Facial Expression Recognition Based on Supervised LLE Analysis of Optical Flow and Ratio Image

Yu-Kuen Wu and Shang-Hong Lai
Department of Computer Science, National Tsing Hua University, Hsinchu 300, Taiwan
Email: lai@cs.nthu.edu.tw

ABSTRACT

In this paper, we propose a new facial expression recognition algorithm based on supervised locally linear embedding (SLLE) analysis on the optical flow and ratio image. In this algorithm, we first extract the face region from the face image to remove factors due to global head motion. Secondly, we compute the optical flow and ratio image between the neutral face and expression images and then apply the SLLE to extract the low-dimensional discriminating features from the expression motion and brightness variation. Finally, we compute the distance between the low-dimensional feature vectors to recognize the facial expression. Experimental results on the JAFFE face database and Yale database show that the proposed algorithm outperforms previous methods for facial expression recognition.

1: INTRODUCTION

In recently years, facial expression analysis has attracted many researchers in computer vision. During the course of conservation, understanding facial expression can help us understand the meaning of the conversation and decide what should be done. For example, when a speaker found the facial expression of the listener is confused, he knows that he needs to explain in more details. Understanding facial expression is easy for human from the motion of facial muscle. However, it is a challenging problem for a computer to recognize human facial expression.

In recent years, many researchers have developed several different facial expression recognition techniques. One popular technique is image-based approach. Principle component analysis (PCA) has been popularly used for dimension reduction and feature extraction. Eigenface extracted from face images by using PCA is usually used for face recognition or authentication. Some researchers used PCA to extract features from the optical flow of facial expression. Liu et al. [6] proposed to use an eigenflow method for face authentication. Recently, Roweis et al. [1] proposed a locally linearly embedding (LLE) technique to achieve non-linear dimension reduction. LLE is an unsupervised learning algorithm that computes low-dimensional features which preserve the neighborhood relation as that of the high-dimensional data. Ridder et al. [2] proposed the supervised LLE (SLLE) method. The difference between LLE and SLLE is that SLLE includes class label information into LLE to enhance the performance. Liang et al. [3] used SLLE on a whole image for facial expression recognition and the experimental result showed SLLE can achieve better performance than those of PCA and LLE.

Another popular approach for facial expression recognition is the motion-based approach. In the motion-based approach, many researchers computed dense optical flow to obtain facial motion information. For example, Lien et al. [4] used dense optical flow with principle component analysis to recognize facial actions defined in Facial Action Coding System (FACS). Note that the recognition result is heavily dependent on the estimation of optical flow. Unfortunately, accurate optical flow estimation is difficult especially when the motion or the brightness variation between two images is large.

In this paper, we employed the optical flow algorithm by Teng et al. [5] to compute the optical flow and ratio image between the neural face and facial expression images. This algorithm can provide accurate estimation of optical flow as well as the associated ratio image for describing the associated brightness variations. Then, SLLE is applied to these two types of features to provide dimensionality-reduced facial expression features. Finally, facial expression is recognized based on the distance in this SLLE feature space.

The rest of this paper is organized as follows: Section 2 briefly reviews the previous works on nonlinear dimension reduction. Then, we describe the system structure and the proposed facial expression recognition algorithm in section 3. The experimental results by applying the proposed algorithm and previous methods on the public facial expression datasets are reported in section 4. Finally, we conclude this paper in section 5.

2: NONLINEAR DIMENSION REDUCTION

2.1: Locally linear embedding

Roweis et al. [1] proposed locally linearly embedding (LLE) to achieve non-linear dimension reduction. LLE is an unsupervised learning algorithm that can compute low-dimensional data representation which preserves the local neighborhood relation in the original high-dimensional data. In the following, we describe the components in the LLE algorithm to compute a lower-dimensional representation of the original data.

In the simplest formulation of LLE, the distance between every two samples in the training dataset is measured by Euclidean distance. In this step, the distances between each sample X_i and the other samples are computed. The samples with distances being the K closest to X_i are the neighbors of X_i.

