Multi-View Matrix Decomposition: A New Scheme for Exploring Discriminative Information

Cheng Deng[1] **Zongting Lv**[1] **Wei Liu**[1,2*] **Junzhou Huang**[3] **Dacheng Tao**[4] **Xinbo Gao**[1]
[1]School of Electronic Engineering, Xidian University, Xi'an 710071, China
[2]IBM T. J. Watson Research Center, Yorktown Heights, NY 10598, USA
[3]University of Texas at Arlington, TX 76019, USA
[4]Faculty of Engineering and Information Technology,
University of Technology, Sydney, NSW 2007, Australia

Abstract

Recent studies have demonstrated the advantages of fusing information from multiple views for various machine learning applications. However, most existing approaches assumed the shared component common to all views and ignored the private components of individual views, which thereby restricts the learning performance. In this paper, we propose a new multi-view, low-rank, and sparse matrix decomposition scheme to seamlessly integrate diverse yet complementary information stemming from multiple views. Unlike previous approaches, our approach decomposes an input data matrix concatenated from multiple views as the sum of low-rank, sparse, and noisy parts. Then a unified optimization framework is established, where the low-rankness and group-structured sparsity constraints are imposed to simultaneously capture the shared and private components in both instance and view levels. A proven optimization algorithm is developed to solve the optimization, yielding the learned augmented representation which is used as features for classification tasks. Extensive experiments conducted on six benchmark image datasets show that our approach enjoys superior performance over the state-of-the-art approaches.

1 Introduction

Many real-world datasets have representations in the form of multiple views, which are collected from different sources or obtained from various feature extractors. For example, in the biological data domain, each human gene is measured by different techniques, such as gene expression, single-nucleotide polymorphism (SNP), and methylation [Cai *et al.*, 2013b]; in the visual data domain, each image/video can be represented by different visual features, such as colour descriptor, local shape descriptor, and spatio-temporal descriptor. These *views* often provide diverse and complementary information to each other. Combining these information introduced by individual views has recently become very popular, and is expected to enhance the overall performance of a learning task at hand.

In the literature, several fusion or combination approaches have been proposed from different perspectives. A naive method is to concatenate the vectors of all views into a new vector, or more generally, to use weights to concatenate them. Such a method is problematic, because it ignores the particular statistical property belonging to an individual view. Kernel-based approaches associate different kernels with different views and combine them either linearly or nonlinearly, among which kernel averaging is a simple and representative method [Bucak *et al.*, 2014]. It is worth noting that such approaches are particularly effective under the assumption that all views are independent of each other.

In contrast, subspace learning methods aim to learn a latent subspace which captures the relevant information shared by all views, and are effective when the views are assumed to be dependent on each other. Canonical Correlation Analysis (CCA) and its variants [Kuss and Graepel, 2003] learn the latent representations shared by all views such that the correlations among the views are maximized. As a version of nonlinear CCA, the shared Gaussian Process Latent Variable Model (sGPLVM) [Shon *et al.*, 2005] builds a common latent space for inferring another view from the observation view. Recently, more methods have been extended to multi-view scenarios. For example, the work of [Quadrianto and Lampert, 2011] constructs embedding projections from multi-view data to produce a shared subspace.

An appealing scheme based on sparse representation has attracted a lot of research interests, which aims to encourage the complementarity by imposing different sparsity constraints on a shared subspace derived from multiple views [Shekhar *et al.*, 2014; Wang *et al.*, 2015]. The used sparsity constraints include ℓ_0-norm [Cai *et al.*, 2013b], tree-structured sparsity norm [Bahrampour *et al.*, 2014], and trace-norm [Liu *et al.*, 2015]. Another popular scheme follows the standard pipeline of low-rank matrix recovery [Wright *et al.*, 2009], in which the low-rankness constraint is employed to discover the underlying subspace structures in multi-view data, while the sparsity constraint is used for outlier removal. Cheng *et al.* [Cheng *et al.*, 2011] imposes $\ell_{2,1}$-norm on a concatenated low-rank matrix to associate multiple views. The work of [Guo *et al.*, 2013] captures the dependencies across multiple views via a shared low-rank coefficient matrix. The work of [Xia *et al.*, 2014] learns a shared low-rank transition probability matrix of all views for

[*]The corresponding author, wliu.cu@gmail.com.

spectral clustering. Although the aforementioned approaches enjoy satisfactory performance in their specific settings, all of them only focus on the shared component across multiple views while overlook the private component of an individual view. Therefore, the main difficulty of integrating multiple views is how to effectively capture the correlative properties across all views (*i.e.*, shared component), and at the same time exploit the discriminative properties of individual views (*i.e.*, private components).

To address the issues presented above, in this paper we propose a novel multi-view low-rank and sparse matrix decomposition method by robustly utilizing the group-structured prior. Specifically, unlike existing methods, we first concatenate input data from multiple views into a new mixed matrix, so the global structure and the hidden correlations across all the views can be preserved well. We then conduct the low-rank and sparse matrix decomposition in the noisy case, and leverage trace norm and group-structured sparsity norm to promote the low-rankness and sparsity properties of the target matrix, respectively. As such, the shared and private components can be simultaneously captured in both instance and view levels. Since our formulated optimization objective is non-smooth, we develop an algorithm based on Augmented Lagrange Multiplier (ALM) to solve the optimization efficiently. We evaluate our method on six widely used image datasets in classification tasks, and on each dataset we integrate six different types of popular visual features. The experimental results demonstrate that our method consistently outperforms the state-of-the-art classification approaches that use traditional multi-view combinations.

