# Transferred Dimensionality Reduction

Zheng Wang, Yangqiu Song, and Changshui Zhang

State Key Laboratory on Intelligent Technology and Systems
Tsinghua National Laboratory for Information Science and Technology (TNList)
Department of Automation, Tsinghua University, Beijing 100084, China

**Abstract.** Dimensionality reduction is one of the widely used techniques for data analysis. However, it is often hard to get a demanded low-dimensional representation with only the unlabeled data, especially for the discriminative task. In this paper, we put forward a novel problem of Transferred Dimensionality Reduction, which is to do unsupervised discriminative dimensionality reduction with the help of related prior knowledge from other classes in the same type of concept. We propose an algorithm named Transferred Discriminative Analysis to tackle this problem. It uses clustering to generate class labels for the target unlabeled data, and use dimensionality reduction for them joint with prior labeled data to do subspace selection. This two steps run adaptively to find a better discriminative subspace, and get better clustering results simultaneously. The experimental results on both constrained and unconstrained face recognition demonstrate significant improvements of our algorithm over the state-of-the-art methods.

**Keywords:** Transfer Learning, Dimensionality Reduction, Clustering.

## 1 Introduction

In many machine learning applications, such as computational biology, appearance-based image recognition and image retrieval, one is confronted with high-dimensional data. However it is considered that the original data naturally reside on lower dimensional manifolds. Finding this compact representation is usually a key step. Using an efficient representation, the subsequent phases, such as clustering or classification, will become much faster and more robust [14]. Thus some dimensionality reduction approaches have been developed. For unsupervised methods, e.g. principle component analysis (PCA) [20] and locality preserving projections (LPP) [14], the compact manifold should preserve the most relevant structure information of the original data point cloud. For supervised case, e.g. linear discriminant analysis (LDA) [1], the low-dimensional representation should find the most discriminative subspace for different classes based on the labeled data. Recently, the semi-supervised method has also been developed [3], which makes use of both labeled and unlabeled data.

In the last few years, several similar works [26,9,23] have been done to couple unsupervised dimensionality reduction with clustering, forming an adaptive dimensionality reduction framework. It performs discriminant analysis and clustering adaptively to select the most discriminative subspace and find a suitable clustering simultaneously. The most recent work [26] uses the method called discriminative k-means (DisKmeans),

W. Daelemans et al. (Eds.): ECML PKDD 2008, Part II, LNAI 5212, pp. 550–565, 2008.

which outgoes the traditional PCA+K-means framework and other similar works in their experiments. However, we observe that this type of methods is efficient only for specific data distributions, which is very limited. For example, we show three cases of a toy problem in Fig. 1.

To alleviate this limitation, additional prior information should be considered. The most straightforward and powerful information is the label, such as the idea of supervised and semi-supervised methods. However, in practice, the label information for these target unknown classes may hardly be obtained. The works from knowledge transfer [22] inspire us to make use of the information from other class domains prior known. Though from different classes, the labeled samples may share some common characteristics with the target task, as they are from the same type of concept.

For example, in face recognition, we want to detect or recognize the face images for a number of persons. When they are all unlabeled, the conventional methods usually cannot get satisfied results, as they cannot use any supervised information. On the other hand, there are already some databases with labeled faces, such as AT&T [18] and Yale [13]. These labeled face data contain some common information for face recognition. So we can use them to improve the original unsupervised learning task. In this situation, though both labeled and unlabeled data appear, the previous semi-supervised methods cannot work, as the labeled and unlabeled data are from different classes. This is a more general problem of learning with both labeled and unlabeled data [15].

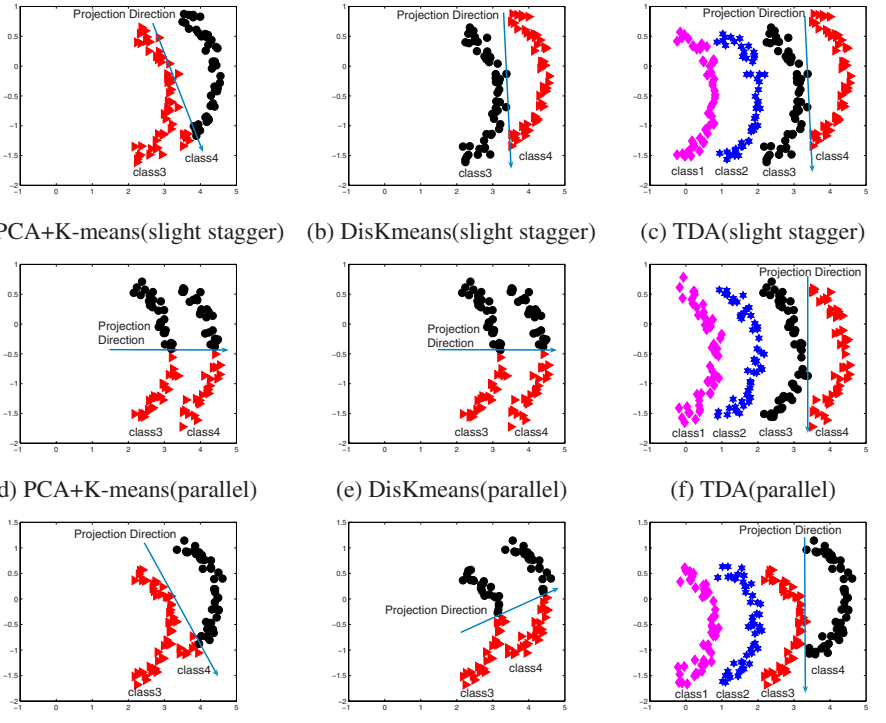
This problem brings forward a novel issue which we call transferred dimensionality reduction (TDR). It transfers the task-related information from the classes prior known to the target unlabeled class domains, and finds a better subspace to discriminate them. In this paper, we propose a method called transferred discriminative analysis (TDA) to tackle the TDR problem. This method extracts the discriminative information from the labeled data and transfers it into unsupervised discriminative dimensionality reduction to revise the results iteratively. Finally, using both these labeled and unlabeled data from different classes, we can find the most discriminative subspace and an optimal clustering result simultaneously. The toy problem in Fig. 1 explains this problem more intuitively. It shows that, the labeled samples from known classes can help us to find a much better subspace to discriminate the unknown classes.

The rest of the paper is organized as follows. In section 2, we briefly review the related works. Then we introduce TDA algorithm in section 3. Experiments are given in section 4. Finally, we give our conclusion and suggest some future works based on the novel problem of TDR in section 5.

