A NEW SPONTANEOUS EXPRESSION DATABASE AND A STUDY OF CLASSIFICATION-BASED EXPRESSION ANALYSIS METHODS

Segun Aina [1], Mingxi Zhou[1], Jonathon A. Chambers [1], Raphael C.-W. Phan [2]

[1] Advanced Signal Processing Group,
School of Electronic, Electrical and Systems Engineering,
Loughborough University, UK
{S.Aina, J.A.Chambers}@lboro.ac.uk, Mingxizhou10@gmail.com

[2] Centre for Information Security and,
Faculty of Engineering,
Multimedia University, Malaysia
raphael@mmu.edu.my

Abstract– In this paper we introduce a new spontaneous expression database, which is under development as a new open resource for researchers working in expression analysis. It is particularly targeted at providing a wider number of expression classes contained within the small number of natural expression databases currently available so that it can be used as a benchmark for comparative studies. We also present the first comparison between kernel-based Principal Component Analysis (PCA) and Fisher Linear Discriminant Analysis (FLDA), in combination with a Sparse Representation Classifier (SRC), based classifier for expression analysis. We highlight the trade-off between performance and computation time; which are critical parameters in emerging systems which must capture the expression of a human, such as a consumer responding to some promotional material.

Keywords– Fisher's Discriminant Analysis, Kernel, Principal Component, Sparsity, Spontaneous Expression Classification.

1. INTRODUCTION

The ability to use algorithms to classify features from facial expressions can potentially allow the emotions that stimulated the expressions to be inferred, giving rise to a range of potential applications. We propose an automated system for advertisers who may wish to identify the expressions of consumers in order to estimate their affective responses and reactions to advertising campaigns. To achieve this, the system should be able to accurately recognise natural expressions of consumer's faces.

There are basic methods for extracting usable facial features, two of which are Principal Component Analysis (PCA, eigen-faces) and Fisher Linear Discriminant Analysis (FLDA, fisher-faces) [1]. The PCA method projects the face images into the space which is composed of principal components, and the reconstructed images are called eigen-faces [2]. On the other hand, the FLDA automatically selects the optimal features for classification by using the theories of linear discriminant projection [3].

Some studies have investigated the possibility of combining the PCA and FLDA methods with a neural network [4] and Support Vector Machine (SVM) [5]. Other research also suggests that applying the kernel method will make it easier to linearly separate each class [6]–[8]. Apart from the choice of classification method, the careful selection of features may also bring significant improvement of the overall performance [8]. Researchers have applied the recently developed approach called Sparse Representation-based Classification (SRC) or sparsity to Facial Expression Recognition (FER) [9]. The SRC approach employs each training image as an atom to represent the test image, and uses the sparse representation to perform classification [10]. Practically, the types of image database (spontaneous or posed expression database) used to conduct training will also affect the effectiveness of methods [11–13].

1.1. Brief Review of Existing Databases

Most of the facial expression databases in use today are posed expression databases, such as the Cohn-Kanade (CK) database, JAFFE database [14] and the MUG database [13] (being used for a comparison in this study). The MUG database contains 630 images, posed by more than 30 participants, comprising the seven prototypical facial expression classes: Anger (AN), Disgust (DI), Fear (FE), Happiness (HA), Neutral (NE), Sadness (SA) and Surprise (SU) [13]. Posed databases are limited due to the unrealism of the often-exaggerated expressions contained therein; as such, systems developed on this basis may not perform well in practical applications.

Because of the difficulty in stimulating certain types of expression and labelling them, there are only a few spontaneous facial expression databases available at this moment, such as the NVIE database [11] and the DISFA database [12]. Where others exist, they have not been made available to other researchers in the field. There are mainly three different ways to stimulate facial expressions: human-human conversation, human-computer interaction, or emotion-inducing videos [11], [15]. Since certain expressions are harder to stimulate and capture (such as AN, FE and SA), a typical spontaneous facial expression database would have *around three expression classes* from approximately 50 - 80 subjects which is one of the pitfalls we will try to address.