In LLE, each sample is reconstructed from a linear combination of its K nearest neighbors, but not all samples can be perfectly reconstructed from its neighbors. The reconstruction error is measured by the square of the distance between a sample X_i and its reconstruction. The cost function which is the summation of all the squares of the reconstruction errors is given as follows:

$$\varepsilon(W) = \sum_i \left\| \vec{X}_i - \sum_j W_{ij} \vec{X}_j \right\|^2 \tag{1}$$

where the weight matrix W can be obtained by minimizing the cost function in Equation (1) subject to two constraints. One is the constraint $W_{ij}=0$ if X_j does not belong to X_i's neighbors. The other constraint is to enforce the combination in the linear combination from the neighbors, i.e.

$$\sum_j W_{ij} = 1 \tag{2}$$

The optimal weight matrix W subject to the above two constraints can be obtained by solving a least squares problem.

In the final steps of LLE, the low-dimensional representation of each sample is reconstructed from the weight matrix W. The d-dimensional Y_i can be obtained by minimizing the following cost function subject to two additional constraints.

$$\Phi(Y) = \sum_i \left\| \overline{Y_i} - \sum_j W_{ij} \overline{Y_j} \right\|^2 \tag{3}$$

These constraints are to center the vectors to the origin and avoid degenerate solutions. For an input testing vector x, there are three steps for finding its d-dimensional LLE representation. (1) Select its K nearest neighbors from the training data set. (2) Compute the weights w_j to reconstruct the testing sample from the linear combination of its neighbors. (3) Compute the linear combination $y= \Sigma_j w_j.Y_j$, where Y_j is the corresponding low-dimensional representation for the training data X_j, to be the d-dimensional representation for x. Thus, the output vector y is computed from the result of LLE training.

2.2: Supervised locally linear embedding

LLE is an unsupervised learning algorithm, because it does not require the class label information. Ridder et al. [2] proposed supervised locally linear embedding (SLLE) analysis, which is an extension of LLE by taking advantage of the class label information. SLLE modified the first step of LLE. When two samples X_i and X_j do not belong to the same class, a penalty term is added into the distance between them as follows:

$$D'(X_i, X_j) = D(X_i, X_j) + \alpha \max_{(k,l)}(D(X_k, X_l))\Delta_{ij} \tag{4}$$

where $D(X_i, X_j)$ is the Euclidean distance between sample X_i and X_j without considering the class label information and $D'(X_i, X_j)$ is the modified distance including the class label information. If X_i and X_j do not belong to the same class, then $\triangle_{ij} = 1$. Otherwise, $\triangle_{ij} = 0$. The value of α is set to be between 0 and 1. This parameter controls the weight of the class label information. In all of our experiments, α is set to 0.2. The inclusion of the penalty term into the LLE computation is to make the selected nearest neighbors to be more likely from the same class.

3: PROPOSED FACIAL EXPRESSION RECOGNITION ALGORITHM

The proposed facial expression recognition is based on the SLLE analysis on the optical flow and ratio image computed from the neutral face and facial expression images of the same person.

3.1: Normalization

To remove the rigid head motion from the expression motion, we extract the facial region based on three facial landmark points. In order to remove the factors that influence the accuracy of the optical flow computation, we estimate an affine transformation to account for the translation, rotation and scaling factors. We manually identify three facial feature points, i.e. the outside canthus of both eyes and the uppermost point on the philtrum, from the face image. Additionally, we need to setup a standard face space and fix the positions of these three facial feature points. The feature point positions in each image are normalized to the standard face space.

3.2: Computing Optical Flow and Ratio Image for Facial Expression Image

We use the neutral face image as the reference image. To compute the facial motion for a facial expression image, we compute the optical flow between the reference image and the facial expression image in the face region.

Moreover, the optical flow only contains the motion information. In many cases, motion information is not enough to describe the facial expression changes in image appearance. The intensity variation information can provide additional information for facial expression recognition.

