Beyond the Graphs: Semi-parametric Semi-supervised Discriminant Analysis*

Fei Wang, Xin Wang and Tao Li
School of Computing and Information Sciences
Florida International University
feiwang@cs.fiu.edu

Abstract

Linear Discriminant Analysis (LDA) is a popular feature extraction method that has aroused considerable interests in computer vision and pattern recognition fields. The projection vectors of LDA is usually achieved by maximizing the between-class scatter and simultaneously minimizing the within-class scatter of the data set. However, in practice, there is usually a lack of sufficient labeled data, which makes the estimated projection direction inaccurate. To address the above limitations, in this paper, we propose a novel semi-supervised discriminant analysis approach. Unlike traditional graph based methods, our algorithm incorporates the geometric information revealed by both labeled and unlabeled data points in a semi-parametric way. Specifically, the final projections of the data points will contain two parts: a discriminant part learned by traditional LDA (or KDA) on the labeled points and a geometrical part learned by kernel PCA on the whole data set. Therefore we call our algorithm Semi-parametric Semi-supervised Discriminant Analysis (SSDA). Experimental results on face recognition and image retrieval tasks are presented to show the effectiveness of our method.

1. Introduction

Dimensionality reduction has been a key problem in computer vision and pattern recognition fields, since (1) the curse of high dimensionality is usually a major cause of limitations of many practical technologies; (2) the large quantities of features may degrade the performances of the classifiers when the size of the training set is small compared to the number of features [14]. In the past several decades, many dimensionality reduction methods have been proposed, in which *Principal Component Analysis* (*PCA*) and *Linear Discriminant Analysis* (*LDA*) are among the most well-known ones.

*The work is partially supported by NSF grants IIS-0546280, DMS-0844513 and CCF-0830659.

PCA [15] is an unsupervised dimensionality reduction method which aims at extracting a linear subspace in which the variance of the projected data is maximized (or, equivalently, the reconstruction error is minimized). If the data set is indeed embedded in a linear subspace, then *PCA* is guaranteed to discover such a subspace and produces a compact representation. Otherwise, we can apply some other techniques (*e.g.*,the kernel trick [19]) to generalize traditional *PCA* to the nonlinear case.

LDA [10] is a supervised dimensionality reduction technique which aims to find a projection subspace on which the data from the same class will be pushed close while the data from different classes will be pulled far away. When sufficient label information are available, *e.g.* for the fully supervised classification task, *LDA* is shown to be capable of achieving significant better performances than *PCA* [1]. However, in practice, there is usually a lack of sufficient labeled data, which makes the estimated projection direction inaccurate and thus degrades the final performance of *LDA*.

To solve the problem, some semi-supervised learning methods, which aims to to learn from both labeled and unlabeled data, have been proposed [6], among which *graph based approaches* [23] are one of the most active research areas. The main theme behind those techniques is to model the whole data set (including both labeled and unlabeled data points) as a graph and then use such geometric information as a prior to guide the final decision process [2]. Recently, *Cai et al.* [5] incorporated such idea into discriminant analysis and proposed *Semi-supervised Discriminant Analysis* (*SDA*), which finds a projection respecting the discriminant structure inferred from the labeled data, as well as the intrinsic geometric structure from all the data points.

In this paper, we propose a novel *Semi-parametric Semi-supervised Discriminant Analysis* (*SSDA*) technique. Unlike *SDA*, our method exploits the geometric information of the data set in a different way. In *SSDA*, we first formulate discriminant analysis as a regression problem, in which the regression function is a *semi-parametric* one that can be decomposed into two parts: the non-parametric part is obtained via performing traditional discriminant analysis al-

978-1-4244-3991-1/09/$25.00 ©2009 IEEE

gorithms (*i.e.*, *LDA* or *Kernel Discriminant Analysis* (*KDA*) [17]) on the labeled data, and the parametric part is achieved by performing *Kernel PCA* on both labeled and unlabeled data points (which will be made clear in the later sections). As a result, *SSDA* is able to learn the data projections by integrating the power of *LDA* and *KPCA*. From another point of view, we use the geometrical information learned from *KPCA* to "correct" the projection directions learned only on the labeled data by *LDA* to make it more "accurate".

It is worthwhile to highlight several aspects of the proposed approach here:

1. The proposed *SSDA* approach is based on semi-parametric regression, unlike traditional *LDA* which is based on generalized eigenvalue decomposition.

2. We derive a new way to exploit the geometrical information contained in the data set, which is different from traditional graph based approaches.

3. *SSDA* is computationally efficient and achieves good empirical results in our experiments.

4. The core idea of *SSDA* can be easily generalized to other classification and regression problems.

The rest of this paper is organized as follows: In Section 2 we will briefly review some works that is closely related to this paper. The detailed algorithm of *SSDA* will be introduced in Section 3. Section 4 will show the experimental results on applying *SSDA* to the problem of face recognition from a single training image and image retrieval, followed by the conclusions and discussions in Section 5.

2. Related Works

In this section we briefly review some previous works that are closely related to this paper. Generally, given a labeled data set $\mathcal{X} = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_l\}$ coming from C different classes, *LDA* finds the optimal projection direction by [10]

$$\mathbf{w}^* = \arg\max_{\mathbf{w}} \frac{\mathbf{w}^T \mathbf{S}_b \mathbf{w}}{\mathbf{w}^T \mathbf{S}_w \mathbf{w}} \tag{1}$$

where

$$\mathbf{S}_b = \sum_{c=1}^{C} n_c (\mathbf{m}_c - \mathbf{m})(\mathbf{m}_c - \mathbf{m})^T \tag{2}$$

$$\mathbf{S}_w = \sum_{c=1}^{C} \left(\sum_{\mathbf{x}_i \in \pi_c} (\mathbf{x}_i - \mathbf{m}_c)(\mathbf{x}_i - \mathbf{m}_c)^T \right) \tag{3}$$

are the between-class and within-class scatter matrices respectively. $\mathbf{m}_c$ is the mean of the c-th vector, $\mathbf{m}$ is the mean of the whole data set, π_c denotes the c-th class with size n_c.

