Generalized Latent Multi-View Subspace Clustering

Changqing Zhang, Huazhu Fu, Qinghua Hu, Xiaochun Cao, Yuan Xie, Dacheng Tao, *Fellow, IEEE*, and Dong Xu, *Fellow, IEEE*

Abstract—Subspace clustering is an effective method that has been successfully applied to many applications. Here we propose a novel subspace clustering model for multi-view data using a latent representation termed Latent Multi-View Subspace Clustering (LMSC). Unlike most existing single-view subspace clustering methods, which directly reconstruct data points using original features, our method explores underlying complementary information from multiple views and simultaneously seeks the underlying latent representation. Using the complementarity of multiple views, the latent representation depicts data more comprehensively than each individual view, accordingly making subspace representation more accurate and robust. We proposed two LMSC formulations: linear LMSC (lLMSC), based on linear correlations between latent representation and each view, and generalized LMSC (gLMSC), based on neural networks to handle general relationships. The proposed method can be efficiently optimized under the Augmented Lagrangian Multiplier with Alternating Direction Minimization (ALM-ADM) framework. Extensive experiments on diverse datasets demonstrate the effectiveness of the proposed method.

Index Terms—Multi-view clustering, subspace clustering, latent representation, neural networks.

1 Introduction

Subspace clustering has been successfully used in numerous applications, especially those involving high-dimensional data [1], [2]. Existing subspace clustering approaches can be categorized into iterative methods [3], [4], algebraic approaches [5], [6], statistical methods and spectral clustering-based methods [7], [8]. Recently proposed subspace clustering methods [9], [10], [11], [12], [13], [14] are based on the assumption that data points are drawn from multiple subspaces corresponding to different clusters, where each data point can be expressed by a linear combination of the data points themselves. The general formulation of existing subspace clustering methods is

$$\min_{\mathbf{Z}} \mathcal{L}(\mathbf{X}, \mathbf{X}\mathbf{Z}) + \lambda\Omega(\mathbf{Z}), \tag{1}$$

where $\mathbf{X} = [\mathbf{x}_1, \cdots, \mathbf{x}_n]$ is the $d \times n$ feature matrix whose columns are the samples, and $\lambda > 0$ is the tradeoff factor. The loss function $\mathcal{L}(\cdot, \cdot)$ and regularization term $\Omega(\cdot)$ are usually defined under specific assumptions. The representative approach - Sparse Subspace Clustering (SSC) [9] - focuses on searching for the sparsest representation from an infinite number of possible representations based on the ℓ_1-norm. Unlike SSC, which separately constructs the sparsest representation for each data point, Low-Rank Representation (LRR) [10] tries to find the lowest rank representation of all data jointly by using the structured sparsity loss. Constrained by graph regularization, SMooth Representation clustering (SMR) [11] investigates theoretically the grouping effect for self-representation based approaches. With the reconstruction coefficient matrix $\mathbf{Z}$, the affinity matrix is obtained by $\mathbf{S} = abs(\mathbf{Z}) + abs(\mathbf{Z}^T)$, where $abs(\cdot)$ is the element-wise absolute operator. Finally, with the affinity matrix $\mathbf{S}$ as the input, the final clustering result is obtained by conducting standard spectral clustering [7].

Although these subspace clustering approaches are effective, they tend to be heavily influenced by the original features, especially when the observations are insufficient and/or grossly corrupted. Fortunately, multi-view subspace clustering methods [15], [16], [17] have been proposed to overcome this issue, in which multiple views describe each data point. The complementary information from multiple views can benefit clustering, and the effectiveness has been empirically proven under different multi-view constraints. Existing multi-view subspace clustering methods usually reconstruct the data points on the original view directly and generate individual, view-specific subspace representations, and generally share the following formulation:

$$\min_{\{\mathbf{Z}^{(v)}\}_{v=1}^{V}} \mathcal{L}\big(\{(\mathbf{X}^{(v)}, \mathbf{X}^{(v)}\mathbf{Z}^{(v)})\}_{v=1}^{V}\big) + \lambda\Omega\big(\{\mathbf{Z}^{(v)}\}_{v=1}^{V}\big), \tag{2}$$

where $\mathbf{X}^{(v)}$ and $\mathbf{Z}^{(v)}$ correspond to the feature matrix and subspace representation of the v^{th} view, respectively. Using the above formulation, existing methods employ differen-

- *C. Zhang and Q. Hu are with the School of Computer Science and Technology, Tianjin University, Tianjin 300072, China (e-mail: zhangchangqing@tju.edu.cn; huqinghua@tju.edu.cn).*
- *X. Cao is with the State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093, China (e-mail: caoxiaochun@iie.ac.cn).*
- *H. Fu is with the Inception Institute of Artificial Intelligence, Abu Dhabi, United Arab Emirates (e-mail: huazhufu@gmail.com).*
- *Y. Xie is with the Research Center of Precision Sensing and Control, Institute of Automation, Chinese Academy of Sciences, Beijing, 100190, China (e-mail: yuan.xie@ia.ac.cn).*
- *D. Tao is with the Centre for Quantum Computation and Intelligent Systems, Faculty of Engineering and Information Technology, University of Technology, Sydney, 235 Jones Street, Ultimo, NSW 2007, Australia (e-mail: dacheng.tao@gmail.com).*
- *D. Xu is with the School of Electrical and Information Engineering, University of Sydney, Sydney, NSW 2006, Australia (e-mail: dong.xu@sydney.edu.au.)*

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2018.2877660, IEEE Transactions on Pattern Analysis and Machine Intelligence

t loss functions $\mathcal{L}(\cdot,\cdot)$ and impose different assumptions (with different regularization terms $\Omega(\cdot)$) to explore relationships between subspace representations of multiple views. Although these methods have achieved promising results, they insufficiently describe data within each view, making reconstruction using only information from one view risky. Moreover, noise - which is ubiquitous - further increases the difficulty in reconstruction on the original feature space.

Here we propose using a latent representation for multiple views to explore the relationships between data points and effectively deal with noise. As discussed in [18], [19], the underlying assumption is that multiple views originate from one underlying latent representation, which depicts the essence of the data and reveals the common underlying structure shared by different views. Based on this assumption, we propose a novel method that we call *Latent Multi-view Subspace Clustering* (LMSC). Our approach learns a latent representation to encode complementary information from multi-view features and produces a common subspace representation for all views rather than that of each individual view. More importantly, and expanding on the linear correlation used in our previous work [20], we further generalize our model for non-linear correlation, and accordingly propose *generalized Latent Multi-view Subspace Clustering* (gLMSC). Our method jointly learns the latent representation and multi-view subspace representation within a unified framework, which can be effectively optimized using the Augmented Lagrangian Multiplier with Alternating Direction Minimization (ALM-ADM) strategy. We conduct extensive experiments to compare our method with the current state-of-the-art to demonstrate our model's performance.

The main contributions of this paper are as follows:

- Based on self-representation-based subspace clustering, we propose a novel multi-view subspace clustering method called Latent Multi-view Subspace Clustering (LMSC), which can integrate multiple views into a comprehensive latent representation.
- The automatically learned latent representation encodes complementary information from different views and can meet the self-expressiveness property thus it well reflects the underlying clustering structure.
- In addition to exploring linear correlations between the latent representation and each view, we further introduce neural networks to explore more general relationships and propose the generalized Latent Multi-view Subspace Clustering (gLMSC) method.
- Finally, our formulation is effectively solved by using the Alternating Direction Minimization (ADM) and our optimization algorithm empirically reaches convergence.

The remainder of the paper is organized as follows. Related works, including multi-view learning, multi-view subspace-based clustering, and latent representation-based clustering methods are briefly reviewed in Section 2. Details of our proposed approach are presented in Section 3. In Section 4, we present experimental results that demonstrate our model's performance using a variety of real-world datasets. Conclusions are drawn in Section 5.

2 Related Work

Based on the ubiquitous multi-view data, multi-view learning [21], [22], [23], [24], [25] has shown remarkable success in a wide range of real-world applications. Most existing multi-view clustering methods are *graph-based models*. One of the early methods presented in [26] focuses on handling two-view data. Under a *matrix factorization* framework, some methods [27], [28] attempt to uncover a common representation to link different views for clustering. The *multi-view subspace clustering* methods [15], [16], [17] relate different data points in a self-representing manner on the original view and simultaneously constrain these subspace representations of different views to exploit complementary information. Based on spectral clustering, [29], [30] *co-regularize* the clustering hypothesis of different views to enforce consistence. For large-scale data, a robust, *large-scale*, multi-view k-means clustering method [31] can be parallelized and run on multi-core processors for large-scale data clustering. *Multiple Kernel Learning (MKL)* can be considered a nature way to integrate multiple views. As a result, the method in [32] directly combines multiple kernels corresponding to different views and validated the approach's effectiveness. Based on MKL, [33] further proposes to automatically weight different views. There are some multi-view methods focusing on other topics, e.g., dimensionality reduction [34] and feature selection [35].

Two groups of multi-view subspace clustering methods are most related to ours. The first employs CCA to project multiple views onto a low-dimensional subspace and then uses the learned representation for clustering [36], [37]. The second group are the self-representation-based methods [15], [16], [17]. Diversity-induced Multi-view Subspace Clustering (DiMSC) [15] explores complementary information with Hilbert-Schmidt Independence Criterion (HSIC) under the self-representation subspace clustering framework. Low-Rank Tensor Constrained Multi-view Subspace Clustering (LT-MSC) [16] explores the high-order correlation among these subspace representations. The method in [17] unifies different views with a common indicator matrix rather than a common subspace representation. These methods reconstruct data points within each single view. Instead, our method constructs a unified similarity matrix for multiple views by using a latent representation, and thus well utilizes complementarity across different views for subspace clustering.

