Binary- and Multi-class Group Sparse Canonical Correlation Analysis for Feature Extraction and Classification

Zhao Zhang, *Student Member*, *IEEE*, Mingbo Zhao, *Student Member*, *IEEE*, and Tommy W.S. Chow, *Senior Member*, *IEEE*

Abstract—This paper incorporates the group sparse representation into the well-known canonical correlation analysis (CCA) framework and proposes a novel discriminant feature extraction technique named group sparse canonical correlation analysis (GSCCA). GSCCA uses two sets of variables and aims at preserving the group sparse (GS) characteristics of data within each set in addition to maximize the global interset covariance. With GS weights computed prior to feature extraction, the locality, sparsity and discriminant information of data can be adaptively determined. The GS weights are obtained from an NP-hard group-sparsity promoting problem that considers all highly correlated data within a group. By defining one of the two variable sets as the class label matrix, GSCCA is effectively extended to multiclass scenarios. Then GSCCA is theoretically formulated as a least-squares problem as CCA does. Comparative analysis between this work and the related studies demonstrate that our algorithm is more general exhibiting attractive properties. The projection matrix of GSCCA is analytically solved by applying eigen-decomposition and trace ratio (TR) optimization. Extensive benchmark simulations are conducted to examine GSCCA. Results show that our approach delivers promising results, compared with other related algorithms.

Index Terms—Canonical correlation analysis, group sparse representation, multiclass classification, feature extraction

1 Introduction

The classical canonical correlation analysis (CCA) [1], [2] is a standard multivariate analysis technique relying on two sets of variables. CCA maximizes the correlations between the projections of the variables onto the obtained pairs of basis vectors. CCA has attracted considerable interests in theoretical studies [3], [4], [5], [6], [7] and practical applications, for example, gene expression [8], [9], [10], [11]. CCA was originally proposed for dealing with two sets of variables in an unsupervised manner. It was extended to handle multiclass classification problems, i.e., multiclass CCA (MCCA) [3], by setting one set of variables being the class label matrix. Then the labels of data samples can be conducted for discriminant feature extraction (FE) and subsequent multiclass classification. According to [3], class information is usually represented in a form of numerical label encodings. In particular, when proper label encodings are applied, MCCA can be equivalent to linear discriminant analysis (LDA) [12]. Another interesting relationship under a mild condition between the least squares (LS) and CCA in multiclass case has been rigorously established in [3] and [13].

Recently, many neighborhood preserving graph algorithms, for example, Laplacian Eigenamps (LE) [14], locally linear embedding (LLE) [15], and ISOMAP [16], were proposed for nonlinear dimensionality reduction. The core step of neighborhood preservation is to construct the neighborhood graph and define the weights [14]. This process essentially involves the operation of finding the neighbors of each point by certain methods, for example, k-neighborhood or ε-neighborhood [14]. But the determination of optimal k and ε values is still an open problem. In addition to the above problems, the adjacency graph with fixed k or ε fails to consider the actual data distribution [25]. More recently, sparse representation (SR) has become increasingly popular and important in various areas, for example, pattern recognition [17], [18], [25] and machine learning [19], [20], [21]. Sparse graph weights are usually calculated using l^1-minimization [17], [20]. As studied in [19], the locality and sparsity of samples can be adaptively determined in l^1-graph. By adapting the elastic net into CCA, a penalized CCA (PCCA) was proposed, in which CCA was penalized as an iterative regression. A Lasso penalty was then applied to find the canonical vectors [9]. PCCA was subsequently generalized for identifying candidate genes for incorporation in the pathway [11]. Note that two parameters for ridge and Lasso penalties are involved, but estimating optimal penalty parameters is still difficult [5]. By including a sparseness parameter to control the number of variables to be contained from each set, a sparse CCA (SCCA) was first used for genomic data integration [8]. Another format of SCCA [7] was proposed by formulating CCA in both primal (input) and dual (kernel) representation for both views. This setting minimizes the number of features in both primal and dual projections while maximizing the correlation between the two variable sets. Note that kernel trick is used in [7], but estimating an optimal

• *The authors are with the Department of Electronic Engineering, Academic Building, City University of Hong Kong, 83 Tat Chee Avenue, Kowloon, Hong Kong. E-mail: itzzhang@ee.cityu.edu.hk, mzhao4@student.cityu.edu.hk, eetchow@cityu.edu.hk.*

Manuscript received 4 Aug. 2011; revised 10 July 2012; accepted 3 Oct. 2012; published online 26 Oct. 2012.
Recommended for acceptance by X. Zhu.
For information on obtaining reprints of this article, please send e-mail to: tkde@computer.org, and reference IEEECS Log Number TKDE-2011-08-0472.
Digital Object Identifier no. 10.1109/TKDE.2012.217.

kernel width is still a difficult issue. Wiesel et al. [19] discussed an SCCA that maximized the correlation by obtaining a pair of linear combinations with prescribed cardinality. Lykou and Whittaker [5] have also developed an extended SCCA algorithm for the first two dimensions with the positivity Lasso. A common property of these SCCA versions is that they generally work similarly as CCA does. All of these kinds of SCCA seek a pair of 1D sparse projection vectors for representing the two sets of variables, but obtaining multiple projection vectors are required in the general pattern recognition applications. However, how to extend these SCCA methods to handle multiclass classification through delivering multiple projection axes is still unclear. Note that virtually all previous SCCA methods are proposed for performing gene selection in the areas of bioinformatics [22]. In contrast, we in this paper are proposing a general multiple projections-based discriminant FE criterion for feature reduction and pattern recognition.

Compared with the other existing SCCA methods, this present work exhibits several properties.

1. For a group of highly correlated samples, SR tends to select one single point from the group [24], so l^1-norm is not an ideal choice for ensuring selection of all correlated samples in a group [23]. Group sparsity [23], [32] was introduced to ensure that all weights for a particular class are selected, delivering a grouping effect. This work mainly considers group sparsity [23] in FE. We incorporate the concept of group sparsity into CCA and establish a novel group sparse canonical correlation analysis (GSCCA) technique. GSCCA is originally proposed for handling binary-class cases as CCA, which may constrain its certain applications, for example, multiclass classification. We also extend GSCCA to multiclass case by setting one set of variables as the class label matrix. GSCCA is motivated by sparsity preserving projections (SPP) [25], but it differs from SPP in three aspects. First, GSCCA relies on two sets of variables as CCA does, but SPP only works on one set of variables. Second, multiclass GSCCA is a supervised method, but SPP is naturally unsupervised. Third, SPP calculates the sparse reconstruction weights by l^1-minimization, while GSCCA considers group sparsity. In extracting the representative features, GSCCA focuses on maximizing the global interset covariance while computing the multiple projections that best preserve the GS characteristics within the two variable sets at the same time.
2. In GSCCA, GS weights are calculated directly from an NP-hard group-sparsity promoting minimization problem solved by faster greedy algorithms prior to FE. As a result, data of a particular class can be well grouped. Also, GSCCA can avoid estimating the model parameters, that is, kernel width and neighborhood size. By adding the sum-to-one constraint to the NP-hard problem, the GS reconstruction procedure can enable GSCCA to preserve certain local information of data [25].
3. GSCCA exhibits a strong generalization power. By comparing GSCCA with the related work, LDA, MCCA, and SPP are treated as special cases of our approach.
4. GSCCA delivers multiple projections for representation. The basis vectors can be calculated by using eigen-decomposition and TR optimization. When eigen-decomposition is applied, we mathematically formulate GSCCA as an LS problem and detail an inherent equivalence between them. When TR criterion is used, specific solution and similarity preservation are simultaneously obtained according to the orthogonal constraints [26]. It is owing to these properties that GSCCA is applicable to various tasks, such as multiclass classification, multidimensional data visualization and feature extraction, but previous SCCA methods are unable to cope with.

The paper is outlined as follows: Section 2 reviews CCA. In Section 3, we mathematically propose GSCCA. Section 4 compares this work with the related works. We also show binary-class and nonlinear GSCCA. Section 5 describes the simulations and results.

Notations. Denote by N the number of samples. Let $X = [x_1, \ldots, x_N] \in \mathbb{R}^{n\times N}$ and $Y = [y_1, \ldots, y_N] \in \mathbb{R}^{c\times N}$ be centered variable sets, i.e., $\sum_{i=1}^{N} x_i = 0$, and $\sum_{i=1}^{N} y_i = 0$, where $x_i \in \mathbb{R}^n$ (or, $y_i \in \mathbb{R}^c$) is the ith column of X (or, Y), n and c are the data dimensionalities of X and Y sets, respectively. $\|\bullet\|$ is l^2-norm, and $\|\bullet\|_F$ denotes the Frobenius norm of a matrix. $Tr(\bullet)$ is trace operator, $A^{\dagger}$ denotes the pseudoinverse of matrix A, A^{T} is the transpose of matrix A, I is an identity matrix, and $e \in \mathbb{R}^N$ is a vector of all ones.

2 Canonical Correlation Analysis

Given N pairs of variables $(x_i, y_i), i = 1, 2, \ldots, N$, CCA seeks two optimal projection vectors $\widehat{l_x}$ and $\widehat{l_y}$ such that the correlation coefficient ρ between the variables $\widehat{l_x}^{\mathrm{T}} x_i$ and $\widehat{l_y}^{\mathrm{T}} y_i$, $i = 1, 2, \ldots, N$, is maximized. Thus, we have

$$\begin{aligned}\rho &= \frac{\widehat{l_x}^{\mathrm{T}} \sum_{i=1}^{N} x_i y_i^{\mathrm{T}} \widehat{l_y}}{\sqrt{\widehat{l_x}^{\mathrm{T}} \sum_{i=1}^{N} x_i x_i^{\mathrm{T}} \widehat{l_x}} \bullet \sqrt{\widehat{l_y}^{\mathrm{T}} \sum_{i=1}^{N} y_i y_i^{\mathrm{T}} \widehat{l_y}}} \\ &= \frac{\widehat{l_x}^{\mathrm{T}} X Y^{\mathrm{T}} \widehat{l_y}}{\sqrt{\widehat{l_x}^{\mathrm{T}} X X^{\mathrm{T}} \widehat{l_x}} \bullet \sqrt{\widehat{l_y}^{\mathrm{T}} Y Y^{\mathrm{T}} \widehat{l_y}}},\end{aligned} \tag{1}$$

where XY^{T} is the interset covariance matrix, XX^{T} and YY^{T} are the intraset covariance matrices. Then the multiple projection matrices $\widehat{L_X} = [\widehat{l_{x_1}} | \widehat{l_{x_2}} | \cdots | \widehat{l_{x_{d_1}}}] \in \mathbb{R}^{n\times d_1}$ and $\widehat{L_Y} = [\widehat{l_{y_1}} | \widehat{l_{y_2}} | \cdots | \widehat{l_{y_{d_2}}}] \in \mathbb{R}^{c\times d_2}$, satisfying $d_1 \le n$ and $d_2 \le c$, can be obtained from the following problems:

$$\begin{aligned}&\begin{pmatrix} XY^{\mathrm{T}}(YY^{\mathrm{T}})^{-1}YX^{\mathrm{T}} & 0 \\ 0 & YX^{\mathrm{T}}(XX^{\mathrm{T}})^{-1}XY^{\mathrm{T}} \end{pmatrix} \begin{pmatrix} \widehat{l_x} \\ \widehat{l_y} \end{pmatrix} \\ &= \lambda \begin{pmatrix} XX^{\mathrm{T}} & 0 \\ 0 & YY^{\mathrm{T}} \end{pmatrix} \begin{pmatrix} \widehat{l_x} \\ \widehat{l_y} \end{pmatrix},\end{aligned} \tag{2}$$

where λ is the canonical correlation to be optimized. CCA is originally derived to handle two data sets directly. By setting $Y \in \mathbb{R}^{c\times N}$ to be the specific class indicator matrix [3], [13], where c is the number of classes, CCA can then be extended for supervised FE in multilabel scenarios [3], [13],

[28], referred to as MCCA. In such case, MCCA only needs to compute $\widehat{L_X}$ for representing the sample set X and applies embedded data for subsequent visualization and classification. The commonly used class-label encoding approaches are one-of-c encoding [29], c-1 label encoding [29], $\{-1, 1\}$ encoding [30] and soft-label encoding [3]. The projection matrix $\widehat{L_X}$ can be obtained by solving the following generalized problem:

$$\begin{aligned} \underset{\widehat{L_X}}{Max}\ & Tr\big(\widehat{L_X}XY^{\mathrm{T}}(YY^{\mathrm{T}}+\alpha_+ I)^{-1}YX^{\mathrm{T}}\widehat{L_X}\big), \\ & S.t.\ \widehat{L_X}XX^{\mathrm{T}}\widehat{L_X}=I, \end{aligned} \tag{3}$$

where $\alpha_+ I(\alpha_+ > 0)$ is a multiply of identity matrix to prevent the overfitting and avoid the singularity of YY^{T}. Note that when the class-label encoding is properly assigned, MCCA is equivalent to LDA [3]. There exits an equivalence between MCCA and LS in multilabel classification scenarios [13]. For X and Y sets, the objective function and solution Ξ^*_{LS} of LS can be defined as

$$\underset{\Xi}{Min}\ \sum_{i=1}^{N}\|\Xi^{\mathrm{T}}x_i - y_i\|^2 = \|\Xi^{\mathrm{T}}X - Y\|_F^2,\ \ \Xi^*_{LS} = (XX^{\mathrm{T}})^{\dagger}XY^{\mathrm{T}}. \tag{4}$$

Based on the equivalence between MCCA and LS, the optimal solution of MCCA can be calculated by solving an LS problem, which is more efficient than directly solving the generalized eigen-problems of MCCA.

