# Flexible Manifold Embedding: A Framework for Semi-Supervised and Unsupervised Dimension Reduction

Feiping Nie, Dong Xu, *Member, IEEE*, Ivor Wai-Hung Tsang, and Changshui Zhang, *Member, IEEE*

***Abstract*—We propose a unified manifold learning framework for semi-supervised and unsupervised dimension reduction by employing a simple but effective linear regression function to map the new data points. For semi-supervised dimension reduction, we aim to find the optimal prediction labels $F$ for all the training samples $X$, the linear regression function $h(X)$ and the regression residue $F_0 = F - h(X)$ simultaneously. Our new objective function integrates two terms related to label fitness and manifold smoothness as well as a flexible penalty term defined on the residue $F_0$. Our Semi-Supervised learning framework, referred to as flexible manifold embedding (FME), can effectively utilize label information from labeled data as well as a manifold structure from both labeled and unlabeled data. By modeling the mismatch between $h(X)$ and $F$, we show that FME relaxes the hard linear constraint $F = h(X)$ in manifold regularization (MR), making it better cope with the data sampled from a nonlinear manifold. In addition, we propose a simplified version (referred to as FME/U) for unsupervised dimension reduction. We also show that our proposed framework provides a unified view to explain and understand many semi-supervised, supervised and unsupervised dimension reduction techniques. Comprehensive experiments on several benchmark databases demonstrate the significant improvement over existing dimension reduction algorithms.**

***Index Terms*—Dimension reduction, face recognition, manifold embedding, semi-supervised learning.**

## I. Introduction

In PAST decades, a large number of dimension reduction techniques [2], [13], [25], [29], [30], [35] have been proposed. Principal component analysis (PCA) [25] pursues the directions of maximum variance for optimal reconstruction. Linear discriminant analysis (LDA) [2], as a supervised algorithm, aims to maximize the inter-class scatter and at the same time minimize the intra-class scatter. Due to the utilization of label information, LDA is experimentally reported to outperform PCA for face recognition, when sufficient labeled face images are provided [2].

To discover the intrinsic manifold structure of the data, nonlinear dimension reduction algorithms such as ISOMAP [23], locally linear embedding (LLE) [18] and Laplacian eigenmap (LE) [3] were recently developed. However, ISOMAP and LE suffer from the so-called out-of-sample problem, i.e., they do not yield a method for mapping new data points that are not included in the training set. To deal with this problem, He *et al.* [12] developed the locality preserving projections (LPP) method, in which the linear projection function is used for mapping new data points. Wu *et al.* [27] proposed a local learning algorithm, referred to as local learning projection (LLP) for linear dimension reduction. Yan *et al.* [30] recently demonstrated that several dimension reduction algorithms (e.g., PCA, LDA, ISOMAP, LLE, LE) can be unified within a proposed graph-embedding framework, in which the desired statistical or geometric data properties are encoded as graph relationships. Recently, Zhang *et al.* [34], [35] further reformulated many dimension reduction algorithms into a unified patch alignment framework. Based on their patch alignment framework, a new subspace learning method called Discriminative Locality Alignment (DLA) was also proposed [34], [35].

While supervised learning algorithms generally outperform unsupervised learning algorithms, the collection of labeled training data in supervised learning requires expensive human labor [8], [38]. Meanwhile, it is much easier to obtain unlabeled data. To utilize a large amount of unlabeled data as well as a relatively limited amount of labeled data for better classification, semi-supervised learning methods such as transductive SVM [26], co-training [5], and graph-based techniques [1], [4], [6], [20], [21], [31], [33], [28], [36], [37] were developed and demonstrated promising results for different tasks. However, most semi-supervised learning methods such as [5], [11], [26], [36], [37] were developed for the problem of classification. The manifold regularization (MR) method [4], [20], [21] can be also used for various learning problems. In practice, MR extended regression and SVM, respectively, to the semi-supervised learning methods Laplacian regularized least squares (LapRLS) and Laplacian support vector machines (LapSVM) by adding a geometrically-based regularization term. Recently, Cai *et al.* [6] extended LDA to semi-supervised discriminant analysis (SDA), and Zhang *et al.* [34] extended DLA to

Manuscript received February 10, 2009; revised January 04, 2010. First published March 08, 2010; current version published June 16, 2010. This work was supported by the Singapore National Research Foundation Interactive Digital Media R&D Program, under research Grant NRF2008IDM-IDM004-018, MOE AcRF Tier-1 Grant (RG63/07) and China 973 Program (2009CB320602) and NSFC Grant (60721003). The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Xuelong Li.

F. Nie is with the School of Computer Engineering, Nanyang Technological University, 639798 Singapore and also with the State Key Laboratory of Intelligent Technologies and Systems Tsinghua National Laboratory for Information Science and Technology (TNList), Department of Automation, Tsinghua University, Beijing 100084, China.

D. Xu and I. Wai-Hung Tsang are with the School of Computer Engineering, Nanyang Technological University, 639798 Singapore (e-mail: dongxu@ntu.edu.sg).

C. Zhang is with the State Key Laboratory of Intelligent Technologies and Systems Tsinghua National Laboratory for Information Science and Technology (TNList), Department of Automation, Tsinghua University, Beijing 100084, China.

Digital Object Identifier 10.1109/TIP.2010.2044958

semi-supervised discriminative locality alignment (SDLA), for semi-supervised dimension reduction.

Many dimension reduction algorithms (e.g., PCA, LDA, LPP, and SDA) use a linear projection function to map the data matrix $X$ in the original feature space to a lower dimensional representation $F$, namely, $F = X^T W$. The low dimensional representation can then be used for faster training and testing in real applications, as well as the interpretation of the data. In this work, we first show that the MR method linear LapRLS (referred to as LapRLS/L) can also utilize a linear function $h(X)$ to connect the prediction labels $F$ and the data matrix $X$ by[1] $F = h(X) = X^T W$. While the linearization techniques provide a simple and effective method to map new data points, we argue that such techniques assume that the lower dimension representation or the prediction labels $F$ lie in the space spanned by the training samples $X$, which is usually overstrict in many real applications.

The prior work [1], [33] employed a regression residue term to relax the hard constraint $F = h(X)$ for binary classification. Inspired by their work [1], [33], we propose a new manifold learning framework for dimension reduction in multi-class setting and our framework naturally unifies many existing dimension reduction methods. Specifically, we set the prediction labels as $F = h(X) + F_0$, where $h(X)$ is a regression function for mapping new data points and $F_0$ is the regression residue modeling the mismatch between $F$ and $h(X)$. With this model, we propose a new framework, referred to as flexible manifold embedding (FME), for semi-supervised dimension reduction. In practice, we aim to find the optimal prediction labels $F$, the linear regression function $h(X)$ and the regression residue $F_0$ of our new objective function simultaneously, which integrates two terms related to the label fitness and the manifold smoothness as well as a flexible penalty term $\|F_0\|^2$. FME can effectively utilize label information from labeled data as well as the manifold structure from both labeled and unlabeled data. We also show that our FME relaxes the hard linear constraint $F = h(X)$ in LapRLS/L. With this relaxation, FME can better deal with the samples which reside on a nonlinear manifold. We also propose a simplified version, referred to as FME/U, for unsupervised manifold learning. It is worth mentioning that FME and FME/U are linear methods, which are fast and suitable for practical applications such as face, object and text classification problems.

The main contributions of this paper include the following.

- We propose a unified framework for semi-supervised and unsupervised manifold learning, which can provide a mapping for new data points and effectively cope with the data sampled from the nonlinear manifold.
- Our proposed framework provides a unified view to explain and understand many semi-supervised, supervised, and unsupervised dimension reduction techniques.
- Our work outperforms existing dimension reduction methods on five benchmark databases, demonstrating promising performance in real applications.

The rest of this paper is organized as follows. Section II gives a brief review of the related dimension reduction methods. We will introduce our proposed framework for semi-supervised and unsupervised dimension reduction in Sections III and IV, respectively. Discussions with other related work are presented in Section V. Comprehensive experimental results are reported in Section VI. Section VII gives conclusive remarks.

[1]Here we ignore the bias term of the linear regression function in LapRLS/L.

## II. Brief Review of the Prior Work

We briefly review the prior semi-supervised learning work: local and global consistency (LGC) [36], Gaussian fields and harmonic functions (GFHF) [37], manifold regularization (MR) [4], [20], [21] and semi-supervised discriminant analysis (SDA) [6]. We denote the sample set as $X = [x_1, x_2, \ldots, x_n, x_{n+1}, \ldots, x_m] \in \mathbb{R}^{f \times m}$, where $x_i|_{i=1}^n$ and $x_i|_{i=n+1}^m$ are labeled and unlabeled data, respectively. For labeled data $x_i|_{i=1}^n$, the labels are denoted as $y_i \in \{1, 2, \ldots, c\}$, where $c$ is the total number of classes. We also define a binary label matrix $Y \in \mathbb{B}^{m \times c}$ with $Y_{ij} = 1$ if $x_i$ has label $y_i = j$; $Y_{ij} = 0$, otherwise. Let us denote $G = \{X, S\}$ as an undirected weighted graph with vertex set $X$ and similarity matrix $S \in \mathbb{R}^{m \times m}$, in which each element $S_{ij}$ of the symmetric matrix $S$ represents the similarity of a pair of vertices. The graph Laplacian matrix $L \in \mathbb{R}^{m \times m}$ is denoted as $L = D - S$, where $D$ is a diagonal matrix with the diagonal elements as $D_{ii} = \sum_j S_{ij}, \forall i$. The normalized graph Laplacian matrix is represented as $\tilde{L} = D^{-(1/2)} L D^{-(1/2)} = I - D^{-(1/2)} S D^{-(1/2)}$, where $I$ is an identity matrix. We also denote $\mathbf{0}, \mathbf{1} \in \mathbb{R}^{m \times 1}$ as a vector with all elements as 0 and a vector with all elements as 1, respectively.

