FACE RECOGNITION USING FEATURE EXTRACTION BASED ON INDEPENDENT COMPONENT ANALYSIS

Nojun Kwak, Chong-Ho Choi*, and Narendra Ahuja***

*School of Electrical Eng. and Computer Science
Seoul National University, Seoul, Korea
{triplea,chchoi}@csl.snu.ac.kr

**Dept. of Electrical and Computer Engineering
University of Illinois at Urbana-Champaign
ahuja@vision.ai.uiuc.edu

ABSTRACT

We have explored a new method of feature extraction for face recognition. It is based on independent component analysis (ICA), but unlike original ICA, one of the unsupervised learning methods, it is developed to be well suited for classification problems by utilizing class information. By using ICA in solving supervised classification problems, we can obtain new features which are made as independent from each other as possible and which convey the class information faithfully. We have applied this method on Yale Face Databases and AT&T Face Databases and compared the performance with those of conventional methods such as principal component analysis (PCA), Fisher's linear discriminant (FLD), and so on. The experimental results show that for both databases the proposed method outperforms the others.

1. INTRODUCTION

In constructing features of an image, subspace methods have been successfully applied [1] – [4]. Among these, Eigenface [1] (based on PCA) and Fisherface [2] (based on FLD) methods are popular, because these methods allow efficient characterization of a low-dimensional subspace preserving the perceptual quality of a very high-dimensional raw image.

Though it is the most popular, Eigenface method [1], by its nature, is not well-fitted for classification problem since it does not make use of any output class information in deciding the principal components. The main drawback of this method is that the extracted features are not invariant under transformation. Merely scaling the attributes changes resulting features. In addition, it does not use higher order statistics and it is reported that the performance of Eigenface method is severely infected by the illumination [2].

Unlike Eigenface, Fisherface method [2] focuses on classification problems to find optimal linear discriminating functions for certain types of data whose classes have Gaussian distribution, the centers of which are well separated. Though it is very simple and powerful method for classification problem, Fisherface cannot produce more than $N_c - 1$ features, where N_c is the number of classes. And also, like Eigenface method, it only uses second order statistics in representing an image. Some researchers have proposed subspace methods using higher order statistics such as evolutionary pursuit and kernel methods for face recognition [3], [4].

Recently, in neural networks and signal processing societies, independent component analysis (ICA), which was devised for blind source separation problems, has received a great deal of attention because of its potential applications in various areas. Bell and Sejnowski [5] have developed an unsupervised learning algorithm performing ICA based on entropy maximization in a single-layer feedforward neural network. Researchers have shown that ICA using higher order statistics is more powerful for face recognition than PCA [6] [7]. Unlike PCA and FLD, ICA uses higher order statistics and some researchers have shown that It has been applied successfully for recognizing faces with changes in pose [6], and classifying facial actions [7]. But it leaves much room for improvement since it does not utilize output class information like the PCA.

In this paper, we propose an ICA-based feature extraction algorithm that utilizes the output class information in addition to having the advantages of the original ICA method. It is an extended version of [8] and well-suited for classification problems in the aspect of constructing new features that are strongly related to output class. We have applied this method for face recognition problems. The proposed algorithm greatly reduces the dimension of feature space while improving classification performance.

This paper is organized as follows. In Section 2, we briefly review ICA and propose a new feature extraction algorithm. In Section 3, experimental results are given to show the advantages of the proposed algorithm and conclusions follow in Section 4.

*This project is partially supported by the Brain Science and Engineering Program of the Korea Ministry of Science and Technology.

