# Kernel Modified Quadratic Discriminant Function for Facial Expression Recognition*

Duan-Duan Yang[1], Lian-Wen Jin[1], Jun-Xun Yin[1], Li-Xin Zhen[2], and Jian-Cheng Huang[2]

[1] Department of Electronic and Communication Engineering, South China University of Technology, 510640 Guangzhou, China
{ddyang, eelwjin, eejxyin}@scut.edu.cn
[2] Motorola China Research Center, 210000, Shanghai, P.R. China
{Li-Xin.Zhen, Jian-Cheng.Huang}@motorola.com

**Abstract.** The Modified Quadratic Discriminant Function was first proposed by Kimura et al to improve the performance of Quadratic Discriminant Function, which can be seen as a dot-product method by eigen-decompostion of the covariance matrix of each class. Therefore, it is possible to expand MQDF to high dimension space by kernel trick. This paper presents a new kernel-based method to pattern recognition, Kernel Modified Quadratic Discriminant Function(KMQDF), based on MQDF and kernel method. The proposed KMQDF is applied in facial expression recognition. JAFFE face database and the AR face database are used to test this algorithm. Experimental results show that the proposed KMQDF with appropriated parameters can outperform 1-NN, QDF, MQDF classifier.

## 1 Introduction

Statistical techniques have been widely used in various pattern recognition problems[1]. Statistical classifiers include linear discriminant function(LDF), quadratic discriminant function(QDF), Parzen window classifier, nearest-neighbor(1-NN) and k-NN rules, etc. Under the assumption of multivariate Gaussian density for each class, the quadratic discriminant function is obtained based on Bayes theory. The modified QDF(MQDF) proposed by Kimura et al. [2]aims to improve the computation efficiency and classification performance of QDF via eigenvalue smoothing, which have been used successfully in the handwriting recognition[2,3]. The difference from the QDF is that the eigenvalues of minor axes are set to a constant. The motivation be-hind this is to smooth the parameters for compensating for the estimation error on finite sample size.

* The paper is sponsored by Motorola Human Interface Lab Research Foundation(No.303D804372), New Century Excellent Talent Program of MOE(No.NCET-05-0736), NSFGD(No.04105938).

N. Zheng, X. Jiang, and X. Lan (Eds.): IWICPAS 2006, LNCS 4153, pp. 66–75, 2006.

On the other hand, kernel-based learning machines, e.g., support vector machines(SVMs)[4], kernel principal component analysis(KPCA)[5], and kernel Fisher discriminant analysis(KFD)[6,7,8], have been got much interest in the fields of pattern recognition and machine learning recently. The basic idea of kernel methods is finding a mapping such that, in new space, problem solving is easier(e.g. linear). But the mapping is left implicit. The kernel represents the similarity between two objects defined as the dot-product in this new vector space. Thus, the kernel methods can be easily generalized to a lot of dot-product (or distance) based pattern recognition algorithms. QDF and MQDF can also be seen as dot-product methods by eigen-decompostion of the covariance matrix. Therefore, it is nature that MQDF can be generalized to a new high-dimension space by kernel trick.

This paper proposes a new kernel-based method to pattern recognition, Kernel Modified Quadratic Discriminant Function(KMQDF), based on kernel methods and MQDF. For testing and evaluating its performance, the proposed KMQDF is applied for facial expression recognition(FER) on two face databases. Experimental results show that KMQDF with appropriated parameters can outperform 1-NN, QDF, MQDF classifier.

