Pattern Recognition Letters 35 (2014) 186–194

Contents lists available at SciVerse ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

Similar handwritten Chinese character recognition by kernel discriminative locality alignment

Dapeng Tao, Lingyu Liang, Lianwen Jin *, Yan Gao

School of Electronic and Information Engineering, South China University of Technology, Guangzhou, PR China

ARTICLE INFO

Article history:
Available online 29 June 2012

Keywords:
Similar handwritten Chinese character recognition
Static candidates generation
Dimension reduction
Manifold learning
Patch alignment framework
Discriminative locality alignment

ABSTRACT

It is essential to extract the discriminative information for similar handwritten Chinese character recognition (SHCCR) that plays a key role to improve the performance of handwritten Chinese character recognition. This paper first introduces a new manifold learning based subspace learning algorithm, discriminative locality alignment (DLA), to SHCCR. Afterward, we propose the kernel version of DLA, kernel discriminative locality alignment (KDLA), and carefully prove that learning KDLA is equal to conducting kernel principal component analysis (KPCA) followed by DLA. This theoretical investigation can be utilized to better understand KDLA, i.e., the subspace spanned by KDLA is essentially the subspace spanned by DLA on the principal components of KPCA. Experimental results demonstrate that DLA and KDLA are more effective than representative discriminative information extraction algorithms in terms of recognition accuracy.

1. Introduction

In recent years, handwritten Chinese character recognition (HCCR) has made great progress in both research and practical applications. Unconstrained online HCCR, however, is still an open problem remaining to be solved, because it is still challenging to reach high recognition rate considering the high diversity of handwriting styles and large category set (Gao and Liu, 2008; Leung and Leung, 2010; Liu et al., 2010; Shao et al., 2011). In constrained HCCR, recognition rate can generally reach to over 98.5%; but in unconstrained online HCCR, the recognition rate drops to 92.39% (Liu et al., 2010).

Many effective methods have been proposed to promote the recognition rate in cursive online or offline HCCR. Gao and Liu (2008) presented a linear discriminant analysis (LDA)-based compound distance method to boost the recognition rate. Leung and Leung (2010) presented critical region analysis, which can distinguish one character from another similar character by emphasizing the critical regions. All of the methods above are concerned with constructing a globally linear transformation to improve recognition accuracy.

In fact, one of the main reasons for the performance drop in unconstrained online HCCR lies in that similar Chinese character often share an analogous structure, and have only presence or absence of a stroke in a specific region. Fig. 1 shows some similar cursive samples from the CASIA-OLHWD1 database (Wang et al., 2009). There is usually only one classifier for all classes in many HCCR systems. Systems of this sort are easy to construct, but fail to distinguish very similar Chinese characters.

Therefore many Chinese character recognition engines (Leung and Leung, 2010) adopt a hierarchical classifier to overcome the shortage of a single classifier. When a newly imported character is identified by the first-level, the recognition results can be reordered by the confidence score in general. The second-level classifier aims to distinguish the top confidence score results. Many methods (Gao and Liu, 2008; Shao et al., 2011; Leung and Leung, 2010) have been presented to identity the small subsets of Chinese characters. These methods aim to effectively extract the discriminative information of the simplest circumstance, i.e., a pair of similar Chinese character classes.

Although the recognition rate can be boosted, there is still room to obtain further improvement. First, using pairwise classifiers to reorder the candidate character in second-level classification task is an expensive approach, because the number of classifiers is $C(C-1)/2$ for a C-class classification problem. The time cost and space cost of this strategy is not accepted easily in general. Second, discriminative information extraction is considerably important in similar handwritten Chinese character recognition (SHCCR). We thus apply the DLA (discriminative locality alignment) manifold learning (Shao et al., 2011) and static candidate generation technique (Liu and Jin, 2007) to address these issues. Fig. 2 shows the diagram of the proposed recognition system. At the first level classification, the similar Chinese candidate sets for each class is generated using the static candidate generation technique

* Corresponding author.
E-mail addresses: dapeng.tao@gmail.com (D. Tao), lianglysky@gmail.com (L. Liang), lianwen.jin@gmail.com (L. Jin), thegoldfishwang@163.com (Y. Gao).

0167-8655/$ - see front matter © 2012 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.patrec.2012.06.014

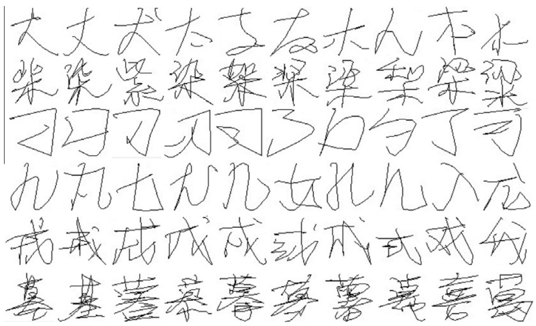

Fig. 1. Similar samples in handwritten Chinese character from CASIA-OLHWD1 database.

(Liu and Jin, 2007). Then when the first candidate is given by the first level classification, the system retrieves the corresponding similar Chinese candidate set according to the first candidate on which the second level classification is carried. DLA or kernel DLA (KDLA, which is proposed in this paper) projection matrix is applied in the selected similar character set on the consideration to improve the recognition accuracy by learning more effective discriminative features in the similar candidate set.

Linear discriminant analysis (LDA) (Fisher, 1936) is one of the widely used similar Chinese characters discriminative feature extraction methods in the literature (Gao and Liu, 2008; Jin et al., 2010; Leung and Leung, 2010; Shao et al., 2011). Traditional LDA, however, suffers from the following drawbacks. First, it ignores the local structure of samples, which makes it fail to discover the nonlinear structure hidden in the high dimensional space. Second, LDA is confronted with the small sample size (SSS) problem (Bian, 2011; Tao et al., 2009; Tao et al., 2007). We need a large number of samples for model training.

Manifold learning based dimension reduction algorithms are the powerful tools for finding the intrinsic structure of a set of samples embedded in a high dimensional ambient space and attracted intensive attention in recent years (Belkin and Niyogi, 2002; Bengio et al., 2004; Cai et al., 2005; Guan, 2012; Guan, 2012; He and Niyogi, 2004; Si, 2010; Tian, 2012; Wang, 2011; Zhou, 2011). Therefore, to overcome the above problems of LDA for performance improvement, we introduce a popular supervised manifold learning approach to SHCCR, which is called discriminative locality alignment (DLA) (Zhang et al., 2008). DLA is developed under the umbrella of patch alignment framework (PAF) (Guan, 2011; Zhang et al., 2009). PAF includes most existing manifold learning based dimension reduction algorithms as special cases and shows that these algorithms can be divided into two steps: patch optimization and global alignment. Based on PAF, we can easily understand the common points and essential differences of different algorithms, and develop new manifold learning based dimension reduction algorithms. DLA is a special implementation of PAF for classification. It contains two stages. In the first stage, DLA preserves the discriminative information in a local patch through integrating two criteria that the distances between the intra-class samples will be as small as possible and the distance between the inter-class samples will be as large as possible. In the second stage, DLA integrates all the weighted part optimizations to form a global subspace structure through an alignment operation.