If we define the total scatter matrix of $\mathcal{X}$ as

$$\mathbf{S}_t = \sum_{i=1}^{l} (\mathbf{x}_i - \mathbf{m})(\mathbf{x}_i - \mathbf{m})^T = \mathbf{S}_b + \mathbf{S}_w, \tag{4}$$

then the objective of *LDA* is equivalent to

$$\mathbf{w}^* = \arg\max_{\mathbf{w}} \frac{\mathbf{w}^T \mathbf{S}_b \mathbf{w}}{\mathbf{w}^T \mathbf{S}_t \mathbf{w}}. \tag{5}$$

The optimal $\mathbf{w}$'s can be obtained by the following generalized eigenvalue-decomposition problem

$$\mathbf{S}_b \mathbf{w} = \lambda \mathbf{S}_t \mathbf{w}$$

If we first subtract the mean from the data set, *i.e.*, centralize the data, then the between-class scatter matrix becomes

$$\mathbf{S}_b = \sum_{c=1}^{C} n_c \mathbf{m}_c \mathbf{m}_c^T = \sum_{c=1}^{C} \bar{\mathbf{X}}^{(c)} \mathbf{G}^{(c)} (\bar{\mathbf{X}}^{(c)})^T \tag{6}$$

where $\bar{\mathbf{X}}^{(c)} = [\bar{\mathbf{x}}_1^{(c)}, \bar{\mathbf{x}}_2^{(c)}, \cdots, \bar{\mathbf{x}}_{n_c}^{(c)}]$ denotes the centralized data matrix of the c-th class, and $\mathbf{G}^{(c)}$ is an $n_c \times n_c$ constant matrix with all its elements equal to $1/n_c$.

If we rearrange the data order and define the centralized data matrix as $\bar{\mathbf{X}} = [\bar{\mathbf{X}}^{(1)}, \bar{\mathbf{X}}^{(2)}, \cdots, \bar{\mathbf{X}}^{(C)}]$, and define a block-diagonal matrix $\mathbf{G} \in \mathbb{R}^{l \times l}$ as

$$\mathbf{G} = \begin{bmatrix} \mathbf{G}^{(1)} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{G}^{(2)} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{G}^{(C)} \end{bmatrix} \tag{7}$$

where we use $\mathbf{0}$ to denotes the zero matrix with appropriate size. Thus we have $\mathbf{S}_b = \bar{\mathbf{X}} \mathbf{G} \bar{\mathbf{X}}^T$, and $\mathbf{S}_t = \bar{\mathbf{X}} \bar{\mathbf{X}}^T$. Then the objective function in Eq.(1) can be rewritten as

$$\mathbf{w}^* = \arg\max_{\mathbf{w}} \frac{\mathbf{w}^T \bar{\mathbf{X}} \mathbf{G} \bar{\mathbf{X}}^T \mathbf{w}}{\mathbf{w}^T \bar{\mathbf{X}} \bar{\mathbf{X}}^T \mathbf{w}} \tag{8}$$

Recently, *Cai et al.* [5] pointed out that when there is no sufficient labeled training samples (which is a common case in practice [6]), the data covariance matrices of each class may not be accurately estimated, which will make the final projection direction $\mathbf{w}$ inaccurately estimated. To solve such a problem, they proposed to first construct a data dependent regularizer

$$J(\mathbf{w}) = \mathbf{w}^T \bar{\mathbf{X}} \mathbf{L} \bar{\mathbf{X}} \mathbf{w} \tag{9}$$

and solve for the optimal $\mathbf{w}$ by

$$\mathbf{w}^* = \arg\max_{\mathbf{w}} \frac{\mathbf{w}^T \bar{\mathbf{X}} \mathbf{G} \bar{\mathbf{X}}^T \mathbf{w}}{\mathbf{w}^T \bar{\mathbf{X}} \bar{\mathbf{X}}^T \mathbf{w} + J(\mathbf{w})} \tag{10}$$

where $\mathbf{L} = \mathbf{D} - \mathbf{S} \in \mathbb{R}^{l \times l}$ is the *combinatorial graph Laplacian* [9], $\mathbf{S} \in \mathbb{R}^{l \times l}$ is the data similarity matrix with its (i, j)-th entry representing the similarity between $\mathbf{x}_i$ and $\mathbf{x}_j$, $\mathbf{D} \in \mathbb{R}^{l \times l}$ is a diagonal data degree matrix with the i-th

Table 1. Spectral Regression LDA

Inputs: Training data set $\mathcal{X}$ from C classes
Outputs: $C-1$ projection directions $\mathbf{w}_1, \cdots, \mathbf{w}_{c-1}$
Procedure:
1. Centralize $\mathcal{X}$
2. Generate the $C-1$ response vectors $\{\bar{\mathbf{y}}^i\}_{i=1}^{C-1}$ by the *Gram-Schmidt* method as introduced in [4]
3. Solving the regularized least square problem (13).

element on its diagonal line $D(i,i) = \sum_j S_{ij}$. To see its physical meaning, we can further expand $J(\mathbf{w})$ by

$$J(\mathbf{w}) = \sum\nolimits_{i=1}^{l} S_{ij}(\mathbf{w}^T\bar{\mathbf{x}}_i - \mathbf{w}^T\bar{\mathbf{x}}_j)^2 \qquad (11)$$

which measures the smoothness of the data set after projection. Therefore, the criterion shown in Eq.(10) just aims to learn a discriminative function which is as smooth as possible on the data manifold.

3. Semi-parametric Semi-supervised Discriminant Analysis (*SSDA*)

In this section, we introduce our *Semi-parametric Semi-supervised Discriminant Analysis* (*SSDA*) algorithm in detail. Since our method is based on the regression interpretation of the traditional *LDA* algorithm, we first introduce how to formulate *LDA* as a regression problem.

3.1. Spectral Regression LDA

Cai et al. [4] recently proposed a unified *spectral regression* framework for subspace learning. We can also adapt *LDA* into such framework by two steps: (1) solve the eigen-problem $\mathbf{G}\bar{\mathbf{y}} = \lambda\bar{\mathbf{y}}$ to get $\bar{\mathbf{y}}$; (2) find $\mathbf{w}$ by solving $\bar{\mathbf{X}}^T\mathbf{w} = \bar{\mathbf{y}}$. Since in practice such a $\mathbf{w}$ may not exist, we can solve the following regression problem to obtain an approximate $\mathbf{w}$

$$\mathbf{w}^* = \arg\min_{\mathbf{w}} \sum\nolimits_{i=1}^{l} (\mathbf{w}^T\bar{\mathbf{x}}_i - \bar{y}_i)^2 \qquad (12)$$

In situations where the number of samples is smaller than the number of features (in which case we can get infinite number solutions for $\bar{\mathbf{X}}^T\mathbf{w} = \bar{\mathbf{y}}$), we can solve the following regularized form of Eq.(12) to obtain the optimal $\mathbf{w}$

$$\mathbf{w}^* = \arg\min_{\mathbf{w}} \left(\sum\nolimits_{i=1}^{l} (\mathbf{w}^T\bar{\mathbf{x}}_i - \bar{y}_i)^2 + \delta\|\mathbf{w}\|^2 \right) \qquad (13)$$

Following the parlance in [4], we call this algorithm *Spectral Regression LDA* (*SRLDA*), and the main procedure of which is summarized in Table 1.