Under the self-representation-based subspace clustering framework, some methods [9], [10] introduce latent representations. Latent Space Sparse Subspace Clustering (LS3C) [38] jointly performs dimensionality reduction and sparse coding on sparse subspace clustering [9]. Latent Low-Rank Representation (LatLRR) [39] is based on LRR [10] and constructs a dictionary by jointly using observed and hidden data. The methodology of ours is quite different from these work: (1) Our algorithm performs subspace clustering with the learned common latent representation, while these methods conduct data self-representation within each single view. (2) The correlations among different views are linear for these methods, while our algorithm gLMSC explores more general correlations by neural networks. There are also some algorithms for multi-view representation learning.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2018.2877660, IEEE Transactions on Pattern Analysis and Machine Intelligence

Some approaches [18], [19], [40] explicitly learn a common representation for multiple views as a joint optimization problem with a common subspace representation matrix. Generalized Multiview Analysis (GMA) [41] is an extension of Canonical Correlational Analysis (CCA), which is designed for cross-view classification and retrieval. Multiview LSA [42] is an algorithm that can efficiently approximate Generalized Canonical Correlational Analysis (GCCA). Beyond kernel technique, Deep Canonical Correlation Analysis (DCCA) [43] explores nonlinear correlation between views with neural networks. Some recent approaches [44], [45] aim to learn a new representation based on auto-encoders. In contrast to these methods, which learn the latent representation by linearly [38] or non-linearly [44], [45] mapping the original single-view data, our method jointly recovers the latent multi-view representation and the mappings corresponding to different views to encode the intrinsic complementary information.

3 Latent Multi-view Subspace Clustering

In our method, subspace clustering is performed based on the latent representation encoding complementary information in multiple views. Specifically, given n multi-view observations $\{[\mathbf{x}_i^{(1)};...;\mathbf{x}_i^{(V)}]\}_{i=1}^n$ consisting of V different views, our model aims to seek a shared multi-view latent representation $\mathbf{h}$ for each data point. The underlying assumption is that these different views originate from one underlying latent representation. Basically, in one respect, the information from different views should be encoded into the learned representation. In another respect, the learned latent representation should meet the specific task (task-oriented goal), e.g., self-representation or subspace reconstruction. Therefore, we consider the general objective function

$$\underbrace{\mathcal{I}(\{\mathbf{X}_v\}_{v=1}^V, \mathbf{H}; \mathbf{\Theta}_1)}_{\text{information preservation}} + \lambda \underbrace{\mathcal{S}(\mathbf{H}; \mathbf{\Theta}_2)}_{\text{task-oriented goal}}. \tag{3}$$

where $\mathbf{H} = [\mathbf{h}_1, \cdots, \mathbf{h}_n] \in \mathbb{R}^{k\times n}$ is the latent representation matrix. The first term $\mathcal{I}(\cdot,\cdot)$ ensures that the latent representation encodes information from the original views, thus avoids the bias of the latent representation towards the specific task. The second term $\mathcal{S}(\cdot,\cdot)$ is the task-oriented term. $\lambda > 0$ balances the two terms. $\mathbf{\Theta}_1$ and $\mathbf{\Theta}_2$ are the parameters corresponding to $\mathcal{I}(\cdot,\cdot)$ and $\mathcal{S}(\cdot,\cdot)$, respectively.

Specifically, for latent multi-view subspace clustering, which aims to explore the subspace structure based on the latent representation, we have the following formulation:

$$\min_{\boldsymbol{\theta}_v,\mathbf{H},\mathbf{Z}} \mathcal{L}_S(\mathbf{H}, \mathbf{HZ}) + \sum_{v=1}^{V} \alpha_v \mathcal{L}_V(\mathcal{F}_v(\mathbf{H};\boldsymbol{\theta}_v), \mathbf{X}^{(v)}) + \lambda\Omega(\mathbf{Z}), \tag{4}$$

where $\mathcal{L}_S(\mathbf{H}, \mathbf{HZ})$ is the loss function for the subspace representation. $\mathcal{L}_V(\mathcal{F}_v(\mathbf{H};\boldsymbol{\theta}_v), \mathbf{X}^{(v)})$ and $\mathcal{F}_v(\mathbf{H};\boldsymbol{\theta}_v)$ are the reconstruction loss and underlying mapping from the latent representation $\mathbf{H}$ to the observations for the v^{th} view, respectively. The tradeoff factors $\alpha_v > 0$ and $\lambda > 0$ are used to control the influence of the v^{th} view and regularization degree of subspace representation, respectively. With objective function (4), we can learn the latent multi-view representation, which benefits from complementarity of all views and is therefore beneficial to subspace clustering. In our work, we propose two latent multi-view subspace clustering (LMSC) methods: linear (l)LMSC and generalized (g)LMSC.

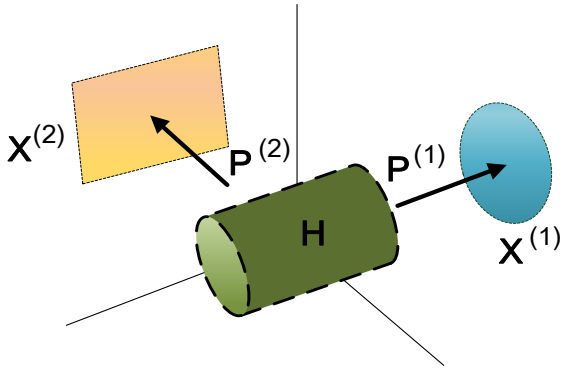


Fig. 1: Illustration of multi-view latent representation. Observations $\{\mathbf{X}^{(v)}\}_{v=1}^V$ ($V \geq 2$) corresponding to different views are partially projected by $\{\mathbf{P}^{(v)}\}_{v=1}^V$ from one underlying latent representation $\mathbf{H}$.

3.1 Linear Latent Multi-view Subspace Clustering

We first model the correlation between the latent representation and each view by using a linear model, termed linear Latent Multi-view Subspace Clustering (lLMSC). As shown in Fig. 1, observations corresponding to different views can be linearly recovered with their respective models $\{\mathbf{P}^{(1)}$, ..., $\mathbf{P}^{(V)}\}$ based on the shared latent representation $\mathbf{h}_i$, i.e., $\mathbf{x}_i^{(v)} = \mathbf{P}^{(v)}\mathbf{h}_i$. Considering noise in observations, we have

$$\mathbf{x}_i^{(v)} = \mathbf{P}^{(v)}\mathbf{h}_i + \mathbf{e}_i^{(v)}, \tag{5}$$

where $\mathbf{e}_i^{(v)}$ is the noise of the i^{th} sample in the v^{th} view. To infer the multi-view latent representation, the objective function becomes

$$\min_{\mathbf{P},\mathbf{H}} \mathcal{L}_V(\mathbf{X}, \mathbf{PH}), \quad \text{with } \mathbf{X} = \begin{bmatrix} \mathbf{X}^{(1)} \\ \cdots \\ \mathbf{X}^{(V)} \end{bmatrix} \text{ and } \mathbf{P} = \begin{bmatrix} \mathbf{P}^{(1)} \\ \cdots \\ \mathbf{P}^{(V)} \end{bmatrix}, \tag{6}$$

where $\mathbf{X}$ and $\mathbf{P}$ are the observations and reconstruction models concatenated and aligned according to multiple views, respectively. The loss function $\mathcal{L}_V(\cdot,\cdot)$ is associated with the reconstruction from the latent (hidden) representation to different views. In this way, complementary information from multiple views is automatically encoded into the latent representation $\mathbf{H}$, making it more comprehensive than that of each single view individually.

For the task-oriented goal (the second term in Eq. (3)), our aim is to perform subspace clustering as in Eq. (1). Therefore, the objective function based on latent representation $\mathbf{H}$ is reformulated as

$$\min_{\mathbf{Z}} \mathcal{L}_S(\mathbf{H}, \mathbf{HZ}) + \lambda\Omega(\mathbf{Z}), \tag{7}$$

where the loss function $\mathcal{L}_S(\mathbf{H}, \mathbf{HZ})$ is defined based on the self-representation-based reconstruction error. The reconstruction coefficient matrix $\mathbf{Z}$ is regularized with $\Omega(\mathbf{Z})$.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2018.2877660, IEEE Transactions on Pattern Analysis and Machine Intelligence