### A. LGC and GFHF

LGC [36] and GFHF [37] estimate a prediction label matrix $F \in \mathbb{R}^{m \times c}$ on the graph with respect to the label fitness (i.e., $F$ should be close to the given labels for the labeled nodes) and the manifold smoothness (i.e., $F$ should be smooth on the whole graph of both labeled and unlabeled nodes). Let us denote $F_{i.}$ and $Y_{i.}$ as the $i$th row of $F$ and $Y$. As shown in [36]–[38], LGC and GFHF minimize the objective function $g_L(F)$ and $g_G(F)$, respectively

$$g_L(F) = \frac{1}{2} \sum_{i,j=1}^{m} \left\| \frac{F_{i.}}{\sqrt{D_{ii}}} - \frac{F_{j.}}{\sqrt{D_{jj}}} \right\|^2 S_{ij} + \lambda \sum_{i=1}^{n} \|F_{i.} - Y_{i.}\|^2$$

$$g_G(F) = \frac{1}{2} \sum_{i,j=1}^{m} \|F_{i.} - F_{j.}\|^2 S_{ij} + \lambda_\infty \sum_{i=1}^{n} \|F_{i.} - Y_{i.}\|^2 \qquad (1)$$

where the coefficient $\lambda$ balances the label fitness and the manifold smoothness, and $\lambda_\infty$ is a very large number such that $\sum_{i=1}^n \|F_{i.} - Y_{i.}\|^2 = 0$, or $F_{i.} = Y_{i.}\ \forall i = 1, 2, \ldots, n$ [38]. Notice that the objective functions $g_L(F)$ and $g_G(F)$ in (1) share the same formulation

$$Tr(F^T M F) + Tr(F - Y)^T U (F - Y) \qquad (2)$$

where $M \in \mathbb{R}^{m \times m}$ is a (normalized) graph Laplacian matrix and $U \in \mathbb{R}^{m \times m}$ is a diagonal matrix.

In LGC [36], $M$ is the normalized graph Laplacian matrix $\tilde{L}$ and $U$ is a diagonal matrix with all elements as $\lambda$. In GFHF [37], $M = L$ and $U$ is also a diagonal matrix with the first $n$ and the rest $m - n$ diagonal elements as $\lambda_\infty$ and 0, respectively.

### B. Manifold Regularization

The MR [4], [20], [21] extends many existing algorithms, such as ridge regression and SVM to their semi-supervised learning methods by adding a geometrically based regularization term. We take LapRLS/L as an example to briefly review MR methods. Let us define a linear regression function $h(x_i) = W^T x_i + b$, where $W \in \mathbb{R}^{f\times c}$ is the projection matrix and $b \in \mathbb{R}^{c\times 1}$ is the bias term. LapRLS/L [21] minimizes the ridge regression errors and simultaneously preserves the manifold smoothness, namely

$$g_M(W,b) = \lambda_A\|W\|^2 + \lambda_I Tr(W^T X L X^T W) + \frac{1}{n}\sum_{i=1}^{n}\left\|W^T x_i + b - Y_{i\cdot}^T\right\|^2 \quad (3)$$

where the two coefficients $\lambda_A$ and $\lambda_I$ balance the norm of $W$, the manifold smoothness and the regression error.

### C. Semi-Supervised Discriminant Analysis

Cai *et al.* extended LDA to SDA [6] by adding a geometrically-based regularization term in the objective function of LDA. The core assumption in SDA is still the manifold smoothness assumption, namely, nearby points will have similar representations in the lower-dimensional space. We define $X_l = [x_1, x_2, \ldots, x_n]$ as the data matrix of labeled data, and denote the number of the labeled samples in the $i$th class as $n_i$. Let us denote two graph similarity matrices $\tilde{S}^w, \tilde{S}^b \in \mathbb{R}^{n\times n}$, where $\tilde{S}^w_{ij} = \delta_{y_i,y_j}/n_{y_i}, \tilde{S}^b_{ij} = (1/n) - \tilde{S}^w_{ij}$. The corresponding Laplacian matrices of $\tilde{S}^w$, $\tilde{S}^b$ are represented as $\tilde{L}_w$ and $\tilde{L}_b$, respectively. According to [30], the intra-class scatter $S_w$ and the inter-class scatter $S_b$ of LDA can be rewritten as $S_w = \sum_{i=1}^{n}(x_i - \bar{x}_{y_i})(x_i - \bar{x}_{y_i})^T = X_l\tilde{L}_w X_l^T$, and $S_b = \sum_{l=1}^{c} n_c(\bar{x}_l - \bar{x})(\bar{x}_l - \bar{x})^T = X_l\tilde{L}_b X_l^T$, where $\bar{x}_l$ is the mean of the labeled samples in the $l$th class and $\bar{x}$ is the mean of all the labeled samples. The objective function in SDA is then formulated as

$$g_S(W) = \frac{\left|W^T X_l \tilde{L}_b X_l^T W\right|}{\left|W^T\left(X_l(\tilde{L}_w + \tilde{L}_b)X_l^T + \alpha X L X^T + \beta I\right)W\right|} \quad (4)$$

where $L \in \mathbb{R}^{m\times m}$ is the graph Laplacian matrix for both labeled and unlabeled data, and $\alpha$ and $\beta$ are two parameters to balance three terms.

## III. Semi-Supervised Flexible Manifold Embedding

It is noteworthy that existing MR work [1], [4], [20], [21], [33] are mainly on binary classification and regression problems only. In this paper, we focus on dimension reduction problems in multi-class setting. We firstly discuss the connection between LapRLS/L and LGC/GFHF. And then we propose a new manifold learning framework, referred to as FME, for semi-supervised dimension reduction.

### A. Connection Between LapRLS/L and LGC/GFHF

LGC [36] and GFHF [37] were proposed based on the motivations of label propagation and random walks, and LapRLS/L [21] was initially proposed as a semi-supervised extension for ridge regression. LGC/GFHF do not present a method for mapping new data points, and LapRLS/L can provide a mapping for unseen data points through the linear regression function $h(x)$. While LGC/GFHF and LapRLS/L are proposed from different motivations, we show that LapRLS/L is a varied out-of-sample extension of LGC/GFHF.

*Proposition 1:* LapRLS/L is a varied out-of-sample extension of LGC/GFHF, when a graph Laplacian matrix $M \in \mathbb{R}^{m\times m}$ satisfying $M\mathbf{1} = \mathbf{0}$ and $\mathbf{1}^T M = \mathbf{0}^T$ is used.

*Proof:* Suppose that the solution $F$ of LGC/GFHF is located in the linear subspace spanned by $X$, i.e., $F = h(X) = X^T W + \mathbf{1}b^T$, where $W \in \mathbb{R}^{f\times c}$ is the project matrix, $b \in \mathbb{R}^{c\times 1}$ is the bias term, then the objective function (2) in LGC/GFHF can be reformulated as

$$\mathrm{Tr}[(X^T W + \mathbf{1}b^T)^T M(X^T W + \mathbf{1}b^T)] + \mathrm{Tr}(X^T W + \mathbf{1}b^T - Y)^T U(X^T W + \mathbf{1}b^T - Y). \quad (5)$$

Then we add a regularization term $(\lambda_A)/(\lambda_I)\|W\|^2$ in (5) and set $M = L$, and the first $n$ and the rest $m-n$ diagonal elements of the diagonal matrix $U$ as $(1)/(n\lambda_A)$ and 0, respectively, it becomes

$$\frac{\lambda_A}{\lambda_I}\|W\|^2 + \mathrm{Tr}(W^T X L X^T W) + \frac{1}{n\lambda_I}\sum_{i=1}^{n}\left\|W^T x_i + b - Y_{i\cdot}^T\right\|^2 \quad (6)$$

which is equal to $(1)/(\lambda_I)g_M(W,b)$. So we have Proposition 1. ∎

### B. Flexible Manifold Learning Framework

From Proposition 1, we observe that the prediction labels $F$ in LapRLS/L are constrained to lie within the space spanned by all the training samples $X$, namely $F = X^T W + \mathbf{1}b^T$. While this linear function can be used to map new data points that are not included in the training set, the number of parameters in $W$ does not depend on the number of samples. Thereafter, this linear function may be overstrict to fit the data samples from a non-linear manifold. To better cope with this problem, we relax this hard constraint by modeling the regression residue. As shown in Fig. 1, we assume that $F = h(X) + F_0 = X^T W + \mathbf{1}b^T + F_0$, where $F_0 \in \mathbb{R}^{m\times c}$ is the regression residue modeling the mismatch between $F$ and $h(X)$. FME aims to find the optimal prediction labels $F$, the regression residue $F_0$, and the linear regression function $h(X)$ simultaneously

$$(F^*, F_0^*, W^*, b^*) = \arg\min_{F,F_0,W,b} \mathrm{Tr}(F-Y)^T U(F-Y) + \mathrm{Tr}(F^T M F) + \mu(\|W\|^2 + \gamma\|F_0\|^2) \quad (7)$$

where the two coefficients $\mu$ and $\gamma$ are parameters to balance different terms, and $M \in \mathbb{R}^{m\times m}$ is the Laplacian matrix and $U \in \mathbb{R}^{m\times m}$ is the diagonal matrix. Note that similar idea was also discussed in the prior work [1], [22], [24], [33] for binary classification problems. Here, we extend this idea for dimension reduction in multi-class setting, in which the class dependency can be captured by the extracted features.

