Neurocomputing 73 (2010) 1550–1555

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Multilinear principal component analysis for face recognition with fewer features

Jin Wang [a], Armando Barreto [a], Lu Wang [a], Yu Chen [a], Naphtali Rishe [b], Jean Andrian [a], Malek Adjouadi [a,*]

[a] Department of Electrical and Computer Engineering, Florida International University, Miami, FL 33174, USA
[b] The School of Computing and Information Science, Florida International University, Miami, FL 33174, USA

ARTICLE INFO

Available online 11 March 2010

Keywords:
Multilinear principal component analysis
Nearest neighbor classifier
Support vector machine
Face recognition

ABSTRACT

In this study, a method is proposed based on multilinear principal component analysis (MPCA) for face recognition. This method utilized less features than traditional MPCA algorithm without downgrading the performance in recognition accuracy. The experiment results show that the proposed method is more suitable for large dataset, obtaining better computational efficiency. Moreover, when support vector machine is employed as the classification method, the superiority of the proposed algorithm reflects significantly.

1. Introduction

In recent years, remarkable efforts have been extended into the face recognition problem [1], especially with the considerable accessibility to new technologies and the wide range of commercial applications that have become available.

Common sense dictates that all automatic face recognition systems should include two key steps. The first step consists of face detection and feature extraction processes, which is necessary step to locate the face and extract thereafter face features in the image that are essential for further processing. The features extracted will then be fed into the second and critical step that of face recognition process. The face recognition process remains a challenging endeavor for researchers due to the myriad of faces that can be considered and the variability in the circumstances and ways under which images of these faces are taken. Therefore, face recognition is considered the focal point of this research. The Eigenface system for face recognition was initially developed by Turk and Pentland [2]. Later, other PCA-based face recognition methods were introduced with the use of independent component analysis [3] and kernel PCA [4] applied using the kernel Hilbert space. For these methods, there is a need to reshape a series of p MN input images into a matrix with a higher dimensional matrix of size MN p; this type of matrices may overburden the computational requirements.

To decrease the effect of computational cost due to the high dimensionality, Yang et al. [5] proposed the 2D PCA approach for face recognition which reduced the computational complexity significantly and improve the recognition accuracy. And Visani et al. [7] prove the better robustness of 2D PCA. Various other studies [9–14] followed this trend with the intent to improve the performance of 2D PCA. Nguyen et al. [15] proposed a random subspace 2D PCA which combines 2D PCA with the random subspace technique. In this method, not only the largest eigenvectors are considered, but also some eigenvectors with non-zero eigenvalues. This method shows better recognition rate than traditional 2D PCA. Sanguansat et al. [16] and Xu et al. [17] introduced similar methods as the one proposed in [15]. Pan and Ruan [18] described a new method to using Gabor features combining with the $(2D)^2$PCA for palmprint recognition. While Yu et al. [19] extended the concept of 2D PCA of both vertical and horizontal direction into the linear discriminant analysis (LDA) algorithm for face recognition application. Moreover, Yu and Bennamoun [6] provided the nD-PCA algorithm based on high-order singular value decomposition and test its validity and performance on the face recognition grand challenge (FRGC) 3D scan facial database. Lu et al. [20,21] provide a new framework of multilinear PCA for dimensionality reduction and feature extraction with an application to gait recognition. Tao et al. [22,23] provided the study of geometric mean for subspace selection and Pang et al. [24] introduced new methods for extracting Gabor features efficiently using kernalized region covariance matrix for high dimensional data. And a comparative study for these traditional methods has been performed in an earlier study [25].

This question posed at this juncture is whether those computationally efficient methods do actually provide the necessary features or the features provided are indeed sufficient for solution to the face recognition problem. In retrospect, what these methods (2D PCA and multilinear PCA) attempt to do is to introduce more features to enhance the face recognition accuracy and at the same time achieve computationally efficiency for feature extraction, in contrast to the traditional PCA. Furthermore, in the classification process, the more features extracted the more computational time will be needed for all the methods.

* Corresponding author.
E-mail address: adjouadi@fiu.edu (M. Adjouadi).

