Semisupervised Dimensionality Reduction and Classification Through Virtual Label Regression

Feiping Nie, Dong Xu, Xuelong Li, *Senior Member, IEEE*, and Shiming Xiang

***Abstract*—Semisupervised dimensionality reduction has been attracting much attention as it not only utilizes both labeled and unlabeled data simultaneously, but also works well in the situation of *out-of-sample*. This paper proposes an effective approach of semisupervised dimensionality reduction through label propagation and label regression. Different from previous efforts, the new approach propagates the label information from labeled to unlabeled data with a well-designed mechanism of *random walks*, in which outliers are effectively detected and the obtained virtual labels of unlabeled data can be well encoded in a weighted regression model. These virtual labels are thereafter regressed with a linear model to calculate the projection matrix for dimensionality reduction. By this means, when the manifold or the clustering assumption of data is satisfied, the labels of labeled data can be correctly propagated to the unlabeled data; and thus, the proposed approach utilizes the labeled and the unlabeled data more effectively than previous work. Experimental results are carried out upon several databases, and the advantage of the new approach is well demonstrated.**

***Index Terms*—Dimensionality reduction, label propagation, label regression, semisupervised learning, subspace learning.**

I. Introduction

THE dimensionality of data is usually very high—more than a thousand or so in many real applications, e.g., face recognition [1]–[4], cross-media retrieval [5], text categorization, gene expression data classification [6], and image segmentation [7]. Directly working on such high dimensional data is not only time consuming but also computationally unreliable. Therefore, dimensionality reduction plays an important role to solve these problems. Over the past years, many approaches of dimensionality reduction have been proposed [8]–[12], and the most representative ones are probably principal components analysis (PCA) and linear discriminant analysis (LDA) [13]. PCA is an unsupervised dimensionality reduction method, which aims to maximize the variance in the low-dimensional space while LDA is supervised and its goal is to maximize the discriminative power in the low-dimensional space.

In general, the supervised approaches of dimensionality reduction are suitable for tasks of classification when there are sufficient (i.e., a *large-enough* pool) labeled data available. Unfortunately, in real cases, labeled data are usually scarce, and to label a large number of data would require expensive human labor in practice. On the other hand, unlabeled data are usually abundant and relatively easier to obtain. To effectively utilize the labeled and unlabeled data simultaneously, semisupervised learning was proposed and has attracted much attention recently [14].

Many traditional semisupervised learning approaches, such as Gaussian Field and Harmonic Function (GFHF) [15] and Local and Global Consistency (LGC) [16] work on transductive setting and cannot deal with arbitrary new coming data, which is known as the *out-of-sample* problem [17]. In contrast, semisupervised dimensionality reduction methods not only reduce the dimension but also naturally solve the out-of-sample problem. It is more practical and therefore has attracted more interest in practice. Many semisupervised dimensionality reduction methods have been proposed in the past few years [18]–[24]. The basic ideas of most of these methods can be categorized into two main types, namely: 1) to consider the manifold regularization term [25] in the original objective function of the approaches for supervised dimensionality reduction; and 2) to construct graph using the information of labeled and unlabeled data and then to directly perform the graph-based dimensionality reduction approaches with the constructed graph [26].

This paper proposes a novel approach for semisupervised dimensionality reduction. The new approach has two main steps, namely: 1) it propagates the label information from labeled to unlabeled data, from which the virtual labels of unlabeled data are obtained; and 2) it then regresses these virtual labels with a linear model to calculate the projection matrix for dimensionality reduction. In here, the first step of label propagation is a double-blade sword. On one hand, when the condition of the manifold assumption or the clustering assumption of data is well satisfied, the labels of labeled data can be correctly propagated to the unlabeled data; and thus, the proposed method utilizes both labeled and unlabeled data more effectively. On the other hand, when the two assumptions however do not hold, the procedure of label propagation might not be reliable, which could severely deteriorate the performance of the proposed approach. To reduce the negative effect, a new label propagation mechanism is adopted in a special way named *random walks*, in which the outlier of data can be effectively detected, and

Manuscript received November 3, 2009; revised June 18, 2010; accepted August 17, 2010. Date of publication November 29, 2010; date of current version May 18, 2011. This work was supported by the National Basic Research Program of China (973 Program) under Grant 2011CB707100 and by the National Natural Science Foundation of China under Grant 61072093. This paper was recommended by Associate Editor L. Wang.

F. Nie and D. Xu are with the School of Computer Engineering, Nanyang Technological University, Singapore 639798.

X. Li is with the Center for OPTptical IMagery Analysis and Learning (OPTIMAL), State Key Laboratory of Transient Optics and Photonics, Xi'an Institute of Optics and Precision Mechanics, Chinese Academy of Sciences, Xi'an 710119, China (e-mail: xuelong_li@opt.ac.cn).

S. Xiang is with the Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSMCB.2010.2085433

the obtained virtual labels are probability values that can be conveniently encoded in the weighted linear regression model.

The newly proposed method can actually be regarded as a methodological framework for extending existing approaches of any supervised/unsupervised dimensionality reduction to the form of semisupervised ones. Its motivation and basic ideas are significantly different from previous efforts for the same purpose. Previous work mainly adds the manifold regularization term into the original objective functions and/or performs original graph-based algorithms with the graph constructed by the labeled and unlabeled information. However, in this new framework, a label propagation with outlier detection is first developed to obtain the virtual labels of unlabeled data, and then any other supervised or unsupervised dimensionality reduction algorithm can be directly extended into the semisupervised counterpart by performing this algorithm with the labels of labeled data and the virtual labels of unlabeled data. More importantly, due to the label propagation procedure, the new framework could explore the distribution of the labeled and unlabeled data more effectively to learn a better subspace for dimensionality reduction when the manifold assumption or the clustering assumption of data, which is the basic assumptions for semisupervised learning, is well satisfied.

II. Transductive Learning and Outlier Detection via Random Walks

Given labeled and unlabeled data $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n \in \mathbb{R}^d$, the goal of transductive learning is to predict the class labels for the unlabeled data. Random walks on graph is an effective approach for transductive learning [15]. Via random walks, the distribution of labeled and unlabeled data can be effectively explored to learn the labels for the unlabeled data. First, we construct a neighborhood weighted graph on given data. There are many methods to construct the graph. For example, we could construct the graph through many manifold methods, such as locally linear embedding [27], local tangent space alignment [28], local spline embedding [29], etc. In this paper, we use a popular method to construct the graph as follows: If $\boldsymbol{x}_i$ is in the k-neighbors of $\boldsymbol{x}_j$ or $\boldsymbol{x}_j$ is in the k-neighbors of $\boldsymbol{x}_i$, then $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are linked by a weight computed by

$$\mathbf{A}_{ij} = e^{-\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2/\sigma^2} \tag{1}$$

otherwise, $\mathbf{A}_{ij} = 0$. Here, σ is the variance, $\|\cdot\|$ is the two-norm of the vector, i.e., $\|\boldsymbol{x}\|^2 = \boldsymbol{x}^T\boldsymbol{x}$. The transition probability matrix on the graph is defined as $\mathbf{P} = \mathbf{D}^{-1}\mathbf{A}$, where $\mathbf{D}$ is a diagonal matrix with the ith diagonal element as $\mathbf{D}_{ii} = \sum_j \mathbf{A}_{ij}$.

In the traditional random walk approaches [15], walks starting from an unlabeled data point can only stop at a labeled data, which indicates that the unlabeled data can only be predicted to one of the labeled classes. While in many real world applications, there are some outliers in the data or the labeled classes might not cover all the possible classes in the application. In order to capture the outliers or the data from the class other than the labeled classes, we consider special random walks on the graph (see Fig. 1) and define a new transition probability matrix $\tilde{\mathbf{P}}$ as

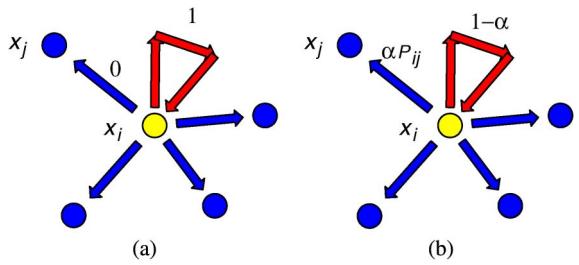

Fig. 1. Each data point x_i randomly walks to its neighbors with the probability determined by P. There is a probability $1-\alpha$ to return to itself at one walk. The walks will stop when they hit one of the data points on the graph twice consecutively. (a) Labeled data. (b) Unlabeled data.

$$\tilde{\mathbf{P}} = \mathbf{I}_\beta + \mathbf{I}_\alpha\mathbf{P} \tag{2}$$

where $\mathbf{I}_\alpha$ is a diagonal matrix; the ith diagonal element of which is zero if $\boldsymbol{x}_i$ is a labeled data point and is $\alpha(0 \le \alpha < 1)$; otherwise, $\mathbf{I}_\beta = \mathbf{I} - \mathbf{I}_\alpha$ in which $\mathbf{I}$ denotes the identity matrix. Note that the sum of each row of $\tilde{\mathbf{P}}$ is equal to one, which indicates $\tilde{\mathbf{P}}$ is a stochastic matrix. Based on the transition probability matrix $\tilde{\mathbf{P}}$, the stop rule of the special random walks is defined as the following: Each point walks randomly on the graph based on the transition probability matrix $\tilde{\mathbf{P}}$ and stops when it consecutively hits one of the points on the graph twice. It is considered to have hit the starting point once before the walks.

