Proceedings of the International MultiConference of Engineers and Computer Scientists 2013 Vol I,
IMECS 2013, March 13 - 15, 2013, Hong Kong

Sparse Kernel Canonical Correlation Analysis

Delin Chu, Li-Zhi Liao, Michael K. Ng and Xiaowei Zhang

***Abstract*—Canonical correlation analysis (CCA) is a multivariate statistical technique for finding the linear relationship between two sets of variables. The kernel generalization of CCA named kernel CCA has been proposed to find nonlinear relations between data sets. Despite the wide usage of CCA and kernel CCA, they have one common limitation that is the lack of sparsity in their solution. In this paper, we consider sparse kernel CCA and propose a novel sparse kernel CCA algorithm (SKCCA). Our algorithm is based on a relationship between kernel CCA and least squares. Sparsity of the dual transformations is introduced by penalizing the ℓ_1-norm of dual vectors. Experiments demonstrate that our algorithm not only performs well in computing sparse dual transformations but also can alleviate the over-fitting problem of kernel CCA.**

***Index Terms*—canonical correlation analysis, kernel, sparsity**

I. Introduction

THE description of relationship between two sets of variables has long been an interesting topic to many researchers. Canonical correlation analysis (CCA) [10] is a multivariate statistical technique for finding the linear relationship between two sets of variables. It seeks a linear transformation for each of the two sets of variables in a way that the projected variables in the transformed space are maximally correlated. In recent years, CCA has been successfully applied in various areas, including genomic data analysis [19], [20] and bilingual analysis [18], where researchers can measure multiple sets of variables on a single subject. For instance, DNA copy number variations, gene expression, and single nucleotide polymorphism (SNP) data might all be available on a common set of patient samples.

Since CCA only consider linear transformation of the original variables, it fails to capture nonlinear relations. However, in a wide range of practical problems linear relations may not be adequate for studying relation among variables. Detecting nonlinear relations among data is important and useful in modern data analysis, especially when dealing with data that are not in the form of vectors, such as text documents, images, microarray data and so on. A natural extension, therefore, is to explore and exploit nonlinear relations among data. Among nonlinear extensions of CCA, one most frequently used approach is the kernel generalization of CCA, named kernel canonical correlation analysis (kernel CCA) [1], [3]. Kernel CCA have been successfully applied in many fields, including content−based image retrieval [9], bioinformatics [21] and independent component analysis [3].

Despite the wide usage of CCA and kernel CCA, they have one common limitation that is the lack of sparsity in their solution. For CCA, the lack of sparsity makes the interpretation of extracted features difficult, while for kernel CCA it can lead to excessive computational time to compute projections of new data since kernel functions must be evaluated at all training data. To handle the limitation of CCA, researchers suggested to incorporate sparsity into weight vectors and many attempts have been made to study sparse CCA [6], [8], [19], [20]. Similarly, we shall find sparse solutions for kernel CCA so that projections of new data can be computed by evaluating the kernel function at a subset of the training data. Another motivation for studying sparse kernel CCA is the over-fitting problem of kernel CCA as pointed out in [3], [9]. Although there are many sparse kernel approaches [5], seldom can be found in the area of sparse kernel CCA [4], [16].

In this paper we consider a new sparse kernel CCA approach. A relationship between CCA and least squares is established so that CCA solutions can be obtained by solving a least squares problem. Since the optimization criteria of CCA and kernel CCA are of the same form, this relationship can be extended to kernel CCA. Based on the relationship, we attempt to introduce sparsity to kernel CCA by penalizing ℓ_1-norm of the solutions, which eventually leads to a ℓ_1-norm regularized least squares problem having the form of the following basis pursuit denoising (BPDN) problem

$$\min_{x\in\mathbf{R}^d} \frac{1}{2}\|Ax-b\|_2^2+\lambda\|x\|_1, \tag{1.1}$$

where $\lambda > 0$ is a regularizer controlling the sparsity of x. We adopt a fixed-point continuation (FPC) method [7] to solve the BPDN problem above, which results in a new sparse kernel CCA algorithm named SKCCA.

The remainder of the paper is organized as follows. In Section II, we present background results of both CCA and kernel CCA. In Section III, we establish a relationship between CCA and least squares problems. In Section IV, we extend the relationship to kernel CCA and incorporate sparsity into kernel CCA by penalizing the least squares with ℓ_1-penalty. Solving the penalized least squares problems by FPC leads to a new sparse kernel CCA algorithm. Numerical results of applying the newly proposed algorithm to content-based image retrieval are presented in Section V. Finally, we draw some conclusions in Section VI.

II. Background

Let $\{x_i\}_{i=1}^n \in \mathbf{R}^{d_1}$ and $\{y_i\}_{i=1}^n \in \mathbf{R}^{d_2}$ be n samples for variables $x \in \mathbf{R}^{d_1}$ and $y \in \mathbf{R}^{d_2}$, respectively. Denote

$$X = [x_1 \ \cdots \ x_n] \in \mathbf{R}^{d_1\times n}, \quad Y = [y_1 \ \cdots \ y_n] \in \mathbf{R}^{d_2\times n},$$

Manuscript received 11 Oct. 2012; accepted 8 Nov. 2012.

Delin Chu and Xiaowei Zhang are with Department of Mathematics, National University of Singapore, Block S17, 10 Lower Kent Ridge Road Singapore 119076. Email: matchudl@nus.edu.sg/g0800887@nus.edu.sg. These two authors are supported in part by NUS Research Grant R-146-000-140-112.

Li-Zhi Liao is with Department of Mathematics, Hong Kong Baptist University, Kowloon Tong, Hong Kong. Email: liliao@hkbu.edu.hk. This author is supported in part by GRF grants HKBU201409 and HKBU201611 from the Research Grant Council of Hong Kong.