This paper introduces a new natural database, which will be a publicly available benchmark dataset for research in spontaneous facial expression analysis. We also provide experimentally verified analyses of classification-based methods along with assertions on eigen-vector selection, notably the first known comparison between kernel-based PCA and FLDA combined with SRC for expression recognition. The remainder of the paper is arranged as follows: Section 2 will introduce our new spontaneous expression database. Section 3 will describe the feature extraction methods employed in the review. Section 4 will discuss classifiers including sparse representation. Performance is analysed in Section 5 and Section 6 contains conclusions and suggestions for future work.

2. NEW SPONTANEOUS EXPRESSION DATABASE

Other potential applications of expression recognition (e.g. real-time assessment of marketing on smart devices) are emerging at a slow pace. This is partly as a result of a limited number of accessible natural databases. Where accessible, there are a limited number of subjects and classes. We aim to develop what will be a publicly available database with sufficiently larger number of subjects, more expression classes and more diverse than currently exist.

The prototype of our spontaneous facial expression database contains 134 images collected under normalised conditions from 50 subjects from the HA, DI, NE, SA and SU classes. However only images from DI, HA, SA and SU were used in the evaluations in this paper. A clip of various emotion-inducing videos, which lasts around 19 minutes, is being shown to each voluntary participant under the same conditions in order to stimulate their facial expressions. Prior to the experiment, volunteers are told that they are participating in a memory/observation test in order to make sure that the expressions are truly natural and as such, the volunteers had no idea that they were being recording until the end of the experiment.

The experiment was setup in the multimedia room of the department. For capture, we used a mini High-Definition (HD) camera with the record lighting concealed, placed on top of the screen to record video of the volunteers while they watched the clip. Prior to the experiment, participants were asked to sign a 'consent to participate' form, which contained details of the fictitious experiment. At the end of the video, following the disclosure of the true purpose of the experiment (at which point recordings continued to capture reactions), the volunteers are also asked to pose the six classes of expression in front of the camera. To conclude, each participant was asked to sign a 'release form'.

The raw video data are then transformed into a useable database of images by cutting the video into short video clips, typically lasting 2-3 seconds; each one of which contains the evolution of an observed facial expression from the onset through the apex to the dissipation. The short video clips are converted to a sequence of JPEG images. The 'peak' expression image, which has the most intense expression in each sequence, is chosen to build up the database. The remaining images in the sequences are also saved for further study.

The labelling and pre-processing (cropping and centring) are performed manually. A judging panel of three people is being used to re-examine the labelling of the images in classes AN, FE and SA, due to the higher chance of confusion between these classes.

3. FEATURE EXTRACTION METHODS

3.1. Principal Component Analysis (PCA)

PCA characterizes objects to be measurable by machines and reduces the dimensionality of objects by combining features. Effectively, PCA chooses a linear projection that maximizes the scatter matrix [1]. To achieve that, a scatter or covariance matrix is formed by: $\boldsymbol{S} = \sum_{k=1}^{N}(\boldsymbol{X}_k - \boldsymbol{\mu})(\boldsymbol{X}_k - \boldsymbol{\mu})^T$. Given that $\boldsymbol{X}_k$ is the column vector created by reshaping each original image, $\boldsymbol{\mu}$ is the mean of all $\boldsymbol{X}_k$, N is the total number

Figure 1: Images from the new spontaneous database (top row) and images from the posed MUG database (bottom row).

of training samples, and $(\cdot)^T$ denotes the transpose operation. Applying linear projection $\boldsymbol{W}^T$, the scatter of the projected feature vector $\{\boldsymbol{e}_1, \boldsymbol{e}_2, \dots, \boldsymbol{e}_N\}$ is $\boldsymbol{W}^T\boldsymbol{S}\boldsymbol{W}$. For PCA the projection matrix $\boldsymbol{W}_{opt}$ is chosen to maximize the determinant of the scatter of the projected feature vector:

$$\boldsymbol{W}_{opt} = arg\max_{\boldsymbol{W}}|\boldsymbol{W}^T\boldsymbol{S}\boldsymbol{W}| \tag{1}$$