Define the initial label matrix $\mathbf{Y} = [\mathbf{Y}_1^T, \ldots, \mathbf{Y}_n^T]^T \in \mathbb{R}^{n\times(c+1)}$, where $\mathbf{Y}_i \in \mathbb{R}^{1\times(c+1)}(1 \le i \le n)$, c is the number of classes. For the labeled data $\boldsymbol{x}_i$, $\mathbf{Y}_{ij} = 1$ if $\boldsymbol{x}_i$ is labeled as j and $\mathbf{Y}_{ij} = 0$ otherwise. For the unlabeled data $\boldsymbol{x}_i$, $\mathbf{Y}_{ij} = 1$ if $j = c+1$ and $\mathbf{Y}_{ij} = 0$ otherwise. Note that we add an additional class $c+1$ in order to detect outlier data.

Denote $\hat{\mathbf{P}} = \mathbf{I}_\alpha\mathbf{P}$ and $(\hat{\mathbf{P}}^n)_{ij}$ as the (i,j)th entry of $\hat{\mathbf{P}}^n$. It can be seen that the value of $(\hat{\mathbf{P}}^n)_{ij}$ is the probability of the ith point reaching the jth point at the nth step before stop.

Denote $\mathbf{G}$ as

$$\mathbf{G} = \mathbf{I}_\beta + \hat{\mathbf{P}}\mathbf{I}_\beta + \hat{\mathbf{P}}^2\mathbf{I}_\beta + \cdots + \hat{\mathbf{P}}^n\mathbf{I}_\beta + \cdots. \tag{3}$$

Note that the value of $(\hat{\mathbf{P}}^k\mathbf{I}_\beta)_{ij}$ is the probability of the ith point stopping the walks at the jth point at the kth step, so $\mathbf{G}_{ij}$ is the probability of the ith point stopping the walks at the jth point.

Note that the ∞-norm of matrix $\hat{\mathbf{P}}$ is lower than one in the case of $0 \le \alpha < 1$. According to the matrix property, the spectral radius of $\hat{\mathbf{P}}$ is less than the ∞-norm, i.e., $\rho(\hat{\mathbf{P}}) < 1$. Therefore, $\mathbf{I} - \hat{\mathbf{P}}$ is invertible, and $\lim_{t\to\infty}\hat{\mathbf{P}}^t = \mathbf{0}$, so we have $\lim_{t\to\infty}\sum_{i=0}^{t-1}(\mathbf{I}_\alpha\mathbf{P})^i\mathbf{I}_\beta = (\mathbf{I} - \mathbf{I}_\alpha\mathbf{P})^{-1}\mathbf{I}_\beta$; and thus, $\mathbf{G}$ can be written as $\mathbf{G} = (\mathbf{I} - \mathbf{I}_\alpha\mathbf{P})^{-1}\mathbf{I}_\beta$. Calculate the soft label matrix $\mathbf{F} = [\mathbf{F}_1^T, \ldots, \mathbf{F}_n^T]^T \in \mathbb{R}^{n\times(c+1)}$ by

$$\mathbf{F} = \mathbf{G}\mathbf{Y}. \tag{4}$$

We can see that $\mathbf{F}_{ij}(j \le c)$ is the probability of the ith point which stops the random walks at the labeled data point whose label is j, and $\mathbf{F}_{ij}(j = c+1)$ is the probability of the ith point which stops the random walks at one of the unlabeled data

point. The random walks that can stop at the unlabeled data points makes our algorithm having the mechanism to discover novel class or outlier in data.

Denote $\mathbf{1}_n = [1, \ldots, 1]^T \in \mathbb{R}^{n\times 1}$, we have

$$\left.\begin{array}{l}\mathbf{P}\mathbf{1}_n = \mathbf{1}_n \\ \mathbf{Y}\mathbf{1}_{c+1} = \mathbf{1}_n\end{array}\right\}$$
$$\Rightarrow \mathbf{I}_\alpha \mathbf{P}\mathbf{1}_n + \mathbf{I}_\beta \mathbf{Y}\mathbf{1}_{c+1} = \mathbf{1}_n$$
$$\Rightarrow \mathbf{I}_\beta \mathbf{Y}\mathbf{1}_{c+1} = (\mathbf{I} - \mathbf{I}_\alpha \mathbf{P})\mathbf{1}_n$$
$$\Rightarrow (\mathbf{I} - \mathbf{I}_\alpha \mathbf{P})^{-1}\mathbf{I}_\beta \mathbf{Y}\mathbf{1}_{c+1} = \mathbf{1}_n. \quad (5)$$

Therefore, the elements in $\mathbf{F}$ are probability values, and $\mathbf{F}_{ij}$ can be seen as an estimation of the posterior probability of $\boldsymbol{x}_i$ belonging to class j. When $j = c+1$, $\mathbf{F}_{i,c+1}$ denotes the probability of $\boldsymbol{x}_i$ belonging to the outlier.

In the special random walks, if an unlabeled sample is similar to one of the labeled samples, the walks starting from this unlabeled sample will stop at the labeled sample with high probability. If this unlabeled sample is an outlier or from a novel class, this sample is not similar to any of the labeled samples, and the walks starting from this unlabeled sample will stop at one of the unlabeled samples with high probability. Therefore, the outliers would be effectively detected by this special random walks, and it is enough to treat all the outliers with one extra class. Moreover, it is worth to emphasize that the obtained virtual label for each data by the special random walks is of probability value. As it can be seen in the next section, using the virtual labels with probability values, it is convenient to construct a weighted label regression model to learn the optimal subspace.

III. Efficient Semisupervised Subspace Learning and Classification Through Virtual Label Regression

Transductive learning only predicts the labels of the given unlabeled data and cannot predict the label for new coming data point. The reason is that transductive learning does not learn a model. To handle this out-of-sample problem, we use a linear model to approximate the predicted labels of the unlabeled data and propose an efficient semisupervised subspace learning through label regression. As the linear model is introduced in the algorithm, new data can also be handled with the learned projection matrix $\mathbf{W}$; and thus, the out-of-sample problem is naturally solved with the algorithm.

A. Label Regression

Consider the linear model as follows:

$$\mathbf{y} = \mathbf{W}^T\mathbf{x} + \mathbf{b} \quad (6)$$

where $\mathbf{W} \in \mathbb{R}^{d\times c}$ is the projection matrix; and $\boldsymbol{b} \in \mathbb{R}^{c\times 1}$ is the bias term.

Denote $\boldsymbol{t}_i = [\underbrace{0, \ldots, 0}_{i-1}, 1, \underbrace{0, \ldots, 0}_{c-i}]^T$ as the class indicator vector for the ith class. We consider a weighted and regularized least squares regression problem as follows:

$$\mathbf{W} = \arg\min \mathcal{J}(\mathbf{W}, \mathbf{b}) \quad (7)$$

where

$$\mathcal{J}(\mathbf{W}, \boldsymbol{b}) = \gamma\|\mathbf{W}\|^2 + \sum_{i=1}^{n}\sum_{j=1}^{c}\mathbf{F}_{ij}\|\mathbf{W}^T\boldsymbol{x}_i + \boldsymbol{b} - \boldsymbol{t}_j\|^2 \quad (8)$$

and γ is the regularization parameter. From (8), we can see that the virtual labels with probability values can be conveniently encoded in this weighted regression model. When $\mathbf{F}_{i,c+1}$ is a large value near one, which implies that the probability of $\boldsymbol{x}_i$ belonging to outlier is large, then the values of $\mathbf{F}_{i,j}(1 \le i \le c)$ are small, and then the effect of the outlier data point $\boldsymbol{x}_i$ is not too much in the regression model according to (8).