Michael K. Ng is with Centre for Mathematical Imaging and Vision, and Department of Mathematics, Hong Kong Baptist University, Kowloon Tong, Hong Kong. E-mail: mng@math.hkbu.edu.hk. This author is supported in part by grants from Hong Kong Baptist University (FRG), and the Research Grant Council of Hong Kong.

and assume both $\{x_i\}_{i=1}^n$ and $\{y_i\}_{i=1}^n$ have zero mean, i.e., $\sum_{i=1}^n x_i = 0$ and $\sum_{i=1}^n y_i = 0$. Then CCA solves the following optimization problem

$$\begin{aligned} \max_{w_x, w_y} \quad & w_x^T X Y^T w_y \\ s.t. \quad & w_x^T X X^T w_x = 1, \\ & w_y^T Y Y^T w_y = 1, \end{aligned} \tag{2.1}$$

to get the first pair of *weight vectors* w_x and w_y, which are further utilized to obtain the first pair of *canonical variates* $w_x^T X$ and w_y^TY, respectively. However, only one pair of weight vectors is not enough for most practical problems. To obtain multiple projections of CCA, we recursively solve the above optimization problem with additional constraint that the current canonical variates must be orthogonal to all previous ones. Specifically, denoting $W_x = [w_x^1 \ \cdots \ w_x^l]$ and $W_y = [w_y^1 \ \cdots \ w_y^l]$, we use the trace formula

$$\begin{aligned} \max_{W_x, W_y} \quad & \text{Trace}(W_x^T X Y^T W_y) \\ s.t. \quad & W_x^T X X^T W_x = I, \ W_x \in \mathbf{R}^{d_1 \times l}, \\ & W_y^T Y Y^T W_y = I, \ W_y \in \mathbf{R}^{d_2 \times l}. \end{aligned} \tag{2.2}$$

as the criterion of CCA to compute multiple projections.

In kernel methods, we first implicitly represent data as elements in reproducing kernel Hilbert spaces associated with positive definite kernels, then apply linear algorithms on the data and substitute the linear inner product by kernel functions, which results in nonlinear variants. The main idea of kernel CCA is that we first virtually map data X into a high dimensional *feature space* $\mathcal{H}_x$ via a mapping ϕ_x such that data in the feature space become

$$\Phi_x = \begin{bmatrix} \phi_x(x_1) & \cdots & \phi_x(x_n) \end{bmatrix} \in \mathbf{R}^{\mathcal{N}_x \times n},$$

where $\mathcal{N}_x$ is the dimension of feature space $\mathcal{H}_x$ that can be very high or even infinite. The mapping ϕ_x from input data to the feature space $\mathcal{H}_x$ is performed implicitly by considering a *positive definite kernel function* κ_x satisfying

$$\kappa_x(x_1, x_2) = \langle \phi_x(x_1), \phi_x(x_2) \rangle, \tag{2.3}$$

where $\langle \cdot, \cdot \rangle$ is an inner product in $\mathcal{H}_x$, rather than by giving the coordinates of $\phi_x(x)$ explicitly. The feature space $\mathcal{H}_x$ is known as the *Reproducing Kernel Hilbert Space (RKHS)* [2] associated with kernel function κ_x. In the same way, we can map Y into a feature space $\mathcal{H}_y$ associated with kernel κ_y through mapping ϕ_y such that

$$\Phi_y = \begin{bmatrix} \phi_y(y_1) & \cdots & \phi_y(y_n) \end{bmatrix} \in \mathbf{R}^{\mathcal{N}_y \times n}.$$

After mapping X to Φ_x and Y to Φ_y, we then apply ordinary linear CCA to data pair (Φ_x, Φ_y).

Let

$$K_x = \langle \Phi_x, \Phi_x \rangle = [\kappa_x(x_i, x_j)]_{i,j=1}^n \in \mathbf{R}^{n \times n}, \tag{2.4}$$

$$K_y = \langle \Phi_y, \Phi_y \rangle = [\kappa_y(y_i, y_j)]_{i,j=1}^n \in \mathbf{R}^{n \times n} \tag{2.5}$$

be matrices consisting of inner products of data sets Φ_x and Φ_y, respectively. K_x and K_y are called *kernel matrices* or *Gram matrices*. Then kernel CCA seeks linear transformation in the *RKHS* by expressing the weight vectors as linear combinations of the training data, that is

$$w_x = \Phi_x \boldsymbol{\alpha} = \sum_{i=1}^n \alpha_i \phi_x(x_i), \quad w_y = \Phi_y \boldsymbol{\beta} = \sum_{i=1}^n \beta_i \phi_y(y_i),$$

where $\boldsymbol{\alpha}$, $\boldsymbol{\beta} \in \mathbf{R}^n$ are called *dual vectors*. The first pair of dual vectors can be determined by solving the following optimization problem

$$\begin{aligned} \max_{\boldsymbol{\alpha}, \boldsymbol{\beta}} \quad & \boldsymbol{\alpha}^T K_x K_y \boldsymbol{\beta} \\ s.t. \quad & \boldsymbol{\alpha}^T K_x^2 \boldsymbol{\alpha} = 1, \\ & \boldsymbol{\beta}^T K_y^2 \boldsymbol{\beta} = 1. \end{aligned} \tag{2.6}$$

To compute multiple pairs of dual vectors, we consider

$$\begin{aligned} \max_{\mathcal{W}_x, \mathcal{W}_y} \quad & \text{Trace}(\mathcal{W}_x^T K_x K_y \mathcal{W}_y) \\ s.t. \quad & \mathcal{W}_x^T K_x^2 \mathcal{W}_x = I, \ \mathcal{W}_x \in \mathbf{R}^{n \times l}, \\ & \mathcal{W}_y^T K_y^2 \mathcal{W}_y = I, \ \mathcal{W}_y \in \mathbf{R}^{n \times l}, \end{aligned} \tag{2.7}$$

where $\mathcal{W}_x = [\boldsymbol{\alpha}^1 \ \cdots \ \boldsymbol{\alpha}^l]$ and $\mathcal{W}_y = [\boldsymbol{\beta}^1 \ \cdots \ \boldsymbol{\beta}^l]$ consist of dual vectors for X and Y, respectively.