In the generalized optical flow computational algorithm [5], the optical flow and the associated brightness change are estimated altogether to compensate for brightness variations between two frames. The intensity variation is described by the multiplier *m(r)* as given in the following equation.

$$I_1(r+\delta r)-I_0(r)=-m(r)I_0(r) \quad (5)$$

where $I_0(r)$ is the image intensity at position r of the neural image, $I_1(r+\delta r)$ is the image intensity at position $r+\delta r$ of the expression image, and δr denotes the optical flow at the location r. In this work, we extract the multiplier field $m(r)$ to represent the intensity variation information. The factor $m(r)$ is a ratio between the intensities of the corresponding points. Therefore, $m(r)$ is also called ratio image in this paper.

Before SLLE is applied for dimension reduction, we need to normalize the ratio image to remove some undesirable factor. For example, if an expression face image is lighter than the neutral face image, it will cause the $m(r)$ value of each pixel to be negative. Since this overall illumination change factor is not useful for expression recognition, we normalized the ratio image $m(r)$ value to zero-mean to remove the illumination factor. Figures 1 depicts an example of a facial expression image and the computed optical flow and ratio image.

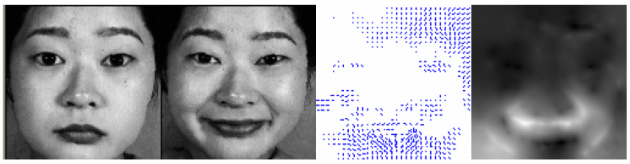

Fig. 1. From left to right: neutral image, expression (happiness) image, optical flow and ratio image.

3.3: Classification

After the lower-dimensional SLLE features are computed from the facial expression images, we employ two different methods for classification. First, we use the minimum class-mean distance classification method means to measure the distance between the testing image and the center mean of each class. If the testing image has the minimum distance with the center mean of a particular class, then we classify the testing image to this particular class.

In the other classification method, we employ the k-nearest-neighbor classification for facial expression recognition. If the feature of a facial expression image is closer to those of samples in a particular class, then the probability of this image belongs to this particular class is larger. Therefore, we use the k-nearest neighbors and apply a voting mechanism for classification.

The k-nearest neighbors have their own corresponding expression classes. Then, we use the k-nearest neighbors in the SLLE space to vote for the class which the input image should belongs to. If a particular expression class has the maximal votes, then this face image will be assigned to this particular class. If two or more expression classes have the same number of votes, we use the sum of distances between SLLE features of the input face image and those of in the expression classes to make the final decision

3.4: Feature Integration

In the proposed method, we compute the optical flow and ratio image for each facial expression image and then apply the SLLE analysis separately for each of these two types of image features. Optical flow represents the motion-based information, while ratio image represents the brightness-based information. Here, we want to combine optical flow and ratio image properly to improve the classification result. To make both types of features equally weighted, we compute a weight such that both types of features have roughly the same level of variations. Let D_{of} represent the sum of distances between the mean of optical flow to all optical flow vectors in the training data, and let D_r represent the sum of distances between the mean of ratio image to all ratio images. We can use the following equation to determine the weight w.

$$w \times D_{of} = (1-w) \times D_r \quad (6)$$

$$w = \frac{D_r}{D_r + D_{of}} \quad (7)$$

A testing image has a d-dimensional optical flow feature vector and a d-dimensional ratio image feature vector after the SLLE dimension reduction. Let dis_{of} and dis_r denote the distances of the optical flow and ratio image feature vectors to the centers of the particular expression in the two SLLE feature spaces. The combined distance is then given by

$$D = w \times dis_{of} + (1-w) \times dis_r \quad (8)$$

Each expression has its own combined distance. If the combined distance to the center of an expression class is shortest, then the expression image is designated to this particular expression.

