Feature extraction using PCA and Kernel-PCA for face recognition

Hala Mousher Ebied
Ain Shams University

Feature Extraction Using PCA and Kernel-PCA for Face Recognition

Hala M. Ebied
Scientific Computing Department
Faculty of Computer and Information Sciences
Ain Shams University
Cairo, Egypt
e-mail: hala_mousher@hotmail.com

***Abstract*—The face recognition system consists of a feature extraction step and a classification step. In this paper, the researcher studies the use of linear and nonlinear methods for feature extraction in the face recognition system. The linear Principal component analysis (PCA) which is widely used in the face recognition is used to construct the feature space and extract features. The Kernel-PCA is extended from PCA to represent nonlinear mappings in a higher-dimensional feature space. Several parameters of Kernel functions are investigated and expected to affect the recognition performance. The k-nearest neighbor classifier with Euclidean distance is used in the classification step. Our experiments are carried out on the ORL face database which contains variability in expression, pose, and facial details. Experimental results show that Kernel-PCA with Gaussian function can give a correct recognition rate similar to PCA and higher than Kernel-PCA with polynomial function.**

I. INTRODUCTION

Face recognition is a pattern recognition problem. Usually the pattern recognition problem suffers from the large size of raw image (pattern). As a pattern recognition problem, there are two critical issues, that is, how to extract features to represent faces (patterns) and how to classify subjects based on the extracted features [1]. Feature extraction is the key to face recognition, as it is to any pattern classification [2]. Extract features from the raw face saves the time for the classify step.

Feature Extraction from still image can be categorize into appearance-based and model-based face recognition. The model techniques, also known as geometric techniques, used feature values including items such as the distance and angle between geometric points (eyes corners, mouth extremities, nostrils and chine top) [3, 4, 5]. Brunelli and Poggio [6] addressed the problem of gender classification and identification using a geometrical description of faces.

In the appearance-based face recognition, also known as view-based, an image is considered as a high-dimensional vector, i.e., a point in a high-dimensional vector space. Many view-based approaches use statistical approaches and Neural Networks approaches [2, 7, 8, 9]. Most of the systems in the statistical approaches are based on the traditional Eigenfaces (KLT) [10]. Brunelli and Poggio [11] compared a geometric feature-based method with a template matching scheme. It was concluded that the intensity templates are superior in recognition to the geometric feature-based methods.

Different researchers for the face recognition system have proposed many linear and nonlinear statistical techniques. The Principal component analysis (PCA) or Eigenfaces method [10], Independent Component Analysis (ICA) [12], and Linear Discriminant Analysis (LDA) [13], and PCA combined with LDA [14] are the most widely used linear statistical techniques reported by research community. Delac et al. [15] presented an independent, comparative study of three most popular appearance-based face recognition projection methods (PCA, ICA, and LDA) in completely equal working conditions.

The non-linear manifold analysis approaches are used the Kernel method, such as kernel principal component analysis (Kernel-PCA) [16]. Kernel-PCA (KPCA) is an extension nonlinear form of PCA [16], computes the principal components in a high-dimensional feature space, related to the input space by some nonlinear map; and thus can extract non-linear principal components. Kernel methods are utilized in many different research fields, such as noisy interpolation and pattern recognition [17-22].

In this paper, the Kernel-PCA nonlinear statistical technique presents for the face recognition. The experimental results compare with the popular linear PCA statistical technique. The classification step chooses to be the simplest classifier; k-nearest neighbor with Euclidean distance. The remainder of this paper is organized as follows: Section 2 explains the Principal component analysis and the Kernel-PCA. In Section 2 the research describes the classifier k-nearest neighbor function. Section 4 introduces the ORL face databases. The results are discussed in Section 5. Finally, Section 6 concludes the paper.

