Facial Image Recognition by Subspace Learning: A Comparative Study

Yongbin Zhang, Lixin Lang and Onur Hamsici

1 Introduction

Facial image analysis and recognition has been gaining great attentions and witnessing significant advances during recent years. Ranging from static photo matching in different kinds of authorized licenses to dynamic analysis of surveillance video images, its successful application is not only due to commercialization and law enforcement, but also a result of the availability of reliable methodologies. In this report, feature-based and appearance-based, will be introduced with our concentration on the latter. Public face databases, FERET and JAFFE face database are used to analyze several subspace methods in face identity recognition as well as facial expression analysis, which includes PCA, ICA, LDA and their kernel versions.

1.1 Feature-based Approach

Feature-based recognition methods denote those approaches that are built on visual face characteristics, to name a few geometrical measurements of mouth extremities distance and chin shape [10, 7]. Based on these features, a model can be obtained to present faces, to which the face features in tested images would be matched to obtain their identity. Although psychologically sounding and insensitive to illumination and view-point variation, it severely suffers from an unreliable measurement and extraction of face features.

1.2 Appearance-based Approach

Unlike feature-based method, appearance-based technology directly uses the high dimensional image intensity feature(or its filter bank output). It is attractive in that the model of each class is directly defined by the selection of

the sample images of that object, without the need to create precise geometrical or algebraic representations. Significant successful applications especially in face recognition area in recent years can be found in [27, 28, 3, 12].

Generally speaking, a training process is designed to find a mapping function which projects the image data to a lower dimension space (subspace), which makes it possible to compare and match tested faces to the trained ones. The reason will be made clear in the following.

As a result of using original image space, the length of feature vector equals the number of pixels in one image, which is in general very large (e.g. 4800 in our case). To make it worse, the curse of dimensionality is extremely severe for classification problem. In the pattern recognition community, it is quite a common practice to use a minimum number of independent samples equal to ten times the number of classes(ie, face identity number) by the number of feature dimensions. To this end, efficient feature extraction methods are required to find a low-dimensional subspace that is capable of explaining the original data. Linear subspace methods are extensively used in appearance-based recognition, among which Principal Component Analysis (PCA) [27, 28], Independent Component Analysis (ICA) [2] and Linear Discriminate Analysis (LDA) [3, 12] are the three most popular ones. However, all these subspace techniques are linear in nature, which is inadequate to describe nonlinearities inherent in the data. Recently, kernel sub-space methods, such as Kernel-PCA (KPCA), Kernel ICA (KICA) and Kernel LDA (KLDA), receive more attentions in pattern recognition community in that they are capable of extracting nonlinear features by mapping input space to some higher dimensional spaces. By using so-call "kernel trick", the nonlinear subspace can be implicitly obtained without the needs to explicitly compute the mapping function. Kernel methods find wide applications in clustering [8], density estimation [15] and image de-noising [22].

With the availability of above-mentioned subspace methods, it is natural to compare their pros and cons, as is one of the purposes of this report. Interested readers are encouraged to refer to other resources [10, 18, 19, 29]. Draper et al[10] compares PCA and ICA in the context of a baseline face recognition system, and showed that their performance are depending on the task, ICA algorithm and subspace metric, and comparable overall. In [29], the authors showed that KPCA outperforms PCA in face recognition. Liu et al [18] compared KLDA and KPCA in face recognition and demonstrated that the KLDA achieves higher recognition rate than KPCA. In [19], KPCA is combined with Gabor-based representations for face recognition. In this contribution, the KPCA is extended to include fractional power polynomial

kernel [1].

1.3 Facial Expression Analysis

Though subspace methods have been largely used in facial identity recognition, there is relatively less literature in subspace-based facial expression analysis. Facial expression recognition presents a natural and challenging way for human-machine interaction(HCI), and this area has been receiving more and more attentions. In general, facial expression can be identified either through face muscle's motions that are extracted from image sequence or with the aid of facial feature detection in a single image. For the latter methods, previous work included, while are not limited to, principle component analysis as shown in [23] and Gabor wavelets coding in [20], which used the same face image database as to be used in our study. In [20], a multi-orientation, multi-resolution set of Gabor filters were used to code each image. Based on this representation, similarity space method was declared to be feasible in rating each face image by the corresponding expression categories.

1.4 Overview of This report

we plan to give a comprehensive study on different subspace methods for identity recognition and facial expression recognition. More specifically, we will focus on studying the performance of PCA, ICA, LDA as well as their kernel versions. To this end, we will experiment them on FERET database for facial identity recognition under pose variations, and on JAFFE database for facial expression recognition. The rest of this report is organized as follows: In section 2 and section 3, we give a review on subspace methods, especially focusing on "kernel" versions. Facial identity and expression problems will be discussed in section 4. Description of the databases and experimental results will be presented in section 5, and Concluding remarks will be given in the final section.

2 Linear Subspace Analysis

PCA, ICA and LDA are three of the most popular linear subspace methods and have been largely used in face recognition applications [27, 28, 3, 12, 2].

[1] A fractional power polynomial kernel is defined by $k(x, y) = (1+ < x, y >)^d$, where $0 < d < 1$.

2.1 Principal Components Analysis

PCA finds the optimal liner projection between the original feature space and the low-dimensional subspace with respect to the mean-square error of the Gaussian that represents the data. To accomplish this, PCA uses the first and central moments of the data; i.e., the sample mean μ and the sample covariance matrix Σ_X. Let $X = \{x_1, x_2, \cdots, x_n\}$ represent n images of size m, the sample covariance matrix can be computed by $\Sigma_X = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{\mu}) * (x_i - \bar{\mu})^T$, where $\bar{\mu} = \frac{1}{n}\sum_{i=1}^{n} x_i$. Given Σ_X, the PCA subspace basis can be constructed by selecting its first k(usually $k \ll m$) leading eigenvectors, more formally:

$$\Sigma_X \Phi = \Lambda \Phi \tag{1}$$

The PCA basis is then given by the first k column of Φ.