2. FEATURE EXTRACTION BASED ON ICA

2.1. Review of ICA

The problem setting of ICA is as follows. Assume that there is an L-dimensional zero-mean source vector $\boldsymbol{s}(t) = [s_1(t), \cdots, s_L(t)]^T$, such that the components $s_i(t)$'s are mutually independent, and an observed data vector $\boldsymbol{x}(t) = [x_1(t), \cdots, x_N(t)]^T$ is composed of linear combinations of sources $s_i(t)$ at each time point t, such that,

$$\boldsymbol{x}(t) = A\boldsymbol{s}(t) \tag{1}$$

where A is a full rank $N \times L$ matrix with $L \leq N$. The goal of ICA is to find a linear mapping W such that each component of signals $\boldsymbol{u}$

$$\boldsymbol{u}(t) = W\boldsymbol{x}(t) = WA\boldsymbol{s}(t) \tag{2}$$

is statistically independent, reproducing $\boldsymbol{s}(t)$ by some scaling and permutations. Also, $\boldsymbol{y} = [y_1, \cdots, y_N]^T$ is obtained by $y_i = g_i(u_i)$, $i = 1, \cdots, N$, for the derivation of the unsupervised learning rule. The $g_i(\cdot)$'s are assumed to be the cumulative density function of u_i's. This function, shaped like a sigmoid, plays a magnitude limiting role so that the resulting $y_i = g_i(u_i)$ is bounded.

From the view of information theory, statistical independence means that mutual information between variables are zero, and this statistical independence between u_i's can be achieved by minimizing the mutual information $I(\boldsymbol{y})$ between y_i's

$$I(\boldsymbol{y}) = \int p(\boldsymbol{y}) \log \frac{p(\boldsymbol{y})}{\prod_{i=1}^{N} p(y_i)} \, d\boldsymbol{y}. \tag{3}$$

In (3), $p(\boldsymbol{y})$ is the joint probability density function (pdf) of vector $\boldsymbol{y}$, and $p(y_i)$ is the marginal pdf of variable y_i. The joint entropy $H(\boldsymbol{y})$ can be written as

$$\begin{aligned} H(\boldsymbol{y}) &= -E\{\log p(\boldsymbol{y})\} \\ &= E\{\log |\det J(\boldsymbol{x})|\} - E\{\log p(\boldsymbol{x})\} \end{aligned} \tag{4}$$

where $J(\boldsymbol{x})$ is the Jacobian matrix whose elements are partial derivatives $\partial y_j / \partial x_i$. In [5], it is shown that maximizing the joint entropy $H(\boldsymbol{y})$ of the output components y_i can minimize the mutual information among outputs. Differentiating $H(\boldsymbol{y})$ with respect to W leads to the learning rule for ICA:

$$\Delta W \propto W^{-T} - \boldsymbol{\varphi}(\boldsymbol{u})\boldsymbol{x}^T \tag{5}$$

and multiplying $W^T W$, we get the natural gradient [9] speeding up the convergence rate

$$\Delta W \propto [I - \boldsymbol{\varphi}(\boldsymbol{u})u^T]W \tag{6}$$

where

$$\boldsymbol{\varphi}(\boldsymbol{u}) = \left[-\frac{\frac{\partial p_1(u_1)}{\partial u_1}}{p_1(u_1)}, \cdots, -\frac{\frac{\partial p_N(u_N)}{\partial u_N}}{p_N(u_N)} \right]. \tag{7}$$

In this paper, we adopt the extended Infomax algorithm in [9] because it is easy to implement with less strict assumptions on source distribution.

2.2. Algorithm : ICA-FX

The main idea of the proposed feature extraction algorithm is very simple. In applying ICA to feature extraction, we include output class information in addition to input features.

ICA is classified as unsupervised learning because it outputs a set of maximal independent component vectors. This unsupervised method by nature is related to the input distribution, but cannot guarantee good performance in classification problems. Instead of using only the input vectors, we treat the output class information as one of the input features and append it to the inputs for ICA. This makes the ICA become a supervised learning.

We restrict our attention on linear transformations from original features $\boldsymbol{x} = [x_1, \cdots, x_N]^T$ to newly extracted features $\boldsymbol{f}_a = [f_1, \cdots, f_M]^T$. Then, the purpose is to find $K_a \in \Re^{M \times N}$ that maximize $I(c; \boldsymbol{f}_a)$, where c represents output class and $\boldsymbol{f}_a = K_a\boldsymbol{x}$.