3 Group Sparse Canonical Correlation Analysis

3.1 Robust Sparse Representation

SR aims to compute the compact representation of images. Extensive studies have demonstrated the effectiveness of SR in representing and recognizing images.

3.1.1 Sparse Reconstruction Weights and SPP

For a set of samples $x_i, i = 1, 2, \ldots, N$ in c classes, SR represents each x_i using as few points in $X = [x_1, x_2, \ldots, x_N] \in \mathbb{R}^{n\times N}$ as possible [25]. Let $s_i = [s_{i,1}, \ldots, s_{i,i-1}, 0, s_{i,i+1}, \ldots, s_{i,N}]^{\mathrm{T}}$ be an n-dimensional coefficient vector, in which the ith entry of s_i is equal to 0 and $s_{i,j}(i \neq j)$ measures the contribution of each x_j for reconstructing x_i. Let $\mathrm{Card}\{s_{i,j}|s_{i,j} \neq 0\}$ be the set cardinality, i.e., the number of nonzero elements in s_i, the pseudo-l^0 norm of s_i, i.e.,$\|s_i\|_{l_0}$, can be define as

$$\underset{s_i}{Min}\ \|s_i\|_{l^0} = \mathrm{Card}\{s_{i,j}|s_{i,j} \neq 0\},\ S.t.\ x_i = Xs_i, \tag{5}$$

which is a combinatorial computational problem, i.e., NP-hard. Note that the above problem can be solved by approximating l^0-norm to l^1-norm (or, Lasso regularization) [31]. Then (5) can be written as

$$\underset{s_i}{Min}\ \|s_i\|_{l^1},\ S.t.\ \|x_i - Xs_i\| < \varepsilon,\ e^{\mathrm{T}}s_i = 1, \tag{6}$$

where ε is an error tolerance, $x_i - Xs_i$ is the compensation for reconstructing x_i, and the sum-to-one constraint enables the solution $S = [s_1^*, s_1^*, \ldots, s_1^*]$ to preserve some local characteristics of data [20], [25], where $s_i^*, i = 1, 2, \ldots, N$ are obtained optimal vectors. Thus, a weighted graph $G_D = (X, S)$ can be obtained, where S is an edge weight matrix representing the sparsity of data sets. The discriminative information can also be preserved by S due to nature of SR.

Based on the weight vectors s_i^*, SPP [25] was proposed by seeking a projection matrix $\mathrm{P} \in \mathbb{R}^{n\times d}$ that best preserve the optimal weight vector s_i^* by minimizing

$$\begin{aligned} \underset{\mathrm{P}}{Min}\ & \sum_{i=1}^{N}\left\|\mathrm{P}^{\mathrm{T}}x_i - \mathrm{P}^{\mathrm{T}}Xs_i^*\right\|^2 \\ & = Tr(\mathrm{P}^{\mathrm{T}}X(I - S - S^{\mathrm{T}} + S^{\mathrm{T}}S)X^{\mathrm{T}}\mathrm{P}), \end{aligned} \tag{7}$$

where d is the reduced dimension. According to [25], remakrable results are delivered by SPP for subspace learning, robust representation and recognition.

3.1.2 Group Sparse Reconstruction Weights

As elaborated in [23] and [32], samples of a class are highly correlated, so l^1-minimization is not the best choice for ensuring selection of all correlated samples from a particular group. To address this issue, group sparse classifier (GSC) [23], [32] was introduced. GSC assumes that training data of a particular class approximately form a linear basis for any sample of the class [23]. Let x_i^k be ith sample of the class k and $\widetilde{s}_i^k$ be the coefficient vector associated with the ith sample of class k, then for a new sample x_p^k from the kth class, we have

$$x_p^k = \widetilde{s}_1^k x_1^k + \widetilde{s}_2^k x_2^k + \cdots + \widetilde{s}_{N_k}^k x_{N_k}^k + \varepsilon,$$

where N_k is the sample number of class k. By extending (7) to all classes, we obtain $x_p = X\widetilde{s_p} + \varepsilon$, where $\widetilde{s_p} = [\widetilde{s}_1^1, \ldots, \widetilde{s}_{N_1}^1, \ldots, \widetilde{s}_1^k, \ldots, \widetilde{s}_{N_k}^k, \ldots, \widetilde{s}_1^c, \ldots, \widetilde{s}_{N_c}^c]^{\mathrm{T}}$. This assumption holds if $\widetilde{s}$ is group sparse [23]. That is, the solution of the inverse problem of $x_p = X\widetilde{s_p} + \varepsilon$ should have nonzero coefficients corresponding to a particular group of training samples and zero elsewhere. The solution is obtained by

$$\underset{s_p}{Min}\ \left\|\widetilde{s_p}\right\|_{l^{2,0}},\ S.t.\ \left\|x_p - X\widetilde{s_p}\right\| < \varepsilon,\ e^{\mathrm{T}}\widetilde{s_p} = 1, \tag{8}$$

where the mixed $l^{2,0}$-norm of $\widetilde{s_p} = [\widetilde{s}_1^1, \ldots, \widetilde{s}_{N_1}^1, \ldots, \widetilde{s}_1^k, \ldots, \widetilde{s}_{N_k}^k, \ldots, \widetilde{s}_1^c, \ldots, \widetilde{s}_{N_c}^c]^{\mathrm{T}}$ is defined as $||\widetilde{s_p}||_{l^{2,0}} = \sum_{k=1}^{c}\Pi(||\widetilde{s_p}|| > 0)$, where $\Pi(||\widetilde{s_p}||) = 1$ if $||\widetilde{s_p}|| > 0$. Note that constraint $e^{\mathrm{T}}\widetilde{s_p} = 1$ for each point p of X set is also imposed to the $l^{2,0}$-norm problem. It is easy to check that the above mixed $l^{2,0}$-norm problem is NP-hard. According to [32], this problem can be convexly relaxed by optimizing the following mixed $l^{2,1}$-norm problem:

$$\underset{s_p}{Min}\ \left\|\widetilde{s_p}\right\|_{l^{2,1}},\ S.t.\ \left\|x_p - X\widetilde{s_p}\right\| < \varepsilon,\ e^{\mathrm{T}}\widetilde{s_p} = 1, \tag{9}$$

where $||\widetilde{s_p}||_{l^{2,1}} = ||\widetilde{s}^1|| + ||\widetilde{s}^2|| + \cdots + ||\widetilde{s}^k|| + \cdots + ||\widetilde{s}^c||$ with $||\widetilde{s}^k|| = [\widetilde{s}_1^k, \ldots, \widetilde{s}_{N_k}^k]$. According to [23], [32], (9) can provide an approximation to (8), but the involved quadratic programming is computational expensive. To solve the $l^{2,0}$-norm problem directly, certain faster greedy algorithms, for example, group orthogonal matching pursuit (GOMP)

and block orthogonal matching pursuit (BOMP), are proposed in [32]. These greedy algorithms first identify the group with nonzero coefficients using block or group selection strategy, then the coefficients for the group indexes are estimated [32]. As a result, all weights for a particular class can be selected. Results verified the effectiveness and efficiency of these greedy algorithms, compared with convex optimization. In this work, we use the GOMP [32] to optimize the NP-hard problem and thus the groups are selected based on the highest individual correlation during iterations. Similarly, $\widetilde{S} = [\widetilde{s}_1^*, \widetilde{s}_2^*, \ldots, \widetilde{s}_N^*]$ can be defined as the edge weights of an adjacency graph $G_S = (X, \widetilde{S})$ on the X set, where $\widetilde{s}_i^*$ is optimal vector calculated from (9).

With analogous arguments, we obtain the group sparse weight vectors for reconstructing each point y_p in Y set from solving the following $l^{2,0}$-norm problem:

$$\underset{\widetilde{r}_p}{Min} \ \|\widetilde{r}_p\|_{l^{2,0}}, \ S.t. \ \|y_p - Y\widetilde{r}_p\| < \varepsilon, \ e^{\mathrm{T}}\widetilde{r}_p = 1, \tag{10}$$

where $\widetilde{r}_p = [\widetilde{r}_1^1, \ldots, \widetilde{r}_{N_1}^1, \ldots, \widetilde{r}_1^k, \ldots, \widetilde{r}_{N_k}^k, \ldots, \widetilde{r}_1^c, \ldots, \widetilde{r}_{N_c}^c]^{\mathrm{T}}$ with $||\widetilde{r}^k|| = [\widetilde{r}_1^k, \ldots, \widetilde{r}_{N_k}^k]$. The optimal solution $\widetilde{R} = [\widetilde{r}_1^*, \widetilde{r}_2^*, \ldots, \widetilde{r}_N^*] \in \mathbb{R}^{N \times N}$ over the Y set can also reflect some local geometric properties of data, where $\widetilde{r}_i^*$ is optimal vector calculated from (10). Similarly, $\widetilde{R}$ can be treated as an edge weight matrix of an adjacency graph $G_R = (Y, \widetilde{R})$ over Y set. We in next subsection detail the proposed algorithm. Since our technique integrates the group sparsity with CCA, this algorithm is referred to as group sparse CCA (GSCCA).

3.2 The GSCCA Objective Function

For given the variable sets X and Y, GSCCA similarly seeks to calculate two optimal projection matrices $\widehat{L_X} = [\widehat{l_{x_1}}|\widehat{l_{x_2}}|\cdots|\widehat{l_{x_{d_1}}}] \in \mathbb{R}^{n \times d_1}$ and $\widehat{L_Y} = [\widehat{l_{y_1}}|\widehat{l_{y_2}}|\cdots|\widehat{l_{y_{d_2}}}] \in \mathbb{R}^{c \times d_2}$ to represent X and Y, respectively. In extracting the informative features, motivated by SPP, GSCCA aims at achieving the projections that best preserve the group sparse coefficient vectors $\widetilde{s}_i^*$ and $\widetilde{r}_i^*$ addressed from the X and Y sets respectively, and at the same time maximizing the global interset covariance. This enables us to define the following optimization problem for calculating the optimal GSCCA projection axes:

$$\underset{\widehat{l_x}, \widehat{l_y}}{Max} \frac{\widehat{l_x}^{\mathrm{T}} \sum_{i=1}^{N} x_i y_i^{\mathrm{T}} \widehat{l_y}}{\sqrt{\widehat{l_x}^{\mathrm{T}} \sum_{i=1}^{N} \left(x_i - X\widetilde{s}_i^*\right)\left(x_i - X\widetilde{s}_i^*\right)^{\mathrm{T}} \widehat{l_x}} \cdot \sqrt{\widehat{l_y}^{\mathrm{T}} \sum_{i=1}^{N} \left(y_i - Y\widetilde{r}_i^*\right)\left(y_i - Y\widetilde{r}_i^*\right)^{\mathrm{T}} \widehat{l_y}}}. \tag{11}$$

Based on simple algebra formulations as [25], we have

$$\sum_{i=1}^{N} \left(x_i - X\widetilde{s}_i^*\right)\left(x_i - X\widetilde{s}_i^*\right)^{\mathrm{T}} = X(I - \widetilde{S} - \widetilde{S}^{\mathrm{T}} + \widetilde{S}^{\mathrm{T}}\widetilde{S})X^{\mathrm{T}}, \tag{12}$$

$$\sum_{i=1}^{N} \left(y_i - Y\widetilde{r}_i^*\right)\left(y_i - Y\widetilde{r}_i^*\right)^{\mathrm{T}} = Y\left(I - \widetilde{R} - \widetilde{R}^{\mathrm{T}} + \widetilde{R}^{\mathrm{T}}\widetilde{R}\right)Y^{\mathrm{T}}. \tag{13}$$

By substituting (12) and (13) into (11), we obtain

$$\begin{aligned} \underset{\widehat{l_x} \in \mathbb{R}^n, \widehat{l_y} \in \mathbb{R}^c}{Max} \ & \widehat{l_x}^{\mathrm{T}} X Y^{\mathrm{T}} \widehat{l_y}, \ S.t. \ \widehat{l_x}^{\mathrm{T}} X P_X X^{\mathrm{T}} \widehat{l_x} \\ & = \widehat{l_y}^{\mathrm{T}} Y P_Y Y^{\mathrm{T}} \widehat{l_y} = 1, \end{aligned} \tag{14}$$

where $P_X = I - \widetilde{S} - \widetilde{S}^{\mathrm{T}} + \widetilde{S}^{\mathrm{T}}\widetilde{S}$ and $P_Y = I - \widetilde{R} - \widetilde{R}^{\mathrm{T}} + \widetilde{R}^{\mathrm{T}}\widetilde{R}$. From (14), the projection matrices $\widehat{L_X}$ and $\widehat{L_Y}$ can be obtained. As a result, FE can be performed in the forms of $\widehat{L_X}^{\mathrm{T}} X$ and $\widehat{L_Y}^{\mathrm{T}} Y$. This enables us to use compact representations of points to subsequent classification or clustering.