2.2 Independent Components Analysis

Compared to PCA which de-correlates input data through second-order statistics (covariance), ICA achieves higher-order independence and thus further reduce redundancies. Basically, it assumes that the input data can be decomposed into several statistically independent components with different weights, and estimates both the components and weights based on the above assumption. In face recognition area, it can be applied in two different schemes regarding to what kind of input data are used [2, 11]. In the first scheme, the input data are face image basis obtained from PCA. Based on the image data, ICA algorithm learns the weights and further recovers independent basis images. In contrast to scheme 1, scheme 2 uses the PCA subspace coefficients and gives independent coefficients for image basis instead of independent image basis. As one of the advantages, the image basis contains more global characteristics and thus allows high-order combinations of features.

ICA does not have a general close-form solution, but iterative methods are available. In our experimental results we use the *Infomax* algorithm defined in [4]. Based on the observation that independence is maximized if the entropy is maximized, *Infomax* thus performs gradient ascent search to find a local optimal de-mixing matrix W:

$$W^* = argmax_w H(u) \tag{2}$$

where $H(u) = -\int f(u) log(f(u)) du$, and $u = W^* X$.

2.3 Linear Discriminant Analysis

Different from PCA and ICA that are unsupervised learning algorithms, LDA is a supervised method. LDA selects those basis vectors that maximize the between-class scatter and minimize the within-class scatter [13]. This can facilitate the task of feature extraction in some applications. In [3], improved recognition results over PCA were obtained on Yale face database with illumination variations. Let's assume that the training set X(as defined above) contains l classes, and each class has n_i samples($\sum_{i=1}^{l} n_i = n$). LDA basis w is given by maximizing the following function:

$$J(w) = \frac{w^T S_b w}{w^T S_w w} \tag{3}$$

where S_b and S_w are between-class and within-class scatter matrix respectively:

$$S_b = \frac{\sum_{k=1}^{l} \sum_{j=1}^{l} (\mu_k - \mu_j)(\mu_k - \mu_j)^T}{l(l-1)} \tag{4}$$

$$S_w = \sum_{k=1}^{l} \frac{1}{n_i} \sum_{i=1}^{n_i} (x_k^i - \mu_l)(x_k^i - \mu_l)^T \tag{5}$$

where μ_l is the sample mean of class l.

Then the LDA basis is given by solving the generalized eigenvalue problem: $S_b\Phi = S_w\Lambda\Phi$. It is obvious that the rank of S_w is at most $l-1$, and thus the dimension of LDA subspace is also at most $l-1$. So the clear disadvantage of LDA is that it is only suitable for discriminating data that is generated from large number of classes.

3 Kernel-Based Subspace Analysis

In this section, we will show the idea of derivations of Kernel Subspaces using KPCA and KLDA as examples. All the notations are consistent with those of linear subspaces.

3.1 Kernel Principal Component Analysis

While PCA finds the principal components by solving the eigenvalues of sample covariance matrix, KPCA will perform the same computation in another transformed space F, which is related to the input space by a nonlinear map:

$$\Phi : x \in R^m \mapsto \Phi(x) \in F. \tag{6}$$

It is important to note that transformed feature space F may have infinite dimensionality, which is prohibitive to deal with. Kernel methods solve this problem by utilizing the inner product of mapping Φ, instead of directly working with F. The sample covariance matrix in F is given by:

$$\bar{C} = \frac{1}{n}\sum_{j=1}^{n} \Phi(x_j)\Phi(x_j)^T \tag{7}$$

Like PCA, we will solve an eigenvalue problem: $\lambda V = \bar{C}V$. Clearly, all eigenvectors in V with non-zero λ lie in the span of $\Phi(x_1), ..., \Phi(x_n)$. Thus this has following two useful consequences . Firstly, there exist coefficients $\alpha_i (i = 1, ..., n)$ such that,

$$V = \sum_{i=1}^{n} \alpha_i \Phi(x_i). \tag{8}$$

Secondly, the following equation also holds.

$$\lambda(\Phi(x_k), V) = (\Phi(x_k), \bar{C}V) \text{ for all k=1,...,n} \tag{9}$$

From Eq.9 and 8

$$\lambda \sum_{i=1}^{n} \alpha_i (\Phi(x_k), \Phi(x_i)) = \frac{1}{n}\sum_{i=1}^{n} \alpha_i (\Phi(x_k) \sum_{j=1}^{n} \Phi(x_j))(\Phi(x_j), \Phi(x_i)) \tag{10}$$

By defining a kernel matrix: $K_{ij} = \langle \Phi(x_i), \Phi(x_j) \rangle$, it follows from Eq. 10 that the problem becomes solving the eigenvalue of kernel matrix K, without even knowing the mapping Φ.

$$\frac{K}{n}\alpha^i = \lambda_i \alpha^i \tag{11}$$

where α^i denotes the column vector, $[\alpha_1^i, ..., \alpha_n^i]^T$.

Let $\lambda_1 \leq \lambda_2 \leq ... \leq \lambda_n$ and $\alpha^1, ..., \alpha^n$ denote the eigenvalues and the corresponding eigenvectors of $\frac{K}{n}$, respectively. We then normalize α^i so that $(\alpha^i, \alpha^i) = 1$ for all $i = \{1, ..., n\}$. From Eq.8, we get the following equation for α^i:

$$1 = \sum_{i,j=1}^{n} \alpha_i^k \alpha_j^k (\Phi(x_i), \Phi(x_j)) = \sum_{i,j=1}^{n} \alpha_i^k \alpha_j^k K_{ij} = (\alpha^k, K\alpha^k) = \lambda_k(\alpha^k, \alpha^k) \tag{12}$$

Given a new testing sample x, with its image $\Phi(x)$ in F, the projections onto the eigenvectors V^k in $F(k = 1, ..., n)$ is thus computed as:

$$(V^k, \Phi(x)) = \sum_{i=1}^{n} \alpha_i^k (\Phi(x_i), \Phi(x)) \tag{13}$$

In summary, to perform kernel based PCA (see Fig. 1), the following steps should be carried out. First compute the matrix $K_{ij} = (k(x_i, x_j))_{ij}$. Next find the eigenvectors and eigenvalues by solving Eq. 11. Then extract the principal components of a test sample x by Eq.13.

