \tag{1}$$

Where A is a Vector function.

2.1 Calculations of Eigenvalues and Eigenvectors

By using Equation (1), Equation (2) presented as:

$$(A - \lambda I)X = 0 \tag{2}$$

Where I is the n x n Identity matrix. This is a homogeneous system of equations, and from fundamental linear algebra, we know that a nontrivial solution exists if and only if,

$$\det(A - \lambda I) = 0 \tag{3}$$

Where det ( ) denotes determinant. When evaluated, becomes a polynomial of degree n. This is known as the characteristic equation of A, and the corresponding polynomial is the characteristic polynomial. The characteristic polynomial is of degree n. If A is n x n, then there are n solutions or n roots of the characteristic polynomial. Thus there are n eigenvalues of A satisfying the Equation (4),

$$AXi = \lambda Xi \tag{4}$$

Where I =1, 2, 3….n

If the eigenvalues are all distinct, there are n associated linearly independent eigenvectors, whose directions are unique, which span an n dimensional Euclidean space [14].

3. PRINCIPAL COMPONENT ANALYSIS (PCA) APPROACH

Eigenfaces employed principal component analysis (PCA). PCA is an unsupervised learning method, which treats samples of the different classes in the same way [16].

Eigenvectors is dependent on the concept of orthogonal linear transformation, which is a non-zero vector and the dominant Eigenvector of a matrix is the one corresponding to the largest Eigenvalue of that matrix. This dominant Eigenvector is important for many real world applications.

PCA steps of feature extraction for facial expressions are as follows:

1. Organizing the data set- Consider the data having a set of M variables that are arranged as a set of N data vectors. Thus the whole data is put into a single matrix X of dimensions M x N.
2. Calculating the mean of matrix X:

   $$\mu_{x=\frac{1}{N}} \sum_{n=1}^{N} X[m,n] \tag{5}$$

   Where μ_x is the mean of the matrix X, m=1, 2… M and n=1, 2… N.
3. Subtracting the mean for each dimension:

$$X = X - \mu_x \tag{6}$$

The new matrix X comprises of the mean-subtracted data. The subtraction of mean is important, since it ensures that the first principal component indicates the direction of maximum variance.

4. Calculating the covariance matrix:

   Covariance has the same formula as that of the variance. Assume a 3-dimensional data set (p, q, r), then covariance either between p and q, q and r or r and p dimensions can be measured. But measuring the covariance between p and p, q and q, r and r dimensions gives the value of variance of the respective p, q, r dimension. Variance is measured on a single dimension were as covariance on multi-dimensions.

   - For 1-dimension:

     $$\mathrm{Var}(x) = \frac{\sum_{i=1}^{N}(X-\mu_x)(X-\mu_x)}{N-1} \tag{7}$$

     Where *Var* is the variance matrix;

   - For 2-dimension say (x, y):

     $$\mathrm{Cov}(x,y) = \frac{\sum_{i=1}^{N}(X-\mu_x)(Y-\mu_y)}{N-1} \tag{8}$$

     Where *Cov(x, y)* is the covariance matrix; μ_y is the mean of another matrix Y.

5. Calculating the Eigenvectors and Eigenvalues of the covariance matrix- For computing the matrix of Eigenvectors that make the diagonal of covariance matrix C

   $$E \cdot \mathrm{Cov} \cdot E^{-1} = D \tag{9}$$

   Where Cov is the covariance matrix; E is the matrix of all the Eigenvectors of Cov, one Eigenvector per column; D is the diagonal matrix of all the Eigenvalues of Cov along its main diagonal, and which is zero for the rest of the elements [17].

4. THE PROPOSED TECHNIQUE

In the following steps, a method for extracting features from 3D video by using Eigenface approach:

4.1 Video Capturing

Video is captured using 3D full HD digital camera with MP4 file format as testing video frames.

4.2 Face Detection

The detection of the face is based on a detector using Haar-like features trained by an An Adaboost cascade classifier, which is applied that detects the subject's face for the sequences of testing video frames, this by using the Viola-Jones algorithm:

- Integral image for feature computation.
- Adaboost for feature selection.
- An attentional cascade for efficient computational resource allocation.