0925-2312/$ - see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.neucom.2009.08.022

Consequently, the proposed method based on the multilinear PCA is designed to reduce the features necessary for face recognition without compromising on the performance of the recognition process. Different feature selections (in terms of the different PCA projections) and classification methods (nearest neighbor classifier and support vector machines) are compared in the experiments.

2. 2D principal component analysis

2D PCA [5] changes the Eigenface algorithm by keeping the original images and doing the decomposition directly on the mean covariance matrix of the images to find the feature basis. Suppose the training data are composed of M face images A_i with size $m \times n$. The detailed algorithm of 2D PCA is described as follows:

(a) Compute the average image of all training images $\overline{A} = (1/M)\sum_{i=1}^{M} A_i$.
(b) Construct the covariance matrix $C = (1/M)\sum_{i=1}^{M} (A_i - \overline{A})(A_i - \overline{A})^T$.
(c) Perform Eigen decomposition on C and let $X = [x_1, x_2, \ldots, x_k]$ be the matrix consisting of the selected eigenvectors corresponding to the largest k eigenvectors.
(d) Project a sample into this subspace to find the features $Y_i = (A_i - \overline{A})^T X$.
(e) Perform the classification based on the features.

3. Multilinear principal component analysis

With the MPCA method, the training images are rearranged into a 3D tensor as $\mathcal{S} \in \mathcal{R}^{I_1 \times I_2 \times I_3}$, where I_1 is the height of the images, I_2 is the width of the images and I_3 presents the number of images used in the training phase. The multilinear PCA algorithm can be summarized as follows:

(a) Compute the mean matrix $\overline{A} = (1/I_3)\sum_{i=1}^{I_3} A_i$.
(b) Center the training tensor $\hat{S} = [A_1 - \overline{A}, A_2 - \overline{A}, \ldots, A_{I_3} - \overline{A}]$.
(c) Unfold tensor into a matrix. For different modes, the unfolding matrices are different. The elements in the mode-n unfolding matrix are defined by $(\hat{S}_{(n)})_{(index)} = a_{i_1 i_2 i_3}$, where $index = \left[i_n, \sum_{m=3}^{n+1} (i_m - 1)\left(\prod_{p=m}^{n+1} I_p\right)\left(\prod_{p=n-1}^{1} I_p\right) + \sum_{m=n-1}^{1} (i_m - 1)\left(\prod_{p=m-1}^{1} I_p\right)\right]$.
(d) For a given mode-n, find the eigenvectors through covariance matrix $C_{(n)} = \hat{T}_{(n)} \hat{T}_{(n)}^T$, and let $X_{(n)} = [x_1, x_2, \ldots, x_{k_{(n)}}]$ be the selected eigenvectors corresponding to the largest $k_{(n)}$ eigenvectors.
(e) Two types of feature selections processes can be performed: (a) using $Y_{i(n)} = (A_i - \overline{A}) \times_n X_{(n)}^T$ for different modes and (b) using $Y_i = X_1^T (A_i - \overline{A}) X_{(2)}$. It is noted that the features obtained from the two methods are different; in that, the first one provides $k_{(1)} \times I_2 + k_{(2)} \times I_1$ features while the second one provides $k_{(1)} \times k_{(2)}$ features.
(f) Perform the classification process on either one of these two sets of features.

It should be noted that the multilinear PCA for images is similar to other 2D PCA algorithm provided in literatures [8–14]. The difference is that when computing covariance matrix for multilinear PCA, the step of finding the mean covariance matrix is ignored. Theoretically the two algorithms provide the same results working on 2D images. However, multilinear PCA can be applied to high dimension tensors while 2DPCA only works for 2D images. Moreover, the superiority of MPCA is already stated in those literatures.

4. Random subspace method

The random subspace method was first introduced by Ho [26] is designed to randomly selects different features and constructs multiple smaller subspaces. This algorithm has been utilized in [15,16,27], using different implementation methods of the random subspace.

In [27], the random subspace method is applied for randomly choose eigenvectors from the whole eigenvector set. The algorithm random choose k from those $m-1$ eigenvectors for features. Recall that for the 2D PCA, there are $m-1$ eigenvectors from the decomposition of the covariance matrix.