## 2 MQDF

In this section we would give a brief review the MQDF. Let us start with the Bayesian decision rule, which classifies the input pattern to the class of maximum a posteriori(MAP) probability out of class. Representing a pattern with a feature vector, the a posteriori probability is computed by Bayes rule:

$$P(w_i|x) = p(x|w_i)P(w_i)/p(x) \tag{1}$$

where $P(w_i)$ is the a priori probability of class, $p(x|w_i)$ is the class probability density function(pdf) and $p(x)$ is the mixture density function. Since $p(x)$ is independent of class label, the nominator of (1) can be used as the discriminant function for classification:

$$g(w_i|x) = p(x|w_i)P(w_i) \tag{2}$$

The Bayesian classifier is reduced to QDF under the Gaussian density assumption with varying restrictions. Assume the probability density function of each class is multivariate Gaussian

$$p(x|w_i) = \frac{1}{2\pi^{\frac{d}{2}}|\Sigma|_i^{\frac{1}{2}}} \exp[\frac{(x-\mu_i)^T \Sigma_i^{-1}(x-\mu_i)}{2}] \tag{3}$$

where $\mu_i$ and $\Sigma_i$ denote the mean vector and the covariance matrix of class, respectively. Inserting (3) into (2), taking the negative logarithm and omitting the common terms under equal priori probabilities, the QDF is obtained as

$$g(w_i|x) = (x-\mu_i)^T \Sigma_i^{-1}(x-\mu_i) + \log|\Sigma_i| \tag{4}$$

The QDF is actually a distance metric in the sense that the class of minimum distance is assigned to the input pattern. By eigen-decompostion, the covariance matrix can be diagonalized as

$$\Sigma_i = B_i \Lambda_i B_i^T \tag{5}$$

where $\Lambda_i$ is a diagonal matrix formed by the eigenvalues of $\Sigma_i$, $B_i$ is formed by the corresponding eigenvectors.

According to (5), the QDF can be rewritten in the form of eigenvectors and eigen-values:

$$\begin{aligned} g(w_i|x) &= B_i^T(x-\mu_i)^T \Lambda_i^{-1} B_i^T(x-\mu_i) + \log|\Lambda_i| \\ &= \sum_{j=1}^{d}(\frac{1}{\lambda_{ij}})[\beta_{ij}^T(x-\mu_i)]^2 + \sum_{j=1}^{d}\log(\lambda_{ij}) \end{aligned} \tag{6}$$

Replacing the minor eigenvalues with a constant, the modified quadratic discriminant function [3] is obtained as follows:

$$\begin{aligned} g_2(w_i|x) &= \sum_{j=1}^{k}(\frac{1}{\lambda_{ij}})[\beta_{ij}^T(x-\mu_i)]^2 + \sum_{j=1}^{k}\log(\lambda_{ij}) \\ &\quad + \sum_{j=k+1}^{d}(\frac{1}{\delta_i})[\beta_{ij}^T(x-\mu_i)]^2 + (d-k)\log\delta_i \\ &= \sum_{j=1}^{k}(\frac{1}{\lambda_{ij}})[\beta_{ij}^T(x-\mu_i)]^2 + \sum_{j=1}^{k}\log(\lambda_{ij}) \\ &\quad + (\frac{1}{\delta_i})r_i(x) + (d-k)\log(\delta_i) \end{aligned} \tag{7}$$

where $k$ denotes the number of principal axed and $r_i(x)$ is the residual of subspace projection:

$$r_i(x) = \|x-\mu_i\|^2 - \sum_{j=1}^{k}[\beta_{ij}^T(x-\mu_i)]^2 \tag{8}$$

The (8) utilizes the invariance of Euclidean distance.

The advantage of MQDF is multifold. First, it overcomes the bias of minor eigen-values (which are underestimated on small sample size) such that the classification performance can be improved. Second, for computing the MQDF, only the principal eigenvectors and eigenvalues are to be stored so that the memory space is reduced. Third, the computation effort is largely saved because the projections to minor axes are not computed[3].

The parameter $\delta_i$ of MQDF can be set to a class-independent constant as the following equation[2,9]:

$$\delta_i = (tr(\Sigma_i) - \sum_{j=1}^{k}\lambda_{ij})/(d-k) = \sum_{j=k+1}^{d}\lambda_{ij}/(d-k) \tag{9}$$

where $tr(\Sigma_i)$ denotes the trace of covariance matrix.