Note that Φ can be an arbitrary nonlinear map into an high-dimensional space F. By using the kernel representation $k(x, y) = (\Phi(x), \Phi(y))$, it allows us to compute the dot product in F without having to carry out the map. Typically the following three kernel function are widely used in practice.

- Polynomial Kernel:
$$k(x, y) = (1 + \langle x, y \rangle)^d \tag{14}$$
- Gaussian Kernel:
$$k(x, y) = exp(-\frac{\|x - y\|^2}{2\sigma^2}) \tag{15}$$
- Sigmoid Kernel:
$$k(x, y) = \tanh(k(x.y) + \Theta). \tag{16}$$

.

3.2 Kernel Linear Discriminant Analysis

The basic idea behind KLDA is to generate a nonlinear discriminant in the input space by using the kernel trick and LDA(Linear Discriminant Analysis). First, the input data is nonlinearly mapped into an implicit feature space F, $\Phi : x \in R^m \mapsto \Phi(x) \in F$. Then try to find a linear transformation in F that can maximize the between class scatter and minimize the within class scatter in F. Also one does not need to calculate Φ explicitly but can compute the inner product in F using kernel function: $k(x, y) = \langle \Phi(x), \Phi(y) \rangle$

The between-class and within-class scatter matrix S_b, S_w in F are defined as:

$$S_b = \frac{1}{l(l-1))} \sum_{i=1}^{l} \sum_{j=l}^{l} (u_i - u_j)(u_i - u_j)^T \tag{17}$$

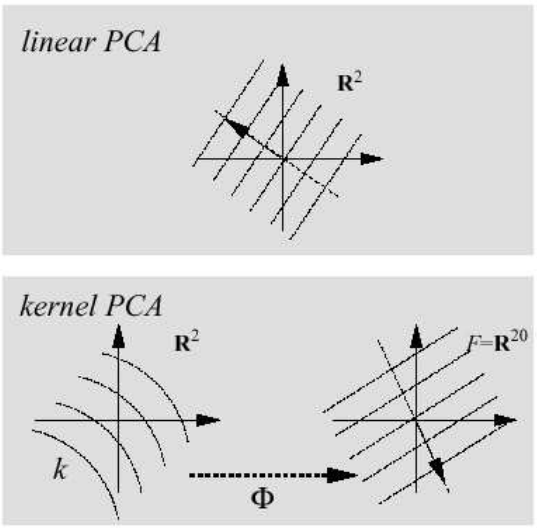


Figure 1: **The basic idea of kernel PCA. We perform a linear PCA in nonlinear high dimensional subspace, just like a PCA in input space(top),courtesy of [26].**

$$S_w = \frac{1}{l}\sum_{i=1}^{l}\frac{1}{n_i}\sum_{j=1}^{n_i}(\Phi(x_j)-u_i)(\Phi(x_j)-u_i)^T \tag{18}$$

where $u_i = \frac{1}{n_i}\sum_{j=1}^{n_i}\Phi(x_j)$ denotes the sample mean of class i in F.
In order to perform KLDA in F, one should maximize the following.

$$J(w) = \frac{w^T S_b w}{w^T S_w w} \tag{19}$$

Using the same conclusion derived in KPCA that any solution $w\epsilon F$ must lie in the span of all the samples in F, there exists coefficients α_i , $i = 1, 2...n$, such that

$$w = \sum_{i=1}^{n}\alpha_i\Phi(x_i) \tag{20}$$

By the above equation , the projection of each class means u_i onto w can be described as:

$$w^T u_i = \alpha^T \begin{pmatrix} \frac{1}{n_i}\sum_{j=1}^{n_i} k(x_1, x_j) \\ \frac{1}{n_i}\sum_{j=1}^{n_i} k(x_2, x_j) \\ \cdots \\ \frac{1}{n_i}\sum_{j=1}^{n_i} k(x_n, x_j) \end{pmatrix} = \alpha^T m_i \tag{21}$$

it follows that

$$w^T S_b w = \alpha^T K_b \alpha \tag{22}$$

where $K_b = \frac{1}{l(l-1)}\sum_{i=1}^{l}\sum_{j=1}^{l}(m_i - m_j)(m_i - m_j)^T$, and a similar transformation as in 22, it can be found that

$$w^T S_w w = \alpha^T K_w \alpha \tag{23}$$

where $K_w = \frac{1}{l}\sum_{i=1}^{l}\frac{1}{n_i}\sum_{j=1}^{n_i}(\zeta(x_j) - m_i)(\zeta(x_j) - m_i)^T$,
where $\zeta_j = (k(x_1, x_j), k(x_2, x_j), ..., k(x_n, x_j))^T$.
Thus maximizing Eq.19 is converted to maximizing:

$$J(\alpha) = \frac{\alpha^T K_b \alpha}{\alpha^T K_w \alpha} \tag{24}$$

Like LDA, this can be easily solved by finding the eigenvectors of $K_w^{-1} K_b$, and the projection of a new sample x onto w in F is given by

$$(w, \Phi(x)) = \sum_{i=1}^{n} \alpha_i k(x_i, x) \tag{25}$$

In practice, K_w^{-1} does not always exist, or has poor rank conditions. This problem can be addressed using regularization to stabilize K_w:

$$K_w^{'} = (K_w + \lambda I)^{-1} \tag{26}$$

where λ is a small number, and I is identity matrix.

