Multi-view Common Component Discriminant Analysis for Cross-view Classification

Xinge You†, Senior Member, IEEE, Jiamiao Xu, Wei Yuan, Xiao-Yuan Jing, Dacheng Tao, Fellow, IEEE, and Taiping Zhang

Abstract—Cross-view classification that means to classify samples from heterogeneous views is a significant yet challenging problem in computer vision. A promising approach to handle this problem is the multi-view subspace learning (MvSL), which intends to find a common subspace for multi-view data. Despite the satisfactory results achieved by existing methods, the performance of previous work will be dramatically degraded when multi-view data lies on nonlinear manifolds. To circumvent this drawback, we propose Multi-view Common Component Discriminant Analysis (MvCCDA) to handle view discrepancy, discriminability and nonlinearity in a joint manner. Specifically, our MvCCDA incorporates supervised information and local geometric information into the common component extraction process to learn a discriminant common subspace and to discover the nonlinear structure embedded in multi-view data. We develop a kernel method of MvCCDA to further boost the performance of MvCCDA. Beyond kernel extension, optimization and complexity analysis of MvCCDA are also presented for completeness. Our MvCCDA is competitive with the state-of-the-art MvSL based methods on four benchmark datasets, demonstrating its superiority.

Index Terms—cross-view classification, local geometry preservation, multi-view learning, subspace learning.

I. INTRODUCTION

OBJECTS captured at various viewpoints or by different sensors can generate multi-view data [1], [2]. Although more information is provided, there is a large discrepancy among views [3], [4]. This results in a challenging problem for categorizing samples when the gallery and probe data are from heterogeneous views. The problem is also known as cross-view classification. Formally speaking, given a trained model T, a gallery set Dv in view v and a probe data qu from view u, cross-view classification intends to obtain the label of qu using a classifier from Du.

Cross-view classification has received much attention in recent years due to its widespread applications in computer vision. Take face recognition as an example, the gallery set is usually frontal face images, whereas faces from any pose may appear in the probe set (e.g., multi-pose faces exist extensively in videos). However, directly recognizing the profile faces via a classifier trained on frontal face set is impractical due to the large view discrepancy between frontal faces and profile faces.

To tackle cross-view classification problem, early work is based on distance metric learning (DML) [5], which attempts to construct a new distance measurement describing the similarity of heterogeneous samples (i.e., samples from diverse views) [6]–[11]. Examples of DML based methods include the Large Margin Nearest Neighbor (LMNN) [12] and the Joint Graph Regularized Heterogeneous Metric Learning (JGRHML) [13]. The recent work in this area is based on multi-view subspace learning (MvSL). Different from DML, MvSL based methods tend to use a subspace learning strategy to project multi-view data into a latent common space, in which the view discrepancy can be removed and the similarity of heterogeneous samples can be measured with traditional Euclidean distance [14]–[25].

Substantial efforts have been made to remove view discrepancy. One of the most well-known method is Canonical Correlation Analysis (CCA) [26], [27]. CCA attempts to learn two view-specific transforms to respectively project data into a latent common subspace in which the correlation between two-view projected samples is maximized. Multi-view Canonical Correlation Analysis (MCMA) [28], [29] was later proposed as a generalization of CCA in multi-view scenario. Another commonly used approach is Partial Least Square (PLS) Regression [30]. Different from CCA, PLS learns two linear mapping functions such that the covariance of projected samples from two views is maximized. Although the view discrepancy can be removed by aforementioned methods, the neglect of the supervised information may give rise to the performance degeneration. To overcome this shortcoming, Generalized Multi-view Analysis (GMA) [3] extends MCCA to a supervised algorithm by taking into consideration intra-view discriminant information. Following this, Multi-view Discriminant Analysis (MvDA) [31] and Multi-view Modular Discriminant Analysis (MvMDA) [32] were proposed to further consider inter-view discriminability, leading to a more discriminant subspace. Despite the satisfactory performance achieved on real applications, the global linearity of above methods limits their effectiveness for nonlinear distributed samples [33]. To circumvent the linear limitation, these methods can be naturally extended to nonlinear algorithms with kernel trick (e.g., Kernel Canonical Correlation Analysis (KCCA) [34], [35] and Kernel Generalized Multi-view Analysis (KGMA) [3]). However, it is trivial to design a desirable kernel [36] and the kernel based methods suffer from “curse of kernelization”
on large-scale dataset \cite{37}. On the other hand, deep neural network (DNN) also offers a feasible solution to handle nonlinearity (e.g., Multi-view Deep Network (MvDN) \cite{36}, Deep Canonical Correlation Analysis (DCCA) \cite{38,39}). Same as other DNN approaches that have been developed in other computer vision tasks, it remains a question on the optimal selection of network topology and the performance of network deteriorates dramatically theoretically and practically when the training data is insufficient. Consequently, there remains a need for an cross-view classification algorithm that can effectively handle nonlinearity.

To capture nonlinear information while circumventing the drawbacks of existing work, a general idea that initiated in manifold learning is to enforce the projected samples to preserve the local geometry of observed samples, thus significantly improving its representation power. Albeit its simplicity, it is hard to implement this idea in cross-view classification. This is because we cannot precisely measure the similarity of heterogeneous samples in observation space due to the large discrepancy between pairwise views, whereas this operation is vital for local geometry preservation algorithms (e.g., Locally Linear Embedding (LLE) \cite{40} and Locality Preserving Projections (LPP) \cite{41}).

This work also deals with nonlinear challenge. Motivated by the idea of local geometry preservation, we propose a novel cross-view classification algorithm, which imposes prior knowledge as regularization terms on the common component extraction process, namely discriminant regularization and local consistency regularization. Specifically, the former enforces the convergence of within-class common components to a small region and between-class common components to be far from each other in latent subspace, whereas the latter enforces the common components to preserve the local geometric structure of objects.

To summarize, our main contributions are threefold:

1) A novel algorithm, named MvCCDA, is proposed for cross-view classification and an alternating algorithm is presented to efficiently optimize MvCCDA.

2) The local geometry preservation, initiated in single-view manifold learning algorithm, has been extended to multi-view scenario, thus enabling us to preserve the local geometry of multi-view data.

3) Extensive experiments conducted on four datasets show that our method can effectively capture the nonlinear structure of multi-view data, which thus boosts the cross-view classification performance.

The remainder of this paper is arranged as follows. Sect. II introduces the related work. In Sect. III, we present the formulation of our MvCCDA, and also develop an alternating algorithm to solve the optimization problem efficiently. Following this, the kernel extension and complexity analysis of MvCCA are conducted. The performance of our method is evaluated on four benchmark datasets in Sect. IV and Sect. V concludes this paper.

