2013 IEEE 13th International Conference on Data Mining

Feature Transformation with Class Conditional Decorrelation

Xu-Yao Zhang† Kaizhu Huang‡ Cheng-Lin Liu†
† NLPR, Institute of Automation, Chinese Academy of Sciences. {xyz, liucl}@nlpr.ia.ac.cn
‡ Dept. of EEE, Xi'an Jiaotong-Liverpool University, China. kaizhu.huang@xjtlu.edu.cn

***Abstract*—The well-known feature transformation model of Fisher linear discriminant analysis (FDA) can be decomposed into an equivalent two-step approach: whitening followed by principal component analysis (PCA) in the whitened space. By proving that whitening is the optimal linear transformation to the Euclidean space in the sense of minimum log-determinant divergence, we propose a transformation model called class conditional decorrelation (CCD). The objective of CCD is to diagonalize the covariance matrices of different classes simultaneously, which is efficiently optimized using a modified Jacobi method. CCD is effective to find the common principal components among multiple classes. After CCD, the variables become class conditionally uncorrelated, which will benefit the subsequent classification tasks. Combining CCD with the nearest class mean (NCM) classification model can significantly improve the classification accuracy. Experiments on 15 small-scale datasets and one large-scale dataset (with 3755 classes) demonstrate the scalability of CCD for different applications. We also discuss the potential applications of CCD for other problems such as Gaussian mixture models and classifier ensemble learning.**

***Keywords*-class conditional decorrelation, simultaneous diagonalization, feature transformation.**

I. Introduction

In pattern classification for high-dimensional data, feature transformation is widely applied as a pre-processing technique. Feature transformation can reduce the computational complexity by dimensionality reduction, and also obtain better generalization performance by reducing irrelevant and redundant information in data, overcoming the estimation problem in statistical classifier learning, and revealing the latent structure of data.

Feature transformation can be divided into linear and nonlinear methods. A large variety of linear methods, such as random projection (RP) [1], principal component analysis (PCA) [20], Fisher linear discriminant analysis (FDA) [15], independent component analysis (ICA) [19], non-negative matrix factorization (NMF) [23], and locality preserving projections (LPP) [17], have been proposed from different statistical or geometrical viewpoints. The nonlinear methods include: (i) kernel extension of the linear methods, such as kernel PCA [33] and kernel FDA [38]; (ii) manifold learning models such as ISOMAP [36], LLE [32] and Laplacian eigenmaps [2]; (iii) deep neural networks [18], [34] which use a deep architecture to learn the nonlinear data mapping.

Fisher linear discriminant analysis (FDA) [15] is one of the most famous linear supervised algorithms. The principle of FDA is to minimize the within-class variance as well as maximize the between-class variance. Under the homoscedastic Gaussian assumption, FDA leads to the optimal projection axes when the reduced dimensionality is $K-1$ (K is the number of classes). However, in many other occasions, FDA is only a suboptimal model, e.g., many models have been proposed: (i) to solve the class separation problem when the reduced dimensionality is much smaller than the number of classes [27], [35], [3], [40]; (ii) to improve FDA under the heteroscedastic case [26], [42]; and (iii) to alleviate the small sample size problem [7], [39].

In this paper, we first decompose FDA into an equivalent two-step approach: whitening and PCA in the whitened space. By proving that whitening is the optimal linear transformation to transform the covariance matrices of different classes to the identity matrices, we further propose a new model called class conditional decorrelation (CCD). The objective of CCD is to learn a linear transformation attempting to diagonalize all the covariance matrices for each class simultaneously. CCD is more flexible than whitening and can find the common principal components among multiple classes [14]. Furthermore, the modified Jacobi method is used to solve the optimization problem of CCD efficiently.

After CCD transformation, the variables become class conditionally uncorrelated. This will benefit the following classification tasks. In this paper, we combine CCD with the nearest class mean (NCM) classification model to learn an improved distance metric. The original NCM classifiers are inferior to other discriminative classifiers, however, with the help of CCD, the improved NCM classifiers can achieve comparable performance with other benchmark classifiers such as the NCM metric learning method [29]. Experiments on 15 small-scale datasets and one large-scale dataset (with $3,755$ classes) demonstrate that: the structure of latent common principal components for multiple classes exists not only in small category problems but also in large category problems. CCD is effective in finding such structure and improving the classification performance. Besides the applications of CCD in this paper, we also extend CCD into a much more generalized formulation and show the potential advantages of CCD for other problems such as Gaussian mixture models and classifier ensemble learning.

The rest of this paper is organized as following: Section II introduces the decomposition of FDA; Section III proves that whitening is the optimal linear transformation to the

1550-4786/13 $31.00 © 2013 IEEE
DOI 10.1109/ICDM.2013.43

Euclidean space; Section IV presents the proposed model of class conditional decorrelation (CCD); Section V describes the combination of CCD for nearest class mean (NCM) classification; Section VI reports the experimental results; Section VII offers some potential extensions of CCD; and Section VIII draws the concluding remarks.

II. Decomposition of FDA

Let $\mu_k \in \mathbb{R}^d$ be the mean vector and $\Sigma_k \in \mathbb{R}^{d\times d}$ be the covariance matrix for class k ($k = 1, \ldots, K$). The within-class and between-class scatter matrices are defined as:

$$S_w = \frac{1}{K}\sum_{k=1}^{K}\Sigma_k \,, \tag{1}$$

$$S_b = \frac{1}{K}\sum_{k=1}^{K}(\mu_k - \mu_0)(\mu_k - \mu_0)^\top \,, \tag{2}$$

where $\mu_0 = \frac{1}{K}\sum_{k=1}^{K}\mu_k$, and we assume the prior probabilities are equal for all the classes. The objective of FDA is to learn a transformation matrix $W \in \mathbb{R}^{d\times d'}$ to transform the feature into a low-dimensional space $x' = W^\top x$ by minimizing the within-class variance as well as maximizing the between-class variance. It is easy to check that the scatter matrices in the transformed space become $W^\top S_w W$ and $W^\top S_b W$. There are many formulations of FDA, and two typical criteria are given in the following [15]:

$$\max_W \ \mathrm{tr}\left\{\left(W^\top S_w W\right)^{-1}\left(W^\top S_b W\right)\right\} \,, \tag{3}$$

$$\max_W \ \ln\left|W^\top S_b W\right| - \ln\left|W^\top S_w W\right| \,. \tag{4}$$

The above two criteria are equivalent to a constrained problem:

$$\begin{aligned}\max_{W\in\mathbb{R}^{d\times d'}} \quad & \mathrm{tr}\left(W^\top S_b W\right) \,, \\ \text{s.t.} \quad & W^\top S_w W = I \,,\end{aligned} \tag{5}$$

where I is the identity matrix. Usually, this model is solved by a two-step approach. The first step is the whitening.

