Data-Dependent Kernels for High-Dimensional Data Classification

Jingdong Wang James T. Kwok Helen C. Shen Long Quan
Department of Computer Science
The Hong Kong University of Science and Technology
Clear Water Bay
Hong Kong
Email: {welleast,jamesk,helens,quan}@cs.ust.hk

Abstract— **For high-dimensional data classification problems such as face recognition, one of the most efficient classifiers is the Nearest Neighbor (NN) classifier. What mostly affects the NN classification performance is the feature extracted by some methods. And the kernel method is one of the efficient methods for extracting features. However, the selection of kernel parameters is still difficult. In this paper, we propose a so-called data dependent kernel (DDK) which is defined by generalizing the Gaussian kernel. Also an efficient and practical method is presented to calculate the DDK parameters. Moreover, one DDK based on subspaces is given to improve the recognition performance. Experiments show that the proposed DDK can achieve promising classification performance in face recognition and SPECT heart diagnosis.**

I. Introduction

In pattern classification, one of the most difficult problems is high-dimensional data classification problem with a small number of training samples, such as face recognition in which there are only several instances of high dimension for each subject. Some powerful classifiers, which usually require a large number of training samples compared with the dimension, may not obtain satisfactory performance. In practice, the Nearest Neighbor classifier can achieve promising performance. Recently, extracting and selecting features for NN classifiers have become a hot topic in high-dimensional data classification problems.

One classical algorithm is the Principal Component Analysis (PCA) [1]. PCA is to extract the features as the projections on the principal subspace whose basis vectors correspond to the maximum variance directions in the original space, while discard the complementary subspace as a noise subspace. In some cases, PCA can obtain satisfactory performance. However, no theory can prove the complementary subspace is useless for recognition, and, on the contrary, experiments show that using the complementary subspace properly may improve recognition performance.

There are other component analysis methods, such as Linear Discriminant Analysis (LDA) [2], and Independent Component Analysis (ICA) [3]. However, both methods usually use PCA dimension reduction as the preprocessing step. Therefore, the two methods still discard the so-called noise subspace. To efficiently utilize the entire space, Wang and Tang [4] proposed the technique of random sampling the two base vectors on the principal and complementary subspaces for face recognition, which, however, in some sense, causes over-selecting features.

Recently, kernel methods, such as kernel PCA [5], kernel LDA [6] and kernel ICA [7], have been introduced to extract features for recognition. However, kernel parameter selection is difficult. One method is by trial-and-error heuristics, which is easy to implement but not efficient and also causes overfitting problem. The second is using boosting method [8] to learn the combination of kernel functions with different kernel types or different kernel parameters. In [9] one transformed kernel function is discussed, which is good in theory but can not give an easy and efficient way to obtain the transformation matrix.

The goal of this paper is to give an efficient and convenient approach to extract features for high-dimensional data classification problems by generalizing the Gaussian kernel function. We analyze the NN classifier for high-dimensional data classification problems. To obtain better performance, we generalize the Gaussian Kernel to the so-called Data Dependent kernel, which can be easier to calculate compared with the invariant kernel in [9] and obtain better performance than conventional Gaussian kernel and Bayesian Face Matching method [10]. Moreover, we explain why the specified DDK based on subspaces works well.

The paper is organized as follows. Section II analyzes the NN classifier for the high dimensional data classification problem and reviews the data dependent kernel in unsupervised problems. In section III, the proposed data dependent kernel is presented. Section IV presents the comparisons with other related methods. Experiment results are given in section V. Final section is about the conclusion.

II. Background

A. Nearest Neighbor Classifier in High-Dimensional Classification Problems

In pattern recognition, low-dimensional problems with large scale seem easy to process except sometimes the large computation cost. However, few algorithms can well deal with high-dimensional classification problems with small scale, especially when the samples in the different classes are much similar. For example, in face recognition, the faces of different subjects seems also very similar, but the between-class difference is much different from the within-class difference.