For convenience, we rewrite (8) in matrix form as

$$\mathcal{J}(\mathbf{W}, \boldsymbol{b}) = \gamma tr(\mathbf{W}^T\mathbf{W}) + tr(\mathbf{W}^T\mathbf{X} + \boldsymbol{b}\mathbf{1}^T)\mathbf{S}(\mathbf{W}^T\mathbf{X} + \boldsymbol{b}\mathbf{1}^T)^T - 2tr\mathbf{F}_c(\mathbf{W}^T\mathbf{X} + \boldsymbol{b}\mathbf{1}^T) \quad (9)$$

where $\mathbf{S} \in \mathbb{R}^{n\times n}$ is a diagonal matrix with the i diagonal element being $\mathbf{S}_{ii} = \sum_{j=1}^{c}\mathbf{F}_{ij}$; $\mathbf{F}_c \in \mathbb{R}^{n\times c}$ is formed by the first c columns of $\mathbf{F}$, $\mathbf{1}_n = [1, 1, \ldots, 1]^T \in \mathbb{R}^{n\times 1}$; and the data matrix $\mathbf{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n] \in \mathbb{R}^{d\times n}$.

By setting the derivative w.r.t. $\boldsymbol{b}$ and $\mathbf{W}$ to zero, respectively, we have

$$\begin{cases}\boldsymbol{b} = \frac{1}{\mathbf{1}_n^T\mathbf{S}\mathbf{1}_n}\left(\mathbf{F}_c^T - \mathbf{W}^T\mathbf{X}\mathbf{S}\right)\mathbf{1}_n, \\ \gamma\mathbf{W} + \mathbf{X}\mathbf{S}\mathbf{X}^T\mathbf{W} + \mathbf{X}(\mathbf{S}\mathbf{1}_n\boldsymbol{b}^T - \mathbf{F}_c) = \mathbf{0}.\end{cases} \quad (10)$$

According to (15), the optimal solution to the regression problem (18) is

$$\mathbf{W} = (\mathbf{X}\mathbf{L}_s\mathbf{X}^T + \gamma\mathbf{I})^{-1}\mathbf{X}\mathbf{C}_s\mathbf{F}_c \quad (11)$$

where $\mathbf{I}$ denotes an identity matrix with proper size; $\mathbf{L}_s = \mathbf{S} - (1/\mathbf{1}_n^T\mathbf{S}\mathbf{1}_n)\mathbf{S}\mathbf{1}_n\mathbf{1}_n^T\mathbf{S}$ is a Laplacian matrix; and $\mathbf{C}_s = \mathbf{I} - (1/\mathbf{1}_n^T\mathbf{S}\mathbf{1}_n)\mathbf{S}\mathbf{1}_n\mathbf{1}_n^T$ is a weighted centering matrix, which is also a Laplacian matrix.

B. Efficient Semisupervised Subspace Learning

Now we obtain the projection matrix $\mathbf{W} \in \mathbb{R}^{d\times c}$ for subspace learning via the label propagation and regression. For any data point $\boldsymbol{x} \in \mathbb{R}^{d\times 1}$, the projected data point $\boldsymbol{y}$ can be calculated by

$$\boldsymbol{y} = \mathbf{W}^T\boldsymbol{x} = \mathbf{F}_c^T\mathbf{C}_s^T\mathbf{X}^T(\mathbf{X}\mathbf{L}_s\mathbf{X}^T + \gamma\mathbf{I})^{-1}\boldsymbol{x}. \quad (12)$$

We describe the proposed semisupervised subspace learning algorithm in Table I. From Table I, we can see that the algorithm needs to calculate the soft label matrix $\mathbf{F}$ by (4) and calculate the projection matrix $\mathbf{W}$ by (11). In fact, the $\mathbf{F}$ and $\mathbf{W}$ can be obtained by solving the linear system of equations, which is very efficient and can be performed on large-scale data sets. In contrast, LDA and semisupervised discriminant analysis (SDA) are to solve the eigenvalue decomposition problem, which is computationally expensive. Therefore, the algorithm proposed in this paper is much more efficient.

TABLE I
EFFICIENT SEMISUPERVISED DIMENSIONALITY REDUCTION ALGORITHM

Input:
Data matrix $\mathbf{X} \in \mathbb{R}^{d\times n}$ (each column is a data point). regularization parameter γ
Output:
The projection matrix $\mathbf{W} \in \mathbb{R}^{d\times c}$.
Algorithm:
1. Construct the neighborhood graph and calculate the weight matrix $\mathbf{A}$.
2. Label propagation and obtain the soft label matrix $\mathbf{F}$ by Eq.(4).
3. Label regression and obtain the projection matrix $\mathbf{W}$ by Eq.(11) or Eq.(13).

Note that $(\mathbf{X}\mathbf{L}_s\mathbf{X}^T + \gamma\mathbf{I})^{-1}\mathbf{X} = \mathbf{X}(\mathbf{L}_s\mathbf{X}^T\mathbf{X} + \gamma\mathbf{I})^{-1}$, so (11) can be rewritten as

$$\mathbf{W} = \mathbf{X}(\mathbf{L}_s\mathbf{X}^T\mathbf{X} + \gamma\mathbf{I})^{-1}\mathbf{C}_s\mathbf{F}_c \tag{13}$$

and thus, for any data point $\boldsymbol{x} \in \mathbb{R}^{d\times 1}$, the projected data point $\boldsymbol{y}$ can be calculated by $\boldsymbol{y} = \mathbf{W}^T\boldsymbol{x} = \mathbf{F}_c^T\mathbf{C}_s^T(\mathbf{X}^T\mathbf{X}\mathbf{L}_s + \gamma\mathbf{I})^{-1}\mathbf{X}^T\boldsymbol{x}$. Note that the algorithm using (13) only involves inner product operation; therefore, the proposed algorithm can be easily extended to the nonlinear version by using the kernel tricks. We can also use the general kernelization framework in [30] to obtain the kernel version of the proposed linear algorithm.

C. Directly Classification With Verification

Once the projection matrix $\mathbf{W}$ is obtained, we can project data by $\mathbf{W}$ and apply any classifier, such as the nearest neighbor classifier on the projected low-dimensional data for classification. Similarly to the regularized least squares and Laplacian regularized least squares (LapRLS/L) approaches, we can also directly classify data using the classification function $\boldsymbol{y} = \mathbf{W}^T\boldsymbol{x} + \boldsymbol{b}$ as they all use the linear model as in (6). However, in some applications, the performance reaches the optimal when the regularization parameter is set to a large value. In this case, the values $\boldsymbol{y} = \mathbf{W}^T\boldsymbol{x}_i + \boldsymbol{b}$ would deviate severely from the true label values to be regressed for the labeled data $\boldsymbol{x}_i$. In this paper, we propose to use an additional affine transformation matrix $\mathbf{W}_o \in \mathbb{R}^{c\times c}$ and bias $\boldsymbol{b}_o \in \mathbb{R}^{c\times 1}$ to rectify the deviation for the labeled data. Without loss of generalization, we assume that the first l data are labeled, and the remaining data are unlabeled. Specifically, we solve the following optimization problem:

$$\{\mathbf{W}_o, \boldsymbol{b}_o\} = \arg\min \sum_{i=1}^{l} \left\|\mathbf{W}_o^T(\mathbf{W}^T\boldsymbol{x}_i + \boldsymbol{b}) + \boldsymbol{b}_o - \boldsymbol{t}_{y_i}\right\|^2 \tag{14}$$

where $\mathbf{W}$ and $\boldsymbol{b}$ are calculated in Section III-A; and y_i is the class label of $\boldsymbol{x}_i$. Denote the labeled data matrix by $\mathbf{X}_l = [\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_l]$, it is easy to derive that the optimal solution to the aforementioned problem is

$$\begin{cases} \mathbf{W}_o = (\tilde{\mathbf{F}}^T\mathbf{L}_c\tilde{\mathbf{F}})^{-1}\tilde{\mathbf{F}}^T\mathbf{L}_c\mathbf{T}, \\ \boldsymbol{b}_o = \frac{1}{l}\left(\tilde{\mathbf{F}}^T - \mathbf{W}_o^T\mathbf{X}_l\right)\mathbf{1}_l \end{cases} \tag{15}$$

where $\mathbf{L}_c = \mathbf{I} - (1/l)\mathbf{1}_l\mathbf{1}_l^T$ is the centering matrix; $\tilde{\mathbf{F}} = \mathbf{X}_l^T\mathbf{W} + \mathbf{1}_l\boldsymbol{b}^T \in \mathbb{R}^{l\times c}$; $\mathbf{1}_l = [1, 1, \ldots, 1]^T \in \mathbb{R}^{l\times 1}$; and $\mathbf{T} = [\boldsymbol{t}_{y_1}, \boldsymbol{t}_{y_2}, \ldots, \boldsymbol{t}_{y_l}]^T \in \mathbb{R}^{l\times c}$.

