Constructing PCA Baseline Algorithms to Reevaluate ICA-Based Face-Recognition Performance

Jian Yang, David Zhang, and Jing-Yu Yang

***Abstract*—The literature on independent component analysis (ICA)-based face recognition generally evaluates its performance using standard principal component analysis (PCA) within two architectures, ICA Architecture I and ICA Architecture II. In this correspondence, we analyze these two ICA architectures and find that ICA Architecture I involves a vertically centered PCA process (PCA I), while ICA Architecture II involves a whitened horizontally centered PCA process (PCA II). Thus, it makes sense to use these two PCA versions as baselines to reevaluate the performance of ICA-based face-recognition systems. Experiments on the FERET, AR, and AT&T face-image databases showed no significant differences between ICA Architecture I (II) and PCA I (II), although ICA Architecture I (or II) may, in some cases, significantly outperform standard PCA. It can be concluded that the performance of ICA strongly depends on the PCA process that it involves. Pure ICA projection has only a trivial effect on performance in face recognition.**

***Index Terms*—Face recognition, feature extraction, image representation, independent component analysis (ICA), principal component analysis (PCA).**

I. Introduction

Face recognition has attracted significant attention in the past decades because of its potential applications in biometrics, information security, and law enforcement. Many methods have been suggested to recognize faces [1]. Perhaps the simplest method is principal component analysis (PCA). PCA was first used to represent images of human faces by Sirovich and Kirby in 1987 [2], [3] and was, subsequently, applied to face recognition by Turk and Pentland [4], [5] who presented the well-known Eigenfaces method in 1991. Since then, PCA has been widely investigated and has become one of the most popular face-recognition approaches [6]–[12].

Recently, a method closely related to PCA, independent component analysis (ICA) [13], has received wide attention. ICA can be viewed as a generalization of PCA, since it is concerned not only with second-order dependencies between variables but also with high-order dependencies between them. PCA makes the data uncorrelated while ICA makes the data as independent as possible. Generally, there are two arguments for using ICA for face representation and recognition. First, the high-order relationships among image pixels may contain information that is important in recognition tasks. Second, ICA seeks to find the directions such that the projections of the data into those directions have maximally "non-Gaussian" distributions. These projections may be interesting and useful in classification tasks [13], [31].

Manuscript received December 22, 2005; revised July 6, 2006. The work was supported in part by the UGC/CRC fund from the HKSAR Government, in part by the central fund from the Hong Kong Polytechnic University, and in part by the National Science Foundation of China under Grants 60503026, 60332010, 60472060, 60473039, and 60632050. Portions of the research in this paper use the FERET database of facial images collected under the FERET program sponsored by the DOD Counterdrug Technology Development Program Office. This paper was recommended by Associate Editor M. Pantie.

J. Yang is with the Biometrics Centre, Department of Computing, Hong Kong Polytechnic University, Kowloon, Hong Kong, and also with the Department of Computer Science, Nanjing University of Science and Technology, Nanjing 210094, China (e-mail: csjyang@comp.polyu.edu.hk).

D. Zhang is with the Biometrics Centre, Department of Computing, Hong Kong Polytechnic University, Kowloon, Hong Kong (e-mail: csdzhang@comp.polyu.edu.hk).

J.-Y. Yang is with the Department of Computer Science, Nanjing University of Science and Technology, Nanjing 210094, China (e-mail: yangjy@mail.njust.edu.cn).

Digital Object Identifier 10.1109/TSMCB.2007.891541

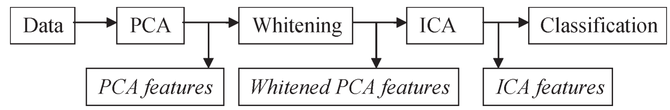

Fig. 1. Illustration of the ICA process for feature extraction and classification.

Bartlett *et al.* [14], [15] were among the first to apply ICA to face representation and recognition. They used the Infomax algorithm [27], [28] to implement ICA and suggested two ICA architectures (i.e., ICA Architectures I and II) for face representation. Both architectures were evaluated on a subset of the FERET face database and were found to be effective for face recognition [15]. Yuen and Lai [16], [17] adopted the fixed-point algorithm [29] to obtain the independent components (ICs) and used a householder transform to gain the least square solution of a face image for representation. Liu and Wechsler [18]–[20] used an ICA algorithm given by Comon [26] to perform ICA and assessed its performance for face identification. All of these researchers claimed that ICA outperforms PCA in face recognition. Other researchers, however, reported differently. Baek *et al.* [21] reported that PCA outperforms ICA while Moghaddam [22] and Jin and Davoine [23] reported no significant performance difference between the two methods. Socolinsky and Selinger [24] reported that ICA outperforms PCA on visible images but PCA outperforms ICA on infrared images.

Recently, Draper *et al.* [25] tried to account for these apparently contradictory results. They retested ICA and PCA on the FERET face database with 1196 individuals and made a comprehensive comparison of the performances of the two methods and found that the relative performance of ICA and PCA mainly depends on the ICA architecture and the distance metric. Their experimental results showed that: 1) ICA Architecture II with the cosine distance significantly outperforms PCA with L1 (city block), L2 (Euclidean), and cosine distance metrics. This is consistent with Bartlett and Liu's results; 2) PCA with the L1 distance outperforms ICA Architecture I. This is in favor of Baek's results; and 3) ICA Architecture II with L2 still significantly outperforms PCA with L2, although the degree of significance is not as great as in the ICA Architecture II with cosine over PCA. Moreover, it should be noted that this last result is still inconsistent with Moghaddam and Jin's results.

An interesting byproduct of comparative research into ICA and PCA is the finding that different versions of ICA algorithms seem to perform similarly in face-recognition tasks. Moghaddam [22] showed that the basis images derived from Hyvärinen's fixed-point algorithm is very similar to those from Cardoso's JADE algorithm [32]. Draper *et al.* [25] showed that the performance difference between Infomax algorithm [27] and FastICA [29], [30] is insignificant.