II. Feature extraction methods

Face recognition has a challenge to perform in real time. Raw face image may consume a long time to recognize since it suffers from a huge amount of pixels. One needs to reduce the amounts of pixels. This is called dimensionality reduction or feature extraction, to save time for the decision step. Feature extraction refers to transforming face space into a feature space. In the feature space, the face database is represented by a reduced number of features that retain most of the important information of the original faces [23]. The most popular method to achieve this target is through applying the Eigenfaces algorithm [10, 24]. The Eigenfaces algorithm is a classical statistical method using the linear Karhumen-Loeve transformation (KLT) (also known as Principal component analysis (PCA)). The KLT calculates the eigenvectors of the covariance matrix of the input face space. These eigenvectors define a new face space where the images are represented [16]. In contrast to linear PCA, schölkopf et al. [16] have developed a nonlinear PCA called Kernel-PCA. The kernel-PCA is not interested in principal components in input space, but rather in principal components of variables which are nonlinearly related to the input variables [25].

A. Principal component analysis

Consider a set of n 2-D training images of size $M \times N$. Each face image is represented as a 1-D column-vector I_i by concatenating each row into a long vector, where $1 \le i \le n$ in a MN-dimensional space. The average face of the set is defined by

$$m = 1/n \sum_{i=1}^{n} I_i \tag{1}$$

Each face differs from the average by vector $\mathbf{x}_i = I_i - m$, $i = 1,\dots, n$. The shifted faces are arranged on a matrix X $=[\mathbf{x}_1\, \mathbf{x}_2 \ldots \mathbf{x}_n]$ of dimension $MN \times n$. The covariance matrix $\mathbf{C}_X$ of the training image set is defined by

$$\mathbf{C}_X = XX^T \tag{2}$$

It is necessary to solve the *eigenvalue problem*

$$\mathbf{C}_X\,\mathbf{U} = U\Lambda \tag{3}$$

where Λ is a diagonal matrix defined by the eigenvalues λ of matrix $\mathbf{C}_X$, that is $\Lambda=\text{diag}[\lambda_1, \lambda_2, \dots, \lambda_{MN}]$, and U is the associated eigenvectors of λ. Now these eigenvectors represent the new face space.

There are MN possible projections of the image vector $\mathbf{x}$,

$$y_j = u_j^T x \qquad ,j = 1,\dots, MN. \tag{4}$$

where the u_j are the eigenvectors of the covariance matrix C_x and the y_j are the projections of $\mathbf{x}$ and called the *principal components* (also known as *eigenfaces*).
The original image vector $\mathbf{x}$ may be reconstruct exactly from the projections y_j as

$$y = [y_1, y_2, \dots, y_{MN}] \tag{5}$$

$$\mathbf{x} = U\,y = \textstyle\sum_{j=1}^{MN} u_j y_j \tag{6}$$

The dimensionality can be reduced by selecting the first n' ($<< MN$) eigenvectors that have large variances and discarding the remaining ones that have small variance. One may then approximate the image vector x by truncating the expansion of Eq. (6) after m terms as follows:

$$\hat{\mathbf{x}} = \textstyle\sum_{j=1}^{n'} u_j y_j \tag{7}$$

Therefore, a few numbers of eigenvectors provide sufficient information for image coding and face recognition.

B. Kernel PCA

Schölkopf et al. [4] extended principal component analysis to a nonlinear form based on kernel methods. They compute PCA in another dot product feature space **F,** which is related to the input space $\mathbf{R}^{MN}$ by a possibly nonlinear map

$$\varphi{:}\mathbf{R}^{MN}\rightarrow\mathbf{F} \tag{8}$$

The $\{\boldsymbol{\varphi}(x_i)\}$, $i = 1, 2, \dots, n$, denote the images of center input vectors x_i included in the feature space [1].
It is assumed that preprocessing has been done to satisfy the zero mean condition of all feature vectors over the training sample, i.e. $1/n \sum_{i=1}^{n} \varphi(x_i)$.

Kernel function is the inner product term and denote as the scalar

$$k(x,x_i) = \varphi^T(x_i)\,\varphi(x), \quad i = 1, 2, ..., n \tag{9}$$

Compute the $k(x_i, x_j)$ as the ij-th element of the n-by-n matrix K,

$$K = \{k(x_i, x_j)\} = \{\varphi^T(x_i)\,\varphi(x_j)\}_{i,j = 1, 2, ..., n} \tag{10}$$