In [15], the random subspace is applied to the features that were already found through projection to the first k largest eigenvectors. The projection to the k eigenvectors is computed as $Y_i = (A_i - \overline{A})^T X$, where $Y_i \in \mathcal{R}^{n \times k}$. Then the random subspace method random choose rows from matrix Y_i, where the subspaces are denoted as $Y_{iR} \in \mathcal{R}^{R \times k}$, where $R < n$.

In [16], the random subspace method is used to construct the X matrix in the 2D PCA algorithm. The first k largest eigenvectors are determined, while k_1 more eigenvectors from the rest non-zero eigenvectors are chosen by the random subspace method. The combination of $k+k_1$ eigenvectors is utilized for further classification.

Since the random subspace method was reported with better performance when combined with 2D PCA in the literature [15,16]. In this study, the performance of the random subspace method combined with the MPCA will be assessed, and the experimental results are provided.

5. Proposed algorithm

The proposed algorithm integrating MPCA to RSM in a 2-step method seek an optimized number of features (steps (c)–(e)) without compromising the recognition accuracy. For the multilinear PCA, both feature selection methods described earlier are taken into consideration. Furthermore, two ways of random subspace methods are implemented. The detail steps for the algorithm are:

(a) Perform the first four steps of the MPCA algorithm.
Branch 1I (RSM1)
Randomly choose $k_{(n)1}$ more eigenvectors with non-zero eigenvalues from the rest eigenvectors.
Generate new eigenvectors set $X_{(n)} = [x_1, x_2, \ldots, x_{k_{(n)}}, x_{k_{(n)}+1}, \ldots, x_{k_{(n)}+k_{(n)1}}]$.
End
(b) Find features of images
Branch 2I (MPCA1)
Find $Y_{i(n)} = (A_i - \overline{A}) \times_n X_{(n)}^T$ for mode-1 and mode-2.
Branch 1II (RSM2)
It is known that $Y_{i(1)} \in \mathcal{R}^{I_2 \times k_{(1)}}$ and $Y_{i(2)} \in \mathcal{R}^{I_1 \times k_{(2)}}$.
Random choose $k_{(1)1}$ and $k_{(2)1}$ rows from $Y_{i(1)}$ and $Y_{i(2)}$.
Generate new $Y_{i(1)} \in \mathcal{R}^{k_{(1)1} \times k_{(1)}}$ and $Y_{i(2)} \in \mathcal{R}^{k_{(2)1} \times k_{(2)}}$.
End
Rearrange those features into a vector y_i.
End
Branch 2II (MPCA2)

Compute $Y_i = X_1^T(A_i - \overline{A})X_{(2)}$.
Rearrange those features into a vector y_i.
End
PCA

(c) If $y_i \in \mathcal{R}^{k\times 1}$, generate new matrix G from the feature vectors for each image, where $G \in \mathcal{R}^{k\times I_3}$. Find the eigenvectors $Z=[z_1,z_2,\ldots,z_p]$ with the first p largest eigenvectors using SVD. For large dataset, SKL can be used to compute the eigenvectors.
(d) Generate the features for each image as $yy_i = Z^T(y_i - \overline{y})$, where $yy_i \in \mathcal{R}^{p\times 1}$.
(e) Perform the classification process based on features yy_i.

Note, for branch 1, either I or II in the branch can be utilized, or none of them is utilized. For branch 2, either I or II should be utilized.

For performance evaluation, different algorithms are implemented including PCA, 2D PCA, MPCA1, MPCA2, 2D PCA+RSM1, 2D PCA+RSM2, and the newly developed MPCA1+RSM1, MPCA1+RSM2, MPCA2+RSM1 (until step (b)). Based on the obtained results, the best two methods in terms of recognition accuracy will then be selected and run through steps (c) through (e) for optimized feature selection (minimum number of features and high recognition accuracy). Since MPCA2 has a comparatively compact feature base of the images, there is no necessary to use the RSM1 for finding subspaces.

6. Experiment results

All the experiments were evaluated using Matlab on Windows Vista based PC with Intel Core 2 1.60 GHz and 2 G RAM. Images from the AT&T database have a resolution of 112×92, and 40 subjects with 10 images each included in the database. Fig. 1 shows some sample images from the database. Those images have different characteristics, such as with or without glasses, and with different facial expressions.