II. RELATED WORK

In this section, we first summarize some notations that will be used throughout this paper and briefly review relevant multi-view subspace learning (MvSL) based approaches. We also introduce some fundamental knowledge on local geometry preservation to make interested readers more familiar with our method that will be elaborated in Sect. III.

A. Multi-view Subspace Learning based Methods

Suppose \( \{o^1, o^2, \ldots, o^m\} \) denotes an object dataset, where \( m \) is the number of objects. \( X = \{x^1_1, x^1_2, \ldots, x^1_n\} \) is its multi-view representations, where \( \{x^1_1, x^1_2, \ldots, x^1_n\} \) are paired samples of object \( o^1 \) and \( n (n \geq 2) \) is the number of views. \( x^1_i \in \mathbb{R}^{d_v} \) is the sample of object \( o^1 \) under the \( v \)-th view embedded in \( d_v \)-dimensional space. \( x^v_{ic} \) denotes that the sample \( x^v_{ic} \) is from the \( c \)-th class, where \( c \in \{1,2,\ldots,C\} \) and \( C \) is the number of categories. Let \( X_v = [x^v_1, x^v_2, \ldots, x^v_m] \) denote data matrix from the \( v \)-th view.

To remove view discrepancy, MvSL based methods aim to learn multiple view-specific mapping functions \( \{P_v\}_{v=1}^n \) to project multi-view data into a latent common subspace \( Z \), where \( P_v \in \mathbb{R}^{d_v \times d} \) stands for the \( v \)-th view mapping function. To make the presentation easier to follow, let \( \mu_v^c \) denote the mean of samples of the \( c \)-th class from the \( v \)-th view in \( Z \), \( \mu^c \) denote the mean of samples of the \( c \)-th class over all views in \( Z \), and \( \mu \) denote the mean of all samples over all views in \( Z \). Also let \( N^c_v \) denote the number of samples of the \( c \)-th class from the \( v \)-th view, and \( N^c \) denote the number of all samples of \( c \)-th class over all views. \( \text{tr}(\cdot) \) denotes the trace operator. We introduce relevant MvSL based approaches for cross-view classification as below.

1) CCA: Canonical Correlation Analysis (CCA) \cite{26} attempts to respectively project two-view data \( X_1 \) and \( X_2 \) into a latent common subspace, in which the projected representations \( P_1^TX_1 \) and \( P_2^TX_2 \) are most correlated, where linear mapping functions \( P_1 \) and \( P_2 \) are optimized as follows:

\[
\max_{P_1, P_2} P_1^TX_1X_2^TP_2 \\
\text{s.t. } P_1^TX_1X_1^TP_1 = 1, \quad P_2^TX_2X_2^TP_2 = 1. \tag{1}
\]

The performance of CCA deteriorates dramatically when two-view data is non-linearly embedded in observation space.

2) KCCA: Kernel Canonical Correlation Analysis (KCCA) \cite{34} was later proposed to extend CCA to a nonlinear model with the famed kernel trick. The objective of KCCA can be formulated via representer theorem \cite{42} as follows:

\[
\max_{A_1, A_2} A_1^TK_1K_2A_2 \\
\text{s.t. } A_1^TK_1K_1A_1 = 1, \quad A_2^TK_2K_2A_2 = 1. \tag{2}
\]

where \( K_1 \) and \( K_2 \) are kernel matrices with respect to \( X_1 \) and \( X_2 \) respectively. \( A_1 \) and \( A_2 \) are atom matrices of corresponding views. The attentive reader will notice that CCA and KCCA are only designed for two-view data.

3) MCCA: Multiview Canonical Correlation Analysis \cite{28} generalizes CCA to multi-view scenario. MCCA is to find multiple linear transforms \( \{P_v\}_{v=1}^n \) to respectively project
\{X_1, X_2, \ldots, X_n\} into a latent subspace where the sum of all pairwise correlations is maximized:
\[
\max_{P_1, P_2, \ldots, P_n} \sum_{i<j} P_i^T X_i X_j^T P_j \\
\text{s.t. } P_i^T X_i X_i^T P_i = 1, \quad i = 1, 2, \ldots, n. \tag{3}
\]

The neglect of supervised information inhibits its performance in cross-view classification.

4) GMA: Generalized Multiview Analysis (GMA) [3] generalizes MCCA to a supervised algorithm by the proper utilization of intra-view discriminant information. The objective of GMA can be expressed as:
\[
\max_{P_1, P_2, \ldots, P_n} \sum_{i=1}^{n} \mu_i P_i^T A_i P_i + \sum_{i<j} 2\lambda_{ij} P_i^T X_i X_j^T P_j \\
\text{s.t. } \sum_{i} \gamma_i P_i^T B_i P_i = 1, \tag{4}
\]
where \(A_i\) and \(B_i\) are within-class and between-class scatter matrices respectively. As a MvSL based supervised method, GMA suffers from the ignorance of inter-view discriminant information.

5) MvDA: Multi-view Discriminant Analysis (MvDA) [31] integrates view correlation, intra-view discriminability and inter-view discriminability in a joint manner. Its objective can be formulated as a generalized Rayleigh quotient:
\[
(P_1^*, P_2^*, \ldots, P_n^*) = \arg \max_{P_1, \ldots, P_n} \frac{\text{tr}(S_B)}{\text{tr}(S_W)}, \tag{5}
\]
where the within-class scatter matrix \(S_W\) and between-class matrix \(S_B\) of the projected samples are defined as:
\[
S_W = \sum_{v=1}^{C} \sum_{i=1}^{N_v} (P_v^T x_{vi} - \mu_v^c)(P_v^T x_{vi} - \mu_v^c)^T, \\
S_B = \sum_{v=1}^{C} N_v (\mu_v - \mu)(\mu_v - \mu)^T. \tag{6}
\]
MvDA-VC [1] was proposed thereafter to boost the performance of MvDA by further considering view-consistency.

6) MvMDA: Multi-view Modular Discriminant Analysis (MvMDA) [32] was recently proposed to separate class centers across different views. The objective of MvMDA is formulated as:
\[
(P_1^*, P_2^*, \ldots, P_n^*) = \arg \max_{P_1, \ldots, P_n} \frac{\text{tr}(S_B)}{\text{tr}(S_W)}, \tag{7}
\]
where the within-class scatter matrix \(S_W\) and between-class matrix \(S_B\) of the projected samples are defined as:
\[
S_B = \sum_{v=1}^{C} \sum_{i=1}^{N_v} \sum_{p=1}^{C} (\mu_{i}^p - \mu_v^c)(\mu_{i}^p - \mu_v^c)^T, \\
S_W = \sum_{v=1}^{C} \sum_{i=1}^{N_v} (P_v^T x_{vi}^c - \mu_v^c)(P_v^T x_{vi}^c - \mu_v^c)^T. \tag{8}
\]

Despite the satisfactory performance achieved by GMA, MvDA or MvMDA, these methods are all incapable of handling nonlinear scenario due to their linear essence [33].