0-7803-9048-2/05/$20.00 ©2005 IEEE

How do we use the property? One method is LDA, which usually suffers from the singularity problem. Another method is based on kernels, which selects a proper kernel parameter according to the property. Both methods usually use the NN classifier. Consider the NN classifier with Euclidean distance measure, and we can get at least two observations: (1) NN actually considers the differences rather than the original samples; (2) To improve the recognition performance, it is important to choose some features to discriminate the with- and between-class differences so as to reduce the within-class distance and increase the between-class distance. One straightforward method is to discover two complementary subspaces: one (called the principal subspace) only for within-class differences and the other (called the complementary) only for between-class differences. Then, by assigning less weight on the principal subspace while more weight on the complementary subspace, we can obtain a better distance measure. In this paper, we will incorporate this idea into kernel design to obtain an efficient data-dependent kernel based on subspace.

B. Data-Dependent Kernels in Unsupervised Learning

Kernel methods have been successfully applied in many problems. Its basic idea is to implicitly map the data from the input space $\mathcal{X}$ to a high-dimensional feature space $\mathcal{H}$ via a nonlinear function $\phi : \mathcal{X} \to \mathcal{H}$, and then a similarity measure in $\mathcal{H}$ is defined as the dot product:

$$\mathbf{k}(\mathbf{x}, \mathbf{x}') \equiv \langle \phi(\mathbf{x}) \cdot \phi(\mathbf{x}') \rangle .$$

Here, the kernel function $\mathbf{k}$ should satisfy *Mercer condition* [11].

However, one major question is how to choose $\mathbf{k}(\cdot, \cdot)$ and its associated kernel parameters. One approach to alleviate this problem is using the data-dependent kernel. The concept of data-dependent kernels is first introduced in [12], as shown in Table I. In that paper, the data-dependent kernel is used in unsupervised problems, such as nonlinear dimension reduction and clustering. In spectral clustering [13], one divisive-normalized kernel is proposed, where the divisive coefficient is determined by all the data. The divisive-normalized kernel is proved to help clustering the data. In Isometric Feature Mapping (Isomap) [14], they used so-called geodesic distance kernel to keep the geodesic distance of the data manifold to reduce the dimension. In Local Linearly Embedding (LLE) [15], they use the kernel to keep the local linear reconstruction. All the methods are applied in unsupervised problems.

TABLE I
DATA-DEPENDENT KERNELS IN UNSUPERVISED LEARNING.

Method	Data-dependent Kernel
KPCA	Centralization Normalization
Isomap	Isomap Kernel
LLE	LLE Kernel
Spectral Clustering	Division Normalization

Actually, the data-dependent kernel can also be used in supervised problems. For example, spectral clustering has been used in recognition. In this paper, we want to design special data dependent kernel for high-dimensional classification problems based on the Gaussian Kernel.

III. Data Dependent Kernels for Classification

In this section, we start with the generalized Gaussian kernel:

$$\mathbf{k}(\mathbf{x}, \mathbf{y}) = \exp\{-\frac{1}{2}(\mathbf{x} - \mathbf{y})' \mathbf{H}^{-1} (\mathbf{x} - \mathbf{y})\}, \tag{1}$$

where $\mathbf{H}$ is a transformation matrix and $'$ denotes the transpose operation. In cases where certain input transformations are known to leave function values unchanged, the use of $\mathbf{H}$ can also allow such invariance to be incorporated into the kernel function [9]. In the following, we propose several methods for obtaining $\mathbf{H}$ from the training data, and the corresponding kernels are called data dependent kernels $\mathbf{k}_d(\mathbf{x}, \mathbf{y})$.

A. Using the Covariance Matrix

An obvious choice for $\mathbf{H}$ is the $d \times d$ covariance matrix:

$$\mathbf{H} = E[(\mathbf{x} - \bar{\mathbf{x}})(\mathbf{x} - \bar{\mathbf{x}})'], \tag{2}$$

where $\mathbf{x} \in \mathbb{R}^d$ is the data vector and $\bar{\mathbf{x}}$ the corresponding mean vector. By assuming that the entries of the sample vector are independent, (2) can be simplified by dropping the off-diagonal elements as:

$$\mathbf{H} = \text{Diag}[\text{diag}[E[(\mathbf{x} - \bar{\mathbf{x}})(\mathbf{x} - \bar{\mathbf{x}})']]].$$

Here, $\text{diag}(\cdot)$ extracts the diagonal elements of (2), which are then used to construct a diagonal matrix by $\text{Diag}(\cdot)$, and the obtained kernel by this matrix $\mathbf{H}$ is called Independent DDK.

