Supervised LLE in ICA Space for Facial Expression Recognition

Qijun Zhao
Department of Computer Scicence and Engineering
Shanghai Jiao Tong University
Shanghai, 200240, China
E-mail: csqjzhao@polyu.edu.hk

David Zhang
Department of Computing
Hong Kong Polytechnic University
Hung Hom, Kowloon
Hong Kong, China
E-mail: csdzhang@comp.polyu.edu.hk

Hongtao Lu
Department of Computer Scicence and Engineering
Shanghai Jiao Tong University
Shanghai, 200240, China
E-mail: lu-ht@cs.sjtu.edu.cn

***Abstract*— Locally linear embedding (LLE) is an unsupervised nonlinear manifold learning algorithm. It performs well in visualizing data yet has a very poor recognition rate in facial expression recognition. In this paper, to improve the performance of LLE in facial expression recognition, we first employ the independent component analysis (ICA) technique to preprocess the face images such that they are represented by some independent components and some noise is filtered from them. We then propose a Supervised LLE (SLLE) algorithm to learn the hidden manifold. SLLE constructs the neighborhood graphs for the data according to the Euclidean distances between them and the cluster information of them. Its embedding step is the same as that of LLE. Finally, we use a generalized regression neural network (GRNN) to learn the implicit nonlinear mapping from the ICA space to the embedded manifold. Experiments on the JAFFE database show promising results.**

I. INTRODUCTION

Recent developments in both data visualization [1][2] and pattern recognition [3][4] show that human face images are much more likely to reside on a low dimensional nonlinear manifold which is embedded in the original high dimensional image space. Learning the manifold underlying the face images is important in applications such as face recognition and facial expression recognition because the manifold reveals the intrinsic structure of the data and the essential relations in them. In this paper, based on the Locally Linear Embedding (LLE) [2] and Independent Component Analysis (ICA) [5] techniques, we propose a novel method, Supervised Locally Linear Embedding (SLLE) in the ICA space, to learn the facial expression manifold for recognizing facial expressions.

Generally, manifold learning algorithms are of two types: linear and nonlinear algorithms. Some representative linear manifold learning algorithms include Principal Component Analysis (PCA) [6], Linear Discriminant Analysis (LDA) [7], and Independent Component Analysis (ICA). These methods effectively see only the Euclidean structure in data [8] and thus often fail to deal well with the nonlinear properties in face images, for example, pose variations, illumination conditions, and facial expressions [9]. Locally Linear Embedding (LLE) [2] is a promising method recently proposed for learning nonlinear manifold. Its applications in visualizing face image data show excellent performance in revealing the underlying nonlinear face manifold [8]. However, LLE has a very low recognition rate when it is used for facial expression recognition, as shown in [3].

Facial expression recognition is of interest to researchers because of its many potential applications such as more intelligent human-computer interface and human emotion interpretation. Many methods have been proposed to recognize facial expressions. The facial action coding system (FACS) [10], uses 44 action units (AU) to describe facial actions according to their locations as well as their intensities. Instead of dividing the whole face into different units, Lyons *et al.* [11] chose 34 feature points (*e.g.* corners of the eye and mouth) on a face for the purpose of analyzing its expression. Another method is to base the facial expression analysis and recognition directly on the whole face image rather than its segmentations or so-called feature points [12]. This method is typically simple and fast. For this sake, we used the whole face images as the input of our proposed algorithm in the experiments.

Chang *et al.* [3] have explored nonlinear manifold learning algorithms for facial expression recognition. They claimed that all face images with various facial expressions lie on a smooth manifold which is embedded in the image space with the neutral face as the central reference point. They used image sequences of a subject which demonstrated all expressions under consideration and employed LLE and Lipschitz embedding to learn the manifold after preprocessing the image sequences by Active Wavelet Networks (AWN). Their results show that the classification performance of LLE is very poor. There are at least two reasons for the failure of LLE in facial expression recognition. First, the neighborhood graphs constructed for the face images can not accurately describe the relations among them in terms of the similarity in facial expressions, not only because of the noise in the data but also because of the mutual, or redundant, information in them. Second, LLE, as an unsupervised algorithm, makes no use of the cluster information of the data, which is often given as an important *a prior* knowledge.