B. Local Geometry Preservation
To discover the underlying nonlinear structure automatically, our algorithm is expected to be able to precisely preserve local geometry of observed objects. Canonical examples of the local geometry preservation paradigm include Local Linear Embedding (LLE) [40], Local Tangent Space Alignment (LTSA) [43] and Locality Preservation Projection (LPP) [41]. Despite their dramatic success and strong representation power, one should note that local geometry preservation technology used in those work cannot be directly applied to multi-view scenario. Before elaborating our solutions in Sect. III without loss of generality, we introduce LPP below for completeness.

Let \(X = \{x_1, x_2, \ldots, x_m\}\) denote a dataset, where \(x_i \in \mathbb{R}^{D \times 1}\) is the i-th sample embedded in D-dimensional observation space and \(m\) is the number of samples. Suppose we are given a mapping function \(f: \mathcal{X} \rightarrow \mathcal{Y}\), where \(\mathcal{Y} = \{y_1, y_2, \ldots, y_m\}\), and \(y_i \in \mathbb{R}^{D \times 1}\). Then, LPP preserves local neighborhood structure of \(X\) by minimizing the following objective:
\[
\sum_{i=1}^{m} \sum_{j=1}^{m} \|y_i - y_j\|^2 W(i, j), \tag{9}
\]
where \(W\) is the similarity matrix of observed samples. \(W(i, j)\) that refers to the similarity of \(x_i\) and \(x_j\) is usually determined by heat kernel:
\[
S(i, j) = \exp\left(-\frac{\|x_i - x_j\|^2}{\sigma}\right), \tag{10}
\]
\[
W(i, j) = \begin{cases} S(i, j), \quad x_i \in N_k(x_j) \text{ or } x_j \in N_k(x_i); \\ 0, \quad \text{others,} \end{cases} \tag{11}
\]
where \(N_k(x_i)\) is the set of \(k\) nearest neighbors of \(x_i\), \(k > 0\) can be determined by cross validation.

It is worth noting that the regularization loss in (9) suffers from a heavy penalty, when adjacent samples \(x_i\) and \(x_j\) are mapped far apart. This property ensures that the projected representations can preserve the local geometry of observed samples.

III. Multi-view Common Component Discriminate Analysis and Its Kernel Extension
In this section, we first detail the motivation and the general framework of our Multi-view Common Component Discriminate Analysis (MvCCDA), and then present its optimization procedure. We also conduct the kernel extension and complexity analysis of MvCCDA.

A. Multi-view Common Component Discriminate Analysis
Existing MvSL based methods fail to simultaneously handle three crucial subproblems: view discrepancy, discriminability and nonlinearity, thus inhibiting their performance in cross-view classification. Motivated by manifold learning, we propose a model which integrates supervised information and local geometric information into the common component extraction process to elegantly and effectively handle all three
subproblems in a joint manner. We term the novel model MvCCDA. An overview of MvCCDA is shown in Fig. 1.

Paired samples come from different views on the same object (e.g., multi-pose face images of a person). Recent work in this area (e.g., 3, 36) has indicated that view discrepancy can be removed by extracting the common characteristics of paired samples. Therefore, we require projected samples \( \{P^i_1 x_1^i, P^i_2 x_2^i, \ldots, P^i_n x_n^i\} \) of the \( i \)-th paired samples \( \{x_1^i, x_2^i, \ldots, x_n^i\} \) converge to a common component \( z^i \) in the latent space \( Z \), where \( z^i \in \mathbb{R}^d \). Thus, the objective can be formulated as follows:

\[
\frac{1}{n} \sum_{v=1}^{n} \log \left( 1 + \frac{\|z^i - P^T x_v^i\|^2}{a^2} \right), \tag{12}
\]

where \( a \) is a constant. Cauchy loss used in (12) helps to reduce the influence of noise or outliers to some degree [42, 43].

By performing feature extraction over all the paired samples, we can obtain the overall objective as:

\[
\min_{z, P} \frac{1}{mn} \sum_{i=1}^{m} \sum_{v=1}^{n} \log \left( 1 + \frac{\|z^i - P^T x_v^i\|^2}{a^2} \right) + \lambda_1 \sum_{v=1}^{n} \|P_v\|^2 + \lambda_2 \sum_{i=1}^{m} \|z^i\|^2, \tag{13}
\]

where \( \lambda_1 \) and \( \lambda_2 \) are regularization parameters. Obviously, common components can be learned by minimizing (13).

To improve discriminability, we first define the label indicator vector \( l \in \mathbb{R}^{C \times 1} \) for convenience. We set \( l(j) = 1 \) if sample \( x \) is from the \( j \)-th class and \( l(j) = 0 \) otherwise, where \( l(j) \) refers to the \( j \)-th element of \( l \). To sum up, if a sample \( x \) is from the \( j \)-th class, its corresponding \( l \) is defined as:

\[
l = \begin{bmatrix} j-1 \vdots 0 \vdots 0 \vdots 0 \end{bmatrix}^T.
\]

Subsequently, we denote \( \{P^i\}_{i=1}^{m} \) as the label indicator vectors of common components \( \{z^i\}_{i=1}^{m} \) that satisfy \( d = C \).

Then supervised information can be incorporated into model using the following discriminant regularization:

\[
\min_{z, P} \frac{1}{mn} \sum_{i=1}^{m} \sum_{v=1}^{n} \log \left( 1 + \frac{\|z^i - P^T x_v^i\|^2}{a^2} \right) + \lambda_1 \sum_{v=1}^{n} \|P_v\|^2 + \lambda_2 \sum_{i=1}^{m} \|z^i - l^i\|^2.
\]

A discriminant subspace can be learned by minimizing (16). One should note that (16) is incapable of discovering the nonlinear structure embedded in multi-view data. To extend Eq. (16) to a nonlinear model, we generalize the traditional local geometry preservation framework illustrated in Sect. II-B to multi-view scenario.

Denote

\[
S_v(i, j) = \exp \left( -\frac{\|x_v^i - x_v^j\|^2}{\sigma_v} \right), \tag{17}
\]

\[
S(i, j) = \frac{1}{n} \sum_{v=1}^{n} S_v(i, j), \tag{18}
\]

where \( S_v \) and \( S \) denote the similarity matrix of samples under the \( v \)-th view and the similarity matrix of objects \{\( o_v \}\}_{i=1}^{m} respectively.

