Available online at www.sciencedirect.com
SCIENCE DIRECT®
Pattern Recognition 38 (2005) 1717–1732
PATTERN RECOGNITION
THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY
www.elsevier.com/locate/patcog

# Face recognition using a fuzzy fisherface classifier

Keun-Chang Kwak[a,b,*], Witold Pedrycz[b,c]

[a]Department of Electrical Engineering, Chungbuk National University, Cheongju 361-763, Korea
[b]Department of Electrical and Computer Engineering, University of Alberta, Edmonton, Canada T6G 2G7
[c]Systems Research Institute, Polish Academy of Sciences, 01-447 Warsaw, Poland

Received 11 March 2004; received in revised form 24 January 2005; accepted 24 January 2005

**Abstract**

In this study, we are concerned with face recognition using fuzzy fisherface approach and its fuzzy set based augmentation. The well-known fisherface method is relatively insensitive to substantial variations in light direction, face pose, and facial expression. This is accomplished by using both principal component analysis and Fisher's linear discriminant analysis. What makes most of the methods of face recognition (including the fisherface approach) similar is an assumption about the same level of typicality (relevance) of each face to the corresponding class (category). We propose to incorporate a gradual level of assignment to class being regarded as a membership grade with anticipation that such discrimination helps improve classification results. More specifically, when operating on feature vectors resulting from the PCA transformation we complete a Fuzzy K-nearest neighbor class assignment that produces the corresponding degrees of class membership. The comprehensive experiments completed on ORL, Yale, and CNU (Chungbuk National University) face databases show improved classification rates and reduced sensitivity to variations between face images caused by changes in illumination and viewing directions. The performance is compared vis-à-vis other commonly used methods, such as eigenface and fisherface.


*Keywords:* Face recognition; Eigenface; Fisherface; Principal component analysis (PCA); Fisher's linear discriminant (FLD); Fuzzy nearest neighbor classifier

## 1. Introductory comments

Biometrics is aimed at capturing and use of physiological or behavioral characteristics for personal identification or individual verification purposes. Face recognition is a natural intuitively appealing and straightforward biometric method. Face recognition has been researched in various areas such as computer vision, image processing, and pattern recognition. In practice, face recognition is a very difficult problem due to a substantial variation in light direction, different face poses, and diversified facial expressions. The most well-known classification techniques used for face recognition are those of eigenface [1] and fisherface [2]. The eigenface method relies on a transformation of feature vectors by utilizing principal components (and is referred to as principal component analysis, PCA); the other naming used there is the Karhunen–Loeve (KL) expansion. In essence, the PCA dwells on a linear projection of a high-dimensional face image space into a new low-dimensional feature space. The major problem coming with the use of the eigenface technique is that it can be affected by variations in illumination conditions and different facial expressions. It is also worth stressing that the PCA is oriented toward the representation in low-dimensional spaces but not necessarily optimal in terms of face classification (as the issue of discrimination between classes is not a part of the problem formulation).

* Corresponding author. Department of Electrical and Computer Engineering, University of Alberta, Edmonton, Canada T6G 2G7. Tel.: +1 780 492 4661; fax: +1 780 492 1811.

*E-mail addresses:* kwak@ece.ualberta.ca (K.-C. Kwak), pedrycz@ece.ualberta.ca (W. Pedrycz).

0031-3203/$30.00 
doi:10.1016/j.patcog.2005.01.018

There are numerous extensions to the standard PCA such as mixture-of-eigenfaces [3], topological PCA [4], kernel PCA [5], eigen-wavelet method [6], to name a few of them. In spite of these expansions, PCA still retains unwanted variations due to different conditions caused by lighting and facial expressions [2].

The second well-known approach coming under the name of fisherface is insensitive to large variation in the conditions we have already enumerated above. It uses both PCA and Fisher's linear discriminant (FLD). It is worth stressing that by maximizing the ratio of between-scatter matrix and within-scatter matrix, FLD produces well separated classes in a low-dimensional subspace, even under severe variation in lighting and facial expressions. There are a various enhancements made to FLD: Enhanced FLD [7], direct linear discriminant analysis (LDA) [8], uncorrelated discriminant transformation [9], most discriminating feature (MDF) [10]. While these two approaches are quite a dominant, there are a number of other techniques developed in the setting of computational intelligence including neural networks (NN) [11,12], fuzzy logic [13], and genetic algorithm (GA) [14].

The objective of this study is to revisit the fisherface technique (which is well established and has already enjoyed a significant level of success) and augment it by some mechanisms of fuzzy sets. By taking advantage of the technology of fuzzy sets [19], a number of studies have been carried out for fuzzy image filtering, fuzzy image segmentation, and fuzzy edge detection with an ultimate objective to cope with the factor of uncertainty being inherently present in many problems of image processing and pattern recognition [15]. From this point of view, we address the uncertainty associated with a significant variation in illumination, viewing directions, and facial expression in the face images.

By analyzing the existing fisherface, we note that the algorithm dwells on the concept of a binary (yes–no) class assignment meaning that the faces come fully assigned to the given classes (categories). Evidently, as the faces are significantly affected by numerous environmental conditions (including illumination, poses, etc.), it is advantageous to investigate these factors and quantify their impact on their "internal" (viz. algorithm-driven) class assignment. In essence, the intent is to reflect all these factors in a "soft" viz. fuzzy class allocation to the individual faces under consideration. Interestingly, the idea of such "fuzzification" of class assignment has been around for a long time and can be dated back to the results published by Keller et al. [16] coming under the notion of a fuzzy $k$-nearest neighbor classifier. We envision that this concept can be used effectively to enhance the performance of the fisherface (in the sequel, this new generalization will be referred to as a fuzzy fisherface).

This material is organized in the following manner. Section 2 provides a concise summary of the well-known techniques of eigenface and fisherface and introduces all required notation. This section can serve as a sound reference point when looking at the expansion of the fisherface technique and serves as a prerequisite to the fuzzy fisherface approach outlined in Section 3. Section 4 reports on comprehensive simulation results completed for several commonly used face databases such as ORL [17], Yale [18], and CNU (Chungbuk National University). Finally, concluding comments are included in Section 5.

## 2. Conventional face recognition methods: a brief overview

It is instructive to summarize the main notation we are about to use in the study. While the notation is standard to a high extent, its careful inspection becomes advantageous when walking through the algorithms presented in the paper. The faces are $n$ by $n$ matrices and these are represented in the form of $n^2$-dimensional vectors $\mathbf{z}_i$. The number of faces is equal to $N$ and they belong to "$c$" classes (categories). Furthermore by $Z$ we denote a training family of faces. The statistical characterization of the faces is standard: by $R$ we denote a covariance matrix of the images, $\bar{\mathbf{z}}$ describes a mean image of the faces in the training set, while $\mathbf{e}_i$ stands for the $i$th eigenvector of the covariance matrix. Likewise by $S_{\mathrm{W}}$ and $S_{\mathrm{B}}$ we describe a within-class and between-class scatter matrices, respectively. To emphasize the origin of some matrices we will be adding pertinent subscripts, say $W_{\mathrm{FLD}}$ indicates a between-class scatter matrix generated by the Fisher's linear discriminant technique.

### 2.1. Eigenface method

PCA is a well-known technique commonly exploited in multivariate linear data analysis. The main underlying concept is to reduce the dimensionality of a data set while retaining as much variation as possible in the data set. Let a face image be a two-dimensional $n \times n$ array of pixels. The corresponding image $\mathbf{z}_i$ is viewed as a vector with $n^2$ coordinates that results from a concatenation of successive rows of the image. Denote the training set of $N$ faces by $Z = \{\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_N\}$. Define the corresponding covariance matrix in the standard manner

$$R = \frac{1}{N} \sum_{i=1}^{N} (\mathbf{z}_i - \bar{\mathbf{z}})(\mathbf{z}_i - \bar{\mathbf{z}})^{\mathrm{T}} = \Phi\Phi^{\mathrm{T}}, \tag{1}$$

where

$$\bar{\mathbf{z}} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{z}_i. \tag{2}$$

Let $E = (\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_r)$ be a matrix formed by "$r$" eigenvectors corresponding to the "$r$" largest eigenvalues. Thus, for a set of original face images $Z$, their reduced feature vectors $X = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N)$ are obtained by projecting them into the PCA-transformed space following the linear transformation:

$$\mathbf{x}_i = E^{\mathrm{T}}(\mathbf{z}_i - \bar{\mathbf{z}}) \tag{3}$$

with $\mathbf{x}_i$ being the result of this transformation.