2 Robust Multi-View, Low-Rank, and Sparse Matrix Decomposition

In this section, we introduce the low-rank and sparse matrix decomposition method for integrating diverse and complementary information from multiple views. We first present the concatenation of multi-view input matrices and then describe our framework. Finally, we output an augmented representation by combining shared component and private component, and extend the proposed method to deal with supervised image classification tasks. Given a set of n data instances $\{\mathbf{x}_i\}_{i=1}^n$, the data matrix in the k-th view is $X^{(k)} = [\mathbf{x}_1^{(k)}, \cdots, \mathbf{x}_n^{(k)}] \in \mathbb{R}^{d_k \times n}$ $(k = 1, \cdots, m)$, where d_k denotes the feature dimension of the k-th view.

2.1 Multi-View Input Matrix Concatenation

Different from previous approaches that handle multi-view input matrices $X^{(k)}$ individually, we first concatenate the input data matrices $X^{(k)}$ of each view to construct a new mixed matrix $X = [X^{(1)}; \cdots; X^{(m)}] \in \mathbb{R}^{d \times n}$, with $d = \sum_{k=1}^m d_k$. Since different views describe different aspects of the same object, the views are intrinsically associated. Thus, the concatenated input matrix has a natural advantage of strengthening the correlation among multiple views, as well as providing a convenient way to explore the complementarity of multiple views both in instance level and view level, which will be detailed in the following subsection.

2.2 Problem Formulation

Given a single-view input data $X^{(k)}$, motivated by classical low-rank matrix recovery model [Wright *et al.*, 2009], previous approaches [Guo *et al.*, 2013; Xia *et al.*, 2014] utilize this model to remove outlier from the observation data by decomposing the single-view data matrix $X^{(k)}$ into two different parts: a shared low-rank matrix L and a separately sparse error matrix $E^{(k)}$. Under fairly general conditions, L can be exactly recovered from $X^{(k)}$ as long as $E^{(k)}$ is sufficiently sparse. Formally, this model can be formulated into:

$$\min_{L, E^{(k)}} ||L||_* + \gamma ||E^{(k)}||_1 \quad \text{s.t.} \quad X^{(k)} = X^{(k)} L + E^{(k)}, \tag{1}$$

where γ is a tradeoff parameter. As aforementioned, this model may not be suitable for coping with multi-view problems. The main reason is that this model oversimply decomposes all input matrices into a low-rank matrix shared by all views and an error matrix belonging to each view. In fact, the removed error matrix instead contains discriminative information specific to each individual view, which is useful to boost the performance of a given learning task.

Therefore, in this paper, we simultaneously capture the shared component and private component in a unified matrix decomposition framework. Specifically, we decompose the concatenated input matrix X into three parts, *i.e.* low-rank matrix L, sparse matrix S and noise matrix E, and then utilize three different regularizations to exploit the underlying relationship among multiple views so as to leverage the correlated information across all views as well as the discriminative information of each view. Hence, we consider the following matrix decomposition problem:

$$\min_{L,S,E} \lambda_1 ||L||_* + \lambda_2 \Omega(S) + \frac{1}{2}||E||_\mathrm{F}^2 \quad \text{s.t.} \quad X = L + S + E,\ L = XZ,\ S = BX, \tag{2}$$

where $L, S, E \in \mathbb{R}^{d \times n}$, $Z \in \mathbb{R}^{n \times n}$ encodes the dependencies among the data instances, $B \in \mathbb{R}^{d \times d}$ denotes a row transformation matrix that maps the input data into a view space, and non-negative parameters λ_1, λ_2 are used to balance the effects of the three parts. The trace norm $\|L\|_*$ is the convex envelope of the rank of L over the unit ball of the spectral norm, and minimizing the trace norm often induces the desirable low-rank structure in practice. $\Omega(S)$ is a regularizer that encourages group-structured sparsity of S. Due to the inherent sparse structures of the real-world data, $\Omega(S)$ can be defined as

$$\Omega(S) \triangleq \sum_{j=1}^{d} \sum_{i=1}^{c} ||S_j^{G_i}||_2. \tag{3}$$

Here, c is the number of groups, and $S_j^{G_i}$ is a row vector containing a subset of entries in the row S_j, that is, those specified by the indices in group G_i [Rakotomamonjy *et al.*, 2008].

Based on matrix norm inequality, Eq. (2) can be relaxed to:

$$\min_{Z,B,E} \lambda_1 ||Z||_* + \lambda_2 \Omega(B) + \frac{1}{2}||E||_\mathrm{F}^2 \quad \text{s.t.} \quad X = XZ + BX + E. \tag{4}$$

In Eq. (4), analogous to low-rank matrix recovery, the matrix Z is equal to the low-rank representation (*i.e.*, shared component) corresponding to the dictionary X. Intuitively, the underlying global structure of the original input matrices is consistent across all views.