## 2 Related Work

### 2.1 Discriminative Dimensionality Reduction and Clustering

Over the past few decades, a lot of attention has been paid to dimensionality reduction. Some algorithms have been developed. A large family of them can be explained in a graph view [25]. The low-dimensional vector representation can be obtained from the eigenvectors corresponding to the eigenvalues of the graph Laplacian matrix with certain constraints. It preserves similarities between the pairs of the data, where




(a) PCA+K-means(slight stagger) (b) DisKmeans(slight stagger) (c) TDA(slight stagger)

(d) PCA+K-means(parallel) (e) DisKmeans(parallel) (f) TDA(parallel)

(g) PCA+K-means(bigger stagger) (h) DisKmeans(bigger stagger) (i) TDA(bigger stagger)

**Fig. 1.** Toy problem: There are four classes of data. Each class contains 50 random samples and forms a moon shape manifold. Suppose the class 3 and 4 are unlabeled, and we want to find the suitable subspace to discriminate them. There are three situations, each one in a row. PCA+K-means framework fails for any case, as in (a), (d) and (g). DisKmeans only works for the case that class 3 and 4 are slightly staggered in (b). When they are paralleled in (e) or staggered too much in (h), it cannot work well either. However, with the help of class 1 and 2, which are labeled beforehand, we can find the suitable subspace for each case as in (c), (f) and (i).

similarity is measured by a graph similarity matrix that characterizes certain statistical or geometric properties of the data set.

To get the discriminative structure of data, supervised methods try to find a transformation that minimizes the within-class scatter and maximizes the between-class scatter simultaneously. Given $l$ labeled samples $\mathbf{X}_L = (\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_l)$ from C classes, where $\boldsymbol{x_i} \in \mathbb{R}^d$. The within-class scatter matrix $\mathbf{S}_w$, the between-class scatter matrix $\mathbf{S}_b$ and the total-scatter matrix $\mathbf{S}_t$ are defined as:

$$\mathbf{S}_w = \sum_{j=1}^{C} \sum_{i=1}^{l_j} (\boldsymbol{x}_i - \boldsymbol{m}_j)(\boldsymbol{x}_i - \boldsymbol{m}_j)^T = \mathbf{X}\mathbf{L}_w\mathbf{X}^T \tag{1}$$

$$\mathbf{S}_b = \sum_{j=1}^{C} l_j(\boldsymbol{m}_j - \boldsymbol{m})(\boldsymbol{m}_j - \boldsymbol{m})^T = \mathbf{X}\mathbf{L}_b\mathbf{X}^T \tag{2}$$

$$\mathbf{S}_t = \sum_{i=1}^{l} (\boldsymbol{x}_i - \boldsymbol{m})(\boldsymbol{x}_i - \boldsymbol{m})^T = \mathbf{S}_b + \mathbf{S}_w = \mathbf{X}\mathbf{L}_t\mathbf{X}^T, \tag{3}$$

where $\boldsymbol{m}_j = \frac{1}{l_j}\sum_{i=1}^{l_j}\boldsymbol{x}_i$ $(j = 1, 2, ..., C)$ is the mean of the samples in class $j$, $l_j$ is the number of samples in class $j$, and $\boldsymbol{m} = \frac{1}{l}\sum_{i=1}^{l}\boldsymbol{x}_i$ is the mean of all the samples. And the corresponding graph Laplacians are [14]:

$$\mathbf{L}_w = \mathbf{I} - \mathbf{H}(\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T \tag{4}$$

$$\mathbf{L}_b = \mathbf{H}(\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T - \tfrac{1}{l}\mathbf{1}_l\mathbf{1}_l^T \tag{5}$$

$$\mathbf{L}_t = \mathbf{I} - \tfrac{1}{l}\mathbf{1}_l\mathbf{1}_l^T, \tag{6}$$

where $\mathbf{H} = \{0, 1\}^{l\times C}$ is an indicator matrix: $H_{ij} = 1$ if $x_i$ belongs to the $j$-$th$ class, and $H_{ij} = 0$ otherwise.

LDA is one of the most popular and representative supervised methods. It is to solve the optimization problem:

$$\max_W \text{trace}((\mathbf{W}^T\mathbf{S}_w\mathbf{W})^{-1}(\mathbf{W}^T\mathbf{S}_b\mathbf{W})). \tag{7}$$

or

$$\max_W \text{trace}((\mathbf{W}^T\mathbf{S}_t\mathbf{W})^{-1}(\mathbf{W}^T\mathbf{S}_b\mathbf{W})). \tag{8}$$

The solution is the eigenvectors corresponding to the $C-1$ largest eigenvalues of $\mathbf{S}_w^{-1}\mathbf{S}_b$ or $\mathbf{S}_t^{-1}\mathbf{S}_b$ [11].

Clustering is another important topic to exploit the discriminative structure of the data. K-means is one of the simplest and most popular algorithms to solve the clustering problem. Given $u$ unlabeled samples $\mathbf{X}_U = (\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_u)$ from K classes. Standard k-means finds the partition of the data to minimize the energy function:

$$J_K = \sum_{k=1}^{K}\sum_{i\in C_k}\|x_i - m_k\|_2^2 = \text{trace}(\mathbf{S}_w). \tag{9}$$

The clustering state can also be specified by an dummy indicator matrix $\widetilde{\mathbf{H}}^{u\times K}$.

It is clear the k-means clustering is to minimize the within-class scatter matrix $\mathbf{S}_w$, or maximize the between class scatter matrix $\mathbf{S}_b$, since the total scatter $\mathbf{S}_t$ is a constant. It also can be represented in graph form using equation (4). On the other hand, its kernelized version can also be explained under the graph view, which has close connection with other spectral clustering methods [8].

The discriminative analysis and clustering methods all emphasize on pursuing the intrinsic discriminative structure of the data. So [9,23,26] combine them together to get better learning result.

Though the combined method of discriminative k-means does a good job in some situations. It focuses too much on the present unlabeled samples, and sometimes is trapped into a very bad result, even worse than the PCA+K-means method, which is shown in the third case in Fig. 1. To overcome this problem, we consider to introduce more information from outer classes within the same concept domain. As the different classes of data in the same concept domain often lie on similar lower dimensional manifolds in certain extent, they should share some common discriminative structure. We can extract this structure easily from the labeled classes using discriminative analysis. Then, we can transfer the shared information to the unlabeled data, and find their discriminative structure using the clustering method.

### 2.2 Transfer Learning and Semi-supervised Learning

TDA has a similar motivation with knowledge transfer, or transfer learning, which has been recognized as an important topic in machine learning field. It is the ability to apply knowledge and skills learned in previous tasks to novel tasks. Early works raised some significative issues [17,21,4]. There are still more and more attentions paid to this topic recently [16,7]. Most of the previous works focus on transferring the related knowledge for supervised learning tasks. In this work, however, we address on the single-task problem, and transfer the supervised information to unsupervised task. Though it seems like semi-supervised learning [5], they have obvious distinctions. In traditional semi-supervised learning the labeled and unlabeled data come from the same class domains. There should be both labeled and unlabeled data in each class. The unlabeled data should have the same distribution with the labeled ones, then a large number of data points can expose the manifold information and improve the learning result of the labeled data.