For multi-view subspace clustering, we jointly conduct latent representation learning in Eq. (6) and subspace clustering in Eq. (7) within one unified objective function

$$\min_{\mathbf{P},\mathbf{H},\mathbf{Z}} \mathcal{L}_V(\mathbf{X},\mathbf{PH}) + \lambda_1 \mathcal{L}_S(\mathbf{H},\mathbf{HZ}) + \lambda_2 \Omega(\mathbf{Z}), \tag{8}$$

where $\lambda_1 > 0$ and $\lambda_2 > 0$ are the tradeoff parameters used to balance the three terms. Generally, the quality of subspace clustering is improved by a comprehensive latent representation, while the quality of the latent representation is ensured by the complementary information from multiple views and the clustering structure identification. Considering outliers, the objective function of lLMSC is

$$\begin{aligned} &\min_{\mathbf{P},\mathbf{H},\mathbf{Z},\mathbf{E}_V,\mathbf{E}_S} \|\mathbf{E}_V\|_{2,1} + \lambda_1 \|\mathbf{E}_S\|_{2,1} + \lambda_2 \|\mathbf{Z}\|_* \\ &s.t.\ \mathbf{X} = \mathbf{PH} + \mathbf{E}_V, \mathbf{H} = \mathbf{HZ} + \mathbf{E}_S \text{ and } \mathbf{PP}^T = \mathbf{I}, \end{aligned} \tag{9}$$

where $\mathbf{E}_V$ and $\mathbf{E}_S$ denote the errors corresponding to reconstruction from the latent representation to each view and subspace representation, respectively. The subspace representation is ensured to be low-rank with matrix nuclear norm $||\cdot||_*$. The $\ell_{2,1}$-norm $||\cdot||_{2,1}$ enforces columns of a matrix to be zero [10]. The definition of $\ell_{2,1}$-norm used for a matrix ($\mathbf{A}$) is: $\|\mathbf{A}\|_{2,1} = \sum_{j=1}^{q} \sqrt{\sum_{i=1}^{p} \mathrm{A}_{ij}^2}$ with $\mathbf{A} \in \mathbb{R}^{p\times q}$. It is robust to outliers due to its underlying assumption that the corruptions are sample-specific. The projection matrix $\mathbf{P}$ is constrained to avoid $\mathbf{H}$ being pushed arbitrarily close to zero, since rescaling $\mathbf{H}/s$ and $\mathbf{P}s$ $(s > 0)$ preserves the same loss. For our objective function, the first term ensures that the latent representation $\mathbf{H}$ is comprehensive, while the second term relates data points with subspace representation. The last term finds the lowest rank subspace representation and prevents a trivial solution. Note that our model holds the robustness from: (1) complementary information in different views enhances robustness compared to each single view, subsequently improving clustering; (2) the structured sparsity regularization with the $\ell_{2,1}$-norm on the error handles outliers well compared to the Frobenius norm.

To ensure that the outliers are consistent with the errors $\mathbf{E}_S$ and $\mathbf{E}_V$, we vertically concatenate them along column. This enforces $\mathbf{E}_S$ and $\mathbf{E}_V$ to be with the same pattern of column-wise sparsity [46]. Accordingly, the final objective function of our lLMSC is formulated as

$$\begin{aligned} &\min_{\mathbf{P},\mathbf{H},\mathbf{Z},\mathbf{E}_V,\mathbf{E}_S} \|\mathbf{E}\|_{2,1} + \lambda \|\mathbf{Z}\|_* \\ &s.t.\ \mathbf{X} = \mathbf{PH} + \mathbf{E}_V,\ \mathbf{H} = \mathbf{HZ} + \mathbf{E}_S, \\ &\quad \mathbf{E} = [\mathbf{E}_V; \mathbf{E}_S] \text{ and } \mathbf{PP}^T = \mathbf{I}. \end{aligned} \tag{10}$$

In our model, only one parameter $\lambda > 0$ is involved to balance the reconstruction error and regularization on subspace representation.

3.1.1 lLMSC Optimization

According to the objective function of our lLMSC in Eq. (10), we simultaneously seek the effective latent representations from different views and obtain the affinity matrix based on the latent representations. Since it is not jointly convex for all the variables, we divide our objective function into subproblems that can be efficiently solved. We employ the Augmented Lagrange Multiplier (ALM) with Alternating Direction Minimization (ADM) strategy [47] for our optimization. To adopt the ADM strategy, the objective function should be separable. Hence, auxiliary variable $\mathbf{J}$ is introduced to replace $\mathbf{Z}$. Accordingly, the following problem, which is equivalent to Eq. (10), is proposed:

$$\begin{aligned} &\min_{\mathbf{P},\mathbf{H},\mathbf{Z},\mathbf{E}_V,\mathbf{E}_S,\mathbf{J}} \|\mathbf{E}\|_{2,1} + \lambda \|\mathbf{J}\|_* \\ &s.t.\ \mathbf{X} = \mathbf{PH} + \mathbf{E}_V,\ \mathbf{H} = \mathbf{HZ} + \mathbf{E}_S, \\ &\quad \mathbf{E} = [\mathbf{E}_V; \mathbf{E}_S],\ \mathbf{PP}^T = \mathbf{I} \text{ and } \mathbf{J} = \mathbf{Z}. \end{aligned} \tag{11}$$

To solve the above objective function, we minimize the following ALM problem:

$$\begin{aligned} &\mathcal{L}(\mathbf{P},\mathbf{H},\mathbf{Z},\mathbf{E}_V,\mathbf{E}_S,\mathbf{J}) \\ &= \|\mathbf{E}\|_{2,1} + \lambda \|\mathbf{J}\|_* \\ &+ \Phi(\mathbf{Y}_1, \mathbf{X} - \mathbf{PH} - \mathbf{E}_V) \\ &+ \Phi(\mathbf{Y}_2, \mathbf{H} - \mathbf{HZ} - \mathbf{E}_S) + \Phi(\mathbf{Y}_3, \mathbf{J} - \mathbf{Z}) \\ &s.t.\ \mathbf{E} = [\mathbf{E}_V; \mathbf{E}_S];\ \mathbf{PP}^T = \mathbf{I}. \end{aligned} \tag{12}$$

Note that, for better presentation, we have the definition as: $\Phi(\mathbf{C},\mathbf{D}) = \frac{\mu}{2}||\mathbf{D}||_F^2 + \langle \mathbf{C},\mathbf{D}\rangle$, where $\langle\cdot,\cdot\rangle$ is known as the Frobenius inner product defined by $\langle \mathbf{A},\mathbf{B}\rangle = tr(\mathbf{A}^T\mathbf{B})$. $\mu > 0$ is the penalty scalar and $\mathbf{C}$ is the Lagrangian multiplier. According to the Alternating Direction Minimization (ADM) strategy [47], we separate our objective into subproblems that can be efficiently optimized. Then, the optimization is cycled over all variables while keeping the previously updated variables fixed. Specifically, each subproblem is solved as follows:

1. P-subproblem: With other variables fixed, we should optimize the following problem for updating $\mathbf{P}$:

$$\begin{aligned} &\mathbf{P}^* = \arg\min_{\mathbf{P}} \Phi(\mathbf{Y}_1, \mathbf{X} - \mathbf{PH} - \mathbf{E}_V) \\ &s.t.\ \mathbf{PP}^T = \mathbf{I}. \end{aligned} \tag{13}$$

To efficiently solve the above problem, we introduce Theorem 1 [48] which is used for "Wahba's problem", i.e., seeking a (orthogonal) rotation matrix between two coordinate systems given a set of observations.

Theorem 1. *Given the objective function* $\min_{\mathbf{R}} ||\mathbf{Q} - \mathbf{GR}||_F^2$ *s.t.* $\mathbf{R}^T\mathbf{R} = \mathbf{RR}^T = \mathbf{I}$, *the optimal solution is* $\mathbf{R} = \mathbf{UV}^T$, *where* $\mathbf{U}$ *and* $\mathbf{V}$ *are left and right singular values of SVD decomposition of* $\mathbf{G}^T\mathbf{Q}$.

We can show that $\mathbf{P}^T = \mathbf{UV}^T$ is the optimal solution for the $\mathbf{P}$-subproblem with $\mathbf{U}$ ($\mathbf{V}$) being the left (right) singular values of $\mathbf{H}(\mathbf{X} + \mathbf{Y}_1/\mu - \mathbf{E}_V)^T$. Specifically, we have

$$\begin{aligned} \mathbf{P}^* &= \arg\min_{\mathbf{P}} \Phi(\mathbf{Y}_1, \mathbf{X} - \mathbf{PH} - \mathbf{E}_V) \\ &= \arg\min_{\mathbf{P}} \frac{\mu}{2} ||\mathbf{X} - \mathbf{PH} - \mathbf{E}_V + \mathbf{Y}_1/\mu||_F^2 \\ &= \arg\min_{\mathbf{P}} \frac{\mu}{2} ||(\mathbf{X} + \mathbf{Y}_1/\mu - \mathbf{E}_V) - \mathbf{PH}||_F^2 \\ &= \arg\min_{\mathbf{P}} \frac{\mu}{2} ||(\mathbf{X} + \mathbf{Y}_1/\mu - \mathbf{E}_V)^T - \mathbf{H}^T\mathbf{P}^T||_F^2. \end{aligned}$$

According to Theorem 1, if $\mathbf{P}$ is constrained to be orthogonal (i.e., $\mathbf{PP}^T = \mathbf{P}^T\mathbf{P} = \mathbf{I}$), $\mathbf{P}^T = \mathbf{UV}^T$ will be the optimal solution. In practice, the constraint for $\mathbf{P}$ could be relaxed (i.e., $\mathbf{PP}^T = \mathbf{I}$, where $\mathbf{P} \in \mathbb{R}^{k\times d}, k \ll d$). Promising performance and convergence results validate this relaxation.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2018.2877660, IEEE Transactions on Pattern Analysis and Machine Intelligence

TABLE 1: Main notations used throughout the paper.