As a manifold learning based method, DLA has many attractive properties for SHCCR compared to LDA. First, DLA focuses on local discriminative structure for each training sample, and it captures the discriminative information at the sample level, and thus it is more powerful than LDA. Second, DLA obtains robust classification performance under the condition of small sample size. Third, it does not need to compute the inverse of a matrix, and thus it does not face the matrix singularity problem. In addition, our experimental results show that a smaller size projection matrix obtained by DLA compared with that obtained by LDA can achieve a high recognition rate for SHCCR. That means DLA is able to keep a higher recognition rate with less computing and storage costs, which is attractive for practical applications.

However DLA is a linear algorithm, which cannot capture the nonlinearity of the samples. Motivated by kernel methods (Geng, 2011; Muller et al., 2001) successfully used for discovering the intrinsic nonlinear structure, we generalize DLA to the kernel feature space as the Kernel discriminative locality alignment (KDLA). According to the so-called "kernel trick", we map the original low-dimensional input Euclidean space to a high-dimensional Hilbert space, in which samples from different classes are almost linearly separable. KDLA obtains a set of optimal discriminative basis vectors in the high-dimensional Hilbert space. Therefore, KDLA performs much better than DLA. We prove that learning KDLA is equivalent to learning DLA in the space spanned by principal components of kernel principle component analysis (KPCA)

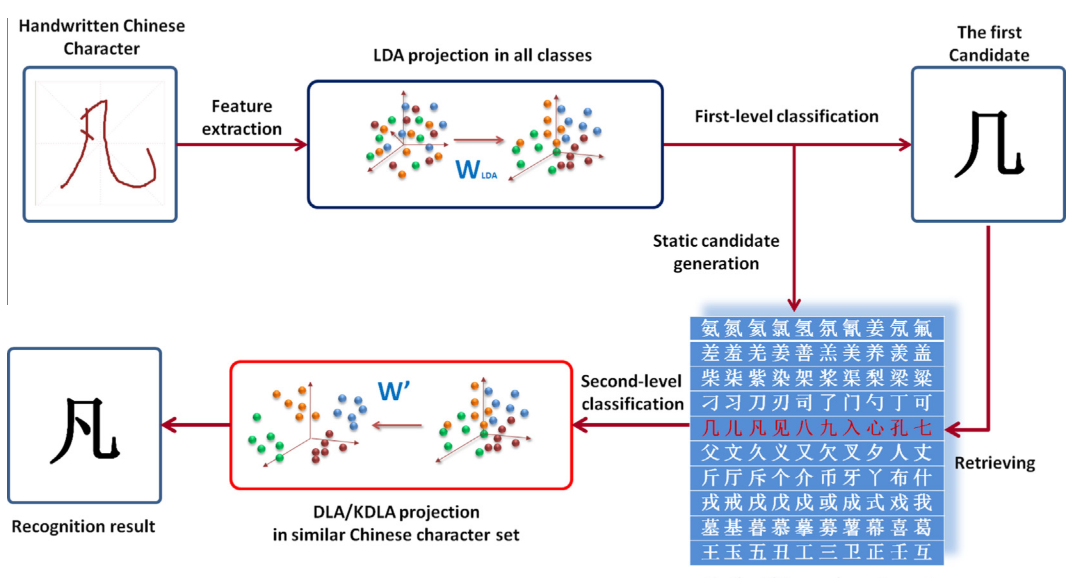

Fig. 2. The static candidate generation technique based SHCCR system.

(Schölkopf et al., 1998). This helps us to understand the difference between KDLA and DLA. In this paper, we conducted experiments on ten difficult recognition collections of similar handwritten Chinese characters to compare DLA and KDLA to popular baseline algorithms. The effectiveness of DLA and KDLA has been demonstrated by experimental results.

The rest of the paper is organized as follows: Section 2 introduces the static candidate generation technique. Section 3 introduces DLA for extracting discriminative features for SHCCR and then details the basic formulation and theoretical analysis of the proposed KDLA algorithm. Experiments and empirical analysis are presented in Section 4. Section 5 concludes this paper.

2. The static candidate character set

It is well known that the most Chinese character recognition engine which only has a single classifier to identify a newly imported character. Although this strategy is easy to implement, it fails to separate very similar Chinese characters. In this paper, we introduce static candidate generation (SCG) (Liu and Jin, 2007) to improve the performance of the similar character sets recognition.

2.1. Static candidate generation (SCG)

There are two ways to implement SCG, which are distance-based similar Chinese character sets generation and frequency-based similar Chinese character sets generation.

The distance-based SCG assumes that the distances of the similar character feature templates are close to each other in the feature space. Given a Chinese character and its feature template C_i, we generate $k-1$ candidates for C_i by selecting $k-1$ nearest feature templates of C_i, i.e., $C_{i_1}, C_{i_2}, \ldots, C_{i_{k-1}}$. Therefore, the static candidate set with respect to C_i is $[C_i, C_{i_1}, C_{i_2}, \ldots, C_{i_{k-1}}]$. Since this technique typically only utilizes the sample mean to calculate the distances, it fails to perform well.

The frequency-based SCG method (Liu and Jin, 2007) first generates the original k candidates for some samples of Chinese character C_i according to the classifier confidence score. The setting of the parameter k is according the expected hitting rate of selecting the correct SCG set corresponding to a given recognition candidate. If the expected hitting rate is high, we need to adjust the k to a large number which will result in larger storage cost. In our system, we set $k = 10$ with a hitting rate about 99%. Afterward, we use the classifier to recognize all the training samples of all the remaining classes other than C_i, and calculate the frequency of the samples that are incorrectly recognized as the character C_i. Finally, we can get the final k similar candidate set of C_i according to the error recognition frequency.

2.2. Similar character collection

The similar character collection is processed by using the frequency-based SCG method. The process is carried out as follows: (1) we generate the original static candidate class set for each Chinese character; (2) samples are collected according to the ten selected class sets which are difficult to recognize in the benchmark dataset; (3) we use the character samples corresponding to the ten generated static candidate sets as the similar character collections in our experiments.

3. Discriminative information extraction

In most of the SHCCR solutions reported in the literature (Gao and Liu, 2008; Leung and Leung, 2010), LDA is applied to extract discriminative information. However, we confront the problem that the number of the training samples is insufficient let alone that LDA ignores the local geometry of the sample distribution. Therefore, we introduce a supervised manifold learning algorithm DLA to improve the performance of discriminative information extraction in SHCCR. Afterward, we present a kernel method called kernel discriminative locality alignment (KDLA) to achieve better performance for subsequent classification. In particular, KDLA benefits from discovering the nonlinearity of the sample distribution.

We consider the general problem of discriminative information extraction. We denote a set of training samples in a high-dimensional space R^D by $X = [x_1, x_2, \ldots, x_N] \in R^{D\times N}$, each of which has a label $C_i \in Z^n$. The objective of discriminative information extraction aims to find a linear project matrix $U \in R^{D\times d}$ to project samples from the high-dimensional space R^D to the corresponding low-dimensional subspace R^d, wherein $d < D$. Therefore, the corresponding low dimensional representation is given by $Y = U^T X = [y_1, y_2, \ldots y_N] \in R^{d\times N}$.