In our problem, on the contrary, the labeled and unlabeled data are from different classes, and they have different distributions. We extract the useful discriminative information from the labeled data to improve the subspace selection of the unlabeled data. It is quit different with semi-supervised learning and cannot be solved using existing semi-supervised methods. As a result, we name this problem as transferred dimensionality reduction.

## 3 Transferred Discriminative Analysis

In learning tasks, it is vital to use the prior information. In traditional methods, the prior is often assumed to be given by the designer's experience. We cannot expect this prior to be always right, as it is hard to propose a suitable prior even for an expert. However, in TDR we extract the information directly from the data prior known, and embed this information into the task using the cross similarity part between the source prior known and the target to be dealt with.

In TDR, suppose we have the labeled source data set, contains $l$ points $\mathbf{D_L} = \{\mathbf{X_L}, \mathbf{Y_L}\}$, $\mathbf{X}_L = (\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_l)$, $\mathbf{Y}_L = (\boldsymbol{y}_1, \boldsymbol{y}_2, ..., \boldsymbol{y}_l)^T$. The label is $\boldsymbol{y} \in \{1, \ldots, C\}$. We want to find the compact subspace of $u$ newly arrived unlabeled points $\mathbf{D_U} = \{\mathbf{X_U}\}$, $\mathbf{X}_U = (\boldsymbol{x}_{l+1}, \boldsymbol{x}_{l+2}, ..., \boldsymbol{x}_{l+u})$, from K classes which are different from the classes in $\mathbf{Y_L}$. Each point $\boldsymbol{x} \in \mathbb{R}^d$ is a $d$-dimensional vector. We denote all data as $\mathbf{D} = \{\mathbf{D_L}, \mathbf{D_U}\}$, and $\mathbf{X} = \{\mathbf{X_L}, \mathbf{X_U}\}$. For simplicity, we assume $n = l + u$, and the sample mean of $\mathbf{D}$ is zero, which is $\boldsymbol{m} = \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{x}_i = 0$.

### 3.1 The Objective Function

The manifold structure is interpreted as that nearby points will have similar embedding. As the labeled and unlabeled data are from different manifolds of the same concept domain. The discriminative structure can be shared to some extent among this manifolds. We can transfer this information from source data $\mathbf{D_L}$ to target data $\mathbf{D_U}$ through the intervention between these two parts.

In our TDR, we measure the between-class information of the data set $\mathbf{D}$ as follows:

$$\mathbf{S}_b = \mathbf{S}_{bl} + \widetilde{\mathbf{S}}_{bu} = \sum_{i=1}^{C} l_i \boldsymbol{m}_i \boldsymbol{m}_i^T + \sum_{j=1}^{K} l_j \widetilde{\boldsymbol{m}}_j \widetilde{\boldsymbol{m}}_j^T. \tag{10}$$

The first part is the between-class scatter of the labled data. However, for the unlabeled data, we can estimate this information using clustering method, which is expressed as the second part, treating each cluster as a class.

In the between-class scatter, the labeled and unlabeled parts are separately presented. To properly describe the structure of all data, we should introduce the relationship between labeled and unlabeled parts.

Under the existence of unlabeled data, Graph Laplacian has been generally used to describe the data structure [5]. We define $G = (V, E)$ as a graph associated with the data. $V$ is the vertex set of graph, which is defined on the observed set, including both labeled and unlabeled data. $E$ is the edge set, which contains the pairs of neighboring vertices $(\boldsymbol{x}_i, \boldsymbol{x}_j)$. A typical adjacency matrix $\mathbf{M}$ of neighborhood graph is defined as:

$$\mathbf{M}_{ij} = \begin{cases} \exp\{-\frac{\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2}{2\sigma^2}\} & if\ (\boldsymbol{x}_i, \boldsymbol{x}_j) \in E \\ 0 & otherwise \end{cases} \tag{11}$$

then the normalized graph Laplacian [6] is:

$$\mathbf{L} = \mathbf{I} - \mathbf{D}^{-\frac{1}{2}} \mathbf{M} \mathbf{D}^{-\frac{1}{2}}, \tag{12}$$

where the diagonal matrix $\mathbf{D}$ satisfies $\mathbf{D}_{ii} = d_i$, and $d_i = \sum_{j=1}^{l+u} \mathbf{M}_{ij}$ is the degree of vertex $\boldsymbol{x}_i$.

Introducing the graph Laplacian into the total scatter, we can make use of both labled and unlabeled information to describe the structure of the data set $\mathbf{D}$ properly. With the zero sample mean, it becomes

$$(\mathbf{S}_t + \lambda \mathbf{X} \mathbf{L} \mathbf{X}^T) = \mathbf{X}(\mathbf{I} + \lambda \mathbf{L}) \mathbf{X}^T \tag{13}$$

It is also can be seen in the regularization of discriminative analysis [10].

As described above, the target of TDA becomes:

$$\max_{\mathbf{W}, \widetilde{\mathbf{H}}_u} \text{trace}((\mathbf{W}^T(\mathbf{S}_t + \lambda \mathbf{X} \mathbf{L} \mathbf{X}^T)\mathbf{W})^{-1}(\mathbf{W}^T(\mathbf{S}_{bl} + \widetilde{\mathbf{S}}_{bu}(\widetilde{\mathbf{H}}_u))\mathbf{W})). \tag{14}$$

It is to optimize the objective function w.r.t two variables. One is the dummy indicator matrix $\widetilde{\mathbf{H}}_u$, representing the clustering structure of the unlabled data, and the other one is the projection direction $\mathbf{W}$ for the dimensionality reduction.

Direct optimizing the objective function is complex and not advisable. Instead, we optimize it alternatively. We can use clustering method to estimate the discriminative structure of the unlabeled data, and project all data into lower dimension by supervised method to revise the clustering result. Using the method in [23], the process will converge to the optimal solution for the objective, while we will using the k-means clustering in our experiment, which gives a local solution but is good enough.

The introduction of the labeled parts in between-class scatter, total scatter and graph Laplcian adds more restriction into the problem. They restrict that in the low-dimensional subspace of unlabeled data, the discriminative structure of labeled data should still be preserved. The labeled data will bring in punishment if the structure is violated. This will force the unlabeled data clustering to form similar discriminative structure with the labeled data, and the information is transferred like this. The alternation process will stop, when the structure consistency of all data in the subspace and the clustering structure within unlabeled data are balanced. Following this process, the knowledge is transferred through the intervention between the labeled and unlabeled structures, and then affects the clustering and projection process.