0-7803-9422-4/05/$20.00 ©2005 IEEE

In this paper, to improve the performance of LLE in facial expression recognition, we present a Supervised LLE (SLLE) algorithm with ICA as its preprocessing step. It learns the manifold of facial expression from static face images of different subjects, rather than image sequences of one subject, with the cluster information (the expressions) of the images given. In our algorithm, we take each pixel in an image as a random variable and each image is represented as a column vector by concatenating its columns. As a result, a face image with n pixels is represented as an n-dimensional random vector. In order to remove the noise and redundant information in the data, we first employ the FastICA [13] to find m ($m<n$) independent components to represent the face image data. We then use SLLE to learn the manifold of facial expression from the face images in the ICA transformed space. A generalized regression neural network (GRNN) [14] is employed to learn the implicit mapping from the ICA space to the facial expression manifold such that new samples can be successfully projected onto the manifold. We finally use a K nearest neighbor (KNN) classifier to recognize facial expressions on face images. Using the JAFFE database, we compared the proposed SLLE in the ICA space with LLE in terms of the clustering performance in their learned embedded spaces and the classification performance as well. The experimental results demonstrate that the SLLE in the ICA space overwhelms the LLE in both clustering and classifying facial expressions. The obtained recognition rates are also better than existing ones on the JAFFE database.

The remainder of this paper is organized as follows. In Section II, we briefly introduce the ICA algorithm used by this paper. In Section III, the supervised LLE algorithm is presented. Section IV provides our experiments on the JAFFE database. Section V offers our conclusion.

II. INDEPENDENT COMPONENT ANALYSIS

Independent Component Analysis (ICA) can be seen as an extension of PCA. The goal of ICA is to seek a number of axes, on which the projections of data are independent, not merely uncorrelated as in PCA. Unlike PCA, ICA has no numerical solution, but is solved in an iterative way with some criteria optimized in the procedure. There are a number of different ICA algorithms which measure the independence differently and optimize their criteria differently. Here, for the sake of efficiency, we employ a highly efficient ICA algorithm, the FastICA, proposed by Hyvarinen [13].

Given N observed samples, $x_1, x_2, ..., x_N \in R^n$, which are centralized, *i.e.* the mean of these samples $\bar{x} = \frac{1}{N}\sum_{i=1}^{N} x_i$ is zero, the first step in FastICA is to whiten the samples. Let $\Sigma = \frac{1}{N}\sum_{i=1}^{N} x_i x_i^T$ denotes the covariance matrix ('T' means the transpose operation). By EVD on Σ, we obtain

$$\Sigma = E\Lambda E^T, \tag{1}$$

where E is the orthogonal matrix of its eigenvectors and Λ is the diagonal matrix of its eigenvalues, $\Lambda = diag(\lambda_1, \lambda_2, ..., \lambda_n)$. Then the whitened sample $\tilde{x}_i$ of x_i is as follows,

$$\tilde{x}_i = E\Lambda^{-1/2}E^T x_i, \quad i = 1, 2, ..., N. \tag{2}$$

After being whitened, the sample data have an identity covariance matrix. By omitting some less dominant eigenvectors, we can reduce the dimension of data in advance and filter the noise in them to some extent.

These whitened sample data $\tilde{X} = [\tilde{x}_1 \tilde{x}_2 ... \tilde{x}_N] \in R^{\tilde{n}\times N}$ ($\tilde{n}$ is the number of retained eigenvectors during whitening) are then used as the input to the general ICA model. This model assumes that the observed data components are generated linearly from some independent sources, say $s_1, s_2, ..., s_m$. Here m is the estimated number of independent sources. Mathematically, the general ICA model can be formulated as follows,

$$Y = W\tilde{X}, \tag{3}$$

where $W \in R^{m\times \tilde{n}}$ is the transformation matrix of ICA from the observed data components to the independent components of them, and where each column of $Y \in R^{m\times N}$ gives the m independent components of an observed sample. The goal of ICA is to find Y and W for $\tilde{X}$, such that the rows in Y are as independent from each other as possible.

To solve this ICA model, we should define a measurement of the independence of components. FastICA defines the negentropy of random vectors to measure the independence of components. Specifically, to find one independent component $y_{ij} = W_j \tilde{x}_i$ (W_j is the jth row of W, $j = 1, 2, ..., m$) for the observed sample $\tilde{x}_i, i = 1, 2, ..., N$, it maximizes the following function J_G:

$$J_G(W_j) = [\varepsilon\{G(W_j\tilde{x}_i)\} - \varepsilon\{v\}]^2, \tag{4}$$

under the constraint of $\varepsilon\{(W_j\tilde{x}_i)^2\} = 1$. Here v is a standard Gaussian variable, ε is the expectation operator, and G is any non-quadratic function. The final W_j is that maximizes the total sum of the above function for all samples. After obtaining one independent component, the others can be calculated by minimizing the mutual information between the calculated components and the existent ones. To solve these optimization problems, Hyvarinen proposed a fixed-point algorithm which solves them efficiently in an iterative way.