3.2. Fisher Linear Discriminant Analysis (FLDA)

PCA provides a basis for feature extraction by performing dimensionality reduction with a linearly selected projection matrix. However, the dimensional reduced images are not always perfectly linearly separable. FLDA was introduced, this transforms the original database into a dimensional reduced space where the differences between classes are maximised and the differences within each class are minimised. To achieve that, let the between-class scatter matrix be: $\boldsymbol{S}_B = \sum_{i=1}^{c} N_i(\boldsymbol{\mu}_i - \boldsymbol{\mu})(\boldsymbol{\mu}_i - \boldsymbol{\mu})^T$ and the within-class scatter matrix be defined as: $\boldsymbol{S}_W = \sum_{i=1}^{c}\sum_{\boldsymbol{X}_k \in \boldsymbol{X}_i}(\boldsymbol{X}_k - \boldsymbol{\mu}_i)(\boldsymbol{X}_k - \boldsymbol{\mu}_i)^T$ where $\boldsymbol{\mu}_i$ is the mean of class $\boldsymbol{X}_i$, $\boldsymbol{\mu}$ is the mean of the entire training set, and N_i is the number of samples in class $\boldsymbol{X}_i$. The optimal projection matrix is then:

$$\boldsymbol{W}_{opt} = arg\max_{\boldsymbol{W}}\frac{|\boldsymbol{W}^T\boldsymbol{S}_B\boldsymbol{W}|}{|\boldsymbol{W}^T\boldsymbol{S}_W\boldsymbol{W}|} = [\boldsymbol{w}_1, \boldsymbol{w}_2, \dots, \boldsymbol{w}_m] \tag{2}$$

where $\{\boldsymbol{w}_i|\ i = 1,2, \dots, m\}$ is the set of generalized eigenvectors of $\boldsymbol{S}_B$ and $\boldsymbol{S}_W$ corresponds to the m largest eigenvalues $\{\lambda_i|\ i = 1,2, \dots, m\}$.

To practically obtain the optimal FLDA projection matrix, PCA is first applied; as such, the FLDA solution is given by:

$$\boldsymbol{W}_{opt} = \boldsymbol{W}_{pca}\ \boldsymbol{W}_{flda} \tag{3}$$

where $\boldsymbol{W}_{pca}$ is given by equation (1) and $\boldsymbol{W}_{flda}$ is given by:

$$\boldsymbol{W}_{flda} = arg\max_{\boldsymbol{W}}\frac{|\boldsymbol{W}^T\boldsymbol{W}_{pca}^T\boldsymbol{S}_B\boldsymbol{W}_{pca}\boldsymbol{W}|}{|\boldsymbol{W}^T\boldsymbol{W}_{pca}^T\boldsymbol{S}_W\boldsymbol{W}_{pca}\boldsymbol{W}|} \tag{4}$$

3.3. Kernel Principal Component Analysis (KPCA)

In general, KPCA maps data into a higher dimensional space $\boldsymbol{R}^F$, where the data could be made more linearly separable [16], which enables the extraction of nonlinear features, thereby achieving better classification performance [17], [18]. The nonlinear mapping function for this projection is given by $\Phi{:}\boldsymbol{R}^M \to \boldsymbol{R}^F$ $M \le F$ where $\boldsymbol{R}^M$ is the space in which the data originally laid. In space $\boldsymbol{R}^F$, the covariance matrix of $\Phi(\boldsymbol{X})$ is then given by: $\boldsymbol{S}^{\Phi} = \sum_{k=1}^{N}\Phi(\boldsymbol{X}_k)\ \Phi(\boldsymbol{X}_k)^T$, where $\Phi(\boldsymbol{X}_k)$ is the projection of image $\boldsymbol{X}_k$ in feature space $\boldsymbol{R}^F$. Notice that the dimensionality of the feature space $\boldsymbol{R}^F$ can be arbitrarily large [16], [17].