The covariance matrix in **F** is

$$R = \frac{1}{n}\sum_{i=1}^{n} \varphi(x_i)\varphi^T(x_i) \quad (11)$$

One now solves the *eigenvalue problem*

$$\mathbf{RV} = \lambda V \quad (12)$$

where λ is an eigenvalue of R and V is the associate eigenvector. There are coefficients α_i, $i = 1, \ldots, n$, such that $V = \sum_{i=1}^{n} \alpha_i \varphi(x_i)$. Premultiply both side of eq. (12) by $\varphi(x_k)$ and substituting eq. (11) and V in eq. (12), we obtain

$$\frac{1}{n}\sum_{i=1}^{n} \alpha_i(\varphi(x_k) \,.\, \sum_{j=1}^{n} \varphi(x_j))(\varphi(x_j).\,\varphi(x_i)) = \lambda \sum_{i=1}^{n} \alpha_i\,(\varphi(x_k).\,\varphi(x_i)) \quad (13)$$

By substituting eq. (10) in eq. (13), one obtains

$$K^2\,\alpha = n\lambda K\alpha \quad (14)$$

Then

$$K\alpha = n\lambda\alpha \quad (15)$$

where λ is an eigenvalue of K and α is the associate eigenvector.

For the purpose of principal component extraction one needs to compute projections onto the eigenvectors $V=[v_1, v_2, \ldots, v_n]$ in F. Let x be a test point, with an image $\varphi(x)$ in F, then

$$y_q = v_q^T \varphi(x) = \sum_{j=1}^{n} \alpha_{q,j}\varphi^T(x_j)\varphi(x) \quad (16)$$

In brief, the following steps were necessary to compute the principal components: first, compute the dot product matrix K defined by eq (10); second, compute its eigenvectors (eq. (15)) and normalize them in F; third, compute projections of a test point onto the eigenvectors by eq. (16).

In one's work, the polynomial kernel and the Gaussian kernel functions are used and the results are compared. The polynomial kernel is $k(x, y) = (x^T y + 1)^p$, where power p is specified a priori by the user. The Gaussian kernel, also called Radial basis kernel is $k(x, y) = \exp(-\frac{\|x-y\|^2}{2\sigma^2})$, where the width σ^2,common to all the kernels, is specified a priori by the user.

III. NEAREST NEIGHBOR CLASSIFIER

The k-nearest neighbor classifier (*k*-NN) is a method for classifying objects by finding the closest k neighbors in the feature space. The *k*-NN rule classifies the query point *z* by assigning it the class label *C* most frequently represented among the k nearest prototypes [26]; i.e., by finding the *k* neighbors with the minimum distances between *z* and all prototype feature points $\{z_{ci}, 1 \le c \le C, 1 \le i \le n_c\}$, as follows:

$$\{d_1(z, z_{c_1 i_1}), \ldots, d_k(z, z_{c_k i_k}\} = \min_{1\le c\le C, 1\le i\le n_c} d(z, z_{ci}) \quad (17)$$

where n_c is the number of samples per class. The number of distance computation is $\sum_{c=1}^{C} n_c$. Different parameters are used with k-NN, such as value of *k* nearest neighbors and distance model. In one's work, the Euclidean norm distance $d(z, z_{ci}) = \|z - z_{ci}\|$ is used and k=1 to find the class of the closest query point.

IV. ORL DATABASE OF FACES

This paper considers the well known ORL face database that is taken at the Olivetti Research Laboratory in Cambridge, Uk [27]. The ORL database contains 400 Grey images corresponding to ten different images of 40 distinct subjects. There are ten different images of each of 40 distinct subjects. Some sample faces are shown in Figure 1.

The images are taken at different times with different specifications, including varying slightly in illumination,

Figure 1. Sample faces of ORL database.

different facial expressions (open/closed eyes, smiling/non-smiling), and facial details (glasses/no-glasses). All images were taken against a dark homogeneous background with the subjects in an upright, frontal position, as well as tolerance for some tilting and rotation of up to 20 degrees. There is some variation in scale of up to about 10%. All the images are 8-bit grayscale with size 112 × 92 pixels. In one's work, the face image is resized to a resolution of 68 × 56 pixels.