Similarly as in LGC, GFHF, and LapRLS/L, the first two terms in (7) represent the label fitness and the manifold smoothness, respectively. Considering that it is meaningless to enforce

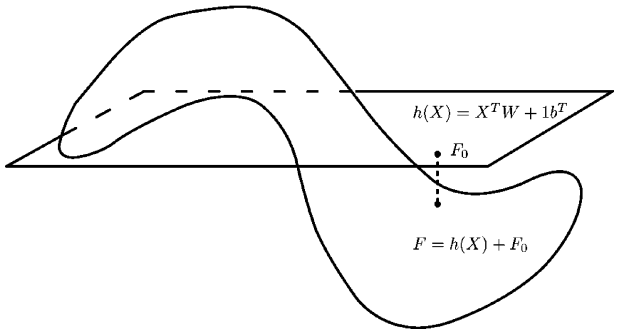



Fig. 1. Illustration of FME. FME aims to find the optimal prediction labels $F$, the linear regression function $h(X)$, and the regression residue $F_0$ simultaneously. The regression residue $F_0$ measures the mismatch between $F$ and $h(X)$.

the prediction labels $F_{i.}$ and the given labels $Y_{j.}$ of different samples (i.e., $j \neq i$) to be close, we set the matrix $U$ as the diagonal matrix with the first $n$ and the rest $m-n$ diagonal elements as 1 and 0, respectively, similarly as in LapRLS/L. In addition, the matrix $M$ should be set as the graph Laplacian matrix in order to utilize the manifold structure (i.e., $F$ should be as smooth as possible on the whole graph) in semi-supervised learning. While it is possible to construct the Laplacian matrix $M$ according to different manifold learning criterions (e.g., [30] and [32]). Similarly as in GFHF and LapRLS/L, we choose the Gaussian function to calculate $M$, namely, $M = D - S$, where $D$ is a diagonal matrix with the diagonal elements as $D_{ii} = \sum_j S_{ij}, \forall i$, and $S_{ij} = \exp(-\|x_i - x_j\|^2/t)$, if $x_i$ (or $x_j$) is among $k$ nearest neighbors of $x_j$ (or $x_i$); $S_{ij} = 0$, otherwise.

The last two terms in (7) control the norm of projection matrix $W$ and the regression residue $F_0$. In the current formulation of $F$, the regression function $h(X)$ and the regression residue $F_0$ are combined. In practice, our work can naturally map the new data points for dimension reduction by using the function $h(X)$. The regression residue $F_0$ can model the mismatch between the linear regression function $X^T W + \mathbf{1}b^T$ and the prediction labels $F$. Compared with LapRLS/L, we do not force the prediction labels $F$ to lie in the space spanned by all the samples $X$. Therefore, our framework is more flexible and it can better cope with the samples which reside on the nonlinear manifold. Moreover, the prior work [14] on face hallucination has demonstrated that the introduction of a local residue can lead to better reconstruction of face images.

Replacing $F_0$ with $F - X^T W - \mathbf{1}b^T$, we have

$$\begin{aligned}(F^*, W^*, b^*) &= \arg\min_{F,W,b} \mathrm{Tr}(F-Y)^T U(F-Y) + \mathrm{Tr}(F^T M F) \\ &\quad + \mu(\|W\|^2 + \gamma\|X^T W + \mathbf{1}b^T - F\|^2). \end{aligned} \tag{8}$$

From then on, we refer to the objective function in (8) as $g(F, W, b)$. First, we prove that the optimization problem in (8) is jointly convex with respect to $F, W$, and $b$.

*Theorem 1:* Denote $U, M \in \mathbb{R}^{m\times m}$, $F, Y \in \mathbb{R}^{m\times c}$, $W \in \mathbb{R}^{f\times c}$, $b \in \mathbb{R}^{c\times 1}$. If the matrices $U$ and $M$ are positive semi-definite, $\mu \geq 0$ and $\gamma \geq 0$, then $g(F, W, b) = \mathrm{Tr}(F-Y)^T U(F-Y) + \mathrm{Tr}(F^T M F) + \mu(\|W\|^2 + \gamma\|X^T W + \mathbf{1}b^T - F\|^2)$ is jointly convex with respect to $F$, $W$, and $b$.

*Proof:* In function $g(F, W, b)$, we remove the constant term $\mathrm{Tr}(Y^T U Y)$, then $g(F, W, b)$ can be rewritten in matrix form as

$$g(F,W,b) = \mathrm{Tr}\begin{bmatrix} F \\ W \\ b^T \end{bmatrix}^T P \begin{bmatrix} F \\ W \\ b^T \end{bmatrix} - \mathrm{Tr}\begin{bmatrix} F \\ W \\ b^T \end{bmatrix}^T \begin{bmatrix} 2UY \\ 0 \\ 0 \end{bmatrix}$$

where

$$P = \begin{bmatrix} \mu\gamma I + M + U & -\mu\gamma X^T & -\mu\gamma \mathbf{1} \\ -\mu\gamma X & \mu I + \mu\gamma X X^T & \mu\gamma X \mathbf{1} \\ -\mu\gamma \mathbf{1}^T & \mu\gamma \mathbf{1}^T X^T & \mu\gamma m \end{bmatrix}.$$

Thus in order to prove that $g(F, W, b)$ is jointly convex with respect to $F$, $W$, and $b$, we only need to prove that the matrix $P$ is positive semi-definite.

For any vector $z = [z_1^T, z_2^T, z_3]^T \in \mathbb{R}^{(m+f+1)\times 1}$, where $z_1 \in \mathbb{R}^{m\times 1}$ $z_2 \in \mathbb{R}^{f\times 1}$, and $z_3$ is a scalar, we have

$$\begin{aligned} z^T P z &= z_1^T(\mu\gamma I + M + U) z_1 - 2\mu\gamma z_1^T X^T z_2 - 2\mu\gamma z_1^T \mathbf{1} z_3 \\ &\quad + z_2^T(\mu I + \mu\gamma X X^T) z_2 + 2\mu\gamma z_2^T X \mathbf{1} z_3 + \mu\gamma m z_3^T z_3 \\ &= z_1^T(M+U) z_1 + \mu z_2^T z_2 + \mu\gamma\left(z_1^T z_1 - 2 z_1^T X^T z_2 \right. \\ &\quad \left. - 2 z_1^T \mathbf{1} z_3 + z_2^T X X^T z_2 + 2 z_2^T X \mathbf{1} z_3 + m z_3^T z_3\right) \\ &= z_1^T(M+U) z_1 + \mu z_2^T z_2 + \mu\gamma(z_1 - X^T z_2 - \mathbf{1} z_3)^T \\ &\quad \times (z_1 - X^T z_2 - \mathbf{1} z_3). \end{aligned}$$

So if $U$ and $M$ are positive semi-definite, $\mu \geq 0$ and $\gamma \geq 0$, then $z^T P z \geq 0$ for any $z$, and thus $P$ is positive semi-definite. Therefore, $g(F, W, b)$ is jointly convex with respect to $F$, $W$, and $b$. ■

To obtain the optimal solution, we set the derivatives of the objective function in (8) with respect to $b$ and $W$ equal to zero. We have

$$\begin{aligned} b &= \frac{1}{m}(F^T \mathbf{1} - W^T X \mathbf{1}) \\ W &= \gamma(\gamma X H_c X^T + I)^{-1} X H_c F = AF \end{aligned} \tag{9}$$

where $A = \gamma(\gamma X H_c X^T + I)^{-1} X H_c$ and $H_c = I - (1/m)\mathbf{1}\mathbf{1}^T$ is used for centering the data by subtracting the mean of the data. With $W$ and $b$, we rewrite the regression function $X^T W + \mathbf{1}b^T$ in (8) as

$$\begin{aligned} X^T W + \mathbf{1}b^T &= X^T A F + \frac{1}{m}\mathbf{1}\mathbf{1}^T F - \frac{1}{m}\mathbf{1}\mathbf{1}^T X^T A F \\ &= H_c X^T A F + \frac{1}{m}\mathbf{1}\mathbf{1}^T F = BF \end{aligned} \tag{10}$$

where $B = H_c X^T A + (1/m)\mathbf{1}\mathbf{1}^T$. Replacing $W$ and $b$ to (8), we arrive at

$$\begin{aligned} F^* = \arg\min_F &\mathrm{Tr}(F-Y)^T U(F-Y) \\ &+ \mathrm{Tr}(F^T M F) + \mu(\mathrm{Tr}(F^T A^T A F) \\ &+ \gamma \mathrm{Tr}(BF - F)^T(BF - F)). \end{aligned}$$

By setting the derivative of this objective function with respect to $F$ as 0, the prediction labels $F$ are obtained by

$$F = (U + M + \mu\gamma(B-I)^T(B-I) + \mu A^T A)^{-1} U Y. \tag{11}$$

Using $H_c H_c = H_c = H_c^T$ and $\mu\gamma A^T X H_c X^T A + \mu A^T A = \mu\gamma A^T X H_c = \mu\gamma H_c X^T A$, the term $\mu\gamma(B-I)^T(B-I) +$

$\mu A^TA$ in (11) can be rewritten as $\mu\gamma(A^TX-I)H_c(X^TA-I)+\mu A^TA$ or $\mu\gamma A^TXH_cX^TA-2\mu\gamma H_cX^TA+\mu\gamma H_c+\mu A^TA$. Then, we have

$$\mu\gamma(B-I)^T(B-I)+\mu A^TA=\mu\gamma H_c-\mu\gamma^2H_cX^T\times(\gamma XH_cX^T+I)^{-1}XH_c. \tag{12}$$

By defining $X_c=XH_c$, we can also calculate the prediction labels $F$ by

$$F=(U+M+\mu\gamma H_c-\mu\gamma^2N)^{-1}UY \tag{13}$$

where $N=X_c^T(\gamma X_cX_c^T+I)^{-1}X_c=X_c^TX_c(\gamma X_c^TX_c+I)^{-1}$.

**Algorithm 1: Procedure of FME**

Given a binary label matrix $Y\in\mathbb{B}^{m\times c}$ and a sample set $X=[x_1,x_2,\ldots,x_m]\in\mathbb{R}^{f\times m}$, where $x_i|_{i=1}^n$ and $x_i|_{i=n+1}^m$ are labeled and unlabeled data, respectively.

1: Set $M$ as the graph Laplacian matrix $L\in\mathbb{R}^{m\times m}$, and $U\in\mathbb{R}^{m\times m}$ as the diagonal matrix with the first $n$ and the rest $m-n$ diagonal entries as 1 and 0, respectively.
2: Compute the optimal $F$ with (13).
3: Compute the optimal projection matrix $W$ with (9).