The eigen-decomposition problem of $\boldsymbol{S}^{\Phi}$ can be simplified as: $\boldsymbol{KX\alpha} = \mathrm{N}\lambda\boldsymbol{\alpha}$, and the projection matrix is given by: $\boldsymbol{W}^{\Phi} = \sum_{i=1}^{N} \alpha_i \Phi(\boldsymbol{X}_i)$ where $\alpha_i, \ldots, \alpha_N$ are the elements of vector $\boldsymbol{\alpha}$. With the projection matrix $\boldsymbol{W}^{\Phi}$, we can then project the images in $\boldsymbol{R}^F$ to a dimensionally reduced space. Assuming that $\boldsymbol{X}_k$ is a sample image, its projection via the kernel method is then given by:

$$\boldsymbol{W}^{\Phi}\Phi(\boldsymbol{X}_k) = \sum_{i=1}^{N} \boldsymbol{\alpha}_i \Phi(\boldsymbol{X}_i)\Phi(\boldsymbol{X}_k) = \sum_{i=1}^{N} \boldsymbol{\alpha}_i k(\boldsymbol{X}_i, \boldsymbol{X}_k) \quad (5)$$

Note that whether or not $\boldsymbol{X}$ is mean-centred in its original space, there is no guarantee that it is zero mean in space $\boldsymbol{R}^F$. Therefore, the mean-centred version of the kernel matrix is given as: $\boldsymbol{Kx}' = \boldsymbol{Kx} - 1_N\boldsymbol{Kx} - \boldsymbol{Kx}1_N + 1_N\boldsymbol{Kx}1_N$ where 1_N is an $N \times N$ matrix whose elements take the value of $1/N$.

3.4. Kernel Fisher Discriminant Analysis (KFDA)

The FLDA method solves Fisher's discriminant problem in the linear space, while the KFDA solves it in nonlinear feature space $\boldsymbol{R}^F$. Effectively, because the extra nonlinear features can easily be extracted in space $\boldsymbol{R}^F$, applying the kernel variant produces better results. Similar to the process of the fisherface method, all the scatter matrices of the KFDA will have KPCA pre-applied for dimensional reduction [7]. Let $\boldsymbol{S}_B^{\Phi}$ denote the between-class scatter matrix and $\boldsymbol{S}_W^{\Phi}$ denote the within-class scatter matrix, which is defined as the fisher's discriminant function in feature space $\boldsymbol{R}^F$ and is given by:

$$\boldsymbol{W}_{opt}^{\Phi} = arg\max_{\boldsymbol{W}} \frac{|\boldsymbol{W}^T\boldsymbol{S}_B^{\Phi}\boldsymbol{W}|}{|\boldsymbol{W}^T\boldsymbol{S}_W^{\Phi}\boldsymbol{W}|} = [\boldsymbol{w}_1^{\Phi}, \boldsymbol{w}_2^{\Phi}, \ldots, \boldsymbol{w}_m^{\Phi}] \quad (6)$$

where $\boldsymbol{S}_B^{\Phi} = \sum_{j=1}^{c} l_j(\boldsymbol{M}_j - \boldsymbol{M}_0)(\boldsymbol{M}_j - \boldsymbol{M}_0)^T$ and $\boldsymbol{S}_W^{\Phi} = \sum_{j=1}^{c} \boldsymbol{K}_j(\boldsymbol{I} - \boldsymbol{1}_{lj})\boldsymbol{K}_j{}^T$, where $(\boldsymbol{K}_j)_{nm} = k(\boldsymbol{X}_n, \boldsymbol{X}_m^j)$. The number of images in class $\boldsymbol{X}_i$, is l_j. The mean of the entire training set is $(\boldsymbol{M}_0)_i = \frac{1}{N}\sum_{k=1}^{N} k(\boldsymbol{x}_i, \boldsymbol{x}_k)$ and $(\boldsymbol{M}_j)_i = \frac{1}{1_j}\sum_{k=1}^{1_j} k(\boldsymbol{X}_i, \boldsymbol{X}_k^j)$ is the mean of class $\boldsymbol{X}_i$.