And then, cropped the face region from the rest background of the whole frame.

4.3 Database Images Loading

A set of 65 3D images (database) of each individual person (man, woman), for each basic expression is loading as training images.

4.4 Resizing Images

Resizing training images database which loading, and testing video frames at the same size in order to process them at the same condition.

4.5 RGB to Gray-scale color image convertor

For each training images and testing video sequences frames, convert RGB color model to gray-scale model.

4.6 Feature Extraction

In the proposed technique, feature vector will be computed for training images and testing video sequences frames by using PCA approach, as follows:

1. For each cropped face from the sequences video frames, compute the following:
   a. Eigenvalues will be computed for each diagonal of the matrix of frame: first mean vectors is obtained for each dimension, then subtracting the mean from the cropped frame. Normalize the data by dividing each feature by its standard deviation. This is especially important if different features correspond to different metrics. Then covariance matrix is obtained by calculating the covariance of the normalized, subtracted mean vectors matrix. Ordered the eigenvalues from the largest to lowest one, where the largest eigenvalues find the most variance in the frame.
   b. Sort the eigenvectors depending on eigenvalues and choose k eigenvectors with the largest eigenvalues to form: V = d x k

      d (original data in each frame), k chosen eigenvectors, V dimensional matrix (where every column represents an eigenvector).
   c. Apply the projection process for data to the new axes (eigenvectors), use V eigenvectors matrix to transform data onto the new subspace. Then the projected data D' is obtained as $D' = V^T * D$
2. For each training image in BOSPHOPUS DATABASE:
   a. Eigenvalues will be computed for each diagonal of the matrix of training image: first mean vectors is obtained for each dimension, then subtracting the mean from the 3D training image. Normalize the data by dividing each feature by its standard deviation.
   b. Then covariance matrix is obtained by calculating the covariance of the normalized, subtracted mean vectors matrix.
   c. Ordered the eigenvalues from the largest to lowest one, where the largest eigenvalues finds the most variance in the training image.
   d. Sort the eigenvectors depending on eigenvalues and choose k eigenvectors with the largest eigenvalues to form: V = d x k
   e. d (original data in each database image), k chosen eigenvectors, V dimensional matrix (where every column represents an eigenvector).
   f. Apply the projection process for data to the new axes (eigenvectors), use V eigenvectors matrix to transform data onto the new subspace.

   Then the projected data D' is obtained as $D' = V^T * D$.

5. EXPERIMENTAL RESULT

The proposed technique is used for feature extraction for facial expression in 3D video frames. This proposed system is implemented on well-known 3D facial expression database (BOSPHOPUS DATABASE), Fig. 1. a, b shows the file of this database and some examples of them individually.

3D training images color model, convert from RGB to gray-scale color model, as shown in the Fig. 2.

Fig. 3. a, b represent the plot of pixel values for colored and gray-scale 3D training one image. The mean for each dimension are obtained for each 3D training image in the database file, as shown in the Fig. 4. Subtracting the mean of each dimension from the 3D training image, as shown in the Fig. 5.

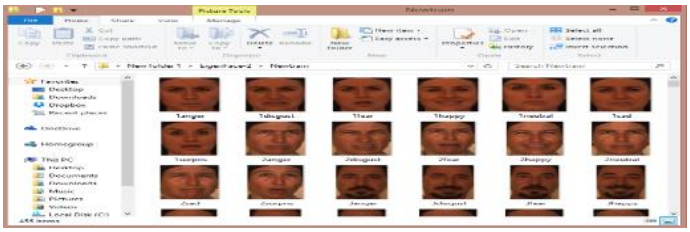

Fig. 1. a. BOSPHOPUS DATABASE file

Fig. 1. B. Examples of some training images (BOSPHOPUS DATABASE).

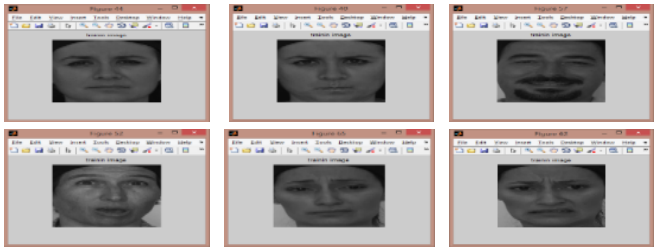

Fig. 2. 3D training images as gray-scale color model.