The classification task is cast in the reduced PCA space as all the computations of the distances are carried out there. Given two images $\mathbf{z}'$ and $\mathbf{z}$ in the original $n^2$-dimensional space, a distance between them is defined in the form

$$\varepsilon(\mathbf{z}, \mathbf{z}') = \|\mathbf{x} - \mathbf{x}'\|, \tag{4}$$

where $\mathbf{x}$ and $\mathbf{x}'$ are PCA-transformed feature vectors of face image $\mathbf{z}$ and $\mathbf{z}'$, respectively.

The method in the form outlined above can lead to extremely large and computationally challenging covariance matrices. This problem can be alleviated by referring to some basic finding known in linear algebra that states that the eigenvalues of $\Phi\Phi^{\mathrm{T}}$ and $\Phi^{\mathrm{T}}\Phi$ are the same. Furthermore, the eigenvectors of $\Phi\Phi^{\mathrm{T}}$ are the same as the eigenvectors of $\Phi^{\mathrm{T}}\Phi$ multiplied by the matrix $\Phi$ and afterwards normalized. This is the motivation behind the snapshot method reported in the literature [1]. This algorithm is used to form the eigenspace on a basis of $N \times N$ matrices, which constitutes a substantial reduction of problem dimensionality originally involving $n^2 \times n^2$ covariance matrices.

Let us again stress that in virtue of the PCA calculations that do not involve any class information, the method is not capable of taking advantage of this discriminatory aspects subsequently being faced with quite a detrimental classification performance. To illustrate this point, let us consider 200 two-dimensional synthetic data points belonging to two classes. The data along with its one-dimensional PCA transformation are shown in Fig. 1. These two classes are drawn from two-dimensional Gaussian probability distribution with the covariance variance matrix $\Sigma$ and mean $\mu$, that is

$$\text{Class 1:}\quad \Sigma_1 = \begin{bmatrix} 0.1^2 & 0 \\ 0 & 0.8^2 \end{bmatrix} \quad \text{and} \quad \mu_1 = [0 \quad 0],$$

$$\text{Class 2:}\quad \Sigma_2 = \begin{bmatrix} 0.01^2 & 0 \\ 0 & 0.8^2 \end{bmatrix} \quad \text{and} \quad \mu_2 = [-1 \quad 2].$$

Noticeably, the PCA projections are optimal as far as dimensionality reduction is concerned but the fail completely when considered as a possible classification environment. The two classes fully overlap and it becomes evident that any classifier is going to fail on this task (even though the classification in the original two-dimensional space does not create any challenge).

### 2.2. Fisherface method

While PCA is commonly used to project face patterns from a high-dimension image space to some low-dimensional space, it is aimed at data representation. In a way it defines a subspace that exhibits the greatest variance of the projected sample vectors among all the subspaces. However, such projection may not be effective for classification since large and unwanted variations may be still retained. Consequently, the projected samples for each class may not be well clustered and result in the patterns being smeared together, cf. [2]. In contrast, the FLD or LDA is an example of a class-specific method that finds the optimal classification-driven projection of patterns. Rather than finding a projection that maximizes the projected variance, FLD determines a projection defined as $V = W_{\mathrm{FLD}}^{\mathrm{T}} X$ (where $W_{\mathrm{FLD}}^{\mathrm{T}}$ denotes the optimal projection matrix). The projection takes advantage of class information and maximizes the ratio between the between-class scatter and the within-class scatter matrices. Consequently, the ensuing classification mechanisms are simplified in the projected space. The fisherface comprises two phases: first it projects the image set to a lower-dimensional space using PCA that is followed by the FLD phase. The use of the LDA helps us achieve non-singularity of the resulting within-class scatter matrix $S_{\mathrm{W}}$ prior to any computations of the optimal projection $W_{\mathrm{FLD}}$.

Proceeding with the presentation of the fisherface approach, let the between-class scatter matrix be defined in the usual manner

$$S_{\mathrm{B}} = \sum_{i=1}^{c} N_i (\mathbf{m}_i - \overline{\mathbf{m}})(\mathbf{m}_i - \overline{\mathbf{m}})^{\mathrm{T}}, \tag{5}$$

where $N_i$ is the number of vectors in the $i$th class $C_i$ and $\overline{\mathbf{m}}$ stands for the mean of all vectors (images), $\mathbf{m}_i$ is the mean of vectors transformed by PCA and dealing with class $C_i$. The within-class scatter matrix reads as

$$S_{\mathrm{W}} = \sum_{i=1}^{c} \sum_{x_k \in C_i} (\mathbf{x}_k - \mathbf{m}_i)(\mathbf{x}_k - \mathbf{m}_i)^{\mathrm{T}} = \sum_{i=1}^{c} S_{\mathrm{W}_i}, \tag{6}$$

where $S_{\mathrm{W}_i}$ is the covariance matrix of class $C_i$. The optimal projection matrix $W_{\mathrm{FLD}}$ is chosen in such a manner so that it forms a matrix with orthonormal columns that maximizes the ratio of the determinant of the between-class scatter matrix of the projected samples and the determinant of the within-class scatter matrix of the projected patterns, i.e.,

$$W_{\mathrm{FLD}} = \arg\max_{W} \frac{|W^{\mathrm{T}} S_{\mathrm{B}} W|}{|W^{\mathrm{T}} S_{\mathrm{W}} W|} = [\mathbf{w}_1 \quad \mathbf{w}_2 \quad \cdots \quad \mathbf{w}_m], \tag{7}$$

where $\{\mathbf{w}_i \,|\, i=1, 2, \ldots, m\}$ is the set of generalized eigenvectors (discriminant vectors) of $S_{\mathrm{B}}$ and $S_{\mathrm{W}}$ corresponding to the $c-1$ largest generalized eigenvalues $\{\lambda_i \,|\, i=1, 2, \ldots, m\}$, that is

$$S_{\mathrm{B}} \mathbf{w}_i = \lambda_i S_{\mathrm{W}} \mathbf{w}_i, \quad i = 1, 2, \ldots, m. \tag{8}$$

However, the rank of $S_{\mathrm{B}}$ is $c-1$ or less because it is the sum of "$c$" matrices of rank one or less. Thus, the upper bound on the values of "$m$" is equal to $c-1$. Similarly, the rank of $S_{\mathrm{W}}$ is at most $N-c$. For a set of $N$ face images of $n^2$ pixels, where $N$ is usually smaller than $n^2$, the within-scatter matrix $S_{\mathrm{W}}$ is always singular. This means that the projected within-scatter matrix can be zero if the projection matrix is not chosen properly. The above problem can be avoided by first projecting the image set to a lower-dimensional space using PCA. Then the resulting within-class scatter matrix

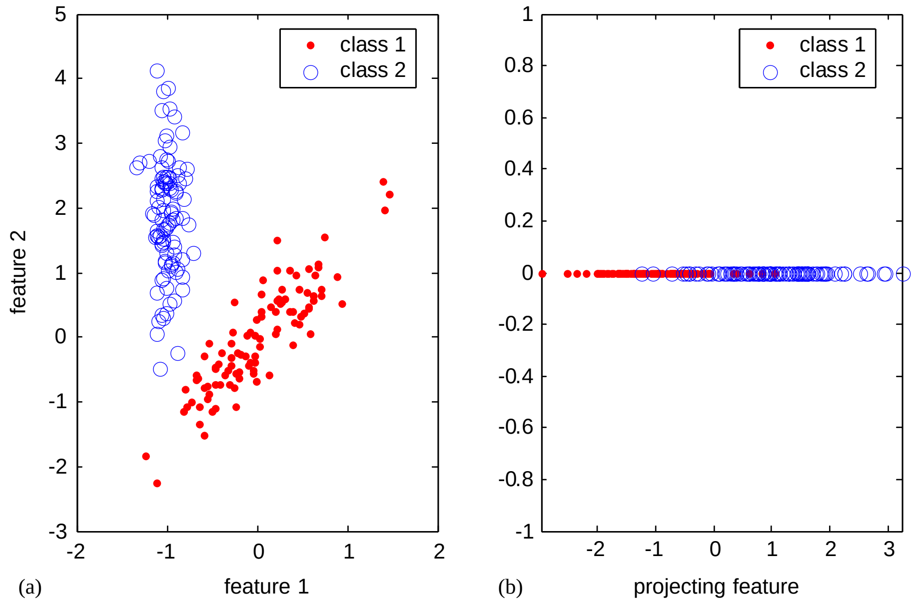


Fig. 1. PCA projection for a two-class problem: (a) original data set and (b) its reduced PCA transformation.