Equation (4) is an approximation to the real negentropy defined as

$$J(y_{ij}) = H(y^g) - H(y_{ij}), \tag{5}$$

where y^g is a Gaussian random variable of the same variance as y_{ij} and *H(s)* is the differential entropy of *s*, which is defined as

$$H(s) = -\int f(s)\log f(s)ds, \tag{6}$$

where *f(s)* is the density of *s*. The reason why this can measure the independence of random variables lies in the fact that a non-Gaussian random variable tends to be more independent, which can be taken as a corollary of the Central Limit Theorem. For more details on the FastICA algorithm, one can refer to [5] and [13].

III. SUPERVISED LOCALLY LINEAR EMBEDDING

Supervised locally linear embedding (SLLE), proposed here, is an improved version of LLE for the purpose of clustering and classifying facial expressions when the cluster information of the sample face image data is available. SLLE also has three steps as LLE. First, a neighborhood graph is constructed for each data point. But SLLE constructs these neighborhood graphs with a strict constraint imposed: only those points in the same cluster as the point under consideration can be its neighbors. In other words, the primary focus of the proposed SLLE is restricted to reveal and preserve the neighborhood in a cluster scope. The neighborhood which is preserved in a cluster scope helps to condense the samples in the same cluster and thus improves the performance of distance based classifiers. There are two ways to find the neighbors $x_j, j \in I_i$, of a given sample $x_i, i = 1,2,...,N$ (here, I_i denotes the index set of the neighbors of x_i): ε-neighborhoods and *k* nearest neighbors. The method of ε-neighborhoods takes all the points which have an Euclidean distance to x_i smaller than ε as the neighbors of x_i, whereas the method of *k* nearest neighbors takes the *k* nearest points to x_i according to the Euclidean distances between them. We employed the method of *k* nearest neighbors in our experiments for the simplicity sake.

SLLE then reconstructs each data point by linear combining its neighbors which are denoted in its neighborhood graph constructed in the first step. The resulting weights W_{ij} should ensure the total reconstruction error as small as possible. In other words, the W_{ij} are calculated to minimize the reconstruction error of x_i from x_j, *i.e.*

$$\left\| x_i - \sum_{j \in I_i} W_{ij} x_j \right\|^2, \tag{7}$$

subject to the constraint $\sum_{j \in I_i} W_{ij} = 1$. These weights play vital roles in the embedding process because the local neighborhood structures in the data are entirely encoded in them.

Finally, SLLE constructs the mappings on the low-dimensional manifold for the whole data set through seeking low-dimensional vectors $\{y_i \mid i = 1,2,...,N\}$ such that the following reconstruction error on the low-dimensional manifold is minimized:

$$\sum_{i=1}^{N} \left\| y_i - \sum_{j \in I_i} W_{ij} y_j \right\|^2, \tag{8}$$

where W_{ij} are the same as those in (7). The neighborhood structures in the data are preserved through conducting the reconstruction process in the low dimensional embedded space with the weights and neighborhoods obtained in the original higher dimensional space.

Algorithm 1 The SLLE Algorithm

Input: $X = \{x_i \in R^m \mid i = 1,2,\cdots,N\}$; $C = \{c_i \mid c_i$ is the cluster label of x_i, $i = 1,2,\cdots,N\}$; $K > 0$ is the size the neighborhood

Output: $Y = \{y_i \in R^l \mid i = 1,2,\cdots,N;\ l \le m\}$

procedure Y = SLLE(X, C, K)
  for $i \leftarrow 1, N$ **do**
    for $j \leftarrow 1, N$ **do**
      if $j \ne i$ and $c_j = c_i$ **then**
        $d_{ij} \leftarrow \|x_i - x_j\|$
      else
        $d_{ij} \leftarrow \infty$
      end if
    end for
    Sort $\{d_{ij}\}$ such that $d_{ij_1} \le d_{ij_2} \le \cdots \le d_{ij_N}$
    Choose $x_{j_1}, x_{j_2}, \cdots, x_{j_K}$ as the neighbors of x_i
    $\{W_{ij_k} \mid k = 1,2,\cdots,K\} \leftarrow arg\ min \|x_i - \sum_{k=1}^{K} W_{ij_k} x_{j_k}\|$ $s.t.$ $\sum_{k=1}^{K} W_{ij_k} = 1$
  end for
  $\{y_i \mid i = 1,2,\cdots,N\} \leftarrow arg\ min \sum_{i=1}^{N} \|y_i - \sum_{k=1}^{K} W_{ij_k} y_{j_k}\|$
end procedure