## IV. Unsupervised FME

We propose a simplified version for unsupervised learning by setting the diagonal elements of matrix $U$ in (8) equal to 0. We also aim to solve for the projection matrix $W$, the bias term $b$ and the latent variable $F$ simultaneously

$$(F^*,W^*,b^*)=\arg\min_{F,W,b,\ F^TVF=I}\mathrm{Tr}(F^TMF)+\mu(\|W\|^2+\gamma\|X^TW+\mathbf{1}b^T-F\|^2) \tag{14}$$

where $V$ is set as $H_c$, $I$ is an identity matrix, and the coefficients $\mu$ and $\gamma$ are two parameters to balance different terms.

In unsupervised learning, the variable $F$ can be treated as the latent variable, denoting the lower dimensional representation. Similar to prior work (e.g., LE [3] and LPP [12]), we constrain that $F$ after centering operation lies in a sphere (i.e., $F^TVF=I$) to avoid the trivial solution $F=0$, where we set $V=H_c$. Beside unsupervised learning, the formulation in (14) is a general formulation, which can be also used for supervised learning by using different matrices $M$ and $V$. Again, FME/U naturally provides a method for mapping new data points through the regression function $h(X)=X^TW+\mathbf{1}b^T$. Compared with the prior linear dimension reduction algorithms (such as PCA, LDA, LPP), the hard mapping function $F=X^TW$ in these methods is relaxed by introducing a flexible penalty term (i.e., regression residue $\|h(X)-F\|^2$) in (14).

Similarly, by setting the derivatives of the objective function in (14) with respect to $W$ and $b$ to zero, $W$ and $b$ can be calculated by (9). Substituting $W$ and $b$ back in (14), then we have

$$F^*=\arg\min_{F,F^TH_cF=I}\mathrm{Tr}(F^TMF)+\mu(\mathrm{Tr}(F^TA^TAF)+\gamma\mathrm{Tr}(BF-F)^T(BF-F)). \tag{15}$$

According to (12), we rewrite (15) as

$$F^*=\arg\min_{F,F^TH_cF=I}\mathrm{Tr}\,F^T(M+\mu\gamma H_c-\mu\gamma^2N)F=\arg\min_{F,F^TH_cF=I}\mathrm{Tr}\,F^T(M-\mu\gamma^2N)F \tag{16}$$

where $N=X_c^T(\gamma X_cX_c^T+I)^{-1}X_c=X_c^TX_c(\gamma X_c^TX_c+I)^{-1}$. This objective function can be solved by generalized eigenvalue decomposition [30].

**Algorithm 2: Procedure of FME/U**

Given the unlabeled sample set as $X=[x_1,x_2,\ldots,x_m]\in\mathbb{R}^{f\times m}$.

1: Set $M$ as the graph Laplacian matrix $L\in\mathbb{R}^{m\times m}$.
2: Compute the optimal $F$ with (16) by generalized eigenvalue decomposition.
3: Compute the optimal projection matrix $W$ with (9).

## V. Discussions With the Prior Work

In this section, we discuss the connection between FME and semi-supervised algorithms LGC [36], GFHF [37], and LapRLS/L [21]. We also discuss the connection between FME/U with graph embedding framework [30] and spectral regression [7].

### *A. Connection Between FME and Semi-Supervised Learning Algorithms*

*Example 1:* LGC and GFHF are two special cases of FME.

*Proof:* If we set $\mu=0$, then the objective function of FME in (8) reduces to (2), which is a general formulation for both LGC and GFHF. Therefore, LGC and GFHF are special cases of FME. ■

*Example 2:* LapRLS/L is also a special case of FME.

*Proof:* If we set $\mu=(\lambda_A)/(\lambda_I)$ and $\gamma\to\infty$ (i.e., $\mu\gamma\to\infty$) in (8), we have $F=X^TW+\mathbf{1}b^T$. Replacing $F$ to (8), then we have a new formulation for FME

$$g(W,b)=\mathrm{Tr}(X^TW+\mathbf{1}b^T)^TM(X^TW+\mathbf{1}b^T)+\mu\|W\|^2+\mathrm{Tr}(X^TW+\mathbf{1}b^T-Y)^TU(X^TW+\mathbf{1}b^T-Y). \tag{17}$$

If we further set $M=L$ and the first $n$ and the rest $m-n$ diagonal elements of the diagonal matrix $U$ in (17) as $(1)/(n\lambda_I)$ and 0, respectively, then $g(W,b)$ is equal to $(1)/(\lambda_I)g_M(W,b)$ in (3). That is LapRLS/L is also a special case of FME. ■

### B. Connection Between FME/U and Graph Embedding Framework

Recently, Yan *et al.* [30] proposed a general graph-embedding framework to unify a large family of dimension reduction algorithms (such as PCA, LDA, ISOMAP, LLE, and LE). As shown in [30], the statistical or geometric properties of a given algorithm are encoded as graph relationships, and each algorithm can be considered as direct graph embedding, linear graph embedding, or other extensions. The objective function in direct graph embedding is

$$F^* = \arg \min_{F,\ F^T V F = I} \mathrm{Tr}(F^T M F) \tag{18}$$

where $V$ is another graph Laplacian matrix (e.g., the centering matrix $H_c$) such that $V\mathbf{1} = \mathbf{0}$ and $\mathbf{1}^T V = \mathbf{0}^T$.

While direct graph embedding computes a low-dimensional representation $F$ for the training samples, it does not provide a method to map new data points. For mapping out-of-sample data points, linearization and other extensions (e.g., kernelization and tensorization) are also proposed in [30]. Assuming a hard linear mapping function $F = X^T W + \mathbf{1}b^T$, the objective function in linear graph embedding is formulated as

$$\begin{aligned} W^* &= \arg \min_{W,\ (X^T W + \mathbf{1}b^T)^T V (X^T W + \mathbf{1}b^T) = I} \\ &\quad \mathrm{Tr}(X^T W + \mathbf{1}b^T)^T M (X^T W + \mathbf{1}b^T) \\ &= \arg \min_{W,\ W^T X V X^T W = I} \mathrm{Tr}(W^T X M X^T W). \end{aligned} \tag{19}$$

*Example 3:* Direct graph embedding and its linearization are special cases of FME/U.

*Proof:* If we set $\mu$ in (14) as 0, then the objective function of FME/U reduces to the formulation of direct graph embedding in (18).

When $\mu \to 0$ and $\mu\gamma \to \infty$ in (14), then we have $F = X^T W + \mathbf{1}b^T$. Replacing $F$ to (14) then the objective function of FME/U reduces to the formulation of linear graph embedding in (19).

Therefore, direct graph embedding and its linearization are special cases of FME/U. ■

Note that one recently published semi-supervised dimension reduction method, transductive component analysis (TCA) [15] is closely related to our proposed FME/U. However, TCA is a special case of graph embedding framework [30], in which the matrix $M$ is a weighted sum of two matrices $M_1$ and $M_2$, i.e., $M = M_1 + \beta M_2$, where $\beta > 0$ is a tradeoff parameter to control the importance between the two matrices. The first matrix $M_1 = (I + \alpha L)^{-1}(\alpha L)$ models two terms related to the manifold regularization and the embedding [similarly as in (14)], where $\alpha > 0$ is a parameter to balance two terms. The second matrix $M_2$ models the average margin criterion of the distance constraints for labeled data. Moreover, the prediction label matrix is constrained as $F = X^T W$. In comparison, the proposed FME and FME/U do not constrain $F = X^T W$ on the prediction labels or the lower-dimensional representation. For semi-supervised setting, (13) in FME can be solved by a linear system, which is much more efficient than solving the eigenvalue decomposition problem as in TCA and many other dimension reduction methods [2], [6], [12].

### C. Connection Between FME/U and Spectral Regression (SR)

Cai *et al.* [7] recently proposed a two-step method, referred to as SR, to solve the projection matrix $W$ for mapping new data points. First, the optimal solution $F$ of (18) is solved. Then, the optimal projection matrix $W$ and the bias term $b$ are computed by solving a regression problem:[2]

$$[W^*, b^*] = \arg \min_{W,b} \|X^T W + \mathbf{1}b^T - F\|^2 + \lambda \|W\|^2. \tag{20}$$

*Example 4:* SR is also a special case of FME/U.

*Proof:* When $\mu \to 0$ and $\gamma = 1/\lambda$ (i.e., $\mu\gamma \to 0$) in (14), then (14) reduces to (18), namely, we need to solve $F$ at first. Then, the objective function in (14) is converted to (20) to find the optimal $W$. Note that the optimal $W^*$ of the objective function of SR [i.e., (20)] is $W^* = (X H_c X^T + \lambda I)^{-1} X H_c F$, which is equal to $W^*$ from FME/U [See (9)]. Therefore, spectral regression is also a special case of FME/U. ■

### D. Discussion

The relationships of our FME framework with other related methods are shown in Fig. 2. Direct graph embedding [30] has unified a large family of dimension reduction algorithms (e.g., ISOMAP, LLE, and LE), and LGC [36] and GFHF [37] are two classical graph-based semi-supervised learning methods. When $\mu$ is set as 0, the objective function of FME (*respectively,* FME/U) reduces to the general formulation of LGC/GFHF (*respectively,* direct graph embedding) [30]. In this case, the whole regularization term related to the weighted sum of the regression residue and $\|W\|^2$ is missing such that LGC/GFHF (*respectively,* direct graph embedding) cannot map new data points that are not included in the training set.

While the objective function of LapRLS/L (*respectively,* linear graph embedding) in not in the objective function of FME (*respectively,* FME/U), they are still special cases of our framework by using different paraments $\mu$ and $\gamma$ (or $\mu\gamma$). When $\mu = (\lambda_A)/(\lambda_I)$ and $\gamma \to \infty$ (*respectively,* $\mu \to 0$ and $\mu\gamma \to \infty$), the objective function of FME (*respectively,* FME/U) reduces to the formulation of LapRLS/L [21] (*respectively,* linear graph embedding [30]). In this case, we have a hard linear constraint $F = X^T W + \mathbf{1}b^T$. While the projection matrix $W$ can be used to cope with the out-of-sample problem for LGC/GFHF and direct graph embedding, we argue that the assumption that the prediction labels or the lower-dimension representation $F$ lies in the space spanned by the training samples $X$ is usually over-strict in the real applications. Similarly as in LapRLS/L (*respectively,* linear graph embedding), FME (*respectively,* FME/U) can find the embedding for the unseen data points. Moreover, FME (*respectively,* FME/U) can better cope with the data points sampled from the nonlinear manifold because it relaxes the hard linear constraint in LapRLS/L (*respectively,* linear graph embedding) by explicitly modeling the regression residue in the objective function.