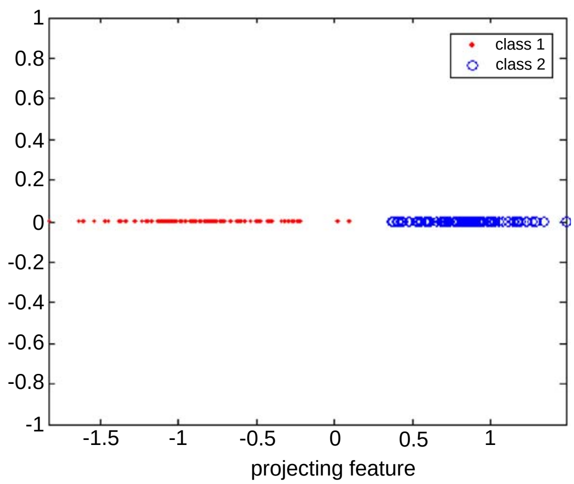


Fig. 2. FLD projection for the two-class synthetic data.

$S_W$ is nonsingular and this facilitates the computations of the optimal projection $W_{FLD}$ [2].

The feature vectors $V = (\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_N)$ for any face images $\mathbf{z}_i$ can be calculated as follows [11]:

$$\mathbf{v}_i = W_{FLD}^T \mathbf{x}_i = W_{FLD}^T E^T (\mathbf{z}_i - \bar{\mathbf{z}}). \tag{9}$$

Alluding again to the synthetic data, Fig. 2 visualizes the FLD transformation—noticeably the results are totally different and the important discriminatory properties are fully retained.

## 3. Fuzzy fisherface approach

The fisherface presented in the previous section has exhibited a substantial advantage over the PCA as far as classification aspects are concerned. The question arises as to further improvements of the approach. A certain alternative that emerges is concerned with more "sophisticated" usage of class assignment of patterns (faces). In particular, we may envision some possibilities to refinement of classification results so that they could affect the within-class and between-class scatter matrices and enhance the performance of the classifier. Having this in mind, an obvious choice is to look at the fundamental results available in the setting of fuzzy nearest neighbor classifiers. The term of a fuzzy partition becomes an important notion to be considered. Given a set of feature vectors transformed by the PCA, $X = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$, a fuzzy "$c$"-class partition of these vectors specifies the degrees of membership of each vector to the classes. As usual, the partition matrix denoted by $U = [\mu_{ij}]$ for $i = 1, 2, \ldots, c$, and $j = 1, 2, \ldots, N$ satisfies two obvious properties

$$\sum_{i=1}^{c} \mu_{ij} = 1, \tag{10}$$

$$0 < \sum_{j=1}^{N} \mu_{ij} < N. \tag{11}$$

The first condition helps us assure sound mathematical tractability. For instance, in a three class problem, the membership grades close to 0.5 indicate that the vector exhibits a high degree of membership to several classes, cf. [16]. The computations of the membership degrees are realized through a sequence of steps:

*Step* 1: Compute the Euclidean distance matrix between pairs of feature vectors in the training.

*Step* 2: Set diagonal elements of this matrix to infinity (practically place large numeric values there).

*Step* 3: Sort the distance matrix (treat each of its column separately) in an ascending order. Collect the class labels of the patterns located in the closest neighborhood of the pattern under consideration (as we are concerned with "$k$" neighbors, this returns a list of "$k$" integers).

*Step* 4: Compute the membership grade to class "$i$" for $j$th pattern using the expression proposed in the literature [16].

$$\mu_{ij} = \begin{cases} 0.51 + 0.49(n_{ij}/k) & \text{if } i = \text{the same as the label} \\ & \text{of the } j\text{th pattern,} \\ 0.49(n_{ij}/k) & \text{if } i \neq \text{the same as the label} \\ & \text{of the } j\text{th pattern.} \end{cases} \tag{12}$$

In the above expression, $n_{ij}$ stands for the number of the neighbors of the $j$th data (pattern) that belong to the $i$th class. After the examination of the membership allocation formula we conclude that the method attempts to "fuzzify" or refine the membership grades of the labeled patterns. The "dominant" membership has not been affected (so the pattern is not moved to a different category) yet we end up with some refinement of membership grades. Intuitively, if there are very few neighbors of the pattern that belong to the same category, the membership grade is kept close to 0.51; refer to (12). Alternatively, if $n_{ij} = k$ meaning that all neighbors are in the same class as the pattern under consideration, then $\mu_{ij}$ returns 1.0 (which is an appealing outcome).

To illustrate this method, we consider nine two-dimensional patterns belonging to three-classes; refer also to Fig. 3.

| No. | $feature_1$ | $feature_2$ | class |
|---|---|---|---|
| 1 | 0.2 | 0.3 | 1 |
| 2 | 0.3 | 0.2 | 1 |
| 3 | 0.4 | 0.3 | 1 |
| 4 | 0.5 | 0.5 | 2 |
| 5 | 0.6 | 0.4 | 2 |
| 6 | 0.5 | 0.6 | 2 |
| 7 | 0.7 | 0.3 | 3 |
| 8 | 0.8 | 0.4 | 3 |
| 9 | 0.7 | 0.5 | 3 |

Proceeding with step [1], the distance matrix comes with the following entries:

| No. | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 0.1414 | 0.2000 | 0.3606 | 0.4123 | 0.4243 | 0.5000 | 0.6083 | 0.5385 |
| 2 | 0.1414 | 0 | 0.1414 | 0.3606 | 0.3606 | 0.4472 | 0.4123 | 0.5385 | 0.5000 |
| 3 | 0.2000 | 0.1414 | 0 | 0.2236 | 0.2236 | 0.3162 | 0.3000 | 0.4123 | 0.3606 |
| 4 | 0.3606 | 0.3606 | 0.2236 | 0 | 0.1414 | 0.1000 | 0.2828 | 0.3162 | 0.2000 |
| 5 | 0.4123 | 0.3606 | 0.2236 | 0.1414 | 0 | 0.2236 | 0.1414 | 0.2000 | 0.1414 |
| 6 | 0.4243 | 0.4472 | 0.3162 | 0.1000 | 0.2236 | 0 | 0.3606 | 0.3606 | 0.2236 |
| 7 | 0.5000 | 0.4123 | 0.3000 | 0.2828 | 0.1414 | 0.3606 | 0 | 0.1414 | 0.2000 |
| 8 | 0.6083 | 0.5385 | 0.4123 | 0.3162 | 0.2000 | 0.3606 | 0.1414 | 0 | 0.1414 |
| 9 | 0.5385 | 0.5000 | 0.3606 | 0.2000 | 0.1414 | 0.2236 | 0.2000 | 0.1414 | 0 |

In step [2], the diagonal elements are replaced by infinity (Inf).

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Inf | 0.1414 | 0.2000 | 0.3606 | 0.4123 | 0.4243 | 0.5000 | 0.6083 | 0.5385 |
| 0.1414 | Inf | 0.1414 | 0.3606 | 0.3606 | 0.4472 | 0.4123 | 0.5385 | 0.5000 |
| 0.2000 | 0.1414 | Inf | 0.2236 | 0.2236 | 0.3162 | 0.3000 | 0.4123 | 0.3606 |
| 0.3606 | 0.3606 | 0.2236 | Inf | 0.1414 | 0.1000 | 0.2828 | 0.3162 | 0.2000 |
| 0.4123 | 0.3606 | 0.2236 | 0.1414 | Inf | 0.2236 | 0.1414 | 0.2000 | 0.1414 |
| 0.4243 | 0.4472 | 0.3162 | 0.1000 | 0.2236 | Inf | 0.3606 | 0.3606 | 0.2236 |
| 0.5000 | 0.4123 | 0.3000 | 0.2828 | 0.1414 | 0.3606 | Inf | 0.1414 | 0.2000 |
| 0.6083 | 0.5385 | 0.4123 | 0.3162 | 0.2000 | 0.3606 | 0.1414 | Inf | 0.1414 |
| 0.5385 | 0.5000 | 0.3606 | 0.2000 | 0.1414 | 0.2236 | 0.2000 | 0.1414 | Inf |

In step [3], the distance matrix is sorted (which is done separately for each column of the matrix)

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 0.1414 | 0.1414 | 0.1414 | 0.1000 | 0.1414 | 0.1000 | 0.1414 | 0.1414 | 0.1414 |
| 0.2000 | 0.1414 | 0.2000 | 0.1414 | 0.1414 | 0.2236 | 0.1414 | 0.1414 | 0.1414 |
| 0.3606 | 0.3606 | 0.2236 | 0.2000 | 0.1414 | 0.2236 | 0.2000 | 0.2000 | 0.2000 |
| 0.4123 | 0.3606 | 0.2236 | 0.2236 | 0.2000 | 0.3162 | 0.2828 | 0.3162 | 0.2000 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 0.4243 | 0.4123 | 0.3000 | 0.2828 | 0.2236 | 0.3606 | 0.3000 | 0.3606 | 0.2236 |
| 0.5000 | 0.4472 | 0.3162 | 0.3162 | 0.2236 | 0.3606 | 0.3606 | 0.4123 | 0.3606 |
| 0.5385 | 0.5000 | 0.3606 | 0.3606 | 0.3606 | 0.4243 | 0.4123 | 0.5385 | 0.5000 |
| 0.6083 | 0.5385 | 0.4123 | 0.3606 | 0.4123 | 0.4472 | 0.5000 | 0.6083 | 0.5383 |
| Inf | Inf | Inf | Inf | Inf | Inf | Inf | Inf | Inf |

