Evaluating the Effect of the Eigenvalues on BDF Classifier in Face Detection

Mohammad Ali Tinati, Ehsan Namjoo, and Mohammad Bagher Akbari Haghighat
Faculty of Electrical and Computer Engineering, University of Tabriz, Tabriz, Iran
tinati@tabrizu.ac.ir, e.namjoo@ieee.org, haghighat@ieee.org

Abstract—**Principal component analysis (PCA) is an effective tool for dimension reduction in classification approaches. Bayesian discriminating features (BDF) is a classifier which effectively utilizes this tool. In this classifier, any of the *M* largest eigenvalues of the training patterns' covariance matrix are individually involved in classification while the arithmetic average of the remaining eigenvalues take part just as a single parameter. In this paper, by suggesting a new classifier, effect of the number of involved eigenvalues in classification performance is studied. In the suggested classifier we ignore the arithmetic average that is utilized in BDF. Our experiments verify that increasing *M* does not lead to an ongoing increase in classifier's detection rate in both BDF and the proposed one. However, by over-increasing *M,* the dependency of classifiers' parameters to the training samples increases which could reduce the performance of the classifiers when they come to make decision about new samples. Furthermore, experimental results verify that arithmetic average of the remaining eigenvalues in BDF improves the classifier performance only when an appropriate number of eigenvalues is selected; hence, ignoring the arithmetic average, as done in proposed classifier, could provide a better performance rather than BDF.**

Keywords—Bayes decision theory; BDF classifier; feature extraction; Hotelling transform.

I. INTRODUCTION

Feature extraction is the most important stage in every pattern recognition approach. Proper selection of the feature vector significantly affects the classification performance and reduces the algorithm complexity. Often, there is no definite and usual way for feature extraction; however, regarding the purpose of the classification, some methods like Fourier transform, discrete cosine transform (DCT), and wavelet transform are implemented for feature extraction.

There are two major points in feature vector selection: firstly, the selected feature vector must contain the most beneficial information about the pattern, and secondly, the feature vector length should be as short as possible in order to reduce the algorithm complexity. Image feature extraction methods are far different from other feature extraction approaches because of the two-dimensional structure of image patterns, variety of them, and also the complicated correlation that exists among neighboring pixels in an image pattern.

One of the most complicated image patterns is the face [1]. Principal component analysis (PCA) is a method for dimension reduction of feature vectors of complicated patterns like face [2]. Appearance-based approaches like neural networks, support vector machines [3], and Hidden Markov Model [4] are also valuable methods for classifying complicated patterns. Bayesian discriminating features (BDF), as another appearance-based method, is also utilized to classify complicated patterns [5,6]. This method is based on PCA. The most important property of PCA, also called Hotelling transform (HT), is its optimal reconstruction property. That is, HT reduces the feature vector length in a way that the new vector contains the maximum amount of information from the previous one [7,8].

The classifiers based on probability theory are more flexible among other appearance-based classifiers like neural networks. These classifiers are usually based on Bayes decision theory. If a proper probability density function is assigned to the training samples, Bayes decision theory will provide a minimum error rate classifier [8].

Long feature vectors lead to large amount of calculations and reduce the decision speed. Designing an accurate and fast classifier is necessary in real-time applications, so the size of long feature vectors should be reduced. Utilizing Hotelling transform helps BDF to reduce the feature vector size and makes it a fast classifier. On the other hand, BDF classifier is based on Bayes decision theory, so it yields the minimum error rate if the proper probability density function is fitted to the train data set [5]. In BDF classifier, the M largest eigenvalues of the covariance matrix, which contain large amount of information about the training patterns, are individually involved in classification while the other eigenvalues take part in classification by their arithmetic average just as a single parameter.

In this paper, the effect of the number of selected eigenvalues in PCA and consequently the arithmetic average of the remaining eigenvalues on classification is studied. For this aim, we propose another classifier based on Bayes decision theory and PCA. In contrast with BDF, in this new classifier, only the M largest eigenvalues of the covariance matrix are involved in classification and other values are disregarded. Our experimental results verify that using arithmetic average of the remaining eigenvalues in BDF does not always lead to better performance of the classifier. Moreover, selecting a large number of eigenvalues does not improve the classifier detection rate either. The experiments are applied on the face as a complicated and different pattern.