Moreover, our framework also reveals that the previously unrelated methods are in fact related. For example, linear graph embedding and SR [7] seem to be unrelated from their objective functions, while both methods aim to find the optimal linear

[2]The biased term $b$ is not used in [17].

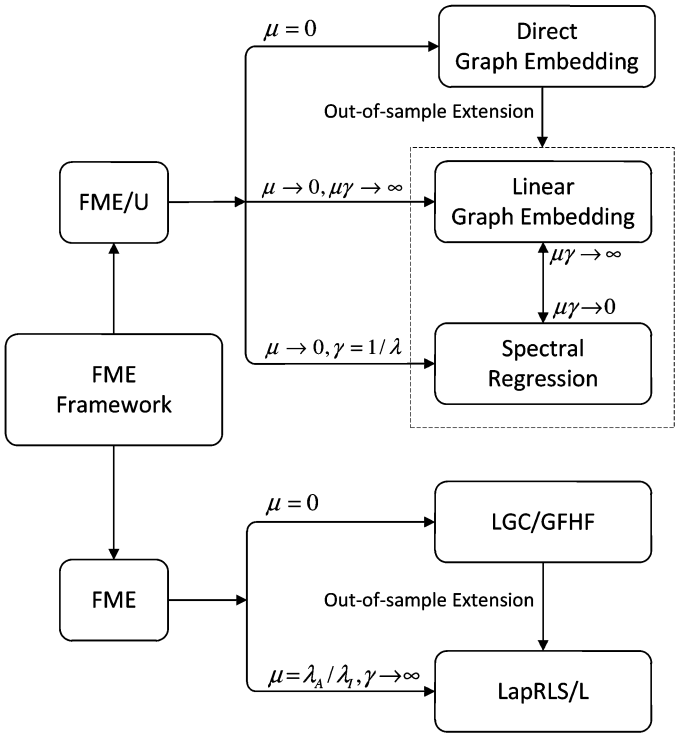



Fig. 2. Relationship of our FME Framework and other related methods.

projection matrix for mapping new data points. However, they are both special cases of FME/U. Specially, FME/U reduces to linear graph embedding, when $\mu \to 0$ and $\mu\gamma \to \infty$. FME/U reduces to SR, when $\mu \to 0$ and $\gamma = 1/\lambda$ (i.e., $\mu\gamma \to 0$).

Finally, our framework can be also used to develop new dimension reduction algorithms. For example, similar as in SR [7], it is also possible to use our FME framework to develop a two-step approach for semi-supervised learning by setting $\mu \to 0$ and $\mu\gamma \to 0$.

## VI. Experiments

In our experiments, we use three face databases UMIST [10], CMU PIE [19], and YALE-B [9], one object database COIL-20 database [16], and one text database 20-NEWS.

**Face Databases:** The UMIST database [10] consists of 575 multi-view images of 20 people, covering a wide range of poses from profile to frontal views. The images are cropped and then resized to $28 \times 23$ pixels. The CMU PIE database [19] 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 this 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 $32 \times 32$ pixels. For the YALE-B database [9], 38 subjects are used in this work, with each person having around 64 near frontal images under different illuminations. The images are cropped and then resized to $32 \times 32$ pixels. In this work, gray-level features are used for face recognition. For each face database, ten images are shown in Fig. 3.

Fig. 3. Ten randomly selected image samples in each image database (From top to bottom: UMIST, YALE-B, CMU PIE, and COIL-20).

**Object Database:** The COIL-20 database [16] consists of images of 20 objects, and each object has 72 images captured from varying angles at intervals of five degrees. We resize each image to $32 \times 32$ pixels, and then extract a 1024 dimensional gray-level feature for each image. Ten images are also shown in Fig. 3.

**Text Database:** The 20-NEWS database[3] is used for text categorization. The topic *rec* which contains *autos*, *motorcycles*, *baseball*, and *hockey* was chosen from the version 20-news-18828. The articles were preprocessed with the same procedure as in [36]. In total, we have 3970 documents. We extract a 8014-dimensional token frequency-inverse document frequency (tf-idf) feature for each document.

### A. Semi-Supervised Learning

We compare FME with LGC [36], GFHF [37], TCA [15], SDA [6], LapRLS/L [21] and MFA [30] for real recognition tasks. For dimension reduction algorithms TCA, SDA, LapRLS/L, MFA, and our FME, the nearest neighbor classifier is performed for classification after dimension reduction. For LGC and GFHF, we directly use the classification methods proposed in [36] and [37] for classification. For GFHF, LapRLS/L, TCA, SDA, and our FME, we need to determine the Laplacian matrix $M$ (or $L$) beforehand. We choose the Gaussian function to calculate $M$ or $L$, in which the graph similarity matrix is set as $S_{ij} = \exp(-\|x_i - x_j\|^2/t)$, if $x_i$ (or $x_j$) is among $k$ nearest neighbors of $x_j$ (or $x_i$); $S_{ij} = 0$, otherwise. For LGC, we used the normalized graph Laplacian matrix $\tilde{L} = I - D^{-(1/2)}SD^{-(1/2)}$, as suggested in [36]. For fair comparison, we fix $k = 10$ and set $t$ according to the method in [17]. For LGC, GFHF and LapRLS/L, the diagonal matrix $U$ is determined according to [36], [37], [21], respectively. For our FME, we set the first $n$ and the rest $m - n$ diagonal elements of the diagonal matrix $U$ as 1 and 0, respectively, similarly as in LapRLS/L.

In all the experiments, PCA is used as a preprocessing step to preserve 95% energy of the data, similarly as in [12], [30]. In order to fairly compare FME with TCA, SDA, LapRLS/L, and MFA, the final dimensions after dimension reduction are fixed as $c$. For SDA, LapRLS/L, TCA, and FME, two regularization parameters (i.e., $\mu$ and $\gamma$ in FME, $\lambda_I$ and $\lambda_A$ in LapRLS/L, $\alpha$ and $\beta$ in SDA and TCA) need to be set beforehand to balance different terms. For fair comparison, we set each parameter to $\{10^{-9}, 10^{-6}, 10^{-3}, 10^{0}, 10^{3}, 10^{6}, 10^{9}\}$, and then we report the top-1 recognition accuracy from the best parameter configuration.

[3]Available at http://people.csail.mit.edu/jrennie/20Newsgroups/.

TABLE I

TOP-1 RECOGNITION PERFORMANCE (MEAN RECOGNITION ACCURACY ± STANDARD DEVIATION %) OF MFA [30], GFHF [37], LGC [36], TCA [15], SDA [6], LAPRLS/L [21], AND FME OVER 20 RANDOM SPLITS ON FIVE DATABASES. FOR EACH DATASET, THE RESULTS SHOWN IN BOLDFACE ARE SIGNIFICANTLY BETTER THAN THE OTHERS, JUDGED BY T-TEST (WITH A SIGNIFICANCE LEVEL OF 0.05). THE OPTIMAL PARAMETERS ARE ALSO SHOWN IN PARENTHESES ($\mu$ AND $\gamma$ IN FME, $\lambda_I$ AND $\lambda_A$ IN LAPRLS/L, $\alpha$ AND $\beta$ IN SDA AND TCA). NOTE THAT WE DO NOT REPORT THE RESULTS FOR MFA WHEN ONLY ONE SAMPLE PER CLASS IS LABELED BECAUSE AT LEAST TWO LABELED SAMPLES PER CLASS ARE REQUIRED IN MFA. CONSIDERING THAT LGC AND GFHF CAN NOT COPE WITH THE UNSEEN SAMPLES, THE RESULTS FOR LGC AND GFHF ON THE TEST DATASET ARE NOT REPORTED

| dataset | method | 1 labeled sample | | 2 labeled samples | | 3 labeled samples | |
|---|---|---|---|---|---|---|---|
| | | Unlabel(%) | Test(%) | Unlabel(%) | Test(%) | Unlabel(%) | Test(%) |
| UMIST | MFA | - | - | 70.5±4.2 | 70.8±3.9 | 81.1±3.9 | 80.6±4.2 |
| | GFHF | **63.6±6.2** | - | **79.1±3.9** | - | **85.8±3.5** | - |
| | LGC | **64.5±5.9** | - | **79.3±3.6** | - | 83.8±3.9 | - |
| | TCA | **63.2±5.2** $(10^3, 10^0)$ | **62.9±5.8** $(10^3, 10^0)$ | **78.0±4.3** $(10^3, 10^6)$ | **77.9±4.2** $(10^3, 10^9)$ | 83.9±4.1 $(10^3, 10^0)$ | 83.6±3.8 $(10^3, 10^0)$ |
| | SDA | 56.2±5.4 $(10^{-9}, 10^{-6})$ | 55.6±5.1 $(10^{-9}, 10^{-9})$ | 76.8±4.2 $(10^0, 10^3)$ | 76.2±4.3 $(10^0, 10^3)$ | 83.7±3.8 $(10^{-3}, 10^3)$ | 83.2±3.9 $(10^0, 10^3)$ |
| | LapRLS/L | 58.1±5.9 $(10^6, 10^9)$ | 57.9±6.1 $(10^3, 10^3)$ | 74.9±4.6 $(10^6, 10^9)$ | 74.3±4.5 $(10^3, 10^6)$ | 82.1±3.9 $(10^6, 10^9)$ | 81.7±3.8 $(10^3, 10^6)$ |
| | FME | **63.5±5.5** $(10^{-9}, 10^{-6})$ | **63.1±5.4** $(10^{-9}, 10^{-6})$ | **79.7±4.5** $(10^{-9}, 10^{-6})$ | **79.1±4.2** $(10^{-9}, 10^{-6})$ | **86.9±3.2** $(10^{-3}, 10^{-6})$ | **86.1±3.1** $(10^{-3}, 10^{-6})$ |