Besides the shared component, we need to discover the private component S specific to each view. In Eq. (4), B is a row transformation matrix projecting the input data into view space, upon which the relevance among views can be discovered. Considering the matrix S being group-structured sparse, we instead enforce group-structured sparsity constraint on B by exploiting regularization relaxation, which encourages sharing within a group and discriminativeness among different groups. Meanwhile, sparsity is also forced between different rows so that the discriminative elements in each view are selected. To obtain the private component, the discriminative information in view space should be connected to the data space, mathematically, $S = BX$.

2.3 Augmented Representation

When the shared component Z and the private component S are learned from Eq. (4), we can derive an augmented multi-view representation by directly concatenating them.

Nevertheless, to make the augmented representation more robust and compact, we first learn the low-dimensional shared component $\tilde{Z} \in \mathbb{R}^{p \times n}$ for all views by enforcing Principal Component Analysis (PCA) on the learned low-rank representation Z. Then, we can obtain more compact private component $\tilde{S} \in \mathbb{R}^{q \times n}$ by filtering out all zero rows in S. The above two procedures still well preserve the inherent properties of Z and S. Therefore, the augmented multi-view representation is denoted as

$$R = [\tilde{Z}; \tilde{S}], \tag{5}$$

where $R \in \mathbb{R}^{(p+q) \times n}$ with $p + q \ll d$. Subsequently, the augmented representation is feasible to a variety of multi-view learning tasks, such as image classification, clustering, object detection and recognition, *etc*.

Here, we apply our method to supervised image classification. Suppose the data instances are labeled into c classes, $\mathbf{y}_i \in \mathbb{R}^c$ is the class label vector of the data instance $\mathbf{x}_i$. The class indicator matrix is represented as $\mathbf{Y} = [\mathbf{y}_1, \cdots, \mathbf{y}_n] \in \mathbb{R}^{c \times n}$, where $\mathbf{Y}_{j,i} = 1$ if data instance $\mathbf{x}_i$ belongs to the j-th class, $\mathbf{Y}_{j,i} = 0$ otherwise. In the training phase, the classifier $W \in \mathbb{R}^{(p+q) \times c}$ can be learned by the augmented representation R_{tr} from the training instances. In the test phase, given test data $\hat{X}$ and the row transformation matrix B, we can obtain the shared component $\hat{Z}$ according to Eq. (4). The private component is written as $\hat{S} = B\hat{X}$. Therefore, the augmented representation for test data is $R_{ts} = [\hat{Z}; \hat{S}]$, the class of an unseen data instance can be determined by $\arg\max_j (W^\top R_{ts} + \mathbf{b})_j$, where $\mathbf{b}$ is bias vector.

3 Optimization

3.1 Algorithm

The optimization problem (4) is challenging due to the simultaneous low-rank and group-structured sparsity regularizations in the objective function. To address this problem, two auxiliary variables are introduced to decouple Z and B [Afonso *et al.*, 2011]. Hence, we reformulate the problem as follows:

$$\min_{Z,B,J,K} \ \lambda_1 ||J||_* + \lambda_2 \Omega(K) + \frac{1}{2} \big\| X - XZ - BX \big\|_{\mathrm{F}}^2 \quad \text{s.t.} \quad Z = J,\ B = K. \tag{6}$$

Algorithm 1 The learning procedure of the proposed method

```
Input: A multi-view data matrix $X$.
Output: The learned matrices $Z, B$.
Initialize: $Z_0 = J_0 = 0, B_0 = K_0 = 0, Y_{1,0} = 0, Y_{2,0} = 0, \mu = \mu_1 = \mu_2 = 10^{-6}, \mu_{max} = 10^6, \rho = 1.1$, and $\epsilon = 10^{-8}$
while not converge do
    fix the others and update $Z$ according to Eq. (8);
    fix the others and update $B$ according to Eq. (9);
    fix the others and update $J$ according to Eq. (10);
    fix the others and update $K$ according to Eq. (11);
    update the multipliers:
        $Y_1 = Y_1 + \mu_1(Z - J)$,
        $Y_2 = Y_2 + \mu_2(B - K)$;
    update the parameter:
        $\mu = \min(\mu_{max}, \rho\mu)$;
    check the convergence condition:
        $\|X - XZ - BX\|_\infty \le \epsilon$, $\|Z - J\|_\infty \le \epsilon$, and $\|B - K\|_\infty \le \epsilon$;
end while
```

The solution to problem (6) is given by ALM, which is implemented by minimizing the following augmented Lagrangian function

$$\begin{aligned} \min_{Z,B,J,K;Y_1,Y_2} \quad & \frac{1}{2} \big\| X - XZ - BX \big\|_{\mathrm{F}}^2 \\ & + \lambda_1 ||J||_* + \langle Y_1, (Z - J) \rangle + \frac{\mu_1}{2} \big\| Z - J \big\|_{\mathrm{F}}^2 \\ & + \lambda_2 \Omega(K) + \langle Y_2, (B - K) \rangle + \frac{\mu_2}{2} \big\| B - K \big\|_{\mathrm{F}}^2, \end{aligned} \tag{7}$$

where Y_1, Y_2 are the Lagrangian multipliers, $\langle X, Y \rangle = tr(X^\top Y)$ is the inner product between two matrices, and μ_1, μ_2 are two non-negative penalty parameters. Due to the separable structure of the objective function in Eq. (7), the optimization to problem (7) can be carried out on each variable separately with others fixed. The optimization procedure is described in Algorithm 1.