Then, for any data point $\boldsymbol{x} \in \mathbb{R}^{d\times 1}$, the projected data point $\boldsymbol{y}$ can be calculated by $\boldsymbol{y} = \mathbf{W}_o^T(\mathbf{W}^T\boldsymbol{x} + \boldsymbol{b}) + \boldsymbol{b}_o$, and the class label i^* of $\boldsymbol{x}$ is predicted by

$$i^* = \arg\max_i \boldsymbol{y}(i) \tag{16}$$

where $\boldsymbol{y}(i)$ is the ith element of $\boldsymbol{y}$.

D. Discussion

Given the weight matrix $\mathbf{A}$ as in (1), the computation complexity of calculating the virtual label matrix $\mathbf{F}$ with (4) is $O(knc)$, where k, n, and c are the number of neighbors, data, and classes, respectively. Following the idea in [31], we can see that the computation complexity of calculating the projection matrix $\mathbf{W}$ with (11) is $O(nsc)$, where s is the averaged number of nonzero of data points. In practice, $k \ll s$; thus, the computation complexity of the whole algorithm proposed in this paper is $O(nsc)$. While traditional dimensionality reduction methods involve eigen-decomposition, the computation complexity is $O(d^3)$ or $O(d^2c)$ if only c eigenvectors are to be calculated, where d is the dimensionality of data points. Obviously, when the training data is very highly dimensional and sparse (i.e., s is very small), the method proposed in this paper is much more efficient than traditional dimensionality reduction methods.

Recently, a series work on spectral regression (SR) was proposed [32], which also uses regression for dimensionality reduction. Although SR can also employ semisupervised dimensionality reduction, the idea is significantly different from us. In SR, the virtual label matrix $\mathbf{G} \in \mathbb{R}^{n\times c}$ is calculated by eigen-decomposition as follows:

$$\mathbf{L}_1\mathbf{G} = \mathbf{L}_2\mathbf{G}\mathbf{\Lambda} \tag{17}$$

where $\mathbf{L}_1$ and $\mathbf{L}_2$ are two Laplacian matrices that lie on a specific graph-based embedding algorithm; and $\mathbf{\Lambda}$ is the eigenvalue matrix of $\mathbf{L}_1$ and $\mathbf{L}_2$. Then, the projection matrix $\mathbf{W}$ is obtained by solving the following regularized least squares regression:

$$\mathbf{W} = \arg\min_{\mathbf{W},\mathbf{b}} \gamma\|\mathbf{W}\|^2 + \sum_{i=1}^{n} \|\mathbf{W}^T\boldsymbol{x}_i + \boldsymbol{b} - \boldsymbol{g}^i\|^2 \tag{18}$$

where $\boldsymbol{g}^i$ denotes the ith row of $\mathbf{G}$.

We can see that both of the two steps of the SR algorithm are different from that of the proposed algorithm in this paper. In the first step, SR directly constructs the Laplacian matrices using labeled and unlabeled data, while our method uses a well-developed label propagation with outlier detection, in which

the distribution of the labeled and unlabeled data are explored more effectively. In the second step, SR uses the ordinary regularized least squares regression to obtain the projection matrix, while our method uses a weighted regularized least squares regression such that the effect of outliers could be alleviated in the regression.

IV. Experiments

We evaluate the proposed semisupervised subspace learning algorithm through virtual label regression (VLR) on image recognition problem and compare it with some state-of-the-art algorithms, including regularized linear discriminant analysis (RLDA) [33], GFHF [15], LGC [16], SDA [19] and LapRLS/L [34].

We use four image databases in the experiments, including University of Manchester Institute of Science and Technology (UMIST), Columbia Object Image Library (COIL)-20, YALE-B, and Carnegie Mellon University Pose, Illumination, and Expression (CMU PIE). The UMIST repository is a multiview face database, consisting of 575 images of 20 people, each covering a wide range of poses from profile to frontal views. The size of each cropped image is 112×92 with 256 gray levels per pixel [35]. We down-sample the size of each image to 28×23, and no other preprocessing is performed. The COIL-20 data set [36] consists of images of 20 objects viewed from varying angles at the interval of five degrees, resulting in 72 images per object. Similarly, each image is down-sampled to the size of 32×32. The YALE-B database [37] used in this experiment contains 38 subjects, with each person having around 64 near frontal images under different illuminations. The images are cropped and then resized to 32×32 pixels. The CMU PIE database [38] contains more than 40 000 facial images of 68 people. The images were acquired over different poses, under variable illumination conditions, and with different facial expressions. In the experiment, we choose the images from the frontal pose (C27), and each subject has around 49 images from varying illuminations and facial expressions. The images are cropped and then resized to 32×32 pixels.

For the semisupervised algorithms GFHF, LGC, SDA, LapRLS/L, and VLR, the weights in the neighborhood graph are computed by (1), and the variance σ is determined by

$$\sigma = \tau \sqrt{-\frac{\bar{d}}{\ln(1/k)}} \tag{19}$$

where $\bar{d}$ is the average of squared Euclidean distances for all the edged pairs on the graph; k is the neighbor number to construct the neighborhood graph; and τ is a parameter. Note that we transform the parameter σ in (1) to τ. The parameter σ should be tuned from the range of $(0, \infty)$, while the parameter τ only need to be tuned from the range of $(0, 1]$. Experimental results show that when $k = 10$ and $\tau = 0.3$, all the compared semisupervised algorithms can obtain good performance in general. Therefore, we fix the parameters $k = 10$ and $\tau = 0.3$ in the experiments. For the dimensionality reduction algorithms RLDA and SDA, the reduced dimension is $c - 1$, while for LapRLS/L and VLR, the reduced dimension is c, where c is the number of classes. There is one regularization parameter in RLDA and VLR, and two regularization parameters in SDA and LapRLS/L. All the regularization parameters are searched from grid $\{10^{-9}, 10^{-6}, 10^{-3}, 10^{0}, 10^{3}, 10^{6}, 10^{9}\}$. The parameter α in VLR is searched from grid {0.1, 0.3, 0.6, 0.9, 0.9999, 0.9999999, 0.9999999999}. Note that there all are two parameters to be tuned in the semisupervised dimensionality reduction algorithms SDA, LapRLS/L, and VLR for fair comparison.

In all the experiments, PCA is used as a preprocessing step to remove the null space of data covariance matrix and preserve 95% energy of data. We randomly select 60% data as the transductive training set and the remaining data as the unseen test set. Among the training set, we randomly label 1, 3, or 5 samples per class and remain the other samples as unlabeled data. For RLDA, only the labeled set is used to learn the subspace, while for GFHG, LGC, SDA, LapRLS/L, and VLR, the whole training set is used to learn the subspace. Note that when only one sample in each class is labeled, the supervised method RLDA cannot be performed. We report the mean recognition accuracy and standard deviation corresponding to the best parameters over 20 random splits on the unlabeled data set and the unseen test data set.

A. Comparison of Performance With Direct Classification Before and After Verification

In this experiment, we verify the performance of LapRLS/L and VLR with direct classification before and after the proposed verification by (15). The experimental results are reported in Table II. From the results, we observe that, in most cases, the proposed verification would improve performance when the direct classification function $\boldsymbol{y} = \mathbf{W}^T \boldsymbol{x} + \boldsymbol{b}$ is used in LapRLS/L and VLR.