In the process of deriving (2.7), we assumed data Φ_x and Φ_y have been centered (that is, the column mean of both Φ_x and Φ_y are zero) as X and Y, otherwise, we need to perform data centering before applying kernel CCA. Unlike data centering of X and Y, we can not perform data centering directly on Φ_x and Φ_y since we do not know their explicit coordinates. However, as shown in [12], [13], data centering in *RKHS* can be accomplished via some operations on kernel matrices. To center Φ_x, a natural idea should be computing $\Phi_{x,c} = \Phi_x(I - \frac{e_n e_n^T}{n})$, where e_n denotes column vector in $\mathbf{R}^n$ with all entries being 1. However, since kernel CCA makes use of the data X through kernel matrix K_x, the centering process can be performed on K_x as

$$\begin{aligned} K_{x,c} = \langle \Phi_{x,c}, \Phi_{x,c} \rangle &= (I - \frac{e_n e_n^T}{n}) \langle \Phi_x, \Phi_x \rangle (I - \frac{e_n e_n^T}{n}) \\ &= (I - \frac{e_n e_n^T}{n}) K_x (I - \frac{e_n e_n^T}{n}). \end{aligned} \tag{2.8}$$

Similarly, we can center testing data $\Phi_{x,t}$ as

$$\begin{aligned} K_{x,t,c} &= \langle \Phi_{x,c}, \Phi_{x,t} - \Phi_x \frac{e_n e_N^T}{n} \rangle \\ &= (I - \frac{e_n e_n^T}{n}) K_{x,t} - (I - \frac{e_n e_n^T}{n}) K_x \frac{e_n e_N^T}{n}, \end{aligned} \tag{2.9}$$

where N is the number of testing data and $K_{x,t}$ denotes the kernel matrix between training and testing data. More details about data centering in *RKHS* can be found in [12], [13]. In the sequel of this paper, we assume the kernel matrices have been centered.

III. CCA and least squares

It is well known that CCA is closely related to linear regression problem, and some relation between CCA and linear regression has been established under the condition that rank$(X) = n - 1$ and rank$(Y) = d_2$ in [14], [15]. In this section, we establish a relation between CCA and linear regression without any additional constraint on X and Y. Before that, we consider the characterization of solutions of (2.2).

Define $r = \text{rank}(X)$, $s = \text{rank}(Y)$, $m = \text{rank}(XY^T)$ and $t = \min\{r, s\}$. Let the (reduced) SVD factorizations of X and Y be, respectively,

$$X = U \begin{bmatrix} \Sigma_1 \\ 0 \end{bmatrix} Q_1^T = \begin{bmatrix} U_1 & U_2 \end{bmatrix} \begin{bmatrix} \Sigma_1 \\ 0 \end{bmatrix} Q_1^T = U_1 \Sigma_1 Q_1^T, \tag{3.1}$$

and

$$Y = V \begin{bmatrix} \Sigma_2 \\ 0 \end{bmatrix} Q_2^T = \begin{bmatrix} V_1 & V_2 \end{bmatrix} \begin{bmatrix} \Sigma_2 \\ 0 \end{bmatrix} Q_2^T = V_1 \Sigma_2 Q_2^T, \quad (3.2)$$

where $U \in \mathbf{R}^{d_1 \times d_1}$, $U_1 \in \mathbf{R}^{d_1 \times r}$, $U_2 \in \mathbf{R}^{d_1 \times (d_1 - r)}$, $\Sigma_1 \in \mathbf{R}^{r \times r}$, $Q_1 \in \mathbf{R}^{n \times r}$, $V \in \mathbf{R}^{d_2 \times d_2}$, $V_1 \in \mathbf{R}^{d_2 \times s}$, $V_2 \in \mathbf{R}^{d_2 \times (d_2 - s)}$, $\Sigma_2 \in \mathbf{R}^{s \times s}$, $Q_2 \in \mathbf{R}^{n \times s}$, U and V are orthogonal, Σ_1 and Σ_2 are nonsingular and diagonal, Q_1 and Q_2 are column orthogonal. It follows from the two orthogonality constraints in (2.2) that

$$l \leq \min\{\text{rank}(X), \text{rank}(Y)\} = \min\{r, s\} = t. \quad (3.3)$$

Next, let

$$Q_1^T Q_2 = P_1 \Sigma P_2^T \quad (3.4)$$

be the singular value decomposition of $Q_1^T Q_2$, where $P_1 \in \mathbf{R}^{r \times r}$ and $P_2 \in \mathbf{R}^{s \times s}$ are orthogonal and $\Sigma \in \mathbf{R}^{r \times s}$, then $m = \text{rank}(Q_1^T Q_2) \leq \min\{r, s\} = t$.

A solution subset of optimization problem (2.2) is described in the following lemma

Lemma 1: Any (W_x, W_y) of the following forms

$$\begin{cases} W_x = U_1 \Sigma_1^{-1} P_1(:, 1:l) + U_2 \mathcal{E}, \\ W_y = V_1 \Sigma_2^{-1} P_2(:, 1:l) + V_2 \mathcal{F}, \end{cases} \quad (3.5)$$

where $P_1(:, 1:l)$ denotes the first l columns of P_1, $\mathcal{E} \in \mathbf{R}^{(d_1 - r) \times l}$ and $\mathcal{F} \in \mathbf{R}^{(d_2 - s) \times l}$ are arbitrary, is a solution of optimization problem (2.2).

The proof of Lemma 1 can be found in [6], where a full characterization of all solutions of optimization problem (2.2) has been established.

Based on the explicit expression of solutions of optimization problem (2.2), we can now establish a relationship between CCA and least squares. Let

$$\begin{aligned} T_x &= Y^T [(YY^T)^{\frac{1}{2}}]^\dagger V_1 P_2(:, 1:l) \Sigma(1:l, 1:l)^{-1} \\ &= Q_2 P_2(:, 1:l) \Sigma(1:l, 1:l)^{-1}, \quad (3.6) \\ T_y &= X^T [(XX^T)^{\frac{1}{2}}]^\dagger U_1 P_1(:, 1:l) \Sigma(1:l, 1:l)^{-1} \\ &= Q_1 P_1(:, 1:l) \Sigma(1:l, 1:l)^{-1}, \quad (3.7) \end{aligned}$$

where $A^\dagger$ denotes the *Moore-Penrose* inverse of a general matrix A and $1 \leq l \leq m$, then we have the following theorem.