V. EXPERIMENTAL RESULTS

The face recognition system consists of a feature extraction step and a classification step. The experiments are constructed a feature spaces by using the PCA or the kernel-PCA feature extraction. The 1-nearest neighbor classifier with Euclidean distance is used to find out the true class of the test patterns. Two kernel functions are used with the kernel-PCA, polynomial and Gaussian functions, to check which one achieved a better performance. The performance of the Gaussian kernel function is affected by its width σ. Besides, the performance of the polynomial kernel function is affected by its degree p. The performance of the nearest neighbor classifier varies as the number of principal components (features) changes and as the number of image samples per person changes.

The face recognition experiments are performed on the ORL face database. Through the experiments, the images of 40 different individuals have been checked; each person has 10 different images, giving a total of 400 images. For the first three experiments, the face space is constructed from one image sample per person. The rest of the 400 images are used for testing, i.e. 360 images (9 image samples per person). The final experiment was performed to check the impact of varying the number of image samples per person from 1 image until 9 images.

The first experiment was performed to determine the impact of varying the value of Gaussian kernel width and to determine the number of principal components that projected from the kernel-PCA, aiming at the lowest error recognition rate. Figure 2 shows the correct recognition rate as a function of Gaussian kernel function width and the number of features. The test correct rate can increase significantly by increasing the Gaussian function width until threshold value is followed by approximately fixing the performance of the classifier. Therefore, Gaussian kernel-PCA is used with a width equal to 3000 in the final experiment. One can see from figure 2 that used fewer numbers between 10 and 20 features results in a similar correct recognition rate, and 30 features results a litter better performance. Therefore, 30 features with the Gaussian kernel-PCA are used in the final experiment.

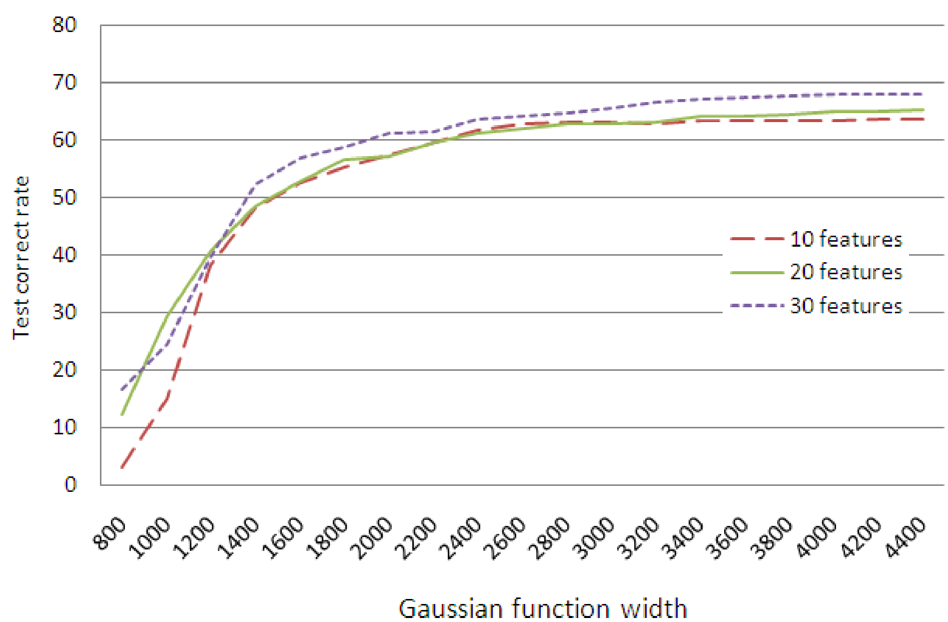

Figure 2. Performance of Kernel-PCA with Gaussian function.

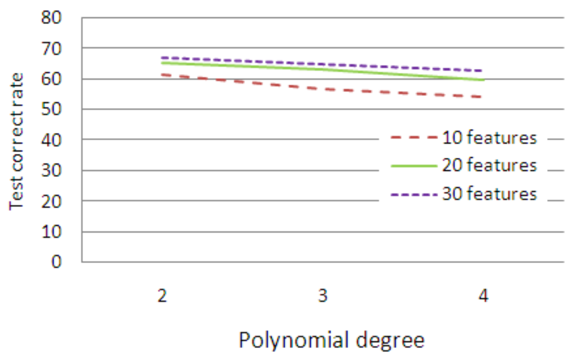

Figure 3. Performance of Kernel-PCA with polynomial function.