3.3 Computational Analysis

Here, we show the method of computing $\widehat{L_X}$ and $\widehat{L_Y}$. Note that $e^{\mathrm{T}}\widetilde{s}_i^* = 1$ for each index i, we then have

$$\begin{aligned} & \sum_i \left[e^{\mathrm{T}}(I - \widetilde{S} - \widetilde{S}^{\mathrm{T}} + \widetilde{S}^{\mathrm{T}}\widetilde{S})\right]_i \\ & = \sum_i \left[e^{\mathrm{T}} I - e^{\mathrm{T}}\widetilde{S} - e^{\mathrm{T}}\widetilde{S}^{\mathrm{T}} + e^{\mathrm{T}}(\widetilde{S}^{\mathrm{T}}\widetilde{S})\right]_i \\ & = \sum_i (e^{\mathrm{T}} I)_i - \sum_i (e^{\mathrm{T}}\widetilde{S})_i - \sum_j e_i^{\mathrm{T}}\widetilde{s}_{i,j}^* + \sum_{j,v} e_v^{\mathrm{T}}\widetilde{s}_{i,v}^*\widetilde{s}_{j,v}^* \\ & = 1 - 1 - \sum_j e_i^{\mathrm{T}}\widetilde{s}_{i,j}^* + \sum_v \widetilde{s}_{i,v}^* = 0. \end{aligned} \tag{15}$$

Thus, the row-sums and column-sums of the symmetric matrix P_X are zeros. By a complete analogous argument, we obtain $\sum_i [e^{\mathrm{T}}(I - \widetilde{R} - \widetilde{R}^{\mathrm{T}} + \widetilde{R}^{\mathrm{T}}\widetilde{R})]_i = 0$ due to the fact that $e^{\mathrm{T}}\widetilde{r}_i^* = 1$ for each i. So P_X and P_Y can be considered as the graph Laplacian matrices as [33]. Thus by setting $W_{i,j}^{(X)} = (\widetilde{S} + \widetilde{S}^{\mathrm{T}} - \widetilde{S}^{\mathrm{T}}\widetilde{S})_{i,j}$ if $i \neq j$, and else 0, we can obtain $\mathrm{H}^{(X)} = D^{(X)} - W^{(X)}$, where $D^{(X)}$ is a diagonal matrix with entries $D_{ii}^{(X)} = \sum_j W_{i,j}^{(X)}$ and $\mathrm{H}^{(X)}$ is the graph Laplacian matrix of $W^{(X)}$. It should be noted that similar definitions exist for P_Y, that is $W_{i,j}^{(Y)} = (\widetilde{R} + \widetilde{R}^{\mathrm{T}} - \widetilde{R}^{\mathrm{T}}\widetilde{R})_{i,j}$ when $i \neq j$, and else 0. Also we can have $D_{ii}^{(Y)} = \sum_j W_{i,j}^{(Y)}$ and $\mathrm{H}^{(Y)} = D^{(Y)} - W^{(Y)}$. So the vectors $\widehat{l_x}$ and $\widehat{l_y}$ can be obtained from

$$\begin{aligned} \underset{\widehat{l_x} \in \mathbb{R}^n, \widehat{l_y} \in \mathbb{R}^c}{Max} \ & \widehat{l_x}^{\mathrm{T}} X Y^{\mathrm{T}} \widehat{l_y}, \ S.t. \ \widehat{l_x}^{\mathrm{T}} X \mathrm{H}^{(X)} X^{\mathrm{T}} \widehat{l_x} \\ & = \widehat{l_y}^{\mathrm{T}} Y \mathrm{H}^{(Y)} Y^{\mathrm{T}} \widehat{l_y} = 1. \end{aligned} \tag{16}$$

The Lagrangian function of (16) with variables $\widehat{l_x}$, $\widehat{l_y}, \lambda_x$, and λ_y can then be expressed as

$$\begin{aligned} \widehat{J}(\widehat{l_x}, \widehat{l_y}, \lambda_x, \lambda_y) = {} & \widehat{l_x}^{\mathrm{T}} X Y^{\mathrm{T}} \widehat{l_y} - \frac{\lambda_x}{2}\left(\widehat{l_x}^{\mathrm{T}} X \mathrm{H}^{(X)} X^{\mathrm{T}} \widehat{l_x} - 1\right) \\ & - \frac{\lambda_y}{2}\left(\widehat{l_y}^{\mathrm{T}} Y \mathrm{H}^{(Y)} Y^{\mathrm{T}} \widehat{l_y} - 1\right). \end{aligned} \tag{17}$$

By taking derivatives to variables $\widehat{l_x}$ and $\widehat{l_y}$, we have

$$\begin{aligned} \partial\widehat{J}/\partial\widehat{l_x} &= XY^{\mathrm{T}}\widehat{l_y} - \lambda_x X \mathrm{H}^{(X)} X^{\mathrm{T}} \widehat{l_x} = 0 \ \Rightarrow \ XY^{\mathrm{T}}\widehat{l_y} \\ &= \lambda_x X \mathrm{H}^{(X)} X^{\mathrm{T}} \widehat{l_x}, \end{aligned} \tag{18}$$

$$\begin{aligned} \partial\widehat{J}/\partial\widehat{l_y} &= YX^{\mathrm{T}}\widehat{l_x} - \lambda_y Y \mathrm{H}^{(Y)} Y^{\mathrm{T}} \widehat{l_y} = 0 \ \Rightarrow \ YX^{\mathrm{T}}\widehat{l_x} \\ &= \lambda_y Y \mathrm{H}^{(Y)} Y^{\mathrm{T}} \widehat{l_y}. \end{aligned} \tag{19}$$

Since $\widehat{l_x}^{\mathrm{T}} X H^{(X)} X^{\mathrm{T}} \widehat{l_x} = 1$ and $\widehat{l_y}^{\mathrm{T}} Y H^{(Y)} Y^{\mathrm{T}} \widehat{l_y} = 1$, by subtracting $\widehat{l_y}^{\mathrm{T}}\times$ (19) from $\widehat{l_x}^{\mathrm{T}}\times$ (18) and zeroing it, we get

$$\begin{aligned} 0 &= \widehat{l_x}^{\mathrm{T}} X Y^{\mathrm{T}} \widehat{l_y} - \lambda_x \widehat{l_x}^{\mathrm{T}} X \mathrm{H}^{(X)} X^{\mathrm{T}} \widehat{l_x} \\ &\quad - \widehat{l_y}^{\mathrm{T}} Y X^{\mathrm{T}} \widehat{l_x} + \lambda_y \widehat{l_y}^{\mathrm{T}} Y \mathrm{H}^{(Y)} Y^{\mathrm{T}} \widehat{l_y} \\ &= \lambda_y \widehat{l_y}^{\mathrm{T}} Y \mathrm{H}^{(Y)} Y^{\mathrm{T}} \widehat{l_y} - \lambda_x \widehat{l_x}^{\mathrm{T}} X \mathrm{H}^{(X)} X^{\mathrm{T}} \widehat{l_x} \\ &= \lambda_y - \lambda_x. \end{aligned} \tag{20}$$

Let $\lambda_y = \lambda_x = \lambda$, from the equation of (19), we have

$$\frac{1}{\lambda}(Y \mathrm{H}^{(Y)} Y^{\mathrm{T}})^{-1} Y X^{\mathrm{T}} \widehat{l_x} = \widehat{l_y}. \tag{21}$$

When $Y = [y_1, \ldots, y_N]$ is set to be a class label set, GSCCA can be applied for multiclass classification. In such cases, GSCCA only needs to compute $\widehat{L_X}$ for representing sample set X. This work mainly discusses multiclass GSCCA. Hereinafter we will still name the multiclass GSCCA as GSCCA. Note that binary-class GSCCA problem will be discussed and evaluated in Section 5.1, that is, both X and Y are sample sets. By substituting $\widehat{l_y}$ into the equation of (18) and left multipling both sides of (18) by $\widehat{l_x}^{\mathrm{T}}$, we obtain the following problem:

$$\underset{\widehat{l_x} \in \mathbb{R}^n}{Max}\ \lambda_x^2 = \frac{\widehat{l_x}^{\mathrm{T}} X Y^{\mathrm{T}} (Y \mathrm{H}^{(Y)} Y^{\mathrm{T}})^{-1} Y X^{\mathrm{T}} \widehat{l_x}}{\widehat{l_x}^{\mathrm{T}} X \mathrm{H}^{(X)} X^{\mathrm{T}} \widehat{l_x}}. \tag{22}$$

Since $Y \mathrm{H}^{(Y)} Y^{\mathrm{T}}$ is a positive and semidefinite matrix, the inverse of $Y \mathrm{H}^{(Y)} Y^{\mathrm{T}}$ may be singular. So a regularized term $\alpha_+ I$ with a parameter α_+ is added to $Y \mathrm{H}^{(Y)} Y^{\mathrm{T}}$ for the stability consideration. Note that, in multiclass GSCCA, Y is set to be the class label matrix and each paired $y_{l,i}$ and $y_{l,j}$ of class l are the same and are equally important, so computing the sparse weight $\widetilde{R}$ is meaningless. This paper simply sets $\mathrm{H}^{(Y)}$ to be an identity matrix due to the computational consideration. Since $\mathrm{H}^{(X)} = D^{(X)} - W^{(X)}$, the above problem can be rewritten as

$$\begin{aligned} \underset{\widehat{l_x} \in \mathbb{R}^n}{Max} \big\{ &(1-\ell_+) \widehat{l_x}^{\mathrm{T}} X Y^{\mathrm{T}} (Y \mathrm{H}^{(Y)} Y^{\mathrm{T}} + \alpha_+ I)^{-1} Y X^{\mathrm{T}} \widehat{l_x} \\ &+ \ell_+ \widehat{l_x}^{\mathrm{T}} X W^{(X)} X^{\mathrm{T}} \widehat{l_x} \big\} \Big/ \big\{ \widehat{l_x}^{\mathrm{T}} X D^{(X)} X^{\mathrm{T}} \widehat{l_x} \big\}, \end{aligned} \tag{23}$$

where $\ell_+ (\in [0,1])$ is a parameter for balancing the tradeoff between $X Y^{\mathrm{T}} (Y \mathrm{H}^{(Y)} Y^{\mathrm{T}} + \alpha_+ I)^{-1} Y X^{\mathrm{T}}$ and $X W^{(X)} X^{\mathrm{T}}$, and $D^{(X)}$ is an identity matrix. Next, we show the solutions.

3.4 Effective Solution Schemes

This section elaborates two solution strategies to calculate the GSCCA projection matrix for embeddings.

3.4.1 Computing GSCCA via Trace Ratio Optimization

We first use the TR optimization to solve (23). Under TR criterion, the orthogonal constraint $\widehat{L_X}^{\mathrm{T}} \widehat{L_X} = I$ is imposed. Let $\widetilde{\Upsilon^{(X)}} = (1-\ell_+) Y^{\mathrm{T}} (Y \mathrm{H}^{(Y)} Y^{\mathrm{T}} + \alpha_+ I)^{-1} Y + \ell_+ W^{(X)}$, (23) can then be expressed in matrix form as the following TR criterion [26], [27], [34] based problem:

$$\underset{\widehat{L_X} \in \mathbb{R}^{n \times d_1},\ \widehat{L_X}^{\mathrm{T}} \widehat{L_X} = I}{Max} \frac{Tr\big(\widehat{L_X}^{\mathrm{T}} X \widetilde{\Upsilon^{(X)}} X^{\mathrm{T}} \widehat{L_X}\big)}{Tr\big(\widehat{L_X}^{\mathrm{T}} X D^{(X)} X^{\mathrm{T}} \widehat{L_X}\big)}. \tag{24}$$

To solve TR problem, Guo et al. [34] have proved that the global optimum of TR problem can be equivalently obtained by solving a trace difference (TD) problem, i.e., to find the best TR value λ^* and optimum matrix $\widehat{L_X}^*$ is equivalent to find the zero point of the TD function:

$$\begin{aligned} F(\lambda) &= \arg \max_{\widehat{L_X}^{\mathrm{T}} \widehat{L_X} = I} Tr\big(\widehat{L_X}^{\mathrm{T}} (X \widetilde{\Upsilon^{(X)}} X^{\mathrm{T}} - \lambda^* X D^{(X)} X^{\mathrm{T}}) \widehat{L_X}\big) \\ &= 0. \end{aligned}$$

Then the optimal $\widehat{L_X}^*$ is given by

$$\widehat{L_X}^* = \arg \max_{\widehat{L_X}^{\mathrm{T}} \widehat{L_X} = I} Tr\big(\widehat{L_X}^{\mathrm{T}} \big(X \widetilde{\Upsilon^{(X)}} X^{\mathrm{T}} - \lambda^* X D^{(X)} X^{\mathrm{T}}\big) \widehat{L_X}\big). \tag{25}$$

Another effective algorithm called iterative trace ratio (ITR) [27] was recently proposed to solve TR problem. ITR tackles the TR problem by directly optimizing the objective $Tr(\widehat{L_X}^{\mathrm{T}} X \widetilde{\Upsilon^{(X)}} X^{\mathrm{T}} \widehat{L_X}) / Tr(\widehat{L_X}^{\mathrm{T}} X D^{(X)} X^{\mathrm{T}} \widehat{L_X})$ when the column vectors of $\widehat{L_X}$ are orthogonal together. For a given λ^v at each iteration v, the projection matrix $\widehat{L_X}^v$ is obtained by solving the following TD problem:

$$\widehat{L_X}^v = \arg \max_{\widehat{L_X}^{\mathrm{T}} \widehat{L_X} = I} Tr\big(\widehat{L_X}^{\mathrm{T}} \big(X \widetilde{\Upsilon^{(X)}} X^{\mathrm{T}} - \lambda^v X D^{(X)} X^{\mathrm{T}}\big) \widehat{L_X}\big). \tag{26}$$

Then λ^{v+1} is renewed using $\widehat{L_X}^v$ as:

$$\lambda^{v+1} = Tr\big(\widehat{L_X}^{v\mathrm{T}} X \widetilde{\Upsilon^{(X)}} X^{\mathrm{T}} \widehat{L_X}^v\big) / Tr\big(\widehat{L_X}^{v\mathrm{T}} X D^{(X)} X^{\mathrm{T}} \widehat{L_X}^v\big)$$

until convergence. Mathematical proof show that ITR converges to the global optimum [26]. Note that ITR initializes $\widehat{L_X}^0$ to be an arbitrary orthogonal matrix, so ITR maybe unstable due to randomness. And it is difficult to guarantee the orthogonal property of initialized $\widehat{L_X}^0$ and a bad initialization may greatly increase the number of iteration. This work initials $\lambda^0 = 0$ instead of initializing $\widehat{L_X}^0$ to be columnly orthogonal matrix as [26]. The procedures of using ITR to solve GSCCA are described in Algorithm 1. Under the TR criterion, the projection matrix $\widehat{L_X}$ is orthogonal, thus similarity preservation and more specific solution can be obtained [26], [27]. We name this scheme as orthogonal GSCCA (o-GSCCA).