## 3 Kernel MQDF

As a statistical algorithm, MQDF can detect stable patterns robustly and efficiently from a finite data sample. Embedding the data sample in a suitable feature by kernel trick, it is possible that MADF can perform better than in the original feature space. According to this idea, we subsequently present the new kernel-based method, KMQDF algorithm, in this section.

For a given nonlinear mapping $\Phi$, the input data space $IR^n$ can be mapped into the feature space $H$. As a result, a pattern in the original input space $IR^n$ is mapped into a potentially much higher dimensional feature vector in the feature space $H$. Since the feature space $H$ is possibly infinite dimensional and the orthogonality needs to be characterized in such a space, it is reasonable to view $H$ as a Hilbert space. An initial motivation of KMQDF is to perform MQDF in the feature space $H$. However, it is difficult to do so directly because it is computationally very intensive to compute the dot products in a high-dimensional feature space. Fortunately, kernel techniques can be introduced to avoid this difficulty. The algorithm can be actually implemented in the input space by virtue of kernel tricks. The explicit mapping process is not required at all.

Given a set of $M$ training samples $x(x_{i1}, x_{i2}, \ldots\ldots, x_{iM})$ in $IR^n$, labeled with the $ith$ class, the covariance operator on the feature space $H$ can be constructed by

$$\Sigma_i^{\Phi} = (\frac{1}{M}) \sum_{j=1}^{M} (\Phi(x_{ij}) - m_{i0}^{\Phi})(\Phi(x_{ij}) - m_{i0}^{\Phi})^T \tag{10}$$

where $m_0^{\Phi} = (\frac{1}{M}) \sum_{j=1}^{M} \Phi(x_{ij})$. In a finite-dimensional Hilbert space, this operator is generally called covariance matrix. Since every eigenvalue of a positive operator is nonnegative in a Hilbert space[10], it follows that all nonzero eigenvalues of are positive. It is the positive eigenvalues that are of interest to us. It is easy to show that every eigenvector of $\Sigma_i^{\Phi}$, $\beta$ can be linearly expanded by

$$\beta = \sum_{j=1}^{M} a_j \Phi(x_{ij}) \tag{11}$$

To obtain the expansion coefficients, let us denote $Q = [\Phi(x_{i1}) \ldots\ldots \Phi(x_{iM})]$, and form an $M * M$ Gram matrix $\tilde{R}_i = Q_i^T Q_i$ ,whose elements can be determined by virtue of kernel tricks:

$$\tilde{R}_{i(u,v)} = \Phi(x_{iu})^T \Phi(x_{iv}) = (\Phi(x_{iu}) \bullet \Phi(x_{iv})) = ker(x_{iu} \bullet x_{iv}) \tag{12}$$

We centralize $\tilde{R}_i$ by $R_i = \tilde{R}_i - 1_M \tilde{R}_i - \tilde{R}_i 1_M + 1_M \tilde{R}_i 1_M$, where $1_M = (\frac{1}{M})_{M \times M}$. On the other hand, We can denote $\Sigma_i^{\Phi}$ and $R_i$ using $Q_i$ as flowing:

$$\Sigma_i^{\Phi} = (\frac{1}{M})(Q_i - Q_i 1_M)(Q_i - Q_i 1_M)^T \tag{13}$$

$$R_i = (Q_i - Q_i 1_M)^T (Q_i - Q_i 1_M) \tag{14}$$

Consider an eignevector-eignevalue pair $\gamma_i$ and $\lambda_i$ of $R_i$, we have

$$\frac{1}{M}(Q_i - Q_i 1_M)(Q_i - Q_i 1_M)^T (Q_i - Q_i 1_M)\gamma_i = \frac{1}{M}\lambda_i (Q_i - Q_i 1_M)\gamma_i \tag{15}$$

Inserting (14) to (15),we can get

$$\Sigma_i^{\Phi}(Q_i - Q_i 1_M)\gamma_i = (\frac{\lambda_i}{M})(Q_i - Q_i 1_M)\gamma_i \tag{16}$$