For high-dimensional training sets with small scale, such an $\mathbf{H} = \text{Diag}(\mathbf{h}_1, \cdots, \mathbf{h}_d)$ obtained[1] from the empirical data is often singular, as the data dimensionality is larger than the sample size. To avoid this problem, we replace the $d - p$ smallest diagonal elements in $\mathbf{H}$ by their average, i.e.,

$$\hat{\mathbf{H}} = \begin{bmatrix} \mathbf{H}_p & \\ & \rho \mathbf{I}_{d-p} \end{bmatrix}. \tag{3}$$

Here, $p \in \{0, 1, \ldots, d\}$ is a user-defined parameter,

$$\rho = \frac{1}{d-p} \sum\nolimits_{i=p+1}^{d} \mathbf{h}_i, \tag{4}$$

$\mathbf{H}_p = \text{Diag}(\mathbf{h}_1, \cdots, \mathbf{h}_p)$ and $\mathbf{I}_{d-p}$ is the $(d-p) \times (d-p)$ identity matrix. (4) can be justified from an information-theoretic point of view. Details can be found in [10].

Notice that when p is set to zero, $\mathbf{H}$ is of the form $\mathbf{H} = \sigma^2 \mathbf{I}$ (where σ is a measure of the data spread) and (1) reduces to the conventional Gaussian kernel:

$$\mathbf{k}(\mathbf{x}, \mathbf{y}) = \exp\{-\frac{\|\mathbf{x} - \mathbf{y}\|^2}{2\sigma^2}\} \tag{5}$$

[1]Here, we assume that the entries of the sample vector have been permutated such that $\mathbf{h}_1 \geq \mathbf{h}_2 \geq \cdots \geq \mathbf{h}_d$.

While (5) implicitly assumes the same variance for each entry of the sample vector (i.e., the data is isotropic, and accordingly the kernel is called Isotropic DDK when σ^2 is calculated as in (4)), this is not necessary for the generalized Gaussian kernel.

In a classification problem, a major deficiency of the $\mathbf{H}$'s defined in (2) and (3) is that they do not utilize the class labels. Thus, they are not discriminative in nature. As will be demonstrated in Section V, experimental results obtained with the corresponding kernels are not satisfactory in practice.

B. Subspace-Based Data Dependent Kernel

As discussed in Section II, it is desirable to have a kernel such that the corresponding intra-class (within-class) difference is reduced while the inter-class (between-class) difference is increased. In this paper, we adopt the subspace method, and use one subspace for the intra-class difference and another for the inter-class difference. Different distance measures are then defined on these two complementary subspaces. Here, the two subspaces are obtained as follows:

1) For each class, obtain all the intra-class differences $\{\mathbf{x}_i^c - \mathbf{x}_j^c\}_{i,j}$, where $\mathbf{x}_i^c, \mathbf{x}_j^c$ are samples from the same class c.
2) Pool the intra-class differences from all classes together, and then perform PCA.
3) The principal subspace is used to represent the intra-class difference, while the remaining subspace for the inter-class difference.

Note that as class information is used, the resulting kernel, called Intra-DDK, is discriminative.

PCA in Step 2 involves eigendecomposition on the $d \times d$ matrix

$$\mathbf{H} = \sum_{c=1}^{C} \mathbf{H}_c \tag{6}$$

where $\mathbf{H}_c = \sum_{i,j=1}^{N_c} (\mathbf{x}_i^c - \mathbf{x}_j^c)(\mathbf{x}_i^c - \mathbf{x}_j^c)'$, $\mathbf{x}_i^c, \mathbf{x}_l^c$'s are samples from class c, N_c is the number of patterns belonging to class c, and C is the total number of classes. As d is assumed to be large here, so, instead of eigendecomposing (6), a common trick is to perform eigendecomposition on the $\frac{1}{2}\sum_{c=1}^{C} N_c(N_c-1) \times \frac{1}{2}\sum_{c=1}^{C} N_c(N_c-1)$ matrix

$$\begin{bmatrix} \vdots \\ (\mathbf{x}_i^c - \mathbf{x}_j^c)' \\ \vdots \end{bmatrix} \begin{bmatrix} \cdots (\mathbf{x}_k^c - \mathbf{x}_l^c) \cdots \end{bmatrix}.$$

However, this is still more computationally expensive than performing standard PCA on the whole data set, which only involves a matrix of size $\sum_{c=1}^{C} N_c \times \sum_{c=1}^{C} N_c$.