The last two steps of SLLE are the same as these in LLE. Roweis and Saul have proven that the above optimization problems can be solved by standard methods in linear algebra: constrained least squares fits to compute the weights and singular value decomposition to perform the

Fig. 1. Some example JAFFE images used in our experiments

embedding [2][8]. Here, however, due to the space limitation, we omit the details of solving the optimization problems.

Algorithm 1 shows the SLLE algorithm in detail. Obviously, SLLE outputs the projections of the input sample data instead of the nonlinear mapping function. In order to project a new sample x^* into the embedded space learned by SLLE, we use a generalized regression neural network, $y = GRNN(x)$, to learn the implicit mapping function with $\{(x_i, y_i) \mid i = 1, 2, ..., N\}$ as the training data. The output of the trained neural network, $y^* = GRNN(x^*)$, then gives the projection of the new sample.

IV. EXPERIMENTS

We have tested the proposed SLLE in the ICA space for facial expression recognition using a standard database, the Japanese Female Facial Expression (JAFFE) database [11]. This database consists of 213 frontal face images of 7 facial expressions (Angry, Depressed, Fear, Happy, Normal, Sad, and Surprised) posed by 10 Japanese female models. These images were taken under the same illumination conditions. Figure 1 shows some example images in the JAFFE database.

A. Testing Methodology

In order to improve the correspondence between features on different face images, we first preprocessed the images in the JAFFE database such that all the face images have the same inter-ocular distance and an average face shape. We then cropped the images to retain only the face area and resized them to the size of 24×24 to further reduce the computational cost.

The parameters used in the experiments were set by experience. Specifically, we set the dimension of the ICA space to 50, the size of neighborhood to 5, and the dimension of the embedded space to 10.

The experiments included two parts. In the first part, we compared the embedded spaces learned from all the face images in the JAFFE database by LLE in the original space, LLE in the ICA space, and SLLE in the ICA space, from the aspect of the cluster property of the data samples in these spaces. In the second part, we tested the training and testing recognition rates achieved by LLE in the original space, SLLE in the original space, LLE in the ICA space, and SLLE in the ICA space to test the performance of SLLE and ICA. Recognition was conducted by a K nearest neighbor classifier (K=5) and recognition rates were measured using the leave-one-out strategy: one image was selected out from each cluster to form a test set with 7 images and the rest 206 images were used as training samples. In the following section, we will give the experimental results of the two parts, respectively.

B. Experimental Results

1) Cluster Property of Samples in the Learned Embedded Spaces: We compared the cluster property of all samples in the embedded spaces learned by LLE in the original space, LLE in the ICA space, and SLLE in the ICA space. The results are shown in Figure 2, where the samples from the same cluster (*i.e.* with the same facial expression) are denoted by a same color. Figures 2(a) and 2(b) are the projections on the first three dimensions in the embedded spaces learned by LLE in the original space and the ICA space, respectively. Obviously, LLE in the ICA space separates the samples better. This owes to that the neighborhood in the ICA space is supposed to describe the similarity between face images with regard to facial expressions more accurately. Figures 2(c) and 2(d) present the results of SLLE in the ICA space. It can be seen that in the embedded space learned by SLLE in the ICA space, the samples in the same cluster become more condensed in the sense of the number of samples which are not only in the neighborhood of a sample but also of the same cluster as it.

2) Recognition Rates: We conducted classification in the learned embedded spaces using a K nearest neighbor classifier (K=5). We compared both training and testing recognition rates for four cases: LLE in the original space (denoted by LLE in Figure 3), SLLE in the original space (SLLE), LLE in the ICA space (ICA+LLE), and SLLE in the ICA space (ICA+SLLE). The average results shown in Figure 3 illustrate that SLLE has a 100% recognition rate in training whereas the best training recognition rate of LLE is less than 70% and its performance in testing (88.56% in the original space and 89.99% in the ICA space) is also much better than both that of LLE (27.16% in the original space and 42.87% in the ICA space) but also that of [11] (70%). Besides, for either SLLE or LLE, the embedded space learned in the ICA space also performs better in classification than that learned in the original space. This proves the effectiveness of ICA as a preprocessing step for manifold learning in classification.