4 Facial Image Analysis and Recognition

4.1 Facial Image pre-processing

The very first task to perform face recognition is to localize the face(and facial features) on the images. There exist many algorithms in the literature. Readers can refer to a recent survey [31]. In our report, we use the SVM(support vector machine) based face detection algorithm described in [17], because it is capable of robustly detecting both facial region and facial features(see

an example in Fig.2(a)). It is important to keep in mind that face is 3D object that is subject to deformations, and these deformations always exist even when we restrict all the faces to be frontal. In appearance-based face recognition, the misalignment of facial features in the image, caused by perspective projection of deformed 3D face, will obviously give misclassification. To address the above problem, warping technique is generally used to warp both training and testing image to shape-free representations through spline interpolation [18] or optical flow [29], and it is shown that higher recognition rate can be achieved using this approach [9].

We use a relatively simpler approach. Once the face and those important facial features have been localized, we will use the differences between the x and y coordinates of the two eyes, the original image is rotated until obtaining a frontal view face where both eyes have the same y value; i.e., $atan(\|y_1 - y_2\|/\|x_1 - x_2\|)$, where (x_1, y_1) and (x_2, y_2) are the right and left eye image coordinates. The face is then aligned to fit a final standard 60×80 rectangle. After warping, the eye centers, the medial line of nose and the center of mouth will be at the same position for all the faces. Fig.2(a) and Fig.2(b) show the image rotation and warping process respectively.

4.2 Learning and Identification

Given N training images $I = \{I_1, I_2, \cdots, I_N\}$ already warped, we first learn the subspace projection matrix Φ, and project the training images onto the subspace by $\hat{I}_i = \Phi * I_i (i$ the index of training image). In Fig.3, we visualize the PCA basis and ICA basis that are learned from FERET database respectively.

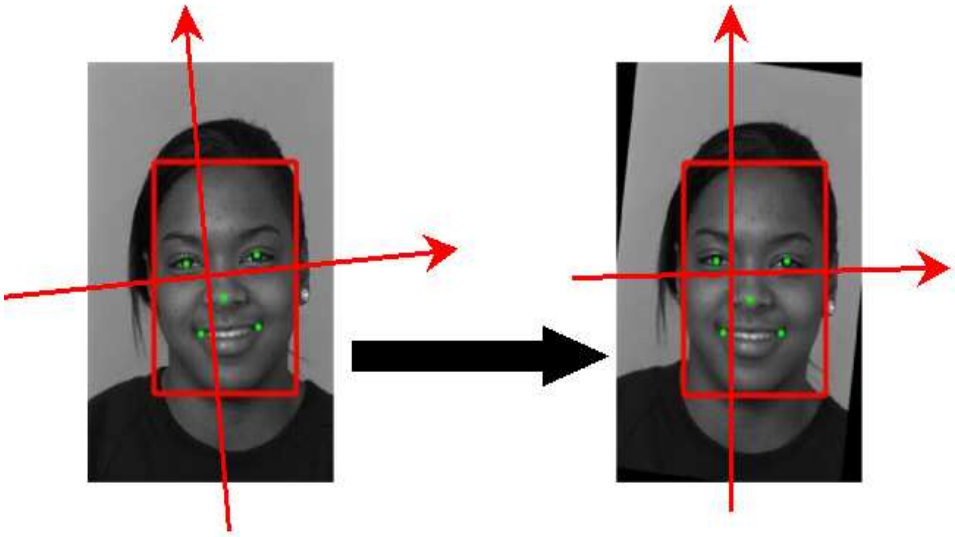
In this report, we will use Nearest Neighbor(NN) to do recognition, that is, a test T is first projected to the subspace by $\hat{T}_i = \Phi * T$ and then classified by assigning to it the class label of the closest training sample. Fig. 4 illustrates the idea of subspace recognition using NN rule. More formally,

$$s = argmin_i \|\hat{I}_i - \hat{T}\|_2 \,, \tag{27}$$

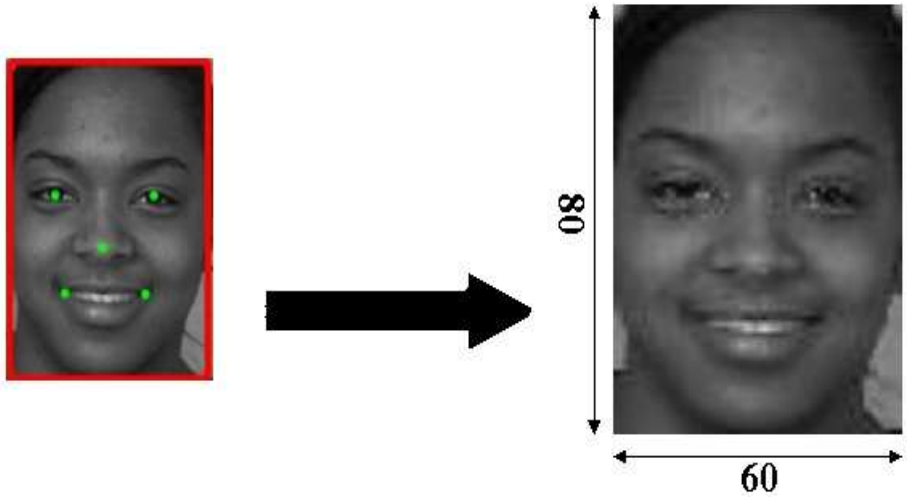
and determine the class label, of $\mathbf{I}_s$.

5 Database and Experimental Results

In this section, we will show the performance of the above different subspace techniques in face identity recognition and facial expression recognition. For face identity recognition, we focus on the recognition of pose-variant faces. To this end, we use 195 subjects from FERET(Face Recognition Technology)

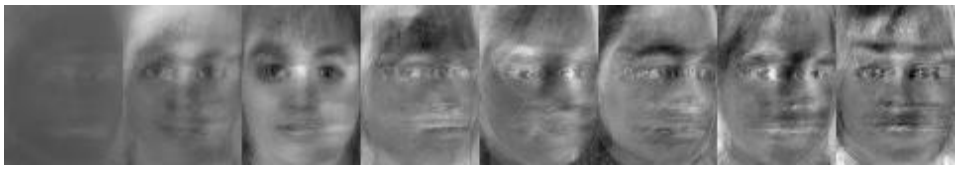


(a)