Model Specification	
Notation	Meaning
$\mathbf{X}^{(v)} \in \mathbb{R}^{d_v \times n}$	The feature matrix of the v^{th} view
$\mathbf{H} \in \mathbb{R}^{k \times n}$	The learned latent representation matrix
$\mathbf{Z} \in \mathbb{R}^{n \times n}$	The subspace representation matrix
$\mathbf{P} \in \mathbb{R}^{k \times d}, d = \sum_v d_v$	The projection from latent representation to all views
$\mathbf{E}_S \in \mathbb{R}^{k \times n}$, $\mathbf{E}_V \in \mathbb{R}^{d \times n}$	The reconstruction errors
$\mathbf{Y}_1$,$\mathbf{Y}_2$, $\mathbf{Y}_3$	Lagrangian multipliers for constraints
$\mathbf{W}_{(1,v)} \in \mathbb{R}^{d_{(1,v)} \times k}$	The neural networks parameters
$\mathbf{W}_{(2,v)} \in \mathbb{R}^{d_{(2,v)} \times d_{(1,v)}}$	The neural networks parameters

2. H-subproblem: To update $\mathbf{H}$, the following objective should be optimized:

$$\begin{aligned}\mathbf{H}^* = \arg\min_{\mathbf{H}} &\Phi(\mathbf{Y}_1, \mathbf{X} - \mathbf{P}\mathbf{H} - \mathbf{E}_V) \\ &+ \Phi(\mathbf{Y}_2, \mathbf{H} - \mathbf{H}\mathbf{Z} - \mathbf{E}_S).\end{aligned} \tag{14}$$

Differentiating the objective function with respect to $\mathbf{H}$ and then setting the derivative to zero, we obtain the following equation:

$$\begin{aligned}&\mathbf{A}\mathbf{H} + \mathbf{H}\mathbf{B} = \mathbf{C} \\ \text{with } &\mathbf{A} = \mu\mathbf{P}^T\mathbf{P}, \mathbf{B} = \mu(\mathbf{Z}\mathbf{Z}^T - \mathbf{Z} - \mathbf{Z}^T + \mathbf{I}), \\ &\mathbf{C} = (\mathbf{P}^T\mathbf{Y}_1 + \mathbf{Y}_2(\mathbf{Z}^T - \mathbf{I})) \\ &+ \mu(\mathbf{P}^T\mathbf{X} + \mathbf{E}_S^T - \mathbf{P}^T\mathbf{E}_V - \mathbf{E}_S\mathbf{Z}^T).\end{aligned} \tag{15}$$

Equation (15) is a Sylvester equation [49]. For stability, matrix $\mathbf{A}$ is enforced to be strictly positive-definite with $\hat{\mathbf{A}} = \mathbf{A} + \epsilon\mathbf{I}$. The matrix $\mathbf{I}$ is an identity matrix and ϵ is a small positive scalar, i.e., $0 < \epsilon \ll 1$.

Proposition 1. *The Sylvester equation (15) has a unique solution.*

Proof. There is a unique solution for Sylvester equation $\hat{\mathbf{A}}\mathbf{H} + \mathbf{H}\mathbf{B} = \mathbf{C}$ with respect to $\mathbf{H}$ if there is no common eigenvalue for $\hat{\mathbf{A}}$ and -$\mathbf{B}$ [49]. Matrix $\hat{\mathbf{A}}$ is positive-definite, so all eigenvalues of $\hat{\mathbf{A}}$ are positive, i.e., $\alpha_i > 0$. Matrix $\mathbf{B}$ is positive semi-definite, so all eigenvalues of $\mathbf{B}$ are non-negative, i.e., $\beta_i \geq 0$. Therefore, $\alpha_i + \beta_j > 0$ holds for any eigenvalues of $\hat{\mathbf{A}}$ and $\mathbf{B}$. Accordingly, there is a unique solution for Sylvester equation (15). □

Remark. We employ the Bartels-Stewart algorithm [49] to solve the Sylvester equation. In this algorithm, the coefficient matrices are transformed into Schur forms by QR decomposition before employing back-substitution to solve the obtained triangular system. Note that, the proposed model can be solved exactly under the condition $\mathbf{P}\mathbf{P}^T = \mathbf{P}^T\mathbf{P} = \mathbf{I}$. That is to say, when $\mathbf{A} = \mathbf{P}^T\mathbf{P}$ is a positive-definite matrix, and $\mathbf{P}$ is orthogonal.

3. Z-subproblem: With the other variables fixed, the subspace representation matrix $\mathbf{Z}$ can be updated by optimizing the following objective function:

$$\mathbf{Z}^* = \arg\min_{\mathbf{Z}} \Phi(\mathbf{Y}_3, \mathbf{J} - \mathbf{Z}) + \Phi(\mathbf{Y}_2, \mathbf{H} - \mathbf{H}\mathbf{Z} - \mathbf{E}_S). \tag{16}$$

Accordingly, the following update rule is obtained:

$$\begin{aligned}\mathbf{Z}^* = (\mathbf{H}^T\mathbf{H} + \mathbf{I})^{-1}[&(\mathbf{J} + \mathbf{H}^T\mathbf{H} - \mathbf{H}^T\mathbf{E}_S) \\ &+ (\mathbf{Y}_3 + \mathbf{H}^T\mathbf{Y}_2)/\mu].\end{aligned} \tag{17}$$

4. E-subproblem: To update the reconstruction error $\mathbf{E}$, we need to solve the following problem:

$$\begin{aligned}\mathbf{E}^* &= \arg\min_{\mathbf{E}} \|\mathbf{E}\|_{2,1} + \Phi(\mathbf{Y}_1, \mathbf{X} - \mathbf{P}\mathbf{H} - \mathbf{E}_V) \\ &\qquad + \Phi(\mathbf{Y}_2, \mathbf{H} - \mathbf{H}\mathbf{Z} - \mathbf{E}_S) \\ &= \arg\min_{\mathbf{E}} \frac{1}{\mu}\|\mathbf{E}\|_{2,1} + \frac{1}{2}\|\mathbf{E} - \mathbf{G}\|_F^2,\end{aligned} \tag{18}$$

where the matrix $\mathbf{G}$ is constructed by vertically concatenating $\mathbf{X} - \mathbf{P}\mathbf{H} + \mathbf{Y}_1/\mu$ and $\mathbf{H} - \mathbf{H}\mathbf{Z} + \mathbf{Y}_2/\mu$. The optimal solution can be obtained by Lemma 3.2 in [10].

5. J-subproblem: With the other variable fixed, we obtain the following objective function with respect to $\mathbf{J}$:

$$\begin{aligned}\mathbf{J}^* &= \arg\min_{\mathbf{J}} \lambda\|\mathbf{J}\|_* + \Phi(\mathbf{Y}_3, \mathbf{J} - \mathbf{Z}) \\ &= \frac{\lambda}{\mu}\|\mathbf{J}\|_* + \frac{1}{2}\|\mathbf{J} - (\mathbf{Z} - \mathbf{Y}_3/\mu)\|_F^2.\end{aligned} \tag{19}$$

This low-rank approximation problem can be solved with the singular value thresholding (SVT) algorithm [50].

6. Updating multipliers: The multipliers can be updated with the following rule:

$$\begin{cases}\mathbf{Y}_1 = \mathbf{Y}_1 + \mu(\mathbf{X} - \mathbf{P}\mathbf{H} - \mathbf{E}_V) \\ \mathbf{Y}_2 = \mathbf{Y}_2 + \mu(\mathbf{H} - \mathbf{H}\mathbf{Z} - \mathbf{E}_S) \\ \mathbf{Y}_3 = \mathbf{Y}_3 + \mu(\mathbf{J} - \mathbf{Z}).\end{cases} \tag{20}$$

The complete algorithm of lLMSC is shown in Algorithm 1.

Algorithm 1: Optimization algorithm for lLMSC

Input: Multi-view matrices: $\{\mathbf{X}^{(1)}, ..., \mathbf{X}^{(V)}\}$, hyperparameter λ and the dimension k of latent representation $\mathbf{H}$.
Initialize: $\mathbf{P} = 0$, $\mathbf{E}_V = 0$, $\mathbf{E}_S = 0$, $\mathbf{J} = \mathbf{Z} = 0$, $\mathbf{Y}_1 = 0$, $\mathbf{Y}_2 = 0$, $\mathbf{Y}_3 = 0$, $\mu = 10^{-6}$, $\rho = 1.2$, $\epsilon = 10^{-4}$, $\max_\mu$ =10^6; Initialize $\mathbf{H}$ with random values.
while *not converged* **do**
  Update variables $\mathbf{P}, \mathbf{H}, \mathbf{Z}, \mathbf{E}_V, \mathbf{E}_S, \mathbf{J}$ according to *subproblems 1-5*;
  Update multipliers $\mathbf{Y}_1, \mathbf{Y}_2, \mathbf{Y}_3$ according to *subproblems 6*;
  Update the parameter μ by $\mu = \min(\rho\mu; \max_\mu)$;
  Check the convergence conditions: $||\mathbf{X} - \mathbf{P}\mathbf{H} - \mathbf{E}_V||_\infty < \epsilon$, $||\mathbf{H} - \mathbf{H}\mathbf{Z} - \mathbf{E}_S)||_\infty < \epsilon$ and $||\mathbf{J} - \mathbf{Z}||_\infty < \epsilon$.
end
Output: $\mathbf{Z}$, $\mathbf{H}$, $\mathbf{P}$ and $\mathbf{E}$.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2018.2877660, IEEE Transactions on Pattern Analysis and Machine Intelligence