Theorem 2: For any l satisfying $1 \leq l \leq m$, suppose $W_x \in \mathbf{R}^{d_1 \times l}$ and $W_y \in \mathbf{R}^{d_2 \times l}$ satisfy

$$W_x = \arg\min\{\|X^T W_x - T_x\|_F^2 : W_x \in \mathbf{R}^{d_1 \times l}\}, \quad (3.8)$$

and

$$W_y = \arg\min\{\|Y^T W_x - T_y\|_F^2 : W_y \in \mathbf{R}^{d_2 \times l}\}, \quad (3.9)$$

where T_x and T_y are defined in (3.6) and (3.7), respectively. Then W_x and W_y form a solution of optimization problem (2.2).

Proof: Since (3.8) and (3.9) have the same form, we only prove the result for W_x, the same idea can be applied to W_y.

We know that W_x is a solution of (3.8) if and only if it satisfies the normal equation

$$XX^T W_x = XT_x. \quad (3.10)$$

Substituting factorizations (3.1), (3.2) and (3.4) into the equation above, we get

$$XX^T = U_1 \Sigma_1^2 U_1^T,$$

and

$$\begin{aligned} XT_x &= U_1 \Sigma_1 Q_1^T Q_2 P_2(:, 1:l) \Sigma(1:l, 1:l)^{-1} \\ &= U_1 \Sigma_1 P_1(:, 1:l), \end{aligned}$$

which yield an equivalent reformulation of (3.10)

$$U_1 \Sigma_1^2 U_1^T W_x = U_1 \Sigma_1 P_1(:, 1:l). \quad (3.11)$$

It is easy to check that W_x is a solution of (3.11) if and only if

$$W_x = U_1 \Sigma_1^{-1} P_1(:, 1:l) + U_2 \mathcal{E}, \quad (3.12)$$

where $\mathcal{E} \in \mathbf{R}^{(d_1 - r) \times l}$ is an arbitrary matrix. Therefore, W_x is a solution of (3.8) if and only if W_x can be formulated as (3.12).

Similarly, W_y is a solution of (3.9) if and only if W_y can be written as

$$W_y = V_1 \Sigma_2^{-1} P_2(:, 1:l) + V_2 \mathcal{F}, \quad (3.13)$$

where $\mathcal{F} \in \mathbf{R}^{(d_2 - s) \times l}$ is an arbitrary matrix.

Now, comparing equations (3.12) and (3.13) with the equation (3.5) in Lemma 1, we can conclude that for any solution W_x of the least squares problem (3.8) and any solution W_y of the least squares problem (3.9), W_x and W_y form a solution of optimization problem (2.2), hence a solution of CCA. ■

Remark 3.1: In Theorem 2 we only consider l satisfying $1 \leq l \leq m$. This is reasonable, since there are m nonzero canonical correlations between X and Y, and weight vectors corresponding to zero canonical correlation do not contribute to the canonical correlation between data X and Y.

IV. Sparse kernel CCA

Since kernel CCA criterion (2.7) and CCA criterion (2.2) have the same form, we can expect a similar characterization of solutions of (2.7) as Lemma 1. Define

$$\hat{r} = \text{rank}(K_x), \quad \hat{s} = \text{rank}(K_y), \quad \hat{m} = \text{rank}(K_x K_y),$$

and let the eigenvalue decomposition of K_x and K_y be, respectively,

$$\begin{aligned} K_x = \mathcal{U} \begin{bmatrix} \Pi_1 & 0 \\ 0 & 0 \end{bmatrix} \mathcal{U}^T &= \begin{bmatrix} \mathcal{U}_1 & \mathcal{U}_2 \end{bmatrix} \begin{bmatrix} \Pi_1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \mathcal{U}_1 & \mathcal{U}_2 \end{bmatrix}^T \\ &= \mathcal{U}_1 \Pi_1 \mathcal{U}_1^T, \quad (4.1) \end{aligned}$$

and

$$\begin{aligned} K_y = \mathcal{V} \begin{bmatrix} \Pi_2 & 0 \\ 0 & 0 \end{bmatrix} \mathcal{V}^T &= \begin{bmatrix} \mathcal{V}_1 & \mathcal{V}_2 \end{bmatrix} \begin{bmatrix} \Pi_2 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \mathcal{V}_1 & \mathcal{V}_2 \end{bmatrix}^T \\ &= \mathcal{V}_1 \Pi_2 \mathcal{V}_1^T, \quad (4.2) \end{aligned}$$

where

$$\mathcal{U} \in \mathbf{R}^{n \times n}, \ \mathcal{U}_1 \in \mathbf{R}^{n \times \hat{r}}, \ \mathcal{U}_2 \in \mathbf{R}^{n \times (n - \hat{r})}, \ \Pi_1 \in \mathbf{R}^{\hat{r} \times \hat{r}},$$

$$\mathcal{V} \in \mathbf{R}^{n \times n}, \ \mathcal{V}_1 \in \mathbf{R}^{n \times \hat{s}}, \ \mathcal{V}_2 \in \mathbf{R}^{n \times (n - \hat{s})}, \ \Pi_2 \in \mathbf{R}^{\hat{s} \times \hat{s}},$$

$\mathcal{U}$ and $\mathcal{V}$ are orthogonal, Π_1 and Π_2 are nonsingular and diagonal. In addition, let

$$\mathcal{U}_1^T \mathcal{V}_1 = \mathcal{P}_1 \Pi \mathcal{P}_2^T \quad (4.3)$$

be the singular value decomposition of $\mathcal{U}_1^T\mathcal{V}_1$, where $\mathcal{P}_1 \in \mathbf{R}^{\hat{r}\times\hat{r}}$ and $\mathcal{P}_2 \in \mathbf{R}^{\hat{s}\times\hat{s}}$ are orthogonal and $\Pi \in \mathbf{R}^{\hat{r}\times\hat{s}}$ is a diagonal matrix. Then we can prove for $1 \le l \le \min\{\hat{r},\hat{s}\}$ that

$$\begin{cases} \mathcal{W}_x = \mathcal{U}_1\Pi_1^{-1}\mathcal{P}_1(:,1:l) + \mathcal{U}_2\mathcal{E}, \\ \mathcal{W}_y = \mathcal{V}_1\Pi_2^{-1}\mathcal{P}_2(:,1:l) + \mathcal{V}_2\mathcal{F}, \end{cases} \tag{4.4}$$

with $\mathcal{E} \in \mathbf{R}^{(n-\hat{r})\times l}$ and $\mathcal{F} \in \mathbf{R}^{(n-\hat{s})\times l}$ being arbitrary matrices, form a subset of solutions to (2.7).