This paper is organized as follows: in Section II, details of the feature extraction used in this paper are discussed. Section

978-1-61284-832-7/11/$26.00 ©2011 IEEE

III describes the BDF and our proposed classifier based on Bayes decision theory. In section IV, the simulation results on the effect of the number of selected eigenvalues and using arithmetic average of remaining eigenvalues are presented; and finally, Section V concludes the paper.

II. Feature Extraction

Feature extractors take proper information from rare patterns. The vectors provided after feature extraction step, are fed to the classifier to be used in making decisions about a specific input pattern. In this experiment, patterns are considered as rectangular 16×16 face sub-images. Suppose that $I(i,j)\epsilon R^{m\times n}$ presents an $M\times N$ sub-image matrix and $X\epsilon R^{mn}$ is a vector that is formed by concatenating the rows or columns of *I(i,j)*. As defined below, the one dimensional Haar representation of *I(i,j)* yields two images, $I_h(i,j)$ *and* $I_v(i,j)$, corresponding to the horizontal and vertical difference images, respectively.

$$I_h(i,j) = I(i+1,j) - I(i,j) \quad 1 \le i \le m-1, 1 \le j \le n \tag{1}$$

$$I_v(i,j) = I(i,j+1) - I(i,j) \quad 1 \le i \le m, 1 \le j \le n-1 \tag{2}$$

As defined, similar vectors $X^h\epsilon R^{(m-1)n}$ and $X^v\epsilon R^{m(n-1)}$ are constructed by concatenating the rows or columns of $I_h(i,j)$ *and* $I_v(i,j)$, respectively. Fig. 1.a and b demonstrate some face sub-images from BioID database and their corresponding sub-sampled 16×16 patterns. Figs 1.c and d also illustrate 1-D horizontal and vertical Haar difference images calculated from (1) and (2). The horizontal (row) and vertical (column) projection vectors of *I(i,j)* are calculated as:

$$X_r(i) = \sum_{j=1}^{n} I(i,j), \quad 1 \le i \le m \tag{3}$$

$$X_c(j) = \sum_{i=1}^{m} I(i,j), \quad 1 \le j \le n \tag{4}$$

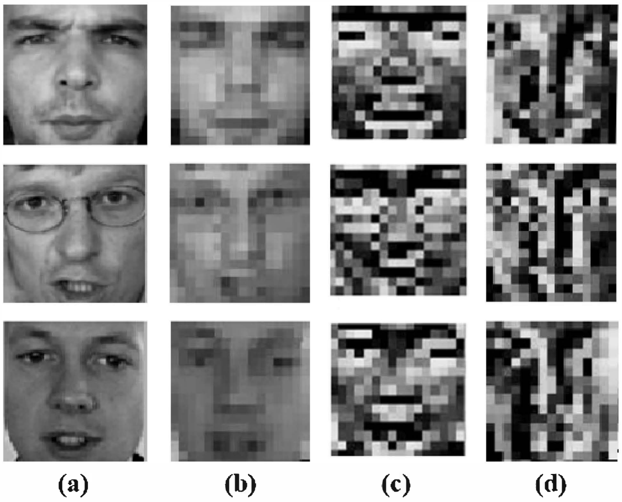

Fig 1. (a) Face sub-images from BioID database. (b) Resized pattern. (c),(d)1-D horizontal and vertical difference images.

In order to normalize these vectors, each vector is subtracted from the mean of its components and then divided by the standard deviation of them. Let $\hat{X}, \hat{X}_h, \hat{X}_v, \hat{X}_r, \hat{X}_c$ be the normalized vectors. A new feature vector is constructed by concatenating these normalized vectors as in (5).

$$\tilde{Y} = \left(\hat{X}, \hat{X}_h, \hat{X}_v, \hat{X}_r, \hat{X}_c\right)^T \tag{5}$$

where $(\ )^T$ represents the transpose operation. Finally the normalized vector Y is defined as a discriminating features vector:

$$Y = \frac{\tilde{Y} - \mu}{\sigma} \tag{6}$$

where μ and σ are the mean and the standard deviation of $\tilde{Y}$, respectively. The achieved normalized vector will be utilized as the feature vector [5].