3.2. SSDA: The Algorithm

One major problem for *SRLDA* is that when there is no sufficient labeled data, $\mathbf{w}$ may be inaccurately estimated. However, in practice, the unlabeled data are usually much easier to obtain. Therefore it is natural to incorporate those unlabeled points into the estimation of $\mathbf{w}$. One most straightforward way is to follow the idea in [2], *i.e.*, we construct a graph based regularizer as in Eq.(9) using the whole data set (both labeled and unlabeled data) and then solve the following optimization problem.

$$\mathbf{w}^* = \arg\min_{\mathbf{w}} \left(\sum_{i=1}^{l} (\mathbf{w}^T\bar{\mathbf{x}}_i - \bar{y}_i)^2 + \delta\|\mathbf{w}\|^2 + \rho\mathbf{w}^T\bar{\mathbf{X}}\mathbf{L}\bar{\mathbf{X}}\mathbf{w} \right)$$

In such a way, the learned $\mathbf{w}$ will not only contain discriminative information, but also the geometrical information contained in the data set.

In the following, we introduce a novel method, called *Semi-parametric Semi-supervised Discriminant Analysis* (*SSDA*), to realize semi-supervised *LDA*. First we review the basics of semi-parametric regression [18].

3.2.1 Semi-parametric Regression

The standard definition of *semi-parametric regression* is

Definition 1 (Semi-parametric Regression). *Semi-parametric regression refers to the regression models in which the predictor contains both parametric and non-parametric components.*

For example, suppose we want to construct a *predictor* $\tilde{f}$ from n input-output pairs $\{(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_n, y_n)\}$ by

$$\tilde{f}^* = \arg\min_{\tilde{f}} \sum_{i=1}^{n} \mathcal{L}(y_i, \tilde{f}(\mathbf{x}_i)) \qquad (14)$$

where $\mathcal{L}(\cdot,\cdot)$ is some loss function. Then for *parametric regression*, $\tilde{f}(\mathbf{x})$ can be written as an explicit function of $\mathbf{x}$ which is dependent on some parameters $\mathbf{w}$ (*e.g.*, linear regression); for *non-parametric regression*, $\tilde{f}(\mathbf{x})$ cannot be estimated via an explicit parametric function, *i.e.*, it can only be estimated from the data (*e.g.*, k-nearest neighbor classifier); for *semi-parametric regression*, $\tilde{f}$ can be decomposed into two parts as $\tilde{f} = f + h$, where f is a non-parametric predictor which can be estimated from the data set, $h \in span\{\psi_p\}$ is a parametric estimator, with $\{\psi_p\}$ a family of parametric functions.

The semi-parametric model can be useful in many cases [19], for example, if one has some additional knowledge that the major properties of the data set are described by a small set of independent basis functions $\{\psi_1(\cdot), \psi_2(\cdot), \cdots, \psi_m(\cdot)\}$, or one may want to correct the data from some (*e.g.* linear) trends. From another point of view, the semi-parametric way can also make the model more *understandable* without sacrificing the accuracy (the non-parametric component usually makes the model accurate since it is estimated from the data, while the paramet-

ric component can make the model more easily understood since it can be written in an explicit form).

Since discriminant analysis can be formulated as a regression problem as we have introduced in Section 3.1, we can also construct a semi-parametric model for discriminant analysis. Specifically, the semi-parametric model consists of two parts: one part (the non-parametric part) is learned from the labeled data which contains some discriminant information, the other part (the parametric part) is learned from both labeled and unlabeled data to incorporate the geometrical information contained in the data set. In other words, in the case when there is no sufficient labeled data, then the discriminative information contained in the data set (the non-parametric part) may not be accurately estimated, therefore we apply a parametric estimator to "correct" the original predictor to make it more accurate.

Before we go into the details of our *SSDA* algorithm, first we introduce two preliminary theorems [19].

Theorem 1 (Semiparametric Representor Theorem). *Suppose we are given a nonempty set $\mathcal{X}$, a positive definite real-valued kernel k on $\mathcal{X} \times \mathcal{X}$, a training set $(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \cdots, (\mathbf{x}_l, y_l) \in \mathcal{X} \times \mathbb{R}$, a strictly monotonically increasing real-valued function Ω on $[0, \infty]$, an arbitrary cost function $\mathcal{L} : (\mathcal{X} \times \mathbb{R}^2)^l \to \mathbb{R} \bigcup \{\infty\}$, and a class of functions*

$$\mathcal{F} = \left\{ f \in \mathbb{R}^{\mathcal{X}} \,\middle|\, f(\cdot) = \sum\nolimits_{i=1}^{\infty} \gamma_i k(\cdot, \mathbf{z}_i), \|f\|_k < \infty \right\}$$

where $\gamma_i \in \mathbb{R}, \mathbf{z}_i \in \mathcal{X}$, and $\|\cdot\|$ is the norm in the Reproducing Kernel Hilbert Space (RKHS) $\mathcal{H}_k$ associated with k, i.e., for any $\mathbf{z}_i \in \mathcal{X}, \gamma_i \in \mathbb{R}$,

$$\left\| \sum\nolimits_{i=1}^{\infty} \gamma_i k(\cdot, \mathbf{z}_i) \right\|^2 = \sum\nolimits_{i,j=1}^{\infty} \gamma_i \gamma_j k(\mathbf{z}_i, \mathbf{z}_j).$$

Moreover, we are also given a set of m real-valued functions $\{\psi_p\}_{p=1}^m$ on $\mathcal{X}$, with the property that the $l \times m$ matrix $(\psi_p(\mathbf{x}_i))_{ip}$ has rank m. Then any $\tilde{f} = f + h$, with $f \in \mathcal{F}$ and $h \in span\{\psi_p\}$, minimizing the regularized risk

$$c((\mathbf{x}_1, y_1, \tilde{f}(\mathbf{x}_1)), \cdots, (\mathbf{x}_l, y_l, \tilde{f}(\mathbf{x}_l))) + \Omega(\|f\|_k)$$

admits a representation of the form

$$\tilde{f}(\cdot) = \sum\nolimits_{i=1}^{l} \alpha_i k(\mathbf{x}_i, \cdot) + \sum\nolimits_{p=1}^{m} \gamma_p \psi_p(\cdot) \qquad (15)$$

with $\alpha_i, \gamma_p \in \mathbb{R}$ for all $p = 1, 2, \cdots, m$.