Equation (16) implys that $(Q_i - Q_i 1_M)\gamma_i$, $\frac{\lambda_i}{M}$ is an eigenvetor-eigenvalue pair of $\Sigma_i^{\Phi}$ . Furthermore, the norm of $(Q_i - Q_i 1_M)\gamma_i$ is given by

$$\|(Q_i - Q_i 1_M)\gamma_i\|^2 = \gamma_i^T (Q_i - Q_i 1_M)^T (Q_i - Q_i 1_M)\gamma_i = \lambda_i \tag{17}$$

so that the corresponding normalized eigenvetor of $\Sigma_i^{\Phi}$ is $\beta_i = (Q_i - Q_i 1_M)\gamma_i/\sqrt{\lambda_i}$.

Calculate the orthonormal eigenvetors $r_{i1}, r_{i2} \ldots r_{im}$ of $R_i$ corresponding to the $m$ largest positive eigenvalues, $\lambda_{i1} \leq \lambda_{i2} \ldots \lambda_{im}$. The orthonormal eigenvetors $\beta_{i1}, \beta_{i2}, \ldots, \beta_{im}$ of $\Sigma_i^{\Phi}$ corresponding to the $m$ largest positive eignevalues, $\frac{\lambda_{i1}}{M}, \frac{\lambda_{i2}}{M}, \ldots, \frac{\lambda_{im}}{M}$, which are $\beta_i = (Q_i - Q_i 1_M)\gamma_i/\sqrt{\lambda_i}, j = 1, 2, 3, \ldots, m$.

Analogizing equation (7), in new feature space, we have KMQDF:

$$\begin{aligned} g_2^{\Phi}(w_i, x) &= \sum_{j=1}^{k}(\frac{1}{\lambda_{ij}^{\Phi}})[{\beta_{ij}^{\Phi}}^T(\Phi(x) - m_i^{\Phi})]^2 + \sum_{j=1}^{k}\log(\lambda_{ij}^{\Phi}) \\ &\quad + \sum_{j=k+1}^{d}(\frac{1}{\delta_{ij}^{\Phi}})[{\beta_{ij}^{\Phi}}^T(\Phi(x) - m_i^{\Phi})]^2 + (d-k)\log(\delta_i^{\Phi}) \\ &= \sum_{j=1}^{k}(\frac{M}{\lambda_{ij}})\left\{[(Q_i - Q_i 1_M)\gamma_i/\sqrt{\lambda_i}]^T[\Phi(x) - Q_i 1_{M\_v}]\right\}^2 \\ &\quad + \sum_{j=k+1}^{d}(\frac{1}{\delta_i^{\Phi}})\left\{[(Q_i - Q_i 1_M)\gamma_i/\sqrt{\lambda_i}]^T[\Phi(x) - Q_i 1_{M\_v}]\right\}^2 \\ &\quad + \sum_{j=1}^{k}\log(\frac{\lambda_{ij}}{M}) + (d-k)\log(\delta_i^{\Phi}) \\ &= \sum_{j=1}^{k}(\frac{M}{{\lambda_{ij}}^2})[{r_{ij}}^T(R_{it} - 1_M R_{it} - \tilde{R}_i 1_{M\_1} + 1_M \tilde{R}_i 1_{M\_1})]^2 \\ &\quad + \sum_{j=k+1}^{d}(\frac{1}{\delta_i^{\Phi}\lambda_{ij}})[{r_{ij}}^T(R_{it} - 1_M R_{it} - \tilde{R}_i 1_{M\_1} + 1_M \tilde{R}_i 1_{M\_1})]^2 \\ &\quad + \sum_{j=1}^{k}\log(\frac{\lambda_{ij}}{M}) + (d-k)\log(\delta_i^{\Phi}) \end{aligned} \tag{18}$$

where $R_{it} = [(\Phi(x_{i1}) \bullet \Phi(x)), (\Phi(x_{i2}) \bullet \Phi(x)), \ldots\ldots, (\Phi(x_{iM}) \bullet \Phi(x))]$,$1_{M\_v} = (\frac{1}{M})_{M\times 1}$ and $\delta_i = \sum_{j=k+1}^{d}(\frac{\lambda_{ij}}{M})/(d-k)$. We can utilizes the invariance of Euclidean distance to simply equation (18):