To improve efficiency, instead of (6), we will use

$$\mathbf{H} = \sum_{c=1}^{C} \mathbf{S}_c, \tag{7}$$

where $\mathbf{S}_c = \sum_{i=1}^{N_c} (\mathbf{x}_i^c - \boldsymbol{\mu}_c)(\mathbf{x}_i^c - \boldsymbol{\mu}_c)'$ is the covariance matrix for class c (with $\boldsymbol{\mu}_c$ being the mean of class c). Now, it can be easily proved that

$$\mathbf{S}_c = \frac{1}{2N_c}\mathbf{H}_c.$$

and so (6) and (7) only differ by the term $\frac{1}{2N_c}$. Let $\mathbf{X}_c = \{\frac{1}{\sqrt{2N_c}}(\mathbf{x}_1^c - \boldsymbol{\mu}_c), \cdots, \frac{1}{\sqrt{2N_c}}(\mathbf{x}_{N_c}^c - \boldsymbol{\mu}_c)\}$. Therefore, we can perform PCA on $\mathbf{X} = \{\mathbf{X}_1, \cdots, \mathbf{X}_C\}$ instead of Step 2, which will only require eigendecomposing a matrix of size $\sum_{c=1}^{C} N_c \times \sum_{c=1}^{C} N_c$.

As mentioned earlier, the leading p-dimensional subspace is used to encode the intra-class difference, while the remaining subspace is for the inter-class difference. To avoid the problem of having a singular matrix, we use the same technique as in Section III-A. Suppose that the eigendecomposition of $\mathbf{H}$ is $\mathbf{H} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}'$, where $\boldsymbol{\Lambda} = \mathrm{Diag}(\lambda_1, \ldots, \lambda_d)$ is the diagonal matrix containing the eigenvalues of $\mathbf{H}$ and $\mathbf{U}$ is the matrix containing the corresponding eigenvectors. The data dependent kernel matrix $\hat{\mathbf{H}}$ is then defined as

$$\hat{\mathbf{H}} = \mathbf{U} \begin{bmatrix} \boldsymbol{\Lambda}_p & \\ & \rho_{d-p} \end{bmatrix} \mathbf{U}', \tag{8}$$

where $p \in \{0, 1, \ldots, d\}$, $\boldsymbol{\Lambda}_p = \mathrm{Diag}(\lambda_1, \cdots, \lambda_p)$, and

$$\rho = \frac{1}{d-p}\sum_{i=p+1}^{d} \lambda_i.$$

IV. Discussion

A. Relationship to Bayesian Face Matching and Relevant Component Analysis

Matrix $\mathbf{H}$ in the subspace-based data dependent kernel is obtained in a similar way as the Bayesian face matching (BFM) algorithm [10] and relevant component analysis (RCA) [16]. In some sense, this paper can be viewed as combining BFM or RCA into kernel PCA. In this paper, (1) the matrix $\mathbf{H}$ in Section III-B is derived from the insight that there exist two subspaces to represent the intra-class and inter-class differences, while the authors in [17] gives the intuitive interpretation. (2) In Section III-B, the principal and complementary subspaces from $\mathbf{H}$ are assigned different weights, while the complementary subspace in RCA is discarded. (3) We give an effective method to calculate the matrix $\mathbf{H}$ and eigendecompose $\mathbf{H}$. (4) We combine the matrix $\mathbf{H}$ into the generalized Gaussian kernel to obtain better recognition performance.

B. Comparison with Invariant Kernels

The invariant kernels in [9] is also closely related to the proposed data-dependent kernels. In [9], PCA is performed on the tangent vector set as a pre-processing step. Let the tangent covariance matrix be $\mathbf{C}$ and $\mathbf{B} = \mathbf{C}^{-\frac{1}{2}}$. Then, the invariant kernel is:

$$\mathbf{k}(\mathbf{x}, \mathbf{y}) = \langle \mathbf{B}\mathbf{x}, \mathbf{B}\mathbf{y} \rangle .$$

In essence, our data-dependent kernel also finds the matrix $\mathbf{B}$, though there are some important differences. On the one hand, our data-dependent kernel provides an easy and feasible way to calculate $\mathbf{B}$, while the work in [9] requires prior knowledge

on the tangent vectors. On the other hand, the subspace based data-dependent kernel, which is derived from the insight that there exist two subspaces to represent the intra-class and inter-class differences, is designed specially for high-dimensional classification problems and in some sense can also be viewed as a discriminative kernel.