Now, we investigate an update rule for each variable among Z, B, J, K while fixing the other variables.

Update Z with the others fixed:

$$Z^* = (\mu_1 I + X^\top X)^{-1} \left(X^\top (X - BX) + \mu_1 J - Y_1 \right). \tag{8}$$

Update B with the others fixed:

$$B^* = \left((X - XZ) X^\top + \mu_2 K - Y_2 \right) (\mu_2 I + X X^\top)^{-1}. \tag{9}$$

Update J with the others fixed:

$$J^* = \arg\min_J \ \frac{\lambda_1}{\mu_1} ||J||_* + \frac{1}{2} \big\| J - Z - \frac{1}{\mu_1} Y_1 \big\|_{\mathrm{F}}^2. \tag{10}$$

Update K with the others fixed:

$$K^* = \arg\min_{K} \ \frac{\lambda_2}{\mu_2}\Omega(K) + \frac{1}{2}\big\|K - B - \frac{1}{\mu_2}Y_2\big\|_{\mathrm{F}}^2. \qquad (11)$$

Note that both Eqs. (8) and (9) give closed-form solutions to Z and B, so they achieve global minima in each iteration. Meanwhile, Eq. (10) can be solved via the singular value thresholding (SVT) operator [Cai *et al.*, 2010] which guarantees the rank of J to reduce until convergence. However, due to the group-structured sparsity regularization, Eq. (11) poses a non-smooth and non-trivial optimization problem. To simplify the optimization of Eq. (11), we seek the following alternative formulation by squaring the regularizer $\Omega(K)$:

$$\arg\min_{K} \ \frac{\lambda_2}{\mu_2}\left(\sum_{j=1}^{d}\sum_{i=1}^{c}\big\|K_j^{G_i}\big\|_2\right)^2 + \frac{1}{2}\big\|K - B - \frac{1}{\mu_2}Y_2\big\|_{\mathrm{F}}^2. \qquad (12)$$

An auxiliary variable $\tau_{j,i}$ is introduced to make Eq. (12) more tractable, that is,

$$\left(\sum_{j=1}^{d}\sum_{i=1}^{c}\big\|K_j^{G_i}\big\|_2\right)^2 \leq \sum_{j=1}^{d}\sum_{i=1}^{c}\frac{\big\|K_j^{G_i}\big\|_2^2}{\tau_{j,i}}, \qquad (13)$$

where $\sum_j\sum_i \tau_{j,i} = 1, \tau_{j,i} \geq 0, \forall j, i$. The condition under which the above inequality holds is

$$\tau_{j,i} = \frac{\big\|K_j^{G_i}\big\|_2}{\sum_{j=1}^{d}\sum_{i=1}^{c}\big\|K_j^{G_i}\big\|_2}. \qquad (14)$$

Obviously, when Eq. (14) is satisfied, the right-hand side of Eq. (13) takes a minimum, which can be regarded as a further relaxation of Eq. (12). Thus, Eq. (12) can be reformulated as

$$\begin{aligned}&\arg\min_{K} \ \frac{\lambda_2}{\mu_2}\sum_{j=1}^{d}\sum_{i=1}^{c}\frac{\big\|K_j^{G_i}\big\|_2^2}{\tau_{j,i}} + \frac{1}{2}\big\|K - B - \frac{1}{\mu_2}Y_2\big\|_{\mathrm{F}}^2 \\ &\quad \text{s.t.} \ \sum_{j}\sum_{i}\tau_{j,i} = 1, \ \tau_{j,i} \geq 0, \ \forall j, i. \qquad (15)\end{aligned}$$

Based on the above analysis, Eq. (15) can be solved by alternatively optimizing K and $\tau_{j,i}$ iteratively until convergence. Denote M_w as the w-th column of matrix M. In each iteration, we first fix the values for K and update $\tau_{j,i}$ according to Eq. (14). Then, we hold the values for $\tau_{j,i}$ as constant and optimize for K. To minimize K, we take the first order derivative of Eq. (15) with respect to K_w and set it to zero, obtaining

$$K_w = \big(2\frac{\lambda_2}{\mu_2}\Pi + I\big)^{-1}\big(B_w + \frac{Y_{2,w}}{\mu_2}\big), \qquad (16)$$

where $\Pi \in \mathbb{R}^{d\times d}$ is a diagonal matrix with $\sum_{i=1}^{c}\frac{1}{\tau_{j,i}}$ being the j-th element in its diagonal.

The following theorem guarantees the convergence of solving problem (11) in each iteration.