In the k-nearest-neighbor classification method, both the optical flow and ratio image have their own k-nearest-neighbor voting results. In the following, we will describe the combination of these two voting results. The main idea is to integrate the two voting results. If a particular expression class has the maximal votes from optical flow and ratio image, we will assign the input image to this particular class. If two or more expression classes has the same number of votes form optical flow and ratio image, we still use the combined distance computed from the SLLE optical flow and ratio image feature vectors to decide the final class. The combined distance is given by

$$NN_{dis} = w \times NN_{dis}^{of} + (1-w) \times NN_{dis}^{r} \quad (9)$$

where w is selected to make both types of features be roughly of the same level of variations as described above in this section. Therefore, each expression class has its own NN_{dis} for an input face image. We compare the combined distances NN_{dis} for all expression classes, select the minimum combined distance and assign the input face image to this class.

4: EXPERIMENTAL RESULTS

The JAFFE facial expression database [7] is used to test and verify our facial expression recognition algorithm based on SLLE analysis on the corresponding optical flow and ratio image. The JAFFE database contains ten Japanese females with seven facial expressions, including anger, disgust, fear, happiness, sadness, surprise and neutral expressions. Each subject usually has three images of each of these seven facial expressions. Some expressions of some subjects may have two or four images. The total database contains 213 face images of different expressions.

4.1: Comparison of PCA and SLLE

In the experiment, we use 183 facial expression images from the JAFFE dataset for comparing the dimension reduction results by using PCA and SLLE. PCA is applied to reduce the feature vector to a 10-dimension representation. For SLLE, we set $K = 10$ and $d = 10$.

Figure 2 and 3 show the results of projecting the 10-dimensional features onto the 2D PCA and SLLE spaces, respectively, by applying Multidimensional Scaling (MDS) for visualization. It is obvious that the feature vectors of different clusters are largely overlapped in the PCA space, while they are well separated for different clusters in the SLLE space.

4.2: Comparison

In this experiment, we select the first two images of six different facial expressions of each subject as the training images. The rest images are used for testing. We compare the expression recognition rates of different methods with varying SLLE feature dimensions by using minimum class-mean distance classification. Figure 4 shows the recognition rate of selecting different dimensions when the neighbor number K is set to 12. This experiment also shows combining the optical flow and ratio image features provide more accurate recognition than those based on other features. Here, we compare the recognition results of using the normalized image and the original image for expression recognition. The combined optical flow and ratio image features provides the best result. We can see that the recognition rate on this test dataset can reach about 92% by using SLLE feature dimension reduction on the optical flow and ratio image in conjunction with a simple classifier

Here, we also compare the expression recognition rates of different methods with varying SLLE feature dimensions by using k-nearest neighbor classification, where $k = 20$. Figure 5 shows the recognition rate of selecting different dimensions when the neighbor number K is set to 12. We can see that the recognition rate of using minimum class-mean distance classification is better than that of using the k-nearest-neighbor classification.

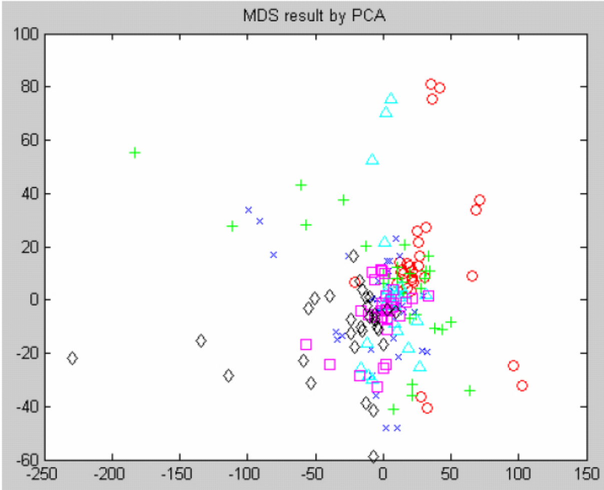

Fig. 2. The PCA distribution of each expression class with projections onto 2D space formed by using MDS.