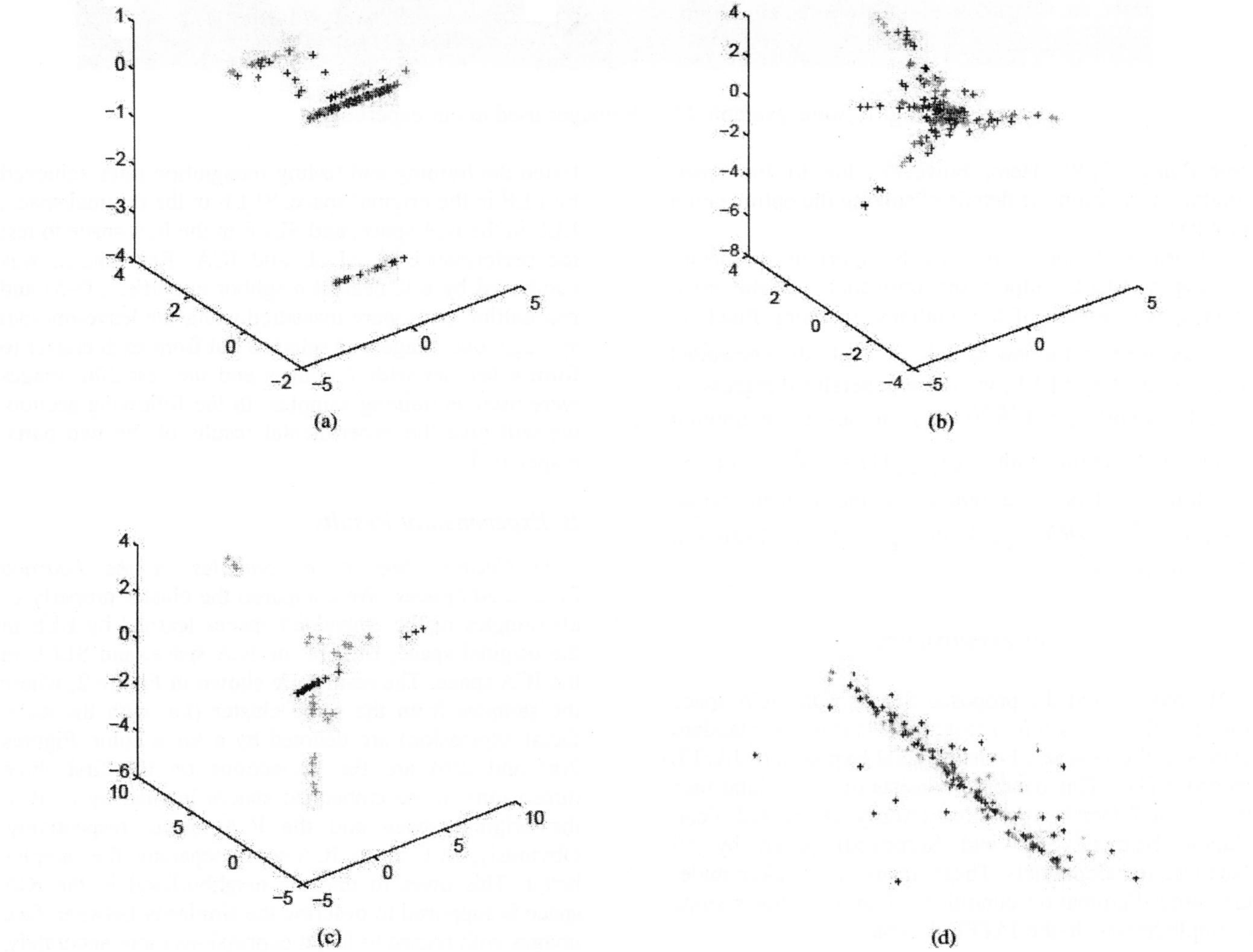

Fig. 2. The projection on the first three dimensions in the embedded spaces learned by (a) LLE in the original space, (b) LLE in the ICA space, and (c) SLLE in the ICA space. (d) shows the details of the central part in (c). The same color means the corresponding samples are in the same cluster

V. CONCLUSIONS

This paper has proposed a supervised locally linear embedding (SLLE) algorithm. Unlike LLE, SLLE confines the neighborhood of a sample in the samples which are in the same cluster as it according to the given cluster information of the sample data. This greatly improves the cluster property of the data in the learned embedded space and thus helps to improve the classification performance on the data. Besides, we employ the FastICA algorithm to preprocess the input face images such that the image data are represented by some independent components and some noise and redundant information in them are removed. This allows the neighborhood graphs constructed from the preprocessed data to more accurately measure the similarity between face images that exhibit a variety of facial expressions.