3.4.2 Computing GSCCA via Eigen-Decomposition

In this case, we aim to solve $Max_{\widehat{L_X}}\ Tr(\widehat{L_X}^{\mathrm{T}} X \widetilde{\Upsilon^{(X)}} X^{\mathrm{T}} \widehat{L_X})$, $S.t. \widehat{L_X}^{\mathrm{T}} X D^{(X)} X^{\mathrm{T}} \widehat{L_X} = I$. The projection axes in $\widehat{L_X}$ include the leading eigenvectors of matrix $(X D^{(X)} X^{\mathrm{T}})^{\dagger} X \widetilde{\Upsilon^{(X)}} X^{\mathrm{T}}$. By performing singular value decomposition (SVD) [35] to $X D^{(X)} X^{\mathrm{T}}$, we can obtain

$$X D^{(X)} X^{\mathrm{T}} = U \begin{pmatrix} \Sigma_t^2 & 0 \\ 0 & 0 \end{pmatrix} U^{\mathrm{T}}, \tag{27}$$

where U is an orthogonal matrix and Σ_t^2 is a diagonal matrix. Let $U = [U_1, U_2]$ be a partition of U such that $U_1 \in \mathbb{R}^{n \times t}$ and

$U_2 \in \mathbb{R}^{n\times(n-t)}$, where t is the rank of $XD^{(X)}X^{\mathrm{T}}$ and U_2 lies in the null space of $XD^{(X)}X^{\mathrm{T}}$, i.e., $U_2^{\mathrm{T}}XD^{(X)}X^{\mathrm{T}}U_2 = 0$. Since $\widetilde{\Upsilon^{(X)}}$ is a symmetrical matrix, it can be decomposed by Cholesky decomposition as $\widetilde{\Upsilon^{(X)}} = \widetilde{G}\widetilde{G}^{\mathrm{T}}$, where $\widetilde{G}$ is a lower triangular matrix. Let $\widetilde{H}_b = X\widetilde{G}$, we can have $X\widetilde{\Upsilon^{(X)}}X^{\mathrm{T}} = \widetilde{H}_b\widetilde{H}_b^{\mathrm{T}}$. Denote $H = \sum_t^{-1}U_1^{\mathrm{T}}\widetilde{H}_b$ and let $H = P\sum_b Q^{\mathrm{T}}$ be the *SVD* of H, where P and Q are orthogonal matrices and $\sum_b$ is a diagonal matrix with rank q, we obtain

$$\sum_t^{-1}U_1^{\mathrm{T}}\widetilde{H}_b\widetilde{H}_b^{\mathrm{T}}U_1\sum_t^{-1} = HH^{\mathrm{T}} = P\sum_b^2 P^{\mathrm{T}}. \tag{28}$$

According to the equations in (27) and (28), we have

$$\begin{aligned}
&(XD^{(X)}X^{\mathrm{T}})^{\dagger}(X\widetilde{\Upsilon^{(X)}}X^{\mathrm{T}})\\
&= U\begin{pmatrix}\sum_t^{-1}\sum_t^{-1} & 0\\ 0 & 0\end{pmatrix}U^T\widetilde{H}_b\widetilde{H}_b^{\mathrm{T}}U\begin{pmatrix}\sum_t^{-1}\sum_t & 0\\ 0 & 0\end{pmatrix}U^{\mathrm{T}}\\
&= U\begin{pmatrix}\sum_t^{-1} & 0\\ 0 & 0\end{pmatrix}P\sum_b^2 P^{\mathrm{T}}\begin{pmatrix}\sum_t & 0\\ 0 & 0\end{pmatrix}U^{\mathrm{T}}\\
&= U\begin{pmatrix}\sum_t^{-1}P & 0\\ 0 & I\end{pmatrix}\begin{pmatrix}\sum_b^2 & 0\\ 0 & 0\end{pmatrix}\begin{pmatrix}P^{\mathrm{T}}\sum_t & 0\\ 0 & I\end{pmatrix}U^{\mathrm{T}}.
\end{aligned} \tag{29}$$

Let $L^*_{GSCCA} = U_1\sum_t^{-1}P_q$, where P_q consists of the first q columns of P when only the first q diagonal elements of $\sum_b$ are nonzero, we have $(XD^{(X)}X^{\mathrm{T}})^{\dagger}(X\widetilde{\Upsilon^{(X)}}X^{\mathrm{T}})L^*_{GSCCA} = \sum_b^2 L^*_{GSCCA}$. That is, L^*_{GSCCA} includes the optimal projection vectors of GSCCA. Note that the cost function of GSCCA in (23) can also be treated as a ratio trace (RT) problem [26], [27], which can be efficiently solved by using the generalized eigen-decomposition (GED) [35]. We still refer to this solution scheme of using GED as GSCCA. Similar to [3], [13], based on the above eigen-decomposition, we can easily establish an equivalent relationship between GSCCA and LS, as addressed in Section 4.1.

Algorithm 1: TR Criterion based Multi-Class GSCCA

Input: Sample set $X = [x_1,...,x_N] \in \mathbb{R}^{n\times N}$ and class label set $Y = [y_1,...,y_m] \in \mathbb{R}^{c\times N}$; the reduced dimension d.

Output: The transforming matrix $\widehat{L_X} \in \mathbb{R}^{n\times d}$.

1. Construct weight matrices $\tilde{S}$ and $\tilde{R}$ by Eqs.8 and 10;
2. Get $P_X = I - \tilde{S} - \tilde{S}^{\mathrm{T}} + \tilde{S}^{\mathrm{T}}\tilde{S}$, $P_Y = I - \tilde{R} - \tilde{R}^{\mathrm{T}} + \tilde{R}^{\mathrm{T}}\tilde{R}$, $\mathrm{H}^{(X)}$ and $\mathrm{H}^{(Y)}$;
3. TR optimization for solving the TR problem in Eq.24:
   3.1. Initialize step $v = 0$ and $\lambda^v = 0$;
   3.2: Solve eigen-problem $\left(X\widetilde{\Upsilon^{(X)}}X^{\mathrm{T}} - \lambda^v XD^{(X)}X^{\mathrm{T}}\right)\pi = \tau^v\pi$ and obtain the vectors $\{\pi_\delta^v\}_{\delta=1}^d$ of $\left(X\widetilde{\Upsilon^{(X)}}X^{\mathrm{T}} - \lambda^v XD^{(X)}X^{\mathrm{T}}\right)$ by eigen-decomposition;
   3.3. Projection matrix $\widehat{L_X}^v = \{\pi_\delta^v\}_{\delta=1}^d$ at step v includes the eigenvectors according to the d largest eigenvalues $\{\tau_\delta^v\}_{\delta=1}^d$ of $\left(X\widetilde{\Upsilon^{(X)}}X^{\mathrm{T}} - \lambda^v XD^{(X)}X^{\mathrm{T}}\right)$;
   3.4. Update $\lambda^{v+1} = Tr\left(\widehat{L_X}^{v\mathrm{T}}X\widetilde{\Upsilon^{(X)}}X^{\mathrm{T}}\widehat{L_X}^v\right) / Tr\left(\widehat{L_X}^{v\mathrm{T}}XD^{(X)}X^{\mathrm{T}}\widehat{L_X}^v\right)$;
   3.5. If $\left|\lambda^{v+1} - \lambda^v\right| < \varepsilon$, o-GSCCA converges; else $v = v+1$, steps 3.2-3.4 repeat;
4. Output $\widehat{L_X}^* = \max_{\widehat{L_X}^{\mathrm{T}}\widehat{L_X}=I} Tr\left(\widehat{L_X}^{\mathrm{T}}\left(X\widetilde{\Upsilon^{(X)}}X^{\mathrm{T}} - \lambda^* XD^{(X)}X^{\mathrm{T}}\right)\widehat{L_X}\right)$.

3.4.3 Computational Complexity of GSCCA and o-GSCCA

The complexity of our algorithms mainly includes two parts. The first part is to calculate the GS weights. Considering that traditional convex optimization for sparse problems is time-consuming [32], to speedup this process, an greedy GOMP algorithm [23], [32] is used to obtain the GS weights. At each iteration of GOMP, the group average of the correlations is calculated and the group with the highest average is selected, and GOMP is proved to be much more efficient than convex optimization with effectiveness guaranteed [23], [32]. The second part is to compute the projection matrix for transformation. For GSCCA, the projection matrix is obtained by eigen-decomposition and the computational complexity is $O(n^3)$ or $O(n^2d)$, if the first d eigenvectors are to be achieved. For o-GSCCA, eigen-composition is also performed in the Step 3 of Algorithm 1, so the computational complexity at each iteration is also $O(n^2d)$. If o-GSCCA converges after t iterations, the total complexity of o-GSCCA is $O(n^2dt)$.

4 Related Work: Connection and Discussion

This section discusses some issues related to our method.

4.1 Equivalence between GSCCA and LS

4.1.1 Relation to Least Squares

Further to the LS problem in (4), if we choose a class indicator matrix as $Y = \widetilde{G}^{\mathrm{T}}$, we have $XY^{\mathrm{T}} = \widetilde{H}_b$. Thus, the solution of LS can be rewritten as $\Xi^*_{LS} = (XX^{\mathrm{T}})^{\dagger}\widetilde{H}_b$, which can also be equivalently reformulated as

$$\begin{aligned}
(XX^{\mathrm{T}})^{\dagger}\widetilde{H}_b &= U\begin{pmatrix}\sum_t^{-1}\sum_t^{-1} & 0\\ 0 & 0\end{pmatrix}U^{\mathrm{T}}\widetilde{H}_b\\
&= U_1\sum_t^{-1}\left(\sum_t^{-1}U_1^{\mathrm{T}}\widetilde{H}_b\right)\\
&= U_1\sum_t^{-1}H = U_1\sum_t^{-1}P\sum_b Q^{\mathrm{T}}\\
&= L^*_{GSCCA}\sum_b Q^{\mathrm{T}}.
\end{aligned} \tag{30}$$

Since Q is an orthogonal matrix, it can be neglected if the similarity of samples is based on euclidean distance. So the main difference between L^*_{GSCCA} and Ξ^*_{LS} is the diagonal matrix $\sum_b$. If $\sum_b$ is an identity matrix, we can have $\Xi^*_{LS} = L^*_{GSCCA}$. This can only be hold when

$$\mathrm{rank}(XX^{\mathrm{T}}) - \mathrm{rank}(X\widetilde{\Upsilon^{(X)}}X^{\mathrm{T}}) = \mathrm{rank}(XX^{\mathrm{T}} - X\widetilde{\Upsilon^{(X)}}X^{\mathrm{T}})$$

[3], [13]. Otherwise, the LS problem can be solved by the two-stage approach [43] detailed in the next section.

4.1.2 Two-Stage Approach for Optimization

For the two-stage approach, one first solves an LS problem by regressing X on Y^{T}, i.e., projecting high-dimensional data set to lower dimensions. We can then calculate an auxiliary matrix $M \in \mathbb{R}^{c\times c}$ and its SVD. Finally, the optimal projection matrix can be obtained from the SVD of M. Since the size of matrix M is small, the computational cost for calculating the SVD of M is lower. The steps of two-stage approach can be summarized as

1. solve the LS problem $Min_{\Xi} \|\Xi^T X - Y\|_F^2$;
2. let $\widehat{X} = \Xi^{\mathrm{T}} X$, by defining an auxiliary matrix M as $M = \widehat{X} Y^{\mathrm{T}}$, we can then calculate the SVD of M as $M = U_M \Sigma_M U_M^{\mathrm{T}}$ and set $V_M^* = U_M \Sigma_M^{-1/2}$; and
3. the projection matrix is given by $V_{TS}^* = \Xi V_M^*$.