3.1. Linear discriminant analysis

LDA is a classical algorithm for discriminative information extraction. It aims to find a subspace by maximizing the trace of the between-class scatter matrix S_b and minimizing the trace of within-class scatter matrix S_w simultaneously (Fisher, 1936). The objective function of LDA is given by:

$$\arg\max_U \frac{tr(U^T S_b U)}{tr(U^T S_w U)}, \tag{1}$$

$$S_w = \sum_{j=1}^{C}\sum_{i=1}^{N_j}(x_i^{(j)} - m_j)(x_i^{(j)} - m_j)^T, \tag{2}$$

$$S_b = \sum_{j=1}^{C} N_j (m_j - m)(m_j - m)^T, \tag{3}$$

where N_j is the training sample size of the jth class, C is the number of classes, m_j is the sample mean of the jth class, and m is the sample mean of all samples. The projection matrix U is obtained by maximizing Eq. (1). If S_w is not singular, $U$is given by the leading d eigenvectors corresponding to the d largest eigenvalues of $S_w^{-1}S_b$.

3.2. Discriminative locality alignment

Different from LDA, DLA aims to preserve the discriminative information locally. In particular, DLA conducts "part optimization" on each training sample, so that in a low dimensional subspace, the average distance between the sample and its intra-class neighbors will be as small as possible, while the average distance between the sample and its inter-class neighbors will be as large as possible. DLA then operates "whole alignment" to integrate all the weighted part optimizations to learn a global subspace structure (Zhang et al., 2009; Zhang et al., 2008). A technical review of DLA is given below for obtaining the kernel version of DLA.

3.2.1. Part optimization

The part optimization of DLA starts from each training sample and the corresponding local patch. Each patch is built by a sample and its neighbors including both intra-class and inter-class samples.

For a given sample x_i and its corresponding patch, we can find m_1 closest samples $x_{i^1}, \ldots, x_{i^{m_1}}$ that from the same class of x_i , and m_2 closest samples $x_{i_1}, \ldots, x_{i_{m_2}}$ that from different classes of x_i. Then the local patch for x_i is denoted by $X_i = [x_i, x_{i^1}, \ldots x_{i^{m_1}}, x_{i_1}, \ldots, x_{i_{m_2}}]$. The part optimization obtains a new low dimensional representation $Y_i = [y_i, y_{i^1}, \ldots y_{i^{m_1}}, y_{i_1}, \ldots, y_{i_{m_2}}]$, in which the inter-class distance is maximized and the intra-class distance is minimized.

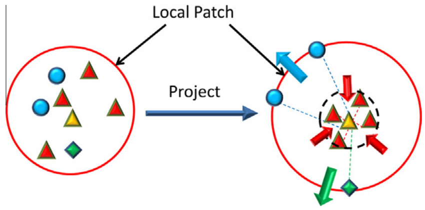

Fig. 3. The process of part optimization.

Fig. 4. Some corresponding handwritten samples of Table I.

Table I
Similar characters sets.

Set#	First Candidate	Similar character set
1	氨	氨 氮 氦 氯 氢 氛 氰 姜 氖 氟
2	差	差 羞 羌 姜 善 羔 美 养 羡 盖
3	柴	柴 柒 紫 染 架 浆 渠 梨 梁 粱
4	刁	刁 习 刀 刃 司 了 门 勺 丁 可
5	凡	凡 风 凤 几 见 叉 冗 丸 贝 八
6	父	父 文 久 义 又 欠 叉 夕 人 丈
7	斤	斤 厅 斥 个 介 币 牙 丫 布 什
8	戎	戎 戒 戌 戊 戍 或 成 式 戏 我
9	墓	墓 基 暮 慕 摹 募 薯 幕 喜 葛
10	王	王 玉 五 丑 工 三 卫 正 壬 互

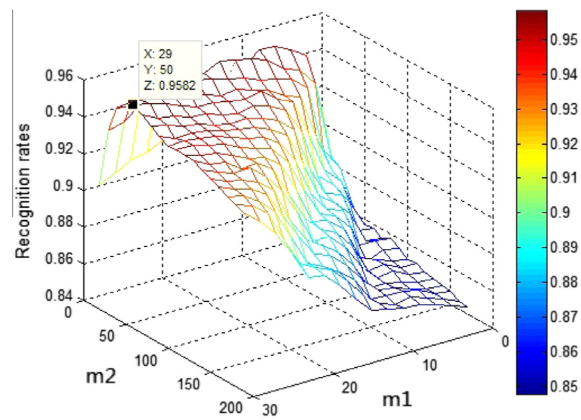

Fig. 5. Recognition rate vs. m_1 and m_2 for DLA parameters optimization.

Fig. 3 illustrates the process of part optimization in the situation when $m_1 = 4$ and $m_2 = 3$. It shows that in the projected subspace, y_i (yellow triangle) is close to the samples from intra-class (red triangle), whereas the distances between y_i and the samples from other classes (blue circle and green square) are large.

The optimization function in part optimization is given by:

$$\arg\min_{y_i}\left(\sum_{j=1}^{m_1}\|y_i - y_{i^j}\|^2 - \beta\sum_{p=1}^{m_2}\left\|y_i - y_{i_p}\right\|^2\right), \tag{4}$$

where $\sum_{j=1}^{m_1}\|y_i - y_{i^j}\|^2$ is the distances between the intra-class samples, $\sum_{p=1}^{m_2}\left\|y_i - y_{i_p}\right\|^2$ is the distance between the inter-class samples, and $\beta \in [0,1]$ is a trade-off parameter, which can balance the contributions of intra-class samples and those of the inter-class samples in part optimization.

By utilizing the coefficients vector $\omega_i = [\overbrace{1,\ldots,1}^{m_1}, \overbrace{-\beta,\ldots,-\beta}^{m_2}]^T$, we deduce (4) to

$$\arg\min_{y_i}\left(\sum_{j=1}^{m_1}\|y_i - y_{i^j}\|^2(\omega_i)_j + \sum_{p=1}^{m_2}\left\|y_i - y_{i_p}\right\|^2(\omega_i)_{p+m_1}\right)$$
$$= \arg\min_{y_i}\left(\sum_{j=1}^{m_1+m_2}\left\|y_{F_i\{1\}} - y_{F_i\{j+1\}}\right\|^2(\omega_i)_j\right), \tag{5}$$

where $F_i = \left\{i, i^1, \ldots, i^{m_1}, i_1, \ldots, i_{m_2}\right\}$.