Different experiments were used to test the different algorithms. The training images are chosen randomly. The first test uses nine images for training; second one uses seven images for training, third one uses five images for training, fourth one uses three images for training, and the remaining images are used in the phase testing.

The first experiment is run to compare 2D PCA, MPCA1 and MPCA2, and to find out which parameters yield high accuracy in the recognition process.

Results given in Table 1 indicate that MPCA1 has a superior performance in best recognition accuracy than the other two methods. Although all the algorithms except PCA yield a 100% accuracy in 9 to 1 test, for all the other tests, MPCA1 has the highest accuracy.

In the second experiment, only three tests are chosen with the parameters obtained in Table 1 with the best recognition accuracy, leaving out the PCA method. The algorithms that were tested include 2D PCA+RSM1, 2D PCA+RSM2, MPCA1+RSM1, MPCA1+RSM2 and MPCA2+RSM1 (these algorithms are combinations of different RSM algorithms in the proposed algorithm before step c under branch 1a and branch 1b as described earlier). Moreover, since MPCA2 already has very compact features, there is no need to test on the combination with RSM2. The results from five different combinations are shown in Table 2.

From the results in Table 2, MPCA1+RSM1 algorithm provides the best recognition accuracy among all other algorithms. When combining with the RSM algorithm, the parameters for best recognition accuracy are used which may restrict the result of RSM. For example, in the 5 to 5 test, the RSM1 cannot improve the recognition accuracy for MPCA1, while it helps MPCA2 to achieve higher recognition accuracy randomly. But from 3 to 7 and 1 to 9 tests, it lets all the algorithms have the chance for better recognition accuracy. Although the highest accuracy obtained is better with the combination of the RSM, the mean recognition

Table 1
Top recognition accuracy (%) among 2DPCA, MPCA1 and MPCA2.

Test	PCA	2D PCA	MPCA1	MPCA2
9 to 1				
Parameters	$k=11$	$k_1=3$	$k_1=1, k_2=3$	$k_1=3, k_2=7$
Recognition accuracy	97.5	100	100	100
7 to 3				
Parameters	$k=76$	$k_1=5$	$k_1=7, k_2=2$	$k_1=6, k_2=11$
Recognition accuracy	96.67	96.67	97.5	96.67
5 to 5				
Parameters	$k=43$	$k_1=5$	$k_1=3, k_2=1$	$k_1=7, k_2=3$
Recognition accuracy	94.5	95.5	97.5	97
3 to 7				
Parameters	$k=64$	$k_1=16$	$k_1=6, k_2=1$	$k_1=7, k_2=6$
Recognition accuracy	90	90.71	92.5	91.07
1 to 9				
Parameters	$k=40$	$k_1=10$	$k_1=10, k_2=1$	$k_1=13, k_2=6$
Recognition accuracy	71.39	71.94	74.44	73.89

(a to b) are the numbers mentioned earlier, where a is the number of training images and b is the number of testing images.

Fig. 1. Image examples from the databases.

Table 2
Recognition accuracy (%) among 2D PCA+RSM1, 2D PCA+RSM2, MPCA1+RSM1, MPCA1+RSM2 and MPCA2+RSM1 (short term in the table: 2DR1, 2DR2, M1R1, M1R2 and M2R1 respectively).