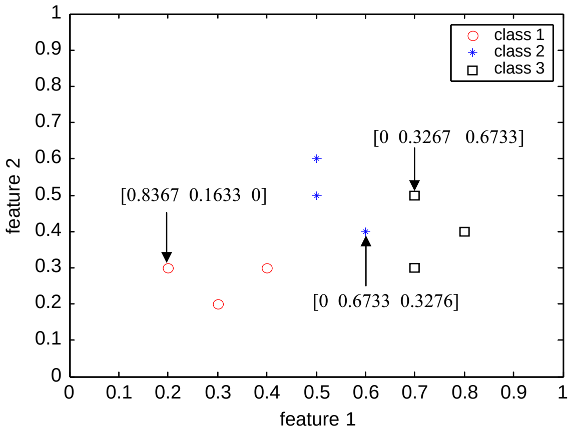


Fig. 3. Fuzzy membership degree using FKNN initialization ($k$=3).

If we consider $k = 3$ neighbors, then the indexes associated with each distance value give rise to the lists

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| $k=1$ | 2 | 1 | 2 | 6 | 4 | 4 | 5 | 9 | 5 |
| $k=2$ | 3 | 3 | 1 | 5 | 9 | 5 | 8 | 7 | 8 |
| $k=3$ | 4 | 4 | 5 | 9 | 7 | 9 | 9 | 5 | 4 |

The classes of $i$th nearest point of $j$th input vector are as follows:

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 2 | 2 | 2 | 2 | 3 | 2 |
| 1 | 1 | 1 | 2 | 3 | 2 | 3 | 3 | 3 |
| 2 | 2 | 2 | 3 | 3 | 3 | 3 | 2 | 2 |

Realizing step [4], the computed membership grades are given as

| | | |
|---|---|---|
| 0.8367 | 0.1633 | 0 |
| 0.8367 | 0.1633 | 0 |
| 0.8367 | 0.1633 | 0 |
| 0 | 0.8367 | 0.1633 |
| 0 | 0.6733 | 0.3267 |
| 0 | 0.8367 | 0.1633 |
| 0 | 0.1633 | 0.8367 |
| 0 | 0.1633 | 0.8367 |
| 0 | 0.3267 | 0.6733 |

To help further clarify the origin of these numbers, let us consider membership grades (0 0.6733 0.3267) of the 5th sample point (that is labeled to belong to class 2). The membership degrees resulting from the calculations (12) are given as

(1) Class 1 $\neq$ assigned class 2, $0.49(n_{ij}/k) = 0.49(0/3) = 0$,
(2) Class 2 = assigned class 2, $0.51 + 0.49(n_{ij}/k) = 0.51 + 0.49(1/3) = 0.6733$,
(3) Class 3 $\neq$ assigned class 2, $0.49(n_{ij}/k) = 0.49(2/3) = 0.3267$.

The above class refinement is intuitively appealing as revealed in Fig. 3. In particular, we note that some patterns become somewhat split between the two classes which is a useful sign of alert pointing out at possible revision of the binary membership allocation.

The results of the FKNN are used in the computations of the statistical properties of the patterns such as the mean value and scatter covariance matrices—the constructs being at heart of the fisherface method. Taking into account the membership grades, the mean vector of each class $\tilde{\mathbf{m}}_i$ is calculated as follows:

$$\tilde{\mathbf{m}}_i = \frac{\sum_{j=1}^{N} \mu_{ij} \mathbf{x}_j}{\sum_{j=1}^{N} \mu_{ij}}, \tag{13}$$

$i = 1, 2, \ldots, c$. The between-class fuzzy scatter matrix $S_{\mathrm{FB}}$ and within-class fuzzy scatter matrix $S_{\mathrm{FW}}$ incorporate the membership values in their calculations

$$S_{\mathrm{FB}} = \sum_{i=1}^{c} N_i (\tilde{\mathbf{m}}_i - \overline{\mathbf{m}})(\tilde{\mathbf{m}}_i - \overline{\mathbf{m}})^{\mathrm{T}}, \tag{14}$$

$$S_{\mathrm{FW}} = \sum_{i=1}^{c} \sum_{x_k \in C_i} (\mathbf{x}_k - \tilde{\mathbf{m}}_i)(\mathbf{x}_k - \tilde{\mathbf{m}}_i)^{\mathrm{T}} = \sum_{i=1}^{c} S_{\mathrm{FW}_i}. \tag{15}$$

The optimal fuzzy projection $W_{\text{F-FLD}}$ and the feature vector transformed by fuzzy fisherface method follows the expressions:

$$W_{\text{F-FLD}} = \arg\max_{W} \frac{|W^{\mathrm{T}} S_{\mathrm{FB}} W|}{|W^{\mathrm{T}} S_{\mathrm{FW}} W|}, \tag{16}$$

$$\tilde{\mathbf{v}}_i = W_{\text{F-FLD}}^{\mathrm{T}} \mathbf{x}_i = W_{\text{F-FLD}}^{\mathrm{T}} E^{\mathrm{T}} (\mathbf{z}_i - \overline{\mathbf{z}}). \tag{17}$$

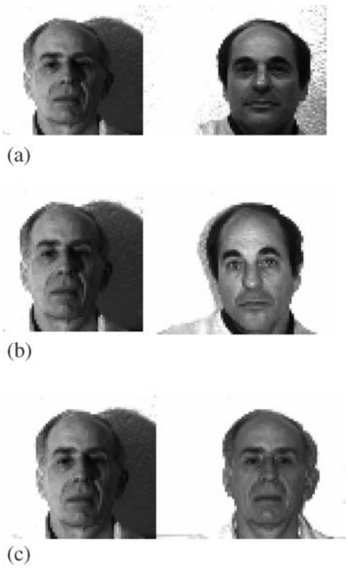


Fig. 4. Comparison of recognition results: (a) eigenface, left: test image (class 5), right: recognized image (class 12), (b) fisherface, left: test image (class 5), right: recognized image (class 12), (c) fuzzy fisherface, left: test image (class 5), right: recognized image (class 5).

Referring to Fig. 3, we note that the mean vectors computed with binary class assignment and membership grades are different

$$\tilde{\mathbf{m}}_1 = \begin{bmatrix} 0.3195 \\ 0.2943 \end{bmatrix}, \quad \tilde{\mathbf{m}}_2 = \begin{bmatrix} 0.5242 \\ 0.4355 \end{bmatrix}, \quad \tilde{\mathbf{m}}_3 = \begin{bmatrix} 0.6964 \\ 0.4127 \end{bmatrix},$$

$$\mathbf{m}_1 = \begin{bmatrix} 0.3000 \\ 0.2667 \end{bmatrix}, \quad \mathbf{m}_2 = \begin{bmatrix} 0.5333 \\ 0.5000 \end{bmatrix}, \quad \mathbf{m}_3 = \begin{bmatrix} 0.7333 \\ 0.4000 \end{bmatrix}.$$

Through a direct inspection, one can conclude that the fuzzy means are shifted closer each other. The between-class fuzzy scatter matrix and within-class fuzzy scatter matrix are obtained by fuzzy FLD as follows:

$$S_{\mathrm{FW}} = \begin{bmatrix} 0.1250 & 0.0217 \\ 0.0217 & 0.0969 \end{bmatrix}, \quad S_{\mathrm{FB}} = \begin{bmatrix} 0.1904 & 0.0604 \\ 0.0604 & 0.0319 \end{bmatrix}.$$

For comparison, the binary class allocation yields the results

$$S_{\mathrm{W}} = \begin{bmatrix} 0.0333 & -0.0100 \\ -0.0100 & 0.0466 \end{bmatrix}, \quad S_{\mathrm{B}} = \begin{bmatrix} 0.2822 & 0.0922 \\ 0.0922 & 0.0822 \end{bmatrix}.$$

Finally, the optimal projection of the proposed method and FLD are obtained, respectively.

$$W_{\text{F-FLD}} = \begin{bmatrix} 0.9769 \\ 0.2135 \end{bmatrix}, \quad W_{\mathrm{FLD}} = \begin{bmatrix} 0.9081 \\ 0.4185 \end{bmatrix}.$$

The following example coming from Yale face database [18] illustrates how the proposed method improves the

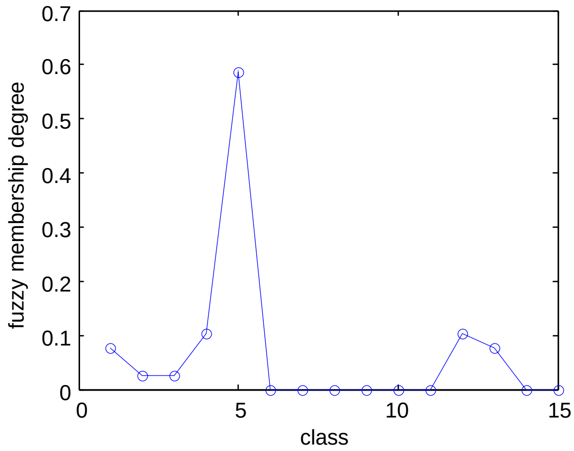


Fig. 5. Membership degrees to different classes—fuzzy fisherface; refer to Fig. 4(c) right image.