3.2.2. Whole alignment

After part optimization, we obtain N different optimizations. In the whole alignment stage, we integrate these part optimizations as a whole

$$\arg\min_{Y}\sum_{i=1}^{N}\left(\sum_{j=1}^{m_1+m_2}\left\|y_{F_i\{1\}} - y_{F_i\{j+1\}}\right\|^2(\omega_i)_j\right). \tag{6}$$

By utilizing the selection matrix $(S_i)_{pq} = 1$, if $p = F_i\{q\}$, otherwise 0, we have $Y_i = YS_i$. Therefore, we have

$$\arg\min_{Y}\sum_{i=1}^{N}\mathrm{tr}\left(Y_i\begin{bmatrix}-e_{m_1+m_2}^T\\ I_{m_1+m_2}\end{bmatrix}\mathrm{diag}(\omega_i)_j\begin{bmatrix}-e_{m_1+m_2}^T & I_{m_1+m_2}\end{bmatrix}Y_i^T\right) =$$
$$\arg\min_{Y}\sum_{i=1}^{N}\mathrm{tr}\left(YS_iL_iS_i^TY^T\right) = \arg\min_{Y}\mathrm{tr}\left(Y\left(\sum_{i=1}^{N}S_iL_iS_i^T\right)Y^T\right) =$$
$$\arg\min_{Y}\mathrm{tr}\left(YLY^T\right), \tag{7}$$

where $e_{m_1+m_2} = [1,\ldots,1]^T \in R^{m_1+m_2}$; $I_{m_1+m_2} = \mathrm{diag}\left(\overbrace{1,\ldots,1}^{m_1+m_2}\right)$; $L_i = \begin{bmatrix}\sum_{j=1}^{m_1+m_2}(\omega_i) & -\omega_i^T\\ -\omega_i & \mathrm{diag}(\omega_i)\end{bmatrix} \in R^{(m_1+m_2+1)\times(m_1+m_2+1)}$; and $L(F_i,F_i) \leftarrow L(F_i,F_i) + L_i$ is the alignment matrix (Zhang and Zha, 2004).

We impose a constraint $U^TU = I_d$ on (7) to determine the projection matrix U according to $Y = U^TX$, wherein I_d is a $d \times d$ identity matrix. Thus (7) is transformed to

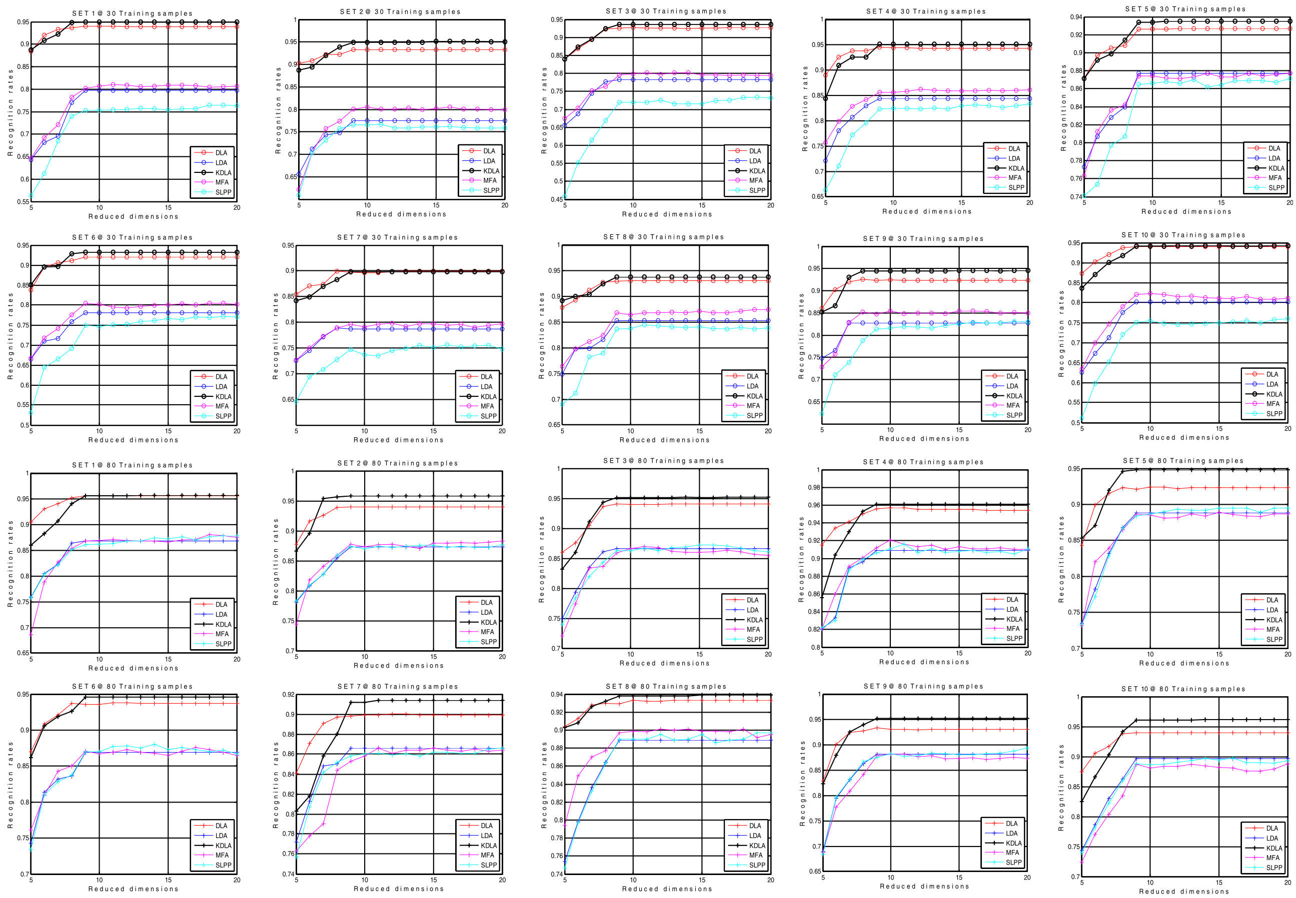

Fig. 6. Recognition rate vs. dimensionality reduction on the ten similar character collections.

Table II
Average recognition rates.N_i is the training sample size in each class. d is the reduced dimensions.

d	$N_i = 30$					$N_i = 80$				
	LDA	SLPP	MFA	DLA	KDLA	LDA	SLPP	MFA	DLA	KDLA
1	0.256	0.231	0.254	0.357	0.403	0.335	0.331	0.284	0.361	0.426
2	0.425	0.347	0.414	0.615	0.633	0.527	0.525	0.478	0.606	0.672
3	0.551	0.472	0.542	0.760	0.747	0.634	0.631	0.590	0.759	0.771
4	0.638	0.556	0.635	0.822	0.814	0.701	0.695	0.682	0.827	0.818
5	0.696	0.635	0.698	0.869	0.860	0.755	0.750	0.743	0.872	0.848
6	0.736	0.686	0.743	0.898	0.886	0.803	0.799	0.805	0.906	0.879
7	0.765	0.724	0.780	0.913	0.906	0.839	0.834	0.835	0.921	0.915
8	0.793	0.751	0.803	0.923	0.925	0.862	0.860	0.856	0.933	0.936
9	0.813	0.780	0.827	0.928	0.937	0.881	0.877	0.879	0.935	0.948
10	NA	0.781	0.828	0.928	0.937	NA	0.878	0.880	0.936	0.948
11	NA	0.782	0.827	0.928	0.938	NA	0.880	0.882	0.936	0.949
12	NA	0.780	0.826	0.928	0.938	NA	0.881	0.881	0.935	0.949
13	NA	0.781	0.827	0.928	0.938	NA	0.882	0.881	0.936	0.949
14	NA	0.781	0.827	0.928	0.938	NA	0.883	0.879	0.936	0.949
15	NA	0.783	0.827	0.928	0.938	NA	0.883	0.879	0.936	0.949
16	NA	0.787	0.827	0.928	0.938	NA	0.883	0.880	0.936	0.949
17	NA	0.787	0.827	0.928	0.938	NA	0.881	0.880	0.935	0.949
18	NA	0.786	0.826	0.928	0.938	NA	0.881	0.881	0.935	0.949
19	NA	0.788	0.827	0.928	0.938	NA	0.883	0.879	0.935	0.949
20	NA	0.788	0.828	0.928	0.938	NA	0.884	0.880	0.935	0.949

$$\arg\min_U \mathrm{tr}(U^T X L X^T U) \quad \mathrm{s.t.} U^T U = I_d. \tag{8}$$