Next, we elaborate that V_{TS}^* is equivalent to that in (29). By solving the LS problem, we can have $\Xi = (XX^{\mathrm{T}})^{-1} XY^{\mathrm{T}}$. We thus obtain $\widehat{X} = \Xi^{\mathrm{T}} X = YX^{\mathrm{T}}(XX^{\mathrm{T}})^{-1} X$ and an auxiliary matrix M is defined as

$$M = \widehat{X} Y^{\mathrm{T}} = YX^{\mathrm{T}}(XX^{\mathrm{T}})^{-1} XY^{\mathrm{T}} = \widetilde{H_b}^{\mathrm{T}} U_1 \sum\nolimits_t^{-1} \sum\nolimits_t^{-1} U_1^{\mathrm{T}} \widetilde{H_b}. \tag{31}$$

The second "=" holds as $\widetilde{H_b} = XY^{\mathrm{T}}$ and $XX^{\mathrm{T}} = U_1 \sum_t^2 U_1^{\mathrm{T}}$. Since $H = \sum_t^{-1} U_1^{\mathrm{T}} \widetilde{H_b}$ and its SVD is $H = P \sum_b Q^{\mathrm{T}}$, we have $M = H^{\mathrm{T}} H = Q \sum_b^2 Q^{\mathrm{T}}$. Equation (31) indicates that $Q \sum_b^2 Q^{\mathrm{T}}$ is the SVD of M, we thus have $V_M^* = Q \sum_b^{-1}$ and the optimal projection matrix by two-stage approach can be given by

$$\begin{aligned} V_{TS}^* = \Xi V_M^* &= (XX^{\mathrm{T}})^{-1} XY^{\mathrm{T}} Q \sum\nolimits_b^{-1} \\ &= U_1 \sum\nolimits_t^{-1} \left(\sum\nolimits_t^{-1} U_1^{\mathrm{T}} \widetilde{H_b} \right) Q \sum\nolimits_b^{-1} \\ &= U_1 \sum\nolimits_t^{-1} P \sum\nolimits_b Q^{\mathrm{T}} Q \sum\nolimits_b^{-1} = U_1 \sum\nolimits_t^{-1} P, \end{aligned} \tag{32}$$

which is just equivalent to the optimal solution in (29).

4.2 Connection between MCCA, LDA, SPP, and Multiclass GSCCA

Here we describe the inherent relationship among MCCA [3], SPP [25] and our GSCCA. Recalling the criterion of GSCCA in (23, if ℓ_+ is set to 0 and $\mathrm{H}^{(Y)}$ is set to be the identity matrix, (23) can be converted to

$$\begin{aligned} &\underset{\widehat{L_X}}{Max}\ Tr\left(\widehat{L_X}^{\mathrm{T}} XY^{\mathrm{T}} (YY^{\mathrm{T}} + \alpha_+ I)^{-1} YX^{\mathrm{T}} \widehat{L_X} \right), \\ &S.t.\ \widehat{L_X}^{\mathrm{T}} XX^{\mathrm{T}}\ \widehat{L_X} = I, \end{aligned} \tag{33}$$

which is just the criterion of MCCA. Thus, MCCA can be considered as a special case of our GSCCA. Based on the relationship between MCCA and LDA [3], when a proper label encoding is used, GSCCA with $\ell_+ = 0$ can also be equivalent to LDA. Similarly, when ℓ_+ is set to constant 1, we obtain the following problem from (23):

$$\begin{aligned} &\underset{\widehat{L_X}}{Max}\ Tr\left(\widehat{L_X}^{\mathrm{T}} XW^{(X)} X^{\mathrm{T}} \widehat{L_X} \right) = \\ &Tr\left(\widehat{L_X}^{T} X \left(\widetilde{S} + \widetilde{S}^{\mathrm{T}} - \widetilde{S}^{\mathrm{T}} \widetilde{S} \right) X^{\mathrm{T}} \widehat{L_X} \right), S.t.\ \widehat{L_X}^{\mathrm{T}} XX^{(T)} \widehat{L_X} = I, \end{aligned} \tag{34}$$

which is just the problem of SPP. So SPP is also treated as a special case of our GSCCA method.

4.3 GSCCA in Binary-Class Case and Evaluation

Note that GSCCA is originally proposed for handling two data sets $X = [x_1, \ldots, x_m] \in \mathbb{R}^{n \times N}$ and $Y = [y_1, \ldots, y_m] \in \mathbb{R}^{c \times N}$. To enable GSCCA to handle multiclass problems, Y set is set to be a class label set, which includes the class labels of samples. This section mainly considers GSCCA in binary-class case, referred to as binary GSCCA (BGSCCA). For BGSCCA, X and Y are all sample sets, so BGSCCA also needs to obtain $\widehat{L_Y} = [\widehat{l_{y_1}} | \widehat{l_{y_2}} | \cdots | \widehat{l_{y_{d_2}}}]$ to represent the samples of Y set. Also, the weight matrices $P_X = I - \widetilde{S} - \widetilde{S}^{\mathrm{T}} + \widetilde{S}^{\mathrm{T}} \widetilde{S}$ and $P_Y = I - \widetilde{R} - \widetilde{R}^{\mathrm{T}} + \widetilde{R}^{\mathrm{T}} \widetilde{R}$ need to be calculated for representing sample sets X and Y, respectively. Note that $\mathrm{H}^{(Y)}$ will be computed from P_Y instead of being simply set as an identity matrix. According to (18), we have

$$\frac{1}{\lambda} \left(X \mathrm{H}^{(X)} X^{\mathrm{T}} \right)^{-1} XY^{\mathrm{T}} \widehat{l_y} = \widehat{l_x}. \tag{35}$$

Since $\mathrm{H}^{(Y)} = D^{(Y)} - W^{(Y)}$, by substituting $\widehat{l_x}$ into (19) and left multipling both sides of (18) by $\widehat{l_y}^{\mathrm{T}}$, we can similarly obtain the following problem as (23):

$$\underset{\widehat{l_y} \in \mathbb{R}^c}{Max}\ \lambda_y^2 = \frac{\widehat{l_y}^{\mathrm{T}} Y \widetilde{\Upsilon^{(Y)}} Y^{\mathrm{T}} \widehat{l_y}}{\widehat{l_y}^{\mathrm{T}} Y D^{(Y)} Y^{\mathrm{T}} \widehat{l_y}}, \tag{36}$$

where $\widetilde{\Upsilon^{(Y)}} = (1 - \ell_+) X^{\mathrm{T}} (X \mathrm{H}^{(X)} X^{\mathrm{T}} + \alpha_+ I)^{-1} X + \ell_+ W^{(Y)}$, $\alpha_+ I$ is a regularization factor and $D^{(Y)}$ is an identity matrix. Note that $\widehat{L_Y}$ can be similarly gained as computing $\widehat{L_X}$ by eigen-decomposition or TR optimization. We still call the binary-class GSCCA as BGSCCA if eigen-decomposition is used. We refer to the ITR-based BGSCCA as orthogonal o-BGSCCA. Note that under TR criterion, the orthogonal constraint $\widehat{L_Y}^{\mathrm{T}} \widehat{L_Y} = I$ is always imposed.

We examine BGSCCA and o-BGSCCA for multivariate visualization. The results are compared with LDA, principal component analysis (PCA), maximum margin criterion (MMC) [36], CCA, MCCA, ITR-based LDA (TR-LDA) [26] and SPP. This study uses the synthetic control chart time series data set, or simply synthetic-control (Available at http://archive.ics.uci.edu/ml/data sets/Synthetic+Control+Chart+Time+Series) from the UCI ML repository. The data set has 600 examples of synthetically generated control charts included in six different classes. So each class has 100 samples. The experimental settings are detailed in Section 6. For visualization by CCA, BGSCCA, and o-BGSCCA, we create a binary-class problem by merging the first, fourth, and fifth classes into a single set (X), and the remaining classes form another sample set (Y). In the simulations, we assume X and Y sets have zero means. For PCA, LDA, MMC, MCCA, TR-LDA, and SPP, each kind of control chart corresponds to a single class. In our study, 20 points from each kind of control chart (totally 120 examples) are selected for learning the basis vectors. After the projection matrix of each method is obtained, all data are projected onto the projection directions for computing low-dimensional embeddings. For CCA, BGSCCA, and o-BGSCCA, paired matrices $\widehat{L_X}$ and $\widehat{L_Y}$ are achieved. Then X and Y sample sets are represented by $\widehat{L_X}$ and $\widehat{L_Y}$, respectively. For feature reduction, we set the number of reduced dimensions $d_1 = d_2 = 2$ in our simulation. Then 2D representations are got.

We illustrate the 2D embedding of each method in Fig. 1. Note that we still use different colors and shapes to represent different kinds of control charts. We have the

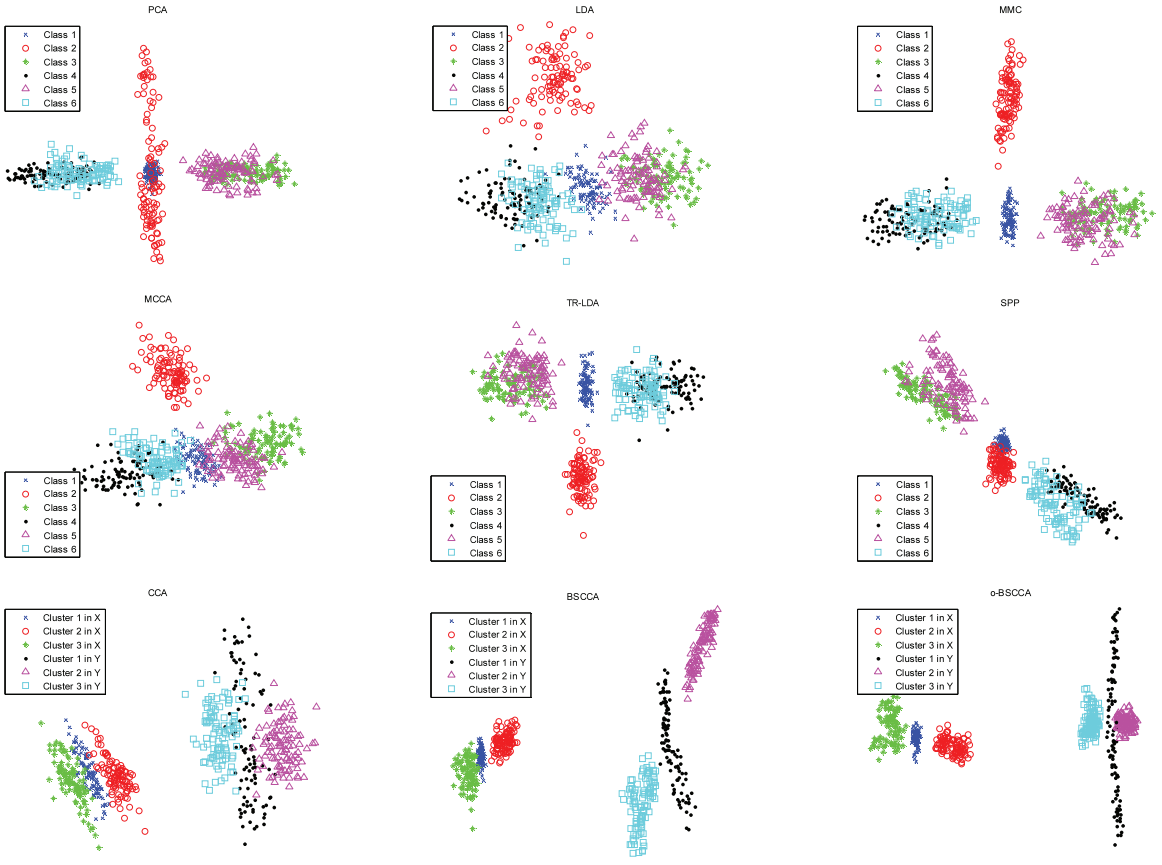


Fig. 1. The 2D embedding result of each method on the synthetic-control data set.

following observations. 1) PCA, LDA, MMC, MCCA, SPP, and TR-LDA represent partial data well, but most points of different control charts are still congregated in the embedding spaces. Specifically, these methods cannot deliver separated embeddings of the fourth and sixth classes effectively. The embedded data of the third and fifth classes are also mixed in their reduced spaces. 2) Compared with PCA, LDA, MMC, MCCA, SPP, and TR-LDA, CCA represents most of points appropriately and exhibits better result. But CCA still suffers from the problem of congregating partial data. On the contrary, our proposed BGSCCA and o-BGSCCA give clearer organizations and can separate the data into their natural clusters with higher intercluster and interclass separation. That is, BGSCCA and o-BGSCCA can implicitly emphasize the natural clusters of points within each class and give separated clusters between different classes.