4. CLASSIFICATION METHOD

Again, as already well covered in literature, but for completeness, the Nearest Neighbour (NN) classifier is often used because of its simplicity and robustness. NN uses a Euclidean distance measure between the projections of the test sample and the training samples to judge their correlation. The test image projection is said to belong to the same expression class as the projection of training image for the lowest value of the Euclidean distance given by:

$$d(samp, training) = \sqrt{\sum_{i=1}^{n}(s_i - t_i)^2} = \|\boldsymbol{s} - \boldsymbol{t}\|_2 \quad (7)$$

4.1. Sparsity Representation

For SRC, let $\boldsymbol{a}_i = [\boldsymbol{x}_1^i, \boldsymbol{x}_2^i, \ldots, \boldsymbol{x}_{N_i}^i] \in \boldsymbol{R}^{m*N_i}$ be the sub-database of class i if the size of class i is large enough, any test sample $\boldsymbol{y}$ belonging to class i will approximately lie in the linear span of the training associated with class i [9]:

$$\boldsymbol{y} = \boldsymbol{\alpha}_1^i\boldsymbol{x}_1^i + \boldsymbol{\alpha}_2^i\boldsymbol{x}_2^i + \cdots + \boldsymbol{\alpha}_{N_i}^i\boldsymbol{x}_{N_i}^i \quad (8)$$

where, $\alpha_j^i \in \boldsymbol{R}, j = 1,2, \ldots, N_i$. The dictionary $\boldsymbol{A}$ can be formed by concatenating all training samples together as $\mathbf{A} = [\mathbf{A}_1, \mathbf{A}_2, \ldots, \mathbf{A}_c] \in \mathbf{R}^{\mathrm{M*N*c}}$.

For SRC to be fully expoited, the database needs to be "overcomplete", which means the dimensionality (M) of the dictionary needs to be larger or at least equal to the total number of training samples (N). This can be further achieved either by artificially expanding the dictionary (combining two different training samples to produce a new one) or applying dimensionality reduction methods e.g. PCA [10],[19]. With an overcomplete dictionary, the linear representation of $\boldsymbol{y}$ can be rewritten in terms of the dictionary as:

$$\boldsymbol{y} = \boldsymbol{Ax}_0 \in \boldsymbol{R}^M \quad (9)$$

where $\boldsymbol{x}_0 = [0, \ldots, 0, \alpha_{i,1}, \alpha_{i,2}, \ldots, \alpha_{i,n_i}, 0, \ldots, 0]^T \in \boldsymbol{R}^M$ is a coefficient vector whose elements are zero except for those corresponding to the i^{th} class. As in the literature, the solution of the linear representation of $\mathbf{y}$ can be simplified as the following ℓ^1 problem [9], [19]:

$$(\ell^1): \quad \hat{\boldsymbol{x}}_1 = \arg\min\|\boldsymbol{x}\|_1 \quad \text{subject to } \boldsymbol{Ax} = \boldsymbol{y}. \quad (10)$$

The ℓ^1 problem can be solved by applying a standard linear programing method [9][20].

4.2. Recognition Based on Sparse Vector

The classification is based on finding the closest approximation of test sample $\boldsymbol{y}$ by the sparse vectors corresponding to the class i. $\delta_i(\hat{\boldsymbol{x}}_1)$ is a characteristic function which selects coefficients related to the i-th class. Using the sparse vectors of class i, we can approximate $\boldsymbol{y}$ as $\hat{\boldsymbol{y}}_1 = \boldsymbol{A}\delta_i(\hat{\boldsymbol{x}}_1)$. Test sample $\boldsymbol{y}$ is therefore classified based on the class, which minimises the residual between $\boldsymbol{y}$ and $\hat{\boldsymbol{y}}_1$ obtained by:

$$l(\boldsymbol{y}) = \underset{i}{\mathrm{argmin}}\|\boldsymbol{y} - \boldsymbol{A}\delta_i(\hat{\boldsymbol{x}}_1)\|_2 \quad (11)$$

5. COMPARATIVE EVALUATION

5.1. Performance Analysis

The Leave-One-Out (LOO) cross validation is applied to examine the performance of all the aforementioned methods. All program codes were run on a HP laptop: OS: Win 7 64-bit, CPU: i5-3317U 1.70 GHz, RAM: 4GB, using MATLAB.