$$g_2^{\Phi}(w_i|x) = \sum_{j=1}^{k}(\frac{M}{{\lambda_{ij}}^2})[{r_{ij}}^T(R_{it} - 1_M R_{it} - \tilde{R}_i 1_{M\_1} + 1_M \tilde{R}_i 1_{M\_1})]^2 + \sum_{j=1}^{k}\log(\frac{\lambda_{ij}}{M}) + \frac{1}{\delta_i^{\Phi}} r_i^{\Phi}(x) + (d-k)\log(\delta_i^{\Phi}) \quad (19)$$

where

$$\begin{aligned} r_i^{\Phi}(x) &= \left\|(\Phi(x) - m_i^{\Phi})\right\|^2 - \sum_{j=1}^{k}[{\beta_{ij}^{\Phi}}^T(\Phi(x) - m_i^{\Phi})]^2 \\ &= (\Phi(x) \bullet \Phi(x)) - 2 * (1_{M\_1})^T \bullet R_{it} + (1_{M\_1})^T \tilde{R}_i (1_{M\_1}) \\ &\quad - \sum_{j=1}^{k}(\frac{1}{\lambda_{ij}})[{r_{ij}}^T(R_{it} - 1_M R_{it} - \tilde{R}_i 1_{M\_1} + 1_M \tilde{R}_i 1_{M\_1})]^2 \end{aligned} \quad (20)$$

It is expected that the KMQDF algorithm can embed the data in a suitable feature space, in which we can use MQDF algorithm to discover pattern easily.

## 4 Facial Expression Recognition Using KMQDF

Facial expression recognition has been an active area of research in the literature for long time. The ultimate goal in this research area is the realization of intelligent and transparent communications between human beings and machines. Several facial expression methods have been proposed in the literature[11,12,13]. In recent years, facial expression recognition based on two-dimensional (2-D) digital images has received a lot of attention by researchers, because it doesn't involve 3-D measurements[13] and is suitable for real time application. A more detailed review on facial expression recognition can be found in[11].

### 4.1 Feature Extraction

In this paper, we use local Gabor filters to extract the features for facial expression recognition. Gabor features have been applied widely in the field of computer vision because of its powerful analysis ability in the conjoint time-frequency domain. Local Gabor filters[14] optimize the structure of global Gabor filters, which can achieve the same performance as global Gabor filters but involve less computation and storage.

Principle component analysis (PCA) and linear discriminant analysis (LDA) are two classical tools widely used in face analysis for data reduction. PCA seeks a projection that best represents the original data in a least-squares sense, and LDA seeks a projection that best separates the data in a least-squares sense. Many LDA-based algorithms suffer from the so-called *"small sample size problem"*(SSS)[15] which exists in high-dimensional pattern recognition tasks, where the number of available samples is smaller than the dimensionality of the samples. Facial expression recognition often meets this problem. The most famous solution to the SSS problem is to utilize PCA concepts in conjunction with LDA (PCA plus LDA)[16,17]. The effectiveness of the method has been demonstrated by [16,17,18,19].

In this paper, the process of the experiments consists of three steps. Firstly, local Gabor filters are used to extract the facial expression features as the description in[14]. Secondly, the local Gabor features will be reduced based on PCA plus LDA. Thirdly, the reduced features would be classified using 1-NN, MQDF and KMQDF respectively.

### 4.2 Experimental Data

Two face databases are used to test KMQDF. The first one is AR face database[19], a subset of AR database is used for our experiments. This subset includes 999 images of 126 individuals with 4 different facial expressions. The images corresponding to the 101 persons are chosen for training (799 samples), while the remaining images are used to test. We repeat the experiments 5 times by changing the training samples and testing samples to obtain an average recognition rate. The second one is JAFFE databases[18]. Total of use the 210 images of 10 individuals are used for our facial expression experiment. (Each expression of one person includes 3 samples). The images corresponding to 8 persons (168 samples) are used as the training samples. The residual images (42samples) are used to test. In the same way, we repeat the experiments 5 times by changing the training samples and testing samples. Fig.1 and Fig. 2 show some example images in AR and JAFFE database.