In theorem 1, the parametric functions $\{\psi_p\}_{p=1}^m$ can be any functions, *e.g.*, in standard *SVM*, $m = 1$ and $\psi_1(\mathbf{x}) = 1$ [19]. Therefore, in the semi-parametric setting, the function $\tilde{f}^*$ minimizing the following structural loss

$$\mathcal{J} = \sum\nolimits_{i=1}^{l} \mathcal{L}(\mathbf{x}_i, y_i, \tilde{f}(\mathbf{x}_i)) + \|f\|_k^2 \qquad (16)$$

would have the form of Eq.(15). Note that in the above loss, the parametric functions $\{\psi_p\}_{p=1}^m$ do not contribute to the regularization term $\|f\|_k^2$. [19] pointed out that this needs not to be a major concern when m is sufficiently small than l. Returning to the *SRLDA* problem Eq.(13), we can easily find that it is just a spacial case of Eq.(16) with $\mathcal{L}$ being the square loss and $\tilde{f}$ begin a linear function without parametric components.

3.2.2 Incorporating the Geometrical Information

Now the only problem remained is how to fit semi-supervised regression into the semi-supervised setting, *i.e.*, how to incorporate the geometrical information contained in the data set into the learning process. A natural choice would be to construct some proper parametric functions which carry those geometrical information.

More concretely, in our *SSDA* algorithm, we use only one parametric function, *i.e.*, $m = 1$, which is learned from both labeled and unlabeled data via *Kernel PCA* (*KPCA*) [19]. The goal of *KPCA* is to find the principal axes which carry the largest variance in the features. Mathematically, let $\{\Phi(\mathbf{x}_1), \Phi(\mathbf{x}_2), \cdots, \Phi(\mathbf{x}_n)\}$ be the data set in the feature space, which has been centralized. Then *KPCA* aims to find the principal axis by solving $\mathbf{C}\mathbf{v} = \lambda\mathbf{v}$, where $\mathbf{C} = \sum_{i=1}^n \Phi(\mathbf{x}_i)\Phi(\mathbf{x}_i)^T/n$ is the data covariance matrix in the feature space. Since the eigenvector $\mathbf{v}$ lies in the space spanned by $\{\Phi(\mathbf{x}_1), \Phi(\mathbf{x}_2), \cdots, \Phi(\mathbf{x}_n)\}$, then $\mathbf{v}$ can be expanded by $\mathbf{v} = \sum_{i=1}^n \mu_i \Phi(\mathbf{x}_i)$, where $\mu_i \in \mathbb{R}$ are the expansion coefficients. Combining all things together, we can derive that $n\lambda\boldsymbol{\mu} = \mathbf{K}\boldsymbol{\mu}$, where $\boldsymbol{\mu} = [\mu_1, \mu_2, \cdots, \mu_n]^T$. Then we have the following theorem [7].

Theorem 2. *The continuous solutions for the discrete cluster membership indicator vectors in Kernel K-means clustering for $\mathcal{X}$ are just the eigenvectors of $\mathbf{K}$ corresponding to its largest $C - 1$ eigenvalues.*

Theorem 2 tells us that the eigenvectors of $\mathbf{K}$ in fact carry some distribution information contained in the data set. For a new test point $\Phi(\mathbf{x})$, its projection on the c-th kernel principal component $\mathbf{v}^c$ is just

$$P^c(\mathbf{x}) = \langle \Phi(\mathbf{x}), \mathbf{v}^c \rangle = \sum\nolimits_{i=1}^{n} k(\mathbf{x}, \mathbf{x}_i)\mu_i^c \qquad (17)$$

where μ_i^c denotes the i-th component of $\boldsymbol{\mu}^c$. According to theorem 2 and the previous analysis, what $P^c(\mathbf{x})$ measures is just the similarity between $\mathbf{x}$ and the c-th cluster.

Therefore, returning back to our *SSDA* algorithm, we can set the parametric function for f

$$\psi(\mathbf{x}) = P(\mathbf{x}) \qquad (18)$$

which incorporates the geometrical information contained

in the data set[1]. In this way, combining Eq.(15) and Eq.(18), we can rewrite the regression function as

$$\tilde{f}(\mathbf{x}) = \sum_{i=1}^{l} \alpha_i k(\mathbf{x}_i, \mathbf{x}) + \gamma\psi(\mathbf{x}) \tag{19}$$

Note that the first term of $\tilde{f}$ only relates to the labeled data points, while the second term of f relates to both labeled and unlabeled data points. In the following subsection we will derive a concrete algorithm on how to get an optimal $\tilde{f}$.

3.2.3 A Concrete Algorithm

In this section, we introduce a concrete *SSDA* algorithm, where the regression loss takes a quadratic form as in *SRLDA*. Then the optimal $\tilde{f}$ can be obtained by minimizing

$$\mathcal{J} = \sum_{i=1}^{l} (\bar{y}_i - \tilde{f}(\mathbf{x}_i))^2 + \delta\|f\|_k^2 \tag{20}$$

where $\bar{y}_i$ is the real label of the i-th data point, and δ is the regularization parameter. Combining Eq.(20) and Eq.(19), we can get that

$$\mathcal{J} = \sum_{i=1}^{l} \left(\bar{y}_i - \sum_{j=1}^{l} \alpha_j k(\mathbf{x}_j, \mathbf{x}_i) - \gamma\psi(\mathbf{x}_i) \right)^2 + \delta\|f\|_k^2,$$

Written in its matrix form, $\mathcal{J}$ is equal to

$$\mathcal{J} = (\bar{\mathbf{y}} - \mathbf{K}^l\boldsymbol{\alpha} - \gamma\boldsymbol{\psi})^T(\bar{\mathbf{y}} - \mathbf{K}^l\boldsymbol{\alpha} - \gamma\boldsymbol{\psi}) + \delta\boldsymbol{\alpha}^T\mathbf{K}^l\boldsymbol{\alpha} \tag{21}$$

where $\mathbf{K}^l$ is the $l \times l$ kernel matrix constructed by the labeled points, $\boldsymbol{\psi} = [\psi(\mathbf{x}_1), \cdots, \psi(\mathbf{x}_l)]^T$. For multi-class problems, we need to solve $C-1$ projection functions as in *SRLDA*. Therefore we can solve them by minimizing the following criterion.