The previous researchers [14]–[25] commonly use standard PCA as the baseline algorithm to evaluate ICA-based face-recognition systems. This, however, begs the question as to whether standard PCA is a good choice for evaluating ICA. The ICA process, as shown in Fig. 1, involves not only a PCA process but also a whitening step. After the whitening step, we get the whitened PCA features of data. How is the performance of these whitened PCA features in contrast to standard PCA features and ICA features? This issue has not been addressed yet. The function of the whitening step, particularly its potential effect on the recognition performance, is still unclear. In the case where the performance of ICA is significantly different from that of PCA, it is critically important to determine what causes this difference, whether it is the whitening process or the subsequent pure ICA projection.

1083-4419/$25.00 © 2007 IEEE

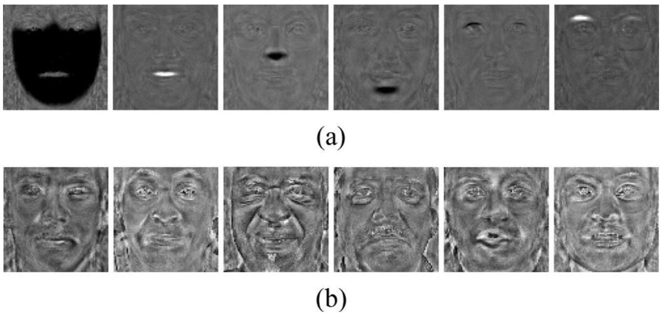

Fig. 2. Basis images corresponding to ICA Architecture I and ICA Architecture II. (a) Basis images corresponding to ICA Architecture I. (b) Basis images corresponding to ICA Architecture II.

If the whitened PCA features can perform as well as ICA features, it is certainly unnecessary to use a computationally expensive ICA projection for further processing. It seems that standard PCA is not as an appropriate baseline algorithm as "PCA + Whitening" (whitened PCA) for evaluating ICA.

In this correspondence, we analyze two ICA-based image-representation architectures and find that ICA Architecture I involves a vertically centered PCA process (PCA I), while ICA Architecture II involves a whitened horizontally centered PCA process (PCA II). Therefore, it is natural to use these two PCA versions as baseline algorithms to reevaluate the performance of ICA-based face-recognition systems.

It should be stated that in this correspondence, our goal is not to find whether ICA or PCA is better but to investigate first what role the PCA whitening step and centering mode play in the ICA-based face-recognition system and second what effect the pure ICA projection has on the performance of face recognition. We also investigate how the performances of two ICA architectures depend on their related PCA versions. It is hoped that this investigation may explain why ICA outperforms PCA in some cases and why not in other cases.

The remainder of correspondence is organized as follows. Section II describes two ICA-based image-representation architectures and their corresponding PCA baseline algorithms. In Section III, we apply these two architectures and baseline algorithms to three face-image databases and compare their performance. Section IV offers some conclusions and outlines future work.

II. Two ICA-Based Image-Representation Architectures and Their Corresponding PCA-Baseline Algorithms

Regardless of the algorithm that is used, ICA for face recognition will generally operate within one of two different architectures, Architecture I or Architecture II. In Architecture I, the observed face images are viewed as a linear mixture of a set of statistically independent basis images. ICA is used to recover the set of statistically independent basis images. To represent the image for use in recognition, ICA makes use of the reconstruction coefficients of a face image that are derived from these basis images. These coefficients for coding each image may be mutually dependent, but the basis images are mutually independent. In Architecture II, however, ICA is used to find a set of statistically independent coefficients to represent an image and the resulting basis images may be mutually dependent. Fig. 2 shows some basis images corresponding to ICA Architecture I and ICA Architecture II. It is shown that ICA Architecture I provides a more localized representation for faces, while ICA Architecture II provides a more holistic representation.

In the following sections, we will analyze these two architectures and give their corresponding PCA-baseline algorithms.

A. ICA Architecture I and Its Baseline Algorithm PCA I

Given a set of M training samples (image column vectors) $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_M$ in $\mathbb{R}^N$, we form the image column data matrix $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_M)$ and its transpose (image row data matrix) $\mathbf{Y} = \mathbf{X}^{\mathrm{T}}$.

In Architecture I, the face images are viewed as random variables and the pixel values provide observations of these variables. This means that ICA is performed on the image row data matrix $\mathbf{Y}$. Rewriting $\mathbf{Y} = (\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_N)$, its column vectors $\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_N$ are used as observation vectors to estimate the unmixing matrix of the ICA model.

1) Centering Data: Let us center the data $\mathbf{Y}$ in an observation space $\mathbb{R}^M$ and obtain its mean vector $\mu_{\mathrm{I}} = E\{\mathbf{y}\} = (1/N)\sum_{j=1}^{N} \mathbf{y}_j$. Denote $\mu_{\mathrm{I}} = (\mu_1, \mu_2, \ldots, \mu_M)^{\mathrm{T}}$. Actually, $\mu_j = E\{\mathbf{x}_j\}$, that is, the mean of all pixel values of the image $\mathbf{x}_j$. Subtracting the mean vector μ_{I} from each observation vector, that is, $\overline{\mathbf{y}}_j = (\mathbf{y}_j - \mu_{\mathrm{I}})$, we get the centered image row data matrix $\mathbf{Y}_h = (\overline{\mathbf{y}}_1, \overline{\mathbf{y}}_2, \ldots, \overline{\mathbf{y}}_N)$.

Let us define $\mathbf{X}_\nu = \mathbf{Y}_h^{\mathrm{T}} = (\widetilde{\mathbf{x}}_1, \widetilde{\mathbf{x}}_2, \cdots, \widetilde{\mathbf{x}}_M)$. $\mathbf{X}_\nu$ is called the vertically centered image column data matrix. Each column $\widetilde{\mathbf{x}}_j$ is a zero-mean image, that is, the original image from whose elements the mean of all pixel values have been removed.