| dataset | method | 1 labeled sample | | 2 labeled samples | | 3 labeled samples | |
|---|---|---|---|---|---|---|---|
| | | Unlabel(%) | Test(%) | Unlabel(%) | Test(%) | Unlabel(%) | Test(%) |
| YALE-B | MFA | - | - | 49.6±4.6 | 49.1±4.9 | 68.1±2.9 | 68.6±2.8 |
| | GFHF | 22.5±2.9 | - | 35.9±3.3 | - | 45.2±3.9 | - |
| | LGC | 29.2±3.1 | - | 42.1±3.1 | - | 49.6±3.5 | - |
| | TCA | 38.5±3.0 $(10^0, 10^0)$ | 39.6±3.2 $(10^0, 10^0)$ | 71.6±3.3 $(10^0, 10^6)$ | 71.4±3.1 $(10^0, 10^6)$ | 81.5±2.9 $(10^3, 10^0)$ | 81.2±2.3 $(10^3, 10^0)$ |
| | SDA | 38.2±3.0 $(10^0, 10^{-9})$ | 39.0±3.1 $(10^0, 10^{-9})$ | 72.1±3.6 $(10^0, 10^{-6})$ | 71.9±3.2 $(10^0, 10^{-6})$ | 83.8±2.1 $(10^0, 10^{-3})$ | 83.0±2.1 $(10^0, 10^{-3})$ |
| | LapRLS/L | **52.9±3.6** $(10^{-3}, 10^{-9})$ | **53.2±3.1** $(10^{-3}, 10^{-9})$ | 73.9±3.2 $(10^0, 10^{-6})$ | 73.6±2.9 $(10^0, 10^{-9})$ | 84.2±2.6 $(10^0, 10^{-9})$ | 83.8±2.5 $(10^0, 10^{-9})$ |
| | FME | **53.9±3.3** $(10^{-6}, 10^6)$ | **54.2±2.9** $(10^{-6}, 10^6)$ | **75.1±2.6** $(10^{-3}, 10^3)$ | **75.0±2.7** $(10^{-3}, 10^3)$ | **85.9±2.1** $(10^{-3}, 10^3)$ | **85.6±2.0** $(10^{-3}, 10^3)$ |

| dataset | method | 1 labeled sample | | 2 labeled samples | | 3 labeled samples | |
|---|---|---|---|---|---|---|---|
| | | Unlabel(%) | Test(%) | Unlabel(%) | Test(%) | Unlabel(%) | Test(%) |
| CMU PIE | MFA | - | - | 72.1±2.6 | 71.8±2.3 | 83.0±1.9 | 83.1±1.8 |
| | GFHF | 33.9±3.3 | - | 47.8±2.6 | - | 55.8±2.1 | - |
| | LGC | 36.2±3.1 | - | 47.9±2.3 | - | 55.9±1.9 | - |
| | TCA | 61.3±3.6 $(10^3, 10^6)$ | 60.8±3.7 $(10^3, 10^3)$ | 78.6±2.4 $(10^3, 10^0)$ | 78.4±2.3 $(10^3, 10^0)$ | 86.9±1.2 $(10^3, 10^0)$ | 86.6±1.1 $(10^3, 10^0)$ |
| | SDA | 59.4±3.2 $(10^0, 10^{-9})$ | 58.7±2.8 $(10^0, 10^{-9})$ | **81.5±2.1** $(10^0, 10^{-3})$ | **81.2±2.0** $(10^0, 10^{-3})$ | **88.6±1.2** $(10^0, 10^{-3})$ | **88.8±1.1** $(10^0, 10^{-3})$ |
| | LapRLS/L | 57.9±3.1 $(10^0, 10^{-9})$ | 57.5±2.6 $(10^0, 10^{-9})$ | 79.1±2.2 $(10^{-3}, 10^{-6})$ | 79.0±1.8 $(10^{-3}, 10^6)$ | 87.8±1.1 $(10^{-3}, 10^{-6})$ | 87.7±1.1 $(10^{-3}, 10^{-6})$ |
| | FME | **63.2±2.8** $(10^{-6}, 10^3)$ | **62.7±2.6** $(10^{-6}, 10^3)$ | **81.8±2.0** $(10^{-6}, 10^3)$ | **81.5±1.9** $(10^{-6}, 10^3)$ | **89.1±1.2** $(10^{-6}, 10^3)$ | **88.9±1.0** $(10^{-6}, 10^3)$ |

| dataset | method | 1 labeled sample | | 2 labeled samples | | 3 labeled samples | |
|---|---|---|---|---|---|---|---|
| | | Unlabel(%) | Test(%) | Unlabel(%) | Test(%) | Unlabel(%) | Test(%) |
| COIL-20 | MFA | - | - | 70.2±2.6 | 70.1±3.2 | 76.5±2.5 | 76.2±2.3 |
| | GFHF | **78.6±2.1** | - | **83.2±2.2** | - | **85.6±2.0** | - |
| | LGC | **78.5±2.6** | - | **82.9±2.1** | - | **85.9±2.1** | - |
| | TCA | 70.6±2.9 $(10^9, 10^3)$ | 70.5±2.8 $(10^9, 10^3)$ | 78.1±2.5 $(10^9, 10^9)$ | 77.9±2.1 $(10^9, 10^9)$ | 81.7±2.2 $(10^9, 10^0)$ | 81.5±2.6 $(10^9, 10^0)$ |
| | SDA | 59.9±2.5 $(10^{-9}, 10^0)$ | 59.8±3.2 $(10^{-9}, 10^0)$ | 73.2±2.7 $(10^3, 10^9)$ | 73.3±2.5 $(10^3, 10^9)$ | 78.3±2.2 $(10^{-9}, 10^6)$ | 78.1±2.5 $(10^0, 10^6)$ |
| | LapRLS/L | 60.5±3.2 $(10^0, 10^6)$ | 60.6±3.5 $(10^0, 10^6)$ | 73.5±2.9 $(10^0, 10^6)$ | 73.1±2.5 $(10^0, 10^6)$ | 78.6±2.6 $(10^0, 10^6)$ | 78.8±2.5 $(10^0, 10^6)$ |
| | FME | 75.1±3.2 $(10^{-9}, 10^{-6})$ | **75.5±3.1** $(10^{-9}, 10^{-6})$ | **82.2±2.9** $(10^{-9}, 10^{-6})$ | **81.9±3.1** $(10^{-9}, 10^{-6})$ | **86.1±2.3** $(10^{-9}, 10^{-6})$ | **85.6±2.6** $(10^{-9}, 10^{-6})$ |

| dataset | method | 10 labeled samples | | 20 labeled samples | | 30 labeled samples | |
|---|---|---|---|---|---|---|---|
| | | Unlabel(%) | Test(%) | Unlabel(%) | Test(%) | Unlabel(%) | Test(%) |
| 20-NEWS | MFA | 46.5±7.2 | 46.2±7.6 | 61.9±5.9 | 61.3±6.2 | 70.9±4.5 | 71.5±4.1 |
| | GFHF | 72.5±7.5 | - | 83.6±2.5 | - | 86.1±1.1 | - |
| | LGC | 80.9±2.3 | - | 83.9±1.5 | - | 85.5±1.0 | - |
| | TCA | 55.2±5.3 $(10^{-3}, 10^{-3})$ | 56.6±5.2 $(10^{-3}, 10^{-3})$ | 68.6±3.3 $(10^{-3}, 10^{-3})$ | 67.5±3.2 $(10^{-3}, 10^{-3})$ | 75.6±3.2 $(10^{-9}, 10^{-9})$ | 73.9±2.5 $(10^{-3}, 10^{-3})$ |
| | SDA | 57.5±5.9 $(10^{-9}, 10^6)$ | 58.1±5.8 $(10^{-9}, 10^6)$ | 72.9±3.8 $(10^{-9}, 10^6)$ | 73.6±3.6 $(10^{-3}, 10^6)$ | 78.9±2.7 $(10^{-9}, 10^6)$ | 80.5±2.2 $(10^{-9}, 10^3)$ |
| | LapRLS/L | 61.9±4.5 $(10^{-9}, 10^3)$ | 62.2±4.6 $(10^{-9}, 10^3)$ | 75.6±2.5 $(10^{-9}, 10^3)$ | 76.2±2.6 $(10^{-9}, 10^3)$ | 80.9±1.7 $(10^{-9}, 10^3)$ | 81.2±1.9 $(10^{-9}, 10^3)$ |
| | FME | **83.2±3.2** $(10^{-9}, 10^{-6})$ | **82.5±3.6** $(10^{-9}, 10^{-6})$ | **88.2±1.9** $(10^3, 10^{-6})$ | **87.6±2.0** $(10^{-9}, 10^{-6})$ | **90.1±1.3** $(10^3, 10^{-6})$ | **89.6±1.5** $(10^3, 10^{-6})$ |