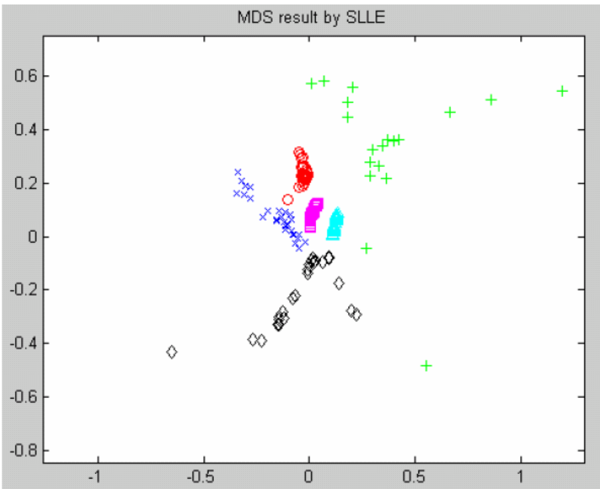

Fig. 3. The SLLE distribution of each expression class with projections onto 2D space formed by using MDS.

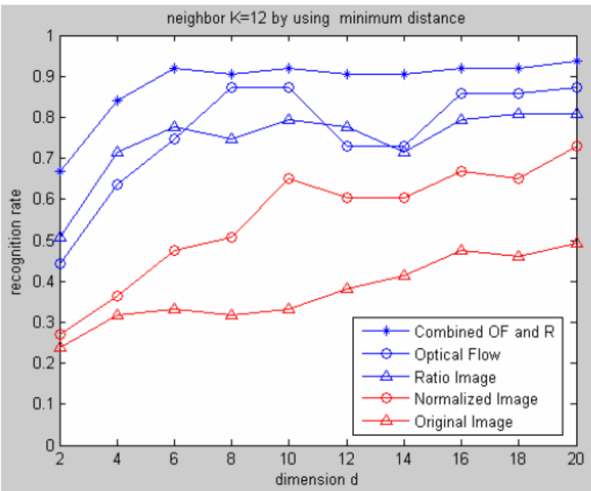

Fig. 4. Comparison of expression recognition rates by using different feature types with varying dimensions and the minimum class-mean distance classifier.

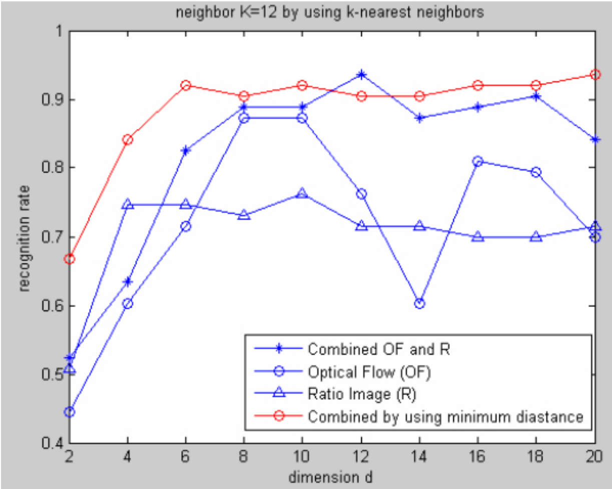

Fig. 5. Comparison of expression recognition rates by using different feature types with varying dimensions and the k-nearest-neighbor classification.

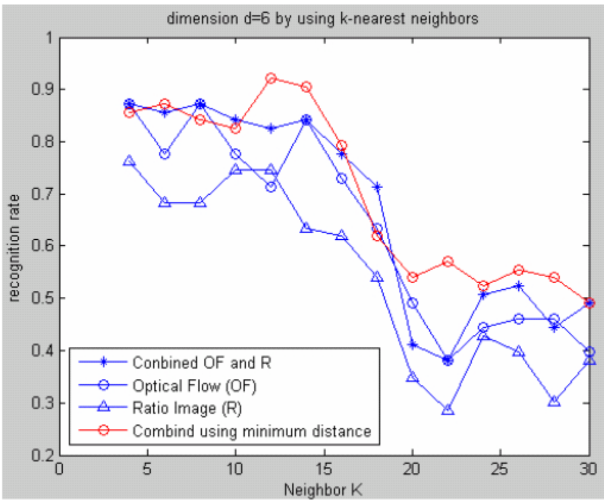

Fig. 7. Comparison of different expression recognition methods of varying neighbor sizes by using the k-nearest-neighbor classification.