To solve this problem, consider the structure shown in Fig. 1. Here, $\boldsymbol{x}$ is fully connected to $\boldsymbol{u} = [u_1, \cdots, u_N]$, c is connected to $\boldsymbol{u}_a = [u_1, \cdots, u_M]$, and $u_{N+1} = c$. Thus, the weight matrix $\boldsymbol{W} \in \Re^{(N+1)\times(N+1)}$ becomes

$$\boldsymbol{W} = \begin{pmatrix} w_{1,1} & \cdots & w_{1,N} & w_{1,N+1} \\ \vdots & & \vdots & \vdots \\ w_{M,1} & \cdots & w_{M,N} & w_{M,N+1} \\ w_{M+1,1} & \cdots & w_{M+1,N} & 0 \\ \vdots & & \vdots & \vdots \\ w_{N,1} & \cdots & w_{N,N} & 0 \\ 0 & \cdots & 0 & 1 \end{pmatrix}. \tag{8}$$

The i-th row of $\boldsymbol{W}$ will produce f_i, $i = M+1, \cdots, N$. Note that $w_{i,N+1} = 0$. This implies that f_i can be made to be independent of c by the assumption.

If we assume that $u_1, \cdots, u_N, u_{N+1}$ are independent of each other, the log likelihood of the given data becomes

$$L(\boldsymbol{u}, c, \boldsymbol{W}) = \log |\det \boldsymbol{W}| + \sum_{i=1}^{N} \log p_i(u_i) + \log p(c), \tag{9}$$

because

$$\begin{aligned} p(\boldsymbol{x}, c) &= |\det \boldsymbol{W}| \, p(\boldsymbol{u}, u_c) \\ &= |\det \boldsymbol{W}| \prod_{i=1}^{N} p_i(u_i) \, p(c). \end{aligned} \tag{10}$$

Using the maximum likelihood estimation criterion, we are to maximize L, and this can be achieved by the steepest

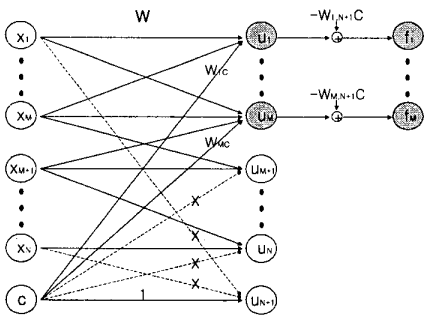

Fig. 1. Feature extraction algorithm based on ICA (ICA-FX)

ascent method. Because the last term in (9) is a constant, differentiating (9) with respect to $\boldsymbol{W}$ leads to

$$\begin{aligned} \frac{\partial L}{\partial w_{i,j}} &= \frac{adj(w_{j,i})}{|\det \boldsymbol{W}|} - \varphi_i(u_i)x_j \qquad i,j = 1,\cdots,N \\ \frac{\partial L}{\partial w_{i,N+1}} &= -\varphi_i(u_i)c \qquad\qquad\qquad i = 1,\cdots,M \end{aligned} \tag{11}$$

where $adj(\cdot)$ is adjoint and $\varphi_i(u_i) = -\frac{\frac{dp_i(u_i)}{du_i}}{p_i(u_i)}$. Note that c has categorical values.

If we denote the upper left $N \times N$ matrix of $\boldsymbol{W}$ in (8) as W, we can see that $|\det \boldsymbol{W}| = |\det W|$ and $\frac{adj(w_{j,i})}{|\det \boldsymbol{W}|} = W_{i,j}^{-T}$. Thus the learning rule becomes

$$\begin{aligned} \Delta W &\propto W^{-T} - \boldsymbol{\varphi}(\boldsymbol{u})\boldsymbol{x}^T \\ \Delta W_{N+1} &\propto -\boldsymbol{\varphi}(\boldsymbol{u}_a)c, \end{aligned} \tag{12}$$

where $W_{N+1} = [w_{1,N+1},\cdots,w_{M,N+1}]^T \in \Re^M$, $\boldsymbol{\varphi}(\boldsymbol{u}) = [\varphi_1(u_1),\cdots,\varphi_N(u_N)]^T$ and $\boldsymbol{\varphi}(\boldsymbol{u}_a) = [\varphi_1(u_1),\cdots,\varphi_M(u_M)]^T$.