The second experiment was performed to determine the impact of varying the degree of polynomial kernel and to determine the number of principal components that projected from the kernel-PCA. Figure 3 shows the correct recognition rate as a function of the polynomial kernel degree and the number of features. One can observe a decrease in the test correct rate when increasing the polynomial degree. Therefore, the polynomial kernel of degree 2 is used in the final experiment. Furthermore, one observes that used fewer numbers between 20 and 30 features results in a similar correct recognition rate. Therefore, 30 features with the polynomial kernel-PCA of degree 2 are used in the final experiment.

The third experiment was performed to determine the impact of varying the number of principal components that projected from the PCA. Table 1 shows the experimental results. The performance of the PCA varies as the number of principal components (features) changes until threshold value is followed by approximately fixing the performance of the classifier. Therefore, 30 features with the PCA are used in the final experiment.

The next experiment compares the performance of the PCA and the two models of Kernel-PCA when the number of training image samples per person changes for constructing the face space. The size of the training set varied from k = 1 to 9 images per person and the remaining (400 - 40k) images form the test set. There was no overlap between the training and test sets. The feature dimensionality is 30 features. One can observe from figure 4 that the test correct rate can increase significantly by increasing the number of training image samples per person until threshold value is followed by approximately fixing the performance of the classifier. Furthermore, one observes that PCA outperforms two models of Kernel-PCA when using few numbers of images per person. The PCA with 3 images per person achieves 81.43% (228/280) correct recognition rate while only 30 features are used. The test correct rate can increase significantly by increasing the number of training images per person until threshold number (7 training images per person) is followed by approximately fixing the performance of the classifier. Overall, one sees that the PCA and the Gaussian Kernel-PCA with 7 training images per person achieves the highest recognition rate 93.33% (112/120) while only 30 features are used.

VI. Conclusions

The face recognition system consists of two important steps, the feature extraction and the classification. In this paper, the researchers focus on the feature extraction step. This paper investigates the nonlinear kernel function to improvement the principal component analysis (PCA) for feature extraction. The experiments carried out to investigate the performance of Kernel-PCA by comparing it with the performance of the PCA. Two kernel functions are used with the kernel-PCA, polynomial and Gaussian functions, to check which one achieved a better performance. The k-nearest neighbor classifier with Euclidean distance is used to investigate the performance of the Kernel-PCA and PCA for classification step.

The experiment on ORL face database showed that PCA outperforms polynomial and Gaussian models of Kernel-PCA when using few numbers of images per

Table I. Performance of PCA when varying the number of principal components.

	Number of features				
	10	20	30	40	50
Test correct rate (correct number)	63.88 (230/360)	66.38 (239/360)	68.88 (248/360)	70.83 (255/360)	70.83 (255/360)

Figure 4. Performance results with varying number of training images per person (using 30 features).

person. Besides, there is a little difference in performance between the PCA and the Gaussian Kernel-PCA when using more than 5 training images per person.

REFERENCES

[1] L. QingShan, H. Rui, L. HanQing, and M. SongDe. Kernel-based nonlinear discriminant analysis for face recognition. *Comput. Sci. & Technol. Journal*, 18(6):788-795, Nov. 2003.

[2] W. Zhao, R. Chellappa, A. Rosenfeld, and P.J. Phillips. Face recognition: a literature survey. *ACM Computing Surveys*, 399-458, 2003.

[3] D. R. Khattab, Y. M. A. El-Latif, M. S. A. Wahab, and M. F. Tolba. Efficient face-based non-split connectivity compression for quad and triangle-quad meshes. *3rd International Conference on Computer Graphics Theory and Applications (GRAPP)*, 31–38, 2008.

[4] D. R. Khattab, Y. M. A. El-Latif, M. S. A.Wahab, and M. F. Tolba. An enhanced edgebreaker compression algorithm for the connectivity of triangular meshes. *2nd International Conference on Computer Graphics Theory and Applications (GRAPP)*, 109–115, 2007.

[5] Fahmy Tolba M., Al-Said Ghonemy M., Taha I. A.-H., and Khalifa A.S. High capacity image steganography using wavelet-based fusion. International Symposium on Computers and Communications, 430-435, 2004.