$$\begin{aligned} \mathcal{J} &= tr\left((\bar{\mathbf{Y}} - \mathbf{K}^l\mathbf{A} - \boldsymbol{\Psi}\boldsymbol{\Gamma})^T(\bar{\mathbf{Y}} - \mathbf{K}^l\mathbf{A} - \boldsymbol{\Psi}\boldsymbol{\Gamma})\right) \\ &\quad + \delta tr\left(\mathbf{A}^T\mathbf{K}^l\mathbf{A}\right) \end{aligned} \tag{22}$$

where $\bar{\mathbf{Y}} = [\bar{\mathbf{y}}^1, \cdots, \bar{\mathbf{y}}^{C-1}]$, and $\bar{\mathbf{y}}^c$ can be obtained by the same manner as in *SRLDA* in table 1. $\mathbf{A} = [\boldsymbol{\alpha}^1, \cdots, \boldsymbol{\alpha}^{C-1}]$ with $\boldsymbol{\alpha}^c$ being the expansion coefficients for $\tilde{f}^c$. $\boldsymbol{\Psi} = [\boldsymbol{\psi}^1, \cdots, \boldsymbol{\psi}^{C-1}]$ with its (i,j)-th element $\Psi_{ij} = \psi_i^j = P^j(\mathbf{x}_i)$, where $P^j(\mathbf{x}_i)$ is defined in Eq.(17). $\boldsymbol{\Gamma}$ is a diagonal matrix with the i-th element on its diagonal line $\Gamma_{ii} = \gamma_i$. Since the derivatives of $\mathcal{J}$ with respect to $\mathbf{A}$ and γ

$$\frac{\partial\mathcal{J}}{\partial\mathbf{A}} = 2\left(\mathbf{K}^l\mathbf{K}^l\mathbf{A} + \mathbf{K}^l\boldsymbol{\Psi}\boldsymbol{\Gamma} - \mathbf{K}^l\bar{\mathbf{Y}} + \delta\mathbf{K}^l\mathbf{A}\right) \tag{23}$$

$$\frac{\partial\mathcal{J}}{\partial\boldsymbol{\Gamma}} = 2\left(\boldsymbol{\Psi}^T\mathbf{K}^l\mathbf{A} + \boldsymbol{\Psi}^T\boldsymbol{\Psi}\boldsymbol{\Gamma} - \boldsymbol{\Psi}^T\bar{\mathbf{Y}}\right) \tag{24}$$

[1]In the multi-class discriminant analysis, we will pursue $C-1$ projection directions, which means that we will regress $C-1$ projection functions $\tilde{f}^1, \tilde{f}^2, \cdots, \tilde{f}^{C-1}$, such that $f^c(\mathbf{x}_i)$ returns the embedding of $\mathbf{x}_i$ on the c-th direction. The parametric part of $\tilde{f}^c$ is P^c.

Table 2. Semi-parametric Semi-supervised Discriminant Analysis

Inputs:
Data set $\mathcal{X} = \mathcal{X}_\mathcal{L} \bigcup \mathcal{X}_\mathcal{U}$, $\mathcal{X}_\mathcal{L}$ is the labeled set, $\mathcal{X}_\mathcal{U}$ is the unlabeled set; Kernel parameters.
Outputs:
Estimated function $\{\tilde{f}^c\}_{c=1}^{C-1}$
Procedure:
1. Choose the kernel $\mathbf{K}$ and do *KPCA* on the whole data set $\mathcal{X}$, get $\psi^c(\cdot) = P^c(\cdot)$ as in Eq.(17).
2. Choose the kernel $\mathbf{K}^l$ on $\mathcal{X}_\mathcal{L}$, and solve the linear equation system Eq.(23) and Eq.(24).
3. Output $\tilde{f}^c(\mathbf{x}) = \sum_{i=1}^{l} \alpha_i^c k(\mathbf{x}_i, \mathbf{x}) + \gamma\psi^c(\mathbf{x})$.

will vanish at its minimum, thus we can solve the optimal $(\mathbf{A}, \boldsymbol{\Gamma})$ by solving the linear equation system $\partial\mathcal{J}/\partial\mathbf{A} = 0, \partial\mathcal{J}/\partial\boldsymbol{\Gamma} = 0$, from which we can derive that

$$\begin{aligned} \boldsymbol{\alpha}^{c*} &= \left(\delta\mathbf{I} - \frac{\boldsymbol{\psi}^c(\boldsymbol{\psi}^c)^T\mathbf{K}^l}{\boldsymbol{\psi}^c(\boldsymbol{\psi}^c)^T} + \mathbf{K}^l\right)^{-1}\left(\mathbf{I} - \frac{\boldsymbol{\psi}^c(\boldsymbol{\psi}^c)^T}{(\boldsymbol{\psi}^c)^T\boldsymbol{\psi}^c}\right)\bar{\mathbf{y}}^c \\ \gamma^{c*} &= \frac{(\boldsymbol{\psi}^c)^T\bar{\mathbf{y}}^c - (\boldsymbol{\psi}^c)^T\mathbf{K}^l\boldsymbol{\alpha}^{c*}}{(\boldsymbol{\psi}^c)^T\boldsymbol{\psi}^c} \end{aligned}$$

The main procedure of *SSDA* is summarized in Table 2.

The Linear Case: We can also linearize *SSDA* to make it more efficient in practice. Specifically, *linear SSDA* aims to solve the $C-1$ projection functions in the following form

$$\tilde{f}^c(\mathbf{x}) = (\mathbf{w}^c)^T\mathbf{x} + \gamma\psi^c(\mathbf{x}), \tag{25}$$

where $\mathbf{w}^c$ is the c-th projection direction, ψ^c is the parametric part of $\tilde{f}^c$ which has the same meaning as in Eq.(18).