Definition 1. *The whitening transformation matrix is*

$$W_{\text{whiten}} = P\Lambda^{-1/2} \in \mathbb{R}^{d\times d} \,, \tag{6}$$

where P is the eigenvector matrix and Λ is the diagonal eigenvalue matrix of the within-class scatter matrix: $S_w = P\Lambda P^\top$. The whitening transformation satisfies

$$W_{\text{whiten}}^\top S_w W_{\text{whiten}} = I \,. \tag{7}$$

The whitening transformation is to transform the within-class scatter matrix into the identity matrix, after that the Euclidean distance becomes a suitable measurement between different classes.

Let $W_{\text{FDA}} = W_{\text{whiten}}W$, we can rewrite FDA of (5) as:

$$\begin{aligned}\max_{W\in\mathbb{R}^{d\times d'}} \quad & \mathrm{tr}\left(W^\top W_{\text{whiten}}^\top S_b W_{\text{whiten}} W\right) \,, \\ \text{s.t.} \quad & W^\top W = I \,.\end{aligned} \tag{8}$$

Hence the second step of FDA is to solve (8). This is exactly the PCA among $W_{\text{whiten}}^\top\mu_1, \ldots, W_{\text{whiten}}^\top\mu_K$. That means FDA is equivalent to whitening followed by PCA of the class-means on the whitened space.

III. Whitening

In this section, we show that, in an information-theoretic viewpoint [10], whitening is the optimal linear transformation to the Euclidean space, which implies the advantages of the two-step approach of FDA.

Theorem 1. *The whitening transformation of (6) minimizes the Log-Determinant divergence between the transformed covariance $W^\top\Sigma_k W$ and the identity matrix I.*

Proof: The Log-Determinant divergence [11] between two $n\times n$ matrices is defined as:

$$D_{ld}(X, Y) = \mathrm{tr}(XY^{-1}) - \log\det(XY^{-1}) - n \,. \tag{9}$$

In this paper, we define the objective of whitening (6) as:

$$\begin{aligned}&\min_{W\in\mathbb{R}^{d\times d}} \quad \mathcal{F} = \sum_{k=1}^{K} D_{ld}\left(W^\top\Sigma_k W, I\right) \\ &= \sum_{k=1}^{K}\left[\mathrm{tr}(W^\top\Sigma_k W) - \log\det(W^\top\Sigma_k W) - d\right] \,.\end{aligned} \tag{10}$$

By setting the derivative of the objective function w.r.t. W to zero, we get

$$\frac{\partial\mathcal{F}}{\partial W} = 2\sum_{k=1}^{K}\Sigma_k W - 2KW^{-\top} = 0 \,, \tag{11}$$

which means $S_w W - W^{-\top} = 0$. Insert $S_w = P\Lambda P^\top$ and $W_{\text{whiten}} = P\Lambda^{-1/2}$ into it, we get

$$P\Lambda P^\top P\Lambda^{-1/2} - P\Lambda^{1/2} = 0 \,. \tag{12}$$

Since $P^\top P = I$, this completes the proof. ∎

From Theorem 1, we can conclude that: even when the covariances $\Sigma_k, k = 1, \ldots, K$ are not equal for all the classes (heteroscedastic), whitening transformation is still a good model to find the optimal Euclidean space. In the whitened space, the Euclidean distance becomes a suitable measurement. Therefore each class can be described by $W_{\text{whiten}}^\top\mu_k$, and the PCA transformation among them (8) can find the optimal separated subspace for classification.

IV. Class Conditional Decorrelation

A. Motivation and Definition

The whitening in (10) is to transform $\Sigma_1, \ldots, \Sigma_K$ into the identity matrix. However, in most cases this is impossible. Although the within-class scatter matrix S_w is transformed to the identity matrix in Eq. (7), the covariance matrices of each class $\Sigma_1, \ldots, \Sigma_K$ are still far from the identity matrix due to the divergences among them. In light of this, we consider a relaxed formulation of the whitening. Instead of

transforming $\Sigma_1, \ldots, \Sigma_K$ into the identity matrix, we would like to diagonalize them as much as possible:

$$\begin{aligned} \min_{W \in \mathbb{R}^{d\times d}} \quad & \sum_{k=1}^{K} \left\| W^\top \Sigma_k W \right\|_{2,\text{off}} , \\ \text{s.t.} \quad & W^\top W = I , \end{aligned} \tag{13}$$

where

$$\|A\|_{2,\text{off}} = \sum_{i\neq j} A_{ij}^2 . \tag{14}$$

The constraint $W^\top W = I$ is to avoid trivial solutions. We call this model class conditional decorrelation (CCD), because we try to diagonalize (or decorrelate) simultaneously $\Sigma_1, \ldots, \Sigma_K$, which are the class conditional covariance matrices for each class. After CCD transformation, the variables become class conditionally uncorrelated, which can benefit the subsequent classification tasks.

Lemma 1. *For arbitrary $A \in \mathbb{R}^{d\times d}$, under any full rank orthogonal transformation $W \in \mathbb{R}^{d\times d}, W^\top W = I$, we have*

$$\left\| W^\top A W \right\|_F^2 = \|A\|_F^2 , \tag{15}$$

where $\|A\|_F^2 = \sum_{i,j=1}^{d} A_{ij}^2$.

Because of Lemma 1, by minimizing the sum of the squared non-diagonal elements, the objective of CCD (13) will concentrate its energy on the diagonal elements, which leads to two main advantages. First, CCD is more flexible than whitening: CCD focuses on diagonalizing all the covariance matrices simultaneously, while the objective of whitening is not only diagonalization but further restricts the transformed matrix to be the identity matrix (10). Second, CCD can find the common principal components among multiple classes (known as common PCA [14], [13]). The goal of PCA is to diagonalize the total scatter matrix of all the samples. Furthermore, the goal of CCD is to diagonalize all the class-wise covariance matrices simultaneously. If there is only one class ($K = 1$), CCD is reduced to PCA. For multiple classes, the transformation axes learned by CCD are the common principal components for all the classes. In the following section, we will give an illustration of these two advantages.

B. Illustration of CCD

We use the well-known handwritten digit database MNIST [22] to give an intuitive understanding of CCD. First, the original covariance matrices of the digits are shown in Figure 1[1]. We can see that all the covariance matrices are very dense. For each matrix, the percentage of the sum of the squares of the diagonal elements is defined as:

$$P(A) = \frac{\sum_{i=1}^{d} A_{ii}^2}{\sum_{i,j=1}^{d} A_{ij}^2} . \tag{16}$$

[1]For better visualization, the 3D bar images represent the absolute values of the elements in each covariance matrix.