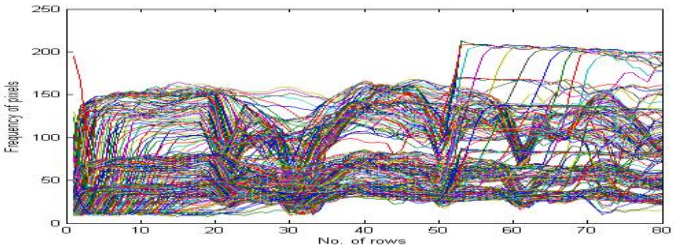

Fig. 3. a. pixel values for one colored 3D training image.

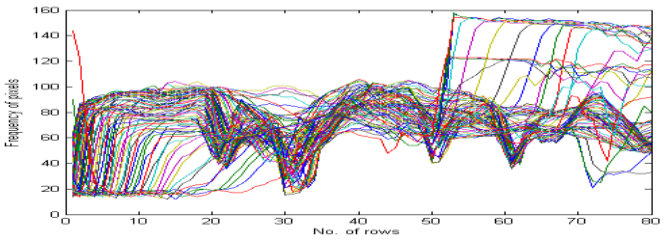

Fig. 3. b. pixel values for one gray-scale 3D training image.

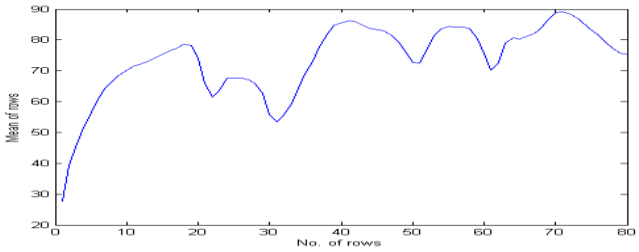

Fig. 4. Mean for x-dimension for one 3D training image.

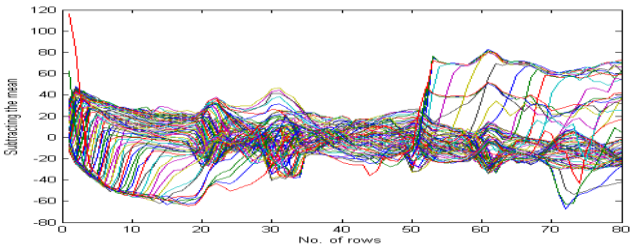

Fig. 5. Subtracting the mean of x-dimension for one 3D training image.

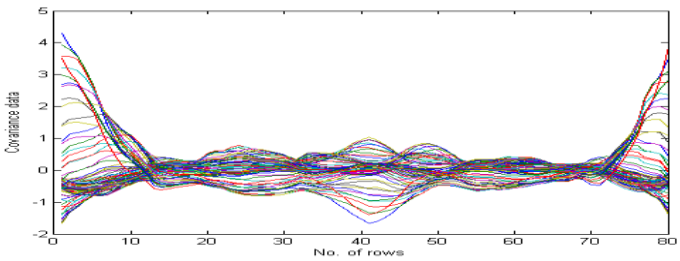

Fig. 6. Data normalization for one 3D training image.

Normalize the data by dividing each feature by its standard deviation, as shown in the Fig. 6.

Covariance matrix is obtained for all 3D training images, by calculating the covariance of the normalized matrix, as shown in the Fig. 7.

Eigenvalues obtained for each diagonal of the covariance matrix, as shown in the Fig. 8. Then ordered eigenvalues from the largest to lowest one, and eigenvectors is obtained for the largest eigenvalues (select 20 largest eigenvalues), as shown in the Fig. 9.

Projected data to the new axes (selected eigenvectors), transform data onto the new subspace, as shown in the Fig. 10.