Similar to *SSDA*, we can solve for the optimal $\mathbf{w}^c$ by minimizing

$$\begin{aligned} \bar{\mathcal{J}} &= tr\left((\bar{\mathbf{Y}} - \mathbf{X}_L^T\mathbf{W} - \boldsymbol{\Psi}\boldsymbol{\Gamma})^T(\bar{\mathbf{Y}} - \mathbf{X}_L^T\mathbf{W} - \boldsymbol{\Psi}\boldsymbol{\Gamma})\right) \\ &\quad + \delta tr\left(\mathbf{W}^T\mathbf{W}\right), \end{aligned} \tag{26}$$

where $\mathbf{W} = [\mathbf{w}^1, \cdots, \mathbf{w}^{C-1}]$, $\mathbf{X}_L \in \mathbb{R}^{d\times l}$ is composed of all the labeled data points, $\bar{\mathbf{Y}}, \boldsymbol{\Psi}, \boldsymbol{\Gamma}$ have the same meaning as in Eq.(22). Then we have

$$\frac{\partial\bar{\mathcal{J}}}{\partial\mathbf{W}} = 2(\mathbf{X}_L\mathbf{X}_L^T\mathbf{W} + \mathbf{X}_L\boldsymbol{\Psi}\boldsymbol{\Gamma} - \mathbf{X}_L\bar{\mathbf{Y}} + \delta\mathbf{W}) \tag{27}$$

$$\frac{\partial\bar{\mathcal{J}}}{\partial\boldsymbol{\Gamma}} = 2(\boldsymbol{\Psi}^T\mathbf{X}_L^T\mathbf{W} + \boldsymbol{\Psi}^T\boldsymbol{\Psi}\boldsymbol{\Gamma} - \boldsymbol{\Psi}^T\bar{\mathbf{Y}}) \tag{28}$$

Table 3. Linear SSDA

Inputs:
Data set $\mathcal{X} = \mathcal{X}_{\mathcal{L}} \bigcup \mathcal{X}_{\mathcal{U}}$, $\mathcal{X}_{\mathcal{L}}$ is the labeled set, $\mathcal{X}_{\mathcal{U}}$ is the unlabeled set; Kernel parameters.
Outputs:
Estimated function $\{\tilde{f}^c\}_{c=1}^{C-1}$
Procedure:
1. Choose the kernel $\mathbf{K}$ and do *KPCA* on the whole data set $\mathcal{X}$, get $\psi^c(\cdot) = P^c(\cdot)$ as in Eq.(17).
2. Solve the linear equation system Eq.(27) and Eq.(28).
3. Output $\tilde{f}^c(\mathbf{x}) = (\mathbf{w}^c)^T\mathbf{x} + \gamma\psi^c(\mathbf{x})$.

By setting $\partial\bar{\mathcal{J}}/\partial\mathbf{W} = 0,\ \partial\bar{\mathcal{J}}/\partial\mathbf{\Gamma} = 0$, we can get that[2]

$$\begin{aligned}\mathbf{w}^{c*} &= \left(\mathbf{X}_L\mathbf{X}_L^T + \delta\mathbf{I} - \frac{\mathbf{X}_L\boldsymbol{\psi}^c(\boldsymbol{\psi}^c)^T\mathbf{X}_L^T}{(\boldsymbol{\psi}^c)^T\boldsymbol{\psi}^c}\right)^{-1} \\ &\quad \cdot\left(\mathbf{X}_L - \frac{\mathbf{X}_L\boldsymbol{\psi}^c(\boldsymbol{\psi}^c)^T\bar{\mathbf{y}}^c}{(\boldsymbol{\psi}^c)^T\boldsymbol{\psi}^c}\right) \qquad (29)\\ \gamma^{c*} &= \frac{(\boldsymbol{\psi}^c)^T\bar{\mathbf{y}}^c - (\boldsymbol{\psi}^c)^T\mathbf{X}_L^T\mathbf{w}^{c*}}{(\boldsymbol{\psi}^c)^T\boldsymbol{\psi}^c} \qquad (30)\end{aligned}$$

The main procedure of *Linear Semi-parametric Semi-supervised Discriminant Analysis* (*LSSDA*) is summarized in Table 3.

3.3. Complexity Analysis

In the last subsection we have introduced our *SSDA* algorithm and its linear counterpart in detail. From the algorithm flowchart in Table 2 we can see that the main computational cost of *SSDA* lies in two parts: (1) perform *KPCA* on the whole data set $\mathcal{X}$; (2) solve the linear equation system. Conventionally, the first part will cost $O(n^3)$ time, where $n = l + u$ is the total number of data points, and the second part will cost $O(l^3)$ time, where l is the number of labeled points. In practice, we can apply some sophisticated methods to make our algorithm efficient, *e.g.*, the *Nyström* [8] or *block quantized* [25] approximation for *KPCA*, and the *iterative* methods for solving linear equation systems [12].

3.4. Relationship with Manifold Regularization

Semi-supervised learning has aroused considerable interests in machine learning and computer vision fields in recent years [6], among which graph based semi-supervised learning is one of the most active research topic. *Belkin et al.* [2] proposed a *manifold regularization* framework for semi-supervised learning, which explores the geometric structure of the marginal distribution of the data set. Specifically, for a classification function f, they considered the following regularization term[3]

$$\|f\|_{\mathcal{I}}^2 = \mathbf{f}^T\mathbf{L}\mathbf{f} = \sum\nolimits_{ij}(f_i - f_j)^2 S_{ij}$$

where S_{ij} is the similarity between $\mathbf{x}_i$ and $\mathbf{x}_j$, $f_i = f(\mathbf{x}_i)$, $\mathbf{f} = [f_1, \cdots, f_n]^T$, and $\mathbf{L}$ is the *combinatorial graph Laplacian* as we have defined in Eq.(9). By incorporating $\|f\|_{\mathcal{I}}^2$, the manifold regularization method aims to obtain an optimal f by

$$\mathbf{f}^* = \arg\min_f \sum_{i=1}^{l}\mathcal{L}(\mathbf{x}_i, y_i, f_i) + \gamma_A\|f\|_k^2 + \gamma_I\|f\|_{\mathcal{I}}^2 \quad (31)$$

Based on this criterion, f should satisfy the following two properties: (1) *Label Consistency*: The predictions of f on labeled points will be sufficiently close to their initial labels; (2) The predictions of f on the whole data set should be sufficiently smooth with respect to the intrinsic data graph.

Although the *label smoothness* property seems intuitive and straightforward, it cannot fit all the cases. For example, it is suitable for the data from the same class, while unsuitable for the data in different classes. However, in our *SSDA* method, we incorporate the geometrical information of the data set by *KPCA*, which carries the underlying cluster information of the data set. Thus we may expect that the resulting f should have more discriminative power.