Further denote

\[
W(i, j) = \begin{cases} S(i, j), & \text{if } o_v \in N_k(o^i) \text{ or } o^j \in N_k(o^i); \\ 0, & \text{others}, \end{cases}
\]

where \( W \) denotes the local similarity matrix of objects.

As revealed in [40, 41], a feasible method to handle nonlinearity is to require common components in \( Z \) precisely
preserve the local geometric structure of objects. To this end, we incorporate local geometry information into model using the following local consistency regularizer:

$$
\sum_{i=1}^{m} \sum_{j=1}^{m} \left\| z^i - z^j \right\|_{2}^{2} W(i, j).
$$

(20)

Note that regularization loss in (20) suffers from a heavy penalty when adjacent objects $o^i$ and $o^j$ are mapped far apart.

Combining (16) with (20), we have

$$
\min_{z^i} \frac{1}{m} \sum_{i=1}^{m} \log \left(1 + \frac{\left\| z^i - P^T x^i \right\|_{2}^{2}}{a^2}\right) + \lambda_1 \sum_{i=1}^{m} \left\| P_v \right\|_{F}^{2} + \lambda_2 \sum_{i=1}^{m} \left\| z^i - l^i \right\|_{2}^{2} + \lambda_3 \sum_{i=1}^{m} \sum_{j=1}^{m} \left\| z^i - z^j \right\|_{2}^{2} W(i, j).
$$

(21)

Eq. (21) is the objective of the proposed MvCCDA.

**B. The Solution to MvCCDA**

Problem (21) can be decomposed into two subproblems with respect to common components $\{z^i\}_{i=1}^{m}$ and mapping functions $\{P_v\}_{v=1}^{m}$ using the alternating optimization method.

**Subproblem I:** The first subproblem decomposed from (21) with respect to common component $z^i$ is defined as:

$$
\min_{z^i} \mathcal{J}_{z^i} = \frac{1}{n} \sum_{v=1}^{n} \log \left(1 + \frac{\left\| z^i - P^T x^i \right\|_{2}^{2}}{a^2}\right) + \lambda_2 \left\| z^i - l^i \right\|_{2}^{2} + \lambda_3 \sum_{j=1}^{m} \left\| z^i - z^j \right\|_{2}^{2} W(i, j).
$$

(22)

Setting the gradient of $\mathcal{J}_{z^i}$ with respect to $z^i$ to 0, we have

$$
\sum_{v=1}^{n} \frac{1}{a^2 + \left\| r_v \right\|_{2}^{2}} + \lambda_2 + \lambda_3 \sum_{j=1}^{m} W(i, j) z^i + \lambda_3 \sum_{j=1}^{m} z^j W(i, j) = 0.
$$

(23)

(23) can be simplified to

$$
\left( \sum_{v=1}^{n} \frac{1}{a^2 + \left\| r_v \right\|_{2}^{2}} + \lambda_2 + \lambda_3 \sum_{j=1}^{m} W(i, j) \right) z^i = \left( \sum_{v=1}^{n} \frac{P^T x^v}{a^2 + \left\| r_v \right\|_{2}^{2}} + \lambda_3 \sum_{j=1}^{m} W(i, j) z^j \right),
$$

(24)

where $r_v = z^i - P^T x^v$. We further define a weight function as:

$$
Q = \left[ \frac{1}{a^2 + \left\| r_1 \right\|_{2}^{2}}, \cdots, \frac{1}{a^2 + \left\| r_n \right\|_{2}^{2}} \right].
$$

(25)

Combining (24) with (25), we have

$$
\left( \sum_{v=1}^{n} Q(v) \right) + \lambda_2 + \lambda_3 \sum_{j=1}^{m} W(i, j) z^i = \left( \sum_{v=1}^{n} P^T Q(v) x^v + \lambda_3 \sum_{j=1}^{m} W(i, j) z^j \right),
$$

(26)

where $Q(v)$ refers to the $v$-th element of $Q$. We thus iteratively update $z^i$ using (26) with a starting value until convergence. The detailed procedure is summarized in Algorithm 1.

**Algorithm 1:** The iterative procedure of Subproblem I

---

**Input:** $\{x^1, x^2, \ldots, x^m\}, \{P_v\}_{v=1}^{m}, \{z^j\}_{j=1}^{m}, l^i$ and $W$

**Output:** $z^i$

1. $(z^i)^0 = z^i$;
2. for $k = 1, \ldots, d$
3. calculate residuals $\{r_v\}_{v=1}^{n} = (z^i)^{k-1}$;
4. update $Q$ using Eq. (25);
5. update $(z^i)^{k}$ using Eq. (26);
6. if the estimate of $z^i$ converges then
   7. break;
8. $(z^i)^{k}$;
9. return $z^i$;

---

**Subproblem II:** The second subproblem decomposed from (21) with respect to mapping function $P_v$ is defined as:

$$
\min_{P_v} \mathcal{J}_{P_v} = \frac{1}{m} \sum_{i=1}^{m} \log \left(1 + \frac{\left\| z^i - P^T x^i \right\|_{2}^{2}}{a^2}\right) + \lambda_1 \left\| P_v \right\|_{F}^{2}.
$$

(27)

Setting the gradient of $\mathcal{J}_{P_v}$ with respect to $P_v$ to zero, we have

$$
\sum_{i=1}^{m} \frac{1}{a^2 + ||r^i||_2^2} + m\lambda_1 = 0.
$$

(28)

(28) can be simplified to:

$$
\left( \sum_{i=1}^{m} \frac{x^v (z^i)^T}{a^2 + \left\| r^i \right\|_{2}^{2}} + m\lambda_1 \right) P_v = \sum_{i=1}^{m} \frac{x^v (z^i)^T}{a^2 + \left\| r^i \right\|_{2}^{2}},
$$

(29)

where $r^i = z^i - P^T x^v$. We further define a weight function as:

$$
G = \left[ \frac{1}{a^2 + \left\| r^1 \right\|_{2}^{2}}, \cdots, \frac{1}{a^2 + \left\| r^n \right\|_{2}^{2}} \right].
$$

(30)

Combining (29) with (30), we have

$$
P_v = \left( \sum_{i=1}^{m} x^v G(i) (z^i)^T \right) + m\lambda_1 \left( \sum_{i=1}^{m} x^v G(i) (z^i)^T \right)^{-1},
$$

(31)

where $G(i)$ refers to the $i$-th element of $G$. Similar to the optimization over the common components, mapping functions $\{P_v\}_{v=1}^{m}$ can also be estimated by Algorithm 1.