Solutions of (2.7) can also be associated with least squares problems. Define

$$\mathcal{T}_x = \mathcal{U}_1\mathcal{P}_1(:,1:l), \quad \mathcal{T}_y = \mathcal{V}_1\mathcal{P}_2(:,1:l), \tag{4.5}$$

with $1 \le l \le \hat{m}$, then each pair of $\mathcal{W}_x$ and $\mathcal{W}_y$, satisfying

$$\mathcal{W}_x = \arg\min\{\|K_x\mathcal{W}_x - \mathcal{T}_x\|_F^2 : \mathcal{W}_x \in \mathbf{R}^{n\times l}\},$$

and

$$\mathcal{W}_y = \arg\min\{\|K_y\mathcal{W}_y - \mathcal{T}_y\|_F^2 : \mathcal{W}_y \in \mathbf{R}^{n\times l}\},$$

respectively, forms a solution of (2.7).

Motivated by research on lasso [17] which shows that sparsity can be obtained by penalizing ℓ_1-norm of the variables, we incorporate sparsity into $\mathcal{W}_x$ and $\mathcal{W}_y$ by solving the following ℓ_1-norm regularized least squares problems

$$\begin{array}{ll} \min & \frac{1}{2}\|K_x\mathcal{W}_x - \mathcal{T}_x\|_F^2 + \sum\limits_{i=1}^{l}\lambda_{x,i}\|\mathcal{W}_{x,i}\|_1 \\ \text{subject to} & \mathcal{W}_x \in \mathbf{R}^{n\times l}, \end{array} \tag{4.6}$$

and

$$\begin{array}{ll} \min & \frac{1}{2}\|K_y\mathcal{W}_y - \mathcal{T}_y\|_F^2 + \sum\limits_{i=1}^{l}\lambda_{y,i}\|\mathcal{W}_{y,i}\|_1 \\ \text{subject to} & \mathcal{W}_y \in \mathbf{R}^{n\times l}, \end{array} \tag{4.7}$$

where $\lambda_{x,i}, \lambda_{y,i} > 0$ are regularization parameters, $\mathcal{W}_{x,i}$ and $\mathcal{W}_{y,i}$ are ith column of $\mathcal{W}_x$ and $\mathcal{W}_y$, respectively.

Since optimization problems (4.6) and (4.7) have the same form, all results holding for one problem can be naturally extended to the other, so we concentrate on (4.6). Note that when $l = 1$ optimization problem (4.6) reduces to a BPDN problem of the form (1.1), which has been intensively studied in the field of compressed sensing. Many efficient approaches have been proposed to solve the BPDN problem, among which we adopt the fixed-point continuation (FPC) method [7], due to its simple implementation and nice convergence property.

Fixed-point algorithm for (1.1) is an iterative method which updates iterates as

$$x^{k+1} = \mathcal{S}_\nu\left(x^k - \tau A^T(Ax - b)\right), \text{ with } \nu = \tau\lambda, \tag{4.8}$$

where $\tau > 0$ denotes the step size, and $\mathcal{S}_\nu$ is the soft-thresholding operator defined as

$$\mathcal{S}_\nu(x) = \operatorname{sign}(x) \odot \max\{|x| - \nu, 0\}, \ x \in \mathbf{R}^d, \tag{4.9}$$

with $\odot$ denoting component-wise multiplication. $\mathcal{S}_\nu(x)$ reduces all components of x with magnitude less than ν to zero, thus reducing the ℓ_1-norm and introducing sparsity.

The fixed-point algorithm can be extended to solve (4.6), which yields

$$\mathcal{W}_{x,i}^{k+1} = \mathcal{S}_{\nu_{x,i}}\left(\mathcal{W}_{x,i}^k - \tau_x K_x^T(K_x\mathcal{W}_{x,i}^k - \mathcal{T}_{x,i})\right), \tag{4.10}$$

where $i = 1,\cdots,l$, $\nu_{x,i} = \tau_x\lambda_{x,i}$ with $\tau_x > 0$ denoting the step size.

Algorithm 1 Sparse kernel CCA (SKCCA)

```
Input: Training data $X \in \mathbf{R}^{d_1\times n}$, $Y \in \mathbf{R}^{d_2\times n}$
Construct and center kernel matrices $K_x$, $K_y$;
Compute matrix factorizations (4.1)-(4.3);
Compute $\mathcal{T}_x$ and $\mathcal{T}_y$ defined in (4.5);
$\nu_{x,i} = \tau_x\lambda_{x,i}$, $\nu_{y,i} = \tau_y\lambda_{y,i}$, $i = 1,\cdots,l$,
repeat
    $\mathcal{W}_{x,i}^{k+1} = \mathcal{S}_{\nu_{x,i}}\left(\mathcal{W}_{x,i}^k - \tau_x K_x(K_x^T\mathcal{W}_{x,i}^k - \mathcal{T}_{x,i})\right)$,
until convergence
repeat
    $\mathcal{W}_{y,i}^{k+1} = \mathcal{S}_{\nu_{y,i}}\left(\mathcal{W}_{y,i}^k - \tau_y K_y(K_y^T\mathcal{W}_{y,i}^k - \mathcal{T}_{y,i})\right)$,
until convergence
Output: Sparse dual transformation matrices $\mathcal{W}_x^k$ and $\mathcal{W}_y^k$.
```

We can prove that fixed-point iterations have some nice convergence properties which are presented in the following theorem. Proof of the theorem can be found in [7].