2) Sphering Data Using PCA: We will sphere the data using PCA based on the centered observation vectors $\overline{\mathbf{y}}_1, \overline{\mathbf{y}}_2, \cdots, \overline{\mathbf{y}}_N$. The covariance matrix is given by

$$\Sigma_{\mathrm{I}} = \frac{1}{N}\sum_{j=1}^{N} \overline{\mathbf{y}}_j \overline{\mathbf{y}}_j^{\mathrm{T}} = \frac{1}{N}\mathbf{Y}_h \mathbf{Y}_h^{\mathrm{T}}. \tag{1}$$

Let $\mathbf{G}_\nu = \mathbf{Y}_h \mathbf{Y}_h^{\mathrm{T}}$. Calculate the orthonormal eigenvectors $\gamma_1, \gamma_2, \ldots, \gamma_m$ of $\mathbf{G}_\nu$ corresponding to the m largest positive eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_m$. Then, the m largest positive eigenvalues of Σ_{I} are $\lambda_1/N, \lambda_2/N, \ldots, \lambda_m/N$, and the associated orthonormal eigenvectors are $\gamma_1, \gamma_2, \ldots,$ and γ_m.

Letting $\mathbf{V} = (\gamma_1, \gamma_2, \ldots, \gamma_m)$ and $\Lambda = \mathrm{diag}(\lambda_1, \lambda_2, \ldots, \lambda_m)$, we obtain the whitening matrix $\mathbf{P} = \mathbf{V}[(1/N)\Lambda]^{-1/2} = \sqrt{N}\ \mathbf{V}\Lambda^{-1/2}$, such that

$$\mathbf{P}^{\mathrm{T}}\Sigma_{\mathrm{I}}\mathbf{P} = \mathbf{I}. \tag{2}$$

The data matrix $\mathbf{Y}_h = (\overline{\mathbf{y}}_1, \overline{\mathbf{y}}_2, \ldots, \overline{\mathbf{y}}_N)$ can be whitened using the transform

$$\mathbf{R} = \mathbf{P}^{\mathrm{T}}\mathbf{Y}_h. \tag{3}$$

Let us construct the following covariance matrix based on the vertically centered image column data matrix $\mathbf{X}_\nu = \mathbf{Y}_h^{\mathrm{T}} = (\tilde{\mathbf{x}}_1, \tilde{\mathbf{x}}_2, \ldots, \tilde{\mathbf{x}}_M)$:

$$\Sigma_\nu = \frac{1}{M}\mathbf{X}_\nu \mathbf{X}_\nu^{\mathrm{T}} = \frac{1}{M}\mathbf{Y}_h^{\mathrm{T}}\mathbf{Y}_h. \tag{4}$$

We may then draw the following conclusion.

Proposition 1: The row vectors of $\mathbf{R} = \mathbf{P}^{\mathrm{T}}\mathbf{Y}_h$ are orthogonal eigenvectors of Σ_ν corresponding to the m largest positive eigenvalues $\lambda_1/M, \lambda_2/M, \ldots, \lambda_m/M$ [36].

3) ICA Processing: We perform ICA on $\mathbf{R}$, producing the matrix $\mathbf{U}_\mathrm{I}$ with m independent basis images in its rows, i.e.,

$$\mathbf{U}_\mathrm{I} = \mathbf{W}_\mathrm{I}\mathbf{R} \tag{5}$$

where $\mathbf{W}_\mathrm{I}$ is the unmixing matrix generated by a given ICA algorithm based on the input data $\mathbf{R}$.

Taking note that the unmixing matrix must be invertible, from (5), it follows that

$$\mathbf{R} = \mathbf{W}_\mathrm{I}^{-1}\mathbf{U}_\mathrm{I}. \tag{6}$$

After vertically centering and projecting a given image $\mathbf{x}$ in a column vector onto the row vectors of $\mathbf{R}$, we have $\mathbf{z} = \mathbf{R}\mathbf{x}$. Since the row vectors of $\mathbf{R}$ are principal eigenvectors of Σ_ν (from Proposition 1), a minimum-square-error-based representation of $\mathbf{x}$ is

$$\hat{\mathbf{x}} = \mathbf{R}^\mathrm{T}\mathbf{z}. \tag{7}$$

Substituting (6) into (7), we have

$$\hat{\mathbf{x}} = \mathbf{U}_\mathrm{I}^\mathrm{T}\left(\mathbf{W}_\mathrm{I}^{-1}\right)^\mathrm{T}\mathbf{z}. \tag{8}$$

Therefore, in the space spanned by the row vectors of $\mathbf{U}_\mathrm{I}$, i.e., a set of m statistically independent basis images, the vector of representation coefficients of image $\mathbf{x}$ is given by

$$\mathbf{s} = \left(\mathbf{W}_\mathrm{I}^{-1}\right)^\mathrm{T}\mathbf{z} = \left(\mathbf{W}_\mathrm{I}^{-1}\right)^\mathrm{T}\mathbf{R}\mathbf{x}. \tag{9}$$

This transform can be decomposed into the following two transforms:

$$\mathbf{z} = \mathbf{R}\mathbf{x} \tag{10}$$

$$\mathbf{s} = \left(\mathbf{W}_\mathrm{I}^{-1}\right)^\mathrm{T}\mathbf{z}. \tag{11}$$

Since the row vectors of $\mathbf{R}$ are principal eigenvectors of Σ_ν (from Proposition 1), the transform in (10) is a special PCA transform in which the data is centered vertically. This transform is thus called vertically centered PCA.

Because the ICA Architecture I involves a vertically centered PCA process, it makes sense to evaluate it using this specific PCA as the baseline algorithm. In this correspondence, the vertically centered PCA algorithm is referred to as PCA Baseline Algorithm I (PCA I).Incidentally, if the unmixing matrix $\mathbf{W}_\mathrm{I}$ is orthogonal,[1] the equation $\mathbf{W}_\mathrm{I}^{-1} = \mathbf{W}_\mathrm{I}^\mathrm{T}$ holds. Then, (9) becomes

$$\mathbf{s} = \mathbf{W}_\mathrm{I}\mathbf{R}\mathbf{x} = \mathbf{U}_\mathrm{I}\mathbf{x}. \tag{12}$$

In this case, we can see that $\mathbf{s}$ is obtained by directly projecting the sample $\mathbf{x}$ onto m independent basis images [36].

B. ICA Architecture II and Its Baseline Algorithm PCA II

Given a set of M training samples (image column vectors) $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_M$ in $\mathbb{R}^N$, we form the image column data matrix $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_M)$.