**C. Kernel Multi-view Common Component Discriminant Analysis (KMvCCDA)**

KMvCCDA first maps observed data to a feature space and then implements MvCCDA in that space. Suppose that $X_v = [x^1_v, x^2_v, \ldots, x^m_v]$ denotes the data matrix of the $v$-th view and there exists a kernel mapping $\phi: x^i_v \rightarrow \phi(x^i_v)$ such that $\kappa(x^i_v, x^j_v) = \langle \phi(x^i_v), \phi(x^j_v) \rangle$, where $\kappa(x, z)$ is a kernel function with respect to $\phi$. $\phi(X_v) = [\phi(x^1_v), \phi(x^2_v), \ldots, \phi(x^m_v)]$ is the data matrix of the $v$-th view in the feature space and
its kernel matrix can be defined as $K_v = \phi^T(X_v)\phi(X_v)$, $K_v \in \mathbb{R}^{m \times m}$. Furthermore, we denote the view-specific mapping function $P_v$ in the feature space as $\phi(P_v)$. Assuming atoms of $P_v$ lie in the space spanned by the input data [42], [46], we have

$$\phi(P_v) = \phi(X_v) A_v,$$

(32)

where $A_v \in \mathbb{R}^{m \times d}$ denotes the atom matrix of the $v$-th view. The kernel extension of Eq. (21) can be obtained through

$$\|z^T - P_v^T x_i^T\|^2_H = \langle z^T - \phi^T(P_v) \phi(x_i), z^T - \phi^T(P_v) \phi(x_i) \rangle$$

$$= (z^T)^T z^T - 2\phi^T(x_i) \phi(X_v) A_v z^T$$

$$+ \phi^T(x_i) \phi(X_v) A_v A_v^T \phi^T(X_v) \phi(x_i),$$

(33)

and

$$\|P_v\|^2_H = \langle \phi(P_v), \phi(P_v) \rangle = tr(A_v^T K_v A_v).$$

(34)

KMvCCDA can be obtained with (33) and (34). Obviously, it can be optimized using the same technology as MvCCDA.

D. Complexity Analysis

Assume that $d_v = D$, ($v = 1, 2, \ldots, n$) for convenience, we now focus on the discussion of the computational complexity of MvCCDA in the $t$-th iteration. We first analyze the computational complexity of subproblem I and subproblem II in one iteration. From Eq. (25), the computational complexity of $Q$ is close to $O(n d D)$. Then from Eq. (26), the computational complexity of $\{z^T n_{i=1}\}$ is approximately $O(m n d D + m^2 D)$. Similarly, the computational complexity of $G$ in Eq. (30) is about $O(n d D)$. Therefore, from Eq. (31), the computational complexity of $\{P_v\}_t$ is approximately $O(m n D^2 + m n D^2 + n d D^2 + n D^3)$. Assume that $T_1$ and $T_2$ are the iterations of subproblem I and subproblem II in the $t$-th iteration of MvCCDA. To summarize, the computational complexity of MvCCDA in the $t$-th iteration is close to $O((m n d D + m^2 D) T_1 + (m n D^2 + m n D^2 + n D^2) T_2)$. In practice, we have $m > D$ and $D > d$, thus the computational complexity of MvCCDA in the $t$-th iteration can be further simplified with $O((m n d D + m^2 D) T_1 + m n D^2 T_2)$.

IV. EXPERIMENTS

We evaluate the performance of our (K)MvCCDA on four benchmark datasets: the MNIST and the USPS handwritten digit databases, the Heterogeneous Face Biometrics (HFB) database, the CMU Pose, Illumination, and Expression (PIE) database (CMU PIE) and the Columbia University Image Library (COIL-100) [3]. We organize this section as follows. Sect. IV-A details the experimental setting. We demonstrate the superiority of (K)MvCCDA on two-view data including handwritten digit datasets in Sect. IV-B and HFB in Sect. IV-C. Following this, we also verify the efficacy of (K)MvCCDA on multi-view data including CMU PIE in Sect. IV-D and COIL-100 in Sect. IV-E. Finally, Sect. IV-F presents the sensitivity analysis and convergence testing.

1http://www.cs.columbia.edu/CAVE/software/softlib/coil-100.php

A. Experimental Setting

In our experiments, five baselines and four state-of-the-art MvSL based approaches are selected for comparison. Baselines selected for comparison include PCA [47], LDA [48], CCA [26], KCCA [34] and MCCA [29], where the first two are classical methods for single-view scenario, whereas the other three are the best-known unsupervised MvSL based methods. State-of-the-art approaches include GMA [6], MvDA [31], MvMDA [32] and MvDA-VC [1], where GMA only considers intra-view discriminability, whereas the other three jointly take into account intra-view and inter-view discriminability. For a fair comparison, each experiment in this paper is repeated 10 times by randomly dividing data into training set and test set, and average of 10 results serves as classification accuracy. Similar to [1], PCA is first used to reduce dimensionality for all methods and PCA dimensions are empirically set to achieve the best performance via traversing all possible values. For KCCA and KMvCCDA, the polynomial kernel is selected. Hyper-parameters and kernel parameters are determined by validation set in handwritten digit datasets or 5-fold cross validation in the other three datasets. During the test phase, newly coming sample is first projected into a common subspace, and then its projected sample is classified via 1-NN classifier [3]. [32], [49] in Z. All experiments are conducted on Matlab 2013a, with CPU i7-4790 and 8.0GB memory size.

B. The Superiority of (K)MvCCDA on Two-view Data

We evaluate the performance of the proposed (K)MvCCDA on MNIST and USPS handwritten digit datasets to demonstrate the superiority of our methods for two-view scenario. The MNIST contains 70,000 normalized grayscale images of size $32 \times 32$ and USPS contains 9698 images of size $16 \times 16$. In our experiments, 1000 images (100 for each digit) from MNIST and 1000 images (100 for each digit) from USPS are selected to construct two-view data set. We use this set to evaluate handwritten digit classification between different handwritten styles. In the following experiments, 50 images of each digit are selected for training, 25 images of each digit compose validation set and the remaining images are used for testing.

1) Component-wise contributions of MvCCDA: In the first experiment, we compare our MvCCDA with its diverse baseline variants to verify the component-wise contributions to the performance improvement. Table I summarized the performance difference among MvCCDA and its three baseline variants. As can be seen, model that considers neither discriminant regularization nor local consistency regularization ranks the lowest, whereas models that consider one regularization achieve a remarkable gain. Our MvCCDA performs the best due to the joint consideration of discriminability and nonlinearity. The results indicate that the improvement of discriminability and the discovery of local geometric structure are essentially important for the task of cross-view classification.