Since the two terms in (12) have different tasks regarding the update of separate matrix W and W_{N+1}, we can divide the learning process and applying natural gradient on updating W, we get the following weight updating rule:

$$\begin{aligned} W^{(t+1)} &= W^{(t)} + \mu_1[I_N - \boldsymbol{\varphi}(\boldsymbol{u})\boldsymbol{f}^T]W^{(t)} \\ W_{N+1}^{(t+1)} &= W_{N+1}^{(t)} - \mu_2\boldsymbol{\varphi}(\boldsymbol{u}_a)c, \end{aligned} \tag{13}$$

where I_N is a $N \times N$ identity matrix, μ_1 and μ_2 are learning rate that can be set differently. Note that the learning rule for W is the same as the original ICA learning rule (5), and also note that $\boldsymbol{f}_a$ corresponds to the first M elements of $W\boldsymbol{x}$.

When u_i's, $i = 1,\cdots,M$, are made to be independent of c, then $f_i = u_i - w_{i,N+1}c$ depends on c if $w_{i,N+1} \neq 0$ and $c \neq 0$. Therefore, f_i contains useful information on c. Moreover, in the following, we can show that $\boldsymbol{f}_a(= [f_1,\cdots,f_M]^T)$ contains the maximal information of c when u_i's, $i = 1,\cdots,N$, are independent of c.

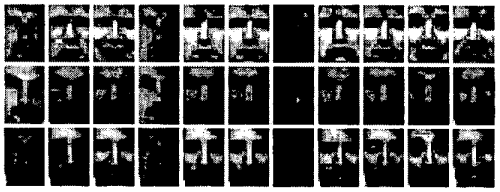

Fig. 2. Yale Database

By the data processing inequality [10], we can see that

$$\begin{aligned} I(\boldsymbol{x};c) &\geq I(W\boldsymbol{x};c) \\ &= I(\{f_1,\cdots,f_M,u_{M+1},\cdots,u_N\};c) \\ &\geq I(\boldsymbol{f}_a;c). \end{aligned} \tag{14}$$

In the first inequality, equality holds if W is nonsingular, and in the second inequality, if $I(u_i;c) = 0$ for $i = M+1,\cdots,N$, equality holds.

Thus $I(\boldsymbol{f}_a;c)$ has its maximum value $I(\boldsymbol{x};c)$, if the assumptions holds.

Therefore, we can extract the optimal features by the proposed algorithm when it find the optimal solution by (13).

Because the proposed ICA-FX algorithm is developed for two-class problems, we need to extend it to multi-class problems for face recognition. Consider there are n_c classes. The simplest way to encode n_c classes is to use n_c bits with 1 at the corresponding bit and 0's at the others. We use this class encoding scheme, instead of using one class bit, with the same learning rule (13).

3. EXPERIMENTAL RESULTS

We have applied ICA-FX to Yale and AT&T face databases. The Yale face database contains 165 grayscale images of 15 individuals. There are 11 images per subject with different facial expressions or configurations. In [2], the authors reports two types of databases: closely cropped set and full face set. We used closely cropped set and downsampled the images into 21 $\times$ 30 pixels for efficiency. Figure 2 represents the images of the first three individuals of the dataset.

We compared the performance of ICA-FX with those of other algorithms: PCA (Eigenface), FLD (Fisherface), ICA, and kernel methods. For ICA, we used extended infomax algorithm in [9]. In ICA-FX, we set learning rate $\mu_1 = 0.002$ and $\mu_2 = 0.1$ which were chosen empirically. In the experiments, we firstly conducted PCA on 630 pixels and used the first 30 principle components as in [2]. After then, we

Table 1. Experimental results on Yale database

Method	Reduced Space	No. of Error	Error Rate (%)
Eigenface	30	37	22.42
ICA	30	39	23.64
Fisherface	14	13	7.88
Kernel Eigenface (d=3)	60	40	24.24
Kernel Fisherface (G)	14	10	6.06
ICA-FX	**14**	**6**	**3.64**