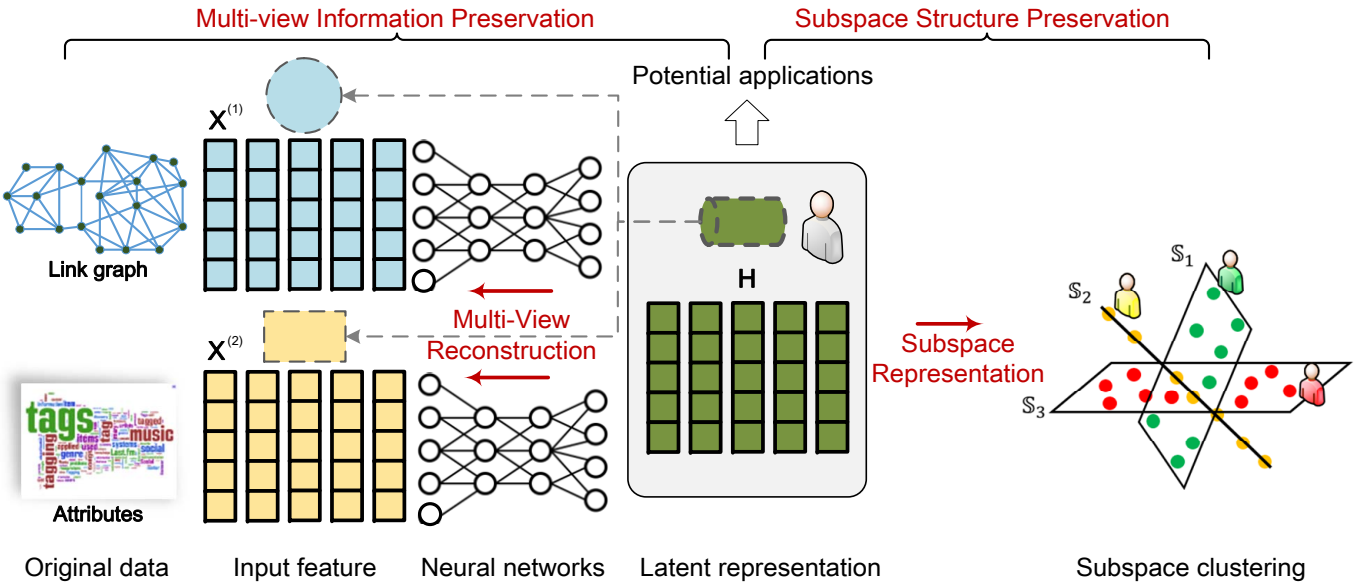


Fig. 2: Illustration of the proposed generalized Latent Multi-view Subspace Clustering (gLMSC). The latent representation non-linearly encodes the information from multiple views with neural networks for uncovering the data distribution in subspaces. Our model can also be considered as an unsupervised multi-view representation learning method, where the learned representation could be used for other potential applications. For comparison, the dashlines indicate the linear LMSC (lLMSC) mentioned in subsection 3.1.

Remark. Several details of our algorithm must be clarified. (1) We employ linear projection which is effective and easy to resolve. The non-linear correlation is addressed in the next subsection. (2) For the $\mathbf{P}$-subproblem optimization, although orthogonal condition is needed for the strict correctness, promising performance and stable convergence are achieved with low-dimensional projection in practice. Moreover, with other constraints for $\mathbf{P}$ (e.g., $||\mathbf{P}(:,j)||^2 \leq 1$), it can be solve with the ADMM algorithm [51]. Although it has similar performance, the inner iteration with ADMM makes the algorithm complexity much greater. (3) It is not appropriate to initialize $\mathbf{H}$ with a zero value. In this case, the optimal solution for $\mathbf{H}$-subproblem will be zero, and subsequent optimizations for the other subproblems (e.g., $\mathbf{Z}$-subproblem in Eq. (16)) will have trivial solutions. Therefore, we initialize $\mathbf{H}$ randomly in our implementation, and $\mathbf{H}$ can also be initialized with other preprocessing (e.g., PCA) to address the instability issue.

3.2 Generalized Latent Multi-view Subspace Clustering

lLMSC assumes a linear relationship between the latent representation and the features from each view. Accordingly, relationships between different views are also linear. Nevertheless, in real-world applications, relationships are usually much more complex and non-linear. The kernel trick is regularly adopted to implicitly address the non-linearity problem by mapping data points onto a high-dimensional space and then solving the learning algorithms in that space. However, the kernel is usually selected in an ad hoc manner and hence suffers from generalization problem. Neural network-based methods [52], [53] can flexibly learn highly non-linear mappings, so here we employ neural networks to address complex relationships between the latent representation and the features from individual views, and the non-linear interactions among multiple views. Accordingly, we propose the generalized Latent Multi-view Subspace Clustering (gLMSC) method shown in Fig. 2.

The objective function of gLMSC is formulated as follows:

$$\begin{aligned}&\min_{\{\boldsymbol{\theta}_v\}_{v=1}^V,\mathbf{H},\mathbf{Z}} \ell(\mathbf{H},\mathbf{HZ}) + \sum_{v=1}^{V} \alpha_v d_v(\mathbf{X}_v, g_{\boldsymbol{\theta}_v}(\mathbf{H})) + \lambda\Omega(\mathbf{Z}) \\ &\text{with } g_{\boldsymbol{\theta}_v}(\mathbf{H}) = \mathbf{W}_{(k,v)} f(\mathbf{W}_{(k-1,v)} \cdots f(\mathbf{W}_{(1,v)}\mathbf{H})),\end{aligned} \tag{21}$$

where $\ell(\cdot,\cdot)$ (corresponding to $\mathcal{L}_S(\cdot,\cdot)$ in (4)) is the loss for subspace representation, and $d_v(\cdot,\cdot)$ (corresponding to $\mathcal{L}_V(\cdot,\cdot)$ in (4)) measures the distortion of reconstruction from the latent representation to the observation in the v^{th} view. The neural network $g_{\boldsymbol{\theta}_v}(\mathbf{H})$ accounts for the non-linear mapping, with $f(\cdot)$ being the activation function and $\mathbf{W}_{(k,v)}$ being the weight matrix of between the k^{th} and $(k+1)^{th}$ layers for the v^{th} view. The tradeoff factor α_v is used to control the fusion portion from the v^{th} view, which encodes the influence of the v^{th} view on the latent representation. By using a three-layer network, we propose the following objective function for gLMSC under the low-rank constraint for subspace representation:

$$\begin{aligned}&\min_{\{\boldsymbol{\theta}_v\}_{v=1}^V,\mathbf{H},\mathbf{Z}} \frac{1}{2}\left\|\mathbf{H}-\mathbf{HZ}\right\|_F^2 \\ &+ \sum_{v=1}^{V} \frac{\alpha_v}{2}\left\|\mathbf{X}_v - \mathbf{W}_{(2,v)} f(\mathbf{W}_{(1,v)}\mathbf{H})\right\|_F^2 + \lambda\left\|\mathbf{Z}\right\|_*,\end{aligned} \tag{22}$$

where the activation function used in our model is the *tanh* function which is defined as:

$$f(a) = tanh(a) = \frac{1-e^{2a}}{1+e^{-2a}}. \tag{23}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2018.2877660, IEEE Transactions on Pattern Analysis and Machine Intelligence

Accordingly, the corresponding derivative can be calculated as:

$$f'(a) = tanh'(a) = 1 - tanh^2(a). \tag{24}$$

To summarize, gLMSC has the following merits. (1) Our model focuses on seeking the comprehensive common representation of multiple views, based on which (and instead of each single view) subspace clustering is performed. (2) Since subspace clustering is specific for high-dimensional data, therefore, for existing methods (e.g., [15], [16]), the data should not have low-dimensional views. In contrast, our method is free of the restraint due to the latent representation. (3) Inter-view correlations are implicitly encoded by the network which non-linearly maps the latent representation to reconstruct each view. (4) Our framework has flexibility due to the use of different components, i.e., both the network and the regularization terms are replaceable (for example with low-rank/sparse/graph regularization); 5) Although our work focuses on subspace clustering, gLMSC can be considered a general multi-view representation learning framework.

3.2.1 gLMSC Optimization

The objective function in Eq. (22) can be solved as follows:
• Update the network parameters, i.e., $\mathbf{W}_{(1,v)}$ and $\mathbf{W}_{(2,v)}$. Letting $\mathbf{M}_v = tanh(\mathbf{W}_{(1,v)}\mathbf{H})$ and imposing regularization on $\mathbf{W}_{(1,v)}$ and $\mathbf{W}_{(2,v)}$, for the v^{th} view, we have

$$\mathcal{L}_W = \frac{\alpha_v}{2}\left\|\mathbf{X}_v - \mathbf{W}_{(2,v)}f(\mathbf{W}_{(1,v)}\mathbf{H})\right\|_F^2 + \gamma\Omega(\mathbf{\Theta}), \tag{25}$$

where $\Omega(\mathbf{\Theta}) = (\left\|\mathbf{W}_{(1,v)}\right\|_F^2 + \left\|\mathbf{W}_{(2,v)}\right\|_F^2)$ and $\gamma > 0$ is the tradeoff parameter for model regularization of the network. Then, we have

$$\mathbf{W}_{(2,v)} = \mathbf{X}_v\mathbf{M}_v^T(\mathbf{M}_v\mathbf{M}_v^T + \frac{\gamma}{\alpha_v}\mathbf{I})^{-1} \tag{26}$$

and

$$\frac{\partial\mathcal{L}_W}{\mathbf{W}_{(1,v)}} = \alpha_v\Big[(\mathbf{1} - \mathbf{M}_v \circ \mathbf{M}_v)\circ (\mathbf{W}_{(2,v)}^T\mathbf{W}_{(2,v)}\mathbf{M}_v - \mathbf{W}_{(2,v)}^T\mathbf{X}_v)\Big]\mathbf{H}^T + \gamma\mathbf{W}_{(1,v)}, \tag{27}$$

where $\circ$ denotes element-wise multiplication, $\mathbf{1}$ is a matrix whose elements are all ones, and $\mathbf{1}-\mathbf{M}_v\circ\mathbf{M}_v$ is the gradient of $\mathbf{M}_v = tanh(\mathbf{W}_{(1,v)}\mathbf{H})$. We update $\mathbf{W}_{(1,v)}$ using the gradient descent (GD) algorithm. The optimization procedure of our neural networks is summarized in Algorithm 2.
• Update $\mathbf{H}$. The update of $\mathbf{H}$ is similar to that of $\mathbf{W}_{(1,v)}$ as follows:

$$\frac{\partial\mathcal{L}_H}{\mathbf{H}} = \sum_{v=1}^{V}\alpha_v\mathbf{W}_{(1,v)}^T\Big[(\mathbf{1} - \mathbf{M}_v \circ \mathbf{M}_v)\circ (\mathbf{W}_{(2,v)}^T\mathbf{W}_{(2,v)}\mathbf{M}_v - \mathbf{W}_{(2,v)}^T\mathbf{X}_v)\Big] + \mathbf{H}(\mathbf{I} - \mathbf{Z} - \mathbf{Z}^T + \mathbf{Z}\mathbf{Z}^T)$$
$$\text{with } \mathcal{L}_H = \frac{1}{2}\left\|\mathbf{H} - \mathbf{H}\mathbf{Z}\right\|_F^2 + \sum_{v=1}^{V}\frac{\alpha_v}{2}\left\|\mathbf{X}_v - \mathbf{W}_{(2,v)}f(\mathbf{W}_{(1,v)}\mathbf{H})\right\|_F^2. \tag{28}$$