Theorem 1. *The objective of Eq.* (11) *with* $\lambda_2 \geq \frac{d\mu_2}{2c}$ *monotonically decreases in each iteration.*

Proof. The solution of Eq. (11) involves alternative optimization of K_w and $\tau_{j,i}$.

$$\begin{aligned}K_w &= \big(2\frac{\lambda_2}{\mu_2}\Pi + I\big)^{-1}\big(B_w + \frac{Y_{2,w}}{\mu_2}\big)\\ &\leq \big(2\frac{\lambda_2}{\mu_2}\Pi\big)^{-1}\big(B_w + \frac{Y_{2,w}}{\mu_2}\big). \qquad (17)\end{aligned}$$

$\big(2\frac{\lambda_2}{\mu_2}\Pi\big)^{-1} = \frac{\mu_2}{2\lambda_2}\Pi^{-1}$, where Π^{-1} is a diagonal matrix with the j-th entry on the diagonal being $\big(\sum_{i=1}^{c}\frac{1}{\tau_{j,i}}\big)^{-1} \leq \frac{1}{c}$. Let $\lambda_2 \geq \frac{d\mu_2}{2c}$, then

$$\left\|\big(2\frac{\lambda_2}{\mu_2}\Pi\big)^{-1}\right\|_{\mathrm{F}} = \big\|\frac{\mu_2}{2\lambda_2}\Pi^{-1}\big\|_{\mathrm{F}} \leq 1. \qquad (18)$$

Suppose that $\{Y_{2,w}\}$ is a cauchy sequence and μ_2 does not grow too fast. Then $\big\|\frac{Y_{2,w}}{\mu_2}\big\|_{\mathrm{F}}$ is bounded with $\big\|\frac{Y_{2,w}^*}{\mu_2}\big\|_{\mathrm{F}}$, where $Y_{2,w}^*$ is the value of $Y_{2,w}$ after the algorithm halts. Therefore, we have

$$\|K_w\|_{\mathrm{F}} \leq \big\|B_w + \frac{Y_{2,w}}{\mu_2}\big\|_{\mathrm{F}}. \qquad (19)$$

Since $\frac{Y_{2,w}}{\mu_2}$ is bounded and B_w monotonically decreases, $\|K_w\|_{\mathrm{F}}$ also monotonically decreases in each iteration.

Without loss of generality, we can suppose $\sum_{j=1}^{d}\sum_{i=1}^{c}\big\|K_j^{G_i}\big\|_2 \geq 1$. According to Eq. (14),

$$|\tau_{j,i}| \leq \big\|K_j^{G_i}\big\|_2. \qquad (20)$$

Since $\|K_w\|_{\mathrm{F}}$ decreases in each iteration, $\big\|K_j^{G_i}\big\|_2$ also decreases. Given $0 \leq |\tau_{j,i}| \leq 1$, we know that the value of $\tau_{j,i}$ monotonically decreases. Since each iteration converges, the convergence of optimizing the objective in Eq. (11) is ensured. □

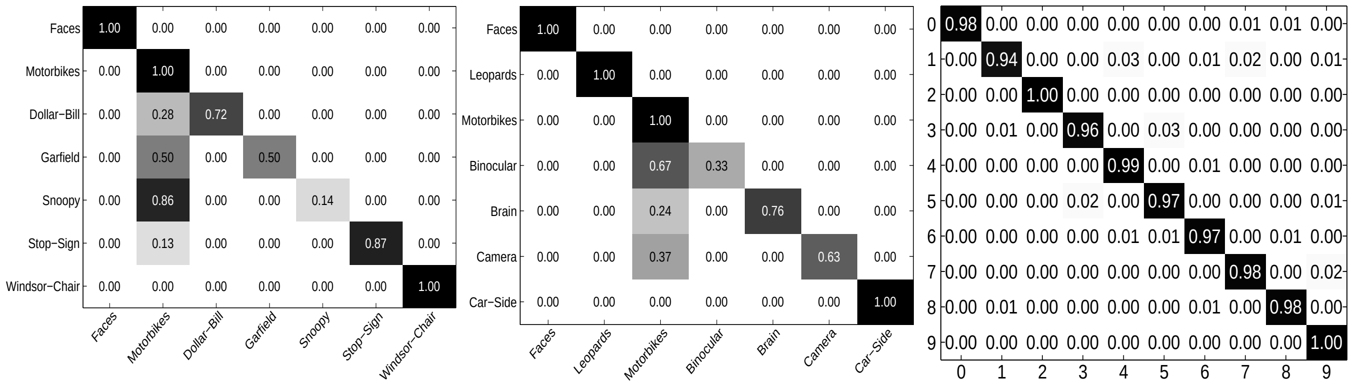
3.2 Complexity Analysis

The complexity of the proposed algorithm (see Algorithm 1) includes four parts, where each part accounts for the optimization with respect to a single variable among Z, B, J, K, respectively. Firstly, the update of Z consists of some matrix multiplication and inversion, leading to a complexity of $O(n^3)$. The update of B also has a similar complexity. The SVD is implemented when solving J, which usually induces a computational cost of $O(n^3)$. Suppose the number of inner iterations for updating K is t, then the corresponding time complexity is $O(2n^3t)$. The computational cost for updating Y_1, Y_2 and μ can be eliminated since they are relatively smaller than that of the others.

Apparently, the computational cost of Algorithm 1 is dominated by the calculation of J. For a maximal number of outer iteration T (step 2 to step 10), the complexity of Algorithm 1 can thus be approximated by $O(Tn^3)$.

4 Experiments

In this section, we evaluate the performance of our method on six benchmark datasets: Caltech101 [Fei-Fei *et al.*, 2007], NUS-WIDE [Chua *et al.*, 2009], Handwritten numeral [Frank



(a) Average accuracy 97%

(b) Average accuracy 83%

(c) Average accuracy 98%

Figure 1: The confusion matrix calculated by our proposed method: (a) Caltech7, (b) Caltech20, and (c) Handwritten numerals. For Caltech20, we only plot the confusion matrix of the top 7 classes for the convenience of displaying.

and Asuncion, 2010], Animal with attributes [Lampert *et al.*, 2009], Scene-15 [Lazebnik *et al.*, 2006].