Theorem 3: Let Ω be the solution set of (4.6), then there exists $M^* \in \mathbf{R}^{n\times l}$ such that

$$K_x^T(K_x\mathcal{W}_x - \mathcal{T}_x) \equiv M^*, \ \forall\, \mathcal{W}_x \in \Omega. \tag{4.11}$$

In addition, define

$$L := \{(i,j) : |M_{i,j}^*| < \lambda_{x,j}\} \tag{4.12}$$

as a subset of indices and let $\lambda_{max}(K_x^TK_x)$ be the maximum eigenvalue of $K_x^TK_x$, and choose τ_x from

$$0 < \tau_x < \frac{2}{\lambda_{max}(K_x^TK_x)},$$

then the sequence $\{\mathcal{W}_x^k\}$, generated by the fixed-point iterations (4.10) starting with any initial point $\mathcal{W}_x^0$, converges to some $\mathcal{W}_x^* \in \Omega$. Moreover, there exists an integer $K > 0$ such that

$$(\mathcal{W}_x^k)_{i,j} = (\mathcal{W}_x^*)_{i,j} = 0, \ \forall (i,j) \in L, \tag{4.13}$$

when $k > K$.

Remark 4.1: 1) Equation (4.11) shows that for any two optimal solutions of (4.6) the gradient of the squared Frobenius norm in (4.6) must be equal.

2) Equation (4.13) means that the entries of $\mathcal{W}_x^k$ with indices from L will converge to zero in finite steps. The positive integer K is a function of $\mathcal{W}_x^0$ and $\mathcal{W}_x^*$, and determined by the distance between them.

Similarly, we can design a fixed-point algorithm to solve (4.7) as follows:

$$\mathcal{W}_{y,i}^{k+1} = \mathcal{S}_{\nu_{y,i}}\left(\mathcal{W}_{y,i}^k - \tau_y K_y^T(K_y\mathcal{W}_{y,i}^k - \mathcal{T}_{y,i})\right), \tag{4.14}$$

where $i = 1,\cdots,l$, $\nu_{y,i} = \tau_y\lambda_{y,i}$ with $\tau_y > 0$ denoting the step size.

Applying fixed-point iterations (4.10) and (4.14) to ℓ_1-norm regularized least squares problems (4.6) and (4.7), we get a new sparse kernel CCA algorithm presented in Algorithm 1.

Since canonical correlations in kernel CCA depend only on kernel matrices K_x and K_y. Therefore, as we shall see from factorizations (4.1)-(4.3), canonical correlations in kernel CCA are determined by singular values of $\mathcal{U}_1^T\mathcal{V}_1$. The following proposition reveals a simple result regarding the distribution of canonical correlations.

Proposition 4: Let $\hat{r} = \text{rank}(K_x)$ and $\hat{s} = \text{rank}(K_y)$. If $\hat{r} + \hat{s} = n + \gamma$ for some $\gamma > 0$, then $\mathcal{U}_1^T\mathcal{V}_1$ has at least γ singular values equal to 1.

Proof: Since $\mathcal{U}_1 \in \mathbf{R}^{n\times\hat{r}}$, $\mathcal{U}_2 \in \mathbf{R}^{n\times(n-\hat{r})}$ and $\mathcal{V}_1 \in \mathbf{R}^{n\times\hat{s}}$ are column orthogonal and $\mathcal{U}_1\mathcal{U}_1^T + \mathcal{U}_2\mathcal{U}_2^T = I_n$, we have

$$(\mathcal{U}_1^T\mathcal{V}_1)^T\mathcal{U}_1^T\mathcal{V}_1 = \mathcal{V}_1^T\mathcal{U}_1\mathcal{U}_1^T\mathcal{V}_1 = I_{\hat{s}} - \mathcal{V}_1^T\mathcal{U}_2\mathcal{U}_2^T\mathcal{V}_1.$$

If there exist $\gamma > 0$ such that $\hat{r} + \hat{s} = n + \gamma$, then $n - \hat{r} = \hat{s} - \gamma < \hat{s}$ and

$$\text{rank}(\mathcal{V}_1^T\mathcal{U}_2\mathcal{U}_2^T\mathcal{V}_1) = \text{rank}(\mathcal{U}_2^T\mathcal{V}_1) \leq n - \hat{r},$$

which implies $\mathcal{V}_1^T\mathcal{U}_2\mathcal{U}_2^T\mathcal{V}_1$ has at least $\hat{s} - (n - \hat{r}) = \gamma$ zero eigenvalues. Thus, $(\mathcal{U}_1^T\mathcal{V}_1)^T\mathcal{U}_1^T\mathcal{V}_1$ has at least γ eigenvalues equal to 1, which further implies that $\mathcal{U}_1^T\mathcal{V}_1$ has at least γ singular values equal to 1. ■

In kernel methods, due to nonlinearity of kernel functions, the rank of kernel matrices is very close to n, which makes most canonical correlations to be 1. For instance, for Gaussian kernel

$$\kappa(x, y) = \exp\left(-\frac{1}{2\sigma^2}\|x - y\|^2\right), \ \sigma \neq 0 \qquad (4.15)$$

we can prove that the kernel matrix K_x given by $(K_x)_{ij} = \exp\left(-\frac{1}{2\sigma^2}\|x_i - x_j\|^2\right)$ has full rank, provided that the points $\{x_i\}_{i=1}^n$ are distinct [12]. Thus, in kernel methods we usually have

$$\hat{r} = \text{rank}(K_x) = n - 1, \quad \hat{s} = \text{rank}(K_y) = n - 1,$$

after centering data. In this case, all nonzero canonical correlations determined by the singular values of $\mathcal{U}_1^T\mathcal{V}_1$ are equal to 1. This means ordinary kernel CCA fails to provide a useful estimation of canonical correlations for general kernels, because for any distinct sample $\{x_i\}_{i=1}^n$ of variable x and distinct sample $\{y_i\}_{i=1}^n$ of variable y the canonical correlations returned by kernel CCA will be 1 even though variables x and y have no joint information.