III. Classifiers

Gaussian probability density function is a powerful choice in modeling many natural events [9]. The conditional probability density function of the face feature vector can be modelled by this distribution as:

$$p(Y \mid \omega_f) = \frac{1}{(2\pi)^{\frac{N}{2}} \left|\Sigma_f\right|^{\frac{1}{2}}} \times \exp\left\{ -\frac{1}{2}(Y - M_f)^T \Sigma_f^{-1} (Y - M_f) \right\} \tag{7}$$

where M_f, Σ_f and N are the mean vector, covariance matrix, and the length of the feature vector of the face class ω_f, respectively. Taking the natural logarithm on both sides of the equation (7), we will have:

$$\ln\left[p(Y \mid \omega_f)\right] = -\frac{1}{2}\left\{(Y - M_f)^T \Sigma_f^{-1} (Y - M_f) + N\ln(2\pi) + \ln(\mid \Sigma_f \mid)\right\} \tag{8}$$

The covariance matrix, Σ_f, can be factorized as:

$$\Sigma_f = \varphi_f \Lambda_f \varphi_f^T \tag{9}$$

where:

$$\begin{cases} \varphi_f^T \varphi_f = \varphi_f \varphi_f^T = I_N \\ \Lambda_f = diag\{\lambda_1, \lambda_2, ..., \lambda_N\} \end{cases}$$

where φ_f is an $N\times N$ eigenvector matrix of the covariance matrix of face feature vectors. Λ_f is a diagonal matrix with the same size containing the corresponding eigenvalues (λ_i) of Σ_f in decreasing order, and I_N is an $N\times N$ identity matrix.

The Hotelling transform is defined in (10):

$$Z = \varphi^T (Y - m_y) \tag{10}$$

where m_y is the mean vector of Y feature vectors. Now, using the Hotelling transform and regarding the fact that φ is an orthogonal matrix, ($\varphi^T = \varphi^{-1}$), we will have:

$$(Y-M_f)^T\Sigma_f^{-1}(Y-M_f) = (Y-M_f)^T(\varphi_f\Lambda_f\varphi_f^T)^{-1}(Y-M_f)$$

$$= (Y-M_f)^T(\varphi_f^T)^{-1}\Lambda_f^{-1}\varphi_f^{-1}(Y-M_f) = Z^T\Lambda_f^{-1}Z$$

So, the natural logarithm of the probability density function of the face samples can be rewritten as:

$$\ln\left[p(Y\mid\omega_f)\right] = -\frac{1}{2}\left\{Z^T\Lambda_f^{-1}Z + N\ln(2\pi) + \ln\left|\Lambda_f\right|\right\} \tag{11}$$

Z components are the principal components. Relying on the optimal reconstruction property of PCA, just M major eigenvalues are enough to estimate the probability density function for face class, where M is much smaller than N. In BDF, the remaining N-M values are estimated by their arithmetic average as:

$$\rho = \frac{1}{N-M}\sum_{k=M+1}^{N}\lambda_k \tag{12}$$

where λ_i's are the eigenvalues of the covariance matrix for face class. Finally, the natural logarithm of the probability density function for the face class will be rewritten as below:

$$\ln\left[p(Y\mid\omega_f)\right] = -\frac{1}{2}\{\sum_{i=1}^{M}\frac{z_i^2}{\lambda_i} + \frac{\left\|Y-M_f\right\|^2 - \sum_{i=1}^{M}z_i^2}{\rho}$$
$$+\ln\left(\prod_{i=1}^{M}\lambda_i\right) + (N-M)\ln\rho + N\ln(2\pi)\ \} \tag{13}$$

Similarly, the natural logarithm of the probability density function for the non-face class is calculated as:

$$\ln\left[p(Y\mid\omega_f)\right] = -\frac{1}{2}\{\sum_{i=1}^{M}\frac{u_i^2}{\lambda_i^{(n)}} + \frac{\left\|Y-M_n\right\|^2 - \sum_{i=1}^{M}u_i^2}{\varepsilon}$$
$$+\ln\left(\prod_{i=1}^{M}\lambda_i^{(n)}\right) + (N-M)\ln\varepsilon + N\ln(2\pi)\ \} \tag{14}$$

where u_i's are the principal components, $\lambda_i^{(n)}$'s are the eigenvalues of the covariance matrix, and M_n is the mean feature vector for non-face class. ε is the estimation of the remaining N-M values, and ω_n represents the non-face class.