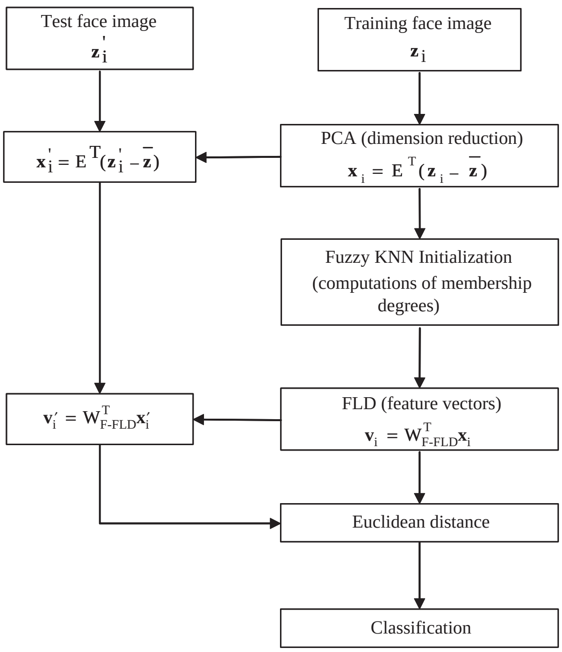


Fig. 6. A general flow of computing for the fuzzy fisherface method.

performance of the classification process. Fig. 4 shows results obtained using eigenface, fisherface, and the proposed method in respectively. As shown there, the misclassification results occur due to large variation in light direction. The fuzzy fisherface approach comes with better performance. The membership degrees are shown in Fig. 5.

Finally, a general flowchart of computing for the fuzzy fisherface is included in Fig. 6. The number of neighbors (as

(a)

(b)

(c)

Fig. 7. Samples of face image in three face databases: (a) ORL face database, (b) Yale face database, (c) CNU face database.

indicated in this flowchart) is usually experiment—driven and needs to be adjusted for a specific dataset at hand.

## 4. Experimental studies

In this section, we elaborate on the experimental findings for a number of well-known and commonly used face databases as shown in Fig. 7. In all scenarios we contrast the results of the fuzzy fisherface approach with the previous techniques.

### 4.1. ORL face databases

The ORL database [17] comprises 400 face images coming from 40 individuals; the pictures were taken in different environments. The total number of images for each individual is equal to 10. These images vary in position, rotation, scale, and facial expression. For some individuals, the images were taken at different times, varying facial details (glasses/no glasses). Each image was digitized and stored as an $112 \times 92$ pixel array whose gray levels ranged between 0 and 255. Some samples from the ORL databases are shown in Fig. 7(a). The training and testing set are selected randomly by choosing for each subject three cases:

Case 1: number of training set for one person: 5, test set: 5.
Case 2: number of training set for one person: 4, test set: 6.
Case 3: number of training set for one person: 6, test set: 4.

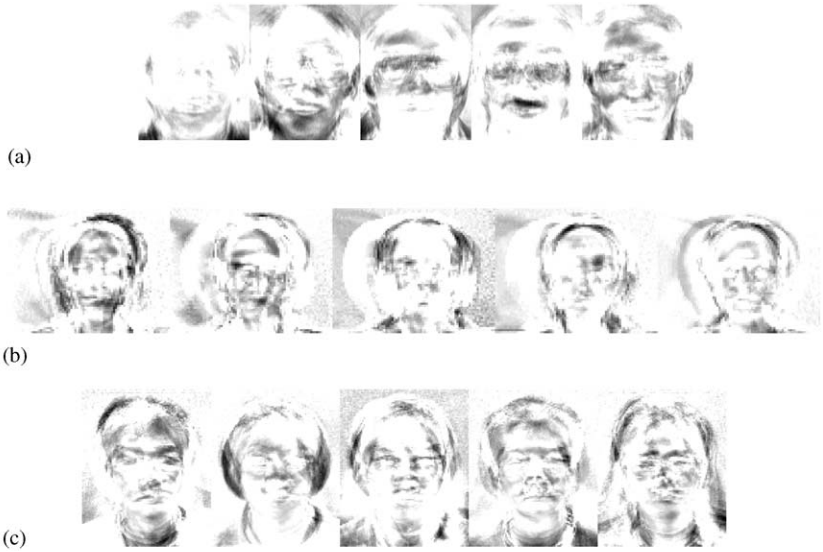


Fig. 8. Fisherfaces obtained by the fuzzy fisherface method: (a) ORL, (b) Yale, (c) CNU.

Table 1
Comparison of mean and standard deviation for recognition rates (ORL)

| | Eigenface (PCA) (%) | Fisherface (PCA + LDA) (%) | Fuzzy fisherface (Fuzzy + PCA+ LDA) (%) |
|---|---|---|---|
| Case 1 | $93.6 \pm 2.55$ | $95.1 \pm 2.10$ | $95.5 \pm 1.99$ |
| Case 2 | $96.12 \pm 1.52$ | $97.37 \pm 0.87$ | $98.37 \pm 0.89$ |
| Case 3 | $91.87 \pm 1.11$ | $92.54 \pm 1.45$ | $94.12 \pm 1.63$ |

Table 2
Minimal and maximal values the of recognition rates—see Fig. 11

| | Min | Max |
|---|---|---|
| 1st random data set | 94.5 | 96.5 |
| 2nd random data set | 94.5 | 95.5 |
| 3rd random data set | 94.5 | 96.5 |
| 4th random data set | 95.5 | 96.5 |

This procedure was repeated 10 times by randomly choosing different training and testing sets. Here, we determined 60 eigenvectors representing the best performance in ten experiments. The number of discriminant vectors corresponding to the $c - 1$ largest generalized eigenvalues is 39. Fig. 8(a) includes some of the fisherface image obtained by the proposed method. The recognition rates for various experiments are shown in Fig. 9. The mean recognition rates for the three cases are shown in Fig. 10. Table 1 contains a comparative analysis of the mean and standard deviation for the obtained recognition rates. The proposed fuzzy fisherface method outperformed other classifiers and this occurred consistently in all cases. Fig. 11 summarizes optimal classification (recognition) rates obtained for optimal values of "$k$" where the experiments were repeated for 10 randomly selected data subsets. The optimal number of neighbors $(k)$ was determined by repeating the experiment for successive values of "$k$" and choosing such $k_0$ that returned the best classification rate. In addition, Table 2 shows the minimal and maximal values of the recognition rates being reported in Fig. 11.

Interestingly, while the number of neighbors exhibits some impact on the performance of the classifier, it is rather limited and does not affect the method in any adverse manner. The 1% difference still makes the fuzzy fisherface outperform the two other methods.