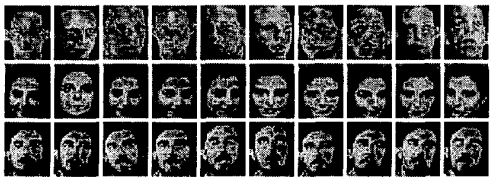

Fig. 3. AT&T Database

applied FLD, ICA, and ICA-FX on these 30 principle components. Table 1 is the error rates of each methods. In the test, the error rates were determined by the 'leave-one-out' strategy and recognition was performed using one nearest neighbor classifier as in [2]. In the table, the performances of kernel Eigenface, and kernel Fisherface are from [4]. The number of features in ICA-FX was determined by conducting several experiments with different numbers of features and selecting the best one.

The AT&T database of faces [4], contains a set of face images. The database was used in the context of a face recognition project carried out in collaboration with the Speech, Vision and Robotics Group of the Cambridge University Engineering Department.

In AT&T database of faces, there are ten different images of each of 40 distinct individuals. We have downsampled the images into 23 × 28 pixels for efficiency. Figure 3 is the images of the first three individuals.

The experiments were performed exactly the same way as in Yale database, except that 40 principle components were used after [4]. Table 2 is the performances of the methods. As in Yale database, the performances of the kernel methods are from [4].

As shown in the tables, we can see that ICA-FX outperforms the other methods for both databases with comparatively less features.

Table 2. Experimental results on AT&T database

Method	Reduced Space	No. of Error	Error Rate (%)
Eigenface	40	17	4.25
ICA	40	16	4.00
Fisherface	39	16	4.00
Kernel Eigenface (d=3)	40	8	2.00
Kernel Fisherface (G)	39	5	1.25
ICA-FX	**10**	**4**	**1.00**

4. CONCLUSIONS

In this paper, we have proposed an algorithm for feature extraction and applied it to face recognition. The proposed algorithm is based on ICA and can generate appropriate features for classification problems.

In the proposed algorithm, we added class information in training ICA. The added class information plays a critical role in the extraction of useful features for classification. With the additional class information we can extract new features containing maximal information about the class. The number of extracted features can be arbitrarily chosen.

Since it uses the standard feed-forward structure and learning algorithm of ICA, it is easy to implement and train. Experimental results for the Yale and the AT&T databases show that the proposed algorithm performs well in face recognition system.

5. REFERENCES

[1] M. Turk and A. Pentland, "Face recognition using eigenfaces," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 1991, pp. 586–591.

[2] P.N. Belhumeur, J.P. Hespanha, and D.J. Kriegman, "Eigenfaces vs. fisherfaces: recognition using class specific linear projection," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 711–720, July 1997.

[3] C. Liu and H. Wechsler, "Evolutionary pursuit and its application to face recognition," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 22, no. 6, pp. 570–582, June 2000.

[4] M.-H. Yang, "Face recognition using kernel methods," *Advances of Neural Information Processing Systems*, vol. 14, 2002, to appear.

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

[6] M.S. Bartlett and T.J. Sejnowski, "Viewpoint invariant face recognition using independent component analysis and attractor networks," *Neural Information Processing Systems-Natural and Synthetic*, vol. 9, pp. 817–823, 1997.

[7] G. Donato, M.S. Bartlett, J.C. Hager, P. Ekman, and T.J. Sejnowski, "Classifying facial actions," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 21, no. 10, pp. 974–989, Oct. 1999.

[8] N. Kwak, C.-H. Choi, and C.-Y. Choi, "Feature extraction using ica," in *Proc. Int'l Conf. on Artificial Neural Networks 2001*, Vienna Austria, Aug. 2001, pp. 568–573.

[9] T-W. Lee, M. Girolami, and T.J. Sejnowski, "Independent component analysis using an extended infomax algorithm for mixed sub-gaussian and super-gaussian sources," *Neural Computation*, vol. 11, no. 2, pp. 417 – 441, Feb. 1999.

[10] T.M. Cover and J.A. Thomas, *Elements of Information Theory*, John Wiley & Sons, 1991.