The Hotelling transform for the non-face class is:

$$U = \varphi_n^T(Y-M_n) \tag{15}$$

where φ_n is an eigenvector matrix that their columns are the eigenvectors of the covariance matrix of the non-face class.

Y represents a face pattern if $P(\omega_f\mid Y) > P(\omega_n\mid Y)$. In this case, Bayes decision theory is defined as:

$$P(\omega_f\mid Y) = \frac{P(\omega_f)p(Y\mid\omega_f)}{p(Y)}$$
$$P(\omega_n\mid Y) = \frac{P(\omega_n)p(Y\mid\omega_n)}{p(Y)} \tag{16}$$

where $P(\omega_f)$ and $P(\omega_n)$ are the priori probabilities of the face and the non-faces classes, respectively, and $p(Y)$ is the mixture probability density function of Y. Finally, BDF classifier can be written as in (17):

$$Y \in \begin{cases}\omega_f & if \quad \delta_f + \tau < \delta_n,\ \delta_f < \theta \\ \omega_n & otherwise\end{cases} \tag{17}$$

where τ and θ are two control parameters which are determined empirically based on training set, and:

$$\delta_f =$$
$$\sum_{i=1}^{M}\frac{z_i^2}{\lambda_i} + \frac{\left\|Y-M_f\right\|^2 - \sum_{i=1}^{M}z_i^2}{\rho} + \ln(\prod_{i=1}^{M}\lambda_i) + (N-M)\ln\rho \tag{18}$$

$$\delta_n =$$
$$\sum_{i=1}^{M}\frac{u_l^2}{\lambda_i^{(n)}} + \frac{\left\|Y-M_n\right\|^2 - \sum_{i=1}^{M}u_\imath^2}{\varepsilon} + \ln(\prod_{i=1}^{M}\lambda_i^{(n)}) + (N-M)\ln\varepsilon \tag{19}$$

Here, another Bayesian classifier similar to BDF is proposed. In contrast with BDF, in the new classifier, the M largest eigenvalues of the covariance matrix are involved in classification and other values are disregarded. That is, N-M remaining values are not taken into account in classification. The decision rule for this new classifier is given as:

$$Y \in \begin{cases}\omega_f & if \quad \eta_f + \tau < \eta_n,\ \eta_f < \theta \\ \omega_n & otherwise\end{cases} \tag{20}$$

where:

$$\eta_f = \frac{1}{2}\left\{Z_M^T\Lambda_M Z_M + N\ln(2\pi) + \ln(\prod_{i=1}^{M}\lambda_i)\right\} \tag{21}$$

$$\eta_n = \frac{1}{2}\left\{U_{M,}^T\Lambda_M^{(n)}U_M + N\ln(2\pi) + \ln(\prod_{i=1}^{M}\lambda_i^{(n)})\right\} \tag{22}$$

and τ and θ are the control parameters determined empirically from training set. These two parameters are selected in a way that maximizes the detection rate in training set [10]. As follows, we will call this classifier as Bayesian classifier (BC).

IV. Experimental Results and Analysis

In this section, performance of each classifier is evaluated by their detection rate on face patterns. Face images used in our experiments are from BioID database [11] containing 1500 images, and Tabriz University database [10] which consists of 2500 face images. In order to have a class of non-face images, 2500 other nature images are selected. Training procedure of the BDF classifier is as follows.

Firstly, the covariance matrix and the mean vector of the training samples are calculated. Then, eigenvalues and eigenvectors of the covariance matrix are calculated. Afterwards, the appropriate number of eigenvalues (M) is selected; and finally, classifier parameters are determined as discussed in Section III. Calculating the covariance matrix and selecting the proper number of eigenvalues are common training procedures in both classifiers. Assuming that the eigenvalues of the covariance matrix are sorted decreasingly, the mean square error for the reconstructed vectors will be computed by equation (23) [12,13].

$$e_{ms} = \sum_{i=1}^{N}\lambda_i - \sum_{i=1}^{k}\lambda_i = \sum_{i=k+1}^{N}\lambda_i \tag{23}$$

Using equation (23) and according to the allowed range of error, we can determine the appropriate number of eigenvalues. For our experiments, we have randomly chosen 1300 training samples, and the number of test samples is 200. Both test and train samples have been selected from BioID database. From non-face set, we have chosen 2000 training samples and 500 test samples.