Because the sum of the squares of all the elements will not change under any orthogonal transformation (Lemma 1), $P(\cdot)$ is a good metric to measure the concentration of the energy on the diagonal elements. We show the $P(\cdot)$ of each covariance matrix above the image. We can find that the percentages $P(\cdot)$ are all below 10%, which indicates the non-diagonal elements are very dense. After whitening, the covariance matrices are shown in Figure 2. Although the covariance matrices after whitening are much like the identity matrices, the non-diagonal elements of them are still very dense. Furthermore, with class conditional decorrelation (CCD), the covariance matrices of the digits are shown in Figure 3. We can find that: after CCD transformation, the covariance matrix of each class becomes very sparse, i.e., most of the non-diagonal elements become very small. The percentages $P(\cdot)$ are dramatically increased. Moreover, the diagonal elements are also sparse, for which the largest elements are concentrated in the first few diagonal elements, and the remaining diagonal elements are very small. Since the largest diagonal elements means the principal components of a particular class, this indicates the advantages of CCD on finding the common principal components among multiple classes.

C. Optimization

In this section, we dive into solving the optimization problem of CCD (13). This problem is known as simultaneous diagonalization [43] which cannot be solved by the eigenvalue decomposition algorithm such as whitening (6). Therefore, we adopt the modified Jacobi method to solve the CCD problem. The original Jacobi method is very effective to find the eigenvectors and eigenvalues of a single symmetric matrix [16], and has been successfully used for sparse high-dimensional covariance matrix estimation [5], [4]. To deal with multiple symmetric matrices, the modified Jacobi method [6], [28] can be used efficiently and effectively.

1) Jacobi Rotation: A basic Jacobi plane rotation $R(i, j, \theta) \in \mathbb{R}^{d\times d}$ $(i \neq j)$ is defined as

$$[R(i,j,\theta)]_{pq} = \begin{cases} 1, & p = q, p \neq i, p \neq j \\ \cos\theta, & p = q = i \\ \cos\theta, & p = q = j \\ \sin\theta, & p = j, q = i \\ -\sin\theta, & p = i, q = j \\ 0, & \text{otherwise.} \end{cases} \tag{17}$$

The $[\cdot]_{pq}$ denotes the pq-th element in a matrix. $R(i, j, \theta)$ is an orthogonal rotation in the plane of the two coordinates i and j with rotation angle θ. Moreover, $R(i, j, \theta)$ is a very sparse matrix with only the diagonal, ij-th and ji-th elements being non-zero.

The basic Jacobi plane rotation $R(i, j, \theta)$ is used sequentially to transform the sum of squares of the off-diagonal elements of all the covariance matrices to be as low as possible. By accumulating the Jacobi plane rotations, we

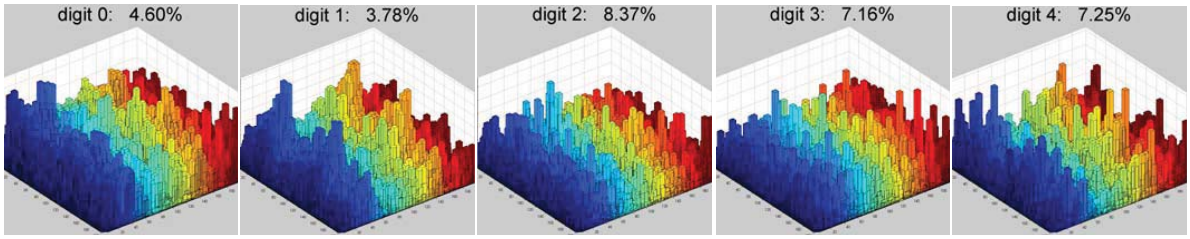


Figure 1. The covariance matrices of 5 digits. The images of digit 5 to digit 9 are omitted due to the high similarity.

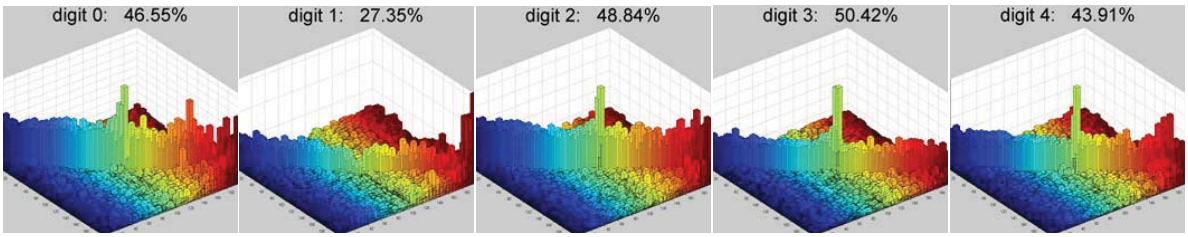


Figure 2. The covariance matrices after whitening.

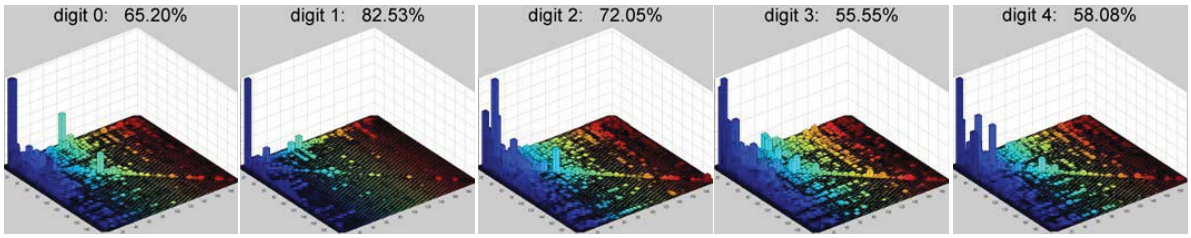


Figure 3. The covariance matrices after class conditional decorrelation (CCD).

can get the final orthogonal transformation matrix of CCD. The basic steps of the modified Jacobi method [28] involve:

- choose an index pair (i, j) that satisfies $1 \leq i < j \leq d$.
- calculate the rotation angle θ such that $\sum_{k=1}^{K} \left\| R(i,j,\theta)^\top \Sigma_k R(i,j,\theta) \right\|_{2,\text{off}}$ is minimized.
- overwrite Σ_k with $\Sigma_k^{\text{new}} = R(i,j,\theta)^\top \Sigma_k^{\text{old}} R(i,j,\theta)$.

With the index pair (i, j) traveling through $\{1, 2, \ldots, d-1\} \times \{i+1, i+2, \ldots, d\}$, the updating is implemented repeatedly until convergence. In each iteration, the objective function of (13) is decreased. At last, we get the CCD matrix as the accumulated Jacobi rotations $W = R(i_1, j_1, \theta_1) R(i_2, j_2, \theta_2) \ldots R(i_n, j_n, \theta_n)$. Therefore, the key problem is to learn the rotation angle when fixing the index pair (i, j).