All images for the experiments are normalized (96*128 pixels) and aligned based on the position of the eyes as Fig.3 shows.

**Fig. 1.** Images of one person with 4 different facial expressions in the AR database

**Fig. 2.** Images of one person with 7 different facial expressions in the JAFFE database

**Fig. 3.** Normalized images corresponding to the images in Fig.1

### 4.3 Experimental Results

A popular kernel, polynomial kernel, is involved in our tests:

$$ker(x, y) = (x \bullet y + 1)^d \tag{21}$$

To achieve the optimal recognition accuracy, the parameters of KMQDF( $k$ in the equation (19) and $d$ in the equation (21)) should be selected appropriately. Experiments show the optimal parameters are different for the different training set. Figure 4 gives an example that shows how the parameters of KMQDF affect the recognition accuracy. Table 1 and Table 2 give the results with the optical parameters on JAFFE and AR database respectively. In both Table1 and Table 2, $[T_1, T_2, \ldots, T_5]$ is used to index different testing sets.

From Table 1 and Table 2, it can be seen that the proposed KMQDF classifier with appropriated parameters can outperform the 1-NN, QDF, MQDF for the

**Table 1.** The recognition results on JAFFE DB

| Test set | 1-NN | QDF | MQDF | KMQDF |
|---|---|---|---|---|
| T1 | 71.43% | 66.67% | 69.05% | 80.95% |
| T2 | 80.95% | 69.04% | 85.71% | 85.71% |
| T3 | 66.67% | 61.90% | 71.43% | 73.81% |
| T4 | 73.81% | 50.00% | 78.57% | 78.57% |
| T5 | 76.19% | 78.57% | 78.57% | 80.95% |
| Average | 73.81% | 65.24% | 76.67% | **80.01**% |

**Table 2.** The recognition results on AR DB

| Test set | 1-NN | QDF | MQDF | KMQDF |
|---|---|---|---|---|
| T1 | 86.5% | 87.5% | 87.0% | 88.5% |
| T2 | 85.5% | 84.5% | 85.5% | 86.5% |
| T3 | 85.5% | 87.0% | 86.5% | 87.0% |
| T4 | 86.5% | 87.5% | 87.5% | 88.0% |
| T5 | 86.0% | 87.0% | 87.5% | 88.5% |
| Average | 86.0% | 86.7% | 86.8% | **87.7%** |

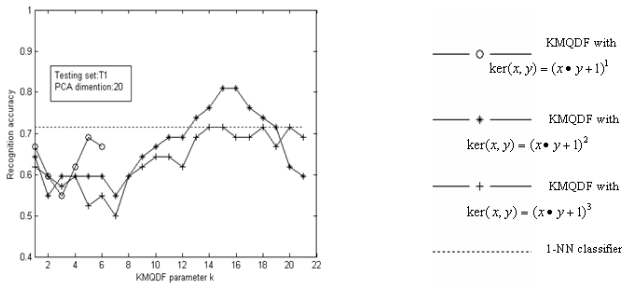


**Fig. 4.** Experiment results of T1 on the JAFFE database. X-axis is the modification parameter($k$ in the equation(19))of KMQDF.

facial expression recognition. Comparing with MQDF, an improvement of 3.3% recognition accuracy for JAFFE database and an improvement of 0.9% for AR database are obtained by the proposed kernel MQDF.

## 5 Conclusion

This paper presents a new kernel-based algorithm: Kernel MQDF, which can perform MQDF algorithm in a potentially much higher dimensional feature space. For testing its classifying capability, the proposed KMQDF is applied for facial expression recognition on the JAFFE face database and the AR face database. Experimental results show that the proposed KMQDF can outperform 1-NN, QDF, MQDF classifier.

Besides, as a new kernel-based algorithm, KMQDF may be expanded to solve other pattern recognition problems, such as characters recognition, face recognition etc, which merits our further study.