V. Experiments

Section V-A demonstrates the difference between the data-dependent kernel based on subspaces and the conventional Gaussian kernel (i.e., Isotropic DDK). In the later subsections, the performance of the proposed data-dependent kernels is then evaluated on two face recognition problems and non-image data sets. All the proposed data-dependent kernels are used in kernel PCA to extract low-dimensional features. Then the low-dimensional features are used in the Nearest Neighbor classifier to classify the test samples. The detailed algorithm is in Table II. The corresponding data dependent kernel PCAs are called Isotropic-KPCA, Independent-KPCA and Intra-KPCA.

TABLE II

Classification using the data dependent kernel.

Step 1. Compute the matrix $\mathbf{H}$ of the data dependent kernel.

Step 2. Compute the kernel matrix $\mathbf{K}_{ij} = \mathbf{k}_d(\mathbf{x}_i, \mathbf{x}_j)$ using the data dependent kernel

Step 3. Solve

$$M\lambda\alpha = \mathbf{K}\alpha,$$

where M is the number of training samples, then normalize the eigenvector expansion coefficients. Let $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_M$ denote the eigenvalues of $\mathbf{K}$ and $\alpha^1, \cdots, \alpha^M$ the corresponding eigenvectors. Normalize the coefficients by requiring $\lambda_i \langle \alpha^i \cdot \alpha^i \rangle = 1$.

Step 4. Extract d principal components $\mathbf{f_x}$ (corresponding to the data dependent kernel) of the training point $\mathbf{x}$ by computing the projections onto the the first p eigenvectors as

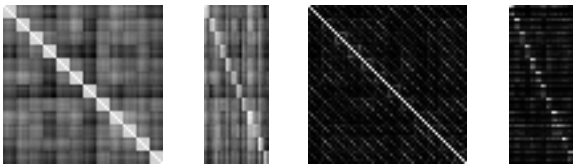
$$\langle w^n \cdot \phi(\mathbf{x}) \rangle = \sum_{i=1}^{M} \alpha_i^n \mathbf{k}_d(\mathbf{x}_i, \mathbf{x}).$$

Step 5. Extract the features $\mathbf{f_t}$ of the test point $\mathbf{t}$ as in step 4.

Step 6. Obtain the same classification label with the training sample whose feature $\mathbf{f_x}$ has the shortest distance to $\mathbf{f_t}$.

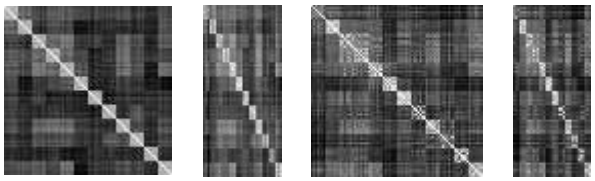
A. Demonstration

Figures 1 and 2 show the training kernel matrix $\mathbf{K}_{tr}$ and the similarity matrix $\mathbf{K}_s$ between the training and testing patterns on AR[2] [18] and ORL[3] face databases. We observe that the within-class similarity is larger than the between-class similarity in both the two matrices (The gray of the pixel represents the similarity of the associated two samples.). If we observe carefully the images, the within-class samples using Intra-DDK are more similar than using Isotropic-DDK, while the between-class samples using Intra-DDK are less similar than using Isotropic-DDK. From this sense, the Intra-DDK is more discriminative than Isotropic-DDK.



(a) $\mathbf{K}_{tr}$ (b) $\mathbf{K}_s$ (c) $\mathbf{K}_{tr}$ (d) $\mathbf{K}_s$

Fig. 1. The training kernel and similarity matrices on the AR database using Intra-DDK ((a), (b)) and Isotropic-DDK ((c), (d)).