Test Parameters	*NN*#	2DR1 $k_{(1)1}=5$	2DR2 $k_{(1)1}=50$	M1R1 $k_{(1)1}=5$ $k_{(2)1}=5$	M1R2 $k_{(1)1}=50$ $k_{(2)1}=50$	M2R1 $k_{(1)1}=5$ $k_{(2)1}=5$
5 to 5	5	95.47 [94.5,96]	94.66 [93,95.5]	97.1 [95,97.5]	96.7 [94.5,97.5]	96.26 [94,97.5]
	10	95.41 [94.5, 96]	94.76 [93, 96]	97.18 [94.5, 97.5]	96.73 [95.5, 97.5]	96.52 [94,97.5]
	15	95.43 [94.5,96]	94.58 [93.5,95.5]	97.20 [95.5,97.5]	96.52 [95,97.5]	96.68 [95,97.5]
3 to 7	5	90.71 [90.71,90.71]	89.99 [87.86,91.07]	92.34 [90.36,93.21]	91.37 [88.57,93.21]	90.96 [90.36,91.07]
	10	90.71 [90.71,90.71]	90.13 [88.21,91.43]	92.36 [90,93.21]	91.63 [90.36,93.21]	90.96 [90,91.07]
	15	90.71 [90.71,90.71]	90.11 [87.5,91.79]	92.24 [90.36,93.21]	91.36 [89.64,92.86]	90.89 [90,91.07]
1 to 9	5	71.98 [71.39,72.22]	71.89 [70,74.17]	73.85 [70.28,74.72]	72.28 [70,75.56]	73.16 [70.83,73.89]
	10	71.88 [70.83,72.22]	71.62 [70,74.17]	73.81 [72.22,74.72]	72.14 [70.28,74.44]	73.19 [71.11,73.89]
	15	72.01 [71.11,72.22]	71.79 [69.44,73.89]	73.94 [73.06,74.44]	72.21 [69.72,73.89]	73.08 [71.11,73.89]

The parameters for 2D PCA, MPCA1 and MPCA2 are adopted from Table 1, which gives the best recognition accuracy. Different numbers of classifiers (5, 10 and 15) are given for the feature selection. The result in this table are presented as the mean accuracy of 50 trails followed by two values within brackets as $[a,b]$ where a is the minimum accuracy and b is the maximum accuracy.

Table 3
Best recognition accuracy (%) for *MPCA* 1_*PCA* and *MPCA* 2_*PCA*.

Tests	Feature # with further reduction	*MPCA* 1_*PCA*		*MPCA* 2_*PCA*	
		Feature #	Accuracy	Feature #	Accuracy
5 to 5	17	388	97	21	97
	19	388	97.5	—	—
3 to 7	33	664	92.14	42	91.07
	42	664	92.5	—	—
1 to 9	37	1032	73.89	78	73.89
	39	1032	74.44	—	—

When the best accuracy for *MPCA* 1_*PCA* is achieved with the minimum dimension, the recognition accuracy for *MPCA* 2_*PCA* is also given. Since the dimension for *MPCA* 2_*PCA* to achieve the best accuracy is less than *MPCA* 1_*PCA*. While the best accuracy for *MPCA* 1_*PCA* is given alone.

accuracy rate did not improve over the different number of classifiers used [15,16]. Therefore, the next experiments do not combine RSM with MPCA1 and MPCA2, and involve only the following combinations: *MPCA* 1_*PCA* and *MPCA* 2_*PCA*. In these experiments, MPCA1 and MPCA2 use the parameters yielding the best accuracy from Table 1.

The results shown in Table 3 prove that the proposed method *MPCA* 1_*PCA* performs better than *MPCA* 2_*PCA* in terms of the recognition accuracy while maintaining similar feature dimensions. Further dimension reduction for MPCA2 is not as necessary as for MPCA1, since MPCA2 has already a highly compact feature basis. Also there are still a lot of redundant features in the feature base of MPCA1. For example, from the tests 1 to 9, it requires 1032 feature dimension after *MPCA* 1 to achieve the best accuracy 74.44%. Applying PCA to the feature shows only 39 are enough to obtain the same accuracy. With less features, the processing time will be reduced as a gain. In the next experiment, the processing time for the nearest neighbor classifier (NNC) is compared. All parameters are set as in the previous experiment (Table 4).

The processing times for the two algorithms using NNC are almost the same and the processing time for *MPCA* 1_*PCA* is longer for the additional decomposition and projection processes.

To verify the benefits gained with the proposed algorithm, a larger dataset is utilized. The face images are collected from the AT&T database [28], the database from University of Essex [29] and the face database from Georgia Institute of Technology [30]. The collective dataset composed of the above three datasets contains 240 subjects with 10 images for each subject for a total of 2400 images. All the images are cropped with the face remaining and normalized into 168×118 resolutions. Moreover, images in the database from the University of Essex and the face database from Georgia Institute of Technology are color images, and they are converted into grayscale for the test.