On the MUG database, our experimental results show that the KFDA + NN classifier achieves the lowest error rate of 7.78%, due to its ability to extract most discriminant (nonlinear) features. However, it took around two hours to complete with cross validation, while the PCA+ SRC only took around eight minutes and provided the second lowest error rate of 8.73% which is better than the recognition error of 10% obtained in [19]. The detailed performance and error rates are shown in Table 1. Figure 2 shows the average error rate and total amount of time taken for different method combinations. It can be seen that after applying the Kernel variants of PCA and FLDA, there is reduction of about 1% to their error rates. However, the price for the lower error rates comes at a disproportionally greater computational time.

Error Rate %	AN	DI	FE	HA	NE	SA	SU	Ave
PCA+NN	14.44	7.78	17.78	10	11.11	8.89	12.22	**11.75**
KPCA+NN	14.44	6.67	14.44	10	11.11	8.89	10	**10.8**
FLDA+NN	12.22	2.22	14.44	3.33	8.89	8.89	15.56	**9.21**
KPCA+SRC	8.89	2.22	20	5.56	11.11	6.67	8.89	**9.05**
KFDA+SRC	14.4	3.33	16.67	5.56	6.67	2.22	13.33	**8.89**
FLDA+SRC	12.22	2.22	16.67	5.56	8.89	6.67	8.89	**8.73**
PCA+SRC	8.89	3.33	16.67	7.78	11.11	6.67	6.67	**8.73**
KFDA+NN	8.89	1.11	14.44	4.44	8.89	4.44	12.22	**7.78**

Table 1: Detailed performance % error rates of the various method combinations on the MUG database.

We can also see that, generally, the SRC classifier achieves better results than the NN classifier and the running time did not increase substantially (taking even less time than NN when it was combined with eigen-face), although note that the SRC wasn't very effective when combined with the KPCA and KFDA methods. This may be due to the extreme nonlinearity of the extracted features, which makes it hard to satisfy the equality constraint $\boldsymbol{Ax} = \boldsymbol{y}$, especially for the KFDA method, which achieved the best performance with the NN classifier.

We also tested our new spontaneous database, and due to the current (smaller) size and the consequent inability to form the overcomplete dictionary required for SRC, we only tested it with the basic feature extraction and classification methods (eigen-face + NN, fisher-face + NN).

PCA + NN					FLDA + NN				
%	DI	HA	SA	SU	%	DI	HA	SA	SU
DI	**15**	35	40	20	DI	**55**	10	30	5
HA	25	**30**	25	20	HA	30	**70**	0	0
SA	40	5	**25**	30	SA	40	5	**45**	10
SU	5	10	45	**40**	SU	20	0	40	**40**
eigen-vectors	Nos. 6 – 10				eigen-vectors	PCA: Nos. 2 – 40 FLDA: Nos. 1 -5			
Av. Acc.	**27.5%**				Av. Acc.	**52.5%**			

Table 2: Confusion matrices for classification results on the *new database*.

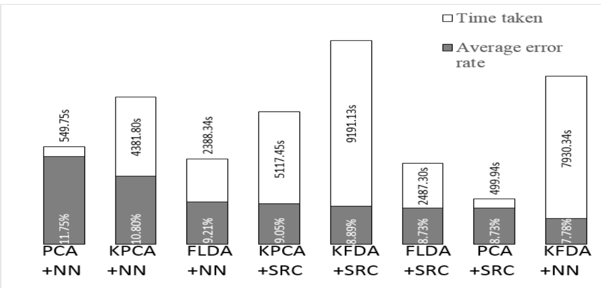

Figure 2: Error performance and running times for the various method combinations on the MUG database.

The confusion matrix in accuracy percentage is shown in Table 2. The result shows consistency in that the fisher-face method has provided greater accuracy compared with the eigen-face method. These poorer results confirm how important it is to have a rich spontaneous database to gain a fair indication of the practical performance of methods.