γ is defined as *reconstruction ratio* in (24) [14]. Fig. 2 illustrates γ as the ratio of the sum of the *M* selected eigenvalues to the sum of the all eigenvalues. So, 1-γ will be defined as *reconstruction error*. TABLE I reveals the reconstruction error values (1-γ) for different number of selected eigenvalues. As it can be seen in TABLE I, selecting more eigenvalues decreases the reconstruction error. This error is about 0.1 when 50 eigenvalues are selected. Considering that the length of the feature vector *Y* and so the number of eigenvalues are 768, selecting 50 eigenvalues means only 6% of them. As it is shown in Fig. 2, this number of eigenvalues contains about 90% of the training patterns information.

$$\gamma = \sum_{i=1}^{M}\lambda_i \Big/ \sum_{i=1}^{N}\lambda_i \tag{24}$$

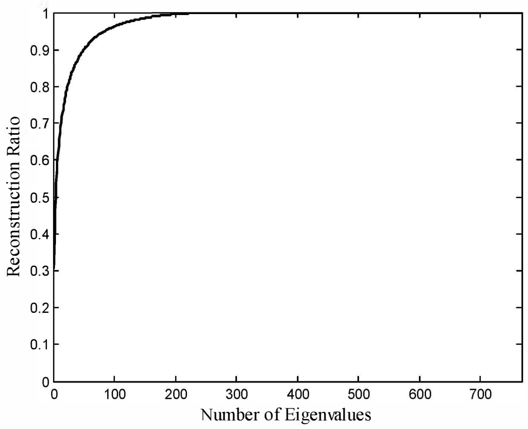

Fig 2. Reconstruction ratio (γ) in terms of number of selected eigenvalues (*M*).

TABLE I

RECONSTRUCTION ERROR FOR DIFFERENT NUMBERS OF SELECTED EIGENVALUES

Number of Eigenvalues	30	40	50	60	70	80
Reconstruction Error (1-γ)	0.17	0.13	0.10	0.09	0.08	0.06

Selecting 50 major eigenvalues, TABLE II and TABLE III demonstrate the detection rate of the examined classifiers. As it can be seen, BDF attains a higher detection rate than the proposed Bayesian classifier. In TABLE III, with an increase in the number of training patterns, performance of both classifiers is improved. However, this improvement is more significant for the Bayesian classifier. This means that Bayesian classifier is an effective classifier when the training data is adequate.

TABLE II

PERFORMANCE EVALUATION OF BDF AND BAYESIAN CLASSIFIERS ON BIOID DATABASE

Classifier	BDF	Bayesian
Number of Training Set	1300	1300
Number of Testing Set	200	200
Number of Detection	190	104
Detection Rate (%)	95.0	52.0

TABLE III

PERFORMANCE EVALUATION OF BDF AND BAYESIAN CLASSIFIERS ON TABRIZ UNIVERSITY DATABASE

Classifier	BDF	Bayesian
Number of Training Set	2000	2000
Number of Testing Set	500	500
Number of Detection	483	459
Detection Rate (%)	96.60	91.80

In another experiment, the effect of the number of selected eigenvalues (*M*) on the performance of classifiers is studied. This experiment is performed on BioID face database with 1300 training and 200 test images. As it can be seen in TABLE IV, unexpectedly, selecting more eigenvalues does not always guarantee the better performance of the classifier. On the other hand, over-increasing *M* may result in more dependency of classifiers on training vectors which will reduce the performance of classifiers in confrontation with new samples. In the table, for *M*=100 proposed Bayesian classifier outperforms the BDF. The different performance of the two classifiers is because of the different strategies performed on the remaining eigenvalues.

TABLE IV

THE COMPLETE RECONSTRUCTION ERROR FOR DIFFERENT NUMBER OF SELECTED EIGENVALUES

Number of Eigenvalues	Detection Rate (%) in BDF classifier	Detection Rate (%) in Bayesian classifier
50	95	52
70	97.23	93.17
100	75.42	83.84
120	42.34	55.23
150	23.01	42.18

In BDF classifier, *M* major eigenvalues of the covariance matrix are involved in classification, and other remaining eigenvalues are involved by their arithmetic average. On the other hand, in proposed Bayesian classifier only the *M* major eigenvalues are involved, and the other remaining values are disregarded. Therefore, regarding to TABLE IV, we can conclude that arithmetic average of the remaining eigenvalues improves the classifier performance only when the number of selected eigenvalues (*M*) is not high. That is, over-increasing *M* does not increase the classifier detection rate. For instance, when *M*=70, detection rate is better than the case of *M*=50; however, if we increase *M* to more than 100, detection rates of the classifiers decrease in a way that the Bayesian classifier performs better than BDF. As a result, the arithmetic average of the *N-M* values employed in BDF leads to a performance failure in classification when the number of selected eigenvalues is high.