(a) $\mathbf{K}_{tr}$ (b) $\mathbf{K}_s$ (c) $\mathbf{K}_{tr}$ (d) $\mathbf{K}_s$

Fig. 2. The training kernel and similarity matrices on the ORL database using Intra-DDK ((a), (b)) and Isotropic-DDK ((c), (d)).

B. SPECT Heart Diagnosis

In this section, experiments are performed on the SPECT/SPECTF heart diagnosis data sets[4] from the UCI Machine Learning Repository. Both are on diagnosing of cardiac Single Proton Emission Computed Tomography (SPECT) images. Each contains 267 samples, which are represented by attributes summarizing the original SPECT images. The SPECT data has 22 binary attributes, whereas the SPECTF data has 44 continuous attributes. The task is to diagnose whether the heart in each patient or image is normal or not. The training set contains 80 samples (40 samples in each category), and the test set contains 187 samples.

For comparison with the proposed data-dependent kernel, we also run the following methods:

1) 1-nearest-neighbor (1-NN) classifier;
2) PCA; and
3) SVM.

For PCA, the dimensionality of the principal subspace is set to 10. The kernel parameter in SVM is well-tuned. Table III shows the accuracy. The known best result on this task, which is achieved by CLIP4 (Cover Learning using Integer Programming) and the ensemble of CLIP4 [19], is shown in the table for comparison.

As can be seen, the proposed data-dependent kernels (Isotropic, Independent, Intra-KPCA) obtain promising results, though not the best, on the SPECT data set. They get the best result on the SPECTF data set.

[2]http://rvl1.ecn.purdue.edu/~aleix/ar.html.

[3]http://www.uk.research.att.com/facedatabase.html.

[4]http://www.ics.uci.edu/~mlearn/MLSummary.html.

TABLE III

CLASSIFICATION ACCURACY ON THE SPECT TASK.

METHOD	SPECT	SPECTF
1-NN	80.2	72.1
PCA	73.3	75.1
SVM	89.9	93.7
[5]CLIP4	**90.4**	77.0
ISOTROPIC-KPCA	88.7	**94.4**
INDEPENDENT-KPCA	89.9	**94.4**
INTRA-KPCA	89.9	**94.4**

C. AR Database

In this subsection, experiments are performed on the AR-face database, which consists of over 3200 color images of the frontal faces of 126 subjects. There are 26 different images for each subject. For each subject, these images were recorded in two different sessions separated by two weeks, each session consisting of 13 images. Each image is of size 768×576.

We choose the first 7 face images of the first session by eliminating occluded face images for each subject. Then, we have 126×7 face images. We manually locate the centers of the eyes and then perform geometric normalization with the eye locations fixed to get geometric normalized face image with size 24×18. Examples of the normalized faces are shown in Figure 3.

Fig. 3. Example face images from the AR database.

In the experiment, we perform ten trials by randomly selecting five faces for training and two for testing (for each subject) in each trial. Results are then averaged over 10 trials. For comparison, we also show the results of 1-NN, PCA and Bayesian face recognition method. As can be seen in Table IV with the accuracy and the standard deviation, Intra-KPCA obtains the best recognition result.

TABLE IV

CLASSIFICATION ACCURACIES ON THE AR DATABASE.

METHOD	ACCURACY
1-NN	85.54 ± 3.4
PCA	87.88 ± 3.1
BAYESIAN	94.08 ± 2.4
ISOTROPIC-KPCA	83.96 ± 3.5
INDEPENDENT-KPCA	83.12 ± 3.8
INTRA-KPCA	$\mathbf{94.67} \pm 2.3$

TABLE V

CLASSIFICATION ACCURACIES ON THE ORL DATABASE.

METHOD	ACCURACY
1-NN	97.50 ± 1.0
PCA	94.16 ± 1.4
BAYESIAN	96.92 ± 1.5
ISOTROPIC-KPCA	97.17 ± 1.0
INDEPENDENT-KPCA	97.08 ± 1.2
INTRA-KPCA	$\mathbf{97.92} \pm 1.3$

D. ORL Database

The ORL (Olivetti Research Laboratory) face database contains a set of face images taken between April 1992 and April 1994 at the Olivetti Research Laboratory. There are ten different images of each of 40 distinct subjects (Figure 4). For some subjects, the images were taken at different times, varying the lighting, facial expressions (open/closed eyes, smiling/not smiling) and facial details (glasses/no glasses). All the images were taken against a dark homogeneous background with the subjects in an upright, frontal position (with tolerance for some side movement). The original image size is 112×92. For convenience, they are downsampled to 28×23.