Because of the largeness of the collective dataset, values for k_1 and k_2 are assigned both a moderate value of 3 based on the prior

Table 4
Total processing times for MPCA1 and *MPCA* 1_*PCA* algorithm based on the parameters obtained in the previous tests using NNC algorithm.

Tests	Feature #	Total processing time (s)
MPCA1		
5 to 5	388	12.3553
3 to 7	664	8.4241
1 to 9	1032	5.6160
MPCA 1_*PCA*		
5 to 5	19	12.6985
3 to 7	42	8.6425
1 to 9	39	6.4740

Table 5
Processing time and recognition accuracy (%) for MPCA1 and *MPCA* 1_*PCA* algorithm using NNC algorithm and SVM algorithm.

Tests	Feature #	NNC		SVM	
		Accuracy	Time (s)	Accuracy	Time (s)
MPCA1					
7 to 3	858	96.67	41.1687	96.67	203.2693
5 to 5	858	94.75	40.9347	95.42	246.9652
3 to 7	858	92.2	62.3068	92.74	222.3638
MPCA 1_*PCA*					
7 to 3	54	97.22	39.5775	96.81	46.6755
5 to 5	40	94.83	30.1862	95.42	53.7111
3 to 7	30	92.44	47.2839	92.26	48.6411
PCA					
7 to 3	54	77.5	19.4377	82.5	30.3578
5 to 5	40	67	20.9977	71	35.5838
3 to 7	30	43.93	15.2101	43.29	35.1314

For MPCA1, choose $k_1=k_2=3$ and for further reduction, the feature number $(k_1 \times I_2+k_2 \times I_1)$ is shown in the table.

results given in Table 1 so as to run a generalized test for both methods (with no guarantee for the best accuracy), since the test is to determine the different processing requirements. A feature number for additional Eigen decomposition is obtained with best accuracy for NNC algorithm (with no guarantee for best accuracy using SVM [31]). Since both MPCA1 and *MPCA* 1_*PCA* assume same processing time for image loading and eigenvector finding, the processing time shown in Table 5 is the total time for the different additional steps required for MPCA1 and *MPCA* 1_*PCA* (as detailed in Proposed Algorithm Section 5). Moreover, in order to highlight the better performance of *MPCA* 1_*PCA*, the comparative results of the direct dimension reduction with PCA to the feature number used by *MPCA* 1_*PCA* are shown in Table 5.

7. Conclusion

Through the comparison between different methods, the *MPCA* 1_*PCA* algorithm can be used for face recognition with fewer features without downgrading the performance in recognition accuracy. A significant contribution of this study is in significantly reducing the feature number and yet consolidating a faster processing time with equally high recognition accuracy. Note that these good results are obtained for both NNC and SVM methods. From the results given earlier in Table 5, it can be observed that for the NNC classification, there are no significant differences in processing time between MPCA1 and *MPCA* 1_*PCA*. However, *MPCA* 1_*PCA* method is faster in all the tests, noting that the larger is the dataset the more pronounced is the difference in processing time between MPCA1 and *MPCA* 1_*PCA*. The recognition accuracy for *MPCA* 1_*PCA* for most times is better than MPCA1; this is on the basis of the pre-assigned value for $k_1=k_2=3$. For the SVM classification, it is obvious that *MPCA* 1_*PCA* algorithm is more computationally efficient than MPCA1 algorithm, although the accuracy is somewhat similar but with the stipulation that the reduced feature number is used with the best accuracy obtained using NNC and not the SVM. Moreover, by comparing the results obtained using PCA of the same feature number as *MPCA* 1_*PCA*, the benefits in term of the recognition rate of the proposed algorithm is obvious.

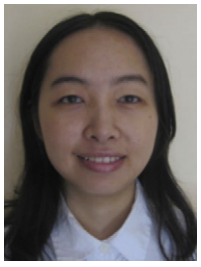
Acknowledgements

The authors would like to thank the support provided by the National Science Foundation under Grants CNS-0426125, HRD-0833093, CNS-0520811, and CNS-0837556. Our thanks also go to those individuals who have contributed and allowed us access to the AT&T database of faces, the face recognition data from the University of Essex, and the Georgia Tech face database.