The goal of ICA Architecture II is to find statistically independent coefficients for the input image data. In this architecture, the face images are viewed as observations, and the pixel values are random variables. ICA is performed directly on the image column data matrix $\mathbf{X}$. In other words, $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_M$ are used as observation vectors to estimate the unmixing matrix of the ICA model.

[1]Ideally, the unmixing matrix should be orthogonal, but in practice, it may be nonorthogonal. Except for FastICA, many ICA algorithms, such as Infomax algorithm or Comon's algorithm, result in a nonorthogonal unmixing matrix.

1) Centering Data: Let us center the data in the observation vector space $\mathbb{R}^N$. The mean vector $\mu_\mathrm{II} = E\{\mathbf{x}\} = (1/M)\sum_{j=1}^M \mathbf{x}_j$. Every observation vector is subtracted by the mean vector μ_II, i.e., $\mathbf{x}_j \leftarrow (\mathbf{x}_j - \mu_\mathrm{II})$, then we get the centered image column data matrix $\mathbf{X}_h = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_M)$. $\mathbf{X}_h$ is called the horizontally centered image column data matrix, in contrast with the vertically centered image column data matrix $\mathbf{X}_\nu$, as described in Section II-A.

2) Sphering Data Using PCA: Based on the horizontally centered image vectors $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_M$, we can construct the following covariance matrix:

$$\Sigma_\mathrm{II} = \frac{1}{M}\sum_{j=1}^{M}\mathbf{x}_j\mathbf{x}_j^\mathrm{T} = \frac{1}{M}\mathbf{X}_h\mathbf{X}_h^\mathrm{T}. \tag{13}$$

Suppose $\beta_1, \beta_2, \ldots,$ and β_m are the orthonormal eigenvectors of Σ_II corresponding to the m largest positive eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_m$. Letting $\mathbf{P}_h = (\beta_1, \beta_2, \ldots, \beta_m)$, the standard PCA transform is

$$\mathbf{z} = \mathbf{P}_h^\mathrm{T}\mathbf{x}. \tag{14}$$

Letting $\mathbf{P}_w = \mathbf{P}_h\Lambda^{-1/2}$, where $\Lambda = \mathrm{diag}(\lambda_1, \lambda_2, \ldots, \lambda_m)$, we have $P_w^\mathrm{T}\Sigma_\mathrm{II}\mathbf{P}_w = \mathbf{I}$. Thus, $\mathbf{P}_w$ is a whitening matrix. The whitened PCA transform is

$$\mathbf{z} = \mathbf{P}_w^\mathrm{T}\mathbf{x}. \tag{15}$$

3) ICA Processing: We then perform ICA on the sphered data $\mathbf{z}_1, \mathbf{z}_2, \ldots,$ and $\mathbf{z}_M$. Suppose the resulting unmixing matrix is $\mathbf{W}_\mathrm{II}$. The whole transform matrix $\mathbf{U}_\mathrm{II}$ of ICA Architecture II is

$$\mathbf{U}_\mathrm{II} = \mathbf{W}_\mathrm{II}\mathbf{P}_w^\mathrm{T}. \tag{16}$$

For a given image $\mathbf{x}$ in a column vector, after being horizontally centered and unmixed by $\mathbf{U}_\mathrm{II}$, we have

$$\mathbf{s} = \mathbf{U}_\mathrm{II}\mathbf{x} = \mathbf{W}_\mathrm{II}\mathbf{P}_w^\mathrm{T}\mathbf{x}. \tag{17}$$

The vector $\mathbf{s}$, containing a set of independent coefficients, is used to represent the image $\mathbf{x}$ for recognition purposes.

It is obvious that the transform in (17) can be decomposed into two items: A whitened PCA transform $\mathbf{z} = \mathbf{P}_w^\mathrm{T}\mathbf{x}$ and a pure ICA projection $\mathbf{s} = \mathbf{W}_\mathrm{II}\mathbf{z}$.

Since the ICA Architecture II involves not only a standard PCA (horizontally centered PCA) but also a whitened PCA process, the two PCA processes should be taken as baseline algorithms for evaluating ICA Architecture II. In this correspondence, the whitened horizontally centered PCA algorithm is refered to as PCA Baseline Algorithm II (PCA II).

C. Two ICA Architectures: A Summary

Fig. 3 illustrates two ICA-based image-representation architectures. Each involves a different version of PCA: ICA Architecture I involves a vertically centered PCA (PCA I), whereas ICA Architecture II involves a whitened horizontally centered PCA (PCA II). Standard PCA removes the mean image of all training samples, while PCA I removes the mean of each image. PCA II is a whitened version of standard PCA. It can normalize the variances of coefficients as well as being able to render these coefficients uncorrelated. To assess the performance of two ICA architectures, it is necessary to compare them with the two different versions of PCA as well as with the standard PCA. In other words, PCA I, PCA II, and standard PCA should all be used as baseline algorithms to evaluate the ICA.

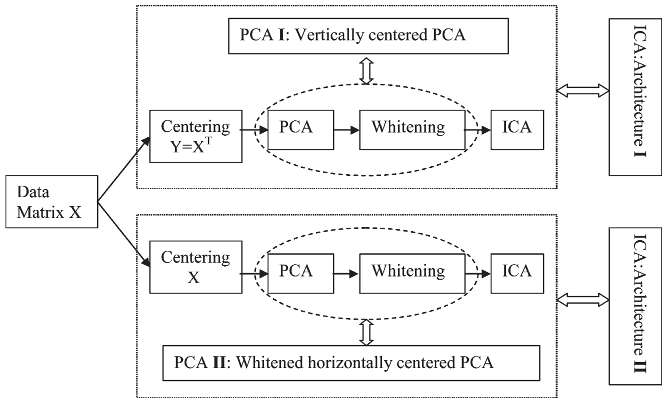

Fig. 3. Illustration of two ICA-based image-representation architectures.

Fig. 4. Sample images of one person in the FERET database.

III. Experiments and Analysis

In this section, we evaluate two ICA architectures using PCA I, PCA II, and standard PCA as baseline algorithms on three face-image databases: the FERET, AR, and AT&T databases.