Our experiments on the JAFFE database demonstrate the excellent performance of the proposed SLLE algorithm and ICA preprocessing step for facial expression recognition. In the experiments, SLLE in the ICA space achieved a recognition rate that is better than both that of LLE and the existing ones. However, in addition to the different intensities of different expressions on different faces considered here, there are still some other factors needing consideration, such as pose variations and illumination conditions. This gives rise to two valuable future research topics: to reveal the effect of these factors on the facial expression manifold and to improve the robustness of the proposed algorithm.

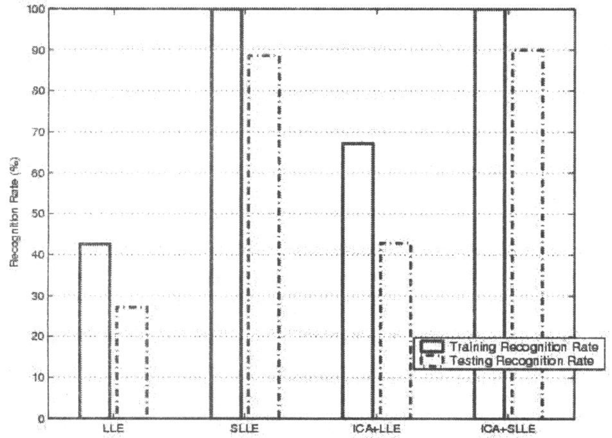

Fig. 3. Recognition rates in both training and testing

ACKNOWLEDGMENT

The authors would like to thank Mr. Martin Kyle for his help on the presentation of this paper.

REFERENCES

[1] J. B. Tenenbaum, V. de Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction", *Science*, vol. 290, no. 5500, pp. 2319-2323, 2000.

[2] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding", *Science*, vol. 290, no. 5500, pp. 2323-2326, 2000.

[3] Y. Chang, C. B. Hu, and M. Turk, "Manifold of facial expression", *IEEE International Workshop on Analysis and Modeling of Faces and Gestures: AMFG 2003*, Nice, France, 2003, pp. 28-35.

[4] X. Geng, D. C. Zhan, and Z. H. Zhou, "Supervised nonlinear dimensionality reduction for visualization and classification", *IEEE Trans. on Systems, Man, and Cybernetics -- Part B: Cybernetics*, vol. 35, no. 6, pp. 1098-1107, 2005.

[5] S. Robert and R. Everson, *Independent Component Analysis: Principles and Practice*, Cambridge University Press, 2001.

[6] I. T. Jolliffe, *Principle Component Analysis*, 2nd Ed. New York: Springer, 2002.

[7] K. Fukunaga, *Introduction to statistical pattern recognition*, 2nd Ed. Academic Press, Inc., 1990.

[8] L. K. Saul and S. T. Roweis, "Think globally, fit locally: Unsupervised learning of low dimensional manifolds", *J. Mach. Learn. Res.*, vol. 4, pp. 119-155, 2003.

[9] W. Zhao, R. Chellappa, P. Phillips, and A. Rosenfeld, "Face recognition: a literature survey," *ACM Comput. Surv.*, vol. 35, no. 4, pp. 399-458, 2003.

[10] P. Elkman and W. V. Friesen, *Facial Action Coding System (FACS)*, Manual. Palo Alto: Consulting Psychologists Press, 1978.

[11] M. J. Lyons, S. Akamatsu, M. Kamachi, and J. Gyoba, "Coding facial expressions with Gabor wavelets", *Proceedings of the 3rd IEEE International Conference on Automatic Face and Festure Recognition*, IEEE Computer Society, 1998, pp. 200-205.

[12] C. Padgett and G. Cottrell, "Identifying emotion in static face images", *Proceedings of the 2nd Joint Symposium on Neural Computing*, San Diego, CA: Univeristy of California, 1995, vol. 5, pp. 91-101.

[13] A. Hyvarinen, "Fast and robust fixed-point algorithms for independent component analysis", *IEEE Trans. on Neural Networks*, vol. 10, no. 3, pp. 626-634, 1999.

[14] P. D. Wasserman, *Advanced Methods in Neural Computing*. New York: Van Nostrand Reinhold, 1993.