To avoid forementioned data over-fitting problem in kernel CCA, researchers suggested to design a regularized kernelization of CCA [3], [9]. On the other hand, as shown in [17], the ℓ_1-penalty term can alleviate data overfitting problem while at the same time introduce sparsity. We can expect that sparse kernel CCA (4.6)-(4.7) enjoys the properties of both computing sparse $\mathcal{W}_x$, $\mathcal{W}_y$ and avoiding data over-fitting similar to regularized kernel CCA.

V. Experiments

In this section, we apply our newly proposed sparse kernel CCA algorithm SKCCA to content-based image retrieval (CBIR) by combining image and text data. CBIR is a challenging aspect of multimedia analysis and has become popular in past few years. Generally, it is the problem of searching for digital images in large databases by their visual content (e.g., color, texture, shape) rather than the metadata such as keywords, labels, and descriptions associated with the images. There exists study utilizing kernel CCA for image retrieval [9].

In the implementation of SKCCA, we need to determine regularization parameters $\{\lambda_{x,i}\}$ and $\{\lambda_{y,i}\}$. We know that x^* is a solution of BPDN problem (1.1) if and only if

$$0 \in A^T(Ax^* - b) + \lambda\partial\|x^*\|_1,$$

where $\partial\|x^*\|_1$ is the subgradient of ℓ_1-norm $\|\cdot\|_1$ at x^*. It follows that $x = 0$ is the solution of (1.1) when $\lambda \geq \|A^T b\|_\infty$. To avoid zero solution, which is meaningless in practice, we chose

$$\lambda_{x,i} = \gamma_x\|K_x^T\mathcal{T}_{x,i}\|_\infty, \ \lambda_{y,i} = \gamma_y\|K_y^T\mathcal{T}_{y,i}\|_\infty, \ i = 1, \cdots, l,$$

where $0 < \gamma_x, \gamma_y < 1$.

A `MATLAB` code implementing FPC algorithm for BPDN problem, named FPC_BB [22], is publicly available. We used this code in our implementation of Algorithm 1 with `xtol`=10^{-5} and `mxitr`=10^4 and all other parameters default.

We experiment on the Ground Truth Image Database [23] created at the University of Washington, which consists of 21 data sets of outdoor scene images. In our experiment we used 852 images form 19 data sets that have been annotated with keywords. We exploited text features and low-level image features, including color and texture, and applied sparse kernel CCA to perform image retrieval from text query. We used the bag-of-words approach to represent the text associated with images, Gabor filters to extract texture features and HSV (hue-saturation-value) color representation as color features.

Following previous work [9], we used Gaussian kernel

$$k_x(I_i, I_j) = \exp\left(-\frac{\|I_i - I_j\|^2}{2\sigma^2}\right),$$

where I_i is a vector concatenating texture features and color features of ith image and σ is the minimum distance between different images, to compute kernel matrix K_x for the first view. The linear kernel was employed to compute kernel matrix K_y using text features for the other view. We used 217 images as training data and the rest were used as testing data.

We compare the performance of CCA, kernel CCA and SKCCA in TABLE I, where the accuracy of image retrieval is measured by average area under the ROC curve (AROC), and for a collection of queries we use the average of retrieval precision of all queries as the average retrieval precision of this collection. More details about the evaluation of retrieval performance can be found in [6]. Results in TABLE I were obtained by letting $l = \hat{m} = \text{rank}(K_x K_y)$, that is, projections corresponding to all nonzero canonical correlations were used. 'Corr' denotes the summation of canonical correlations between testing data, 'Sparsity' column records sparsity of both $\mathcal{W}_x$ and $\mathcal{W}_y$, which is the percentage of zero entries in the matrices. The first component records sparsity of $\mathcal{W}_x$ while the second component records sparsity of $\mathcal{W}_y$. The '(γ_x, γ_y)' column records value of regularization parameters in SKCCA.

TABLE I: CCA, kernel CCA and SKCCA for content-based image retrieval.

Algorithms	AROC	Corr	Sparsity (%)	l	(γ_x, γ_y)
CCA	0.7396	11.53	(0, 7.7)	124	-
KCCA	0.8259	19.37	(0, 0)	124	-
SKCCA	0.8489	24.98	(91.1, 88.4)	124	(0.5, 0.3)

From TABLE I, we observe that SKCCA have the best retrieval performance in terms of precision. It also obtains

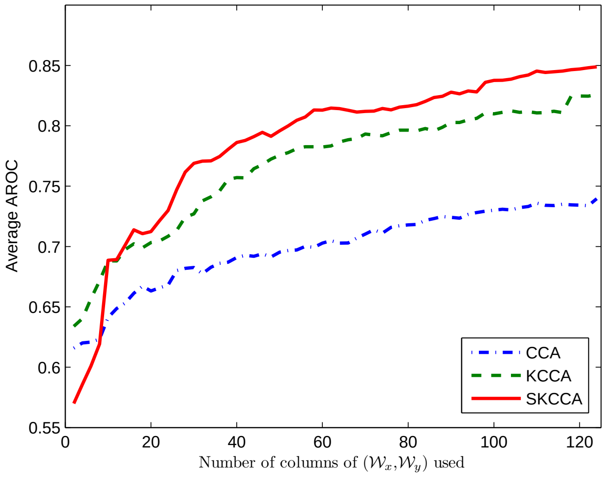

Fig. 1: Content-based image retrieval using CCA, kernel CCA and SKCCA with 217 images in the training data and 635 images in testing data.

larger summation of canonical correlations between testing data than other two approaches, which empirically shows that SKCCA is better than CCA for finding nonlinear relations and alleviates the over-fitting problem of KCCA. We can also see that sparsity of the dual projections $\mathcal{W}_x$ and $\mathcal{W}_y$ computed by SKCCA is greater than 88%, which can excessively reduce the computational time of computing projection of a new data in practice as we only need to evaluate kernel functions between the new data and a small subset of training data.