2) Rotation Angle θ for Fixed i and j: When fixing i and j, the learning of the rotation angle θ can be formulated as:

$$\min_{\theta} \quad \sum_{k=1}^{K} \left\| R(i,j,\theta)^\top \Sigma_k R(i,j,\theta) \right\|_{2,\text{off}} . \tag{18}$$

Lemma 2. *$\left\| R(i,j,\theta)^\top A R(i,j,\theta) \right\|_{2,\text{off}}$ will only be changed on the ij-th elements w.r.t different θ.*

Hence, the problem in (18) is equivalent to

$$\min_{\theta} \quad \sum_{k=1}^{K} \left[R(i,j,\theta)^\top \Sigma_k R(i,j,\theta) \right]_{ij}^2 . \tag{19}$$

Moreover

$$\begin{aligned} &\left[R(i,j,\theta)^\top A R(i,j,\theta) \right]_{ij} \\ &= (\cos^2\theta - \sin^2\theta) A_{ij} + \cos\theta \sin\theta (A_{jj} - A_{ii}) . \end{aligned} \tag{20}$$

Therefore the rotation angle problem in (19) becomes:

$$\begin{aligned} &\min_{\theta} \ \cos^2 2\theta D + (1/4) \sin^2 2\theta E + \sin 2\theta \cos 2\theta F \\ &= \frac{D - E/4}{2} \cos 4\theta + \frac{F}{2} \sin 4\theta + \frac{D + E/4}{2} . \end{aligned} \tag{21}$$

Here

$$D = \sum_{k=1}^{K} (\Sigma_{ij}^k)^2 , \quad E = \sum_{k=1}^{K} \left(\Sigma_{jj}^k - \Sigma_{ii}^k \right)^2 , \tag{22}$$

$$F = \sum_{k=1}^{K} \Sigma_{ij}^k \left(\Sigma_{jj}^k - \Sigma_{ii}^k \right) . \tag{23}$$

We use Σ_{ij}^k to denote the ij-th elements in Σ_k. By some trigonometric analysis [28], the optimal θ can be computed

as:

$$\theta = \begin{cases} \pi/4, & D - E/4 > 0, F = 0 \\ 0, & D - E/4 \leq 0, F = 0 \\ (3\pi/2 - \phi)/4, & D - E/4 \geq 0, F > 0 \\ (\pi/2 - \phi)/4, & D - E/4 \geq 0, F < 0 \\ (\pi/2 - \phi)/4, & D - E/4 < 0, F < 0 \\ (-\pi/2 - \phi)/4, & D - E/4 < 0, F > 0 \end{cases} \tag{24}$$

where

$$\phi = \arctan\left(\frac{D - E/4}{F}\right) \in (-\frac{\pi}{2}, \frac{\pi}{2}) \,. \tag{25}$$

3) The CCD Algorithm: The Jacobi plane rotation (17) is used to find the optimal rotation angle θ when fixing i and j (18). To solve the CCD model in (13), the basic Jacobi plane rotations are used sequentially to minimize the sum of squares of the off-diagonal elements. The complete procedures are shown in Algorithm 1. After we get the optimal Jacobi plane rotation, all the covariance matrices are rotated in Eq. (26). The transformations are accumulated in Eq. (27) to get the final transformation of CCD.

Input: the covariance matrices $\{\Sigma_1, \ldots, \Sigma_K\} \in \mathbb{R}^{d \times d}$
Initial: $W = I$
Do the following steps repeatedly until convergence:
for $i = 1 : d-1$, for $j = i+1 : d$,

get $R(i, j, \theta)$ by Eq. (24) and Eq. (17)

$$\Sigma_k^{\text{new}} = R(i, j, \theta)^\top \Sigma_k^{\text{old}} R(i, j, \theta), \forall k \tag{26}$$

$$W^{\text{new}} = W^{\text{old}} R(i, j, \theta) \tag{27}$$

Return: $W \in \mathbb{R}^{d \times d}$

Algorithm 1: Class Conditional Decorrelation

Theorem 2. *The returned transformation matrix of Algorithm 1 satisfies the orthogonal constraint in (13).*

Proof: The output of Algorithm 1 is

$$W = R(i_1, j_1, \theta_1) R(i_2, j_2, \theta_2) \ldots R(i_n, j_n, \theta_n) \,. \tag{28}$$

It is easy to check that each Jacobi plane rotation is orthogonal:

$$R(i, j, \theta)^\top R(i, j, \theta) = I \,. \tag{29}$$

This leads to $W^\top W = I$. ■

Theorem 3. *The objective function in (13) is non-increasing under the updating rules in (26) and (27).*

Proof: When fixing i and j, the Jacobi plane rotation angle is selected to minimize (18). Therefore the objective function of (13) at the optimal solution of current Jacobi plane rotation will not be larger than the former step, since the searching space of (18) covers $\theta = 0$ which reduces the Jacobi plane rotation to the identity matrix. Therefore, with accumulated Jacobi plane rotations in (26) and (27), the objective function in (13) will be non-increasing. ■

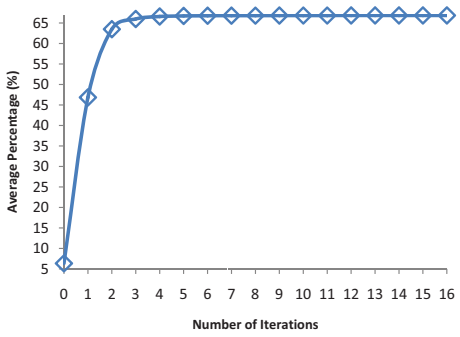


Figure 4. Convergence analysis.

4) Computational Complexity: In each rotation, the sum of squares of off-diagonal elements is decreased. When the calculated rotation angle θ goes to zero, there will be no more transformations (the Jacobi rotation with $\theta = 0$ is an identity matrix). This means that the objective function of CCD becomes invariant to any transformations. We stop the iterations in Algorithm 1 when the change of the objective function in (13) is lower than a pre-defined threshold.

The main computational complexity of CCD (as shown in Algorithm 1) is the outside iteration of the traveling of index pair. For each fixed index pair i and j, the main computations are: (i) calculate the Jacobi rotation (17); (ii) rotate the covariance matrices (26); and (iii) accumulate the transformation (27). These steps are linear dependent with the number of classes K. By considering the outside index traveling steps of $i = 1 : d-1,\ j = i+1 : d$ in Algorithm 1 as one iteration, we show the average percentage $P(\cdot)$ of the 10 covariance matrices (see Section IV-B) with respect to the number of iterations in Figure 4. We can find that: (i) with one iteration, the percentage is dramatically increased; and (ii) after three iterations, the algorithm is nearly converged. This indicates the effectiveness of the modified Jacobi method in solving the CCD problem.

V. Nearest Class Mean Classification with CCD

In this section, we integrate CCD into the nearest class mean (NCM) classification models [37], [29]. NCM represents each classes by their mean feature vector of its samples, and assigns a new pattern to the class $k \in \{1, \ldots, K\}$ with the closest mean:

$$x \in \text{class } \arg\min_{k=1}^{K} d(x, \mu_k) \,, \tag{30}$$

$$\mu_k = \frac{1}{N_k} \sum_{i:y_i=k} x_i \,, \tag{31}$$

where μ_k is the mean vector for class k, and $d(x, \mu_k)$ is a distance metric between a pattern x and class mean μ_k, and y_i is the ground-truth label of pattern x_i, and N_k is the number of training samples in class k. Contrary to the

k-NN classifier, NCM is much more efficient, because only the class-wise mean vectors are needed to be estimated and saved for future prediction. Furthermore, the NCM classifier is much more efficient and effective in generalizing to new classes [29] by adding or adjusting the new class mean.