Fig. 4. Example face images from the ORL database.

We perform ten trials by randomly selecting seven faces for training and three for testing (for each subject) in each trial. Results are then averaged over 10 trials. For comparison, we also show the results on 1-NN, PCA, Bayesian face recognition method's results. As can be seen in Table V, Intra-KPCA again obtains the best recognition result.

VI. CONCLUSION

This paper addressed the data-dependent kernels for high-dimension classification problems, which can be efficient to calculate and would improve the recognition performance. We give and analyze the different candidates for the matrix $\mathbf{H}$ in data dependent kernels. The experiments show that the proposed data dependent kernels especially subspace based DDK can get better recognition performance.

REFERENCES

[1] I. Jolliffe, *Principle Component Analysis*. New York: Springer-Verlag, 1986.

[2] W. Zhao, R. Chellappa, and P. Phillips, "Subspace linear discriminant analysis for face recognition," Tech. Rep. CAR-TR-914, Center for Automation Research, University of Maryland, College Park, 1999.

[3] A. Hyvärinen, "Survey on independent component analysis," *Neural Computing Surveys*, vol. 2, pp. 94–128, 1999.

[4] X. Wang and X. Tang, "Random sampling LDA for face recognition.," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. 259–265, 2004.
[5] B. Schölkopf, A. Smola, and K. Müller, "Nonlinear component analysis as a kernel eigenvalue problem," *Neural Computation*, vol. 10, pp. 1299–1319, 1998.
[6] S. Mika, G. Rätsch, J. Weston, B. Schölkopf, and K.-R. Müller, "Fisher discriminant analysis with kernels," in *Neural Networks for Signal Processing IX* (Y.-H. Hu, J. Larsen, E. Wilson, and S. Douglas, eds.), pp. 41–48, IEEE, 1999.
[7] F. Bach and M. Jordan, "Kernel independent component analysis," *Journal of Machine Learning Research*, vol. 3, pp. 1–48, 2002.
[8] K. Crammer, J. Keshet, and Y. Singer, "Kernel design using boosting," in *Advances in Neural Information Processing Systems 15* (S. T. S. Becker and K. Obermayer, eds.), pp. 537–544, Cambridge, MA: MIT Press, 2003.
[9] B. Schölkopf, P. Simard, A. J. Smola, and V. Vapnik, "Prior knowledge in support vector kernels.," in *NIPS*, 1997.
[10] B. Moghaddam and A. Pentland, "Probabilistic visual learning for object representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 696–710, 1997.
[11] V. Vapnik, *The nature of statistical learning theory*. Statistics for Engineering and Information Science, Berlin: Springer Verlag, 2000.
[12] Y. Bengio, O. Delalleau, N. L. Roux, J.-F. Paiement, P. Vincent, and M. Ouimet, "Learning eigenfunctions links spectral embedding and kernel pca.," *Neural Computation*, vol. 16, no. 10, pp. 2197–2219, 2004.
[13] A. Ng, M. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," 2001.
[14] J. B. Tenenbaum, V. d. Silva, and J. C. Langford, "A Global Geometric Framework for Nonlinear Dimensionality Reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, 2000.
[15] S. T. Roweis and L. K. Saul, "Nonlinear Dimensionality Reduction by Locally Linear Embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, 2000.
[16] N. Shental, T. Hertz, D. Weinshall, and M. Pavel, "Adjustment learning and relevant component analysis," in *Proceedings of the Seventh European Conference on Computer Vision*, vol. 4, (Copenhagen, Denmark), pp. 776–792, 2002.
[17] X. Wang and X. Tang, "Unified subspace analysis for face recognition.," in *ICCV*, pp. 679–686, 2003.
[18] A. Martinez and R. Benavente, "The AR face database," Tech. Rep. CVC 24, 1998.
[19] L. A. Kurgan, K. J. Cios, R. Tadeusiewicz, M. R. Ogiela, and L. S. Goodenday, "Knowledge discovery approach to automated cardiac spect diagnosis.," *Artificial Intelligence in Medicine*, vol. 23, no. 2, pp. 149–169, 2001.