The above explanation is intuitive. We can also explanation this intervention more explicitly from kernel learning view. [26] analyzes that the clustering step of the adaptive framework is just the kerneled version of k-means clustering, using kernel matrix

$$\mathbf{X_U^T W}(\mathbf{W^T}(\mathbf{X_U X_U^T} + \lambda \mathbf{L_U})\mathbf{W})^{-\mathbf{1}}\mathbf{W^T X_U}, \tag{15}$$

which is learned from the unlabeled data. In our method, the kernel matrix becomes

$$\mathbf{X^T W}(\mathbf{W^T}(\mathbf{X X^T} + \lambda \mathbf{L})\mathbf{W})^{-\mathbf{1}}\mathbf{W^T X}, \tag{16}$$

which is learned from all available data, both the source and target. So, the prior information from the source is embedded in the kernel matrix and transferred adaptive to the target task. Finally, we can find the most discriminative projection direction, and get a reasonable clustering result of the unlabeled data at the same time.

### 3.2 The Algorithm

Given the labeled data $\mathbf{D}_L = \{\mathbf{X}_L, \mathbf{Y}_L\}$ belong to C classes, and unlabeled data $\mathbf{D}_U = \{\mathbf{X}_U\}$ with their class number K. The TDA algorithm is stated below:

**Step 1. Initialization:** Initially assign the cluster index for the K classes of unlabeled data using k-means. Construct the graph matrix **M** as in equation (11), and calculate the graph Laplacian **L** as in equation (12).

**Step 2. Supervised Dimensionality Reduction:** Find the optimal subspace with dimension $m = C + K - 1$, using eigenvalue decomposition for the objective function (14) w.r.t **W**, which is similar to LDA. Then the optimal solution is given by:

$$(\mathbf{S}_t + \lambda_1 \mathbf{X L X}^T + \lambda_2 \mathbf{I})\boldsymbol{w}_j^* = \eta_j(\mathbf{S}_{bl} + \widetilde{\mathbf{S}}_{bu}(\widetilde{\mathbf{H}}_u))\boldsymbol{w}_j^*,$$

$j = 1, ..., m$, where $\boldsymbol{w}_j^*$ $(j = 1, ..., m)$ are the eigenvectors corresponding to the $m$ largest eigenvalues of $(\mathbf{S}_t + \lambda_1 \mathbf{X L X}^T + \lambda_2 \mathbf{I})^{-1}(\mathbf{S}_{bl} + \widetilde{\mathbf{S}}_{bu}(\widetilde{\mathbf{H}}_u))$, with fixed $\widetilde{\mathbf{H}}_u$. $\lambda_2 \mathbf{I}$ is a regularization term, which ensures the nonsingularity of the matrix $\mathbf{S}_t + \lambda_1 \mathbf{X L X}^T$, and $\lambda_2$ is an arbitrary small real number.

**Step 3. Compact Clustering for Target Data:** Cluster the unlabeled data in the subspace finding in step 2. It is to fix projection direction **W** and use the clustering

method to get an optimal indicator matrix $\widetilde{\mathbf{H}}_u$ for the unlabeled data. K-means is used in this step to solve the problem

$$\max_{(\widetilde{\mathbf{H}}_u)} \widetilde{\mathbf{S}}_{bu}(\widetilde{\mathbf{H}}_u)$$

**Step 4. Stop Condition:** Goto step 2 until convergence. It is to stop when the clustering result, the indicator matrix $\widetilde{\mathbf{H}}_u$, for previous two iterations is unchanged.

**Step 5. TDA Embedding:** Let the projection matrix $\mathbf{W}_{tda} = [\boldsymbol{w}_1^*, \ldots, \boldsymbol{w}_m^*]$. The samples can be embedded into $m$ dimensional subspace by: $\mathbf{x} \rightarrow \mathbf{z} = \mathbf{W}_{tda}^T \mathbf{x}$.

### 3.3 Kernelization

In this section we present the generalized version of our algorithm using the kernel trick. We show a simple method under a graph view, using the similar treatment with [2]. It performs TDA in Reproducing Kernel Hilbert Space (RKHS), getting kernel TDA.

Let $\phi : x \rightarrow \mathcal{F}$ be a function mapping the points in the input space to feature space, which is a high-dimensional Hilbert space. We try to replace the explicit mapping with the inner product $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = (\phi(\boldsymbol{x}_i) \cdot \phi(\boldsymbol{x}_j))$. According to Representer Theorem [19], the optimal solution $\boldsymbol{w}_j^*$ can be given by:

$$\boldsymbol{w}_j^{\phi *} = \sum_{i=1}^{l+u} \boldsymbol{\alpha}_{ji}^* \phi(\boldsymbol{x}_i) \quad j = 1, ..., m \tag{17}$$

where $\boldsymbol{\alpha}_{ji}$ is the weight that defines how $\boldsymbol{w}_j^{\phi *}$ is represented in the space spanned by a set of over-complete bases $\{\phi(\boldsymbol{x}_1), \phi(\boldsymbol{x}_2), ..., \phi(\boldsymbol{x}_{l+u})\}$.

For convenience, we rewrite the data matrix in RKHS as $\mathbf{X}_L^\phi = [\phi(\boldsymbol{x}_1), \phi(\boldsymbol{x}_2), ..., \phi(\boldsymbol{x}_l)]$, $\mathbf{X}_U^\phi = [\phi(\boldsymbol{x}_{l+1}), \phi(\boldsymbol{x}_{l+2}), ..., \phi(\boldsymbol{x}_{l+u})]$, and $\mathbf{X}^\phi = (\mathbf{X}_L^\phi, \mathbf{X}_U^\phi)$. Then, $\mathbf{W}^\phi$ can be expressed as $\mathbf{W}^\phi = \mathbf{X}^\phi \boldsymbol{\alpha}$. The kernel matrices are defined as $\mathbf{K} = \mathbf{X}^{\phi T} \mathbf{X}^\phi$. Thus we have

$$\begin{aligned} \mathbf{W}^{\phi T} \mathbf{S}_b^\phi \mathbf{W}^\phi &= \boldsymbol{\alpha}^T \mathbf{K}^T \mathbf{L}_b \mathbf{K} \boldsymbol{\alpha} \\ \mathbf{W}^{\phi T} \mathbf{S}_t^\phi \mathbf{W}^\phi &= \boldsymbol{\alpha}^T \mathbf{K}^T \mathbf{I} \mathbf{K} \boldsymbol{\alpha} \\ \mathbf{W}^{\phi T} \mathbf{X}^\phi \mathbf{L} \mathbf{X}^{\phi T} \mathbf{W}^\phi &= \boldsymbol{\alpha}^T \mathbf{K}^T \mathbf{L} \mathbf{K} \boldsymbol{\alpha} \\ \mathbf{W}^{\phi T} \mathbf{W}^\phi &= \boldsymbol{\alpha}^T \mathbf{K} \boldsymbol{\alpha} \end{aligned} \tag{18}$$

Using the graph expression of (1) $\sim$ (6) and the graph Laplacian (12).