5.2. Eigen-feature Selection

The approaches discussed above include the process of first applying PCA or KPCA. It has been proven in several studies that different eigenvectors may carry information that are relevant to certain facial properties [8], [21]. Figure 4 demonstrates that the Eigen-feature selection, which is often performed manually, influences results. Although the eigen

Figure 3: Examples of eigenvectors NOs.1-5 in the order of eigenvalues

vectors shown in Figure 3 can be very significant for recognition, they are often not relevant to the facial expression classification. For example, the first eigenvector shows the common background. The following few eigenvectors reflect the lighting conditions and gender information. We experimentally discovered that discarding the first few eigenvectors produces better results in the case of natural expression recognition evident by the lower error rates.

The approaches we discussed above are selecting eigen-features manually while the feature extraction methods such as PCA, KPCA, FLDA and KFDA performed automatic eigen-feature selection. The FDA is applied in both FLDA and KFDA, which theoretically should produce the optimal solution for classification, and generally, the FDA would need a comparatively greater number of eigen-features to select the optimal ones. However, if we also feed the first few principal eigenvectors into the FDA, both FLDA and KFDA would produce inferior results. It is likely because those eigenvectors contain the common features in the face, which exist both within classes and between classes, this causes the FDA to produce less optimal output. This also reflected the conclusion we draw above that discarding the first few eigenvectors in the order of eigenvalues would increase the total performance.

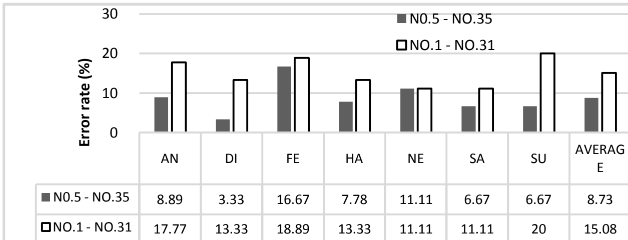

Figure 4: Results from PCA+SRC using different sets of eigenvectors

6. CONCLUSION

Through the comparison presented in this paper, it is clear that the combination of eigen-face plus SRC method is superior to any other method. On the MUG database, the eigen-face plus SRC method can easily reach the average error rate of 8.73% which is the second lowest among all methods, and it also is the fastest algorithm which only took 500 seconds to complete the LOO cross validation.

Meanwhile, the trade-off between the accuracy and the computational complexity can be very expensive. To be specific, although KFDA with NN classifier method has the lowest recorded average error rate of 7.78%, it took 7930 seconds to complete the LOO cross validation, which indicates a considerably larger computational complexity. On the other side, the KFDA with SRC classifier is also a time consuming method, which took 2.5 hours to run. Therefore, regardless of the low error rate the KFDA provides, it may not be the best choice of practical feature extraction method.

Generally, by changing the classifier from NN to SRC, most of the combinations achieved better performance. On average, all combinations acquired more than 1% of improvement on recognition accuracy. This demonstrates the excellence of sparsity compared with regular approaches.

The new spontaneous facial expression database being measured has already shown great potential in the spontaneity of the expressions being collected, the number of expression classes being recorded, the diversity of the subjects and in providing data for future study such as understanding the evolution of certain types of expressions, other analyses of natural facial expressions or prediction of affective states. Once complete, our new database will serve to mitigate the current shortfall in freely available *natural* training data for developers of FER systems. It will have more classes from more diverse (age, race, gender) than any other available spontaneous database.

The above all serve as motivation for the continued expansion and development of the spontaneous facial expression database.

7. REFERENCES

[1] J. Ruiz-del-Solar and P. Navarrete, “Eigenspace-based face recognition: a comparative study of different approaches,” *Syst. Man, Cybern. Part C Appl. Rev. IEEE Trans.*, vol. 35, no. 3, pp. 315–325, 2005.

[2] M. Turk, “Eigenfaces for Recognition,” *J. Cogn. Neurosci.*, vol. 3, no. 1, pp. 71–86, Jan. 1991.

[3] P. Belhumeur and J. Hespanha, “Eigenfaces vs. fisherfaces: Recognition using class specific linear projection,” *Pattern Anal. Mach. Intell. IEEE Trans.*, vol. 19, no. 7, pp. 711–720, 1997.