We can transform (8) to a generalized eigenvalue problem (Jolliffe, 2002) and U is given by d eigenvectors associated with d smallest eigenvalues of XLX^T.

3.3. Kernel discriminative locality alignment

DLA is a linear algorithm, so we conduct DLA in the reproducing kernel hilbert space (RKHS), which results in kernel discriminative locality alignment (KDLA). We consider that the linear input space can be mapped to a kernel feature space by a non-linear mapping:

Set#	First Candidate	Similar character set
1	氨	氨 氮 氦 氯 氢 氛 氰 姜 氖 氟
2	差	差 羞 羌 姜 善 羔 美 养 羡 盖
3	柴	柴 柒 紫 染 架 桨 渠 梨 梁 粱
4	刁	刁 习 刀 刃 司 了 门 勺 丁 可
5	凡	凡 风 凤 几 见 叉 冗 丸 贝 八
6	父	父 文 久 义 又 欠 叉 夕 人 丈
7	斤	斤 厅 斥 个 介 币 牙 丫 布 什
8	戎	戎 戒 戍 戊 戌 或 成 式 戏 我
9	墓	墓 基 暮 慕 摹 募 薯 幕 喜 葛
10	王	王 玉 五 丑 工 三 卫 正 壬 互

Fig. 7. The recognition boxplots of different methods. There are two subfigures, each of which corresponds to the performance obtained a particular number (30, 80) of labeled training samples of each class.

$$\varphi : R^D \rightarrow F, \tag{9}$$

where F is a sufficiently high-dimensional feature space obtained by utilizing a proper nonlinear mapping function φ. In practice, it is difficult to find such a nonlinear mapping function. However, we can achieve the so-called RKHS through the kernel dot product trick, i.e., $k(x, x\prime) = \varphi(x)^T\varphi(x\prime)$. The kernel matrix formed by the given samples is positive semi-definite (Shawe-Taylor and Cristianini, 2004). The commonly used kernels include Gaussian kernel $k(x, x\prime) = \exp\left(-\|x - x\prime\|^2/2\sigma^2\right)$ and polynomial kernel $k(x, x\prime) = (1 + x^T x\prime)^d$.In KDLA, the local patch $\tilde{X}_i = [\varphi(x_i), \varphi\left(x_{i^1}\right), \ldots, \varphi(x_{i^{m_1}}), \varphi(x_i), \ldots, \varphi\left(x_{i_{m_2}}\right)]$ and $\tilde{F}_i = \left\{i, i^1, \ldots, i^{m_1}, i_1, \ldots, i_{m_2}\right\}$ is the set of indices on the patch. The alignment matrix L is obtained in an iterative procedure $\tilde{L}\left(\tilde{F}_i, \tilde{F}_i\right) \leftarrow \tilde{L}\left(\tilde{F}_i, \tilde{F}_i\right) + \tilde{L}_i$ with the initialization $\tilde{L} = 0$, where $\tilde{L}_i = \begin{bmatrix} \sum_{j=1}^{m_1+m_2}(\omega_i)_j & -\omega_i^T \\ -\omega_i & \text{diag}(\omega_i) \end{bmatrix}$. Then we have to solve the eigenvalue problem $\tilde{X}\tilde{L}\tilde{X}^T\tilde{u} = \tilde{\lambda}\tilde{u}$, where $\tilde{X} = \Phi = [\varphi(x_i), \ldots \varphi(x_N)]$. According to the representer theorem (Schölkopf et al., 2001), we have $\tilde{u} = \sum_{i=1}^N \tilde{\theta}_i\varphi(x_i)$, Then $\tilde{u} = \sum_{i=1}^N \tilde{\theta}_i\varphi(x_i) = \Phi\tilde{\theta}_i$ and the eigenvalue problem can be rewritten as $\Phi\tilde{L}\Phi^T\Phi\tilde{\theta} = \tilde{\lambda}\Phi\tilde{\theta}$, which is

$$\tilde{L}K\tilde{\theta} = \tilde{\lambda}\tilde{\theta}, \tag{10}$$

where K is the gram matrix, in which an entry is given by $K_{ij} = k(x_i, x_j)$.

The low-dimensional representation is given by

$$\tilde{y}_i = (\widetilde{U})^T\varphi(x_i) = \widetilde{\Theta}^T k_i, \tag{11}$$

where $\widetilde{U} = [\tilde{u}_1, \ldots, \tilde{u}_N]$, $\widetilde{\Theta} = \left[\tilde{\theta}_1, \ldots, \tilde{\theta}_N\right]$ and k_i is the ith column of K.

In fact, the subspace spanned by KDLA is essentially the subspace spanned by DLA on the principal components of kernel principle component analysis (KPCA) (Schölkopf et al., 1998). This fact is proved in the following theorem and is particularly important to better understand KDLA.

Theorem. *Learning a KDLA subspace is equal to conducting DLA in the space spanned by the principal components of KPCA.*

Proof. The kernel PCA is conducted by obtaining the principal components on the kernel space. Denote the covariance matrix of the training samples $X = [x_1, \ldots, x_N] \in R^{D\times N}$ in the RKHS by $C = (1/N)\sum_{i=1}^N \varphi(x_i)\varphi(x_i)^T$. For KPCA, we have find eigenvector u and eigenvalue λ satisfying $Cu = \lambda u$. This is equivalent to solving

$$\varphi(x_i)^T Cu = \lambda\varphi(x_i)^T u, \quad i = 1, \ldots, N. \tag{12}$$

We can prove that $u = \sum_{i=1}^N \beta_i\varphi(x_i)$, which is similar to the proof the representer theorem (Muller et al., 2001). Then (12) is equivalent to following optimization problem:

$$K\beta = \lambda\beta, \tag{13}$$

the solution of (13) is the eigenvector $\beta_i = [\beta_{i,1}, \ldots, \beta_{i,N}]^T$ and the corresponding eigenvalue λ_i. The projection matrix of KPCA is given by $U = [u_1, \ldots, u_N]$, each $u_i = \sum_{j=1}^N \beta_{ij}\varphi(x_j) = \Phi\beta_i$, where $\Phi = [\varphi(x_i), \ldots \varphi(x_N)]$. Normalize $\beta_i \leftarrow \beta_i/\left(\|\beta_i\|\sqrt{\lambda_i}\right)$, then we have $u_i^T u_j = \beta_i^T\Phi^T\Phi\beta_i = \beta_i^T K\beta = \lambda\beta_i^T\beta = 1$ and $u_i^T u_j = 0$. So we have $U^T U = I$. Let $B = [\beta_1, \ldots, \beta_N]$, since $U^T U = (\Phi B)^T\Phi B = B^T KB$, we obtain $B^T KB = I$. This result in $B^T B = K^{-1}$. Consequently, the feature of projected data in the KPCA is given by $\hat{y}_i = U^T\varphi(x_i) = B^T k_i$, k_i is the ith column of K. Therefore, the data processed by KPCA is $\hat{X} = B^T K$.