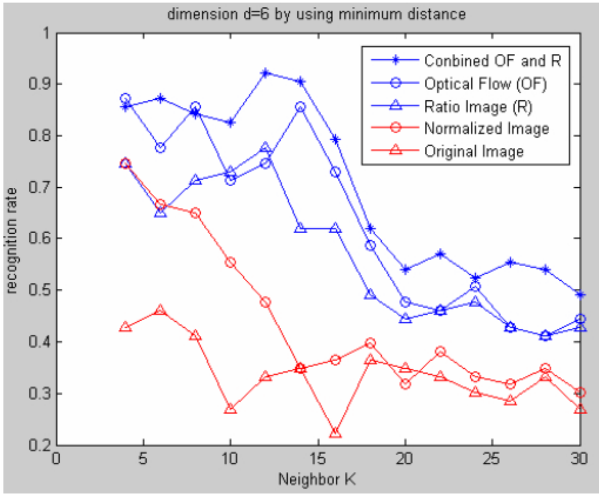

Fig. 6. Comparison of different expression recognition methods of varying neighbor sizes by using the minimum class-mean distance classifier.

Figure 6 shows the recognition rate with different K nearest neighbors when dimension d = 6. From the recognition results, we can see using the optical flow and/or ratio image are significantly better than those based on the normalized image or original image. This means that optical flow and ratio image are more representative features for facial expression recognition. If more neighbors are used, the neighbors assigned to each sample may be from different classes. Thus, the SLLE analysis will be affected by samples from different classes, thus degrading the recognition rate.

Here, we use k-nearest-neighbor classification to replace the minimum class-mean distance classification for the same experiment, and the results for $k = 20$ are shown in Fig. 7. We can see that the recognition rate by using the minimum class-mean distance classification is better than that by using the k-nearest-neighbor classification.

4.3: Yale database

In the previous section, the experiments work well on the JAFFE database. To test its generalization, we used the Yale Face database for testing the proposed expression recognition system trained from the JAFFE dataset.

In the training stage, we used all 183 images for our training. In the Yale database, it contains 165 images of 15 subjects. There are 11 images per subject, one for each of the 11 facial expressions or configurations. We select the happy, normal, sad and surprised images for our testing expression images. The normal image is used as the reference image. Therefore, we have 45 expression images for testing. In the Yale Face database, some subjects wear glasses or are with a beard. These situations did not appear in the JAFFE database. Therefore, using the JAFFE database for training and Yale Face database for testing is very challenging

In this experiment, we first applied the minimum class-mean distance classifier for classification. Figure 8 shows the result of using the Yale Face database for testing. The best accuracy is only up to 71% accuracy. In addition, we used the k-nearest-neighbor classifier to replace the minimum distance classifier and the recognition result on the Yale Face database is shown in Figure 9. The experimental result has better recognition rate than that by using the minimum class-mean distance classifier. The red line in Figure 9 shows the result of using the minimum class-mean distance classifier, and the blue lines are the results by using the k-nearest-neighbor classifier with different feature combinations. The best recognition rate is about 90% by using the k-NN classifier with optical flow and ratio image features. This accuracy is close to the recognition rate by using the JAFFE database for testing.