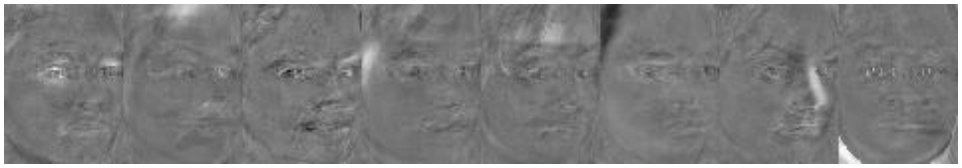


(b)

Figure 2: **(a) After face and facial feature detection, we rotate the face so that the centers of the two eyes have the same y coordinates. (b) Warp all the faces in the database to a standard** 60×80 **array so that the facial features are at the same locations.**

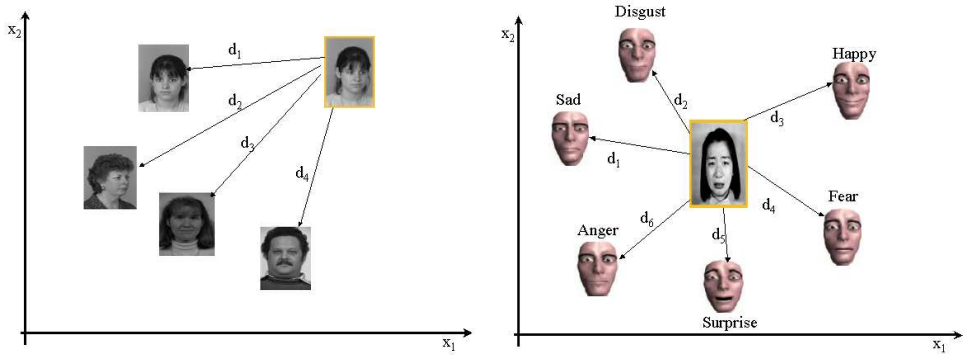


(a)



(b)

Figure 3: **(a) Eigen-faces(PCA basis).(b) ICA basis.**



(a) (b)

Figure 4: **(a),(b) illustrate facial identity recognition and expression recognition in subspace using nearest neighbor rule, respectively. The image in the box is the testing image.**

face database [25]. Each subject contains 9 samples with different poses. For facial expression recognition, we will use JAFFE(Japanese Female Facial Expression) database. The database contains 213 images of 7 facial expressions.

5.1 Pose Variant Face Recognition

The face images in FERET database display diversity across age, gender and ethnicity, and contain images taken under varying poses, illumination conditions, facial expression and time. FERET is now the standard for evaluating face recognition systems. In our experiment, we use faces from 195 subjects, each subject gives 9 images with head orientation ranging from -60^o to 60^o. Fig.5 shows sample images of a subject after cropping and warping process as detailed in Section.4.1.

Figure 5: **The images of a subject in Feret database.**

To evaluate the performance, We test all the algorithms by a leave-one-pose-out procedure. Each time, we use images of eight poses for training and the one left for testing. The recognition rate is thus the average of those procedures. We will show the experiment results we have got so far. Fig.6 shows the results achieved by using PCA, ICA and LDA.

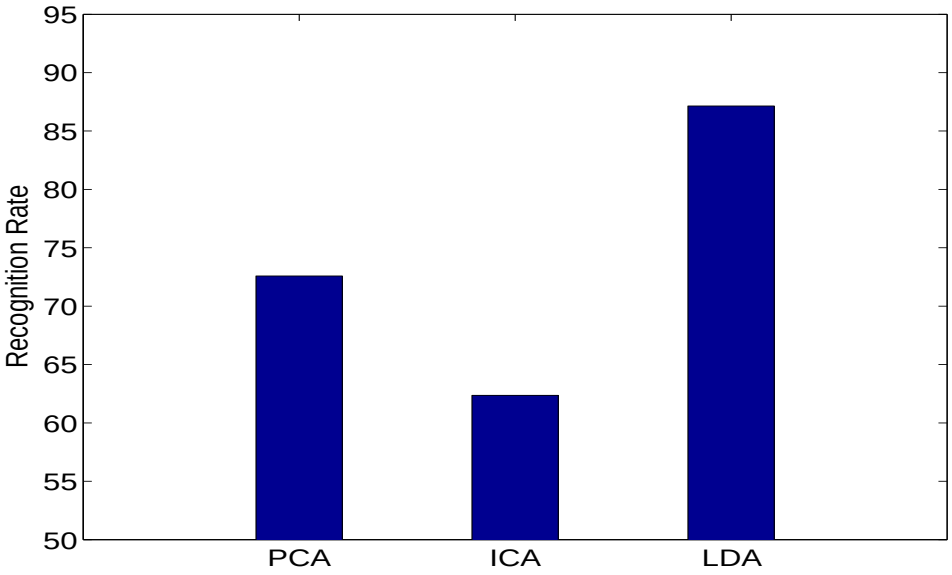


Figure 6: **Recognition rates of PCA, ICA and LDA.**

In Fig.7, we show the recognition rate obtained using KPCA with polynomial kernels. We test the performance using four different degrees of

polynomial: d=2, d=3, d=0.5, d=0.8. Our results are coherent with the observations in [20] that fractional power polynomial($0 < d < 1$) is a good choice in some scenarios. We also demonstrate the performance of KLDA using different degrees of polynomial kernels in Fig.8. While in the KLDA case, the fractional power kernel does not show superiority over integer power ones. Comparing with other subspace representation, KLDA gives the best recognition performance.