A. *Experiment Using the FERET Face Database*

The FERET face database [34], [35] has become a standard database for testing and evaluating state-of-the-art face-recognition algorithms. The basic gallery of the FERET 1996 standard subset contains 1196 face images of 1196 subjects (one image per subject). There are four sets of probe images used for testing, where the fafb probe set contains 1195 images taken at the same time as the gallery images but with different facial expressions. The fafc probe set contains 194 images taken under different lighting conditions. The duplicate I probe set contains 722 images taken anywhere between 1 min and 1031 days after their respective gallery matches. Finally, the duplicate II probe set is a subset of the duplicate I set, containing 234 images taken at least 18 months after their gallery entries. In this correspondence, the face portion of each original image is automatically cropped based on the location of eyes and mouth (i.e., the known coordinates of two eyes and the mouth in the images) and resized to an image of 80×80 pixels. The resulting image is then preprocessed using a histogram-equalization algorithm. Fig. 4 shows some example images after preprocessing.

1) Parameter Selection in the ICA Model: As we know, an ICA-based image-recognition system is generally implemented in two phases. In the first phase, PCA I (or II) is used for preprocessing, and in the second phase, a pure ICA projection is performed. It is difficult, however, to determine how many principle components (PCs) are required in the PCA phase of ICA. Liu [20] has shown that the number of PCs selected has a substantial effect on the performance of ICA-based face-recognition systems.

In this correspondence, we tried to find an optimal number of PCs, that is, a number which maximizes the performance of ICA. To this end, 500 images were randomly selected from the gallery to form the training sample set, and the four probe sets were united to form a testing set. Here, FastICA [33] with a contrast function $G_1(u) = (1/4)u^4$ (which is closely related to kurtosis) was used as an example to perform parameter selection. Note that, here, there is only training sample available per class, so the feature selection mechanism based on the ratio of between-class variance to within-class variance suggested in [15] cannot be used. Here, ICA features are automatically ordered by the FastICA code [33]. A nearest neighbor classifier with cosine distance was employed. Our parameter-selection strategy was to allow the number of PCs m to vary from 20 to 300 with an interval of 20. For each m, we allowed the number of ICs d to vary from ten to m with an interval of ten. This produces an optimal d^*_{m} which maximizes the recognition rate of the ICA. Finally, we chose an optimal m^* (and the corresponding d^*_{m}) which achieves the best performance. Fig. 5 shows the best recognition rates of ICA Architectures I and II versus the variation of the PC number m. Table I gives the optimal parameters corresponding to two ICA architectures.

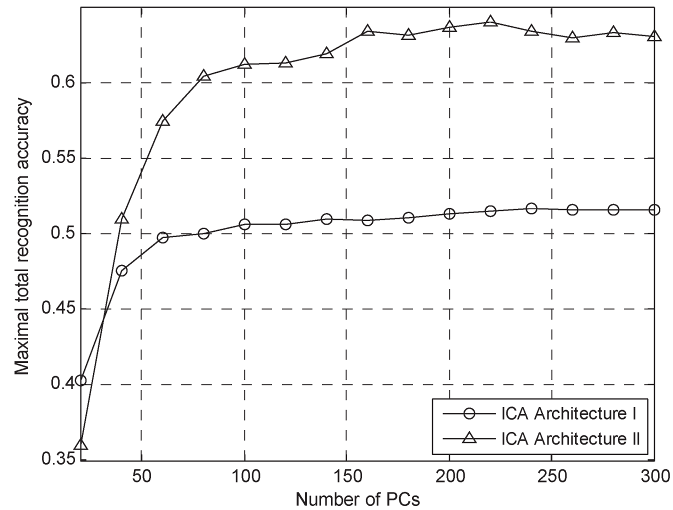

Fig. 5. Recognition rates of two ICA architectures versus the number of PCs.

TABLE I
Optimal Parameters for Two ICA Architectures on the FERET Database

Parameters	ICA Arch. I	ICA Arch. II
Number of PCs	240	220
Number of ICs	210	210

2) Performance Comparison and Analysis: In this section, we compare the performance of face-recognition systems using standard PCA, PCA I, PCA II, and ICA Architectures I and II. In order to alleviate the effect on the recognition performance caused by the choice of the training sample set, we run each system ten times. Each time, the training sample set containing 500 images is randomly selected from the gallery so that the training sample sets are different for each of the ten tests.

The projection matrix of each method is obtained by training, and for each face image, 210 features are extracted for use in recognition. Here, we calculate the unmixing matrix by using the FastICA algorithm with the contrast function $G_1(u) = (1/4)u^4$. After feature extraction, we classify using a nearest neighbor classifier with different distance metrics. Three distance metrics, the L2 distance, the L1 distance, and the cosine distance, are used in standard PCA. Only the cosine distance is used in ICA Architectures I and II, because this metric has been shown to be most effective for both of them [25]. As evaluators of ICA Architectures I and II, PCA I and PCA II also use the cosine distance. Table II lists the average recognition rate and standard deviation (std) across ten tests for each method and each probe set.

TABLE II
AVERAGE RECOGNITION RATES AND STD OF STANDARD PCA, PCA I, PCA II, AND ICA ARCHITECTURES I AND II USING DIFFERENT DISTANCE METRICS ON THE FERET DATABASE

Probe set	ICA Arch. I	PCA I	Standard PCA			PCA II	ICA Arch. II
	Cosine	Cosine	L2	L1	Cosine	Cosine	Cosine
fafb	77.63 ± 0.39	77.67± 0.32	77.27± 0.36	76.26± 0.54	76.64± 0.37	81.66± 0.59	81.36 ± 0.49
fafc	14.28 ± 1.31	14.47± 1.32	14.89± 1.27	38.57± 1.26	11.06± 0.45	64.17± 1.19	64.58 ± 1.89
Dup.I	32.37 ± 0.35	32.48± 0.32	32.03± 0.46	33.61± 0.86	33.79± 0.60	46.76± 0.86	46.13 ± 1.15
Dup. II	10.50 ± 0. 81	10.59± 0.77	10.19± 0.63	12.69± 1.57	12.69± 0.55	26.67± 1.67	26.62 ± 1.90
Total	51.76	51.83	51.49	53.67	51.64	63.98	63.66