3D Video is captured as testing video frames (first video frame with 105 frames, the second one with 39 frames), face detection is obtained for two side of 3D frames, as shown in the Fig. 11. And cropping process is applied for each side of frame, as shown in the Fig. 12.

Frames of video color model, convert from RGB to gray-scale color model, as shown in the Fig. 13.

Fig. 14. a, b represent the plot of pixel values for colored and gray-scale for one of cropping frame. The mean of each dimension are obtained for each sequences of cropping frames, as shown in the Fig. 15. Subtracting the mean of each dimension from the cropping frames, as shown in the Fig. 16.

Normalize the data in cropping frames by dividing each feature by its standard deviation, as shown in the Fig. 17.

Covariance matrix is obtained for all cropping frames, by calculating the covariance of the normalized matrix, as shown in the Fig. 18.

Eigenvalues obtained for each diagonal of the matrix of covariance matrix, as shown in the Fig. 19. Then ordered eigenvalues from the largest to lowest one and eigenvectors is obtained for the largest eigenvalues (select 20 largest eigenvalues), as shown in the Fig. 20.

Projected data to the new axes (selected eigenvectors), transform the data onto the new subspace, as shown in the Fig. 21.

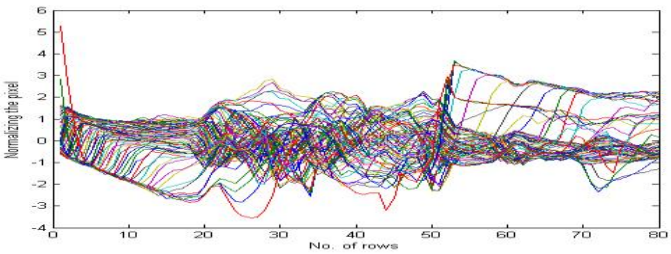

Fig. 7. Covariance matrix for one 3D training image.

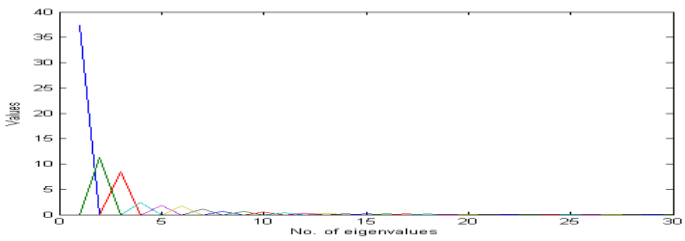

Fig. 8. Eigenvalues for one 3D training image.

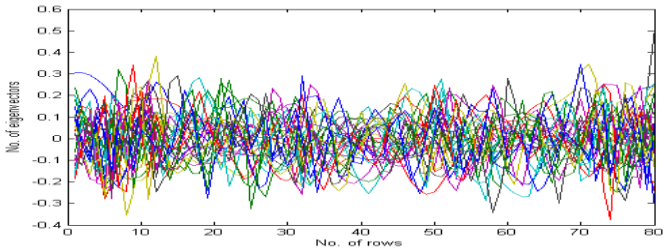

Fig. 9. Eigenvectors for one 3D training image.

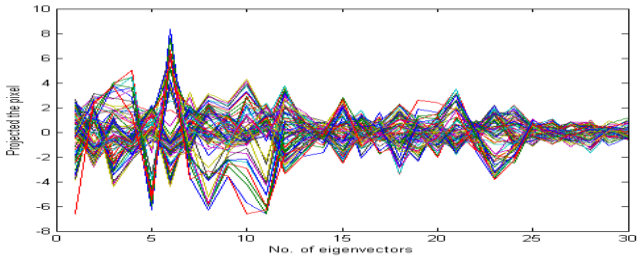

Fig. 10. Projected data to the new subspace for one 3D training image.

Fig. 11. Examples of Face detection process for one frame of different videos.

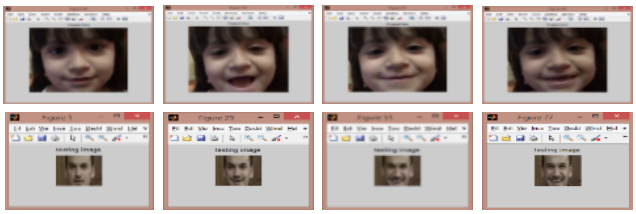

Fig. 12. Examples of Cropping process for multiple frames of different videos.