4.4 Kernelized GSCCA for Nonlinear FE

This section considers nonlinear GSCCA. We call the kernelized GSCCA as KGSCCA that seeks a projection matrix $\widehat{\Delta_X} = [\widehat{\eta_{x_1}}|\widehat{\eta_{x_2}}|\cdots|\widehat{\eta_{x_{d_1}}}]$ to represent sample set X . Let ϕ be a mapping from $\mathbb{R}^n$ to a high-dimensional space $\mathbb{Z}^p (p \gg n)$ [41]. This mapping can be implicitly defined by a kernel. Specifically, the (i,j)th entry of a kernel matrix K is given by $K_{ij} = K(x_i, x_j) = \phi(x_i)^{\mathrm{T}}\phi(x_j)$. Gaussian RBF kernel [41] is a typical choice of kernels and is defined as

$$K(x_i, x_j) = \exp(-||x_i - x_j||^2/2\sigma^2). \tag{37}$$

Rewriting every basis vector in $\mathbb{Z}^p$ as an expansion in terms of mapped data, $\widehat{l_x}$ and $\widehat{l_y}$ can be written as the projections of data onto the directions $\widehat{\eta_x}$ and $\widehat{\eta_y}$, respectively, i.e., $\widehat{l_x}^{\phi} = \phi(X)\widehat{\eta_x}$ and $\widehat{l_y}^{\phi} = \phi(Y)\widehat{\eta_y}$, where $\widehat{\eta_y}$ is another projection vector. Then GSCCA in kernel space can be written as

$$\begin{aligned} &\underset{\widehat{\eta_x}\in\mathbb{R}^N,\, \widehat{\eta_y}\in\mathbb{R}^N}{Max} \ \widehat{\eta_x}^{\mathrm{T}}\phi(X)^{\mathrm{T}}\phi(X)\phi(Y)^{\mathrm{T}}\phi(Y)\widehat{\eta_y}, \\ &S.t.\ \widehat{\eta_x}^{\mathrm{T}}\phi(X)^{\mathrm{T}}\phi(X)\mathrm{H}^{(X)}\phi(X)^{\mathrm{T}}\phi(X)\widehat{\eta_x} = 1 \\ &\quad \widehat{\eta_y}^{\mathrm{T}}\phi(Y)^{\mathrm{T}}\phi(Y)\mathrm{H}^{(Y)}\phi(Y)^{\mathrm{T}}\phi(Y)\widehat{\eta_y} = 1. \end{aligned} \tag{38}$$

Let $K_{XX} = \phi(X)^{\mathrm{T}}\phi(X)$ and $K_{YY} = \phi(Y)^{\mathrm{T}}\phi(Y)$ be $N \times N$ kernel matrices over X and Y sets, respectively. By substituting the matrix inner product into (38), we have

$$\begin{aligned} &\underset{\widehat{\eta_x}\in\mathbb{R}^N,\, \widehat{\eta_y}\in\mathbb{R}^N}{Max} \ \widehat{\eta_x}^{\mathrm{T}}K_{XX}K_{YY}\widehat{\eta_y} \\ &S.t.\ \widehat{\eta_x}^{\mathrm{T}}K_{XX}\mathrm{H}^{(X)}K_{XX}\widehat{\eta_x} = \widehat{\eta_y}^{\mathrm{T}}K_{YY}\mathrm{H}^{(Y)}K_{YY}\widehat{\eta_y} = 1. \end{aligned} \tag{39}$$

Observing from (14) and (39), we see they are of similar expression. The major difference is that data and class label matrices are now represented in the form of matrix inner product. So the basis vectors $\widehat{\eta_x}$ and $\widehat{\eta_y}$ can be obtained by similar methods as solving GSCCA. The detailed computations will not be provided in this paper due to page limitation. So, the problem for computing the projection axes in $\widehat{\Delta_X}$ can be formulated as

TABLE 1
Performance Comparison of Each Method Based on 12 Standard UCI Data Sets

Dataset / Method	Breast	SPECT	Sonar	CMC	Monks1	Monks2	Monks3	Vote	Heart	Control	Hepatitis	Glass
# *Dim*	9	22	60	9	6	6	6	16	13	60	19	10
# *Num*	699	267	208	1473	556	601	554	435	270	600	155	214
#*Classes* (c)	2	2	2	3	2	2	2	2	2	6	2	6
# *Train*	15×c	30×c	30×c	50×c	50×c	50×c	50×c	40×c	20×c	30×c	45×c	7×c
PCA	58.25%	46.85%	70.18%	34.63%	49.51%	50.19%	50.24%	56.46%	54.21%	93.67%	52.66%	93.31%
	0.0036s	0.0053s	0.0088s	0.0304s	0.0080s	0.00855	0.0089s	0.0047s	0.0034s	0.0108s	0.0031s	0.0035s
LDA	59.12%	48.45%	67.79%	35.15%	49.65%	51.63%	50.73%	55.40%	53.71%	94.30%	57.56%	89.81%
	0.0048s	0.0068s	0.0081s	0.0353s	0.0093s	0.0091s	0.0108s	0.0058s	0.0039s	0.0252s	0.0039s	0.0047s
MMC	57.32%	48.86%	70.50%	35.64%	50.05%	51.17%	50.56%	56.76%	52.21%	95.72%	56.32%	93.47%
	0.0049s	0.0061s	0.0089s	0.0350s	0.0088s	0.0094s	0.0101s	0.0053s	0.0037s	0.0212s	0.0035s	0.0044s
MCCA	59.24%	48.87%	66.25%	35.35%	48.97%	50.59%	50.71%	56.38%	53.04%	92.71%	52.71%	86.52%
	0.0045s	0.0065s	0.0083s	0.0361s	0.0106s	0.0099s	0.0110s	0.0062s	0.0043s	0.0238s	0.0056s	0.0053s
SPP1	60.07%	52.06%	71.85%	36.48%	50.17%	50.69%	50.52%	57.71%	53.39%	94.24%	53.81%	94.23%
	0.1516s	0.1561s	0.1968s	0.2025	0.1768s	0.1910s	0.1983s	0.1620s	0.1467s	2.2324s	0.1297s	0.1171s
SPP2	61.14%	51.33%	70.89%	35.86%	50.34%	50.71%	50.63%	57.23%	53.12%	93.65%	53.25%	93.89%
	0.0103s	0.0673s	0.0931s	0.1047s	0.1057s	0.0921s	0.1145s	0.0809s	0.0636s	0.9586s	0.0655s	0.0672s
TR-LDA	62.57%	50.69%	69.92%	35.57%	50.23%	51.35%	50.60%	61.00%	52.61%	92.81%	54.56%	94.13%
	0.0055s	0.0072s	0.0091s	0.0383s	0.0212s	0.0108s	0.0175s	0.0087s	0.0064s	0.0309s	0.0052s	0.0055s
SLPP	60.92%	49.69%	72.03%	36.06%	49.89%	49.84%	50.93%	57.13%	52.97%	93.48%	54.56%	94.06%
	0.0082s	0.0415s	0.0401s	0.0637s	0.0514s	0.0681s	0.0776s	0.0413s	0.0330s	0.6468s	0.0423s	0.0488s
SRDA	62.23%	50.39%	73.26%	36.43%	50.14%	50.58%	50.53%	58.65%	53.12%	94.57%	55.83%	94.15%
	0.0088s	0.0407s	0.0454s	0.0624s	0.0509s	0.0622s	0.0743s	0.0402s	0.0302s	0.6540s	0.0395s	0.0445s
GSCCA	61.56%	53.40%	73.45%	36.44%	50.33%	50.84%	50.67%	61.50%	53.72%	95.54%	55.37%	94.64%
	0.0156s	0.0761s	0.0949s	0.1099s	0.1081s	0.1113s	0.1155s	0.0910s	0.0816s	1.1042s	0.0759s	0.0723s
o-GSCCA	62.94%	51.21%	74.01%	36.52%	50.81%	51.28%	50.75%	60.07%	54.36%	94.46%	56.22%	93.37%
	0.0164s	0.0823s	0.1005s	0.1102s	0.1099s	0.1161s	0.1172s	0.0946s	0.0825s	1.1316s	0.0763s	0.0734s

$$\underset{\widehat{\eta}_x \in \mathbb{R}^N}{Max} \frac{\widehat{\eta}_x^{\mathrm{T}} K_{XX} \widetilde{\Gamma^{(X)}} K_{XX} \widehat{\eta}_x}{\widehat{\eta}_x^{\mathrm{T}} K_{XX} D^{(X)} K_{XX} \widehat{\eta}_x}, \tag{40}$$

where $\widetilde{\Gamma^{(X)}} = (1-\ell_+)K_{YY}(K_{YY}\mathrm{H}^{(Y)}K_{YY} + \alpha_+ I)^{-1}K_{YY} + \ell_+ W^{(X)}$. Note that the above problem in multidimensional case can be effectively solved by using the generalized eigendecomposition or orthogonal TR optimization shown in Section 3.3. After obtaining the optimal projection matrix $\widehat{\Delta_X} = [\widehat{\eta_{x_1}}|\widehat{\eta_{x_2}} \mid \cdots |\widehat{\eta_{x_{d_1}}}]$ and the corresponding eigenvalues $\lambda_1^\phi \geq \lambda_2^\phi \geq \cdots \geq \lambda_{d_1-1}^\phi \geq \lambda_{d_1}^\phi$. feature reduction can be similarly conducted. Specifically, the embedding, $\nabla(x_r)$, of $\phi(x_r)$ from the sample set X can be represented as

$$\nabla(x_r) = \left(\sqrt{\lambda_1^\phi}\widehat{\eta_{x_1}} \middle| \sqrt{\lambda_2^\phi}\widehat{\eta_{x_2}} \middle| \cdots \middle| \sqrt{\lambda_{d_1-1}^\phi}\widehat{\eta_{x_{d_1-1}}} \middle| \sqrt{\lambda_{d_1}^\phi}\widehat{\eta_{x_{d_1}}}\right)^{\mathrm{T}} \begin{pmatrix} K(x_1, x_r) \\ K(x_2, x_r) \\ \cdots \\ K(x_N, x_r) \end{pmatrix}. \tag{41}$$

As KGSCCA represents data by inner product, it allows us to conduct feature reduction on the nonvectorial data, for example, [42]. But kernelized methods heavily depend on the selection of kernel function and its width, since different kernels deliver different properties [41]. But to date there is still no theoretical guarantee of optimal selection of the kernels, so we mainly examine the linear methods.

5 Simulation Results and Analysis

In this section, we conduct simulations on benchmark UCI and real data sets to examine GSCCA and o-GSCCA. The performance of our methods is compared with PCA, LDA, MMC, MCCA, TR-LDA, SPP, sparse locality preserving projection (SLPP) [21] and spectral regression discriminant analysis (SRDA) [44], where SPP, SLPP, and SRDA deliver sparse solutions. All algorithms are implemented in MATLAB 7.1. The codes of SLPP and SRDA with default parameter settings, available from http://www.cad.zju.edu.cn/home/dengcai/Data/SR.html, are applied here. For classification, one-nearest-neighbor (1NN) classifier with euclidean metric is used. The setting of ε is the same as of [25] and the parameter ℓ_+ is chosen by cross validation. For semi-definite matrix inverse operation involved methods, the regularization factor is set to 0.01 in all studies. For MCCA, GSCCA and o-GSCCA, the one-of- c encoding [29] is used to define the class label matrix, but it needs not to be centered according to [3], [29]. We performed all simulations on a PC with Intel(R) Core (TM) i5 CPU 650 at 3.20 GHz 3.19 GHz 4G.

In this study, 12 standard data sets from the UCI machine learning repository (available from http://www.ics. uci.edu/~mlearn/MLRepository.html) and two real databases are evaluated. For classification, training samples selected from the data set are used for training a learner and the remaining are for testing. The training set in all data sets is preliminarily processed by PCA to eliminate the null space before FE. After a 1NN classifier is trained, test samples are projected onto the reduced space. The learner is then used for evaluating the accuracy. Notice that for LDA and TR-LDA, there are at most $c-1$ nonzero

Fig. 2. Typical sample images of the ETH-80 database.

eigenvalues, and so an upper bound on the dimensionality of the reduced space is $c-1$ [12]. That is, LDA and TR-LDA can only extract $c-1$ meaningful features. We first test our methods on 12 standard UCI data sets.

5.1 Classification on UCL Data Sets

In this study, 12 UCI data sets, i.e., Breast-cancer, SPECT, Sonar, Monks1, Monks2, Monks3, Vote, Heart-statlog, Synthetic-control, Hepatitis, Glass-identification, and Contraceptive-method-choice (CMC), are evaluated. In all the simulations, fixed number of samples from each class is selected from the data set for training. For each case, the classification results are averaged over 20 random splits of training/test samples. For each set, we report the mean test accuracy and averaged running time (in seconds) of each method under different reduced dimensionalities in Table 1. We also elaborate the data set descriptions and settings in Table 1. The l^1-norm and mixed $l^{2,0}$-norm are applied to compute the sparse weights for SPP, which are, respectively, called SPP1 and SPP2. From the results, we have the following observations.

1. For classification, our GSCCA and o-GSCCA algorithms deliver comparable or even better results than other methods in most cases.
2. MCCA is always comparative with LDA due to the fact that MCCA can be equivalent to LDA [3]. LDA and TR-LDA outperforms our algorithms over Monks2 data set. MMC and LDA work well on Monks2, Synthetic-control and Hepatitis sets. SRDA is also good in most cases.
3. Compared with LDA, MMC, MCCA, SRDA, and TR-LDA, sparse SLPP, SPP1, and SPP2 can deliver promising results in most cases and obtain comparable results to our methods on the SPECT, CMC, Monks2, and Glass-identification data sets. PCA also works well in some cases.
4. Considering the running time performance, results of SPP2 and our methods are comparable to PCA, LDA, MMC, MCCA, SLPP, SRDA, and TR-LDA when the training sample size is relatively smaller. When the number of training samples increases, more running time is required in SPP2 and our algorithms, because of the involvement of sparse reconstruction. Compared with the l^1-norm based SPP1, SPP2 is more efficient.