The success of the NCM classifier critically depends on the used distance metric $d(x, \mu_k)$. The simplest metric is the Euclidean distance (ED):

$$\text{NCMED:} \quad d(x, \mu_k) = \|x - \mu_k\|^2 \ . \tag{32}$$

However, in many situations, the Euclidean distance is not the optimal measurement.

Given the mean vector μ_k and covariance matrix Σ_k for each class, the optimal Bayes classifier is a quadratic discriminant function:

$$d(x, \mu_k) = \log|\Sigma_k| + (x - \mu_k)^\top \Sigma_k^{-1}(x - \mu_k) \ . \tag{33}$$

For many applications (e.g. large category K or high dimensionality d), the computation and storage for the inverse covariance matrix Σ_k^{-1} are both very expensive due to the singular problem and $Kd(d+1)/2$ free parameters. Therefore, derived from the diagonal assumption of covariance matrices, the weighted distance (WD) is widely used as an approximation of the original distance:

$$\begin{aligned} \text{NCMWD:} \quad & d(x, \mu_k) = \log|\text{diag}(\Sigma_k)| + \\ & (x - \mu_k)^\top \text{diag}(\Sigma_k)^{-1}(x - \mu_k) \ , \end{aligned} \tag{34}$$

where $\text{diag}(A)$ is a $d \times d$ matrix with i, i-th elements equal to $A_{i,i}$ and i, j-th ($\forall j \neq i$) elements equal to zero. In this way, the inverse matrix can be efficiently calculated as $[\text{diag}(A)^{-1}]_{i,i} = \frac{1}{A_{i,i}}, \forall i$ and $[\text{diag}(A)^{-1}]_{i,j} = 0, \forall j \neq i$. In the singular situation, the zero diagonal elements $A_{i,i}$ are replaced with a positive constant (selected via cross-validation).

A. Using CCD to Improve NCMED and NCMWD

NCMED and NCMWD are only suboptimal models, however, we can use CCD to improve their performance by learning a more suitable distance metric $d(x, \mu_k)$. Suppose the CCD transformation matrix $W_{\text{CCD}} \in \mathbb{R}^{d \times d}$ has already been learned from the training data, we can define the multiple dimensional scaling as

$$\delta_i = \frac{1}{K} \sum_{k=1}^{K} \left[W_{\text{CCD}}^\top \Sigma_k W_{\text{CCD}} \right]_{ii}, i = 1, \ldots, d \ . \tag{35}$$

We use $\Lambda \in \mathbb{R}^{d \times d}$ to represent the matrix with $\Lambda_{i,i} = \sqrt{\delta_i}, \forall i$ and $\Lambda_{i,j} = 0, \forall j \neq i$. For NCMED, the improved metric is defined as:

$$\begin{aligned} & \text{CCD-NCMED:} \\ & d(x, \mu_k) = \left\| \Lambda^{-1} W_{\text{CCD}}^\top (x - \mu_k) \right\|^2 \ , \end{aligned} \tag{36}$$

For NCMWD, the improved metric is defined as

$$\begin{aligned} & \text{CCD-NCMWD:} \\ & d(x, \mu_k) = \log \left| \text{diag}(\widehat{\Sigma_k}) \right| + \\ & (x - \mu_k)^\top W_{\text{CCD}} \text{diag}(\widehat{\Sigma_k})^{-1} W_{\text{CCD}}^\top (x - \mu_k) \ , \end{aligned} \tag{37}$$

where $\widehat{\Sigma_k} = W_{\text{CCD}}^\top \Sigma_k W_{\text{CCD}}$. After CCD transformation, the variables become class conditionally uncorrelated, therefore, the classification performance will be improved for both NCMED and NCMWD. In other words, CCD is used to learn a much better distance metric for NCM classification by considering the class conditional correlation information of different classes simultaneously.

B. Comparison with Other Models

There are also other models attempting to learn a distance metric for NCM classification.

1) NCMML: Nearest Class Mean Metric Learning: The recently proposed NCMML [29] model learns a low-rank Mahalanobis distance metric for NCM:

$$\begin{aligned} \text{NCMML:} \quad d(x, \mu_k) &= \|W(x - \mu_k)\|^2 \\ &= (x - \mu_k)^\top W^\top W (x - \mu_k) \ , \end{aligned} \tag{38}$$

where $W \in \mathbb{R}^{d' \times d}$ ($d' < d$) is a dimensionality reduction matrix and $W^\top W$ is a low-rank Mahalanobis distance metric. Using a multi-class logistic regression formulation:

$$p(k|x) = \frac{\exp\left(-\frac{1}{2} d(x, \mu_k)\right)}{\sum_{k'=1}^{K} \exp\left(-\frac{1}{2} d(x, \mu_{k'})\right)} \ , \tag{39}$$

the projection matrix W can be learned via maximizing the log-likelihood of the correct predictions of the training samples: $\max_W \ \sum_{i=1}^{N} \log p(y_i|x_i)$.

Compared with the original NCMED and NCMWD, NCMML is more flexible and accurate by learning a Mahalanobis distance from the data. However, the covariance information of different classes are not taken into consideration. Furthermore, the learned Mahalanobis distance matrix is shared for all classes, which cannot reflect the difference between the covariance matrices of different classes. By taking the second-order covariance information into the learning process, CCD-NCMED and CCD-NCMWD can learn better distance metrics for NCM classification.

2) LDF: Linear Discriminant Function: The classical LDF model assumes all the classes sharing the same covariance matrix [15]: $\Sigma_0 = \frac{1}{K} \sum_{k=1}^{K} \Sigma_k$, and defines the distance metric as:

$$\text{LDF:} \quad d(x, \mu_k) = (x - \mu_k)^\top \Sigma_0^{-1} (x - \mu_k) \ . \tag{40}$$

LDF is equivalent to whitening (Section III) followed by NCMED. For heteroscedastic problems (different classes have different covariance matrices), LDF is only a suboptimal model and can not achieve good performance.

Table I
INFORMATION OF 15 DATASETS.

	#class	#feature	#sample	testing
german.numer	2	24	1000	cv-10
mushrooms	2	112	8124	cv-10
australian	2	14	690	cv-10
breast-cancer	2	10	683	cv-10
heart	2	13	270	cv-10
ionosphere	2	34	351	cv-10
liver-disorders	2	6	345	cv-10
iris	3	4	150	cv-10
svmguide2	3	20	391	cv-10
wine	3	13	178	cv-10
vehicle	4	18	846	cv-10
svmguide4	6	10	612	cv-10
glass	6	9	214	cv-10
segment	7	19	2310	cv-10
vowel	11	10	990	cv-10

3) SVM: Support Vector Machine: SVM is a state-of-the-art classifier in many domains. Equipped with the large margin training (hinge loss and regularization) and kernel tricks, SVM is effective to find the optimal classification boundaries due to the convex formulation. The performance of NCMED and NCMWD should be inferior to SVM. However, with the help of CCD, we will show in the following sections that the performance of CCD-NCMED and CCD-NCMWD will become competitive with SVM.