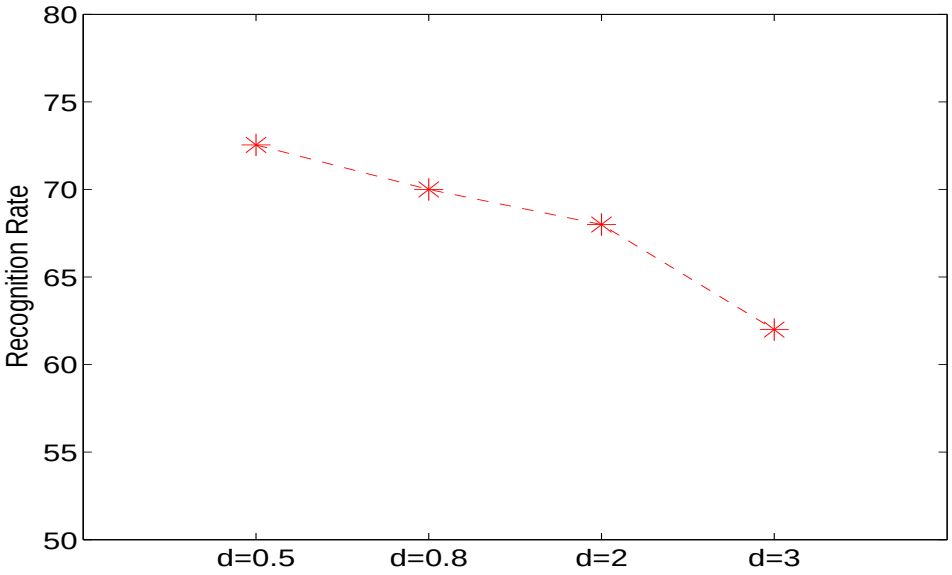


Figure 7: **Recognition rates of KPCA using different degrees of polynomial kernels.**

5.2 Facial Expression Recognition

The JAFFE database contains 213 images of neutral and 6 basic facial expressions(happiness, sadness, surprise, anger, disgust, fear). These images are from 10 Japanese female. Example images of these expressions are shown as Fig. 9.

Excluding the images with neutral expression, we have 168 images, which consist of 6 expressions with 28 images each emotion. To test the performance of the algorithms, we use 14 images of each expression, which gives a total of 84 images, for training, and the rest for testing. Since the way to pose a expression varies significantly across different individuals, it is even a difficult problem for human to judge a given expression. Therefore, unlike the procedure used in recognition of identity, it is generally more meaningful to

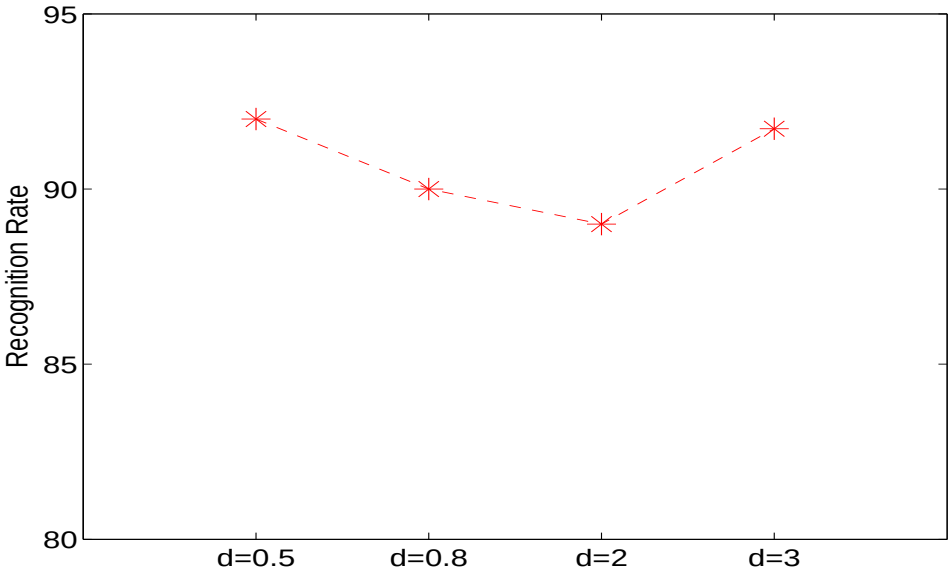


Figure 8: **Recognition rates of KLDA using different degrees of polynomial kernels.**

Figure 9: **The images of a subject in JAFFE database.**

rate a given image according to six expressions, instead of simply assigning it a "hard" class label. In JAFFE database, each image has been rated on 6 emotion adjectives by 60 Japanese subjects. Thus, each images has six scores ranging from $1-5$ that represent its similarity to each expression(refer to Fig.10).

In order to evaluate the algorithms, we compute the correlation between the similarity scores to each class calculated by the algorithms and the scores rated by human. Experimental results using PCA and ICA for recognizing six expressions are given in Fig.11(LDA is not considered because, as we described in section.2, there are only 6 classes here, that is, only up to six features are useful). In average, ICA works well for expression analysis with a correlation score of about 93%, while PCA can achieve 89%.

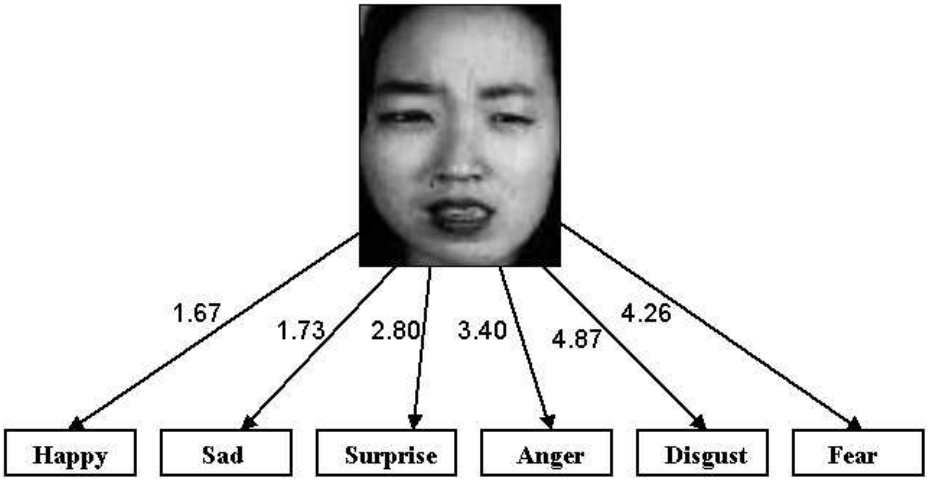


Figure 10: **This figure illustrates the rating of a given face by human, the ground-truth expression of the face in the figure is disgust.**