B. Comparison of Performance With Nearest Neighbor Classifier

In this experiment, we verify the performance of VLR, and compare it with other state-of-the-art dimensionality reduction methods with the one-nearest neighbor classifier. The experimental results are reported in Table III. From the results we have the following observations:

1) The compared semi-supervised dimension reduction algorithms outperform supervised LDA, which demonstrates that unlabeled data can be used to improve the recognition performance.
2) Our method VLR outperforms LDA, SDA and LapRLS/L in most cases in terms of the recognition accuracy on the unlabeled data set and the unseen test data set. Although GFHF and LGC outperforms VLR in some cases, GFHF and LGC cannot cope with the unseen data.
3) When the manifold assumption or the clustering assumption is well satisfied such as in UMIST and COIL-20 databases, GFHF, LGC and our method VLR perform much better than SDA and LapRLS/L.
4) When the manifold assumption or the clustering assumption is not well satisfied such as in Yale-B and CMU PIE databases, traditional label propagation algorithms GFHF

TABLE II
RECOGNITION PERFORMANCE (MEAN RECOGNITION ACCURACY ± STANDARD DEVIATION %) OF LAPRLS/L AND VLR OVER 20 RANDOM SPLITS ON FOUR DATABASES. THE CLASSIFIER IS USING THE CLASSIFICATION FUNCTION $\boldsymbol{y} = \mathbf{W}^T\boldsymbol{x} + \boldsymbol{b}$ WITH OR WITHOUT THE VERIFICATION BY (15)

dataset	method		1 labeled sample		3 labeled samples		5 labeled samples	
			Unlabel(%)	Test(%)	Unlabel(%)	Test(%)	Unlabel(%)	Test(%)
UMIST	LapRLS/L	Before Verif.	54.5±3.9	55.1±5.3	74.2±4.1	74.8±4.6	83.5±3.9	82.7±4.1
		After verif.	55.2±3.6	56.0±5.5	76.8±4.2	76.9±4.8	84.6±3.9	83.7±3.6
	VLR	Before Verif.	73.5±3.8	72.1±4.9	83.9±3.6	82.1±3.6	91.8±3.5	90.1±3.1
		After verif.	74.2±3.9	72.9±4.5	84.6±3.5	83.2±3.6	91.7±3.6	90.2±2.9
COIL-20	LapRLS/L	Before Verif.	59.5±2.5	60.2±3.6	74.7±2.4	74.5±2.6	80.5±2.3	80.2±2.9
		After verif.	59.3±2.6	60.0±3.9	75.1±2.5	74.9±2.3	80.6±2.1	80.5±2.7
	VLR	Before Verif.	83.5±2.1	81.6±2.5	86.9±2.1	84.5±2.3	89.5±1.5	87.1±1.3
		After verif.	83.3±2.2	81.5±2.7	87.7±2.1	85.1±2.2	90.1±1.2	88.2±1.2
YALE-B	LapRLS/L	Before Verif.	54.3±2.5	54.5±2.7	88.6±2.5	89.1±2.3	95.5±1.3	95.8±1.5
		After verif.	54.1±2.6	54.2±2.5	88.8±2.5	89.5±2.2	95.9±1.2	96.1±1.5
	VLR	Before Verif.	54.6±3.3	54.3±3.7	88.9±2.3	89.3±2.1	95.7±1.3	95.9±1.2
		After verif.	54.3±3.1	54.1±3.6	89.2±2.2	89.6±2.2	96.1±1.3	96.6±1.2
CMU PIE	LapRLS/L	Before Verif.	60.3±2.3	60.3± 2.7	86.6±1.1	86.3±1.2	90.9±1.1	90.2±1.2
		After verif.	60.0±2.6	60.1±2.8	86.8±1.2	86.5±1.3	91.2±1.0	90.5±1.3
	VLR	Before Verif.	60.5±2.6	60.4±2.8	86.9±1.2	86.5±1.3	91.1±1.0	90.6±1.2
		After verif.	61.3±2.1	61.2±2.2	87.2±1.0	86.6±1.5	91.6±1.2	91.2±1.1

TABLE III
RECOGNITION PERFORMANCE (MEAN RECOGNITION ACCURACY ± STANDARD DEVIATION %) OF RLDA, GFHF, LGC, SDA, LAPRLS/L, AND VLR OVER 20 RANDOM SPLITS ON FOUR DATABASES

dataset	method	1 labeled sample		3 labeled samples		5 labeled samples	
		Unlabel(%)	Test(%)	Unlabel(%)	Test(%)	Unlabel(%)	Test(%)
UMIST	RLDA	-	-	79.1±4.2	79.6±5.1	90.8±3.3	90.7±3.1
	GFHF	75.1±4.5	-	83.3±4.1	-	91.1±3.5	-
	LGC	76.8±3.9	-	84.3±3.9	-	89.9±3.4	-
	SDA	61.1±3.8	61.7±4.9	86.2±4.8	86.5±5.1	92.5±2.8	92.1±3.1
	LapRLS/L	68.5±4.2	70.1±5.1	86.6±4.9	86.5±5.2	92.7±2.5	92.5±2.7
	VLR	75.8±4.5	75.5±4.7	87.3±3.5	87.5±4.1	93.5±2.9	93.1±2.5
dataset	method	1 labeled sample		3 labeled samples		5 labeled samples	
		Unlabel(%)	Test(%)	Unlabel(%)	Test(%)	Unlabel(%)	Test(%)
COIL-20	RLDA	-	-	78.8±1.9	78.5±2.6	85.6±1.3	85.3±1.6
	GFHF	84.9±2.3	-	87.9±2.1	-	90.3±1.5	-
	LGC	86.9±1.5	-	88.2±2.1	-	90.9±1.1	-
	SDA	61.4±2.6	60.9±3.9	80.2±1.9	80.1±2.5	87.2±2.2	87.1±2.3
	LapRLS/L	63.2±3.1	63.1±3.7	80.9±2.5	80.6±2.4	86.1±1.8	85.9±2.2
	VLR	84.3±2.1	83.5±2.7	88.2±1.8	86.8±2.3	90.6±1.2	89.9±1.1
dataset	method	1 labeled sample		3 labeled samples		5 labeled samples	
		Unlabel(%)	Test(%)	Unlabel(%)	Test(%)	Unlabel(%)	Test(%)
YALE-B	RLDA	-	-	64.3±2.3	64.5±2.5	81.8±2.2	81.9±2.8
	GFHF	27.6±3.3	-	46.9±2.6	-	58.5±2.9	-
	LGC	36.3±2.4	-	51.8±2.1	-	60.7±2.3	-
	SDA	34.3±2.1	33.8±2.5	84.2±2.7	83.6±2.8	93.4±1.5	93.3±2.1
	LapRLS/L	52.1±2.9	52.6±2.6	82.8±2.3	82.7±2.1	92.9±1.6	92.9±2.1
	VLR	52.7±2.7	52.5±3.2	86.1±2.4	85.8±2.2	93.9±1.5	94.3±1.8
dataset	method	1 labeled samples		3 labeled samples		5 labeled samples	
		Unlabel(%)	Test(%)	Unlabel(%)	Test(%)	Unlabel(%)	Test(%)
CMU PIE	RLDA	-	-	83.4±1.8	83.3±1.7	91.5±1.1	91.2±1.0
	GFHF	41.3±2.9	-	56.6±1.9	-	67.9±1.8	-
	LGC	43.5±2.8	-	57.6±1.3	-	67.7±1.7	-
	SDA	58.9±3.3	58.7±3.8	89.2±1.2	88.8±1.3	93.8±0.8	93.5±0.9
	LapRLS/L	61.5±2.9	61.6±3.3	89.5±1.2	88.9±1.1	94.1±0.8	93.7±1.0
	VLR	61.9±2.8	61.8±3.1	89.8±1.2	89.3±1.3	94.2±0.8	93.9±0.9

and LGC perform very bad. While our method VLR was not affected too much in this case due to the effectiveness of the proposed label propagation method to detect the outliers in data.

C. Performance Analysis on the Parameters

In this experiment, we explore how the parameters of algorithms influence the performance. The experimental setting is the same as the experiment in Section IV-B. The parameters setting are the same as the previous experiments except the specific parameter to be explored. In the training data set, three samples per class are randomly selected as the labeled data and remain the other samples as the unlabeled data. The mean accuracies of the training unlabeled data and the test data with different parameters are reported in this experiment.

1) Performance Analysis on the Regularization Parameter: In RLDA, SDA, LapRLS/L, and VLR, the Tikhonov regularization [39] term $\gamma tr\mathbf{W}^T\mathbf{W}$ is used to improve the generalization performance. The experimental results are shown in Fig. 2. We can see that the performance of RLDA and VLR are not influenced too much when the regularization parameter γ is

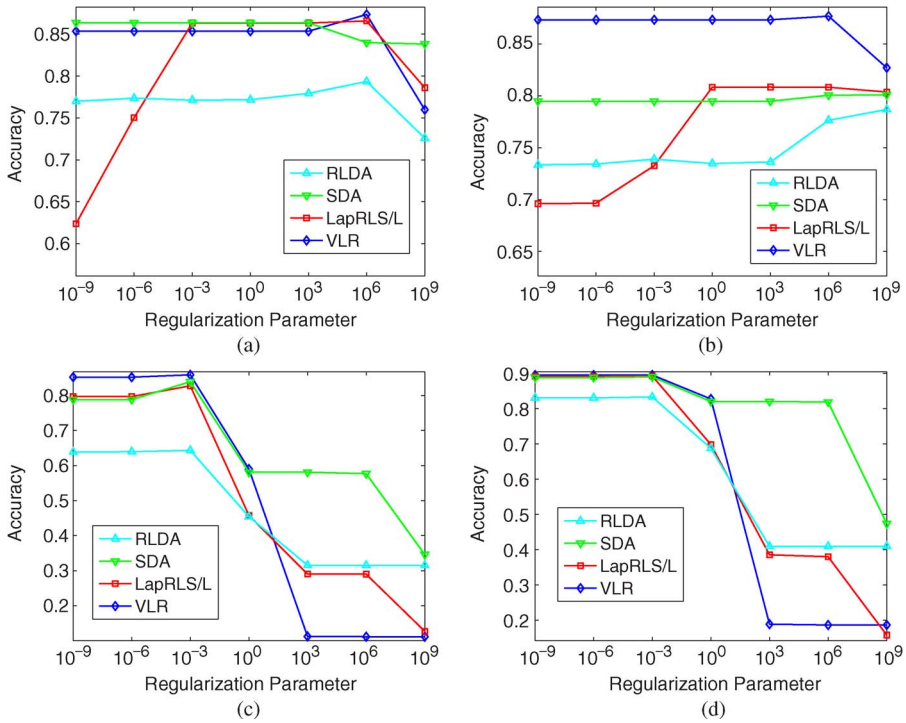

Fig. 2. Accuracy versus the regularization parameter in RLDA, SDA, LapRLS/L, and VLR on the four databases. (a) UMIST. (b) COIL-20. (c) YALE-B. (d) CMU PIE.

small, while SDA and LapRLS/L might not perform well when γ is small and thus should be carefully tuned in the application.