VI. EXPERIMENTS

In this section, we first compare different classification models on 15 small-scale databases. We also evaluate the possibility of CCD on simultaneously diagonalizing thousands of covariance matrices.

A. Classification on 15 Databases

We use the datasets collected on the LIBSVM website[2] to evaluate different models. The complete information of different databases are shown in Table I. For the preprocessing of data, each attribute is linearly scaled to $[-1, 1]$ or $[0, 1]$. In the SVM training process [8], the cost parameter C was set as 1 and the γ in RBF kernel was set as $\frac{1}{\text{num\_features}}$. For multi-class problems, the one-versus-one strategy is adopted for SVM, while the NCM-based classifiers are intrinsically multi-class models. For each database, we randomly partition the data into two subsets: using 90% of them for training and the remaining 10% for testing. This "partition-evaluation" process is repeated 10 times, and we report the average accuracy and standard deviation for each model.

The comparisons of NCMED, NCMWD, LDF, NCMML, SVM-Linear, SVM-RBF, CCD-NCMED, and CCD-NCMWD are shown in Table II. We can find that: the performance of NCMED and NCMWD are very poor when compared with the SVM classifiers. This is because NCMED and NCMWD are only generative models which are not optimized to minimize the empirical loss on training data (such as the discriminative model of SVM). Furthermore, only the Euclidean distance and weighted distance are equipped with NCMED and NCMWD, therefore, the rich discriminative information embedded in the training data are not fully explored by NCMED and NCMWD. However, with the help of class conditional decorrelation (CCD), the improved distance metrics of CCD-NCMED and CCD-NCMWD can significantly boost the accuracies as shown in Table II. The performance of CCD-NCMED and CCD-NCMWD are comparable with SVM classifiers. This is because CCD can decorrelate the class conditional variables, and the covariance information of different classes are taken into consideration simultaneously, which can learn a much better distance metric for NCM classification.

Compared with other metric learning methods for NCM classification such as LDF and NCMML, CCD achieves superior performance. This is because LDF is based on the homoscedastic assumption which is equivalent to whitening followed by Euclidean distances, and CCD is more flexible than the whitening model (see Section IV-A). NCMML directly learns a low-rank Mahalanobis distance metric for NCM classification by maximizing the likelihood of data under a multi-class logistic regression formulation, therefore, its performance is comparable with SVM classifiers as shown in [29]. However, the second-order covariance information is not taken into consideration for the learning process of NCMML.

Taking all the comparisons together, we can conclude: with class conditional decorrelation (CCD), the performance of NCMED and NCMWD can be significantly improved, which become very competitive to other classifiers.

B. CCD for Large Scale Application

Previous evaluations are conducted on some small-scale databases. In this section, we use a large-scale database to evaluate the possibility of simultaneously diagonalizing thousands of covariance matrices, and also evaluate the performance of CCD on improving the classification accuracies for problems with thousands of classes..

The used database is the $3,755$-class handwritten online Chinese character database CASIA-OLHWDB1.1 [24]. Handwritten Chinese character recognition is a challenging problem due to the large number of classes and handwriting style variation across individuals [41]. For representing a character sample, we use a benchmark feature extraction method [25]: 8-direction histogram feature extraction combined with pseudo 2D bi-moment normalization. The feature dimensionality is 512. The number of training and testing samples are $898,573$ and $224,559$ respectively, for which the statistical significance of evaluations should be sufficient. The extracted feature data can be downloaded from website[3].

[2]http://www.csie.ntu.edu.tw/%7Ecjlin/libsvmtools/datasets

[3]http://www.nlpr.ia.ac.cn/databases/handwriting/Download.html

Table II
CLASSIFICATION ACCURACIES AND STANDARD DEVIATIONS (%) OF DIFFERENT MODELS ON 15 DATASETS.

	NCMED	NCMWD	LDF	NCMML	SVM-Linear	SVM-RBF	CCD-NCMED	CCD-NCMWD
german.numer	68.70(3.62)	70.50(4.14)	73.36(2.16)	73.60(3.86)	**77.90**(2.64)	75.10(3.00)	73.10(3.21)	71.80(3.43)
mushrooms	89.47(0.87)	99.05(0.32)	99.66(0.12)	99.96(0.06)	**100.00**(0.00)	99.85(0.10)	99.77(0.24)	99.98(0.05)
australian	87.00(2.97)	87.43(3.35)	86.00(4.03)	87.93(2.66)	86.71(2.43)	84.29(3.81)	88.14(3.23)	**88.43**(5.93)
breast-cancer	96.38(1.96)	95.36(1.91)	96.50(1.56)	96.52(1.83)	95.36(1.50)	96.67(2.37)	**97.39**(2.72)	96.67(2.27)
heart	80.37(5.53)	80.00(9.49)	84.12(3.55)	84.44(4.88)	82.22(6.72)	**85.19**(7.20)	82.96(6.81)	81.48(6.05)
ionosphere	69.17(7.23)	85.83(4.62)	89.33(3.71)	92.06(3.94)	86.39(6.61)	**95.00**(2.55)	86.67(4.68)	93.89(3.15)
liver-disorders	54.86(5.18)	53.43(9.34)	63.10(5.31)	63.14(7.91)	63.71(5.56)	57.71(4.43)	**64.57**(7.76)	63.43(5.99)
iris	89.33(6.44)	92.00(8.20)	98.82(1.16)	**99.33**(2.11)	96.00(4.66)	96.67(3.51)	98.00(4.50)	96.00(4.66)
svmguide2	78.78(8.22)	75.61(6.30)	80.30(3.02)	**81.95**(6.82)	56.10(0.00)	56.10(0.00)	80.00(5.24)	80.49(5.01)
wine	96.84(3.68)	97.37(4.47)	98.56(1.16)	**100.00**(0.00)	97.37(3.72)	98.95(2.22)	98.42(2.54)	**100.00**(0.00)
vehicle	43.26(5.50)	46.63(6.31)	79.20(2.15)	79.30(3.37)	78.84(3.79)	71.51(4.43)	76.98(3.98)	**80.23**(4.78)
svmguide4	20.95(6.34)	44.76(5.39)	48.61(5.21)	49.37(5.37)	29.37(3.76)	21.11(3.18)	50.95(4.39)	**51.59**(5.46)
glass	48.26(6.94)	41.74(7.16)	61.32(5.32)	60.00(8.15)	63.91(6.50)	60.87(7.67)	**65.22**(8.45)	56.52(6.80)
segment	84.46(2.93)	80.95(1.90)	92.11(1.23)	92.21(1.44)	**94.33**(1.25)	91.52(1.75)	90.30(2.16)	91.30(1.70)
vowel	48.28(3.77)	68.69(4.31)	65.41(5.66)	71.81(5.08)	71.72(5.39)	**77.68**(3.79)	53.54(5.57)	72.32(3.93)