[4] H. Sahoolizadeh and Y. Aliyari Ghassabeh, “Face recognition using eigen-faces, fisher-faces and neural networks,” *2008 7th IEEE Int. Conf. Cybern. Intell. Syst.*, pp. 1–6, Sep. 2008.

[5] L. Wu, P. Wu, F. Meng, and W. Yu, “Study on face recognition with combined of fisher algorithm and support vector machine,” *2nd Int. Conf. Inf. Sci. Eng.*, pp. 5444–5447, Dec. 2010.

[6] S. Mika and G. Ratsch, “Fisher discriminant analysis with kernels,” *Neural Networks Signal Process. IX, 1999. Proc. 1999 IEEE Signal Process. Soc. Work.*, pp. 41–48, Jul. 1999.

[7] J. Yang, A. F. Frangi, J.-Y. Yang, D. Zhang, and Z. Jin, “KPCA plus LDA: a complete kernel Fisher discriminant framework for feature extraction and recognition.,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 2, pp. 230–44, Feb. 2005.

[8] G. Bebis and S. J. Louis, “Genetic feature subset selection for gender classification: a comparison study,” *Sixth IEEE Work. Appl. Comput. Vision, 2002. (WACV 2002). Proceedings.*, pp. 165–170.

[9] J. Wright and A. Yang, “Robust face recognition via sparse representation,” *Pattern Anal. Mach. Intell.*, vol. 31, no. 2, pp. 210–227, 2009.

[10] M. Aharon, M. Elad, and A. Bruckstein, “K - SVD□: An Algorithm for Designing Overcomplete Dictionaries for Sparse Representation,” *IEEE Trans. SIGNAL Process.*, vol. 54, no. 11, pp. 4311–4322, 2006.

[11] S. Wang, Z. Liu, S. Lv, Y. Lv, G. Wu, P. Peng, F. Chen, and X. Wang, “A Natural Visible and Infrared Facial Expression Database for Expression Recognition and Emotion Inference,” *IEEE Trans. Multimed.*, vol. 12, no. 7, pp. 682–691, Nov. 2010.

[12] P. Trinh and J. Cohn, “DISFA: A Spontaneous Facial Action Intensity Database,” *IEEE Trans. Affect. Comput.*, vol. 4, no. 2, pp. 151–160, 2013.

[13] A. Aifanti, N., Papachristou, C., & Delopoulos, “The MUG Facial Expression Database,” *Image Anal. Multimed. Interact. Serv.*, pp. pp. 1–4, 2010.

[14] Z. Zeng, M. Pantic, G. I. Roisman, and T. S. Huang, “A survey of affect recognition methods: audio, visual, and spontaneous expressions.,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 1, pp. 39–58, Jan. 2009.

[15] J. Coan and J. Allen, *Handbook of Emotion Elicitation and Assessment.* Oxford University Press, 2007.

[16] B. Schölkopf, A. Smola, and K. Müller, “Nonlinear component analysis as a kernel eigenvalue problem,” *Neural Comput.*, vol. 1319, pp. 1299–1319, 1998.

[17] M. Yang, N. Ahuja, and D. Kriegman, “Face recognition using kernel eigenfaces,” *Image Process. 2000. Proc. Int. Conf.*, vol. vol. 1, pp. pp. 37–40, 2000.

[18] M. H. Yang, “Kernel Eigenfaces vs. Kernel Fisherfaces: Face recognition Using Kernel Methods,” *Proc. Fifth IEEE Int’l Conf. Autom. Face Gesture Recognit.*, pp. 215 – 220, 2002.

[19] S. Zafeiriou and M. Petrou, “Sparse Representations for Facial Expressions Recognition via l1 Optimization,” *2010 IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. - Work.*, pp. 32–39, Jun. 2010.

[20] S. Cotter, “Sparse representation for accurate classification of corrupted and occluded facial expressions,” *Int. Conf. Acoust. Speech Signal Process.*, pp. 838–841, 2010.

[21] B. Draper, “Analyzing pca-based face recognition algorithms: Eigenvector selection and distance measures,” *Empir. Eval. Methods Comput. Vis.*, pp. 1–14, 2002.