2) Superiority of (K)MvCCDA: In the second experiment, we compare the proposed (K)MvCCDA with all competing methods except for MCCA. Its performance is omitted, be-
TABLE I: The average classification accuracy (%) of our MvCCDA (the last row) and its degraded baseline variants on handwritten digit datasets. The best two results are marked with red and blue.

|                      | discriminant regularization | local consistency regularization | MNIST-USPS | USPS-MNIST | Average |
|----------------------|----------------------------|----------------------------------|------------|------------|---------|
|                      | ✓                          | ✓                                | 67.0±4.1   | 58.5±5.8   | 62.3±3.6|
|                      | ✓                          | ✓                                | 75.3±3.1   | 59.0±5.7   | 67.1±4.0|
|                      | ✓                          | ✓                                | 79.8±2.3   | 76.8±3.2   | 78.3±1.6|
|                      | ✓                          | ✓                                | 80.4±2.3   | 80.6±2.3   | 80.5±1.3|

TABLE II: The average classification accuracy (%) on handwritten digit datasets. The best two results are marked with red and blue.

|                      | Gallery-Probe | PCA     | LDA     | CCA     | KCCA    | GMA     | MvDA    | MvMDA   | MvDA-VC | MvCCDA  | KMvCCDA |
|----------------------|---------------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|
|                      |               | 18.2±2.1| 34.4±3.1| 69.6±2.0| 78.8±5.2| 77.9±2.7| 68.8±3.6| 57.8±4.1| 79.6±2.0| 80.4±2.3| 88.1±2.2|
| MNIST-USPS           |               | 26.3±2.6| 35.0±2.8| 65.6±3.9| 72.7±3.4| 75.3±2.0| 68.8±3.1| 69.5±2.5| 79.9±2.3| 80.6±2.3| 92.1±1.0|

Fig. 2: Illustration of 2D embedding on handwritten digit datasets. The embedding from PCA, LDA, CCA, MvDA, MvMDA, MvDA-VC, the proposed method MvCCDA and its kernel extension KMvCCDA. Different colors denote different classes, and different markers stand for different views.

TABLE III: The average classification accuracy (%) on HFB. The best two results are marked with red and blue.

|                      | Gallery-Probe | NIR-VIR | VIS-NIR | PCA     | LDA     | CCA     | KCCA    | GMA     | MvDA    | MvMDA   | MvDA-VC | MvCCDA  | KMvCCDA |
|----------------------|---------------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|
|                      |               | 18.6±6.7| 19.1±4.7| 8.1±2.7 | 13.5±3.3| 22.5±4.7| 22.4±5.0| 31.6±6.3| 25.1±5.6| 32.9±4.4| 36.5±5.2| 44.8±4.7| 42.8±5.4|
| NIR-VIR              |               | 22.5±4.7| 27.5±4.1| 22.4±5.0| 27.5±4.1| 31.6±6.3| 25.1±5.6| 25.1±5.6| 28.0±6.0| 16.7±4.8| 37.0±4.4| 43.4±5.1| 45.8±6.5|

As can be seen, single-view methods fail to provide reasonable accuracy as expected. CCA performs poorly among MvSL based methods, whereas KCCA achieves a large improvement compared with CCA due to its kernel extension. Moreover, GMA also performs better than CCA, which we argue can be attributed to the consideration of intra-view discriminant information. Although MvDA and MvMDA further consider inter-view discriminant information, their performance is unexpectedly worse than GMA. One possible reason is that the view discrepancy can be more effectively removed by performing CCA on the data set. As expected, our MvCCDA outperforms MvDA-VC due to effectively incorporating view correlation, discriminability and nonlinearity in a joint manner. KMvCCDA further achieves a significant gain with the famed kernel trick, which proves the necessity of kernel extension. Fig. 2 displays the embeddings generated by (K)MvCCDA and their competitive methods. As shown in this figure, the embedding generated by (K)MvCCDA is more discriminative and it is obvious that Fig. 2 corroborates the results shown in Table III.

C. The Efficacy of (K)MvCCDA on HFB

We compare (K)MvCCDA with MvSL based counterparts on HFB to further demonstrate the efficacy of our methods on two-view data. The HFB [50], [51] contains 100 subjects. Each subject has four visual (VIS) light images and four near-infrared (NIR) images. In experiments, we convert colorful images to gray scale ones and resize them to 32 × 32 pixels. We use HFB to evaluate visual (VIS) light images versus near-infrared (NIR) images heterogeneous face classification.
TABLE IV: The average classification accuracy (%) of eight cases in terms of mean accuracy (mACC) on CMU PIE. The best two results are marked with red and blue.

| Methods     | Case 1 | Case 2 | Case 3 | Case 4 | Case 5 | Case 6 | Case 7 | Case 8 |
|-------------|--------|--------|--------|--------|--------|--------|--------|--------|
| PCA         | 42.3±10.6 | 33.6±7.1 | 23.5±5.8 | 40.2±5.5 | 28.3±5.0 | 21.3±4.5 | 21.3±4.5 | 9.7±2.6 |
| LDA         | 19.5±3.0  | 12.0±4.0 | 10.5±5.5 | 21.2±2.5 | 11.8±3.1 | 9.7±2.6   | 20.9±2.4 | 18.1±2.2 |
| MCCCA       | 54.2±7.4  | 44.5±9.0 | 28.2±4.6 | 57.4±3.7 | 39.2±5.3 | 32.6±4.2 | 48.4±3.4 | 39.4±1.8 |
| GMA         | 58.3±8.2  | 53.0±9.0 | 34.0±6.4 | 60.3±6.9 | 43.2±4.0 | 37.0±3.8 | 55.0±3.7 | 47.4±4.4 |
| MvDA        | 64.9±7.2  | 60.1±5.6 | 42.3±4.8 | 70.8±5.5 | 58.1±5.2 | 51.0±3.4 | 66.8±4.2 | 63.0±4.4 |
| MvMDA       | 68.2±9.0  | 52.3±6.4 | 48.1±5.2 | 69.2±6.1 | 54.6±3.4 | 48.7±4.0 | 60.3±2.7 | 49.0±2.2 |
| MvDA-VC     | 70.4±10.9 | 62.4±4.9 | 48.4±4.5 | 78.0±4.0 | 65.4±3.4 | 53.2±2.6 | 76.2±3.7 | 69.1±2.1 |
| MvCCDA      | 77.6±6.5  | 70.7±6.6 | 58.6±3.4 | 81.9±4.0 | 70.4±3.4 | 62.6±4.4 | 79.3±1.9 | 72.9±2.1 |
| KMvCCDA     | 81.1±7.9  | 72.5±5.6 | 59.0±4.1 | 83.6±4.4 | 70.9±2.6 | 61.2±4.4 | 80.2±2.2 | 73.2±3.3 |

We repeat the similar experimental setting in Sect. IV-B and summarize the experimental results in Table III when 60 subjects are selected as training data and the rest are used for testing. As expected, our MvCCDA and KMvCCDA achieve the best two results on HFB. It indicates that the superiority of (K)MvCCDA also holds on HFB. One should note that MvMDA performs unexpectedly badly on HFB. One possible reason is that there is a large class-center discrepancy between views on HFB, whereas MvMDA fails to remove it due to the limitation of within-class scatter.