TABLE III
RECOGNITION RATES OF INFOMAX ICA AND FASTICA WITH DIFFERENT CONTRAST FUNCTIONS

Probe set	PCA I	ICA Arch. I			ICA Arch. II			PCA II
		Infomax	FastICA with G_2	FastICA with G_3	Infomax	FastICA with G_2	FastICA with G_3	
fafb	77.67	78.07	77.72	77.65	80.47	81.01	81.33	81.66
fafc	14.47	12.11	14.37	14.22	64.75	63.49	63.86	64.17
Dup.I	32.48	33.43	32.43	32.36	46.17	45.67	46.09	46.76
Dup. II	10.59	11.38	10.59	10.46	27.36	25.94	25.82	26.67
Total	51.83	52.21	51.84	51.75	63.31	63.18	63.50	63.98

TABLE IV
AVERAGE RECOGNITION RATES AND STD OF STANDARD PCA, PCA I, PCA II, AND ICA ARCHITECTURES I AND II WITH DIFFERENT TRAINING SAMPLE SIZES ON THE AR DATABASE

# / class	ICA Arch. I		PCA I	Standard PCA	PCA II	ICA Arch. II	
	FastICA	Infomax				FastICA	Infomax
2	93.1 ± 1.2	92.5 ± 1.5	93.2 ± 1.2	73.5 ± 9.4	92.1 ± 1.7	92.1 ± 1.7	91.8 ± 1.9
4	96.2 ± 0.4	95.2 ± 0.8	95.9 ± 0.5	88.2 ± 7.1	97.6 ± 0.5	97.6 ± 0.5	97.6 ± 0.6
6	96.8 ± 0.9	96.4 ± 1.2	97.0 ± 1.0	90.7 ± 8.4	98.0 ± 0.5	97.9 ± 0.4	98.1 ± 0.7
8	98.0 ± 1.1	97.8 ± 1.2	98.1 ± 1.0	96.0 ± 1.2	98.6 ± 0.6	98.7 ± 0.6	98.6 ± 0.7

Uniting four probe sets as a total testing set, the total recognition rate of each method is also listed in this table.

There are two main points to be taken from Table II. First, ICA Architecture II with cosine distance significantly outperforms standard PCA no matter what distance metric PCA uses. Second, PCA with L1 distance is slightly better than ICA Architecture I in terms of the total recognition rate. All of the results in Table II are consistent with Draper *et al.*'s studies [25], which concluded that ICA Architecture II is better than PCA for identifying faces. However, if we reevaluate the performance of ICA using the proposed PCA-baseline algorithms, we can draw some very different conclusions. As shown in Table II, PCA II (I) can perform as well as (even slightly better than) ICA Architecture II (I). There is no significant performance difference between ICA Architecture II (I) and PCA II (I). It seems that the effect of pure ICA projection on the performance of face recognition is trivial. Given this, we can conclude that the significant performance difference between ICA Architecture II and standard PCA arises from the whitening step, rather than from pure ICA projection.

To determine whether the above conclusion depends on the choice of ICA algorithms or contrast functions adopted in FastICA, we tested two ICA architectures using an Infomax algorithm and FastICA with two different contrast functions: $G_2(u) = (1/a_1)\log\cosh(a_1 u)$, where the parameter $a_1 = 1$; $G_3(u) = (1/a_2)\exp(-a_2 u^2/2)$, where the parameter $a_2 = 1$ [30]. Table III shows the average recognition rates of two ICA architectures corresponding to each ICA algorithm. Although ICA Architecture I with an Infomax algorithm slightly outperforms PCA I, this does not make sense, since it performs much worse than ICA Architecture II. The performance difference between ICA Architecture I (II) and PCA I (II) stays insignificant, no matter what ICA algorithm or contrast function is used.

B. Experiment Using the AR Database

The AR face database [40], [41] contains face images with different facial expressions, under lighting conditions, and with a variety of occlusions. The pictures of 120 individuals (65 men and 55 women) were taken in two sessions (separated by two weeks), and each session of one person contains 13 color images. The nonoccluded 14 face images (each session with seven) of each person of these 120 individuals are selected and used in this correspondence. The face portion of each image is manually cropped and then normalized to 50×40 pixels. Please refer to [11] for some sample images.

A training sample set is formed by randomly selecting t images from each session of each individual. The remaining images are used for testing. Let t vary from one to four. Thus, the number of training samples per class k is 2, 4, 6, and 8, respectively. For each k, we perform 20 random splits and obtain into 20 different training and testing sets. The first ten training and testing sets are used for ICA parameter selection and the rest for performance evaluation. Standard PCA, PCA I, PCA II, and ICA Architectures I and II are used for face representation. Two ICA algorithms, FastICA and Infomax, are used within each architecture. Since the number of training samples per class in this correspondence is larger than one, to select the most discriminative features for each method, we adopt a feature selection mechanism based on the ratio of between-class variance to within-class variance [15]. Finally, we use a nearest neighbor classifier with

TABLE V
AVERAGE RECOGNITION RATES AND STD OF STANDARD PCA, PCA I, PCA II, AND ICA ARCHITECTURES I AND II WITH DIFFERENT TRAINING SAMPLE SIZES ON THE ORL DATABASE

# / class	ICA Arch. I		PCA I	Standard PCA	PCA II	ICA Arch. II	
	FastICA	Infomax				FastICA	Infomax
2	78.3 ± 3.2	77.2 ± 3.3	78.1 ± 3.8	82.5 ± 2.1	81.0 ± 3.4	81.1 ± 3.5	80.8 ± 3.2
3	85.8 ± 3.5	84.3 ± 4.3	85.4 ± 4.0	88.4 ± 3.4	87.2 ± 3.8	87.6 ± 3.5	86.8 ± 3.8
4	90.7 ± 2.4	89.6 ± 2.6	89.8 ± 2.8	91.9 ± 2.5	91.3 ± 2.5	91.4 ± 2.3	91.3 ± 2.1
5	92.8 ± 1.8	91.7 ± 2.1	92.3 ± 1.9	94.0 ± 1.9	92.9 ± 2.4	93.0 ± 2.2	92.5 ± 2.4
6	94.8 ± 1.5	93.8 ± 1.4	94.2 ± 1.5	*95.7 ± 1.7*	93.9 ± 1.8	93.9 ± 1.8	93.6 ± 1.7

cosine distance for classification. The average recognition rate and the std across ten runs of tests of each method are shown in Table IV.