As

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}_l^{l \times C} & \mathbf{0} \\ \mathbf{0} & \widetilde{\mathbf{H}}_u^{u \times K} \end{bmatrix}, \mathbf{L}_b = \begin{bmatrix} \mathbf{L}_{bl}^{l \times C} & \mathbf{0} \\ \mathbf{0} & \widetilde{\mathbf{L}}_{bu}^{u \times K}(\widetilde{\mathbf{H}}_u) \end{bmatrix},$$

the indicator matrix composed with two parts for labeled and unlabeled samples individually, and the between-class scatter is also composed by two parts respectively.

We can then give the objective function of kernel TDA (KTDA) as:

$$\max_{\boldsymbol{\alpha}, \widetilde{\mathbf{H}}_u} \text{trace}((\boldsymbol{\alpha}^T \mathbf{K}^T (\mathbf{L}_t + \lambda_1 \mathbf{L} + \lambda_2) \mathbf{K} \boldsymbol{\alpha})^{-1} (\boldsymbol{\alpha}^T \mathbf{K}^T \mathbf{L}_b \mathbf{K} \boldsymbol{\alpha})). \tag{19}$$

The solution is obtained by solving the generalized eigenvalue decomposition problem:

$$\mathbf{K}^T(\mathbf{L}_t + \lambda_1\mathbf{L} + \lambda_2)\mathbf{K}\boldsymbol{\alpha}_j^* = \eta_j\mathbf{K}^T\mathbf{L}_b\mathbf{K}\boldsymbol{\alpha}_j^* \tag{20}$$

where $\boldsymbol{\alpha}^* = (\boldsymbol{\alpha}_1^*, \boldsymbol{\alpha}_2^*, ..., \boldsymbol{\alpha}_m^*)$ corresponds to the $m$ largest eigenvalues. $\boldsymbol{\alpha}_j^*$ should be resized as $\frac{1}{\sqrt{\boldsymbol{\alpha}_j^{*T}\mathbf{K}\boldsymbol{\alpha}_j^*}}\boldsymbol{\alpha}_j^*$ to satisfy the constraint of $\boldsymbol{\alpha}^{*T}\mathbf{K}\boldsymbol{\alpha}^* = \mathbf{I}$.

### 3.4 The Computational Complexity

The TDA contains both dimensionality reduction and clustering. The process may have several iterations. The empirical result shows it converges vary fast. The number of iterations is often less than ten. In supervised dimensionality reduction, it needs to solve a generalized eigenvalue decomposition, which is of order $O(d^2nt)$. $d$ is the dimension of data, $n = l + u$ is the number of total data points, and $t$ is the number of iterations. For clustering method, we use k-means. The computational complexity is $O(dnt)$. As a result, the total computational complexity is of order $O(d^2nt)$, and the complexity is focus on the original dimension of data. As a result, we can use PCA to initially reduce the dimension of the data, and this can accelerate the running speed of our algorithm. The computational complexity of kernel TDA is $O(n^2dt)$ analyzed in the same way.

## 4 Experiments

We have already shown a toy problem in the introduction. Using TDA we can find the true structure with the help of the labeled data using only a few iterations, which is very fast. In this case, the data prior known can exactly express the discriminative information of the unlabeled samples, which is an ideal situation.

In this section, however, we will give the examples of real problems and show that, most of the time, the transferred information is helpful. We perform the comparisons under the problem of face recognition, both constrained and unconstrained. We compare our TDA method with two of the most popular and representative appearance-based methods including *Eigenface* (based on PCA)[20] and *Laplacianface* (based on LPP)[14], and the adaptive dimensionality reduction method with clustering *DisKmeans* [26].

All images in our experiments are preprocessed to the same size of $56 \times 46$ pixels with 256 gray levels. In each experiment, we randomly choose C classes as labeled data, and K classes unlabeled. TDA runs on all of these data. The comparison methods are operated on the K classes of unlabeled data. We compare their clustering results in the corresponding subspace which they have found. TDA and DisKmeans can cluster the data at the same time of subspace selection. However for the other two methods, we use the k-means for clustering, and run k-means 10 times for each projection then choose the best result in each experiment. For each fixed (C, K), we run the experiment for 50 times, each time on randomly selected labeled and unlabeled classes, then show the average result. For comparison of different methods, we use the clustering result as the measurement of dimensionality reduction performance. We use two standard clustering

performance measures, which are Clustering Accuracy (ACC) and Normalized Mutual Information (NMI) [24,26].

The heuristic parameter in TDA and DisKmeans is the Laplacian weight $\lambda_1$. We set it to a fixed value of 1. As a matter of fact, the algorithm is not sensitivity to this parameter for a wide range. For the heuristic parameter of PCA and LPP, the reduced dimensionality, we choose them using cross validation.

### 4.1 What to Transfer

Usually there are several classes of labeled samples in the data set prior known. But not all of them are helpful for a specific unsupervised task. Because each of them has different discriminative structure. Only some of them are the same with the unlabeled samples. The others are not, and using these data is harmful, on the contrary. On the other hand, using all prior data needs much more computational time, which is not practical. As a result, we choose a proper subset of labeled data for our learning task. As the task is to maximize the discriminative ability of the target data, we just use this as the selection criterion. In following experiments, we randomly select C classes from the prior data set, and repeat for R times. Each time we will find an optimal pair of $(\mathbf{W_i}^T, \mathbf{H_{i}}_u)$, and use the best one. This is,

$$\max_{i \in R} \text{trace}((\mathbf{W_i}^T(\widetilde{\mathbf{S}}_{wu}(\widetilde{\mathbf{H_{i}}_u}) + \lambda \mathbf{X_U L X_U}^T)\mathbf{W_i})^{-1}(\mathbf{W_i}^T \widetilde{\mathbf{S}}_{bu}(\widetilde{\mathbf{H_{i}}_u})\mathbf{W_i})). \quad (21)$$

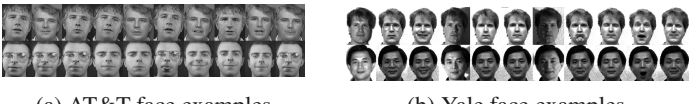
As a result, the computational complexity will be multiplied by R to $O(d^2 ntR)$. We fix R = 10. The complexity will not be changed significantly and remain in the same level.