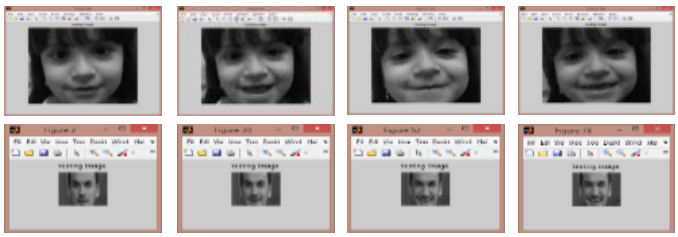

Fig. 13. Examples of cropping frames as gray-scale color model.

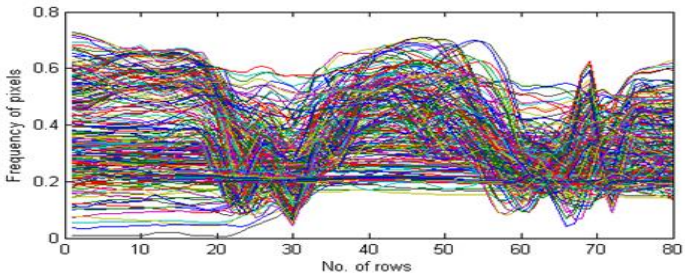

Fig. 14. a. pixel values for one colored cropping frame.

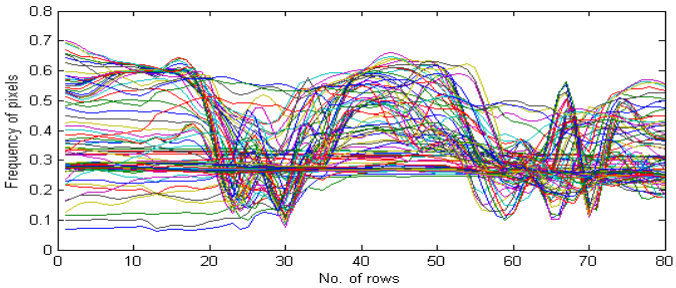

Fig. 14. b. pixel values for one gray-scale cropping frame.

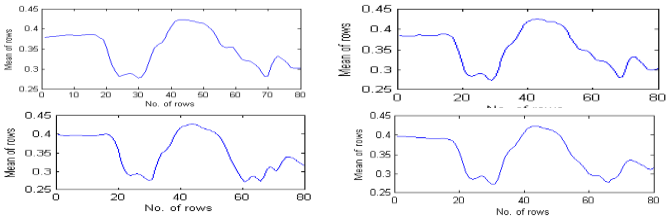

Fig. 15. Mean for x-dimension for some cropping frames.

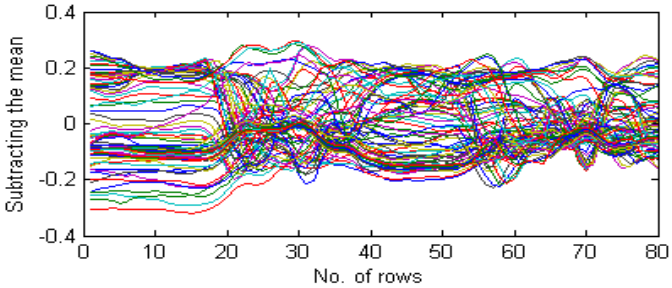

Fig. 16. Subtracting the mean of x-dimension for one cropping frame.

Subtracting the mean of image or video sequences frames, will centered data around the origin of axis, such that its average becomes zero, as shown in Fig. 5. And Fig. 16.