We update $\mathbf{H}$ using the gradient descent (GD) algorithm.
• Update $\mathbf{Z}$. To update $\mathbf{Z}$, we introduce an auxiliary variable $\mathbf{J}$ and iteratively update $\mathbf{Z}$, $\mathbf{J}$ and the multiplier $\mathbf{Y}$ with ADMM as follows:

$$\begin{aligned}&\mathbf{Z} = (\mathbf{H}^T\mathbf{H} + \mu\mathbf{I})^{-1}(\mu\mathbf{J} - \mathbf{Y} + \mathbf{H}^T\mathbf{H}),\\ &\mathbf{J} = \arg\min_{\mathbf{J}}\frac{\lambda}{\mu}\left\|\mathbf{J}\right\|_* + \frac{1}{2}\left\|\mathbf{J} - (\mathbf{Z} + \mathbf{Y}/\mu)\right\|_F^2,\\ &\mathbf{Y} = \mathbf{Y} + \mu(\mathbf{J} - \mathbf{Z}),\end{aligned} \tag{29}$$

where it can be solved by singular value thresholding [50] for updating $\mathbf{J}$. The optimization procedure is summarized in Algorithm 3.

Algorithm 2: Update networks with the GD algorithm

Input: Multi-view data $\{\mathbf{X}^{(1)}, ..., \mathbf{X}^{(V)}\}$, latent representation $\mathbf{H}$, hyperparameter λ, learning rate η, dimensionality k of latent representation $\mathbf{H}$, and maximal iteration number T.
Initialization: Initialize randomly $\mathbf{W}_{(1,v)}$ and $t = 1$.
while $t < T$ *and not converged* **do**
  $v = 1$;
  for $v \leq V$ **do**
    Update $\mathbf{M}_v$ by $\mathbf{M}_v = tanh(\mathbf{W}_{(1,v)}\mathbf{H})$;
    Update $\mathbf{W}_{(2,v)}$ according to (26);
    Update $\mathbf{W}_{(1,v)}$ by $\mathbf{W}_{(1,v)} = \mathbf{W}_{(1,v)} - \eta\frac{\partial\mathcal{L}_W}{\mathbf{W}_{(1,v)}}$;
    $v = v + 1$;
  end
  Check the convergence conditions: $\sum_{v=1}^{V}\alpha_v\left\|\mathbf{X}_v - g_{\boldsymbol{\theta}_v}(\mathbf{H})\right\|_F^2 < \epsilon$.
  $t = t + 1$;
end
Output: $\{\mathbf{W}_{(1,v)}, \mathbf{W}_{(2,v)}\}_{v=1}^{V}$.

Algorithm 3: Optimization algorithm for gLMSC

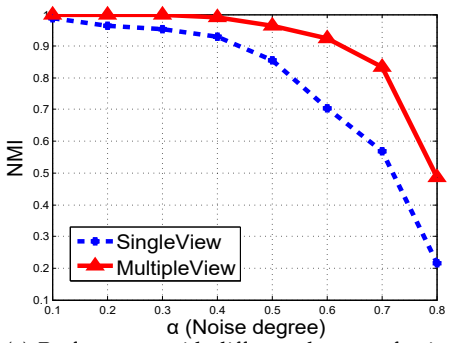
Input: Multi-view matrices: $\{\mathbf{X}^{(1)}, ..., \mathbf{X}^{(V)}\}$, hyperparameter λ and the dimension K of the latent representation $\mathbf{H}$.
Initialization: $\mu = 10^{-6}$, $\rho = 1.2$, $\epsilon = 10^{-4}$, $\max_\mu$ =10^6; randomly initialize $\mathbf{H}$ and $\mathbf{W}_{(1,v)}$.
while *not converged* **do**
  Update the networks by using Alg. 2;
  Update the latent representation $\mathbf{H}$ according to (28);
  Update the subspace representation $\mathbf{Z}$, $\mathbf{J}$ and $\mathbf{Y}$ according to (29);
  Update the parameter μ by μ = min($\rho\mu$; $\max_\mu$);
  Check the convergence conditions: $||\mathbf{J} - \mathbf{Z}||_\infty < \epsilon$.
end
Output: $\mathbf{Z}$ and $\mathbf{H}$.

3.3 Complexity and Convergence

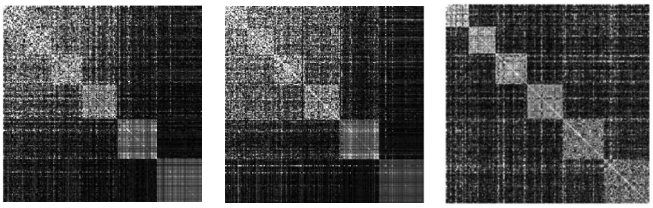
The optimization of lLMSC comprises six sub-problems. For clarification, we define k, d, and n as the dimensionality of the latent representation, the sum of the dimensionalities for multiple views, and the size of data, respectively. Then, the complexities of the six sub-problems are induced as follows. For updating $\mathbf{P}$ and $\mathbf{J}$ (the nuclear norm proximal operator), the complexities are $O(k^2d + d^3)$ and $O(n^3)$,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2018.2877660, IEEE Transactions on Pattern Analysis and Machine Intelligence

respectively. The complexity of updating $\mathbf{H}$ with Bartels-Stewart algorithm [49] is $O(k^3)$. The main computational cost of updating $\mathbf{Z}$ is the matrix inversion, and the complexity is $O(n^3)$. The complexity of updating $\mathbf{E}$ and multipliers is $O(dkn + kn^2)$ due to the matrix multiplication. Then, the overall complexity of lLMSC is $O(k^2d+d^3+k^3+n^3+dkn+kn^2)$. Since the dimension of latent representation is usually much lower than that of original views, i.e., $k \ll d$, then the complexity is basically $O(d^3 + n^3)$. For the complexity of gLMSC, the main computational cost arises from three sub-problems. For the meanings of $d_{(1,v)}$, $d_{(2,v)}$, please refer to Table. 1. The complexities are $O(d_{(1,v)}kn + d^2_{(1,v)}d_{(2,v)} + d^2_{(1,v)}n)$, $O(d_{(1,v)}kn)$, and $O(d^2_{(1,v)}k + d^3_{(1,v)})$ for updating $\mathbf{M}$, $\mathbf{W}_{(1,v)}$, and $\mathbf{W}_{(2,v)}$, respectively. For updating $\mathbf{H}$ and $\mathbf{Z}$, the complexity is $O(d_{(2,v)}d_{(1,v)}n+d^2_{(1,v)}d_{(2,v)}+d^2_{(1,v)}n+n^3+kn^2)$ and $O(n^3)$, respectively. Similarly, under the condition $d_1 = max(\{d_{(1,v)}\}_{v=1}^V)$, $d_2 = max(\{d_{(2,v)}\}_{v=1}^V)$, and $k \ll min(d_1, d_2)$, the total complexity of gLMSC is $O(d_1^3 + n^3 + d_1^2n + d_1^2d_2 + d_1d_2n)$. It is difficult to provide a general proof of the convergence for our algorithm. Fortunately, comprehensive results on both synthesized and real data empirically demonstrate that the proposed algorithm has very strong and stable convergence, even with random $\mathbf{H}$ initialization.



(a) Performance with different degrees of noise.



(b) Visualization of similarity matrices corresponding to single view (left and middle) and multiple views (right).

Fig. 3: Experiments to evaluate the robustness of multi-view and single-view methods on synthetic data.

4 Experiments

4.1 Experimental Setting

To comprehensively evaluate our model, both synthetic and real-world benchmark datasets are employed. We conduct experiments on synthetic data to test the effectiveness of using multiple views compared with a single view. We also employ datasets from diverse applications including general images, medical images, text, and community networks. Specifically, we use the following datasets. **ADNI**[1] consists of 360 samples with Magnetic Resonance (MR) and Positron Emission Tomography (PET) images, where 93 ROI-based neuroimaging features for each neuroimage (i.e., MRI or PET) are extracted. Multilingual dataset **Reuters** [54] consists of 2000 samples with 5 types of languages and the documents are represented as a bag of words using a TFIDF-based weighting scheme. **Football**[2] is a collection of 248 English Premier League football players and clubs active on Twitter. The disjoint ground truth communities correspond to the 20 clubs in the league. **Politicsie**[3] is a collection of Irish politicians and political organizations assigned to seven disjoint ground truth groups according to their affiliation. The two Twitter datasets are associated with 9 different views. **MSRCV1** [55] consists of 210 images from 7 classes. There are 6 types of features extracted: CENT, CMT, GIST, HOG, LBP, and SIFT. **BBCSport**[4] consists of documents of sports news corresponding to 5 topics, where for each document two different types of features are extracted [56]. The dataset **Animals with Attributes** [57] consists of 30475 images of 50 animals classes. We sampled 1/3 data points from each class with equal interval to generate a subset with 10158 samples.