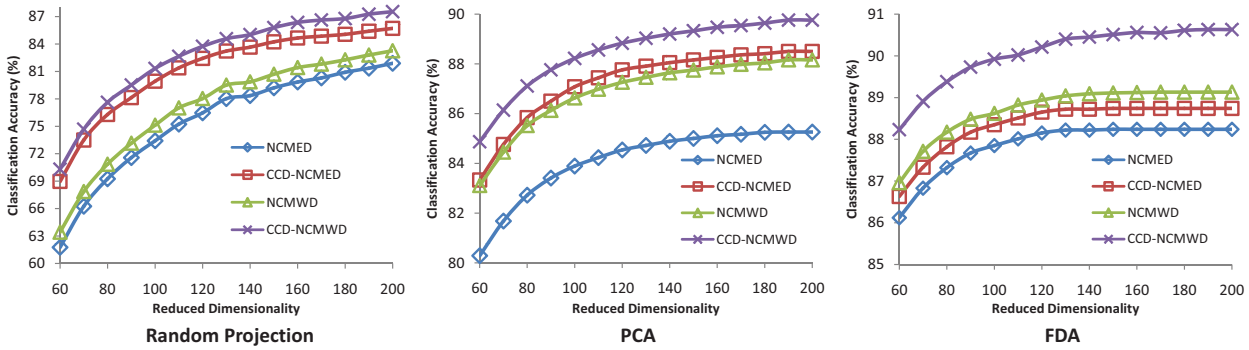


Figure 5. Effectiveness of CCD on the $3,755$-class problem.

We use different dimensionality reduction methods (random projection, PCA, and FDA) to reduce the high-dimensional data into different subspaces. In each subspace, the classification accuracies of NCMED and NCMWD are compared with CCD-NCMED and CCD-NCMWD respectively. Because our goal is to evaluate the effectiveness of CCD for thousands of classes problem, we did not compare other methods (NCMML and SVM). The comparison results are shown in Figure 5. We can find that: for both NCMED and NCMWD, the classification accuracies can be significantly improved when equipped with CCD transformation.

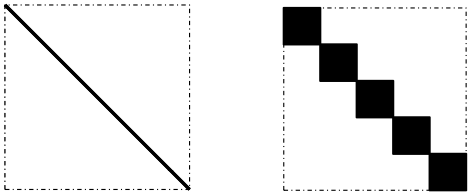
Random projection (RP) [1] is a data-independent model, while PCA [20] is an unsupervised model. The supervised class information is ignored by RP and PCA. Contrarily, the objective of CCD is to decorrelate the covariance matrices of each class simultaneously, which can learn a more suitable distance metric. Therefore CCD can improve the accuracies significantly for RP and PCA. FDA [15] is the most well-known supervised dimensionality reduction model, which is equivalent to a two-step approach: whitening followed by PCA in the whitened space (Section II). However, CCD can still significantly improve the classification accuracies in the FDA transformed subspaces as shown in Figure 5. This is because CCD can decorrelate the variables in each class, while after whitening the non-diagonal elements of the covariance matrices are still dense (see Section IV-B). This indicates the advantages of CCD against whitening.

Altogether, from the analyses above, we can conclude that: even when the number of classes is as large as $3,755$, CCD is still effective in improving the classification accuracies. This indicates that: (i) the structure of latent common principal components for multiple classes exists not only in small category problems but also in large category problems; and (ii) the modified Jacobi algorithm used in CCD is efficient and effective to find such structure via simultaneous diagonalization of even thousands of covariance matrices. This makes CCD scalable for both small-scale and large-scale applications.

VII. FURTHER EXTENSIONS

Besides the previous applications of CCD in nearest class mean (NCM) classification, CCD can be also used for many other problems in pattern recognition and machine learning.

The Gaussian mixture model (GMM) widely used to approximate arbitrarily complicated distributions, has been successfully applied in many applications such as speak-



(a) (b)

Figure 6. Two structure templates. $S_{ij} = 0$ for the black area and $S_{ij} = 1$ for the white area.

er verification [31], object representation [30] and image classification [9]. However, GMM has many parameters to estimate which is prone to overfit the training data. A widely used method to alleviate the overfitting and reduce the computational complexity is to use diagonal covariance matrices for each mixture component. Therefore the CCD algorithm can be incorporated into the expectation-maximization (EM) learning process of GMM to make the diagonal assumption more reasonable.

We can also extend the CCD (13) into a more generalized formulation (GCCD) by integrating some structure information into the learning process

$$\min_{W\in\mathbb{R}^{d\times d}} \quad \sum_{k=1}^{K} \left\|W^\top \Sigma_k W\right\|_{p,S} , \quad \text{s.t.} \quad W^\top W = I , \tag{41}$$

where

$$\|A\|_{p,S} = \sum_{i,j=1}^{d} S_{ij}|A_{ij}|^p . \tag{42}$$

The $S \in \{0,1\}^{d\times d}$ is the pre-defined structure template. For example, in this paper, we use a structure template as Figure 6(a) to get *uncorrelated dimensionalities* (see the definition of $\|A\|_{2,\text{off}}$ in Eq. (14)). We can also use other templates such as Figure 6(b) to transform the covariance matrices into block-like forms to get *uncorrelated subspaces*. In this way, different classifiers trained in different subspaces will contain complementary information, and the combination of them can be used to further boost the classification accuracy. This is known as classifier ensemble learning [12]. Another benefit of this ensemble is that the number of parameters can significantly be reduced for the classifiers which have the number of parameters superlinear dependent on the dimensionality.

The modified Jacobi method (Algorithm 1) can be used directly to solve the GCCD problem with $p = 2$. Because the Jacobi rotation (17) is an elementary operator which is easy for incorporating structure information, we only need to change the traveling of the index in Algorithm 1 from $i = 1 : d-1,\ j = i+1 : d$ to $S_{ij} = 1, \forall i, j$.

We can also consider other values of p, e.g., the L1-norm of $p = 1$ which is proven to be less sensitive and more robust to outliers [21]. However, for $p \neq 2$ the Lemma 2 no longer holds, hence the modified Jacobi method cannot be used to solve the $p \neq 2$ problems. Finding efficient algorithms to solve GCCD with arbitrary p is an interesting topic.

VIII. Conclusion

In this paper, motivated from the whitening (Theorem 1) used in the classical Fisher linear discriminant analysis (two-step decomposition), we proposed the class conditional decorrelation (CCD) model for simultaneous diagonalization of covariance matrices for all classes. The modified Jacobi method is adopted to solve the optimization problem efficiently. After CCD transformation, the variables become class conditionally uncorrelated which can benefit the following classification tasks. Combining CCD with the nearest class mean (NCM) classification model is show to be competitive with other classifiers. CCD is also shown to be effective for large scale problems which have thousands of classes. Besides the presented applications in this paper, CCD can also be hopefully extended to other applications such as Gaussian mixture models and classifier ensemble learning. The nonlinear extension of CCD such as kernelization is also an interesting direction.