[6] R. Brunelli and T. Poggio. Caricatural effects in automated face perception. *Biological Cybernetics Journal*, 69:235-241, 1993.

[7] M.E. Fathy, A.S. Hussein, and M.F.Tolba. Fundamental matrix estimation: a study of error criteria. *Pattern Recognition Letters*, 32(2):383–391, 2011.

[8] A. S. Ali., A. S. Hussien, M. F. Tolba, and A. H. Youssef. Visualization of large time-varying vector data. *3rd IEEE International Conference on Computer Science and Information Technology*, 5565176:210-215, 2010.

[9] A. S. Ali, A. S. Hussein, M. F. Tolba, and A. H. Yousef. Large-scale vector data visualization on high performance computing. *Journal of Software*, 6(2):298-305, 2011.

[10] M. Turk and A. Pentland. Eigenfaces for face recognition. *Cognitive Neuroscience Journal*, 3(1): 71-86, March 1991.

[11] R. Brunelli, and T. Poggio. Face recognition: features versue templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(10):1042-1052, 1993.

[12] K. J. Karande, S. N. Talbar. Independent component analysis of edge information for face recognition. *Image Processing Journal*, 3(3):120–130, 2009.

[13] H. Zhang, W. Deng, J. Guo, and J. Yang. Locality preserving and global discriminant projection with prior information. *Machine Vision and Applications Journal*, 21: 577–585, 2010.

[14] J. Li, B. Zhao, and H. Zhang. Face recognition based on PCA and LDA combination feature extraction. *1st IEEE International Conference on Information Science and Engineering*, 1240–1243, 2009

[15] K. Delac, M. Grgic, and S. Grgic. Independent comparative study of PCA, ICA, and LDA on the FERET data set, *Imaging Systems And Technology Journal*, 15: 252–260, 2005.

[16] B. Schölkopf, A. Smola, and K. R. Müller. Nonlinear component analysis as a Kernel eigenvalue problem. *Neural computing Journal*, 10(5):1299-1319, 1998.

[17] J. Lu, K. N. Plataniotis, and A. N. Venetsanopoulos. Face recognition using kernel direct discriminant analysis algorithms. *IEEE Trans Neural Networks*, 14(1):117–126, 2003.

[18] F. R. Bach, and M. I. Jordan. Kernel independent component analysis, *Machine Learning Research Journal*, 3:41–48, 2002.

[19] D. Zhou and Z. Tang. Kernel-based improved discriminant analysis and its application to face recognition. *Soft Comput Journal*,14:103–111, 2010.

[20] A. S. Khalifa, R. A. Ammar, M. F. Tolba, and T. Fergany. Dynamic online allocation of independent task onto heterogeneous computing systems to maximize load balancing. *8th IEEE International Symposium on Signal Processing and Information Technology*, 4775659: 418-425, 2008.

[21] M. F. Tolba, M. S. Abdel-Wahab, I. A. Taha, and A. A. Anbar. Directed acyclic graphs scheduling in grid computing environments. *International Conference on Internet Computing*, 260-266, 2004.

[22] P. Y. Han, A. T. B. Jin, and T. K. Ann. Kernel discriminant embedding in face recognition. *Journal of Visual Communication and Image Representation*, 22 (7): 634-642, 2011.

[23] S. Haykin. Neural networks and learning machines. 3rd edition, *Person*, 2009.

[24] T. Pentland, B. Moghadam, and T. Starner. View based and modular eigenspaces for face recognition. *IEEE Conference on Computer Vision and Pattern Recognition*, 84–91, 1994.

[25] G. Jarillo, W. Pedrycz, and M. Reformat. Aggregation of classifiers based on image transformations in biometric face recognition. *Machine Vision and Applications Journal*. 19:125–140, 2008.

[26] M. Orozco-Alzate, and C. G. Castellanos-domínguez. comparison of the nearest feature classifiers for face recognition. *Machine Vision and Applications Journal*, 17:279–285, July 2006.

[27] AT&T Laboratories Cambridge Database of faces. Available at: http://www.uk.research.att.com:pub/data/att_faces.zip