1. http://adni.loni.usc.edu/
2. http://mlg.ucd.ie/aggregation/
3. http://mlg.ucd.ie/aggregation/
4. http://mlg.ucd.ie/datasets/

Two types of deep features (i.e., extracted with DECAF [53] and VGG19 [58]) are used. We also extract two types of deep features (extracted with DECAF and VGG19) for **Caltech-101** which contains 8677 images from 101 classes.

We conduct experiments on multiple benchmark datasets to compare the following methods:
(1) **LRR$_{\text{BestSV}}$** [10] performs subspace clustering with the low-rank constraint for each single view with the best performance reported.
(2) **RMSC** [56] recovers a shared low-rank transition probability matrix as the input to the standard Markov chain.
(3) **DiMSC** [15] enforces subspace representations of different views to be diverse to reduce redundancy and then integrates them all into an affinity matrix.
(4) **LT-MSC** [16] employs a low-rank tensor to enforce the consistence in high-order manner to make use of the complementary information of multiple views.
(5) **t-SVD-MSC** [59] imposes a new type of low-rank tensor constraint on the rotated tensor to capture the complementary information from multiple views.
(6) **DSSC** [60] proposes a deep extension of Sparse Subspace Clustering, termed Deep Sparse Subspace Clustering (DSSC). We employ PCA to reduce the number of dimensions for each view and then concatenate together all views.
(7) **MLAP** [61] performs multi-view subspace clustering by concatenating subspace representations of different views together and imposing low-rank constraint to explore the complementarity.
(8) **MSSC** [62] exploits the complementarity by using a common representation across different modalities.
(9-10) **lLMSC/gLMSC** are the proposed linear/generalized Latent Multi-View Subspace Clustering algorithms.

For clustering measures, NMI (normalized mutual information), ACC (accuracy), F-measure, and RI (Rand index)

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2018.2877660, IEEE Transactions on Pattern Analysis and Machine Intelligence

TABLE 2: Performance comparison of different clustering methods.

Datasets	Methods	NMI	ACC	F-measure	RI
ADNI	LRR_{BestSV}	6.28 ± 0.19	42.28 ± 0.21	39.90 ± 0.47	55.67 ± 0.16
	RMSC	6.81 ± 0.30	42.78 ± 0.46	38.34 ± 0.63	55.65 ± 0.12
	DiMSC	5.84 ± 0.12	39.17 ± 0.36	40.12 ± 0.33	50.88 ± 0.23
	LT-MSC	8.63 ± 0.03	42.78 ± 0.05	39.40 ± 0.13	56.57 ± 0.00
	t-SVD-MSC	4.37 ± 0.43	42.38 ± 0.59	37.76 ± 0.23	55.47 ± 0.07
	DSSC	6.98 ± 0.53	44.17 ± 0.56	39.82 ± 1.20	55.50 ± 0.49
	MLAP	**9.68 ± 0.81**	45.27 ± 0.67	39.30 ± 0.18	**56.61 ± 0.02**
	MSSC	5.89 ± 0.45	44.45 ± 0.60	38.47 ± 0.21	55.44 ± 0.02
	lLMSC	8.20 ± 0.19	**45.56 ± 0.21**	**40.78 ± 0.40**	55.50 ± 0.16
	gLMSC	**10.98 ± 0.15**	**46.67 ± 0.23**	**41.91 ± 0.20**	**57.20 ± 0.11**
Reuters	LRR_{BestSV}	20.69 ± 0.62	39.90 ± 0.31	32.55 ± 0.48	68.11 ± 0.07
	RMSC	19.00 ± 0.75	39.46 ± 1.29	31.86 ± 1.40	68.05 ± 0.92
	DiMSC	18.21 ± 0.33	40.00 ± 1.13	28.68 ± 0.39	67.49 ± 0.28
	LT-MSC	17.93 ± 1.32	36.20 ± 1.46	28.29 ± 0.95	68.16 ± 0.53
	t-SVD-MSC	**24.88 ± 0.03**	43.40 ± 0.68	33.17 ± 0.04	**69.54 ± 0.02**
	DSSC	12.86 ± 1.25	42.78 ± 2.03	35.61 ± 2.19	66.90 ± 0.78
	MLAP	17.04 ± 2.24	38.40 ± 1.63	32.15 ± 1.83	63.69 ± 0.52
	MSSC	20.56 ± 0.63	**44.50 ± 1.04**	**37.23 ± 1.12**	62.09 ± 0.36
	lLMSC	**27.99 ± 0.79**	**47.90 ± 0.64**	**40.15 ± 0.50**	**70.08 ± 0.39**
	gLMSC	23.00 ± 1.00	42.70 ± 0.99	34.76 ± 1.21	65.37 ± 0.63
Football	LRR_{BestSV}	81.07 ± 1.56	75.40 ± 2.36	66.36 ± 2.57	96.66 ± 0.15
	RMSC	84.34 ± 2.04	78.55 ± 3.84	70.97 ± 4.01	97.08 ± 0.44
	DiMSC	82.16 ± 1.45	75.40 ± 2.26	67.13 ± 1.19	96.74 ± 0.59
	LT-MSC	84.22 ± 1.17	79.03 ± 2.01	71.32 ± 1.37	97.19 ± 0.55
	t-SVD-MSC	**85.65 ± 0.73**	80.15 ± 0.88	73.04 ± 0.40	97.34 ± 0.22
	DSSC	78.16 ± 1.38	76.81 ± 1.25	48.44 ± 2.14	92.52 ± 0.63
	MLAP	85.19 ± 1.89	80.64 ± 2.36	73.35 ± 2.04	97.36 ± 0.34
	MSSC	84.27 ± 0.93	**84.65 ± 1.37**	**74.78 ± 2.16**	**97.50 ± 0.43**
	lLMSC	83.96 ± 2.08	80.24 ± 2.18	70.82 ± 1.09	97.14 ± 0.82
	gLMSC	**89.31 ± 2.22**	**86.25 ± 1.45**	**79.40 ± 1.40**	**97.97 ± 0.73**
Politicsie	LRR_{BestSV}	72.94 ± 3.37	64.94 ± 4.58	64.59 ± 3.06	85.36 ± 2.06
	RMSC	70.88 ± 3.22	63.30 ± 4.17	60.61 ± 3.38	84.09 ± 1.56
	DiMSC	76.63 ± 4.16	80.46 ± 3.21	77.57 ± 2.19	89.97 ± 1.19
	LT-MSC	68.61 ± 1.22	64.08 ± 1.56	62.69 ± 1.53	84.59 ± 0.90
	t-SVD-MSC	76.86 ± 1.55	78.86 ± 2.10	75.58 ± 1.60	89.39 ± 0.77
	DSSC	75.79 ± 3.87	70.52 ± 3.99	70.05 ± 2.50	87.69 ± 1.35
	MLAP	78.10 ± 2.01	71.26 ± 2.37	72.26 ± 1.34	88.66 ± 1.72
	MSSC	69.27 ± 2.53	66.38 ± 2.06	63.05 ± 1.49	84.86 ± 1.01
	lLMSC	**81.46 ± 0.89**	**83.33 ± 0.94**	**80.66 ± 0.69**	**91.42 ± 0.19**
	gLMSC	**78.65 ± 1.16**	**82.18 ± 1.71**	**78.42 ± 0.91**	**90.48 ± 0.22**
MSRCV1	LRR_{BestSV}	56.47 ± 2.09	66.19 ± 2.73	51.72 ± 3.56	68.34 ± 1.28
	RMSC	64.99 ± 2.21	75.00 ± 4.81	62.78 ± 2.34	89.42 ± 0.69
	DiMSC	62.87 ± 2.18	68.57 ± 3.92	57.92 ± 2.44	89.72 ± 1.10
	LT-MSC	70.04 ± 0.13	80.00 ± 0.09	68.48 ± 0.03	91.12 ± 0.00
	t-SVD-MSC	**96.03 ± 0.03**	**98.10 ± 0.01**	**96.16 ± 0.03**	**98.93 ± 0.00**
	DSSC	63.34 ± 0.24	71.01 ± 0.10	63.29 ± 0.35	86.91 ± 0.25
	MLAP	66.71 ± 0.52	72.86 ± 0.76	64.45 ± 0.38	89.98 ± 0.08
	MSSC	63.10 ± 0.16	70.99 ± 0.22	62.87 ± 0.19	86.54 ± 0.07
	lLMSC	65.34 ± 1.17	80.55 ± 1.41	65.17 ± 1.62	90.40 ± 0.20
	gLMSC	**75.25 ± 1.03**	**84.81 ± 1.27**	**73.80 ± 1.79**	**92.51 ± 0.23**
BBCSport	LRR_{BestSV}	69.02 ± 0.19	78.72 ± 0.26	76.98 ± 0.23	87.35 ± 0.13
	RMSC	60.84 ± 0.75	73.72 ± 0.37	65.51 ± 0.20	92.29 ± 0.33
	DiMSC	85.11 ± 0.13	95.10 ± 2.17	91.02 ± 0.14	95.72 ± 0.10
	LT-MSC	77.54 ± 0.46	90.26 ± 0.73	80.16 ± 0.59	90.36 ± 0.27
	t-SVD-MSC	**91.82 ± 0.08**	**97.61 ± 0.21**	**94.90 ± 0.06**	**97.57 ± 0.11**
	DSSC	72.56 ± 0.32	89.43 ± 0.13	81.19 ± 0.26	92.91 ± 0.01
	MLAP	71.23 ± 0.36	85.29 ± 0.15	73.53 ± 0.19	85.27 ± 0.02
	MSSC	69.96 ± 0.39	79.78 ± 0.92	76.13 ± 0.51	87.27 ± 0.34
	lLMSC	82.59 ± 0.81	91.07 ± 0.59	88.65 ± 0.77	94.53 ± 0.15
	gLMSC	**88.66 ± 0.46**	**96.32 ± 0.78**	**92.54 ± 0.26**	**96.49 ± 0.11**

are employed to conduct comprehensively evaluation. Note that a higher value indicates a better performance for each metric. Since there are different accuracy definitions in clustering, we specify the definition used in our experiments. Given a sample $\mathbf{x}_i$, we denote the cluster and class labels as ω_i and c_i, respectively, giving:

$$ACC = \frac{\sum_{i=1}^{N} \delta(c_i, map(\omega_i))}{n}, \tag{30}$$

where $\delta(a, b) = 1$ when $a = b$, otherwise $\delta(a, b) = 0$. $map(\omega_i)$ is the permutation map function, which maps the cluster labels into class labels. n is the number of samples. The best map can be obtained by the Kuhn-Munkres algorithm.