### 4.2. Yale face databases

The Yale face database contains 165 face images of 15 individuals. There are 11 images per subject, one for each facial expression or configuration: center-light, glasses/no glasses, happy, normal, left-light, right-light, sad, sleepy, surprised and wink. Each image was digitized and presented by a $61 \times 80$ pixel array whose gray levels ranged between 0 and 255. Some of face images coming under 10 different conditions except wink Samples of the Yale databases are shown in Fig. 7(b). The training and testing set are se-

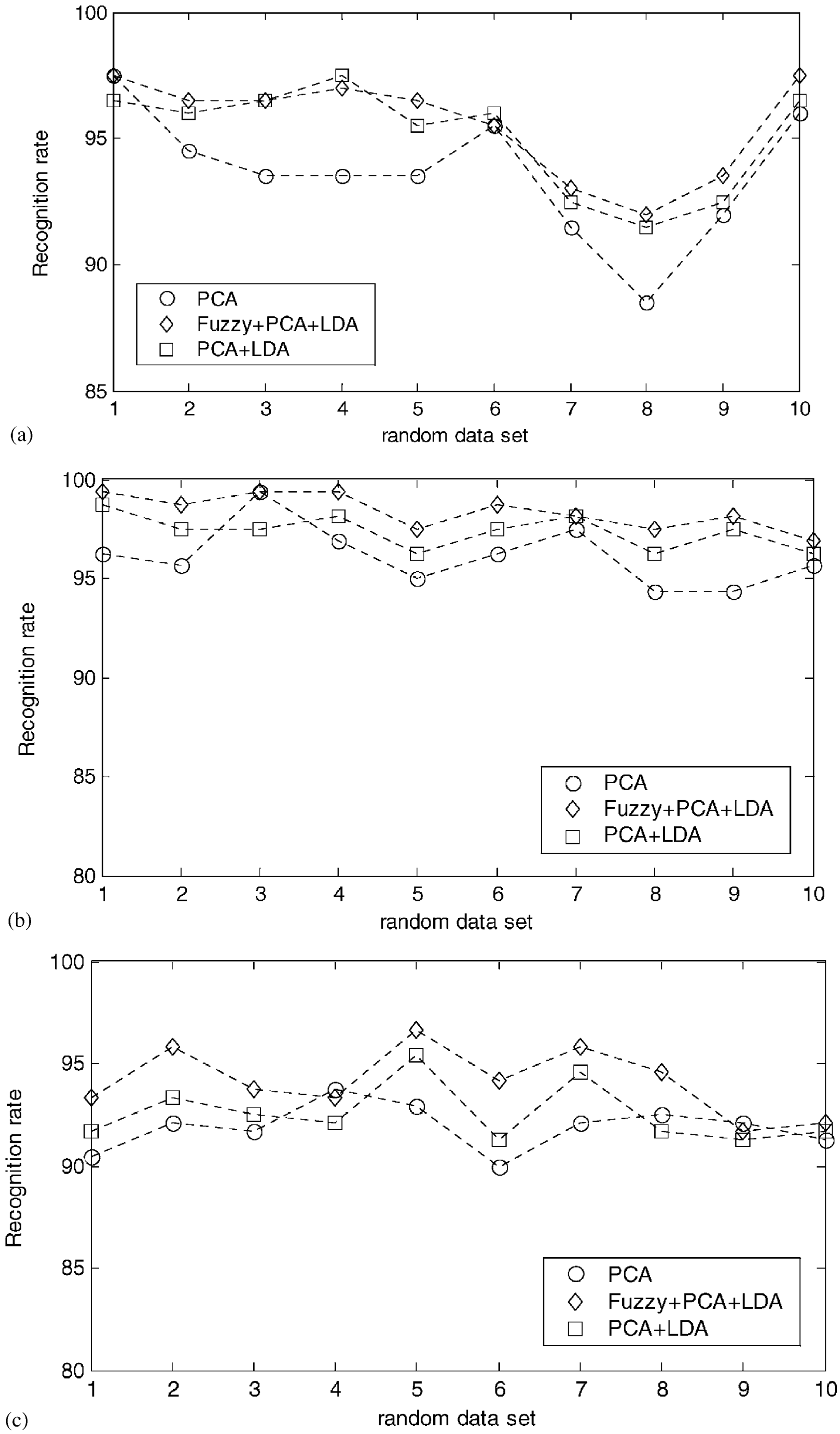


Fig. 9. Recognition rates for ORL database: (a) case 1, (b) case 2, (c) case 3.

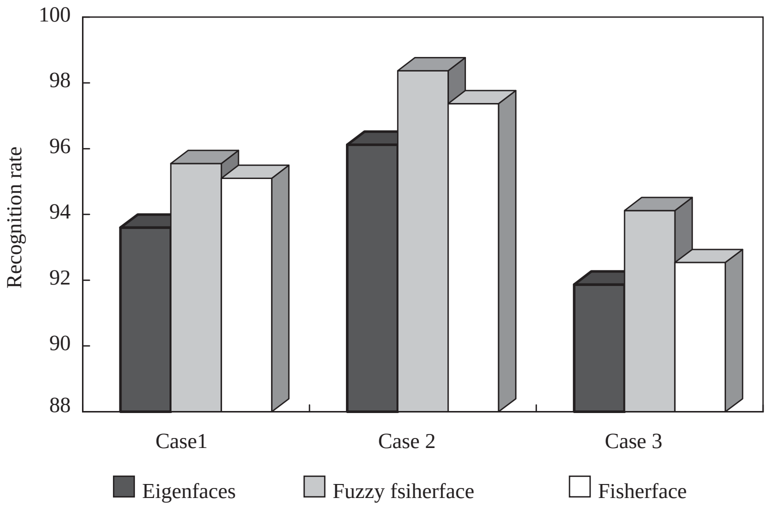


Fig. 10. Comparison of recognition rates for the ORL database.

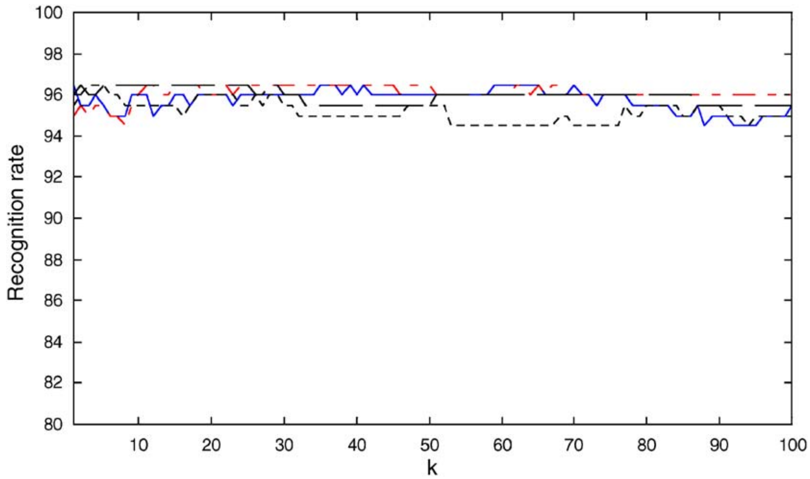


Fig. 11. Recognition rates treated as a function of “$k$” in four random test data sets (case 1).

lected randomly. This split procedure has been repeated for the 10 times in each case. Here, we determined 40 eigenvectors representing the best performance in these ten experiments. The number of discriminant vectors was set up to be 14. Fig. 8(b) shows some of the fisherface image obtained by the proposed method. The experiment results for face recognition are shown in Fig. 12. The mean recognition rates for three cases are shown in Fig. 13. Table 3 lists the comparison of mean and standard deviation for recognition rates. Again, as summarized in these figures and Table 3, the proposed method outperformed other clas-

Table 3
Comparison of mean and standard deviation of recognition rates (Yale)

| | Eigenface (PCA) (%) | Fisherface (PCA + LDA) (%) | Fuzzy fisherface (Fuzzy + PCA+ LDA (%) |
|---|---|---|---|
| Case 1 | $79.6 \pm 3.44$ | $91.86 \pm 3.57$ | $94.8 \pm 3.04$ |
| Case 2 | $76.66 \pm 4.16$ | $91.67 \pm 5.15$ | $95 \pm 2.48$ |
| Case 3 | $72.22 \pm 3.59$ | $87.44 \pm 3.14$ | $89.33 \pm 2.41$ |