**D. The Efficacy of (K)MvCCDA on Multi-view Data**

In this section, we compare (K)MvCCDA with all the other involved competitors on CMU PIE to demonstrate the efficacy of our methods for multi-view scenario. One may note that the performance of CCA and KCCA is omitted due to their limitation on multi-view data. The CMU PIE [52] contains 41,368 images of 68 people with 13 different poses, 43 diverse illumination conditions, and four various expressions. Seven poses, i.e., C14, C11, C29, C27, C05, C37 and C02, are selected to construct multi-view data.

1) **Superiority of (K)MvCCDA**: Experimental results in 8 cases are listed in Table IV when 45 people are selected for training and the rest are used for testing. As can be seen, MCCA performs the worst among MvSL based approaches due to the neglect of discriminant information. GMA outperforms MCCA by improving the intra-view discriminability. Benefited from the consideration of inter-view discriminant information, MvDA, MvMDA and MvDA-VC achieve a remarkable improvement. The proposed (K)MvCCDA achieve the best two results in all cases by further taking into account nonlinearity. Fig. 4 displays the embeddings generated by (K)MvCCDA and all the other competitive methods in case 5. It is obvious that embeddings corroborate the classification performance shown in Table IV.

**TABLE V**: The training time (seconds) of seven MvSL based methods in case 5 of CMU PIE dataset. **Bold** denotes the best result.

| Methods     | Case 1 | Case 2 | Case 3 | Case 4 | Case 5 | Case 6 | Case 7 | Case 8 |
|-------------|--------|--------|--------|--------|--------|--------|--------|--------|
| GMA         | 14     | 27     | 64     | 117    |
| MvDA        | 12     | 22     | 52     | 92     |
| MvMDA       | 14     | 26     | 57     | 101    |
| MvDA-VC     | 12     | 23     | 51     | 92     |
| MvCCDA      | 135    | 165    | 233    | 348    |
| KMvCCDA     | 86     | 125    | 228    | 344    |

2) **Robustness of (K)MvCCDA**: In this section, we evaluate the robustness of (K)MvCCDA and all the other competitors with respect to PCA dimensions and training set size. Robustness experiments are conducted in case 5 of CMU PIE, where PCA dimensions are changed from 25 to 525 at an interval of 25 and training set size are traversed from 20 to 45 at an interval of 5. 45 people are used for training and remaining people serve as test set. We plot the changing trends of accuracy in Fig. 5. It is worth noting that singular value decomposition of MvDA does not converge when PCA dimension exceeds 425, leading to the lack of performance. As shown in Fig. 5(a), the performance of (K)MvCCDA is more insensitive to PCA dimensions compared with GMA, MvDA and MvDA-VC. As shown in Fig. 5(b), our methods consistently outperform their competitors with the increment of training data size. Moreover, we also report the training time of case 5 in Table V. As can be seen, our methods present inferiority in computational complexity. However, considering the overwhelming performance gain, the extra cost in training time is still acceptable.
In experiments, 70 objects are selected for training and the remaining are used for testing. We report results of 4 cases including case 1: \{V1, V2\}, case 2: \{V1, V2, V3\}, case 3: \{V1, V2, V3, V4\} and case 4: \{V1, V2, V3, V4, V5\}. Same as experiments on CMU PIE, we report classification accuracy in terms of mean accuracy (mACC).

In experiments, 70 objects are selected for training and the remaining are used for testing. We report results of 4 cases in Fig. 6. As expected, our (K)MvCCDA achieve satisfactory results in all cases. To evaluate the performance of different methods with respect to different PCA dimensions and training set size, PCA dimensions are changed from 25 to 575 at an interval of 25 and the training set size is traversed from 30 to 70 at an interval of 5. We plot the changing trends of classification accuracy in case 2 in Fig. 8. As can be seen, the general trends shown in Fig. 8 are basically consistent with those in Fig. 5. The results on COIL-100 further demonstrate the superiority of our methods for multi-view scenario.

**E. The Superiority of (K)MvCCDA on COIL-100**

In this section, we further demonstrate the superiority of (K)MvCCDA on multi-view data. The COIL-100 contains 100 objects. Each object rotated from 0° to 355° at an interval of 5° is captured, leading to 72 images. Five poses, i.e., V1: [0°, 15°], V2: [40°, 55°], V3: [105°, 120°], V4: [160°, 175°] and V5: [215°, 230°], are selected to construct multi-view data (see Fig. 6 for exemplar objects). Images used in experiments are grayscale and resized to 48×48 pixels. Similar to [4], [5], experiments on COIL-100 are conducted in four cases including case 1: \{V1, V2\}, case 2: \{V1, V2, V3\}, case 3: \{V1, V2, V3, V4\} and case 4: \{V1, V2, V3, V4, V5\}. Same as experiments on CMU PIE, we report classification accuracy in terms of mean accuracy (mACC).

In experiments, 70 objects are selected for training and the remaining are used for testing. We report results of 4 cases in Fig. 7. As expected, our (K)MvCCDA achieve satisfactory results in all cases. To evaluate the performance of different methods with respect to different PCA dimensions and training set size, PCA dimensions are changed from 25 to 575 at an interval of 25 and the training set size is traversed from 30 to 70 at an interval of 5. We plot the changing trends of classification accuracy in case 2 in Fig. 8. As can be seen, the general trends shown in Fig. 8 are basically consistent with those in Fig. 5. The results on COIL-100 further demonstrate the superiority of our methods for multi-view scenario.

**F. Sensitivity Analysis and Convergence Testing**

In this section, we evaluate the performance of the proposed (K)MvCCDA with different hyper-parameter values. This ex-

---

Fig. 4: Illustration of 2D embedding in case 5 of CMU PIE dataset. The embedding from PCA, LDA, MCCA, GMA, MvDA, MvMDA, MvDA-VC, the proposed method MvCCDA and its kernel extension KMvCCDA. Different colors denote different classes, and different markers stand for different views. To make the figure clear, only the first 8 classes are shown.

```
Fig. 5: Experimental results in terms of mean accuracy (mACC) in case 5 of CMU PIE dataset: (a) shows the classification accuracy of all competing methods with different dimensions; (b) shows the classification accuracy for all competing methods with respect to different training size.
```

Fig. 6: Exemplar objects from COIL-100. Five poses, i.e., V1, V2, V3, V4 and V5, are selected to construct multi-view data.
Fig. 7: Object classification accuracy of nine algorithms in terms of mean accuracy (mACC) on COIL-100.