In another paper *Sindhwani et al.* [21] pointed out that what manifold regularization does is to warp the original functional space (which is a *Reproducing Kernel Hilbert Space* (*RKHS*)) towards the data distributions and define a different data dependent function norm. For our *SSDA* method, we warp the *RKHS* in a different way. We extend the original *RKHS* by incorporating a parametric part such that the learned function can reflect the data distributions without changing its norm.

4. Experiments

In this section, we investigate the performance of our proposed *SSDA* methods on face recognition from a single training image [3], and image retrieval tasks.

4.1. Face Recognition from a Single Training Image

The databases we used in this part of experiments are

[2]Note that using Eq.(29) to get the optimal $\mathbf{w}^c$ needs to compute the inverse of a $d \times d$ matrix. It would be very time consuming when $d \gg l$. A trick is to apply the *Woodbury formula* [12] such that $\mathbf{w}^c = \left(\mathbf{I} - \mathbf{X}_L(\delta\mathbf{I} + \mathbf{B}\mathbf{X}_L^T\mathbf{X}_L)^{-1}\mathbf{B}\mathbf{X}_L^T\right)/\delta$, where $\mathbf{B} = \mathbf{I} - (\boldsymbol{\psi}^c(\boldsymbol{\psi}^c)^T/(\boldsymbol{\psi}^c)^T\boldsymbol{\psi}^c)$. In this way we only need to compute a matrix of size $l \times l$.

[3]Note that when we relax the constraint that the return values of f should be discrete, then the classification problem can also be regarded as a regression problem as discriminant analysis. The only difference is that in classification, the desired target value y_i is the initial label of $\mathbf{x}_i$, while in discriminant analysis, the desired target vale y_i is the ideal embedding of $\mathbf{x}_i$.

Table 4. Recognition accuracies on ORL (mean±std-dev%)

	Transduction acc	Induction acc
PCA [22]	54.4 ± 2.1	53.5 ± 1.9
Kernel PCA [19]	59.9 ± 2.0	59.4 ± 2.5
Consistency [26]	68.4 ± 2.3	—
LapSVM [2]	69.3 ± 1.8	68.7 ± 1.7
LapRLS [2]	68.6 ± 1.9	67.9 ± 2.0
SDA [5]	68.9 ± 2.0	68.3 ± 1.8
LSSDA	$\mathbf{69.1 \pm 2.1}$	$\mathbf{68.7 \pm 1.9}$
SSDA	$\mathbf{72.1 \pm 1.9}$	$\mathbf{71.3 \pm 2.2}$

1. The *ORL* face dataset[4]. There are ten images for each of the 40 human subjects. The original images (with 256 gray levels) have size 92×112, which are resized to 32×32 for efficiency;

2. The *UMIST* face database [11] consists of 564 gray-level images from 20 persons. The original pre-cropped images are of size 112×92. In our experiment, the images were also resized to 32×32;

3. The *CMU PIE* face dataset [20]. It contains 68 individuals with 41,368 face images as a whole. In our experiments, we choose the frontal pose (C27) with varying lighting and illumination which leaves us 43 images for each individual, and all the images were also resized to 32×32.

In out experiments, we first split each data set into a training set and a testing set, then the training set is further split into a labeled set (which contains one images per person) and an unlabeled set [5]. The algorithms will first be applied on the training set to learn a subspace (for *PCA, KPCA, SDA, LSSDA* and *SSDA*) or a classifier (for *Coonsistency, LapSVM* and *LapRLS*, then the nearest neighbor classifier will be performed in the learned subspace. The final performances of those approaches will be evaluated via two ways, the classification accuracy on the training unlabeled set (*transduction accuracy*) and the testing set (*induction accuracy*). For our *SSDA* and *LSSDA* method, the kernels used to perform *KPCA* and construct projection functions are all Gaussian kernels with their variances setting by 5-fold cross validation. The parameters in other approaches are set by standard ways as in their respective references.

The experimental results are reported in Table 4, 5 and 6, where all the values are summarized from 50 independent runs. From the tables we can clearly observe the superiorities of our methods.

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

[5]For *ORL* data set, we randomly select 7 images per person to construct the training set; for the *UMIST* data set, we randomly select 20 images per person to construct the training set; for the *PIE* data set, we randomly select 30 images per person to construct the training set.

Table 5. Recognition accuracies on UMIST (mean±std-dev%)

	Transduction acc	Induction acc
PCA [22]	51.3 ± 2.3	50.8 ± 2.1
Kernel PCA [19]	55.2 ± 1.9	53.8 ± 2.0
Consistency [26]	59.4 ± 2.2	—
LapSVM [2]	59.8 ± 2.0	58.9 ± 1.6
LapRLS [2]	58.6 ± 1.8	57.4 ± 2.0
SDA [5]	60.0 ± 1.5	59.2 ± 1.8
LSSDA	$\mathbf{60.4 \pm 2.2}$	$\mathbf{58.9 \pm 1.8}$
SSDA	$\mathbf{63.1 \pm 1.9}$	$\mathbf{62.6 \pm 1.8}$

Table 6. Recognition accuracies on PIE (mean±std-dev%)

	Transduction acc	Induction acc
PCA [22]	35.8 ± 1.7	35.6 ± 1.6
Kernel PCA [19]	39.2 ± 2.0	38.9 ± 1.5
Consistency [26]	52.0 ± 1.8	—
LapSVM [2]	56.5 ± 1.7	55.9 ± 2.0
LapRLS [2]	57.2 ± 2.0	56.8 ± 1.8
SDA [5]	59.2 ± 1.6	58.9 ± 1.9
LSSDA	$\mathbf{60.0 \pm 2.3}$	$\mathbf{59.1 \pm 1.8}$
SSDA	$\mathbf{62.4 \pm 2.0}$	$\mathbf{61.9 \pm 1.6}$

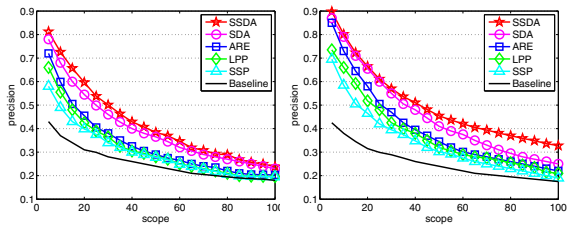
4.2. Relevance Feedback for Image Retrieval

Relevance feedback is an effective framework for narrowing down the gap between low-level visual features and high-level semantic concepts in *Content-Based Image Retrieval* (*CBIR*) [5]. Due to the high dimensionality of the feature space, we hope to find a subspace such that the semantic relationship between images can be better revealed. Clearly the relevance feedback setting is a semi-supervised setting, with a large number of unlabeled data (images in the database) and a small number of labeled data (feedbacks provided by the user).