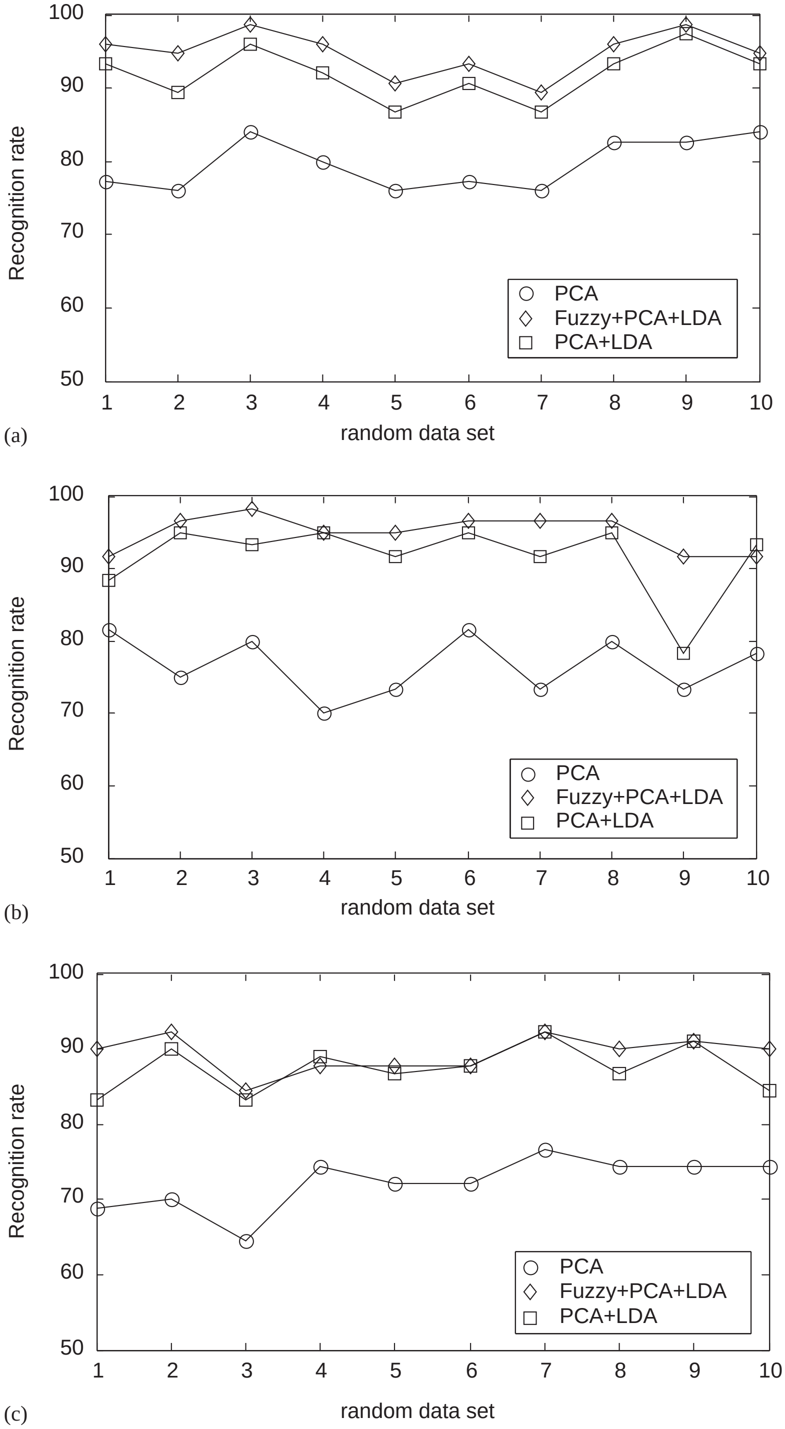


Fig. 12. Recognition rates for the Yale database: (a) case 1, (b) case 2, (c) case 3.

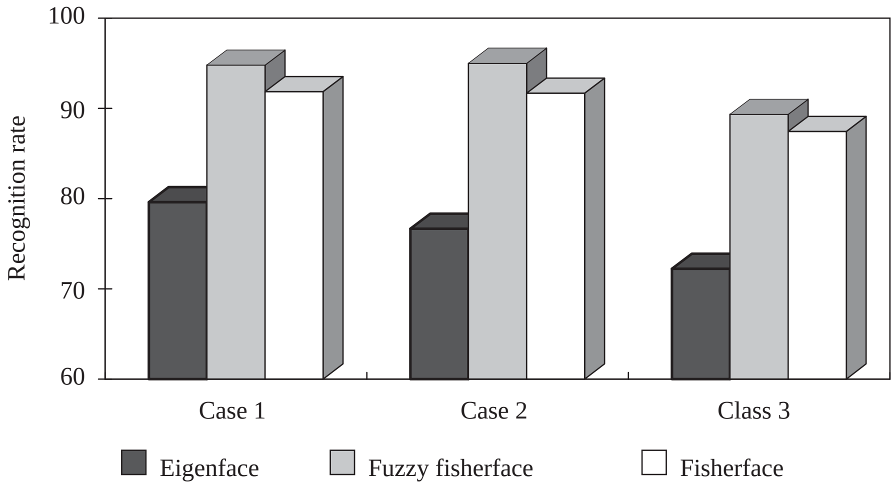


Fig. 13. Comparison of recognition rates for the Yale database.

Table 4
Comparison of mean and standard deviation of the recognition rates (CNU)

| | Eigenface (PCA) (%) | Fisherface (PCA + LDA) (%) | Fuzzy fisherface (Fuzzy + PCA+ LDA) (%) |
|---|---|---|---|
| Case 1 | $77 \pm 3.91$ | $94.8 \pm 3.29$ | $96.8 \pm 1.68$ |
| Case 2 | $75 \pm 5.13$ | $95.75 \pm 2.37$ | $97.5 \pm 2.35$ |
| Case 3 | $74.83 \pm 3.28$ | $91.67 \pm 4.0$ | $95.17 \pm 2.88$ |

sification techniques. Since PCA retains unwanted variations due to lighting and facial expression, the recognitions show a poor performance. In contrast, we observe that the proposed method can be useful in large illumination variation.

### *4.3. CNU (Chungbuk National University) face databases*

The CNU database contains 100 face images from 10 individuals in different states. The total number of images for each person is equal to 10. They vary in face pose and light variation. The size of original image is $640 \times 480$. Each image was resized by a $112 \times 92$ pixel array whose gray levels ranged between 0 and 255. Samples of the CNU databases are shown in Fig. 7(c). The training and testing set is selected, by randomly choosing three cases as already experimented with the ORL and Yale databases. This procedure has been repeated for 10 times in each case. Here, we determined 40 eigenvectors representing the best performance in the 10 times experiments. Here, the number of discriminant vectors is 9. Fig. 8(c) shows some of the fisherface image obtained by the proposed method. The results are reported in the same format as before.

Fig. 14 shows the recognition rates for three cases, respectively. Fig. 15 shows the mean recognition rates for three cases. Again, the findings are consistent with those obtained so far: the fuzzy fisherface approach outperformed the two other methods used in this classification (Table 4).

## 5. Concluding remarks

We have proposed a generalized version of the fisherface method for face recognition by including refined information about class membership of the binary labeled faces (patterns). This in turn allowed us to compute fuzzy within and in-between class scatter matrices forming the core portion of the original fisherface classifier. By doing this we were able to reduce sensitivity of the method to substantial variations between face images caused by varying illumination, viewing conditions, and facial expression. Experimental results showed a consistently better classification rates in comparison to other "standard" methods such as eigenface and fisherface when applied to ORL, Yale, and CNU face databases. In particular, it is worth stressing that the method developed in the setting of fuzzy sets revealed more robust characteristics as far as the uncertainty occurring due to large variation including illumination and facial expression (Yale and CNU) is concerned. The reason why the presented method yields a better performance can be attributed to the fact that fuzzy sets can efficiently manage the vagueness and ambiguity of the face images being degraded by poor illumination component.