2) Performance Analysis on the Parameter to Control the Unlabeled Data: In semisupervised dimensionality reduction methods SDA and LapRLS/L, there is one parameter used to control the tradeoff between the labeled and the unlabeled data. In our method, VLR, the parameter α in (2) is also a parameter to control the effect of unlabeled data on the label regression procedure. If $\alpha \to 0$, then $F_{ij}(j <= c) \to 0$; and thus, the unlabeled data will have little effect on the label regression function in (8). If $\alpha \to 1$, then $F_{ij}(j = c+1) \to 0$; and thus, the virtual labels of the unlabeled data will play an important role on the label regression function in (8).

In the experiment, we explore the performance of SDA and LapRLS/L with different values of the parameter ranged from 10^{-9} to 10^{9} and the performance of VLR with different values of the parameter α ranged from 0 to 1. The experimental results are shown in Fig. 3. Note that the values of the abscissa for VLR in the figure are ranged from {0.1, 0.3, 0.6, 0.9, 0.9999, 0.9999999, 0.9999999999}. From the results, we can see that the parameter in SDA, LapRLS/L, and VLR has significant influence on the performance. Interestingly, we find that the performance is better when the parameter α is larger on the data sets UMIST and COIL-20, while on the data sets YALE-B and CMU PIE, the performance is better when the parameter α is smaller. From the results of GFHF and LGC, we can infer that the data sets UMIST and COIL-20 have clear manifold structures while the data sets YALE-B and CMU PIE do not have. In practice, we can set the parameter α in VLR to be a large value near one if the data has a clear manifold structure and set it to be a small value near zero if the manifold structure of data is not very clear.

3) Performance Analysis on the Parameters to Construct the Graph: In the semisupervised dimensionality reduction methods SDA, LapRLS/L, and VLR, there are two parameters that need to be predefined to construct the graph, including the number of neighbors k and the σ in (1). In the experiment, we explore the influence of these two parameters on performance. The results on these two parameters are shown in Figs. 4 and 5, respectively. The baseline value σ_0 in Fig. 5 is calculated by (19) where $\tau = 1$. From the results, we can see that the parameter k has a relatively small influence on the performance in a large range from 10 to 50, while the the parameter σ has a relatively significant influence on the performance. Generally, the performance would be well when the value of σ belongs to [0.1, 0.4] in all the three methods.

D. Performance Analysis on the Number of Unlabeled Data

An effective semisupervised method would improve the performance when the number of the available unlabeled data increases. To explore how the semisupervised dimensionality reductions utilize the unlabeled data, in this experiment, we verify the performance of SDA, LapRLS/L, and VLR on different numbers of unlabeled data. The experimental and the parameter settings are the same as the experiment in Section IV-B, except the split percentage between the training and the test data set. In the training data set, three samples per class are randomly selected as the labeled data, and the other samples are remained as the unlabeled data. The mean accuracies of the training unlabeled and the test data with different parameters are shown in Fig. 6.

From the experimental results, we can see that the performances of all the three semisupervised dimensionality

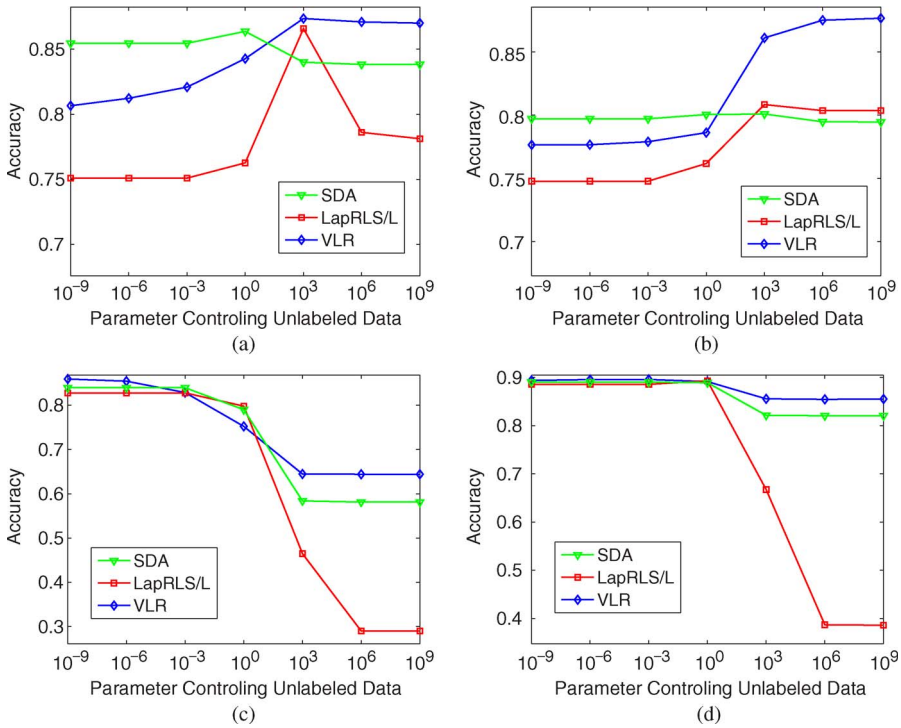

Fig. 3. Accuracy versus the parameter that controls the tradeoff between the labeled and the unlabeled data in SDA, LapRLS/L, and VLR on the four databases. Note that the values of the abscissa for VLR in the figure are ranged from {0.1, 0.3, 0.6, 0.9, 0.9999, 0.9999999, 0.9999999999}. (a) UMIST. (b) COIL-20. (c) YALE-B. (d) CMU PIE.

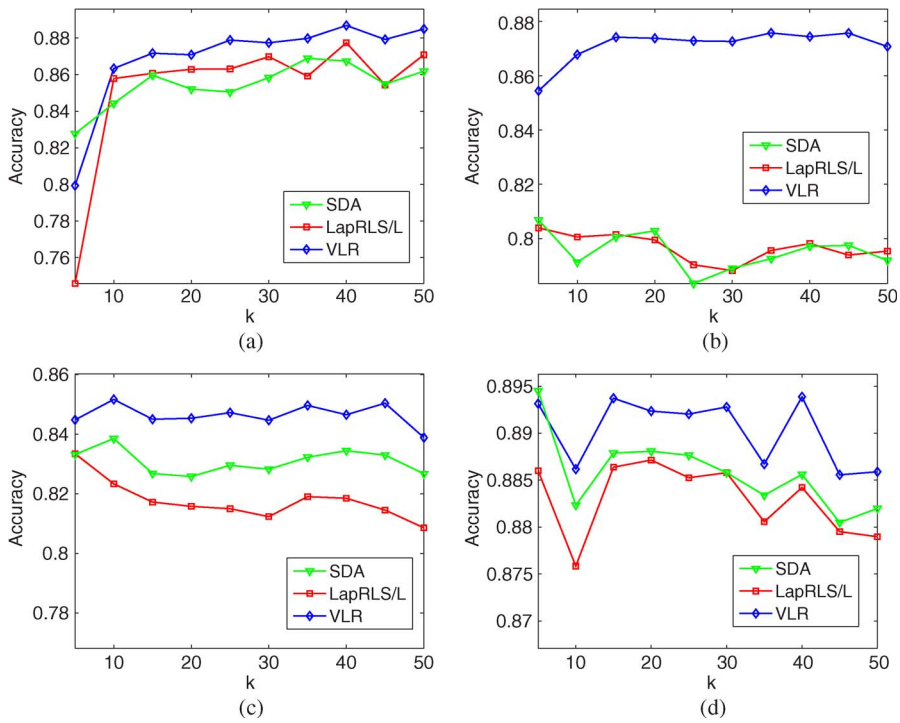

Fig. 4. Accuracy versus the number of neighbors k in SDA, LapRLS/L, and VLR on the four databases. (a) UMIST. (b) COIL-20. (c) YALE-B. (d) CMU PIE.

reduction methods are improved when the number of the available unlabeled data increases, which indicates that semisupervised dimensionality reduction method could indeed improve the performance when using the unlabeled data. The results also show that the improvement speed of our VLR method is much faster than that of SDA and LapRLS/L, which indicates that the proposed VLR method could utilize the unlabeled data more effectively. The improvement speeds of VLR on the data sets UMIST and COIL-20 are much faster than on the data sets YALE-B and CMU PIE, which further verifies that the manifold structures of UMIST and COIL-20 are clearer than those of YALE-B and CMU PIE.