In Fig. 1, we plot AROC of CCA, kernel CCA and SKCCA as a function of the number of projections used (i.e., different l). As visible in Fig. 1, the AROC of all approaches gradually increases when more projections are used for retrieval. This is reasonable, beacause when we increase l more projections corresponding to nonzero canonical correlations are used for retrieval and these added projections may convey information contained in the training data. In addition, we observe that the AROC of SKCCA is at first smaller than and then exceeds that of kernel CCA. This indicates that when suitable number of dual projections are used for retrieval SKCCA can improve the performance of kernel CCA.

VI. Conclusions

In this paper we proposed a novel sparse kernel CCA algorithm called SKCCA. This algorithm is based on a relationship between kernel CCA and least squares which is an extension of a similar relationship between CCA and least squares. We incorporated sparsity into kernel CCA by penalizing the ℓ_1-norm of dual vectors. The resulting ℓ_1-regularized minimization problems were solved by a fixed-point continuation (FPC) algorithm. Empirical results show that SKCCA not only performs well in computing sparse dual transformations, but also alleviates the over-fitting problem of kernel CCA.

Several interesting questions and extensions remain. In many applications such as genomic data analysis, CCA is often performed on more than two data sets. It will be helpful to extend sparse kernel CCA to deal with multiple data sets. In the derivation of SKCCA, we did not discuss the choice of kernel function. However, it is believed that the performance of kernel CCA depends on the choice of the kernel. As for future research, we plan to study the problem of finding the optimal kernel of kernel CCA for different applications, as in the case of kernel FDA [11]. Moreover, we also plan to generalize the idea of sparse kernel CCA in this paper to involve multiple kernels.

References

[1] S. Akaho, “A Kernel Method For Canonical Correlation Analysis”, *In Proceedings of the International Meeting of the Psychometric Society*, 2001.

[2] N. Aronszajn, “Theory of Reproducing Kernels,” *Transactions of the American Mathematical Society*, vol. 68, no. 3, pp. 337-404, 1950.

[3] F. Bach and M. Jordan, “Kernel Independent Component Analysis”, *Journal of Machine Learning Research*, vol. 3, pp. 1-48, 2003.

[4] S. Balakrishnan, K. Puniyani and J. Lafferty, “Sparse Additive Functional and Kernel CCA,” *Proceedings of 29th International Conference of Machine Learning*, 2012.

[5] C. M. Bishop, *Pattern Recognition and Machine Learning*, Springer, 2006.

[6] D. Chu, L. Liao, M. K. Ng and X. Zhang, “Sparse Canonical Correlation Analysis: New Formulation and Algorithm,” *Submitted.*

[7] E. T. Hale, W. Yin and Y. Zhang, “Fixed-Point Continuation for ℓ_1-Minimization: Methodology and Covergence,” *SIAM J. Opt.*, vol. 19, no. 3, pp. 1107-1130, 2008.

[8] D. R. Hardoon and J. R. Shawe-Tayler, “Sparse Canonical Correlation Analysis,” *Machine Learning*, vol. 83, no. 3, pp. 331-353, 2011.

[9] D. R. Hardoon, S. R. Szedmak and J. R. Shawe-Taylor, “Canonical Correlation Analysis: an Overview with Application to Learning Methods,” *Neural Computation*, vol. 16, no. 12, pp. 2639-2664, 2004.

[10] H. Hotelling, “Relations between Two Sets of Variables,” *Biometrika*, vol. 28, pp. 321-377, 1936.

[11] S. Kim, A. Magnani and S. Boyd, “Optimal Kernel Selection in Kernel Fisher Discriminant Analysis,” *The 23th Int’l Conf. Machine Learning*, 2006.

[12] B. Schölkopf and A. Smola, *Learning with Kernels : Support Vector Machines, Regularization, Optimization, and Beyond*, MIT Press, 2002.

[13] B. Schölkopf, A. Smola and K. Müller, “Nonlinear Component Analysis as a Kernel Eigenvalue Problem,” *Neural Computation*, vol. 10, pp. 1299-1319, 1998.

[14] L. Sun, S. Ji and J. Ye, “A Least Squares Formulation for Canonical Correlation Analysis,” In *The 25th Int’l Conf. Machine Learning*, pp. 1024-1031, 2008.

[15] L. Sun, S. Ji and J. Ye, “Canonical Correlation Analysis for Multi-Label Classification: A Least Squares Formulation, Extensions and Analysis,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 1, pp. 194-200, 2011.

[16] L. Tan and C. Fyfe, “Sparse Kernel Canonical Correlation Analysis”, *Proceedings of 9th European Symposium on Artificial Neural Networks*, pp. 335-340, 2001.

[17] R. Tibshirani, “Regression Shrinkage and Selection via the Lasso,” *Journal of the Royal Statistical Society (Series B)*, vol. 58, pp. 267-288, 1996.

[18] A. Vinokourov, J. R. Shawe-Taylor and N. Cristianini, “Inferring a Semantic Representation of Text via Cross-Language Correlation Analysis,” In S.Becker, S.Thrun and K. Obermayer (eds.), *Advances in Neural Information Processing Systems*, Cambridge:MIT Press, 2003.

[19] D. M. Witten and R. Tibshirani, “Extensions of Sparse Canonical Correlation Analysis with Applications to Genomic Data,” *Statistical Applications in Genetics and Molecular Biology*, 8, 2009. Issue 1, Article 28.

[20] D. M. Witten, R. Tibshirani and T. Hastie, “A Penalized Matrix Decomposition, with Applications to Sparse Principal Components and Canonical Correlation Analysis,” *Biostatistics*, vol. 10, no. 3, pp. 515-534, 2009.

[21] Y. Yamanishi, J. P. Vert, A. Nakaya and M. Kanehisa, “Extraction of Correlated Gene Clusters from Multiple Genomic Data by Generalized Kernel Canonical Correlation Analysis,” *Bioinformatics*, vol. 19(Suppl 1), pp. i323–i330, 2003.

[22] MATLAB Code for FPC_BB, http://www.caam.rice.edu/~optimization/L1/fpc/.

[23] Ground Truth Image Database, http://www.cs.washington.edu/research/imagedatabase/groundtruth/.