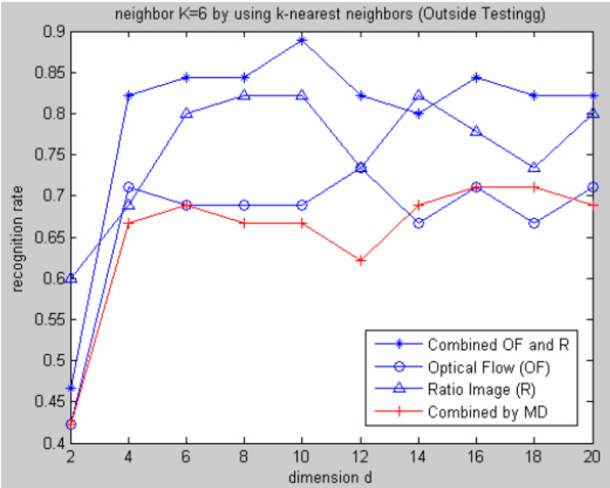

Fig. 8. Comparison of expression recognition rates with varying dimensions by using the minimum class-mean distance classifier on Yale database.

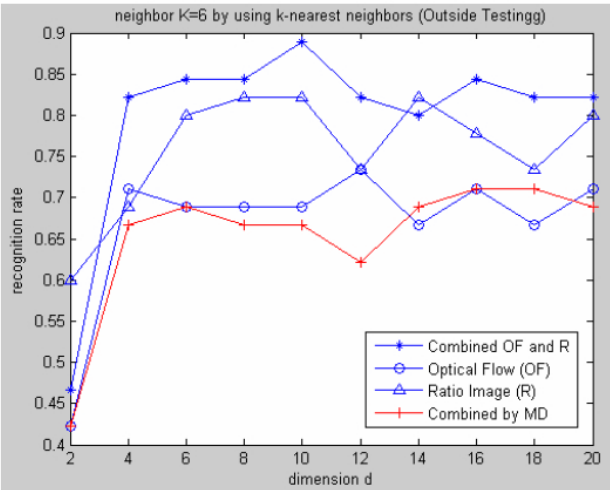

Fig. 9. Comparison of expression recognition rates with varying dimensions by using the k-nearest-neighbor classifier on Yale database.

5: CONCLUSION

In this paper, we proposed a facial recognition algorithm based on computing the optical flow and ratio image from the facial expression image and applying the SLLE for nonlinear dimensionality reduction. The experimental results on the JAFFE database show the proposed facial expression algorithm can provide more accurate recognition rate than previous facial expression methods. We also used the Yale Face database for testing the expression recognition system trained from the JAFFE database, and we can still achieve accurate recognition rate. In the future work, we will try to develop an automatic facial expression system based on the proposed algorithm.

REFERENCES

[1] D. de Ridder, O. Kouropteva, O. Okun, M. Pietik¨ainen and R. P.W. Duin. Supervised locally linear embedding. Proc. Intern. Conf. on Artificial Neural Networks, 2003.

[2] S.T. Roweis and L.K Saul. Nonlinear dimensionality reduction by locally linear embedding. Science 290, 2323–2326, 2000.

[3] D. Liang, J. Yang, Z. Zheng, Y. Chang et al. A facial expression recognition system based on supervised locally linear embedding. Pattern Recognition Letters 26, 2374–2389, 2005.

[4] J. Lien, T. Kanade, J. Cohn, and C. Li. Automated facial expression recognition based on FACS action units, Proc. 2nd Intern. Conf. on Automatic Face and Gesture Recognition, Nara, Japan, 1998.

[5] C.-H. Teng, S.-H. Lai, Y.-S. Chen, W.-H. Hsu. Accurate optical flow computation under non-uniform brightness variations. Computer Vision and Image Understanding, pp. 315-346, 2005.

[6] X. Liu, T. Chen, and B.V.K. V. Kumar, Face authentication for multiple subjects using eigenflow. Pattern Recognition, Vol. 36, pp. 313-328, 2003.

[7] M. J. Lyons, S. Akamatsu, M. Kamachi, and J. Gyoba. Coding facial expressions with Gabor wavelets. Proc. 3rd IEEE Intern. Conf. on Automatic Face and Gesture Recognition, pp. 200-205, 1999.