Caltech101 dataset is an object recognition dataset containing 8677 images, belonging to 101 categories. Following [Cai *et al.*, 2013a], we choose widely used 7 and 20 classes to construct two datasets, *i.e.* **Caltech7** and **Caltech20**, in which there are 1474 images and 2386 images, respectively. In order to obtain the different views, we extract GIST [Oliva and Torralba, 2001] with dimension 512, CENTRIST [Wu and Rehg, 2008] with dimension 1302, LBP [Ojala *et al.*, 2002] with dimension 256, histogram of oriented gradient (HOG) with dimension 576, SIFT-SPM [Lazebnik *et al.*, 2006] with dimension 1000, color histogram (CH) with dimension 64.

NUS-WIDE dataset contains 30000 images and 31 classes. We use six published features to do multi-view classification, which contains 225 dimension block-wise color moments (CM), 64 dimension CH, 144 dimension color correlogram (CoRR), 128 dimension wavelet texture (WT), 73 dimension edge distribution (EDH), and 500 dimension SIFT BoW feature.

Handwritten numerals (HW) dataset consists of 2000 data instances for 0 to 9 ten digit classes. We use six published features to represent multiple views. Specifically, these features include 76 Fourier coefficients of the character shapes (FOU), 216 profile correlations (FAC), 64 Karhunenlove coefficients (KAR), 240 pixel averages in 2×3 windows (PIX), 47 Zernike moment (ZER), and 6 morphological features (MOR).

Animal with attributes (AWA) dataset contains 50 classes, 30475 instances. We use all the published features for all the images, that is, 2688 dimension color histogram (CQ), 2000 dimension local self-similarity (LSS), 252 dimension pyramidHOG (PHOG), 2000 dimension SIFT, 2000 dimension colorSIFT, and 2000 dimension SURF.

Scene-15 dataset consists of 15 classes, 3000 images, mainly stemming from the COREL collection, personal photographs and Google image search. We extract the same six visual features from each image as Caltech101 dataset.

4.1 Experimental Setup

To demonstrate the advantage of our proposed method, we compare the proposed method against several representative approaches: (1) **Single View (SV)**: Using the single-view feature to verify the classification performance, where Type 1 to Type 6 represent six different features belonging to the corresponding datasets, respectively. (2) **Direct Concatenation (DC-SVM)**: Concatenating features of all views in a straightforward way, and then performing classification by SVM directly on the concatenated feature. (3) **SimpleMKL**: Constructing a kernel for each view, and then learning a linear combination of the different kernels in SVM [Rakotomamonjy *et al.*, 2008]. (4) **SMML**: Integrating multiple views by imposing joint structured sparsity regularizations [Wang *et al.*, 2013]. (5) **I^{2}SCA**: CCA-based multi-view supervised feature learning [Jing *et al.*, 2014]. (6) **MT-SRC**: Tree-structured sparse model for multi-view classification [Bahrampour *et al.*, 2014]. (7) **lrMMC**: Low-rank multi-view matrix completion method [Liu *et al.*, 2015].

Besides, to further demonstrate the superior performance of the proposed method, we derive three variants of the proposed method. First, in Eq. (4), we enforce ℓ_1-norm to constrain the sparse matrix S. We denote it as "Our method (ℓ_1-norm)". Second, we use $\ell_{2,1}$-norm to constrain the sparse matrix S. We denote this degenerate version of the proposed method as "Our method ($\ell_{2,1}$-norm)". Finally, the full version of the proposed method by Eq. (4) is operated and named as "Our method".

To quantitatively measure the performance of the compared methods, we use accuracy to measure the classification performances. Since we mainly investigate the multi-view supervised classification problem, hence the labels of training samples are known. For each dataset, we randomly choose 30% of the data for training, and the rest for testing. All reported experimental results are averaged over 10 runs with random initializations.

In all the experiments, we implement standard 5-fold cross-validation and report the average results. Specifically, the parameters λ_1 and λ_2 in Eq. (4) are finely tuned by searching the grid of $\{10^{-3}, 10^{-2}, \cdots, 10^2, 10^3\}$, and then the best values are chosen based on validation performance. For SVM

Table 1: Classification accuracy comparison.

Methods	Datasets					
	Caltech7	Caltech20	NUS-WIDE	HW	AwA	Scene-15
Type1	0.8342	0.5378	0.1521	0.9204	0.0573	0.6648
Type2	0.8035	0.7126	0.1495	0.8277	0.0623	0.9053
Type3	0.6872	0.3346	0.1469	0.9328	0.0501	0.7020
Type4	0.8829	0.3146	0.1507	0.4673	0.0548	0.5986
Type5	0.8124	0.3275	0.1412	0.9457	0.0653	0.5633
Type6	0.7951	0.6093	0.1490	0.8293	0.0726	0.1193
DC-SVM	0.8549	0.5873	0.1331	0.9600	0.0747	0.8378
simpleMKL	0.8912	0.6787	0.1428	0.9623	0.0788	0.8111
SMML	0.8792	0.6802	0.1451	0.9610	0.0812	0.8220
I^{2}SCA	0.8741	0.6011	0.1583	0.9650	0.0798	0.8837
MTSRC	0.8938	0.6177	0.1537	0.9640	0.0787	0.8522
lrMMC	0.9012	0.6937	0.1598	0.9680	0.0792	0.8234
Our method (ℓ_1-norm)	0.9098	0.7041	0.1607	0.9690	0.0812	0.8356
Our method ($\ell_{2,1}$-norm)	0.9115	0.7103	0.1621	0.9703	0.0810	0.8325
Our method	**0.9674**	**0.8264**	**0.1645**	**0.9764**	**0.0823**	**0.9800**