By first preprocessing the data with KPCA, i.e. $\hat{X} = B^T K$, the eigenvalue problem of DLA becomes $\hat{X}\hat{L}\hat{X}^T\hat{u} = \hat{\lambda}\hat{u}$. Similarly, we have $\hat{u} = \sum_{i=1}^N \hat{\theta}_i\hat{x}_i$ according to the representer theorem. Then $\hat{u}_i = \sum_{j=1}^N \hat{\theta}_{ij}\hat{x}_j = \hat{X}\hat{\theta}_i$ and the eigenvalue problem can be rewritten as $B^T K\hat{L}KBB^T K\hat{\theta} = \hat{\lambda}B^T K\hat{\theta}$. By multiplying B on the left side of the equation we have $BB^T K\hat{L}KBB^T K\hat{\theta} = \hat{\lambda}BB^T K\hat{\theta}$. That is

$$\hat{L}K\hat{\theta} = \hat{\lambda}\hat{\theta}. \tag{14}$$

The low dimensional representation is given by

$$\hat{y}_i = (\hat{U}\prime)^T\hat{x}_i = \hat{\Theta}^T k_i, \tag{15}$$

where $\hat{U}\prime = [\hat{u}_1', \ldots, \hat{u}_N']$ and $\hat{\Theta} = \left[\hat{\theta}_1, \ldots, \hat{\theta}_N\right]$. It is direct that $\hat{L} = \tilde{L}$ since the similarity matrix $\hat{X}^T\hat{X} = KBB^T K = K$ is the same as in KDLA. Therefore, (14) is equivalent to (10) and $\hat{\theta} = \tilde{\theta}$. Then we have $\hat{y}_i = \tilde{y}_i$, which ends the proof. □

4. Experiments

We conduct experiments of SHCCR which contains the following three steps:

Table III
Best recognition rates. The number in the parentheses is the reduced dimensions.

	SET 1	SET 2	SET 3	SET 4	SET 5	SET 6	SET 7	SET 8	SET 9	SET 10	
$N_i = 30$	LDA	0.797(9)	0.775(9)	0.782(9)	0.844(9)	0.877(9)	0.781(9)	0.789(8)	0.853(9)	0.828(8)	0.802(9)
	SLPP	0.779(20)	0.761(10)	0.722(18)	0.844(20)	0.858(9)	0.748(13)	0.774(19)	0.848(11)	0.827(19)	0.756(20)
	MFA	0.81(11)	0.805(10)	0.802(11)	0.862(12)	0.877(14)	0.805(18)	0.798(12)	0.875(19)	0.854(10)	0.823(10)
	DLA	0.94(9)	0.933(9)	0.927(10)	0.945(9)	0.927(12)	0.921(9)	0.899(8)	0.931(10)	0.926(8)	0.94(9)
	KDLA	0.949(9)	0.951(15)	0.936(9)	0.951(9)	0.935(11)	0.932(9)	0.898(9)	0.938(9)	0.946(15)	0.943(12)
$N_i = 80$	LDA	0.869(9)	0.873(9)	0.866(9)	0.909(9)	0.888(9)	0.869(9)	0.866(9)	0.889(9)	0.882(9)	0.897(9)
	SLPP	0.879(19)	0.877(20)	0.872(15)	0.916(11)	0.895(15)	0.88(14)	0.866(20)	0.897(19)	0.894(20)	0.898(16)
	MFA	0.881(18)	0.883(20)	0.87(11)	0.921(10)	0.889(15)	0.876(17)	0.866(11)	0.901(12)	0.882(10)	0.888(20)
	DLA	0.956(9)	0.94(9)	0.941(9)	0.957(10)	0.924(10)	0.938(11)	0.9(12)	0.933(10)	0.934(9)	0.94(9)
	KDLA	0.958(13)	0.958(9)	0.952(14)	0.961(9)	0.948(9)	0.946(9)	0.914(11)	0.939(15)	0.952(9)	0.962(14)

(a) Similar samples collection and feature extraction: In this paper, the benchmark dataset is the SCUT-COUCH2009 dataset (Jin et al., 2011). SCUT-COUCH2009 is an online unconstrained Chinese handwriting dataset, which contains 11 subsets of different vocabularies, including GB1, GB2, Letters, Digit, Symbol, Word8888 etc., and all the samples are collected from more than 190 subjects. In the following experiments, the GB1 subset is used, which contains 3755 frequently used simplified Chinese characters in GB-2312-80 standard. Ten difficult recognition collections of similar handwritten Chinese characters are obtained by using the aforementioned SCG method (Liu and Jin, 2007). Then the elastic meshing (ELM) technique (Jin and Wei, 1998) is utilized as a normalized method to solve the stroke location and shape variability of intra-class character and the 8-directional features (Bai and Huo, 2005) are extracted with $D = 512$ dimensions. We randomly divided each similar samples collection into two separate subsets, which are the training set and the test set. We utilize one of similar character collections to tune the model parameters of DLA and KDLA. Table I lists the ten similar character sets we used in the following experiments. Fig. 4 shows some of the corresponding handwritten samples of Table I.

(b) Discriminative information extraction: We evaluate the performance of DLA and KDLA by comparing them with three representative algorithms, including LDA (Fisher, 1936), supervised locality preserving projections (SLPP) (Cai et al., 2005) and marginal fisher analysis (MFA) (Xu, 2007). These algorithms have certain merits in their own rights. LDA is a linear algorithm. SLPP, MFA and DLA are all popular manifold learning algorithms which perform better than LDA in many practical applications. It is worth noting that we employ principal component analysis (PCA) (Hotelling, 1933) to remove redundant information before we conduct LDA, LPP, MFA, and DLA. In the PCA step, we retain $N - C$ dimensions to ensure $X(D^p - W^p)X^T$ (Xu, 2007) in MFA and within-scatter matrix S_w in LDA (Lu et al., 2003) are non-singular, because the number of the original features of the training samples is much larger than the number of training samples. We retain $N - 1$ dimensions in SLPP and DLA to preserve all the energy in this step in order to accelerate the learning process. For KDLA, we use 100 leading eigenvectors of KPCA to form the space for the subsequent DLA. In our experiments, we select Gaussian kernel and σ is empirically set to 6. The implementation of KDLA is based upon the theorem. In particular, we conduct KPCA followed by DLA in the space spanned by the principal components of KPCA.