These observations motivate us to design more efficient approaches to speed-up the sparse reconstructive process with the algorithmic effectiveness guaranteed in our future studies.

5.2 Object Recognition on ETH80 Database

This study addresses an object categorization task on the ETH80 database [37]. This database contains images of eight big categories: *apple*, *car*, *cow*, *cup*, *dog*, *horse*, *pear*, and *tomato*. Every category has 10 subcategories, each of which contains 41 images from different viewpoints. Overall, the database contains 3,280 images of 80 objects. In our studies, we resize the images to 32×32 pixels. Each pixel is considered as an input variable and so each image corresponds to a point in a 1,024D space. We show some typical images in Fig. 2.

5.2.1 Visualization of the Transforming Matrix

We first examine the visual properties of the projection matrices of our methods and compare them with PCA, LDA, MMC, MCCA, TR-LDA, and SPP. In the following simulations, group sparsity is always considered in SPP. In this simulation, the cup category with 10 objects is tested and each object corresponds to a single class. Then a 10-class case is created. For each method, we randomly select eight images from each object to learn the optimal

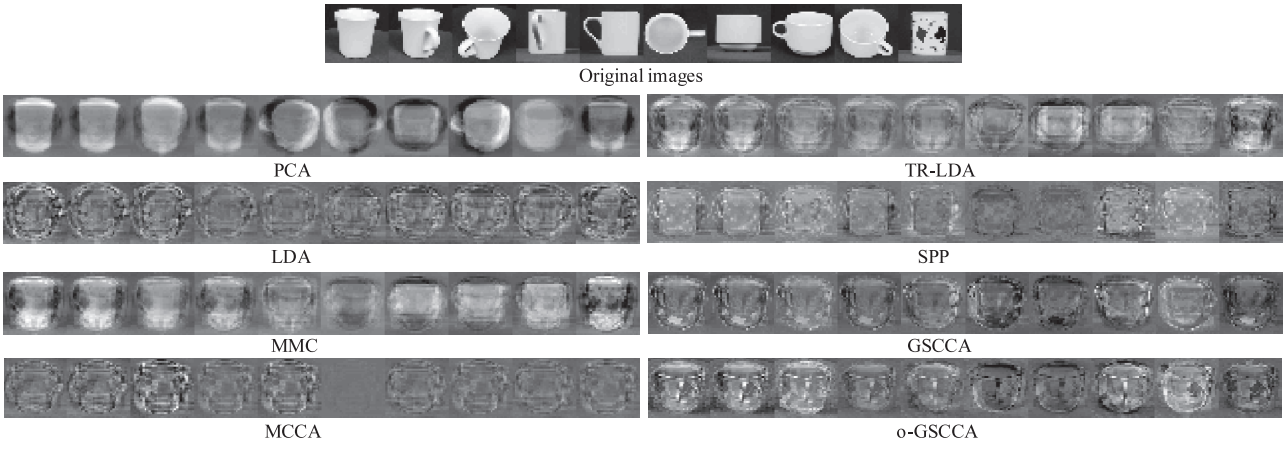


Fig. 3. Visualization of the projection matrices of PCA, LDA, MMC, MCCA, TR-LDA, SPP and our algorithms on the cup object category.

TABLE 2
Performance Comparisons of the Algorithms on the ETH80 Object Database

Result / Method	Simulation Setting: ETH80 (D=1024, 15 train)			ETH80 (D=1024, 25 train)			ETH80 (D=1024, 35 train)		
	Mean	Std	Best	Mean	Std	Best	Mean	Std	Best
PCA	66.75%	0.0304	68.98%	70.26%	0.0413	72.17%	72.59%	0.0463	74.69%
LDA	52.09%	0.0246	53.10%	56.77%	0.0164	57.61%	59.70%	0.0493	62.84%
MMC	65.44%	0.0241	66.10%	68.62%	0.0269	69.94%	71.49%	0.0312	72.80%
MCCA	50.86%	0.0159	52.79%	54.91%	0.0087	56.17%	60.50%	0.0273	62.22%
SPP	64.46%	0.0398	66.91%	69.11%	0.0557	72.33%	72.79%	0.0572	75.03%
TR-LDA	63.09%	0.0434	65.22%	65.24%	0.0381	68.03%	67.18%	0.0414	69.67%
GSCCA	67.78%	0.0355	69.13%	72.27%	0.0597	75.17%	74.70%	0.0481	77.13%
o-GSCCA	68.33%	0.0268	70.77%	71.11%	0.0540	73.53%	74.72%	0.0415	77.24%
Result / Method	Simulation Setting: ETH80 (D=1024, 45 train)			ETH80 (D=1024, 55 train)			ETH80 (D=1024, 65 train)		
	Mean	Std	Best	Mean	Std	Best	Mean	Std	Best
PCA	77.60%	0.0576	80.04%	79.77%	0.0574	82.03%	82.58%	0.0604	85.12%
LDA	63.54%	0.0604	67.57%	66.65%	0.0647	70.05%	70.51%	0.0665	74.52%
MMC	73.41%	0.0349	74.90%	76.59%	0.0318	77.90%	79.19%	0.0361	80.60%
MCCA	64.17%	0.0320	66.76%	67.74%	0.0323	70.40%	72.24%	0.0320	74.31%
SPP	77.73%	0.0571	81.44%	81.01%	0.0595	83.47%	83.73%	0.0575	86.52%
TR-LDA	68.70%	0.0418	72.21%	71.57%	0.0521	74.32%	75.41%	0.0515	78.03%
GSCCA	79.27%	0.0691	81.93%	82.13%	0.0612	85.03%	84.10%	0.0614	87.03%
o-GSCCA	81.25%	0.0506	83.30%	83.57%	0.0539	85.66%	85.48%	0.0501	87.91%

subspaces. We illustrate the first 10 eigenvectors (or, eigen-pictures) of the transforming matrix obtained by each method. The eigen-pictures are then reshaped into a matrix according to the original object image size, i.e., 32×32. The eigen-pictures gained in these methods are exhibited in Fig. 3. For SPP and our methods, we also show the first 10 eigenvectors. The objects can then be projected onto the sparse reconstructive discriminant subspaces spanned by the eigenvectors. Observing from Fig. 3, we see that the eigen-pictures of PCA look like cups, reflecting the principal feature of images. It is also observed that the eigen-pictures of LDA, MMC, MCCA, TR-LDA, SPP and our methods are more noisy compared with PCA, demonstrating that they are capable of capturing more discriminant information about image details.

5.2.2 Object Recognition

This section focuses on representing and recognizing the object images of ETH80 database. In this study, each of the eight big categories is considered as a single class. So an eight-class classification problem is created. In our simulation, 200 images from each of the eight big categories are selected for the experiments. Six simulation settings over different training sample sizes are evaluated.

The validity of the algorithms is evaluated by varying the training sample sizes and numbers of reduced dimensions for image recognition. The results are illustrated in Fig. 3. We compute the accuracies by averaging the test results over 20 random splits of training/test samples. The performance of our methods is also compared with PCA, LDA, MMC, MCCA and TR-LDA, and SPP. Observing from Fig. 3, we conclude that: 1) The performance of all methods increase with the increasing numbers of training samples and the reduced dimensions. 2) The unsupervised PCA performs better over this data set and obtains the recognition accuracies that are higher than those of the supervised LDA, MMC, MCCA and TR-LDA. Also, PCA obtains comparative results to SPP in some cases, which may be attributed to the orthogonal projection vectors of PCA. Due to the equivalence relationship between LDA and MCCA, their accuracies are close and both are worse than other methods. MMC can deliver comparable or even higher accuracy to TR-LDA in each case. This is mainly because MMC can be treated as a special case of the TR framework [26]. 3) Based on characterizing data with the sparsest representation, SPP outperforms LDA, MMC, MCCA, and TR-LDA in each case. By considering the class information of data into the SR, the performance is further improved by our GSCCA and o-GSCCA, compared with other algorithms.

We report the means of recognition accuracies and the standard deviations (Std) in Table 2. The best records are also described here. Seeing from Table 2, we find:

1. The performance superiorities, including the mean and best results, of the algorithms keep consistent with the results in Fig. 4.
2. The standard deviations actually reflect the smooth degree of the curve's trends. We can observe from Table 2 that the standard deviation of each method is comparable in each setting.
3. Our methods are capable of delivering better results than other methods in each tested case.
4. The results also demonstrate that the delivered mean and best accuracies of these algorithms also increase as the training sample sizes increase.

5.3 Application to Natural Image Segmentation

We also prepare an interactive image segmentation task using the benchmark Berkeley segmentation database [38]. This task focuses on extracting the foreground regions from the natural images. Many efforts have been made, for instance [39], [40], but image segmentation is still a challenging problem. When dealing with interactive image

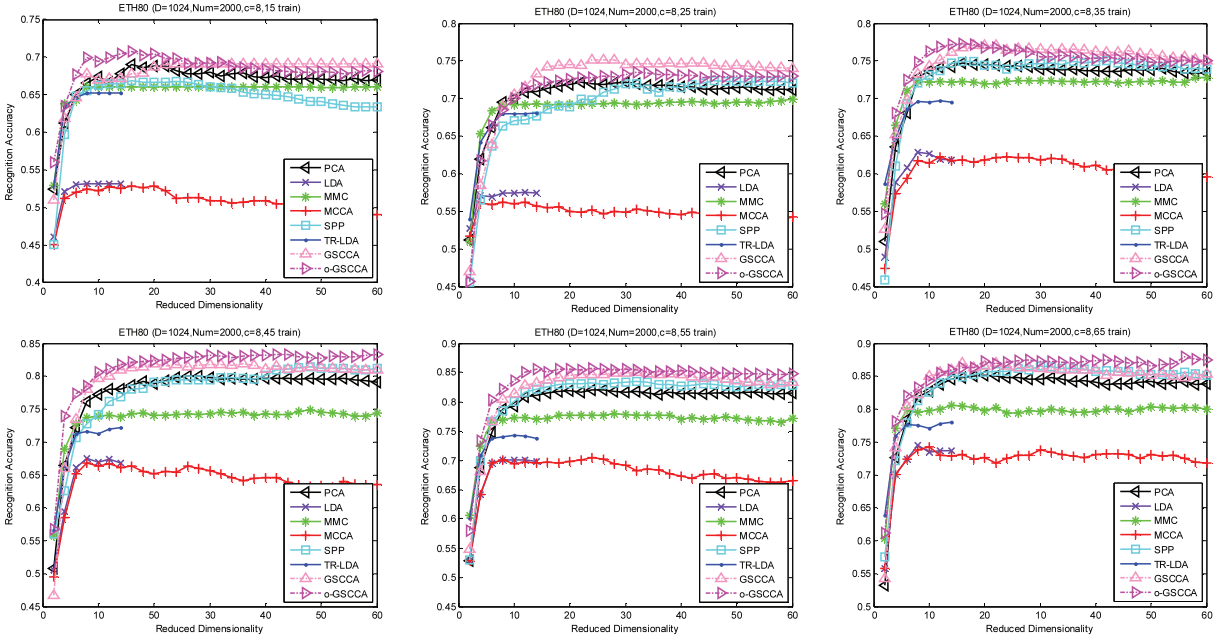


Fig. 4. Recognition accuracy versus number of reduced dimensions on the ETH80 object recognition database.

segmentation, the most important issue is to collect the user specified pixels about foreground and background. In this simulation, eight natural images from the Berkeley database are tested. Each extracted pixel from the images is represented by a 5D vector φ , i.e., $\varphi = [R, G, B, \chi, \eta]^{\mathrm{T}}$, where (R, G, B) denotes the normalized color of the pixel and (χ, η) denotes the spatial coordinate with image width and height. After the pixels are extracted, the dimensionalities are then reduced to 4 by PCA LDA, MMC, MCCA, TR-LDA, SPP, GSCCA, and o-GSCCA. The 1NN classifier is finally applied to determine the class labels of the pixels. Based on the obtained class labels of pixels, the object image regions are classified into foregrounds and backgrounds.

Fig. 5 exhibits the results. Row (a) shows the original images. Row (b) shows the source images with the user specified pixels, where blue and red colors indicate different segments. Rows (c), (d), (e), (f), (g), (h), (i), and (j) illustrate the segmentation result of each method, respectively. It can be observed that: 1) Our GSCCA and o-GSCCA deliver visually comparable and even better performance than other methods in most cases, especially on determining the image details and boundaries. We take the owl image as an example, which needs to segment the owl and stick region from the background. This segmentation task is relatively difficult, as colors of foreground and background of the images are similar. We show the segmented regions with the original image resolution for comparison. We see clearly that more details can be captured by our methods. 2) From the results of the investigated cases, our GSCCA and o-GSCCA methods are able to deliver better segmentation results than LDA, MMC, MCCA, TR-LDA, and SPP, especially on the first, second, third, fifth, sixth, seventh, and eighth images. It can be clearly seen that more pixels from the foreground and background of the images are incorrectly classified by PCA, LDA, MMC, MCCA, TR-LDA, and SPP in most cases. In contrast, our GSCCA and o-GSCCA work better in classifying the pixels and are able to deliver satisfactory results. 3) Compared with LDA, MMC, MCCA, TR-LDA, and SPP, PCA delivers better segmentation results on the second, fifth, sixth, and eighth images. The boundaries of foreground and background are correctly segmented by PCA. LDA, and MCCA clearly detect the boundaries on the fourth and seventh images. MMC delivers satisfactory results on the fifth and eighth images. In contrast, SPP is capable of representing the fifth, sixth, and eighth images appropriately and effectively capture the details in the images.