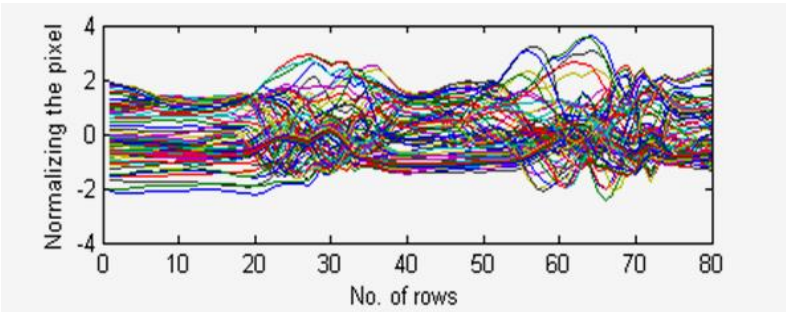

Fig. 17. Data normalization for one cropping frame.

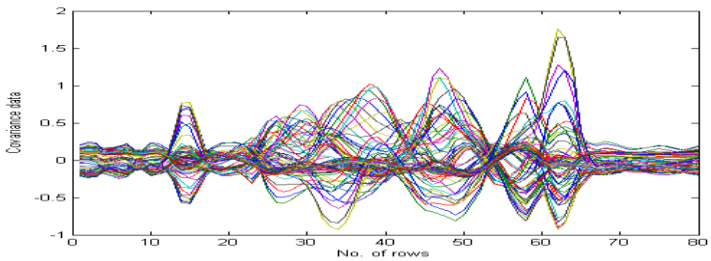

Fig. 18. Covariance matrix for one cropping frame.

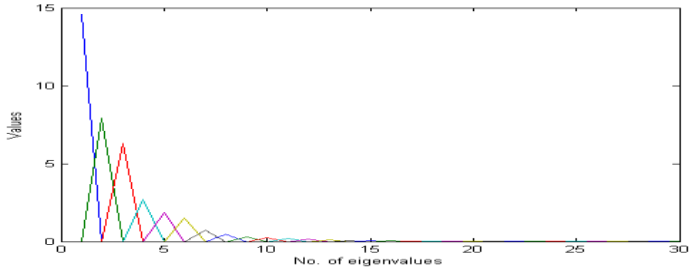

Fig. 19. Eigenvalues for one cropping frame.

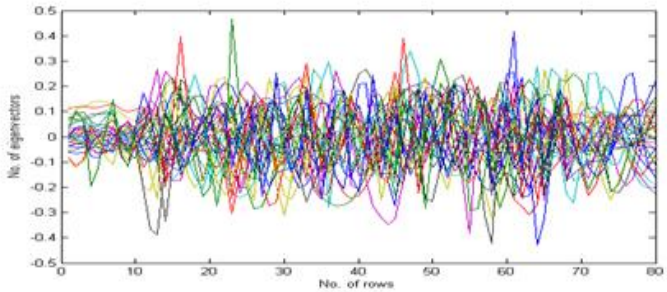

Fig. 20. Eigenvectors for one cropping frame.

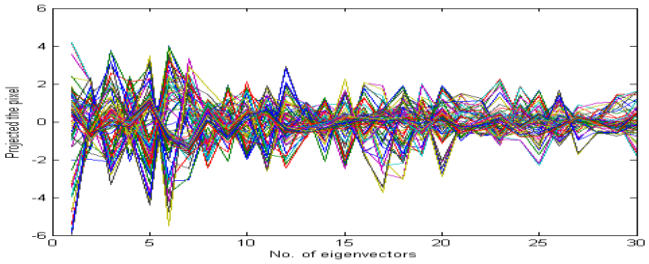

Fig. 21. Projected data to the new subspace for one cropping frame.

Normalize the data to avoid scale-dependent nature of PCA, which especially important if different features correspond to different metrics, as shown in Fig. 6. And Fig. 17.

Covariance matrix is directly related to a linear transformation of uncorrelated data, this linear transformation is completely defined by the eigenvectors and eigenvalues of the data. From Fig. 7. And Fig. 18. all pixels values of training images and video frames are centered on the origin except the values which represent the variance.

Eigenvectors represent the rotation matrix (new axes of projection data for 3D training images and video sequences frames), as shown in Fig. 9, Fig. 20. Eigenvalues correspond to the square of the scaling factor in each dimension as shown in Fig. 8, Fig. 19.