classifier, we use Gaussian kernel as the kernel matrix for each method, which is defined as $\mathcal{K}(x_i, x_j) = exp(-\sigma||x_i - x_j||^2)$. The standard deviation σ is tuned in the same range used as our method. The tradeoff parameter C of SVM is selected from the range $\{0.01, 0.1, 1, 10, 100, 1000\}$. We use LIBSVM[1] software package to implement SVM in all our experiments.

4.2 Classification Results

The classification results of the compared methods on all considered datasets are reported in Table 1. The confusion matrices of Caltech7, Caltech20 and Handwritten numerals are shown in Figure 1. First of all, in order to verify multi-view combination power of our method, we compare classification performances between using all the views and using only one view. It is obvious that single-view feature may perform better on some of the datasets while perform poorly on the other datasets. For example, the classification accuracy of HOG on Caltech7 is 88.29%, while on the Caltech20, the accuracy is only 31.46%. Meanwhile, for each dataset, some features are more effective than the others. Hence, it is nearly impossible for a single feature to perform best on all datasets. To better utilize the diversity and complementarity of multiple views, the naive approach is to directly concatenate multiple features together. However, as shown in Table 1, the performance of this method is not better than the single feature. For instance, on the NUS-WIDE dataset, the classification accuracy of DC-SVM is only 13.31%, compared to 15.21% achieved by the CM feature alone. The reason is that direct feature concatenation may not promote the complementarity of multiple views. Instead, some feature may even introduce noise to other features, deteriorating the whole performance.

As shown in Table 1, more sophisticated multi-view combination methods are compared. Although these approaches all achieve significant performance gains on the six datasets, about 10%$\sim$12% improvement compared with the naive method. However, most of these methods only focus on the shared component across multiple views. Comparatively, our proposed method achieves the best results on all six datasets, and outperforms the previous best results nearly by 7% on average. It is mainly because that our method simultaneously consider the shared component and the private component, which is beneficial to boosting the classification performance.

5 Conclusions

In this paper, we proposed a novel multi-view, low-rank, and sparse matrix decomposition scheme which can robustly integrate diverse yet complementary information from multiple views. Different from traditional approaches, we decomposed a concatenated input matrix into three parts including the low-rank, sparse, and noisy parts, and imposed particular constraints on them. In the presented unified optimization framework, on one hand the low-rankness regularizer was leveraged to capture the shared component across all views in the instance level; on the other hand the group-structured sparsity regularizer was employed to extract the private component from each view in the view level. The final multi-view fusion was conducted by combining the learned shared and private components. The experimental results on six popular image classification datasets demonstrated that our proposed method consistently achieves superior performance over the state-of-the-arts.

In future work, we plan to extend our multi-view matrix decomposition method to work under semi-supervised settings like [Liu *et al.*, 2010; 2012a]. To make our method scalable to massive heterogeneous datasets, we intend to draw on the storage and computational merits of learning based data-dependent hashing techniques [Liu *et al.*, 2011; 2012b; 2014; Shen *et al.*, 2015], and develop new hashing schemes to cater for large-scale multi-view matrix decomposition.

Acknowledgements This work is supported by National High Technology Research and Development Program of China (2013AA01A602), Program for New Century Excellent Talents in University (NCET-12-0917), National Natural Science Foundation of China (No. 61125204 and No. 61472304), and Australian Research Council Projects (FT-130101457, DP-140102164, and LP-140100569).

[1] http://www.csie.ntu.edu.tw/$\sim$cjlin/libsvm/

References

[Afonso *et al.*, 2011] M. V. Afonso, J. M. Bioucas-Dias, and M. A. T. Figueiredo. An augmented lagrangian approach to the constrained optimization formulation of imaging inverse problems. *IEEE Transactions on Image Processing*, 20(3):681–695, 2011.

[Bahrampour *et al.*, 2014] S. Bahrampour, A. Ray, N. M. Nasrabadi, and K. W. Jenkins. Quality-based multimodal classification using tree-structured sparsity. In *Proc. CVPR*, 2014.

[Bucak *et al.*, 2014] S. S. Bucak, R. Jin, and A. K. Jain. Multiple kernel learning for visual object recognition: A review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(7):1354–1369, 2014.

[Cai *et al.*, 2010] J.-F. Cai, E. J. Candès, and Z. Shen. A singular value thresholding algorithm for matrix completion. *SIAM Journal on Optimization*, 20(4):1956–1982, 2010.

[Cai *et al.*, 2013a] X. Cai, F. Nie, W. Cai, and H. Huang. Heterogeneous image features integration via multi-modal semi-supervised learning model. In *Proc. ICCV*, 2013.

[Cai *et al.*, 2013b] X. Cai, F. Nie, and H. Huang. Multi-view k-means clustering on big data. In *Proc. IJCAI*, 2013.