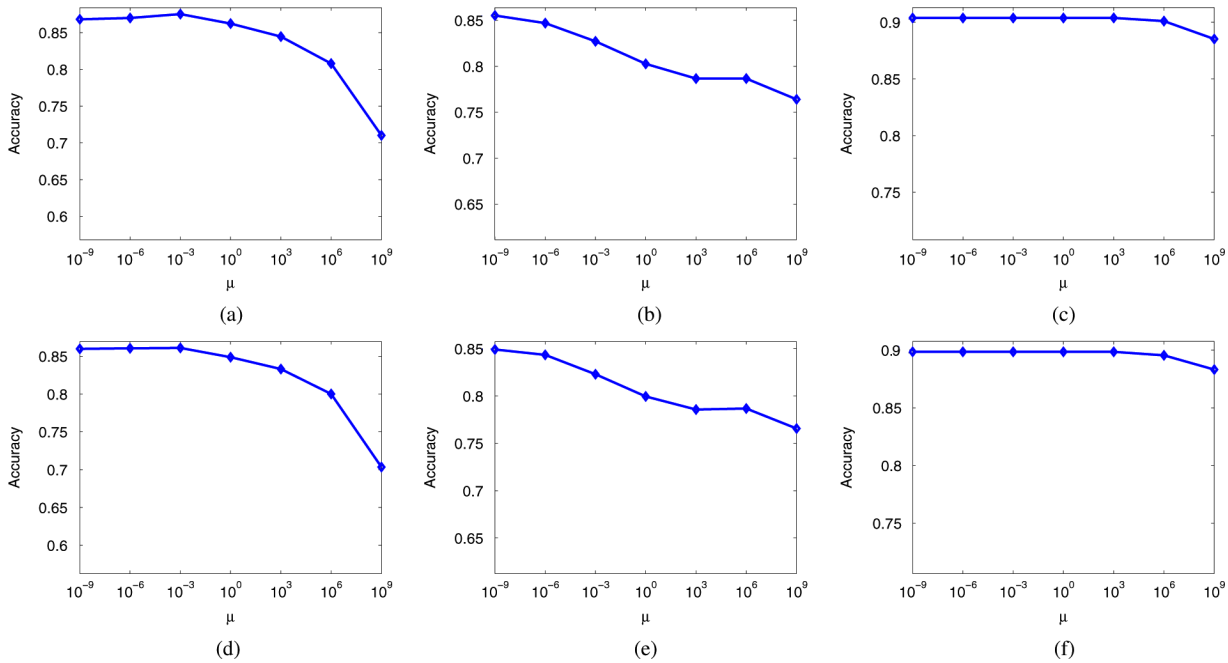



Fig. 4. Recognition accuracy variation with different parameter $\mu$ for FME. The two rows show the results on the unlabeled dataset and unseen test dataset, respectively. Three labeled samples per class are used in UMIST, and COIL-20 databases, and 30 labeled samples per class are used in 20-NEWS database. (a) UMIST, (b) COIL-20, (c) 20-NEWS, (d) UMIST, (e) COIL-20, (f) 20-NEWS.

We randomly select 50% data as the training dataset and use the remaining 50% data as the test dataset. Among the training data, we randomly label $p$ samples per class and treat the other training samples as unlabeled data. The above setting (referred to as semi-supervised setting) has been used in [6] and it is also a more natural setting to compare different dimension reduction algorithms. For UMIST, CMU PIE, YALE-B, and COIL-20 databases, we set $p$ as 1, 2, and 3, respectively. For the 20-NEWS text database, we set $p$ as 10, 20, and 30, respectively, because each class has much more training samples in this database. All the training data are used to learn a subspace (i.e., a projection matrix) or a classifier, except that we only use the labeled data for subspace learning in MFA [30]. We report the mean recognition accuracy and standard deviation over 20 random splits on the unlabeled dataset and the unseen test dataset, which are referred to as *Unlabel* and *Test*, respectively, in Table I. In Table I, the results shown in boldface are significantly better than the others, judged by t-test with a significance level of 0.05. We have the following observations.

1) Semi-supervised dimension reduction algorithms TCA, SDA, and LapRLS/L outperform supervised MFA in terms of mean recognition accuracy, which demonstrates that unlabeled data can be used to improve the recognition performance.
2) When comparing TCA, SDA, and LapRLS/L, there is no consistent winner on all the databases. Among the three algorithms, we observe TCA achieves the best results on UMIST and COIL-20 databases, LapRLS/L is the best on YALE-B and 20-NEWS databases, and SDA is generally better on CMU PIE database, in terms of mean recognition accuracy.
3) The mean recognition accuracies of LGC and GFHF are generally better than TCA, SDA, and LapRLS/L on the unlabeled dataset of UMIST, COIL-20, and 20-NEWS databases, which demonstrate the effectiveness of label propagation. But we also observe that the recognition accuracies from LGC and GFHF are much worse than TCA, SDA, and LapRLS/L on the unlabeled dataset of CMU PIE and Yale-B databases, possibly because of the strong lighting variations of images in the two databases. The labels may not be correctly propagated in this case, which significantly degrades the performances of LGC and GFHF.
4) Our method FME outperforms MFA and semi-supervised dimension reduction methods TCA, SDA, and LapRLS/L in all the cases in terms of mean recognition accuracy. Judged by t-test (with a significance level of 0.05), FME is significantly better than MFA, TCA, SDA, and LapRLS/L in 20 out of 30 cases. On unlabeled dataset, FME significantly outperforms GFHF and LGC in 9 out of 15 cases. While GFHF/LGC is significantly better than FME in one case on COIL-20 database, LGC and GFHF cannot cope with the unseen data.

Finally, we plot the recognition accuracy variation with different parameter $\mu$ for FME in Fig. 4, in which three labeled samples per class are used in UMIST and COIL-20 databases, and 30 labeled samples per class are used in 20-NEWS database. We observe that FME is relatively robust to the parameter $\mu$ when $\mu$ is small (i.e., $\mu \leq 10^{-3}$). We have similar observations on other databases as well as with different number of labeled samples. It is also interesting to observe that the results of FME are the best when using small values for $\mu$ and $\gamma$ when LGC or GFHF performs well (see some cases on UMIST and COIL-20 databases). For these cases, the terms related to label fitness and manifold smoothness are more important and therefore small values for $\mu$ and $\gamma$ can lead to the best performances for FME.

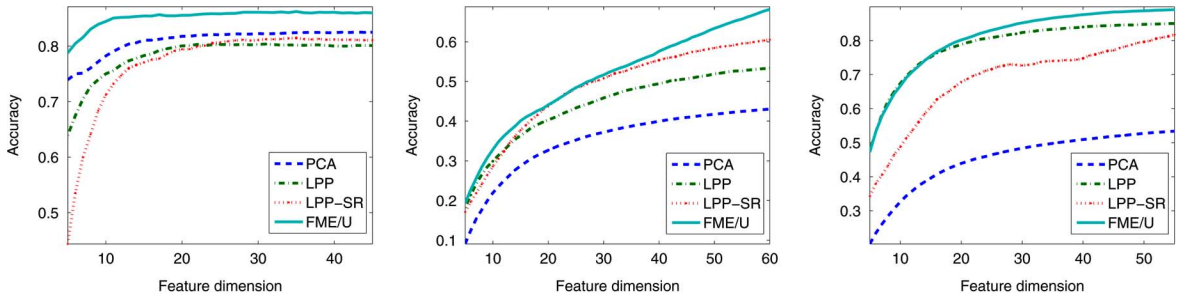



Fig. 5. Top-1 recognition rates (%) with different feature dimensions on the UMIST, YALE-B, and CMU PIE databases. (a) UMIST, (b) YALE-B, (c) CMU PIE.

TABLE II
TOP-1 RECOGNITION PERFORMANCE (MEAN RECOGNITION ACCURACY ± STANDARD DEVIATION %) OF PCA [25], LPP [12], LPP-SR [7], AND FME/U OVER 20 RANDOM SPLITS ON THREE FACE DATABASES. FOR EACH DATASET, THE RESULTS SHOWN IN BOLDFACE ARE SIGNIFICANTLY BETTER THAN THE OTHERS, JUDGED BY T-TEST (WITH A SIGNIFICANCE LEVEL OF 0.05). NOTE THE LAST NUMBERS IN PARENTHESES ARE THE OPTIMAL DIMENSIONS AFTER DIMENSION REDUCTION. THE FIRST NUMBER IN LPP-SR IS THE OPTIMAL $\lambda$ AND THE FIRST TWO NUMBERS IN FME/U ARE THE OPTIMAL PARAMETERS $\mu$ AND $\gamma$

| method | UMIST | YALE-B | CMU PIE |
|---|---|---|---|
| PCA | 82.6±3.2 (43) | 43.1±1.2 (60) | 53.4±1.5 (55) |
| LPP | 80.4±3.5 (31) | 53.3±3.1 (60) | 85.0±1.2 (55) |
| LPP-SR | 81.5±3.2 ($10^6$, 35) | 60.5±3.0 ($10^{-3}$, 60) | 81.7±2.1 ($10^{-5}$, 55) |
| FME/U | **86.3±2.8** ($10^3$, $10^0$, 35) | **68.2±2.5** ($10^{-9}$, $10^9$, 60) | **89.1±1.0** ($10^{-9}$, $10^9$, 55) |

However, the best results of FME are obtained by using small value $\mu$ and large value $\gamma$ in some cases on YALE-B and CMU PIE databases when LGC/GFHF performs poor. For these cases, the term related to the regression residue which models the mismatch between $F$ and $h(X) = X^T W + \mathbf{1}b^T$ becomes more important. However, it is still an open problem to determine the optimal parameters for FME, which will be investigated in the future.

### B. Unsupervised Learning

We also compare FME/U with the unsupervised learning algorithms PCA [25] and LPP [12] on three face databases UMIST, CMU PIE and YALE-B. We also report the results from LPP-SR, in which the Spectral Regression method [7] is used to find the projection matrix in the objective function of LPP. The nearest neighbor classifier is used again for classification after dimension reduction. Five images per class are randomly chosen as the training dataset and remaining images are used as the test dataset. Again, PCA is used as a preprocessing step to preserve 95% energy of the data in all the experiments. The optimal parameters $\mu$ and $\gamma$ in FME/U are also search from the set $\{10^{-9}, 10^{-6}, 10^{-3}, 10^0, 10^3, 10^6, 10^9\}$, and we report the best results from the optimal parameters. For LPP-SR, we use a more dense set $\{10^{-9}, 10^{-8}, \ldots, 10^8, 10^9\}$ for the parameter $\lambda$ and report the best results. For PCA, LPP, LPP-SR, and FME/U, we run all the possible lower dimensions and choose the optimal dimensions corresponding to the best recognition accuracies. We also report the mean recognition accuracy and standard deviation over 20 random splits in Table II. Fig. 5 plots the recognition accuracy with respect to the number of features.

We have the following observations.

1) LPP outperforms PCA on CMU PIE and YALE-B databases, which is consistent with the prior work [12]. We also observe that LPP is slightly worse than PCA on UMIST database, possibly because the limited training data cannot correctly characterize the nonlinear manifold structure in this database.
2) When comparing LPP and LPP-SR, there is no consistent winner on all three databases.
3) Our work FME/U achieves the best results in all the cases, which demonstrates that FME/U is an effective unsupervised dimension reduction method.