6 Concluding Remarks

This paper discusses a group sparse representation-based discriminant feature extraction problem. An effective feature extraction technique, termed group sparse canonical correlation analysis (GSCCA), is developed. GSCCA is originally proposed for handling binary-class case by computing two sets of projection axes for representing two sets of variables. In extracting the informative features, GSCCA relies on preserving the desirable group sparse relationships of highly correlated samples within a group while at the same time maximizing the between-set covariance. Also, the locality and sparsity of data can be automatically determined. By defining one of the two sets to be a class label matrix, GSCCA is naturally extended for multiclass feature extraction and classification. We also establish the equivalence between GSCCA and the least-squares problem. Comparison and analyses between our method and other related techniques demonstrate that our algorithm is more general. To effectively calculate the projection axes, two solution schemes based on eigen-decomposition and trace ratio optimization are proposed. We also present kernelized GSCCA for mining the nonlinear data structures by the standard kernel trick.

This work mainly evaluates the proposed linear methods and the validity is examined by visualization, classification, object recognition, and image segmentation. By visualizing UCI data set, BGSCCA delivers more

Fig. 5. Image segmentation results, where (a) original images, (b) the partially labeled images with user specified pixels, (c) PCA+1NN, (d) LDA+1NN, (e) MMC+1NN, (f) MCCA+1NN, (g) TR-LDA+1NN, (h) SPP+ 1NN, (i) GSCCA+1NN, and (j) o-GSCCA+1NN.

separated embeddings of different clusters. For classification and object recognition, most of our tested cases indicate that the overall performance of our GSCCA is comparative or even outperforms other related techniques. From image segmentation, our GSCCA can represent the pixels of images appropriately through detecting the foreground and background regions effectively. In our future work, investigating the approach of accelerating the sparse representation process is required. Also, we must admit that, in machine learning and pattern recognition areas, determination of optimal reduced dimensions still remains an open problem that needs further exploration.

ACKNOWLEDGMENTS

This work was partially supported by a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (CityU8/CRF/09).

REFERENCES

[1] D.R. Hardoon, S. Szedmak, and J. Shawe-Taylor, "Canonical Correlation Analysis: An Overview with Applications to Learning Methods," *Neural Computation*, vol. 16, no. 12, pp. 2639-2664, 2004.

[2] H. Hotelling, "Relations between Two Sets of Variates," *Biometrika*, vol. 28, pp. 321-377, 1936.

[3] T. Sun and S.C. Chen, "Class Label versus Sample Label-Based CCA," *Applied Math. and Computation*, vol. 185, no. 1, pp. 272-283, 2007.

[4] L. Sun, S.W. Ji, S.P. Yu, and J.P. Ye, "On the Equivalence between Canonical Correlation Analysis and Orthonormalized Partial Least Squares," *Proc. 21st Int'l Joint Conf. Artificial Intelligence (IJCAI)*, pp.1230-1235, 2009.

[5] A. Lykou and J. Whittaker, "Sparse CCA Using a Lasso with Positivity Constraints," *Computational Statistics and Data Analysis*, vol. 54, pp. 3144-3157, 2010.

[6] M. Loog, B. Ginneken, and R.P.W. Duin, "Dimensionality Reduction by Canonical Contextual Correlation Projections," *Proc. European Conf. Computer Vision (ECCV)*, pp. 562-573, 2004.

[7] D.R. Hardoon and J. Shawe-Taylor, "Sparse Canonical Correlation Analysis," *Machine Learning*, vol. 83, no. 3, pp. 331-353, 2011.

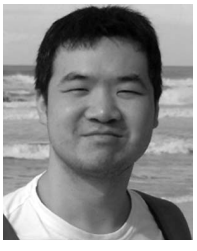
[8] P. Elena, T. David, and B. Joseph, "Sparse Canonical Correlation Analysis with Application to Genomic Data Integration," *Statistical Applications in Genetics and Molecular Biology*, vol. 8, no. 1, pp. 1-34, 2009.

[9] S. Waaijenborg, P.C.V. de Witt Hamer, and A.H. Zwinderman, "Quantifying the Association between Gene Expressions and DNA-Markers by Penalized Canonical Correlation Analysis," *Statistical Applications in Genetics and Molecular Biology*, vol. 7, no. 1, p. 3, 2008.

[10] M. Naylor, X. Lin, S. Weiss, B. Raby, and C. Lange, "Using Canonical Correlation Analysis to Discover Genetic Regulatory Variants," *PLoS One*, vol. 5, no. 5, p. e10395, 2010.

[11] S. Waaijenborg and A.H. Zwinderman, "Sparse Canonical Correlation Analysis for Identifying, Connecting and Completing Gene-Expression Networks," *BMC Bioinformatics,* vol. 10, article 315, 2009.
[12] A. Martinez and A. Kak, "PCA versus LDA," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 23, no. 2, pp. 228-233, Feb. 2001.
[13] L. Sun, S.W. Ji, and J.P. Ye, "Canonical Correlation Analysis for Multilabel Classification: A Least-Squares Formulation, Extensions, and Analysis," *IEEE Trans. Patten Analysis and Machine Intelligence,* vol. 33, no. 1, pp. 194-200, Jan. 2011.
[14] M. Belkin and P. Niyogi, "Laplacian Eigenmaps for Dimensionality Reduction and Data Representation," *Neural Computation,* vol. 15, no. 6, pp. 1373-1396, 2003.
[15] S. Roweis and L. Saul, "Nonlinear Dimensionality Reduction by Locally Linear Embedding," *Science,* vol. 290, no. 5500, pp. 2323-2326, 2000.
[16] J.B. Tenenbaum, V. Silva, and J.C. Langford, "A Global Geometric Framework for Nonlinear Dimensionality Reduction," *Science,* vol. 290, no. 5500, pp. 2319-2323, 2000.
[17] J. Wright, A. Yang, S. Sastry, and Y. Ma, "Robust Face Recognition via Sparse Representation," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 31, no. 2, pp. 210-227, Feb. 2009.
[18] R. He, B.G. Hu, W.S. Zheng, and Y.Q. Guo, "Two-Stage Sparse Representation for Robust Recognition on Large-Scale Database," *Proc. AAAI Conf. Artificial Intelligence,* 2010.
[19] A. Wiesel, M. Kliger, and A.O. Hero III, "A Greedy Approach to Sparse Canonical Correlation Analysis," *Annals of Statistics,* available in arXiv:0801.2748v1, 2008.
[20] B. Cheng, J.C. Yang, S.C. Yan, Y. Fu, and T.S. Huang, "Learning with l^1 -Graph for Image Analysis," *IEEE Trans. Image Processing* vol. 19, no. 4, pp. 858-866, Apr. 2010.
[21] D. Cai, X. He, and J. Han, "Spectral Regression: A Unified Approach for Sparse Subspace Learning," *Proc. IEEE Seventh Int'l Conf. Data Mining (ICDM),* 2007.
[22] S. Zhu, D. Wang, K. Yu, T. Li, and Y. Gong, "Feature Selection for Gene Expression Using Model-Based Entropy," *IEEE/ACM Trans. Computational Biology and Bioinformatics,* vol. 7, no. 1, pp. 25-36, Jan.-Mar. 2010.
[23] A. Majumdar and R.K. Ward, "Robust Classifiers for Data Reduced via Random Projections," *IEEE Trans. Systems, Man and Cybernetics, Part B-Cybernetics,* vol. 40, no. 5, pp. 1359-1371, Oct. 2010.
[24] H. Zou and T. Hastie, "Regularization and Variable Selection via the Elastic Net," *J. Royal Statistical Soc. B,* vol. 67, no. 2, pp. 301-320, 2005.
[25] L.S. Qiao, S.C. Chen, and X.Y. Tan, "Sparsity Preserving Projections with Applications to Face Recognition," *Pattern Recognition,* vol. 43, no. 1, pp. 31-341, 2010.
[26] Y. Jia, F. Nie, and C. Zhang, "Trace Ratio Problem Revisited," *IEEE Trans. Neural Network,* vol. 20, no. 4, pp. 729-735, Apr. 2009.
[27] H. Wang, S. Yan, D. Xu, X. Tang, and T. Huang, "Trace Ratio vs. Ratio Trace for Dimensionality Reduction," *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR),* 2007.
[28] L. Sun, S.W. Ji, and J.P. Ye, "A Least Squares Formulation for Canonical Correlation Analysis," *Proc. 25th Int'l Conf. Machine Learning (ICML),* pp. 1024-1031, 2008.
[29] M. Barker and W. Rayens, "Partial Least Squares (PLS) for Discrimination," *J. Chemometrics,* vol. 17, pp. 166-173, 2003.
[30] T.V. Gestel, J.A.K. Suykens, J. De Brabanter, B. De Moor, and J. Vandewalle, "Kernel Canonical Correlation Analysis and Least Squares Support Vector Machines," *Proc. Int'l Conf. Artificial Neural Networks (ICANN),* 2001.
[31] A.C. Yau, X.C. Tai, and M.K. Ng, "Compression and Denoising Using l_0 -Norm," *Computational Optimization and Applications,* vol. 50, no. 2, pp. 425-444, 2011.
[32] A. Majumdar and R.K. Ward, "Classification via Group Sparsity Promoting Regularization," *Proc. IEEE Int'l Conf. Acoustics, Speech and Signal Processing (ICASSP),* 2009.
[33] 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. on Pattern Analysis and Machine Intelligence,* vol. 29, no. 1, pp. 40-51, Jan. 2007.
[34] Y. Guo, S. Li, J. Yang, T. Shu, and L. Wu, "A Generalized Foley-Sammon Transform Based on Generalized Fisher Discriminant Criterion and Its Application to Face Recognition," *Pattern Recognition Letters,* vol. 24, nos. 1-3, pp. 147-158, 2003.
[35] K. Fukunaga, *Introduction to Statistical Pattern Recognition,* second ed. Academic Press, 1991.
[36] H. Li, T. Jiang, and K. Zhang, "Efficient and Robust Feature Extraction by Maximum Margin Criterion," *IEEE Trans. Neural Networks,* vol. 17, no. 1, pp. 157-165, Jan. 2006.
[37] B. Leibe and B. Schiele, "Analyzing Appearance and Contour Based Methods for Object Categorization," *Proc. IEEE CS Conf. Computer Vision and Pattern Recognition (CVPR),* 2003.
[38] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A Database of Human Segmented Natural Images and Its Application to Evaluating Segmentation Algorithms and Measuring Ecological Statistics," *Proc. IEEE Eighth Int'l Conf. Computer Vision (ICCV),* pp. 416-423, 2001.
[39] J.F. Ning, L. Zhang, D. Zhang, and C.K. Wu, "Interactive Image Segmentation by Maximal Similarity Based Region Merging," *Pattern Recognition,* vol. 43, no. 2, pp. 445-456, 2010.
[40] J.D. Wang, F. Wang, C.S. Zhang, H.C. Shen, and L. Quan, "Linear Neighborhood Propagation and Its Applications," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 31, no. 3, pp. 1600-1615, Sept. 2009.
[41] B. Schölkopf and A. Smola, *Learning with Kernels.* MIT Press, 2002.
[42] Y. Guo, J. Gao, and P. Kwan, "Kernel Laplacian Eigenmaps for Visualization of Non-Vectorial Data," *Proc. 19th Australian Joint Conf. Artificial Intelligence: Advances in Artificial Intelligence,* pp. 1179-1183, 2006.
[43] L. Sun, B. Ceran, and J.P. Ye, "A Scalable Two-Stage Approach for a Class of Dimensionality Reduction Techniques," *Proc. 16th ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (KDD),* 2010.
[44] D. Cai, X.F. He, and J.W. Han, "SRDA: An Efficient Algorithm for Large-Scale Discriminant Analysis," *IEEE Trans. Knowledge and Data Eng.,* vol. 20, no. 1, pp. 1-12, Jan. 2008.

Zhao Zhang (S'11) is currently working toward the PhD degree in the Department of Electronic Engineering, City University of Hong Kong. He was a visiting research engineer in the Department of Electrical and Computer Engineering, National University of Singapore, in 2012. His current research interests include machine learning, pattern recognition and applications. He is a student member of the IEEE.



Mingbo Zhao (S'11) is currently working toward the PhD degree in the Department of Electronic Engineering, City University of Hong Kong. His current interests include machine learning, data mining, and pattern recognition. He is a student member of the IEEE.

Tommy W.S. Chow (M'93-SM'03) received the BSc (First Hons) and PhD degrees from the University of Sunderland, Sunderland, United Kingdom. He is currently a professor in the Electronic Engineering Department, City University of Hong Kong. He has authored or coauthored more than 200 technical papers. He is an associate editor of *Pattern Analysis and Applications* and the *International Journal of Information Technology*. He is a senior member of the IEEE.