[Cheng *et al.*, 2011] B. Cheng, G. Liu, J. Wang, Z. Huang, and S. Yan. Multi-task low-rank affinity pursuit for image segmentation. In *Proc. ICCV*, 2011.

[Chua *et al.*, 2009] T.-S. Chua, J. Tang, R. Hong, H. Li, Z. Luo, and Y. Zheng. Nus-wide: A real-world web image database from national university of singapore. In *Proc. CIVR*, 2009.

[Fei-Fei *et al.*, 2007] L. Fei-Fei, R. Fergus, and P. Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. *Computer Vision and Image Understanding*, 106(1):59–70, 2007.

[Frank and Asuncion, 2010] A. Frank and A. Asuncion. Uci machine learning repository. 2010.

[Guo *et al.*, 2013] X. Guo, D. Liu, B. Jou, M. Zhu, A. Cai, and S.-F. Chang. Robust object co-detection. In *Proc. CVPR*, 2013.

[Jing *et al.*, 2014] X. Y. Jing, R. M. Hu, Y. P. Zhu, S. S. Wu, C. Liang, and J. Y. Yang. Intra-view and inter-view supervised correlation analysis for multi-view feature learning. In *Proc. AAAI*, 2014.

[Kuss and Graepel, 2003] M. Kuss and T. Graepel. The geometry of kernel canonical correlation analysis. Technical Report TR-108, Max Planck Institute for Biological Cybernetics, Tübingen, Germany, 2003.

[Lampert *et al.*, 2009] C. H. Lampert, H. Nickisch, and S. Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *Proc. CVPR*, 2009.

[Lazebnik *et al.*, 2006] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bag of features: Spatial pyramid matching for recognition natural scene categories. In *Proc. CVPR*, 2006.

[Liu *et al.*, 2010] W. Liu, J. He, and S.-F. Chang. Large graph construction for scalable semi-supervised learning. In *Proc. ICML*, 2010.

[Liu *et al.*, 2011] W. Liu, J. Wang, S. Kumar, and S.-F. Chang. Hashing with graphs. In *Proc. ICML*, 2011.

[Liu *et al.*, 2012a] W. Liu, J. Wang, and S.-F. Chang. Robust and scalable graph-based semisupervised learning. *Proceedings of the IEEE*, 100(9):2624–2638, 2012.

[Liu *et al.*, 2012b] W. Liu, J. Wang, R. Ji, Y.-G. Jiang, and S.-F. Chang. Supervised hashing with kernels. In *Proc. CVPR*, 2012.

[Liu *et al.*, 2014] W. Liu, C. Mu, S. Kumar, and S.-F. Chang. Discrete graph hashing. In *NIPS 27*, 2014.

[Liu *et al.*, 2015] M. Liu, Y. Luo, D. Tao, C. Xu, and Y. Wen. Low-rank multi-view learning in matrix completion for multi-label image classification. In *Proc. AAAI*, 2015.

[Ojala *et al.*, 2002] T. Ojala, M. Pietikäinen, and T. Mäenpää. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):971–987, 2002.

[Oliva and Torralba, 2001] A. Oliva and A. Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *International Journal of Computer Vision*, 42(3):145–175, 2001.

[Quadrianto and Lampert, 2011] N. Quadrianto and C. H. Lampert. Learning multi-view neighborhood preserving projections. In *Proc. ICML*, 2011.

[Rakotomamonjy *et al.*, 2008] A. Rakotomamonjy, F. R. Bach, S. Canu, and Y. Grandvalet. Simplemkl. *Journal of Machine Learning Research*, 9:2491–2521, 2008.

[Shekhar *et al.*, 2014] S. Shekhar, V. M. Patel, N. M. Nasrabadi, and R. Chellappa. Joint sparse representation for robust multimodal biometrics recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(1):113–126, 2014.

[Shen *et al.*, 2015] F. Shen, C. Shen, W. Liu, and H. T. Shen. Supervised discrete hashing. In *Proc. CVPR*, 2015.

[Shon *et al.*, 2005] A. Shon, K. Grochow, A. Hertzmann, and R. P. Rao. Learning shared latent structure for image synthesis and robotic imitation. In *NIPS 18*, 2005.

[Wang *et al.*, 2013] H. Wang, F. Nie, H. Huang, and C. Ding. Heterogeneous visual features fusion via sparse multimodal machine. In *Proc. CVPR*, 2013.

[Wang *et al.*, 2015] Z. Wang, W. Yuan, and G. Montana. Sparse multi-view matrix factorisation: a multivariate approach to multiple tissue comparisons. In *arXiv:1503.01291v1 [stat.ML]*, 2015.

[Wright *et al.*, 2009] J. Wright, A. Ganesh, S. Rao, Y. Peng, and Y. Ma. Robust principal component analysis: Exact recovery of corrupted low-rank matrices via convex optimization. In *NIPS 22*, 2009.

[Wu and Rehg, 2008] J. Wu and J. M. Rehg. Where am i: Place instance and category recognition using spatial pact. In *Proc. CVPR*, 2008.

[Xia *et al.*, 2014] R. Xia, Y. Pan, L. Du, and J. Yin. Robust multi-view spectral clustering via low-rank and sparse decomposition. In *Proc. AAAI*, 2014.