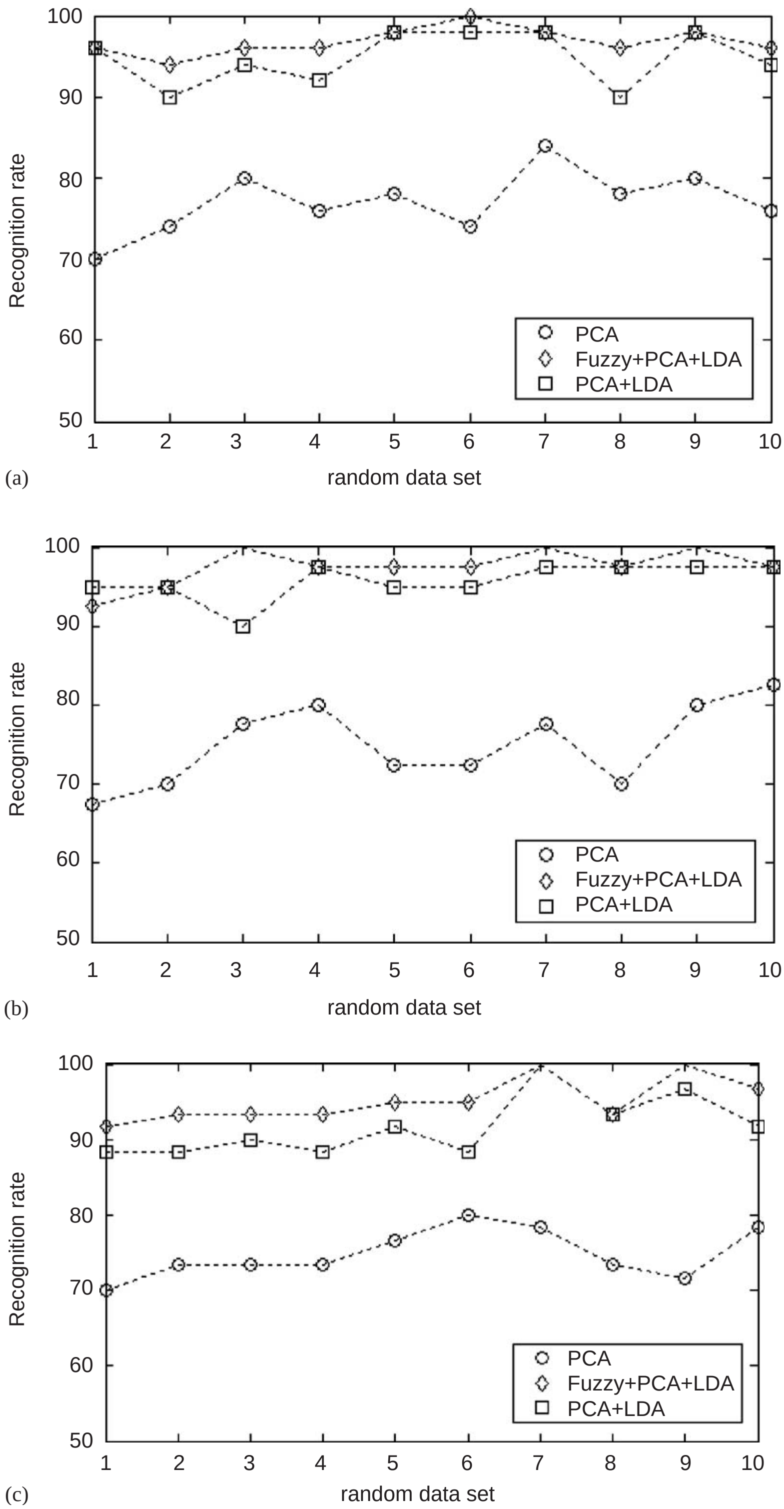


Fig. 14. Recognition rates for the CNU database: (a) case 1, (b) case 2, (c) case 3.

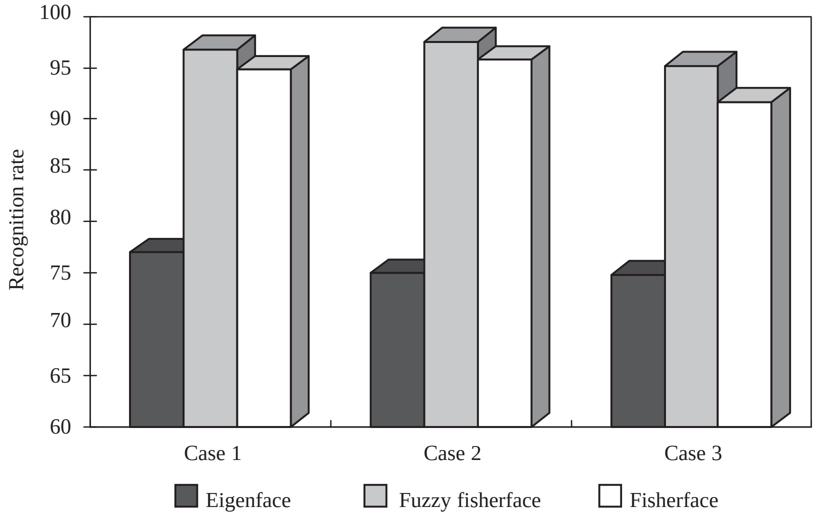


Fig. 15. Comparison of recognition rates for the CNU database.

## Acknowledgements

This work was supported by the Post-doctoral Fellowship Program of Korea Science and Engineering Foundation (KOSEF). Support from the Canada Research Chair (CRC) Program (W. Pedrycz) and Natural Sciences and Engineering Research Council (NSERC) is gratefully acknowledged.

## References

[1] M. Turk, A. Pentland, Face recognition using eigenfaces, Proceeding of IEEE Conference on Computer Vision and Pattern Recognition, Maui, HI, USA, June 1991, pp. 586–591.
[2] P.N. Belhumeur, J.P. Hespanha, D.J. Kriegman, Eigenfaces vs. fisherfaces: recognition using class specific linear projection, IEEE Trans. Pattern Anal. Mach. Intell. 19 (7) (1997) 711–720.
[3] H.C. Kim, D. Kim, S.Y. Bang, Face recognition using the mixture-of-eigenfaces method, Pattern Recognition Lett. 23 (2002) 1549–1558.
[4] A. Pujol, J. Vitria, F. Lumbreras, J.J. Villanueva, Topological principal component analysis for face encoding and recognition, Pattern Recognition Lett. 22 (2001) 769–776.
[5] K.I. Kim, K. Jung, H.J. Kim, Face recognition using kernel principal component analysis, IEEE Signal Process. Lett. 9 (2) (2002) 40–42.
[6] B. Li, Y. Liu, When eigenfaces are combined with wavelets, Knowledge Systems 15 (2002) 343–347.
[7] C. Liu, H. Wechsler, Gabor feature based classification using the enhanced fisher linear discriminant model for face recognition, IEEE Trans. Image Process. 11 (4) (2002) 467–476.
[8] H. Yu, J. Yang, A direct LDA algorithm for high-dimensional data-with application to face recognition, Pattern Recognition 34 (2001) 2067–2070.
[9] Z. Jin, J.Y. Yang, Z.S. Hu, Z. Lou, Face recognition based on the uncorrelated discriminant transformation, Pattern Recognition 34 (2001) 1405–1416.
[10] D.L. Swets, J. Weng, Using discriminant eigenfeatures for image retrieval, IEEE Trans. Pattern Anal. Mach. Intell. 18 (8) (1996) 831–836.
[11] M.J. Er, S. Wu, H.L. Toh, Face recognition with radial basis function (RBF) neural networks, IEEE Trans. Neural Networks 13 (3) (2002) 697–710.
[12] S. Lawrence, C.L. Giles, A.C. Tsoi, A.D. Back, Face recognition: a convolutional neural-network approach, IEEE Trans. Neural Networks 8 (1) (1997) 98–113.
[13] V. Chatzis, A.G. Bors, L. Pitas, Multimodal decision-level fusion for person authentication, IEEE Trans. Syst. Man Cybernet.-Part A 29 (6) (1999) 674–680.
[14] C. Liu, H. Wechsler, Evolutionary pursuit and its application to face recognition, IEEE Trans. Pattern Anal. Mach. Intell. 22 (6) (2000) 570–582.
[15] J.C. Bezdek, J. Keller, R. Krishnapuram, Fuzzy Models and Algorithms for Pattern Recognition and Image Processing, Kluwer Academic Publishers, Dordrecht, 1999.
[16] J.M. Keller, M.R. Gray, J.A. Givens, A fuzzy $k$-nearest neighbor algorithm, IEEE Trans. Syst. Man Cybernet. 15 (4) (1985) 580–585.
[17] ORL face database, http://www.uk.research.att.com/facedatabase.html.
[18] Yale face database, http://cvc.yale.edu/projects/yalefaces/yalefaces.html.
[19] L.A. Zadeh, Fuzzy sets, Info. Control 8 (1965) 338–353.

**About the Author**—WITOLD PEDRYCZ is a Professor and Canada Research Chair in Computational Intelligence in the Department of Electrical and Computer Engineering, University of Alberta, Edmonton, Canada. He is also a professor in the Systems Research Institute of the Polish Academy of Sciences, Warsaw, Poland. He is actively pursuing research in Computational Intelligence, fuzzy modeling, knowledge discovery and data mining, fuzzy control including fuzzy controllers, pattern recognition, knowledge-based neural networks, relational computation, and Software Engineering. He has published numerous papers in this area. He is also author of 9 research monographs covering various aspects of Computational Intelligence and Software Engineering. Dr. Pedrycz is an IEEE Fellow. He currently serves as an Associate Editor of IEEE Transactions on Systems Man and Cybernetics and IEEE Transactions on Fuzzy Systems. He is an editor-in-chief of INFORMATION SCIENCES.

**About the Author**—KEUN-CHANG KWAK received B.Sc., M.Sc., and Ph.D. degrees in the Department of Electrical Engineering from Chungbuk National University, Cheongju, South Korea, in 1996, 1998, and 2002, respectively. During 2002–2003, he worked as a researcher in the Brain Korea 21 Project Group, Chungbuk National University. He is currently a Postdoctoral Fellow in the Department of Electrical and Computer Engineering, University of Alberta, Edmonton, AB, Canada. His research interests include biometrics, computational intelligence, pattern recognition, and intelligent control.