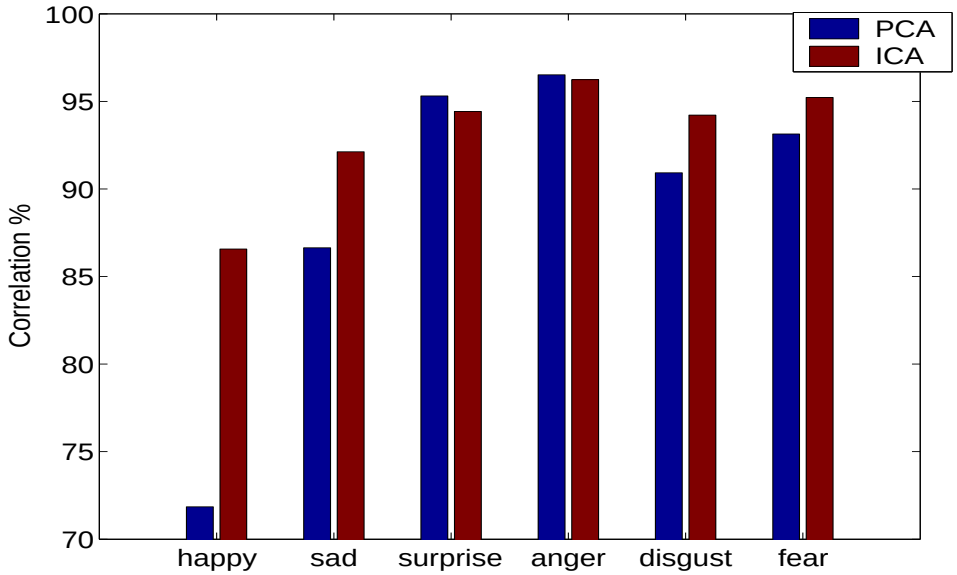


Figure 11: **This figure shows the expression recognition results using PCA and ICA, which are measured by the correlation scores with the human rating.**

In Fig.12, we also show the results using KPCA with polynomial kernels. We test the performance using four different degrees of polynomial: d=2, d=3, d=0.5, d=0.8. We find that the results are quite similar, with a average

correlation score about $88\% - 89\%$.

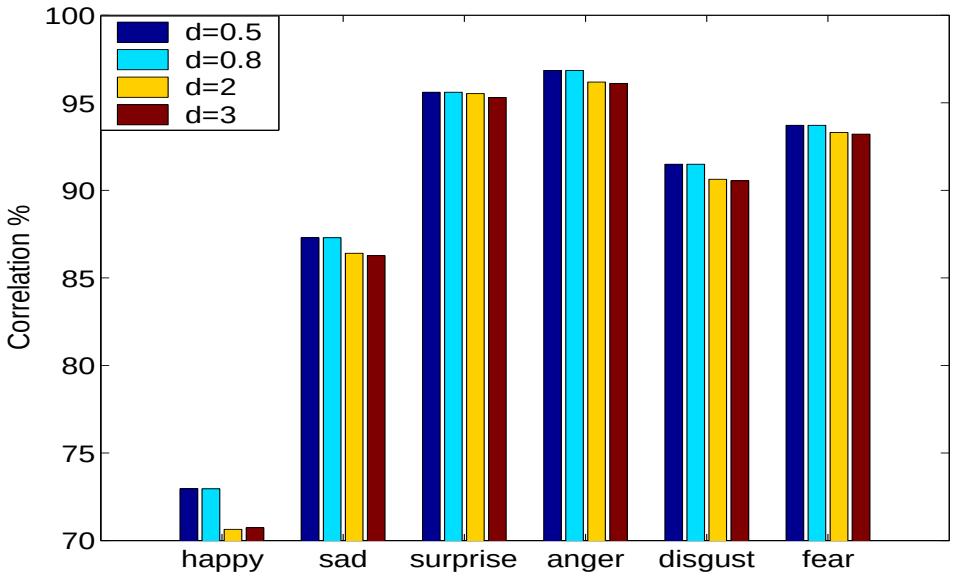


Figure 12: **This figure shows the expression recognition results using KPCA with different degrees of polynomial kernels.**

6 Conclusion

In this report, we have studied various subspace methods in identity recognition and facial expression recognition. For identity recognition, we specifically investigate the generalization performance of subspace methods to pose variations. FERET face database is used to test all the algorithms using leave-one-pose-out strategy. Experimental results shows: 1). KLDA using polynomial kernel gives the best performance. 2). LDA is better than all the others except KLDA. 3). PCA achieves similar rate as KPCA, but ICA performs the worst. It is straightforward to justify the good performance of LDA and its kernel version, since it utilizes the class information, and the data is quite friendly for discriminant analysis(large number of class and samples). Since ICA seeks to use all the high order statistics, it might cause over-fitting problem for the data, which in a way explains its low recognition rate. It is also worthy mentioning that KPCA with polynomial kernels gives the best results for d=0.5 than others. In fact, the degree of the polynomial means the order of the statistics we use to fit the data. When d=1, we only use up to second order statistics, which is equivalent to PCA; when d¿1, we use higher order statistics like kurtosis, and when d¡1(fractional power), we are actually

fitting a sub-gaussian distribution, which might best explain the data in our case. For facial expression problem, we propose to use the correlation between calculated rating vector and that given by human as the performance measure, because of the complex nature of expression perception. JAFFE database is utilized to test the algorithms on rating six basic human expressions. Though ICA is unfavorable in pose-variant identity recognition, it is demonstrated in the experimental results that ICA shows superiority in dealing with expressions over all the other algorithms, whose performance are quite similar. This is because ICA might well capture the statistics of the image data with expression variations, which is supposed to be much more complex than that of pose variations.