### 4.2 Face Recognition Using Conventional Benchmarks

**Face Data Sets.** In the experiments for this section, we use the face data sets, AT&T [18] and Yale [13]. The typical faces of these data sets are shown in Fig. 2.

**Transferred within the Same Data Set.** In these experiments we use the labeled data and unlabeled data in the same data set.

For AT&T database, we chose each integer C from $\{2,\ldots,10\}$ and K from $\{2,\ldots,10\}$. Table 1 gives a part of the results using ACC measure as the limit of space. However, we show the result of all comparisons in both two measures in Fig. 3, where each point represent an average result of a fixed (C,K). We only show the improvement over DisKmeans in the figures, as it is the second best among all comparison methods. The results tell that TDA is much better than the unsupervised method. For Yale database, we chose C traversing all integers from $\{2,\ldots,7\}$, and K traversing from $\{2,\ldots,7\}$. The result is also shown in Fig. 3.



(a) AT&T face examples (b) Yale face examples

**Fig. 2.** Face Data Examples

**Table 1.** Results in AT&T, using ACC measure ($mean \pm std$)

| **AT&T** | PCA | LPP | DisKmeans | TDA |
|---|---|---|---|---|
| C=2,K=2 | 0.80(0.15) | 0.72(0.12) | 0.91(0.16) | **1.00(0.02)** |
| C=2,K=3 | 0.90(0.11) | 0.78(0.08) | 0.90(0.13) | **0.96(0.10)** |
| C=3,K=2 | 0.84(0.13) | 0.70(0.13) | 0.89(0.16) | **1.00(0.02)** |
| C=3,K=3 | 0.93(0.08) | 0.81(0.08) | 0.89(0.15) | **0.97(0.07)** |
| C=2,K=4 | 0.86(0.10) | 0.80(0.08) | 0.89(0.12) | **0.91(0.11)** |
| C=4,K=2 | 0.84(0.14) | 0.69(0.11) | 0.89(0.20) | **1.00(0.02)** |
| C=3,K=4 | 0.88(0.08) | 0.83(0.06) | 0.88(0.10) | **0.92(0.10)** |
| C=4,K=3 | 0.90(0.11) | 0.80(0.10) | 0.86(0.14) | **0.97(0.07)** |
| C=4,K=4 | 0.88(0.08) | 0.79(0.09) | 0.91(0.11) | **0.92(0.10)** |

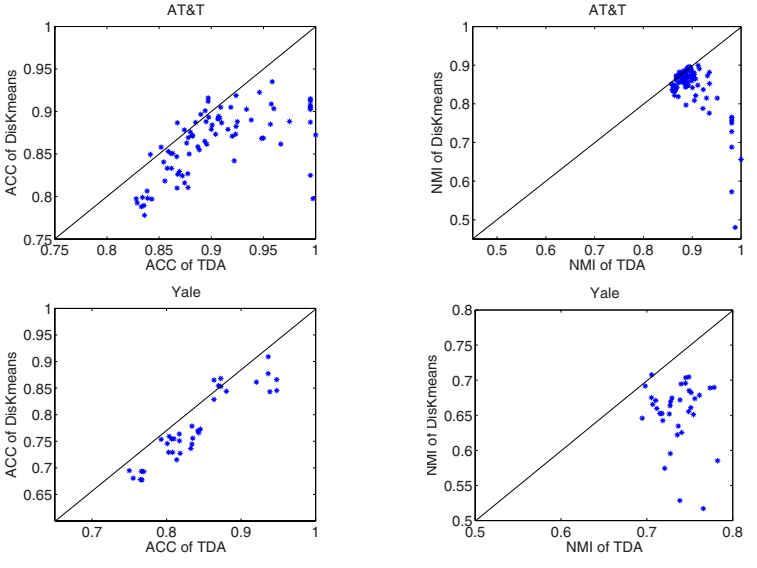



**Fig. 3.** Comparison results of TDA and DisKmeans in ACC and NMI measures, for transfer within either AT&T or Yale, each point represents 50 times average for a fixed (C, K) pair

As the above result cannot show how the change of (C,K) will affect the performance improvement, we give another representation in Table 2. It is the difference matrix between the clustering result of TDA and DisKmeans for each pair of (C,K). In Table 2, we can find that TDA improves significantly over other unsupervised methods for small K, which is the number of unlabeled classes. However, the improvement becomes less significant, with the increase of K. This is because the unknown target discriminative structure becomes more and more complex, and the limited prior cannot describe it properly. However, the increase of the number of labeled classes can not affect the result distinctively. This is because discriminative structure among the labeled data also

**Table 2.** Difference matrix of TDA and DisKmeans for AT&T, each element is calculated as $DM_{kc} = ACC_{kc}^{TDA} - ACC_{kc}^{Dis}$. The bold items show significant improvements of TDA.

| **AT&T** | C = 2 | C = 3 | C = 4 | C = 5 | C = 6 | C = 7 | C = 8 | C = 9 | C = 10 |
|---|---|---|---|---|---|---|---|---|---|
| K=2 | **0.22** | **0.29** | **0.41** | **0.34** | **0.22** | **0.23** | **0.25** | **0.51** | **0.22** |
| K=3 | **0.08** | **0.14** | **0.16** | 0.05 | 0.05 | **0.11** | **0.10** | **0.09** | **0.13** |
| K=4 | 0.02 | 0.06 | 0.02 | 0.04 | 0.01 | 0.02 | 0.03 | 0.06 | **0.09** |
| K=5 | 0.00 | 0.04 | **0.09** | 0.00 | 0.03 | 0.04 | 0.03 | 0.00 | 0.06 |

becomes more and more complex. On one hand it brings more information, on the other hand it contains some structure not consistent with the unlabeled data and may confuse the unsupervised dimensionality reduction. Another capable reason is the limit of the number of samples in each labeled class. There are only tens of samples in each labeled class, which cannot fully express their class characteristics. The discriminative information should increase exponentially fast in order of the labeled classes number, while the increase of the labeled samples actually in linear order. So the description ability becomes less and less, and the result cannot be much improved. As described above, using limited number of samples in each labeled class, we can only expect significant improvements for not too many classes of unlabeled data.

**Transferred between the Different Data Sets.** It is a more interesting and practical problem to transfer the information from one exiting data set to a newly collected one. We randomly choose the labeled classes from AT&T and unlabeled classes from Yale for every integer C from {2,...,10} and K from {2,...,10}. The result is shown in Table 3. We can get a similar result to transfer Yale into AT&T. Both comparison plots are shown in Fig. 4.

From these experiments, we can see that though from different data set, the face images still share some common characteristics. This is helpful knowledge to improve the learning result. It suggests that we can use existing labeled data set to handle other unlabeled classes of data, which is a novel and promising learning problem.