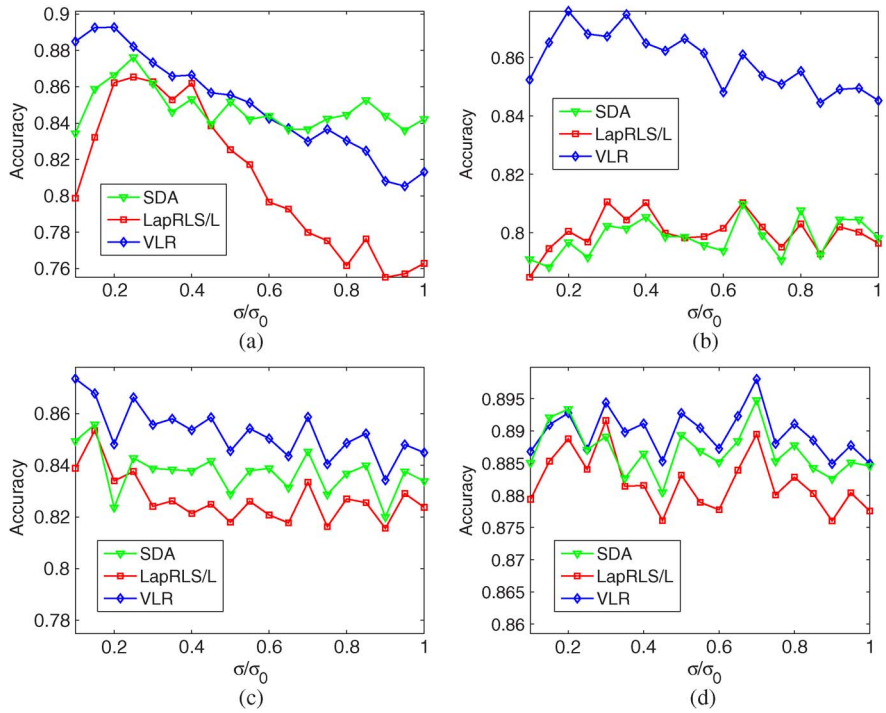

Fig. 5. Accuracy versus the σ in (1) in SDA, LapRLS/L, and VLR on the four databases. The baseline value σ_0 is calculated by (19) where $\tau = 1$. (a) UMIST. (b) COIL-20. (c) YALE-B. (d) CMU PIE.

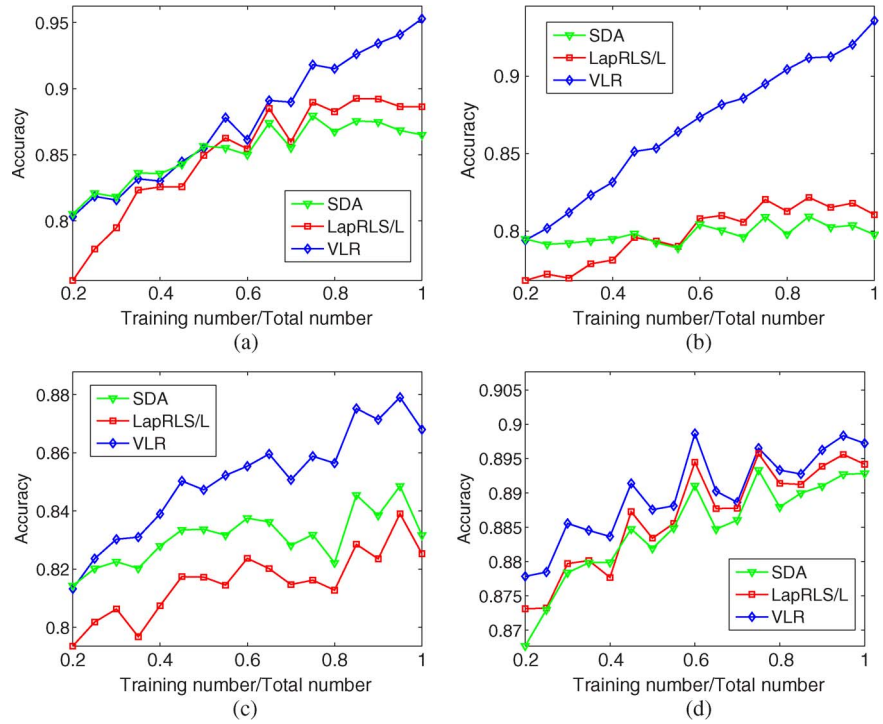

Fig. 6. Accuracy versus the number of training data (including the labeled and the unlabeled data) in SDA, LapRLS/L, and VLR on the four databases. (a) UMIST. (b) COIL-20. (c) YALE-B. (d) CMU PIE.

V. Conclusion

In this paper, we have proposed an efficient semisupervised dimensionality reduction algorithm through label propagation and regression. The basic idea is different from the previous methods, and the distribution of labeled and unlabeled data is effectively explored by using a special designed label propagation procedure, in which the outlier in the data can be effectively detected and the obtained virtual labels of unlabeled data can be easily encoded in a weighted regression model. The projection matrix for dimensionality reduction can be efficiently calculated by the label regression model. Experimental results on several image databases demonstrated the advantage of the proposed method over the state-of-the-art algorithms.

REFERENCES

[1] Y. Pang, Y. Yuan, and X. Li, "Gabor-based region covariance matrices for face recognition," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 18, no. 7, pp. 989–993, Jul. 2008.

[2] D. Dai and P. C. Yuen, "Face recognition by regularized discriminant analysis," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 37, no. 4, pp. 1080–1085, Aug. 2007.

[3] X. Li, S. Lin, S. Yan, and D. Xu, "Discriminant locally linear embedding with high-order tensor data," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 38, no. 2, pp. 342–352, Apr. 2008.

[4] Y. Pang, Y. Yuan, and X. Li, "Effective feature extraction in high dimensional space," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 38, no. 6, pp. 1652–1656, Dec. 2008.

[5] Y. Yang, F. Wu, D. Xu, Y. Zhuang, and L.-T. Chia, "Cross-media retrieval using query dependent search methods," *Pattern Recognit.*, vol. 43, no. 8, pp. 2927–2936, Aug. 2010.

[6] R. Dudoit, J. Fridly, and T. P. Speed, "Comparison of discrimination methods for the classification of tumors using gene expression data," *J. Amer. Stat. Assoc.*, vol. 97, no. 457, pp. 77–87, Mar. 2002.

[7] S. Xiang, F. Nie, C. Zhang, and C. Zhang, "Interactive natural image segmentation via spline regression," *IEEE Trans. Image Process.*, vol. 18, no. 7, pp. 1623–1632, Jul. 2009.

[8] Y. Pang, D. Tao, Y. Yuan, and X. Li, "Binary two-dimensional PCA," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 38, no. 4, pp. 1176–1180, Aug. 2008.

[9] Y. Pang, Y. Yuan, and X. Li, "Iterative subspace analysis based on feature line distance," *IEEE Trans. Image Process.*, vol. 18, no. 4, pp. 903–907, Apr. 2009.

[10] Y. Yuan, X. Li, Y. Pang, X. Lu, and D. Tao, "Binary sparse nonnegative matrix factorization," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 5, pp. 772–777, May 2009.

[11] X. Li and Y. Pang, "Deterministic column-based matrix decomposition," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 1, pp. 145–149, Jan. 2010.

[12] X. Li, Y. Pang, and Y. Yuan, "L1-norm-based 2DPCA," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 40, no. 4, pp. 1170–1175, Aug. 2010, DOI: 10.1109/TSMCB.2009.2035629.

[13] K. Fukunaga, *Introduction to Statistical Pattern Recognition*, 2nd ed. Boston, MA: Academic, 1990.