(c) Classification: minimum euclidean distance classifier (MEDC) is used for recognition.

4.1. Parameters selection for DLA and KDLA

Since the parameters setup for DLA is essential for its performance, we carried out the DLA parameter optimization experiments before we conduct SHCCR. We aim to find a proper range for the dominant parameters m_1 and m_2 in DLA, wherein m_1 is the number of the samples from intra-class in a given patch, and m_2 is the number of the samples from inter-classes in the same patch. Parameter β is set to an empirical value 0.15 and the reduced dimension is set to 9.

Suppose N_i is the number of the training samples in the ith class and N is the number of the total training samples. Then, m_1 and m_2 can be chosen in the ranges of $[1, N_i - 1]$ and $[1, N - N_i]$, respectively.

Fig. 5 shows the recognition rate against different m_1 and m_2 on the similar character collection "氨". When $N_i = 30$, different combinations of m_1, m_2 pairs result in different recognition rates. It is worth noting that the red region represents the best performance obtained by DLA. The best combination of m_1 and m_2 is $m_1 = 29$ and $m_2 = 50$, and the corresponding accuracy is 95.82%. When the parameters are set to $m_1 = 10$ and $m_2 = 30$, the recognition rate reaches 95.4%, which slightly lower than the best recognition rate. In this paper, we choose to use this sub-optimal setting $m_1 = 10$ and $m_2 = 30$ in the following experiments for the other similar character sets to save computational cost. When $N_i = 80$, we use the same setting.

Parameters of KDLA are tuned in a way similar to the above procedure used to tune parameters of DLA.

4.2. Evaluation experiments for SHCCR

In the experiments, we evaluate the performance of DLA and KDLA by comparing with three representative algorithms, including LDA, SLPP and MFA. In the training stage, we randomly selected (30,80) training samples for each class from the ten similar characters collection listed in Table I. Afterward, we randomly selected 100 samples for each class for test. The training set and the test set are disjoint. For different algorithms, we used the same training set and test set for performance evaluation. Fig. 6 shows the recognition rate versus reduced dimensionalities for the ten similar character sets. We observe that DLA/KDLA outperform others. We also carry out experiments on the candidate sets directly without dimensionality reduction. The average recognition rates over ten similar character sets are 89.3% and 90.7% for 30 and 80 training samples settings respectively.

For comparison convenience, we arrange the experiment results in Tables II and III for the ten similar character sets. Table II lists the average recognition rates over ten similar character sets for all the five algorithms under two different settings. Table III lists the best recognition rate with the corresponding reduced dimensionalities for all the five algorithms under two different settings.

In addition, to further illustrate the recognition accuracy of the proposed dimensionality reduction algorithm, we have compared DLA and KDLA with the state-of-the-art algorithms, such as linear and nonlinear support vector machines (SVM) (Geng, 2012; Tao et al., 2006). In our experiments, we used LIBSVM (Chang and Lin, 2001) to conduct the SVM classification experiments. The linear kernel and the RBF kernel have used in SVM. Fig. 7 reports the associated recognition performances with statistical significance.

4.3. Analysis of the results

From Fig. 5, Tables II and III, the performance of DLA in SHCCR can be analyzed in three aspects: first, in Fig. 5 and Table II, it is shown that in the same reduced dimensions, the recognition rates of both DLA and KDLA are significantly higher than other algorithms. It also shows when the recognition rate is in the same level, DLA and KDLA have a better performance of discriminative information extraction than others under the same condition. For example, we can see from Table II that the DLA and KDLA with reduced dimension of 4 outperforms LDA with reduced dimension of 9 under the condition that the N_i is set to 30. Second, in Fig. 5 and Table II, it can be seen that in the same reduced dimensions, recognition rates of DLA and KDLA under the different condition of N_i have just a little variation; whereas the recognition rates of other algorithms vary greatly. It demonstrates that the robustness of DLA and KDLA is better than other algorithms under the condition of small size of training samples. Last but not the least, in Fig. 5, it is shown that KDLA can further improve the performance of discriminative feature extraction.

In Fig. 7, we show the recognition rates boxplots of different methods. There are two subfigures, each of which corresponds to the performance obtained from a particular number (30,80) of labeled training samples of each class. These boxplots show KDLA plus minimum Euclidean distance classifier is superior to the baseline methods. It suggests the effectiveness of KDLA for discriminative feature extraction.

Similar to other kernel methods, KDLA needs more time and space costs. The space costs mainly raised by the number of training samples and the eigenvectors of each similar Chinese character sets. In general, kernel methods need to keep all the training samples to construct the kernel Gram matrix. For example, if the class number is 3755, feature dimension is 512, the number of training samples of each class is 80, when the 4-byte floating number is used, the space cost is $3755 \times 512 \times 80 \times 4$ bytes, about 586 M. In addition, if the projection matrix W contains 12 eigenvectors, and the number of candidate characters is 10, the space cost is $3755 \times 80 \times 10 \times 12 \times 4$ bytes, about 137 M. We conduct experiments on a Core2 E7500 2.93 GHz computer with a 4-Gbyte memory to test KDLA's time cost. All experiments were done in Matlab. Different numbers (30,80) of training samples were selected for each class and the number of candidates is set to $k = 10$. The computational time of 1000 test samples is 0.08 s and 0.17 s, respectively. Given limited storage resource in a smart phone, in general, it is impossible to run an SHCCR system based on KDLA. However, with the remarkable progress of the Internet technology, cloud computing (Vaquero et al., 2011) becomes mature and already offers a lot of services to public services (Gao et al., 2011). Thus, KDLA is applicable on the cloud computing for smart phone users.

5. Conclusion

In this paper, a new manifold learning based subspace learning algorithm, discriminative locality alignment (DLA), has been introduced to similar handwritten Chinese character recognition (SHCCR). Afterward, we propose the kernel version of DLA, Kernel discriminative locality alignment (KDLA), and carefully prove that learning KDLA is equal to conducting KPCA followed by DLA. Comparing to conventional LDA and representative manifold learning based discriminative dimension reduction algorithms, such as supervised LPP and MFA, DLA and KDLA have shown many competitive and attractive properties, and they are superior to these algorithms in terms of recognition accuracy. From our experiments, we have the following observations:

(1) DLA and KDLA consistently achieve better classification performance than representative algorithms in the SHCCR experiments;
(2) In SHCCR, DLA and KDLA are robust and promising, and have no matrix singular problem;
(3) DLA and KDLA are potentially useful for real world applications, because they perform well with a smaller size projection matrix than that obtained by LDA. That results in a much lower storage cost with higher recognition performance.
(4) By proving the equivalence between KDLA and KPCA combined with DLA, we can better understand KDLA, i.e., the subspace spanned by KDLA is equal to the subspace spanned by DLA on the principal components of KPCA.

Acknowledgment

We would like to thank all anonymous reviewers for their valuable suggestions. This work is supported in part by NSFC (Grant no. 61075021), GDNSF (No. S2011020000541) and GDSTP (No. 2010B090400397, 2011B090400146).