In summary, kernel methods in feature extraction don't appear to show much superiority over other algorithms from our experimental results, though it is now very popular in the community. Since the good performances are achieved only by those models that are able to explain the given data set, we argue that one of the future attempts on kernel methods should be around how to adaptively select kernels from the data set.

Acknowledgments

We are grateful for Profs. Prem Goel and Joe Verducci for helpful discussion.

References

[1] F.Bach and M.I.Jordan, "Kernel Independent Component Analysis", Technical Report CSD-01-1166, Computer Science Division, University of California, Berkeley, 2001.

[2] M.S.Bartlett, "Face Image Analysis by Unsupervised Learning," Kluwer Academic,Dordrecht,2001.

[3] P.N.Belhummeour, J.P.Hespanha and D.J.Kriegman, "Eigenfaces vs.Fisherfaces:Recogniton using class specific linear projection" IEEE Trans. PAMI 19(7):711-720,1997.

[4] A.J.Bell and T.J.Sejnowski,"An information-maximization approach to blind seperation and blind deconvolution," Neural Computation 7(1995) 1129-1159

[5] D.Beymer and T.Poggio,"Face Recognition from one Example View," ICCV 1995.

[6] M.J. Black and P. Anandan, "The robust estimation of multiple motions: Parametric and piecewise-smooth flow fields," Comp Vis. Image Under 63(1):75-104, 1996.

[7] R.Brunelli and T.Poggio, "Face Recognition: Features versus Templates", IEEE Trans. on PAMI. vol.15(10), 1993.

[8] F.Camastra, "Kernel Methods for Computer Vision: Theory and Applications",

[9] I.Craw, N.Costen, T.Kato and S.Akamatsn, "How should We Represent Faces for Automatic Recognition", IEEE Trans. on PAMI, vol.21(8), pp.725-736, 1999.

[10] I.J. Cox, J. Ghosn and P.N. Yianilos. "Feature-based face recognition using mixture distance". In NEC TR, 1996.

[11] B.A.Draper, K.Baek, M.S.Bartlett and J.R.Beverige, "Recognizing faces with PCA and ICA", Computer Vision and Image Understanding, vol.91,pp.115-137. 2003.

[12] K. Etemad and R. Chellapa, "Discriminant analysis for recognition of human face images," Journal of Optics of American A 14(8):1724-1733, 1997.

[13] R.A.Fisher,"The Statistical Utilization of Multiple Measurements," Annals of Eugenics,8:376-386,1938.

[14] K.Fukunaga, "Introduction to Statistical Pattern Recognition(second edition)," Academic Press, 1990.

[15] M.Girolami, Orthogonal Series Density Estimation and the Kernel Eigenvalue Problem", Neural Computation, vol.14(3),pp.669-688.2001.

[16] P.L. Hallinan, G.G. Gordon, A.L. Yuille, P. Giblin and D. Mumford, "Two- and Three-Dimensional Patterns of the Face," AK Peters, 1999.

[17] B. Heisel, T. Sere, M. Pontil, T. Vetter, T. Poggio, "Categorization by Learning and Combining object parts", In Proc. NIPS, 2001.

[18] A. Lanitis, C.J. Taylor and T.F. Cootes, "Automatic Classification and Coding of Face Images Using Flexible Models", IEEE Trans.PAMI 19(7):743-756,1997.

[19] Q.S.Liu, R.Huang, H.Lu and S.D.Ma, "Face Recognition Using Kernel Based Fisher Discriminate Analysis", Proc. of the Fifth IEEE Int. Conf. on Automatic Face and Gesture Recognition, 2002.

[20] C.Liu "Gabor-Based Kernel PCA with Fractional Power Polynomial Models For Face Recognition", IEEE Trans.on PAMI, vol.26(5), pp.572-580, 2004.

[21] Michael Lyons and etc., Coding Facial Expressions with Gabor Wavelets, Proceedings, Third IEEE International Conference on Automatic Face and Gesture Recognition, April 14-16 1998, Nara Japan, IEEE Computer Society, pp. 200-205

[22] S.Mika, G.Ratsch and J.Weston etc, "Fisher Discriminant Analysis with Kernels", Neural Networks for Signal Processing IX, 41-48, 1999.

[23] S.Mika, B.Scholkopf, A.Smola and etc, "Kerenl PCA and denoising in Feature Space", Advances in Neural Information Processing Systems 11, MIT Press, 1999.

[24] Curtis Padgett, Garrison Cottrell. Identifying emotion in static face images. In Proceedings of 2nd Joint Symposium on Neural Computation, vol. 5, pp.91-101, La Jolla, CA, 1995

[25] P.J. Phillips, H. Moon,P. Rauss and S.A.Rizvi, "The FERET evaluation methodology for face recognition algorithms," IEEE Trans PAMI 22(10):1090-1104, 2000.

[26] B.Schelkopf, A.Smola and K.Muller, "Nonlinear Component Analysis as a Kernel Eigenvalue Problem", Neural Compuation, vol.10, pp.1299-1319, 1998.

[27] L.Sirovich and M.Kirby, "Low-dimensional procedure for the characterization of human faces," Journal of the Optical Society of America A 4 (1987) 519–524

[28] M. Turk and A. Pentland, "Eigenfaces for recogntion," Journal of Cognitive Neuroscience 3(1):71-86,1991.

[29] T. Vetter and T. Poggio,"Linear Object Class and Image Synthesis from single example view," IEEE Trans. PAMI 19(7):733-742, 1997.

[30] M.H.Yang, Narendra Ahuja and D.Kriegman. "Face Recognition Using Kernel Eigenface". Int.Conf on Image Processing, 2000.

[31] M.H.Yang, D.Kriegman and N.Ahuja, "Detecting Faces In Images: A Survey", IEEE trans. on PAMI,vol.24(1),pp.34-58, 2002.

[32] W.Y. Zhao, R. Chellappa, A. Rosenfeld and J.P. Phillips, "Face Recognition: A Literature Survey", ACM Computing Surveys, 2003.