From Table IV, we can see that both ICA architectures significantly outperform standard PCA with the same cosine metric. However, the performance difference between ICA Architecture I (II) and PCA I (II) is still insignificant, irrespective of the variation in training sample size. This fact demonstrates again that the effect of pure ICA projection on the performance of face recognition is trivial. The significant performance difference between ICA Architecture I and standard PCA arises from the centering mode rather than from the pure ICA projection. We know that PCA I centers the data by removing the mean of each image (i.e., vertical centering) while standard PCA by removing the mean image of all training samples (i.e., horizontal centering). Vertical centering may be helpful to recognize faces in varying illumination, especially that caused by the variation of lighting intensity. This leads to the result that PCA I outperforms standard PCA on this database. Moreover, the significant performance difference between ICA Architecture II and standard PCA in this correspondence certainly arises from the whitening step.

C. *Experiment Using the AT&T Database*

The AT&T database (formerly the ORL database) [39] contains face images from 40 subjects, each providing ten different images. All the images were taken against a dark homogeneous background with the subjects in an upright frontal position (with tolerance for some side movement). For some subjects, the images were taken at different times, varying the lighting, facial expressions (eyes open or closed, smiling or not smiling, with or without glasses). Each image is 92×112 pixels, with 256 gray levels per pixel.

On this database, we test and compare the performance of standard PCA, PCA I, PCA II, and ICA Architectures I and II using the same procedure as we used on the AR database. The experimental results are shown in Table V. In this case, we receive very different results. Both ICA architectures perform worse than standard PCA using the same cosine metric. However, the difference between ICA Architecture I (II) and PCA I (II) is still insignificant. This once again demonstrates that the performance of ICA strongly depends on the PCA process that it involves and that pure ICA projection has only a trivial effect on performance in face recognition.

IV. CONCLUSION AND FUTURE WORK

In this correspondence, we examined two ICA-based image representation architectures and found that ICA Architecture I involves a vertically centered PCA process (PCA I), while ICA Architecture II involves a whitened horizontally centered PCA process (PCA II). We then used these two PCA versions as baseline algorithms to reevaluate the performance of ICA-based face-recognition systems. We performed experiments using the FERET, AR, and AT&T face databases and drew the following conclusions. First, there is no significant performance difference between ICA Architecture I (II) and PCA I (II), although in some cases, there is a significant difference between ICA Architecture I (II) and standard PCA. Second, the performance of ICA strongly depends on the PCA process that it involves. Pure ICA projection seems to have only a trivial effect on performance in face recognition. Third, the centering mode and the whitening step in PCA I (or II) play a central role in inducing the performance difference between ICA Architecture I (II) and standard PCA.

As demonstrated in our correspondence, the added discriminative power of the "independent features" produced by pure ICA projection is not so satisfying. Therefore, the future task is to explore effective ways to obtain more powerful independent features for face representation. Recently, Bressan and Vitria [37] proposed class-conditional ICA (CC-ICA) and, Amoto *et al.* [38] suggested IC discriminant analysis (ICDA). Both methods seek to improve the discriminative power of ICA by embedding class-supervised information into the ICA model. How to generalize these methods and make them suitable for small-sample-size face-recognition problems remains, however, an open question.

Finally, it should be mentioned that Vicente *et al.*'s paper [42] can be viewed as a sister work of ours. In their paper, the authors briefly described the connection between PCA and ICA and argued that whitened PCA may yield identical results to ICA in some cases. They also described some specific situations in which whitened PCA and ICA may perform quite differently.

ACKNOWLEDGMENT

The authors would like to thank the Associate Editor and the anonymous reviewers for their constructive advice.

REFERENCES

[1] W. Zhao, R. Chellappa, A. Rosenfeld, and P. Phillips, "Face recognition: A literature survey," Univ. Maryland, College Park, MD, Tech. Rep. CAR-TR-948, Aug. 2002. UMD CS-TR-4167R.
[2] L. Sirovich and M. Kirby, "Low-dimensional procedure for characterization of human faces," *J. Opt. Soc. Amer.*, vol. 4, no. 3, pp. 519–524, Mar. 1987.
[3] M. Kirby and L. Sirovich, "Application of the KL procedure for the characterization of human faces," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 12, no. 1, pp. 103–108, Jan. 1990.
[4] M. Turk and A. Pentland, "Eigenfaces for recognition," *J. Cogn. Neurosci.*, vol. 3, no. 1, pp. 71–86, 1991.
[5] M. Turk and A. Pentland, "Face recognition using Eigenfaces," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 1991, pp. 586–591.
[6] A. Pentland, B. Moghaddam, and T. Starner, "View-based and modular eigenspaces for face recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 1994, pp. 84–91.
[7] L. Zhao and Y. Yang, "Theoretical analysis of illumination in PCA-based vision systems," *Pattern Recognit.*, vol. 32, no. 4, pp. 547–564, Apr. 1999.
[8] X. Chen, P. J. Flynn, and K. W. Bowyer, "PCA-based face recognition in infrared imagery: Baseline and comparative studies," in *Proc. IEEE Int. Workshop Anal. and Model. Faces and Gestures*, Oct. 2003, pp. 127–134.

[9] H. C. Kim, D. Kim, S. Y. Bang, and S. Y. Lee, "Face recognition using the second-order mixture-of-eigenfaces method," *Pattern Recognit.*, vol. 37, no. 2, pp. 337–349, Feb. 2004.

[10] J. Yang and J. Y. Yang, "From image vector to matrix: A straightforward image projection technique—IMPCA vs. PCA," *Pattern Recognit.*, vol. 35, no. 9, pp. 1997–1999, Sep. 2002.

[11] J. Yang, D. Zhang, A. F. Frangi, and J.-Y. Yang, "Two-dimensional PCA: A new approach to face representation and recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 1, pp. 131–137, Jan. 2004.