**Table 3.** Results for AT&T transferred to Yale, using ACC measure ($mean \pm std$)

| **AT&T**to **Yale** | PCA | LPP | DisKmeans | TDA |
|---|---|---|---|---|
| C=2,K=2 | 0.90(0.03) | 0.68(0.12) | 0.94(0.13) | **0.99(0.02)** |
| C=2,K=3 | 0.84(0.14) | 0.70(0.12) | 0.91(0.12) | **0.95(0.10)** |
| C=3,K=2 | 0.94(0.03) | 0.68(0.08) | 0.93(0.12) | **0.99(0.02)** |
| C=3,K=3 | 0.83(0.15) | 0.71(0.12) | 0.89(0.15) | **0.95(0.10)** |
| C=2,K=4 | 0.90(0.11) | 0.72(0.13) | 0.92(0.10) | **0.97(0.06)** |
| C=4,K=2 | 0.91(0.05) | 0.71(0.08) | 0.96(0.09) | **0.98(0.02)** |
| C=3,K=4 | 0.88(0.12) | 0.73(0.11) | 0.91(0.08) | **0.95(0.08)** |
| C=4,K=3 | 0.83(0.16) | 0.68(0.11) | 0.91(0.12) | **0.97(0.07)** |
| C=4,K=4 | 0.89(0.13) | 0.73(0.10) | 0.90(0.11) | **0.94(0.07)** |

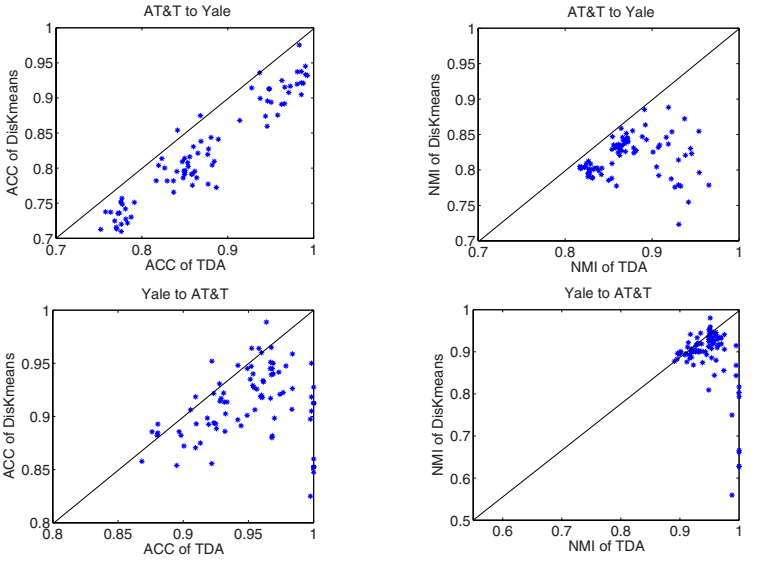



**Fig. 4.** Comparison results of TDA and DisKmeans in ACC and NMI measures, for transfers between different databases. Each point represents 50 times average for a fixed (C, K) pair.

### 4.3 Unconstrained Face Recognition

The databases in the last subsection are created under controlled conditions to facilitate the study of specific parameters on the face recognition problem, such as position, pose, lighting etc. Practically there are also many applications in which the practitioner has little or no control over such parameters. This is provided as a unconstrained face recognition problem. It is much more difficult than the constrained problems and needs novel approaches to solve.

In following experiments, we will use a recently published unconstrained data set and test the performance of our TDA algorithm.

**Unconstrained Face Data Set.** *Labeled Faces in the Wild (LFW):* This is a database of face photographs designed for studying the problem of unconstrained face recognition. The database contains more than 13,000 images of faces collected from the web. 1680 of the people pictured have two or more distinct photos in the database. More details can be found in [12]. To make the data set more balanced and comparable with the constrained data set, we only take the images of persons who have more than 10 and

(a) Original Images (b) Preprocessed Images

**Fig. 5.** LFW Face Data Examples

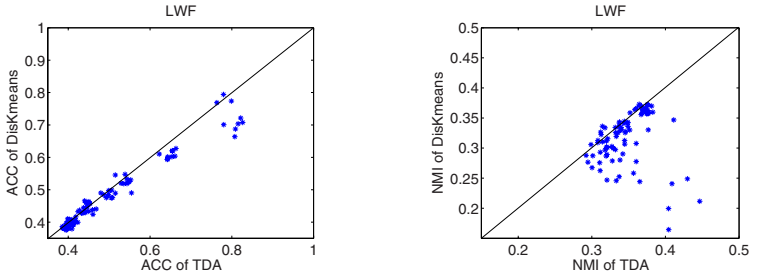



**Fig. 6.** Comparison results of TDA and DisKmeans for LWF, in ACC and NMI measures. Each point represents 50 times average for a fixed (C, K) pair.

**Table 4.** Results for AT&T transferred to LFW, using ACC measure ($mean \pm std$)

| **AT&T** to **LWF** | PCA | LPP | DisKmeans | TDA |
|---|---|---|---|---|
| C=2,K=2 | 0.72 (0.14 ) | 0.63 (0.09) | 0.73 (0.15 ) | **0.78 (0.16)** |
| C=3,K=2 | 0.71 (0.14 ) | 0.63 (0.08) | 0.71 (0.17 ) | **0.81 (0.15)** |
| C=4,K=2 | 0.72 (0.15) | 0.63 ( 0.09) | 0.72 (0.17 ) | **0.81 (0.15)** |
| C=5,K=2 | 0.69 (0.12) | 0.61(0.09) | 0.71(0.16) | **0.80(0.16)** |
| C=2,K=3 | 0.60 (0.12 ) | 0.60 (0.11 ) | 0.58 (0.09 ) | **0.61 (0.11)** |

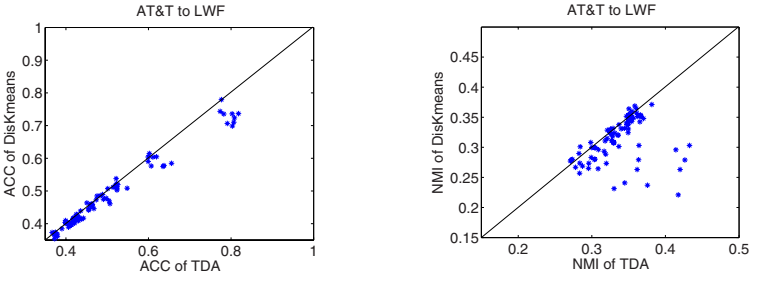



**Fig. 7.** Comparison result of TDA and DisKmeans in ACC and NMI measures, for AT&T transferred to LFW. Each point represents 50 times average for a fixed (C, K) pair.

less than 20 photos in LFW, which are 1401 images for 101 persons. Then take out the head part of the images, resize them to $56 \times 46$ pixels, and turn into gray images. The typical images are shown in Fig. 5.