Although selecting a higher number of eigenvalues decreases the reconstruction error (see TABLE I), this error reduction does not result in a performance improvement in classifier. However, over-increasing *M* may result in more dependency of classifier on training vectors that leads to the performance failure of the classifier in confrontation with new samples. That is, if more eigenvalues are selected, the classifier loses its flexibility in dealing with new samples.

V. Conclusion

In this paper, the effect of the number of selected eigenvalues and the arithmetic average of the remaining values on classification are studied. As it was shown, selecting a higher number of eigenvalues of the covariance matrix does not always lead to better performance of the classifier. Comparing the proposed Bayesian classifier and BDF, it was concluded that the arithmetic average of the remaining eigenvalues help the classification only when the number of selected eigenvalues is not so large. So, a proper selection in the number of eigenvalues will lead to more efficacy of the arithmetic average of the remaining values in classification.

On the other hand, it was concluded that selecting a higher number of features does not guarantee better performance of classifiers. Because, it will lead to a high dependency of the classifier on training set which will result in a performance failure in dealing with new samples. In order to have a better performance in BDF classifier, and to prevent the dependency of the classifiers' parameters to the training samples, the reconstruction error should not be chosen around zero. Selecting 1-γ in the interval [0.08, 0.1] is suggested.

References

[1] M.H. Yang, D.J. Kriegman, and N. Ahuja, “Detection Faces in Images: A Survey,” *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 24, no. 1, pp. 34-58, 2002.

[2] J. Shlens, “A Tutorial on Principal Component Analysis,” Institute for Nonlinear Science, UCSD, 2005. Available from: http://www.cs.cmu.edu/~elaw/papers/pca.pdf.

[3] C. J. C. Burges, “A Tutorial on Support Vector Machines for Pattern Recognition,” *Data Mining and Knowledge Discovery*, vol. 2, no. 2, pp. 121-167, 1998.

[4] L. R. Rabiner, “A Tutorial on Hidden Markov Model and Selected Applications in Speech Recognition,” *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257-286, 1989.

[5] C. Liu, “A Bayesian Discriminating Features Method for Face Detection,” *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 25, no. 6, pp. 725-740, 2003.

[6] P. Shih and C. Liu, “Face Detection Using Discriminating Feature Analysis and Support Vector Machine,” *Pattern Recognition*, vol. 39, no. 2, pp. 260-276, 2006.

[7] S. Theodoridis and K. Koutroumbas, “*Pattern Recognition*,” Academic Press, 3rd Edition, 2006.

[8] R. O. Duda, P. E. Hart, and D. G. Stork, “*Pattern Classification*,” John Wiley and Sons, 2nd Edition, 2000.

[9] K. Kim and G. Shevlyakov, “Why Gaussianity?,” *IEEE Signal Processing Magazine,* vol. 25, no. 2, pp. 102-113, 2008.

[10] E. Namjoo, “Face Detection Using Skin Color Features”, M.Sc. Thesis, Faculty of Electrical and Computer Engineering, University of Tabriz, Tabriz, Iran, 2006.

[11] Available from: http://www.bioid.com/download-center/software/bioid-face-database.html.

[12] B. Moghaddam and A. Pentland, “Probabilistic Visual Learning for Object Representation, “*IEEE Transactions on Pattern Analysis and Machine Intelligence*,” vol. 19, no. 7, pp. 696-710, 1997.

[13] B. Moghaddam, “Principal Manifolds and Probabilistic Subspaces for Visual Recognition,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 6, pp. 780-788, 2002.

[14] X. Jiang, “Linear Subspace Learning-Based Dimensionality Reduction,” *IEEE Signal Processing Magazine*, vol. 28, no. 2, pp. 16-26, 2011.