[12] S. Chen and Y. Zhu, "Subpattern-based principle component analysis," *Pattern Recognit.*, vol. 37, no. 5, pp. 1081–1083, May 2004.

[13] A. Hyvärinen, J. Karhunen, and E. Oja, *Independent Component Analysis*. New York: Wiley, 2001.

[14] M. S. Bartlett and T. J. Sejnowski, "Independent components of face images: A representation for face recognition," in *Proc. 4th Annu. Jount Symp. Neural Comput.*, Pasadena, CA, May 17, 1997.

[15] M. S. Bartlett, J. R. Movellan, and T. J. Sejnowski, "Face recognition by independent component analysis," *IEEE Trans. Neural Netw.*, vol. 13, no. 6, pp. 1450–1464, Nov. 2002.

[16] P. C. Yuen and J. H. Lai, "Independent component analysis of face images," in *Proc. IEEE Int. Conf. Biological Motivated Comput. Vis.*, May 2000, pp. 545–553.

[17] P. C. Yuen and J. H. Lai, "Face representation using independent component analysis," *Pattern Recognit.*, vol. 35, no. 6, pp. 1247–1257, Jun. 2002.

[18] C. Liu and H. Wechsler, "Comparative assessment of independent component analysis for face recognition," in *Proc. 2nd Int. Conf. Audio- and Video-Based Biometric Person Authentication*, Washington, DC, Mar. 22–24, 1999, pp. 211–216.

[19] C. Liu and H. Wechsler, "Independent component analysis of Gabor features for face recognition," *IEEE Trans. Neural Netw.*, vol. 14, no. 4, pp. 919–928, Jul. 2003.

[20] C. Liu, "Enhanced independent component analysis and its application to content based face image retrieval," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 34, no. 2, pp. 1117–1127, Apr. 2004.

[21] K. Baek, B. A. Draper, J. R. Beveridge, and K. She, "PCA vs ICA: A comparison on the FERET data set," in *Proc. Joint Conf. Inf. Sci.*, Durham, NC, 2002, pp. 824–827.

[22] B. Moghaddam, "Principal manifolds and probabilistic subspaces for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 6, pp. 780–788, Jun. 2002.

[23] Z. Jin and F. Davoine, "Orthogonal ICA representation of images," in *Proc. 8th Int. Conf. Control, Autom., Robot. and Vis.*, Kunming, China, Dec. 6–9, 2004, pp. 369–374.

[24] D. Socolinsky and A. Selinger, "A comparative analysis of face recognition performance with visible and thermal infrared imagery," in *Proc. Int. Conf. Pattern Recog.*, Quebec City, QC, Canada, 2002, pp. 217–222.

[25] B. A. Draper, K. Baek, M. S. Bartlett, and J. R. Beveridge, "Recognizing faces with PCA and ICA," *Comput. Vis. Image Underst.*, vol. 91, no. 1/2, pp. 115–137, Jul. 2003.

[26] P. Comon, "Independent component analysis: A new concept?" *Signal Process.*, vol. 36, no. 3, pp. 287–314, Apr. 1994.

[27] A. J. Bell and T. J. Sejnowski, "An information-maximization approach to blind separation and blind deconvolution," *Neural Comput.*, vol. 7, no. 6, pp. 1129–1159, Nov. 1995.

[28] A. J. Bell and T. J. Sejnowski, "The 'independent components'of natural scenes are edge filters," *Vis. Res.*, vol. 37, no. 23, pp. 3327–3338, Dec. 1997.

[29] A. Hyvärinen and E. Oja, "A fast fixed-point algorithm for independent component analysis," *Neural Comput.*, vol. 9, no. 7, pp. 1483–1492, Oct. 1997.

[30] A. Hyvärinen, "Fast and robust fixed-point algorithms for independent component analysis," *IEEE Trans. Neural Netw.*, vol. 10, no. 3, pp. 626–634, May 1999.

[31] A. Hyvärinen and E. Oja, "Independent component analysis: Algorithms and applications," *Neural Netw.*, vol. 13, no. 4/5, pp. 411–430, May/Jun. 2000.

[32] J.-F. Cardoso, "High-order contrasts for independent component analysis," *Neural Comput.*, vol. 11, no. 1, pp. 157–192, Jan. 1999.

[33] A. Hyvärinen, *The FastICA Package for MATLAB*. [Online]. Available: http://www.cis.hut.fi/projects/ica/fastica/

[34] P. J. Phillips, H. Moon, S. A. Rizvi, and P. J. Rauss, "The FERET evaluation methodology for face-recognition algorithms," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 10, pp. 1090–1104, Oct. 2000.

[35] P. J. Phillips, H. Wechsler, J. Huang, and P. Rauss, "The FERET database and evaluation procedure for face recognition algorithms," *Image Vis. Comput.*, vol. 16, no. 5, pp. 295–306, Apr. 1998.

[36] J. Yang, D. Zhang, and J.-Y. Yang, "Is ICA significantly better than PCA for face recognition?" in *Proc. 10th IEEE ICCV*, vol. 1, 2005, pp. 198–203.

[37] M. Bressan and J. Vitria, "On the selection and classification of independent features," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 10, pp. 1312–1317, Oct. 2003.

[38] U. Amato, A. Antoniadis, and G. Gregoire, *Independent Component Discriminant Analysis*. [Online]. Available: http://xoomer.virgilio.it/dahama/

[39] *The AT&T Face Database*. [Online]. Available: http://www.uk.research.att.com/facedatabase.html

[40] A. M. Martinez and R. Benavente, *The AR Face Database*. [Online]. Available: http://rvl1.ecn.purdue.edu/~aleix/aleix_face_DB.html

[41] A. M. Martinez and R. Benavente, "The AR face database," CVC, Barcelona, Spain, CVC Tech. Rep. 24, Jun. 1998.

[42] M. Asunción Vicente, P. O. Hoyer, and A. Hyvarinen, "Equivalence of some common linear feature extraction techniques for appearance-based object recognition tasks," *IEEE Trans. Pattern Anal. Mach. Intell*, vol. 29, no. 5, pp. 896–900, May 2007.