[14] X. J. Zhu, "Semi-supervised learning literature survey," Dept. Comput. Sci., Univ. Wisconsin-Madison, Madison, Tech. Rep. 1530, 2007.

[15] X. Zhu, Z. Ghahramani, and J. D. Lafferty, "Semi-supervised learning using Gaussian fields and harmonic functions," in *Proc. ICML*, 2003, pp. 912–919.

[16] D. Zhou, O. Bousquet, T. N. Lal, J. Weston, and B. Schölkopf, "Learning with local and global consistency," in *Proc. NIPS*, 2004, pp. 321–328.

[17] Y. Bengio, J.-F. Paiement, P. Vincent, O. Delalleau, N. L. Roux, and M. Ouimet, "Out-of-sample extensions for LLE, ISOMAP, MDS, eigenmaps, and spectral clustering," in *Proc. NIPS*, 2003, pp. 177–184.

[18] D. Zhang, Z.-H. Zhou, and S. Chen, "Semi-supervised dimensionality reduction," in *Proc. SDM*, 2007, pp. 629–634.

[19] D. Cai, X. He, and J. Han, "Semi-supervised discriminant analysis," in *Proc. ICCV*, 2007, pp. 1–7.

[20] Y. Song, F. Nie, C. Zhang, and S. Xiang, "A unified framework for semi-supervised dimensionality reduction," *Pattern Recognit.*, vol. 41, no. 9, pp. 2789–2799, Sep. 2008.

[21] M. Sugiyama, T. Idé, S. Nakajima, and J. Sese, "Semi-supervised local fisher discriminant analysis for dimensionality reduction," in *Proc. PAKDD*, 2008, pp. 333–344.

[22] Y. Song, F. Nie, and C. Zhang, "Semi-supervised sub-manifold discriminant analysis," *Pattern Recognit. Lett.*, vol. 29, no. 13, pp. 1806–1813, Oct. 2008.

[23] F. Wu, W. Wang, Y. Yang, Y. Zhuang, and F. Nie, "Classification by semi-supervised discriminative regularization," *Neurocomputing*, vol. 73, no. 10–12, pp. 1641–1651, Jun. 2010.

[24] C. Hou, C. Zhang, Y. Wu, and F. Nie, "Multiple view semi-supervised dimensionality reduction," *Pattern Recognit.*, vol. 43, no. 3, pp. 720–730, Mar. 2010.

[25] M. Belkin, P. Niyogi, and V. Sindhwani, "Manifold regularization: A geometric framework for learning from labeled and unlabeled examples," *J. Mach. Learn. Res.*, vol. 7, pp. 2399–2434, Dec. 2006.

[26] S. Yan, D. Xu, B. Zhang, H. Zhang, Q. Yang, and S. Lin, "Graph embedding and extensions: A general framework for dimensionality reduction," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 1, pp. 40–51, Jan. 2007.

[27] S. T. Roweis and E. Al, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, Dec. 2000.

[28] Z. Zhang and H. Zha, "Principal manifolds and nonlinear dimension reduction via local tangent space alignment," *SIAM J. Sci. Comput.*, vol. 26, no. 1, pp. 313–338, 2005.

[29] S. Xiang, F. Nie, C. Zhang, and C. Zhang, "Nonlinear dimensionality reduction with local spline embedding," *IEEE Trans. Knowl. Data Eng.*, vol. 21, no. 9, pp. 1285–1298, Sep. 2009.

[30] C. Zhang, F. Nie, and S. Xiang, "A general kernelization framework for learning algorithms based on kernel PCA," *Neurocomputing*, vol. 73, no. 4–6, pp. 959–967, Jan. 2010.

[31] C. C. Paige and M. A. Saunders, "Algorithm 583, LSQR: Sparse linear equations and least-squares problems," *ACM Trans. Math. Softw.*, vol. 8, no. 2, pp. 195–209, Jun. 1982.

[32] D. Cai, "Spectral regression: A regression framework for efficient regularized subspace learning," Ph.D. dissertation, Dept. Comput. Sci., Univ. Illinois Urbana-Champaign, Urbana, May, 2009.

[33] J. H. Friedman, "Regularized discriminant analysis," *J. Amer. Stat. Assoc.*, vol. 84, no. 405, pp. 165–175, Mar. 1989.

[34] V. Sindhwani, P. Niyogi, M. Belkin, and S. Keerthi, "Linear manifold regularization for large scale semi-supervised learning," in *Proc. ICML Workshop*, 2005, pp. 80–83.

[35] D. B. Graham and N. M. Allinson, "Characterizing virtual eigensignatures for general purpose face recognition," in *Face Recognition: From Theory to Applications*, vol. 163, *NATO ASI Series F, Computer and Systems Sciences*. Berlin, Germany: Springer-Verlag, 1998, pp. 446–456.

[36] S. A. Nene, S. K. Nayar, and H. Murase, "Columbia object image library (COIL-20)," Columbia Univ., New York, Tech. Rep. CUCS-005-96, 1996.

[37] A. Georghiades, P. Belhumeur, and D. Kriegman, "From few to many: Illumination cone models for face recognition under variable lighting and pose," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 6, pp. 643–660, Jun. 2001.

[38] T. Sim and S. Baker, "The CMU pose, illumination, and expression database," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 12, pp. 1615–1618, Dec. 2003.

[39] A. N. Tikhonov and V. Y. Arsenin, *Solutions of Ill-Posed Problems*. New York: Wiley, 1977.

Feiping Nie received the B.S. degree in computer science from the North China University of Water Conservancy and Electric Power, Zhengzhou, China, in 2000, the M.S. degree in computer science from Lanzhou University, Lanzhou, China, in 2003, and the Ph.D. degree in computer science from Tsinghua University, Beijing, China, in 2009.

His research interests include machine learning, pattern recognition, data mining, and image processing.

Dong Xu received the B.Eng. and Ph.D. degrees from the University of Science and Technology of China, Hefei, China, in 2001 and 2005, respectively.

During his Ph.D. study, he worked at Microsoft Research Asia and The Chinese University of Hong Kong, Shatin, Hong Kong for more than two years. He also worked at Columbia University, New York, NY, for one year as a Postdoctoral Research Scientist. He is currently an Assistant Professor with Nanyang Technological University, Singapore. Since 2008, he has received over $2.8M in research grant funding, from Singapore National Research Foundation, A*STAR, Ministry of Education, and Microsoft Research Asia. He has published more than 35 papers in top venues including T-PAMI, T-IP, T-CSVT, CVPR, ACM MM, ICML, and IJCAI. His publications have been cited in Google Scholar more than 900 times, and his H-Index in Google Scholar is 15. His research focuses on new theories, algorithms, and systems for intelligent processing and understanding of visual data, such as images and videos.

Dr. Xu is an Associate Editor of *Neurocomputing* (Elsevier), and he is an editorial board Member of the *Journal of Multimedia* (Academy). He is currently the Guest Editor of a forthcoming special issue on Social Media in ACM TOMCCAP. He has also been the Guest Editor of three special issues on video and event analysis in T-CSVT, CVIU, and PRL. He has been the workshop Cochair of The ACM SIGMM Workshop on Social Media and The ICME Workshop on Visual Content Identification and Search, a Track Chair of ICME 2009, and a Theme Chair of PSIVT 2009. Moreover, he has regularly served on the program committees of the major computer vision and multimedia conferences including ICCV, CVPR, ECCV, and ACM MM. He was coauthor (with his Ph.D. student Lixin Duan) of a paper that won the Best Student Paper Award in the prestigious IEEE International Conference on Computer Vision and Pattern Recognition (CVPR 2010). Ph.D. students Yi Huang and Lixin Duan in his research group were awarded the prestigious MSRA Fellowship Awards in 2008 and 2009, respectively.

Xuelong Li (M'02–SM'07) is a Researcher (Full Professor) with the State Key Laboratory of Transient Optics and Photonics and the Director of the Center for OPTical IMagery Analysis and Learning (OPTIMAL), Xi'an Institute of Optics and Precision Mechanics, Chinese Academy of Sciences, Xi'an, China.

Shiming Xiang received the B.S. degree from the Department of Mathematics, Chongqing Normal University, Chongqing, China, in 1993, the M.S. degree from the Department of Mechanics and Mathematics, Chongqing University, Chongqing, in 1996, and the Ph.D. degree from the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China, in 2004.

He was a Postdoctoral Fellow with the Department of Automation, Tsinghua University, Beijing, China, until 2006. He is currently an Associate Researcher with the Institute of Automation, Chinese Academy of Science, Beijing, China. His interests include computer vision, pattern recognition, machine learning, etc.