Acknowledgements

This work was supported by National Basic Research Program of China (973 Program) Grant 2012CB316300 and National Natural Science Foundation of China (NSFC) Grant 61075052.

References

[1] N. Ailon and B. Chazelle. Faster dimension reduction. *Communications of the ACM*, 53(2):97–104, 2010.

[2] M. Belkin and P. Niyogi. Laplacian eigenmaps and spectral techniques for embedding and clustering. *Proc. Advances in Neural Information Processing Systems*, 14:585–591, 2001.

[3] W. Bian and D. Tao. Max-Min distance analysis by using sequential SDP relaxation for dimension reduction. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 33(5):1037–1050, 2011.

[4] G. Cao, L.R. Bachega, and C.A. Bouman. The sparse matrix transform for covariance estimation and analysis of high dimensional signals. *IEEE Trans. Image Processing*, 20(3):625–640, 2011.

[5] G. Cao and C.A. Bouman. Covariance estimation for high dimensional data vectors using the sparse matrix transform. *Advances in Neural Information Processing Systems*, 2008.

[6] J.F. Cardoso and A. Souloumiac. Jacobi angles for simultaneous diagonalization. *SIAM J. Matrix Analysis and Applications*, 17(1):161–164, 1996.

[7] H. Cevikalp, M. Neamtu, M. Wilkes, and A. Barkana. Discriminative common vectors for face recognition. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 27(1):4–13, 2005.

[8] C.-C. Chang and C.-J. Lin. LIBSVM: A library for support vector machines. *ACM Trans. Intelligent Systems and Technology*, 2:27:1–27:27, 2011.

[9] K. Chatfield, V. Lempitsky, A. Vedaldi, and A. Zisserman. The devil is in the details: an evaluation of recent feature encoding methods. *Proc. British Machine Vision Conference*, 2011.

[10] J.V. Davis, B. Kulis, P. Jain, S. Sra, and I.S. Dhillon. Information-theoretic metric learning. In *Proc. Int'l Conf. Machine Learning*, pages 209–216, 2007.

[11] I.S. Dhillon and J.A. Tropp. Matrix nearness problems with Bregman divergences. *SIAM J. Matrix Analysis and Applications*, 29(4):1120–1146, 2007.

[12] T. Dietterich. Ensemble methods in machine learning. *Multiple Classifier Systems*, pages 1–15, 2000.

[13] B.K. Flury. Two generalizations of the common principal component model. *Biometrika*, 74(1):59–69, 1987.

[14] B.N. Flury. Common principal components in k groups. *J. American Statistical Association*, 79(388):892–898, 1984.

[15] K. Fukunaga. *Introduction to Statistical Pattern Recognition*. Academic Press, 1990.

[16] G.H. Golub and C.F. Van Loan. *Matrix Computations*, volume 3. Johns Hopkins University Press, 1996.

[17] X. He and P. Niyogi. Locality preserving projections. *Proc. Advances in Neural Information Processing Systems*, 2004.

[18] G.E. Hinton and R.R. Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 313(5786):504–507, 2006.

[19] A. Hyvärinen and E. Oja. Independent component analysis: algorithms and applications. *Neural Networks*, 13(4-5):411–430, 2000.

[20] I.T. Jolliffe. *Principal Component Analysis*. Springer-Verlag, 1986.

[21] N. Kwak. Principal component analysis based on L1-norm maximization. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 30(9):1672–1680, 2008.

[22] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proc. IEEE*, 86(11):2278–2324, 1998.

[23] D.D. Lee and H.S. Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791, 1999.

[24] C.-L. Liu, F. Yin, D.-H. Wang, and Q.-F. Wang. CASIA online and offline Chinese handwriting databases. In *Proc. Int'l Conf. Document Analysis and Recognition*, pages 37–41, 2011.

[25] C.-L. Liu and X.-D. Zhou. Online Japanese character recognition using trajectory-based normalization and direction feature extraction. *Proc. Int'l Workshop Frontiers in Handwriting Recognition*, 2006.

[26] M. Loog and R.P.W. Duin. Linear dimensionality reduction via a heteroscedastic extension of LDA: the Chernoff criterion. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 26(6):732–739, 2004.

[27] M. Loog, R.P.W. Duin, and R. Haeb-Umbach. Multiclass linear dimension reduction by weighted pairwise Fisher criteria. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 23(7):762–766, 2001.

[28] M. Maleko. A Jacobi algorithm for simultaneous diagonalization of several symmetric matrices. *Department of Numerical Analysis and Computer Science, Royal Institute of Technology, Stockholm*, 2003.

[29] T. Mensink, J. Verbeek, F. Perronnin, and G. Csurka. Distance-based image classification: Generalizing to new classes at near-zero cost. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 35(11):2624–2637, 2013.

[30] B. Moghaddam and A. Pentland. Probabilistic visual learning for object representation. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 19(7):696–710, 1997.

[31] D.A. Reynolds, T.F. Quatieri, and R.B. Dunn. Speaker verification using adapted gaussian mixture models. *Digital Signal Processing*, 10(1-3):19–41, 2000.

[32] S.T. Roweis and L.K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326, 2000.

[33] B. Schölkopf, A. Smola, and K.R. Müller. Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, 10(5):1299–1319, 1998.

[34] A. Stuhlsatz, J. Lippel, and T. Zielke. Feature extraction with deep neural networks by a generalized discriminant analysis. *IEEE Trans. Neural Networks and Learning Systems*, 23(4):596–608, 2012.

[35] D. Tao, X. Li, X. Wu, and S.J. Maybank. Geometric mean for subspace selection. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 31(2):260–274, 2009.

[36] J.B. Tenenbaum, V. De Silva, and J.C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000.

[37] A. R. Webb. Statistical pattern recognition. *New-York, NY, USA: Wiley*, 2002.

[38] J. Yang, A.F. Frangi, J. Yang, D. Zhang, and Z. Jin. KPCA plus LDA: a complete kernel Fisher discriminant framework for feature extraction and recognition. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 27(2):230–244, 2005.

[39] H. Yu and J. Yang. A direct LDA algorithm for high-dimensional data with application to face recognition. *Pattern Recognition*, 34(10):2067–2070, 2001.

[40] X.-Y. Zhang and C.-L. Liu. Evaluation of weighted Fisher criteria for large category dimensionality reduction in application to Chinese handwriting recognition. *Pattern Recognition*, 46:2599–2611, 2013.

[41] X.-Y. Zhang and C.-L. Liu. Writer adaptation with style transfer mapping. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 35(7):1773–1787, 2013.

[42] M. Zhu and A.M. Martinez. Subclass discriminant analysis. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 28(8):1274–1286, 2006.

[43] A. Ziehe, P. Laskov, G. Nolte, and K.R. Müller. A fast algorithm for joint diagonalization with non-orthogonal transformations and its application to blind source separation. *J. Machine Learning Research*, 5:777–800, 2004.