References

Bai, Z., Huo, Q., 2005. A study on the use of 8-directional features for online handwritten Chinese character recognition. ICDAR, 232–236.

Belkin, M., Niyogi, P., 2002. Laplacian eigenmaps and spectral techniques for embedding and clustering. Adv. Neural Inform. Process. Systems, 585–591.

Bengio, Y., Paiement, J., Vincent, P., Dellallaeu, O., Roux, L., Quimet, M., 2004. Out-of-sample extensions for LLE, isomap, MDS, eigenmaps, and spectral clustering. Adv. Neural Inform. Process. Systems.

Bian, W., 2011. Max–min distance analysis by using sequential SDP relaxation for dimension reduction. IEEE Trans. Pattern Anal. Mach. Intell. 33 (5), 1037–1050.

Cai, D., He, X., Han, J., 2005.''Using graph model for face analysis'', Department of Computer Science, Technical Report No. 2636, University of Illinois at Urbana-Champaign, September.

C.-C. Chang, C.-J. Lin, ''LIBSVM: a library for support vector machines'', <http://www.csie.ntu.edu.tw/~cjlin/libsvm>, 2001.

Fisher, R.A., 1936. The use of multiple measurements in taxonomic problems. Ann. Eugenics 7, 179–188.

Gao, T.-F., Liu, C.-L., 2008. High accuracy handwritten Chinese character recognition using LDA-based compound distances. Pattern Recognition 41 (11), 3442–3451.

Gao, Y., Jin, L., He, C., Zhou, G., 2011. Handwriting character recognition as a service: A new handwriting recognition system based on cloud computing. ICDAR, 885–889.

Geng, B., 2011. DAML: Domain adaptation metric learning. IEEE Trans. Image Process. 20 (10), 2980–2989.

Geng, B., 2012. Ensemble manifold regularization. IEEE Trans. Pattern Anal. Mach. Intell. 34 (6), 1227–1233.

Guan, N., 2011. Non-negative patch alignment framework. IEEE Trans Neural Networks 22 (8), 1218–1230.

Guan, N., 2012. Online non-negative matrix factorization with robust stochastic approximation. IEEE Trans. Neural Networks Learning Systems. http://dx.doi.org/10.1109/TNNLS.2012.2197827.

Guan, N., 2012. NeNMF: An optimal gradient method for nonnegative matrix factorization. IEEE Trans. Signal Process. 60 (6), 2882–2898.

He, X., Niyogi, P., 2004. Locality preserving projections. Adv. Neural Inform. Process. Syst..

Hotelling, H., 1933. Analysis of a complex of statistical variables into principal components. J. Education. Psychol. 24, 417–441.

Jin, L., Wei, G., 1998. Handwritten Chinese character recognition with directional decomposition cellular features. J. Circuit System Comput. 8 (4), 517–524.

Jin, L., Ding, K., Huang, Z., 2010. Incremental learning of LDA model for Chinese writer adaptation. Neural Comput. 73, 1614–1623.

Jin, L., Gao, Y., Liu, G., Li, Y., Ding, K., 2011. A comprehensive online unconstrained Chinese handwritten database and benchmark evaluation. IJDAR 14 (1), 53–64.

Jolliffe, I.T., 2002. Principal Component Analysis, 2nd edition. Springer-Verlag, New York.

Leung, K.C., Leung, C.H., 2010. Recognition of handwritten Chinese characters by critical region analysis. Pattern Recognition 43 (3), 949–961.

Liu, Z., Jin, L., 2007. ''A static candidates generation technique and its application in two-stage LDA Chinese character recognition'', In: Chinese Control Conference.

Liu, C.-L., Yin, F., Wang, D.-H., Wang, Q.-F., 2010. Chinese handwritten recognition contest 2010. CCPR, 1–5.

Lu, J., Plataniotis, K., Venetsanopoulos, A., 2003. Face recognition using LDA-based algorithms. IEEE Trans. Neural Networks 14 (1), 195–200.

Muller, K., Mika, S., Ratsch, G., Tsuda, K., Scholkopf, B., 2001. An introduction to kernel-based learning algorithms. IEEE. Trans. Neural Networks 12 (2), 181–201.

Schölkopf, B., Smola, A.J., Müller, K.R., 1998. Nonlinear component analysis as a kernel eigenvalue problem. Neural Comput. 10, 1299–1319.

Schölkopf, B., Herbrich, R., Smola, A.J., 2001. ''A generalized representer theorem,'' In: Proceedings of the 14th Annual Conference on Computational Learning Theory, pp. 416–426.

Shao, Y., Wang, C., Xiao, B., Zhang, R., Zhang, L., 2011. Modified two-class LDA based compound distance for similar handwritten Chinese characters discrimination. ICDAR.

Shawe-Taylor, J., Cristianini, N., 2004. Kernel Methods for Pattern Analysis. Cambridge University Press.

Si, S., 2010. Bregman divergence-based regularization for transfer subspace learning. IEEE Trans. Knowl. Data Eng. 22 (7), 929–942.

Tao, D., Tang, X., Li, X., Wu, X., 2006. Asymmetric bagging and random subspace for support vector machines-based relevance feedback in image retrieval. IEEE Trans. Pattern Anal. Mach. Intell. 28 (7), 1088–1099.

Tao, D., Li, X., Wu, X., Maybank, S.J., 2007. General tensor discriminant analysis and gabor features for gait recognition. IEEE Trans. Pattern Anal. Mach. Intell. 29 (10), 1700–1715.

Tao, D., Li, X., Wu, X., Maybank, S.J., 2009. Geometric mean for subspace selection. IEEE Trans. Pattern Anal. Machine Intell. 31 (2), 260–274.

Tian, X., 2012. Sparse transfer learning for interactive video search reranking. ACM Trans. Multimedia Comput. Commun. Appl..

Vaquero, L.M., Caceres, J., Morán, D., 2011. The challenge of service level scalability for the cloud. IJCAC 1 (1), 34–44.

Wang, X., 2011. Subspaces indexing model on grassmann manifold for image search. IEEE Trans. Image Process. 20 (9), 2627–2635.
Wang, D.-H., Liu, C.-L., Yu, J.-L., Zhou, X.-D., 2009. CASIA-OLHWDB1: A database of online handwritten Chinese characters. ICDAR, 1206–1210.
Xu, D., 2007. Marginal Fisher analysis and its variants for human gait recognition and content-based image retrieval. IEEE Trans. Image Process. 16 (11), 2811–2821.
Zhang, Z., Zha, H., 2004. Principal manifolds and nonlinear dimension reduction via local tangent space alignment. SIAM J. Sci. Comput. 26 (1), 313–338.
Zhang, T., Tao, D., Yang, J., 2008. Discriminative locality alignment. ECCV, 725–738.
Zhang, T., Tao, D., Li, X., Yang, J., 2009. Patch alignment for dimensionality reduction. IEEE Trans. Knowledge Data Eng. 21 (9), 1299–1313.
Zhou, T., 2011. Manifold elastic net: A unified framework for sparse dimension reduction. Data Min. Knowl. Discov. 22 (3), 340–371.