## VII. CONCLUSION

In this paper, we propose a unified manifold embedding framework for both semi-supervised and unsupervised learning, and most of existing dimension reduction methods are also unified under the proposed framework. For semi-supervised dimension reduction, FME can provide mappings for unseen data points through a linear regression function and effectively cope with the data sampled from the nonlinear manifold by modeling the regression residue. FME also utilizes the label information from the labeled data as well as the manifold smoothness from both labeled and unlabeled data. A simplified version referred to as FME/U is also proposed for unsupervised dimension reduction. The comprehensive experiments on five benchmark databases clearly demonstrate that FME and FME/U outperform existing dimension reduction algorithms. In the future, we plan to extend FME and FME/U to kernel FME and kernel FME/U by using kernel trick as well as examine how to choose optimal parameters for $\mu$ and $\gamma$.

## REFERENCES

[1] J. Abernethy, O. Chapelle, and C. Castillo, "Web spam identification through content and hyperlinks," in *Proc. Int. Workshop Adversarial Inf. Retrieval Web*, 2008, pp. 41–44.
[2] P. Belhumeur, J. Hespanha, and D. Kriegman, "Eigenfaces vs. fisherfaces: Recognition using class specific linear projection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 711–720, Jul. 1997.

[3] M. Belkin and P. Niyogi, "Laplacian eigenmaps and spectral techniques for embedding and clustering," in *Proc. Adv. Neural Inf. Process. Syst.*, 2001, pp. 585–591.
[4] M. Belkin, P. Niyogi, and V. Sindhwani, "Manifold regularization: A geometric framework for learning from labeled and unlabeled examples," *J. Mach. Learn. Res.*, vol. 12, pp. 2399–2434, 2006.
[5] A. Blum and T. Mitchell, "Combining labeled and unlabeled data with co-training," in *Proc. Ann. Conf. Computational Learn. Theory*, 1998, pp. 92–100.
[6] D. Cai, X. He, and J. Han, "Semi-supervised discriminant analysis," presented at the IEEE Int. Conf. Comput. Vision, Rio de Janiero, Brazil, 2007.
[7] D. Cai, X. He, and J. Han, "Spectral regression for efficient regularized subspace learning," presented at the IEEE Int. Conf. Comput. Vision, Rio de Janiero, Brazil, 2007.
[8] W. Chu, V. Sindhwani, Z. Ghahramani, and S. S. Keerthi, "Relational learning with Gaussian processes," in *Proc. Adv. Neural Inf. Process. Syst.*, 2006, pp. 289–296.
[9] 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.
[10] D. B. Graham and N. M. Allinson, "Characterizing virtual eigensignatures for general purpose face recognition," *NATO ASI Series F*, pp. 446–456, 1998.
[11] Z. Guo, Z. Zhang, E. Xing, and C. Faloutsos, "Semi-supervised learning based on semiparametric regularization," in *Proc. SIAM Int. Conf. Data Min.*, 2008, pp. 132–142.
[12] X. He, S. Yan, Y. Hu, P. Niyogi, and H. Zhang, "Face recognition using laplacianfaces," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 3, pp. 328–340, Mar. 2005.
[13] 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, Feb. 2008.
[14] C. Liu, H. Y. Shum, and W. T. Freeman, "Face hallucination: Theory and practice," *Int. J. Comput. Vision*, vol. 75, no. 1, pp. 115–134, 2007.
[15] W. Liu, D. Tao, and J. Liu, "Transductive component analysis," in *Proc. IEEE Int. Conf. Data Min.*, 2008, pp. 433–442.
[16] S. A. Nene, S. K. Nayar, and H. Murase, "Columbia object image library (Coil-20)," Columbia Univ., Tech. Rep. CUCS-005-96, Feb. 1996.
[17] F. P. Nie, S. M. Xiang, Y. Q. Jia, and C. S. Zhang, "Semi-supervised orthogonal discriminant analysis via label propagation," *Pattern Recog.*, vol. 42, no. 11, pp. 2615–2627, 2009.
[18] S. Roweis and L. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 22, pp. 2323–2326, 2000.
[19] T. Sim and S. Baker, "The cmu pose, illumination, and expression database," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 12, pp. 1615–1617, Dec. 2003.
[20] V. Sindhwani, P. Niyogi, and M. Belkin, "Beyond the point cloud: From transductive to semi-supervised learning," in *Proc. Int. Conf. Mach. Learn.*, 2005, pp. 824–831.
[21] V. Sindhwani, P. Niyogi, and M. Belkin, "Linear manifold regularization for large scale semi-supervised learning," presented at the Workshop Learn. With Partially Classified Train. Data, Int. Conf. Mach. Learn., Bonn, Germany, 2005.
[22] V. Sindhwani and P. Melville, "Document-word co-regularization for semi-supervised sentiment analysis," in *Proc. IEEE Int. Conf. Data Min.*, 2008, pp. 1025–1030.
[23] J. Tenenbaum, V. Silva, and J. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 22, pp. 2319–2323, 2000.
[24] I. W. Tsang and J. T. Kwok, "Large-scale sparsified manifold regularization," in *Proc. Adv. Neural Inf. Process. Syst.*, 2006, pp. 1401–1408.
[25] M. Turk and A. Pentland, "Face recognition using eigenfaces," in *Proc. IEEE Comput. Soc. Conf. Comput. Vision Pattern Recog.*, 1991, pp. 586–591.
[26] V. Vapnik, *Statistical Learning Theory*. New York: Wiley-Interscience, 1998.
[27] M. Wu, K. Yu, S. Yu, and B. Schölkopf, "Local learning projections," in *Proc. Int. Conf. Mach. Learn.*, 2007, pp. 1039–1046.
[28] D. Xu and S. Yan, "Semi-supervised bilinear subspace learning," *IEEE Trans. Image Process.*, vol. 18, no. 7, pp. 1671–1676, Jul. 2009.
[29] D. Xu, S. Yan, S. Lin, T. S. Huang, and S. F. Chang, "Enhancing bilinear subspace learning by element rearrangement," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 10, pp. 1913–1920, Oct. 2009.
[30] 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.
[31] X. Yang, H. Fu, H. Zha, and J. L. Barlow, "Semi-supervised nonlinear dimensionality reduction," in *Proc. Int. Conf. Mach. Learn.*, 2006, pp. 1065–1072.
[32] Y. Yang, D. Xu, F. Nie, J. Luo, and Y. Zhuang, "Ranking with local regression and global alignment for cross media retrieval," in *Proc. ACM Multimedia*, 2009, pp. 175–184.
[33] T. Zhang, A. Popescul, and B. Dom, "Linear prediction models with graph regularization for web-page categorization," in *Proc. ACM SIGKDD Int. Conf. Knowledge Discovery Data Min.*, 2006, pp. 821–826.
[34] T. Zhang, D. Tao, and J. Yang, "Discriminative locality alignment," in *Proc. Eur. Conf. Comput. Vision*, 2008, pp. 725–738.
[35] T. Zhang, D. Tao, X. Li, and J. Yang, "Patch alignment for dimensionality reduction," *IEEE Trans. Knowledge Data Eng.*, vol. 21, no. 9, pp. 1299–1313, Sep. 2009.
[36] D. Zhou, O. Bousquet, T. N. Lal, J. Weston, and B. Schölkopf, "Learning with local and global consistency," in *Proc. Adv. Neural Inf. Process. Syst.*, 2004, pp. 321–328.
[37] X. Zhu, Z. Ghahramani, and J. Lafferty, "Semi-supervised learning using gaussian fields and harmonic functions," in *Proc. Int. Conf. Mach. Learn.*, 2003, pp. 912–919.
[38] X. Zhu, "Semi-supervised learning literature survey," Univ. Wisconsin-Madison, 2007.

**Feiping Nie** received the B.S. degree in computer science from North China University of Water Conservancy and Electric Power, 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.

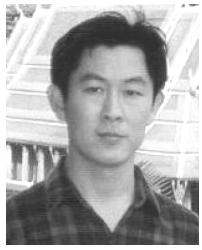
**Dong Xu** (M'07) received the B.Eng. and Ph.D. degrees from the Electronic Engineering and Information Science Department, University of Science and Technology of China, Beijing, China, in 2001 and 2005, respectively.

He is currently an Assistant Professor with Nanyang Technological University, Singapore. During his Ph.D. studies, he worked with Microsoft Research Asia and The Chinese University of Hong Kong. He also spent one year at Columbia University, New York, as a Postdoctoral Research Scientist. His research interests include computer vision, machine learning, and multimedia content analysis.

**Ivor Wai-Hung Tsang** received the Ph.D. degree in computer science from the Hong Kong University of Science and Technology (HKUST), Hong Kong, in 2007.

He is currently an Assistant Professor with the School of Computer Engineering, Nanyang Technological University (NTU), Singapore. He is currently also the Deputy Director of the Centre for Computational Intelligence of NTU. His scientific interests include machine learning, kernel methods, and large-scale optimization, and their applications to data mining and pattern recognitions.

Dr. Tsang was awarded the prestigious IEEE TRANSACTIONS ON NEURAL NETWORKS Outstanding 2004 Paper Award in 2006. In 2009, he clinched the second-class prize of the National Natural Science Award 2008, China. He was also awarded the Microsoft Fellowship in 2005, the Best Paper Award from the IEEE Hong Kong Chapter of Signal Processing Postgraduate Forum in 2006, and also the HKUST Honor Outstanding Student in 2001.



**Changshui Zhang** (M'02) received the B.S. degree in mathematics from Peking University, Beijing, China, in 1986 and the M.S. and Ph.D. degrees in control science and engineering from Tsinghua University, Beijing, China, in 1989 and 1992, respectively.

In 1992, he joined the Department of Automation, Tsinghua University, and is currently a Professor. His research interests include pattern recognition, machine learning, etc. He has authored more than 200 papers.

Prof. Zhang is currently an Associate Editor of the *Pattern Recognition Journal*. He is also a member of the Standing Council of Chinese Association of Artificial Intelligence.