References

[1] W. Zhao, R. Chellappa, P.J. Phillips, A. Rosenfeld, Face recognition: a literature survey, ACM Computing Surveys (CSUR) 35 (4) (2003) 399–458.
[2] M. Turk, A. Pentland, Eigenfaces for recognition, Journal of Cognitive Neuroscience 3 (1) (1991) 71–86.
[3] M.S. Bartlett, JavierR. Movellan, T.J. Sejnowski, Face recognition by independent component analysis, IEEE Transactions on Neural Networks 13 (6) (2002).
[4] K. Kim, K. Jung, H.J. Kim, Face recognition using kernel principal component analysis, IEEE Signal Processing Letters 9 (2) (2002).
[5] J. Yang, D. Zhang, et al., Two-dimensional PCA: a new approach to appearance-based face representation and recognition, IEEE Transactions on Pattern Analysis and Machine Intelligence 26 (1) (2004) 131–137.
[6] H. Yu, M. Bennamoun, 1D-PCA, 2D-PCA to nD-PCA, in: The 18th International Conference on Pattern Recognition, 2006.
[7] M. Visani, C. Garcia, C. Laurent, Comparing robustness of two-dimensional PCA and eigenfaces for face recognition, in: ICIAR 2004, Lecture Notes in Computer Science, vol. 3212, 2004, pp. 717–724.
[8] A. Xu, X. Jin, Y. Jiang, P. Guo, Complete two-dimensional PCA for face recognition, in: The 18th International Conference on Pattern Recognition, 2006.
[9] C. Lu, W. Liu, X. Liu, S. An, Double sides 2DPCA for face recognition, in: Lecture Notes in Computer Science, vol. 5226, Proceedings of the 4th International Conference on Intelligent Computing: Advanced Intelligent Computing Theories and Applications—with Aspects of Theoretical and Methodological Issues, 2008, pp. 446–459.
[10] Y. Kim, Y. Song, U. Chang, D. Kim, T. Yun, J. Ahn, Face recognition using a fusion method based on bidirectional 2dpca, Applied Mathematics and Computation 205 (2) (2008) 601–607.
[11] H. Kong, L. Wang, E.K. Teoh, X. Li, J. Wang, R. Venkateswarlu, Generalized 2d pca for face image representation and recognition, Neural Networks 18 (5–6) (2005) 585–594 (Special issue: IJCNN 2005).
[12] D. Zhang, Z. Zhou, $(2d)^2$ PCA: two-directional two-dimensional PCA for efficient face representation and recognition, Neurocomputing 69 (1–3) (2005) 224–231.
[13] C. Kim, C. Choi, Image covariance-based subspace method for face recognition, Pattern Recognition 40 (5) (2007) 1592–1604.
[14] V. Nhat, S. Lee, New feature extraction approaches for face recognition, in: Lecture Notes in Computer Science, vol. 3809/2005, pp. 489–497.
[15] N. Nguyen, W. Liu, S. Venkates, Random subspace two-dimensional PCA for face recognition, in: Lecture Notes in Computer Science, vol. 4810, Advances in Multimedia Information Processing-PCM 2007, December 2007, pp. 655–664.
[16] P. Sanguansat, W. Asdornwised, S. Marukatat, S. Jitapunkul, Two-dimensional random subspace analysis for face recognition, in: International Symposium on Communications and Information Technologies, 2007, 17–19 October 2007, pp. 628–631.
[17] Y. Xu, D. Zhang, J. Yang, J. Yang, An approach for directly extracting features from matrix data and its application in face recognition, Neurocomputing 71 (10–12) (2008) 1857–1865.
[18] X. Pan, Q. Ruan, PaImprint recognition using Gabor feature-based (2D)(2)PCA, Neurocomputing 71 (13–15) (August 2008) 3032–3036.