6. CONCLUSION

In this paper, the proposed technique is presented for video feature extraction. This approach using Eigenface and PCA considered the Eigen space of the whole face, which is quite robust in the treatment of face images with varied facial expression. It is also quite simple in the facial emotion computation.

The Eigenface method involves the characteristics and features that distinguish it at work for the rest of the recovery methods characteristics, which is characterized viability recognize faces with age and appearance of the signs of aging. This method is being extracted features for whole face and not characteristic of the regions, so the signs of aging and Lighting does not affect. The extraction process as well as for different age groups.

This proposed work is applied with multiple illumination conditions. Technique has been proposed using MATLAB version R2013b and Intel(R) Core (TM) i3, Windows 8.1 Ultimate (64 bit), and 3D Full HD 20.4 mega pixels with 12 Extended Zoom. Viola-Jones algorithm which used for face detection is very fast and accurate detection algorithm.

In future work, the Eigenvectors for 3D training images (database) and sequences frames of 3D video, will be used as robust features in the emotion recognition system.

REFERENCES

[1] R. Niese, A. Alhamadi, A. Panning, and B. Michaelis (2010), "Emotion Recognition based on 2D-3D Facial Feature Extraction from Color Image Sequences", Journal of Multimedia, **5**(5), 488-500.
[2] V. Rishiwal and A. Gupta (2012), "Improved PCA Algorithm for Face Recognition", World Applied Programming for International e-Conference on Computer Engineering (IeCCE), **2**(1), 55-59.
[3] K. Roy, P. Bhattacharya, and C. Suen (2011), "Iris recognition using shape-guided approach and game theory", Pattern Anal.Appl, **14**(4), 329-348.
[4] S. Shin, S. Kim, and H. Choi (2007), "Generalized elastic graph matching for face Recognition", Pattern Recognit. Lett, **28**(9), 1077-1082.
[5] M. Black and Y. Yacoob (1997), "Recognizing facial expressions in image sequences using local parameterized models of image motion", Comput.Vis, **25**(1), 23-48.
[6] M. Pantic and L. Rothkrantz (2000), "Expert system for automatic analysis of facial expression", Image Vis. Comput. J., **18**(11), 881-905.
[7] C. Darwin (1965) "The Expression of the Emotions in Man and Animals", Chicago IL: Univ. of Chicago Press, 18-72.
[8] M. Lewis and J. Haviland (2000), "Handbook of Emotions", Jones Eds. Guilford Press, New York, 236-249.
[9] P. Ekman and W. Friesen (1978), "Facial Action Coding System", Palo Alto, CA: Consulting Psychol. Press.
[10] K. Scherer and P. Ekman (1982), "Handbook Methods in Non-Verbal Behavior Research, Eds.", Cambridge Univ. Press, Cambridge, MA.
[11] S. Gangwar and K. Kumar (2012), "3D Face Recognition Based On Extracting PCA Methods", International Journal of Engineering Research and Applications (IJERA), ISSN: 2248-9622, **2**(2), 693-696.
[12] A. Matinez and A. Kak (2001), "PCA versus LDA", IEEE Trans. Pattern Anal. Mach. Intell, **23**(2), 228-233.
[13] V. Gaidhane, Y. Hote, and V. Singh (2014), "An efficient approach for face recognition based on common eigenvalues", Pattern Recognition.Elsevier Ltd, 1869-1879.
[14] M. Turk and A. Pentland (1991), "Face Recognition Using Eigenfaces", Proceeding of IEEE Conference on Computer Vision and Pattern Recognition, 586-591.
[15] A. Savran (2009), "BOSPHORUS DATABASE", Copyright © Boğaziçi University.
[16] H. Deng, L. Jin, L. Zhen, and J.Huang (2005), "A New Facial Expression Recognition Method Based on Local Gabor Filter Bank and PCA plus LDA", Motorola Labs Research Foundation. International Journal of Information Technology, **11**(11).
[17] J. Kalita and K. Das (2013), "Recognition of Facial Expression Using Eigenvector Based Distributed Feature and Euclidean Distance Based Decision Making Technique", International Journal of Advanced Computer Science and Applications (IJACSA), **4**(2).