Fig. 8: Experimental results in terms of mean accuracy (mACC) in case 2 of COIL-100 dataset: (a) shows the classification accuracy of all competing methods with different dimensions; (b) shows the classification accuracy for all competing methods with respect to different training size.

Experiments are conducted in case 5 of CMU PIE and we repeat similar experimental setting in Sect. IV-D. Hyper-parameters are first divided into two groups: (1) $\{\lambda_1, \lambda_2\}$ and (2) $\lambda_3$. When we test the effect of one group, the other groups are set to default values, i.e., $\{\lambda_1, \lambda_2\} = \{0.6, 0.002\}$, $\lambda_3 = 0.0005$ for MvCCDA and $\{\lambda_1, \lambda_2\} = \{0.02, 0.002\}$, $\lambda_3 = 0.01$ for KMvCCDA. The classification accuracy of (K)MvCCDA are shown in Fig. 9.

As can be seen, we observe that the performance of (K)MvCCDA is insensitive to hyper-parameters within a reasonable range. Specifically, Fig. 9(a) and Fig. 9(b) indicate that MvCCDA can achieve an satisfactory result when $\lambda_1 \in [0.05, 0.5]$ and $\lambda_2 \in [5 \times 10^{-3}, 10^{-2}]$ or $\lambda_1 \in [0.5, 1]$ and $\lambda_2 \in [5 \times 10^{-3}, 10^{-2}]$, whereas KMvCCDA achieve a high accuracy when $\lambda_1 \in [5 \times 10^{-3}, 10^{-2}]$ and $\lambda_2 \in [10^{-4}, 10^{-3}]$ or $\lambda_1 \in [0.05, 0.1]$ and $\lambda_2 \in [5 \times 10^{-3}, 5 \times 10^{-2}]$. However, the performance of (K)MvCCDA deteriorates dramatically when $\lambda_1$ and $\lambda_2$ are out of those ranges. If we further compare MvCCDA with KMvCCDA, it is interesting to find that KMvCCDA is more robust to $\lambda_3$, whereas MvCCDA suffers from a sharp performance drop when $\lambda_3$ exceeds $10^{-2}$. We suggest using cross validation to determined the appropriate value of $\lambda_3$.

We finally evaluate the algorithm convergence. Experiments are also conducted in case 5 of CMU PIE, and we report the results in Fig. 10. As shown in this figure, we can find that reconstruction error monotonically decreases with the increment of iterations, and the classification accuracy achieves its maximum after a certain number of iterations. Trends of curves indicate that the optimization technology shown in Sect. III-B is effective and can lead to a fast convergence.

V. CONCLUSION

In this paper, we present a novel MvCCDA algorithm for cross-view classification. Extensive experiments conducted on four benchmark datasets demonstrate that our methods hold superiority in both two-view and multi-view scenarios. Moreover, our methods can efficiently discover the non-linear structure hidden in multi-view data, thus significantly improving cross-view classification accuracy. In future, we are interested in developing a new version of MvCCDA that can handle the scenario when multi-view data is unpaired or incomplete.
Fig. 9: Classification accuracy of our methods with different parameter values in case 5 of CMU PIE dataset: (a) and (b) show classification accuracy with different $\lambda_1, \lambda_2$ for MvCCDA and KMvCCDA respectively, whereas (c) show classification accuracy with different $\lambda_3$ for MvCCDA and KMvCCDA.

Fig. 10: Classification accuracy and reconstruction error with the increase of iterations: (a) and (b) shows the convergence curve (i.e., blue curve) and classification accuracy curve (i.e., red curve) with different iterations for MvCCDA and KMvCCDA respectively.

ACKNOWLEDGMENT

This work was supported partially by National Key Technology Research and Development Program of the Ministry of Science and Technology of China (No. 2015BAK36B00), in part by the Key Science and Technology of Shen zhen (No. CXZZ20150814155434903), in part by the Key Program for International S&T Cooperation Projects of China (No. 2016YFE0121200), in part by the National Natural Science Foundation of China (No. 61571205), in part by the Key Science and Technology of China (No. 2015BAK36B00), in part by the National Key Technology Research and Development Program of the Ministry of Science and Technology of China (No. 61571205), in part by the National Natural Science Foundation of China (No. 61772220).

REFERENCES

[1] M. Kan, S. Shan, H. Zhang, S. Lao, and X. Chen, “Multi-view discriminant analysis,” IEEE transactions on pattern analysis and machine intelligence, vol. 38, no. 1, pp. 188–194, 2016.
[2] Y. Zhao, X. You, S. Yu, C. Xu, W. Yuan, X.-Y. Jing, T. Zhang, and D. Tao, “Multi-view manifold learning with locality alignment,” Pattern Recognition, vol. 78, pp. 154–166, 2018.
[3] A. Sharma, A. Kumar, H. Duame, and D. W. Jacobs, “Generalized multiview analysis: A discriminative latent space,” in Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on. IEEE, 2012, pp. 2160–2167.
[4] Z. Ding and Y. Fu, “Low-rank common subspace for multi-view learning,” in Data Mining (ICDM), 2014 IEEE International Conference on. IEEE, 2014, pp. 110–119.
[5] P. Moutafis, M. Leng, and I. A. Kakadiaris, “An overview and empirical comparison of distance metric learning methods,” IEEE transactions on cybernetics, vol. 47, no. 3, pp. 612–625, 2017.
[6] J. Goldberger, G. E. Hinton, S. T. Roweis, and R. R. Salakhutdinov, “Neighbourhood components analysis,” in Advances in Neural Information Processing Systems 17. MIT Press, 2005, pp. 513–520.
[7] D. Xu, S. Yan, D. Tao, S. Lin, and H.-J. Zhang, “Marginal Fisher analysis and its variants for human gait recognition and content-based image retrieval,” IEEE transactions on image processing, vol. 16, no. 11, pp. 2811–2821, 2007.
[8] N. Quadrianto and C. H. Lampert, “Learning multi-view neighborhood preserving projections,” in Proceedings of the 28th international conference on machine learning (ICML-11), 2011, pp. 425–432.
[9] A. Mignon and F. Jurie, “Cmmil: a new metric learning approach for cross-modal matching,” in Asian Conference on Computer Vision, 2012, pp. 14–pages.
[10] A. Bellet, A. Habrard, and M. Sebban, “A survey on metric learning for feature vectors and structured data,” arXiv preprint arXiv:1306.6709, 2013.
[11] P. Moutafis and I. A. Kakadiaris, “Semi-coupled basis and distance metric learning for cross-domain matching: Application to low-resolution face recognition,” in Biometrics (ICB), 2014 IEEE International Joint Conference on. IEEE, 2014, pp. 1–8.
[12] K. Q. Weinberger and L. K. Saul, “Distance metric learning for large