[19] W. Yu, Z. Wang, W. Chen, A new framework to combine vertical and horizontal information for face recognition, Neurocomputing 72 (4–6) (2009) 1084–1091.
[20] H. Lu, K.N. Plataniotis, A.N. Venetsanopoulos, multilinear principal component analysis of tensor objects for recognition, in: The 18th International Conference on Pattern Recognition, 2006.
[21] H. Lu, K.N. Plataniotis, A.N. Venetsanopoulos, MPCA: multilinear principal component analysis of tensor objects, IEEE Transactions on Neural Networks 19 (1) (2008).
[22] D. Tao, X. Li, X. Wu, S. Maybank, Tensor rank one discriminant analysis—a convergent method for discriminative multilinear subspace selection, Neurocomputing 71 (10–12) (2008) 1866–1882.
[23] D. Tao, X. Li, X. Wu, S. Maybank, Geometry mean for subspace selection, IEEE Transaction on Pattern Analysis and Machine Intelligence 31 (2) (2009) 260–274.
[24] Y. Pang, Y. Yuan, X. Li, Effective feature extraction in high dimensional space, IEEE Transaction on Systems, Man and Cybernetics—Part B 38 (6) (2008) 1652–1656 B.C.
[25] J. Wang, Y. Chen, M. Adjouadi, A comparative study of multilinear principal component analysis for face recognition, AIPR, in: 2008 37th IEEE Applied Imagery Pattern Recognition Workshop, 2008, pp. 1–6.
[26] T.K. Ho, The random subspace method for constructing decision trees, IEEE Transactions on Pattern Analysis and Machine Intelligence 20 (8) (1998) 832–844.
[27] N.V. Chawla, K. Bowyer, Random subspaces and subsampling for 2-D face recognition, in: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, 2005, pp. 582–589.
[28] AT&T The database of faces, Online available: 〈http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html〉.
[29] Face Recognition Data, University of Essex, UK. Online available: 〈http://cswww.essex.ac.uk/mv/allfaces/index.html〉.
[30] Georgia Tech face database, Online available: 〈http://www.anefian.com/facereco.htm〉.
[31] C.C. Chang, C.J. Lin, LIBSVM: A Library for Support Vector Machines, 2001. Software available at: 〈http://www.csie.ntu.edu.tw/~cjlin/libsvm〉.



Jin Wang received the B.Eng. degree in Telecommunication from the Wuhan University of Technology, China in 2005 and now is a Ph.D. student with the Electrical and computer Engineering Department of Florida International University. Her research interested focus on the application of machine learning and pattern recognition in computer vision.

Armando Barreto is an Associate Professor in the departments of Electrical and Computer Engineering and Biomedical Engineering. He is the director of the Digital Signal Processing Laboratory. Armando's academic background is in electrical engineering: B.S. degree from UNAM, Mexico, MS Degree from FIU, and Ph.D. degree from UF, Gainesville.

Lu Wang received the B.Eng. degree in Electrical Engineering from Zhejiang University in China in 2001 and the Master degree in Computer Science from Florida International University in 2005 and Ph.D. degree in Electrical and Computer Engineering from Florida International University in 2007. His research interest is the computer vision and image processing.

Yu Chen received the B.S. degree in computer science from Xuzhou Normal University, Jiangsu, China, and the M.S. degree in Electrical Engineering from Florida International University, FL, in 2006. He is a Ph.D. candidate of Electrical and Computer Engineering Department and working as a research assistant at Center for Advanced Technology and Education of Florida International University. His research interests include image processing, computer vision and embedded system.

Naphtali Rishe is the Author of three books on database design and geography; Editor of five books on database management; Inventor of four U.S. patents; Author of over 200 papers; Awardee of over $35 million in research grants; the inaugural FIU Outstanding University Professor; Founder of Terra-Fly.com.

Jean Andrian is actively investigating the applications of Wavelet Transforms to non stationary signal decomposition (music) and time or space-frequency localized signal analysis (edge detection). He also has been involved in semiconductor device research mainly in modeling and characterization. He is interested in applying the well established microelectronic technology to the domain of solid-state electrochemical sensors: solid-state micro-ionics.

Malek Adjouadi is a Professor with the Department of Electrical and Computer Engineering with a joint appointment in Biomedical Engineering. Dr. Adjouadi is the founding Director of the Center for Advanced Technology and Education, funded by NSF. He received his B.S., M.S. and Ph.D. degrees in Electrical Engineering.