## References

1. A.K.Jain, R.P.W.Duin, and J.mao, "Statistical pattern recognition: A review", IEEE Trans-actions on Pattern Analysis and Machine Intelligence,vol.22,pp.4-37,Jan.2000.

2. F.Kimura, K.Takashina, S.Tsuruoka, and Y.Miyake, "Modified quadratic discriminant functions and the application to Chinese character recognition" IEEE Transactions on Pattern Analysis and Machine Intelligence, vol9,pp.149-153,Jan1987.
3. Liu CL, Sako H, Fujisawa H, "Discriminative Learning Quadratic Discriminant Function for Handwriting Recognition", IEEE Transaction on Neural networks Vol.15.No.2 March 2004.
4. V.Vapnik, the nature of stastical leaning Theory. New York: Springer, 1995.
5. B.Schölkopf, A. Smola, and K.-R.Müller, "Nonlinear ComponentAnalysis as a Kernel Eigenvalue Problem", Neural Computation,vol. 10, no. 5, pp. 1299-1319, 1998.
6. S.Mika, G.Rätsch, J. Weston, B. Schölkopf, and K.-R.Müller, "Fisher Discriminant Analysis with Kernels", Proc. IEEE Int'l Workshop Neural Networks for Signal Processing IX, pp. 41-48, Aug. 1999.
7. S.Mika, G.Rätsch, B.Schölkopf, A.Smola, J.Weston, and K.-R.Müller, "Invariant Feature Extraction and Classification in Kernel Spaces", Advances in Neural Information Processing Systems 12, Cambridge, Mass.: MIT Press, 1999
8. Jian Yang, Alejandro F. Frangi, Jing-yu Yang, David Zhang, "KPCA Plus LDA: A Complete Kernel Fisher Discriminant Framework for Feature Extraction and Recognition", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 27, No. 2, February 2005.
9. B.Moghaddam and A.Pentland, "Probabilistic visual learning for obeject representation", IEEE Transaction Pattern Analysis and Machining Intelligence, vol. 19,pp.696-710, July 1997.
10. W.Rudin, Functional Analysis. McGraw-Hill, 1973.
11. G. Donata, M.S.Bartlett, J.C.Hager, P.Ekman, and T.J.Sejnowski, "Classifying facial action", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 21, pp.974-989, Oct.1999.
12. Y.Inada, Y.Xiao, and M.Oda, "facial expression recognition using Vector Matching of Special Frequency Components", IEICE Tech. Rep. TR-90-7,1990.
13. Y.Xiao, N.P.Chandrasiri, Y.Tadokora, and M.Oda, "Recognition of facial expressions using 2-d dct and neural network", Nerual network, vol. 9,no. 7,pp.1233-1240,1996.
14. Hong-Bo Deng,Lian-Wen Jin ,Li-Xin Zhen,Jian-Cheng Huang, "A New Facial Expression Recognition Method Based on Local Gabor Filter Bank and PCA plus LDA", Vol 11, no5, International Journal of Information Technology, 2005
15. K.Liu, Y.-Q.Cheng, J.-Y.Yang, and X.Liu, "An Efficient Algorithm for Foley-Sammon Optimal Set of Discriminant Vectors by Algebraic Method", Int'l J. Pattern Recognition and Artificial Intelligence, vol. 6, no. 5, pp. 817-829, 1992.
16. P.N.Belhumeur, J.P.Hespanha, and D.J.Kriegman, "Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection", IEEE Trans. Pattern Anal. Machine Intell., vol. 19, pp. 711-720, July 1997.
17. D.L.Swets and J.Weng, "Using Discriminant Eigenfeatures for Image Retrieval", IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 18, no.8, pp.831-836, Aug.1996.
18. Lyons M J, Budynek J, Akamatsu S. Automatic Classification of Single Facial Images. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1999, 21(12): 1357-1362.
19. A.M.Martinez and R.Benavente, "The AR-face database",CVC Technical Report #24, June 1998.