In our experiments, we use a subset of the *COREL* data set containing 7900 images from 79 categories. We use the same type of features combining the 64-dimensional color histogram and 64-dimensional color texture moment to represent the images as in [5], and we also plot the precision-scope curves to evaluate the performances of the algorithms.

In our experiments, we divide the whole data set into five folds and use one fold as image queries, the other four folds as are used for retrieval, and we report the averaged precision-scope curves as 5-fold cross-validation. We also apply the same automatic feedback scheme as in [5] to model the retrieval process. For each submitted query, our algorithm retrieves and ranks the images in the database. The top 10 ranked images were selected as the feedback images, and their label information (relevant or irrelevant) is used for re-ranking. The images which have been selected at previous iterations are excluded from later selections. For each query, the automatic relevance feedback mechanism is performed for four iterations.

Figure 1 shows the average precision-scope curves of the different algorithms for the 2nd and 4th feedback iterations. The baseline curve corresponds the initial retrieval



(a) Iteration 2 (b) Iteration 4

Figure 1. Image retrieval with user feedback.

result without feedback information. Specifically, at the beginning of retrieval, the Euclidean distances in the original 128-dimensional space are used to rank the images in the database. After the user provides relevance feedbacks, the *Locality Preserving Projections* (*LPP*) [13], *Augmented Relation Embedding* (*ARE*) [16], *Semantic Subspace Projection* (*SSP*) [24], *Semi-supervised Discriminant Analysis* (*SDA*) [5] and our *SSDA* algorithms are then applied to rerank the images in the database. From Fig.1 we can see that our *SSDA* algorithm outperforms the other three algorithms on the entire scope.

5. Conclusions and Discussions

This paper presents a novel semi-supervised discriminant method based on semi-parametric regression other than graph regularization. Our method can easily be generalized to out-of-sample data and the core idea can also be extended to other regression approaches. Finally the experiments on single training image face recognition and image retrieval show the effectiveness of our method.

References

[1] P. N. Belhumeur, J. Hespanda and D. Kiregeman. Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection. *IEEE Trans. on PAMI*. 1997.

[2] M. Belkin, P. Niyogi, and V. Sindhwani. Manifold Regularization: A Geometric Framework for Learning from Labeled and Unlabeled Examples. *JMLR* 7(Nov):2399-2434, 2006.

[3] D. Beymer and T. Poggio. Face Recognition from One Example View. In *Proc. of ICCV*. 1995.

[4] D. Cai, X. He, and J. Han. Spectral Regression for Efficient Regularized Subspace Learning. *Proc. of ICCV*, 2007.

[5] D. Cai, X. He, and J. Han. Semi-Supervised Discriminant Analysis. *Proc. of ICCV*, 2007.

[6] O. Chapelle, B. Schölkopf and A. Zien. *Semi-Supervised Learning*. 508, MIT Press, Cambridge, Mass., USA.

[7] C. Ding and X. He. K-means Clustering via Principal Component Analysis. *Proc. of ICML*, 225-232. 2004.

[8] P. Drineas and M. W. Mahoney. On the Nyström Method for Approximating a Gram Matrix for Improved Kernel-Based Learning. *JMLR* 6, 2153-2175, 2005.

[9] F. R. K., Chung. *Spectral Graph Theory*. Volume 92 of CBMS Regional Conference Series in Mathematics. Published for the Conference Board of the Mathematical Sciences, Washington, DC. 1997.

[10] K. Fukunaga. *Introduction to Statistical Pattern Recognition*. Academic Press, New York, 2nd edtion. 1990.

[11] D. B. Graham and N. M. Allinson. Characterizing Virtual Eigensignatures for General Purpose Face Recognition. In H. Wechsler, P. J. Phillips, V. Bruce, F. Fogelman-Soulie, and T. S. Huang, editors, *Face Recognition: From Theory to Applications, NATO ASI Series F, Computer and Systems Sciences,* vol. 163, 446-456. 1998.

[12] G. Golub and Van Loan. *Matrix Computations*. The Johns Hopkins University Press, Baltimore, 3rd edition, 1996.

[13] X. He and P. Niyogi. Locality Preserving Projections. In *NIPS* 16.

[14] A. K. Jain and B. Chandrasekaran. Dimensionality and Sample Size Considerations in Pattern Recognition Practice. In *Handbook of Statistics*. Amsterdam, North Holland. 1982.

[15] I. T. Jolliffe. *Principal Component Analysis*. Springer-Verlag, New York. 1986.

[16] Y.-Y. Lin, T.-L. Liu, and H.-T. Chen. Semantic manifold learning for image retrieval. In Proceedings of the ACM Conference on Multimedia, 2005.

[17] S. Mika, G. Rätsch, J. Weston, B. Schölkopf and K.-R. Müller. Fisher Discriminant Analysis with Kernels. *Neural Networks for Signal Processing IX*, IEEE. 1999.

[18] D. Ruppert, M. P. Wand, and R. J. Carroll. *Semiparametric Regression*. Cambridge University Press, 2003.

[19] B. Schölkopf and A. Smola. *Learning with Kernels*. The MIT Press. Cambridge, Massachusetts. London, England. 2002.

[20] T. Sim, S. Baker, and M. Bsat. The CMU pose, illuminlation, and expression database. *IEEE Trans. on PAMI*. 2003.

[21] V. Sindhwani, P. Niyogi and M. Belkin. Beyond the Point Cloud: from Transductive to Semi-supervised Learning. *Proc. of ICML*, 2005.

[22] M. A. Turk and A. P. Pentland. Eigenfaces for Recognition. *Journal of Cognitive Neuroscience*, 3(1): 71-96, 1991.

[23] F. Wang and C. Zhang. Label Propagation Through Linear Neighborhoods. *Proc. of ICML*, 2006.

[24] J. Yu and Q. Tian. Learning image manifolds by semantic subspace projection. In Proceedings of the ACM Conference on Multimedia, 2006.

[25] K. Zhang and J. T. Kwok. Block-Quantized Kernel Matrix for Fast Spectral Embedding. *Proc. of ICML*, 1097-1104, 2003.

[26] D. Zhou, O. Bousquet, T. N. Lal, J. Weston, and B. Schölkopf. Learning with Local and Global Consistency. *NIPS 16*. 2004.