**Transferred within LFW Data Set.** In this part we use the labeled data and unlabeled data all in the LFW database. We choose C from $\{2,\ldots,10\}$ and K from $\{2,\ldots,10\}$. The results are shown in Fig. 6. Though TDA outperforms other methods, in practice, we cannot always expect that the unconstrained data set is labeled. In this situation, can we use the constrained ones? If yes, it will make the transfer strategy more powerful.

**Transferred from Conventional Data Set.** In this part, we will transfer the information from exiting constrained data set to this unconstrained data set. It is a practical problem of how to deal with new complex data set based on much easier one.

We choose the labeled classes from AT&T and unlabeled classes from LFW. Use the same setting of (C,K) pairs as in the last experiment. The result is shown in Table 4 and Fig. 7.

The improvement of TDA over the unsupervised methods shows the advantage of our TDA method and gives a new approach to tackle a complex problem using the helpful information from other easier works already solved. It is to solve a difficult problem with the knowledge of more easier problems, which is similar with how human learns things.

## 5 Conclusion and Discussion

In this paper, we bring forward a problem of transferred dimensionality reduction. It uses the labeled and unlabeled data from different class domains, which is different from the traditional semi-supervised learning method. And it is more practical for nowadays drastic increase of various sorts of unlabeled information through internet. To solve this problem, we introduce the algorithm, transferred discriminative analysis. It transfers the specific discriminative information from supervised knowledge to the unlabeled samples in other class domains, and finds more suitable subspace for the lower dimensional embedding. It is fast and robust to run. The experimental results demonstrate its effectiveness and usefulness.

The TDR problem is a practical problem for nowadays computer techniques. In many cases, however, we cannot even know the class number of the data. It is a more challenging issue for our further research, which needs better clustering step of the TDA algorithm. Another interesting issue for the task-specified problems is to introduce more types of knowledge from many other source domains, which may expose the relationship of different concepts.

## Acknowledgments

This research was supported by National 863 Project ( 2006AA01Z121 ) and National Science Foundation of China ( 60675009 ).

## References

1. Belhumeur, P., Hespanha, J., Kriegman, D.: Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection. IEEE Trans. on PAMI 19(7), 711–720 (1997)
2. Baudat, G., Anouar, F.: Generalized Discriminant Analysis Using a Kernel Approach. Neural Computation 12(10), 2385–2404 (2000)
3. Cai, D., He, X., Han, J.: Semi-Supervised Discriminant Analysis. In: Proceedings of IEEE International Conference on Computer Vision (ICCV), pp. 1–7 (2007)
4. Caruana, R.: Multitask Learning. Machine Learning 28(1), 41–75 (1997)

5. Chapelle, O., Schölkopf, B., Zien, A.: Semi-Supervised Learning. MIT Press, Cambridge (2006)
6. Chung, F. (ed.): Spectral Graph Theory. CBMS Regional Conference Series in Mathematics, vol. 92. American Mathematical Society (1997)
7. Dai, W., Yang, Q., Xue, G., Yu, Y.: Boosting for Transfer Learning. In: Proceedings of International Conference on Machine Learning (ICML), pp. 193–200 (2007)
8. Dhillon, I., Guan, Y., Kulis, B.: A Unified View of Kernel K-means, Spectral Clustering and Graph Partitioning. Technical Report TR-04-25, UTCS (2005)
9. Ding, C., Li, T.: Adaptive Dimension Reduction Using Discriminant Analysis and K-means Clustering. In: Proceedings of International Conference on Machine Learning (ICML) (2007)
10. Friedman, J.: Regularized Discriminant Analysis. Journal of the American Statistical Association 84(405), 165–175 (1989)
11. Fukunaga, K.: Introduction to Statistical Pattern Recognition, 2nd edn. Academic Press (1990)
12. Huang, G.B.: Ramesh, M., Berg, T., Miller, E.: Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments. Technical Report 07-49, University of Massachusetts, Amherst (2007)
13. Georghiades, A., Belhumeur, P., Kriegman, D.: From Few to Many: Illumination Cone Models for Face Recognition under Variable Lighting and Pose. IEEE Trans on PAMI 6(23), 643–660 (2001)
14. He, X., Yan, S., Hu, Y., Niyogi, P., Zhang, H.: Face Recognition Using Laplacianfaces. IEEE Trans. on PAMI 27(3), 328–340 (2005)
15. Miller, D., Browning, J.: A Mixture Model and EM-Based Algorithm for Class Discovery, Robust Classification, and Outlier Rejection in Mixed Labeled/Unlabeled Data Sets. IEEE Trans on PAMI 25(11), 1468–1483 (2003)
16. Raina, R., Battle, A., Honglak, L., Ng, A.: Self-taught Learning: Transfer Learning from Unlabeled Data. In: Proceedings of International Conference on Machine Learning (ICML) (2007)
17. Schmidhuber, J.: On Learning How to Learn Learning Strategies. Technical Report FKI-198-94, Fakultat fur Informatik 28(1), 711–720 (1994)
18. Samaria, F., Harter, A.: Parameterisation of a stochastic model for human face identification. In: IEEE Workshop on Applications of Computer Vision, pp. 138–142 (1994)
19. Schölkopf, B., Herbrich, R., Smola, A.: A Generalized Representer Theorem. In: Helmbold, D.P., Williamson, B. (eds.) COLT 2001 and EuroCOLT 2001. LNCS (LNAI), vol. 2111, pp. 416–426. Springer, Heidelberg (2001)
20. Turk, M., Pentland, A.: Face Recognition Using Eigenfaces. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 586–591 (1991)
21. Thrun, S., Mitchell, T.: Learning One More Thing. In: IJCAI, pp. 1217–1223 (1995)
22. Thrun, S., Pratt, L.: Learning To Learn. Kluwer Academic Publishers, Boston (1998)
23. Torre, F., Kanade, T.: Discriminative cluster analysis. In: Proceedings of International Conference on Machine Learning (ICML), pp. 241–248 (2006)
24. Wu, M., Schölkopf, B.: A Local Learning Approach for Clustering. In: Proceedings of Proceedings of Neural Information Processing Systems (NIPS), pp. 1529–1536 (2007)
25. Yan, S., Xu, D., Zhang, B., Zhang, H., Yang, Q., Lin, S.: Graph Embedding and Extension: A General Framework for Dimensionality Reduction. IEEE Trans. on PAMI 29(1), 40–51 (2007)
26. Ye, J., Zhao, Z., Wu, M.: Discriminative K-Means for Clustering. In: Proceedings of Neural Information Processing Systems (NIPS), pp. 1–8 (2007)