For our algorithm, we tune the tradeoff parameter λ from the set $\{0.01, 0.1, 1, 10, 100\}$. For simplicity, we set α_1 =...= α_V = α and tune α from $\{0.1, 0.2, \cdots, 1.0\}$ on all

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2018.2877660, IEEE Transactions on Pattern Analysis and Machine Intelligence

TABLE 3: Performance comparison of different clustering methods.

Datasets	Methods	NMI	ACC	F-measure	RI
ANIMAL	LRR_{BestSV}	34.59 ± 0.60	28.83 ± 0.33	16.99 ± 0.47	96.36 ± 0.31
	RMSC	70.46 ± 1.84	61.58 ± 4.50	54.30 ± 4.16	**97.95 ± 0.35**
	DiMSC	44.62 ± 0.89	32.61 ± 1.81	20.66 ± 1.10	96.30 ± 0.23
	LT-MSC	41.29 ± 0.40	33.65 ± 0.67	21.65 ± 0.49	96.53 ± 0.16
	t-SVD-MSC	**70.66 ± 0.19**	**63.44 ± 0.23**	**54.40 ± 0.26**	97.91 ± 0.01
	DSSC	–	–	–	–
	MLAP	69.98 ± 0.03	63.32 ± 0.06	52.61 ± 0.11	97.88 ± 0.18
	MSSC	66.93 ± 0.35	59.24 ± 0.32	50.12 ± 0.15	97.22 ± 0.02
	lLMSC	70.11 ± 0.25	59.86 ± 0.29	51.90 ± 0.64	97.86 ± 0.01
	gLMSC	**72.66 ± 0.35**	**64.47 ± 0.44**	**54.54 ± 0.37**	**97.97 ± 0.08**
CALTECH	LRR_{BestSV}	77.59 ± 1.23	52.58 ± 2.00	36.86 ± 1.82	97.48 ± 0.70
	RMSC	81.41 ± 1.57	56.02 ± 2.10	27.35 ± 2.63	97.58 ± 0.56
	DiMSC	63.72 ± 0.99	37.09 ± 1.81	25.47 ± 2.11	97.02 ± 0.34
	LT-MSC	80.38 ± 1.58	56.02 ± 1.11	39.86 ± 1.26	**97.59 ± 0.73**
	t-SVD-MSC	81.51 ± 1.40	56.60 ± 0.79	40.43 ± 1.10	97.58 ± 0.46
	DSSC	–	–	–	–
	MLAP	**82.03 ± 1.09**	**57.62 ± 1.57**	**42.30 ± 0.77**	97.57 ± 0.36
	MSSC	78.14 ± 0.45	55.90 ± 0.66	**42.11 ± 0.29**	97.02 ± 0.07
	lLMSC	76.26 ± 1.11	52.84 ± 1.30	37.72 ± 0.96	97.48 ± 0.22
	gLMSC	**81.63 ± 1.10**	**59.68 ± 0.60**	41.90 ± 0.41	**97.68 ± 0.28**

TABLE 4: Performance comparison between single view and the learned latent representation.

Datasets	Methods	NMI	ACC	F-measure	RI
ADNI	View1	10.06 ± 0.20	**48.33 ± 0.34**	40.42 ± 0.09	55.32 ± 0.15
	View2	2.33 ± 0.16	41.50 ± 0.70	38.28 ± 0.38	54.12 ± 0.06
	GCCA	1.49 ± 0.21	41.94 ± 0.37	41.43 ± 0.14	52.45 ± 0.06
	DCCA	3.93 ± 0.32	37.33 ± 0.64	41.37 ± 0.25	54.08 ± 0.13
	Latent(lLMSC)	10.21 ± 0.11	44.72 ± 0.51	**46.58 ± 0.35**	54.85 ± 0.40
	Latent(gLMSC)	**11.15 ± 0.39**	45.00 ± 0.29	46.38 ± 0.70	**56.31 ± 0.17**
Reuters	View1	19.89 ± 6.56	41.74 ± 7.00	39.26 ± 4.12	51.05 ± 6.77
	View2	16.64 ± 7.35	40.94 ± 6.65	34.19 ± 2.56	51.03 ± 3.21
	View3	21.18 ± 8.58	42.22 ± 3.85	36.05 ± 4.04	58.56 ± 5.92
	GCCA	28.18 ± 5.26	37.43 ± 4.00	33.96 ± 2.73	**69.58 ± 3.68**
	DCCA	17.40 ± 3.57	42.67 ± 5.39	36.79 ± 3.86	54.37 ± 5.11
	Latent(lLMSC)	31.21 ± 0.65	38.98 ± 2.77	39.32 ± 1.20	60.71 ± 4.02
	Latent(gLMSC)	**31.23 ± 4.82**	**42.94 ± 3.63**	**39.79 ± 2.92**	68.76 ± 4.11
Football	View1	64.13 ± 2.31	52.82 ± 2.42	29.61 ± 2.66	85.31 ± 1.38
	View2	67.21 ± 2.35	62.10 ± 2.24	38.17 ± 3.32	89.04 ± 1.07
	View8	62.65 ± 2.33	50.81 ± 2.36	25.72 ± 2.45	85.93 ± 1.03
	GCCA	39.87 ± 1.42	25.81 ± 2.07	12.25 ± 2.67	63.41 ± 1.44
	DCCA	79.56 ± 1.99	64.19 ± 2.14	54.08 ± 2.46	94.35 ± 0.73
	Latent(lLMSC)	70.61 ± 2.40	61.69 ± 3.71	44.81 ± 2.19	93.56 ± 1.17
	Latent(gLMSC)	**83.29 ± 1.95**	**70.56 ± 1.32**	**66.69 ± 1.72**	**95.56 ± 0.89**
Politicsie	View1	56.47 ± 1.86	45.11 ± 1.91	48.76 ± 1.90	77.61 ± 0.76
	View2	44.04 ± 2.13	43.97 ± 1.53	36.22 ± 2.86	68.10 ± 1.22
	View8	18.01 ± 1.69	39.37 ± 2.05	34.33 ± 3.45	62.04 ± 1.01
	GCCA	20.65 ± 2.67	52.30 ± 1.88	41.29 ± 2.09	42.60 ± 1.43
	DCCA	56.19 ± 1.07	60.06 ± 1.55	49.31 ± 2.62	78.42 ± 1.23
	Latent(lLMSC)	72.36 ± 2.17	67.58 ± 1.99	60.28 ± 2.00	83.81 ± 1.46
	Latent(gLMSC)	**74.10 ± 2.81**	**68.10 ± 2.72**	**64.86 ± 2.58**	**85.26 ± 1.20**
MSRCV1	View1	51.95 ± 3.12	54.00 ± 5.94	47.91 ± 4.30	83.80 ± 0.41
	View3	62.03 ± 0.72	70.42 ± 0.51	58.95 ± 0.85	88.39 ± 0.13
	View4	53.45 ± 1.41	60.63 ± 1.69	49.79 ± 2.13	85.57 ± 1.46
	GCCA	62.51 ± 2.12	69.05 ± 1.57	58.48 ± 1.64	86.89 ± 0.85
	DCCA	41.20 ± 0.16	54.29 ± 0.70	38.32 ± 0.38	81.63 ± 0.06
	Latent(lLMSC)	71.67 ± 1.31	80.76 ± 1.27	68.92 ± 1.76	90.64 ± 0.96
	Latent(gLMSC)	**72.99 ± 1.36**	**82.36 ± 1.41**	**70.15 ± 1.66**	**91.37 ± 0.55**
BBCSport	View1	59.64 ± 17.04	64.17 ± 15.26	62.43 ± 13.46	73.73 ± 15.41
	View2	23.17 ± 16.86	44.85 ± 8.47	44.47 ± 7.80	42.69 ± 12.36
	GCCA	59.59 ± 7.78	75.92 ± 3.89	70.50 ± 5.57	84.92 ± 6.65
	DCCA	35.52 ± 12.63	64.52 ± 6.98	48.57 ± 6.63	76.81 ± 11.54
	Latent(lLMSC)	62.18 ± 12.45	66.66 ± 12.15	63.96 ± 12.18	76.72 ± 12.30
	Latent(gLMSC)	**76.13 ± 13.21**	**77.21 ± 13.68**	**73.32 ± 12.98**	**87.02 ± 11.55**

datasets. The network parameter γ (for regularization) is fixed to 0.001. For the baseline approaches, we tune all the parameters to report their best performances according to the authors. The dimensionality of the latent representation

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2018.2877660, IEEE Transactions on Pattern Analysis and Machine Intelligence

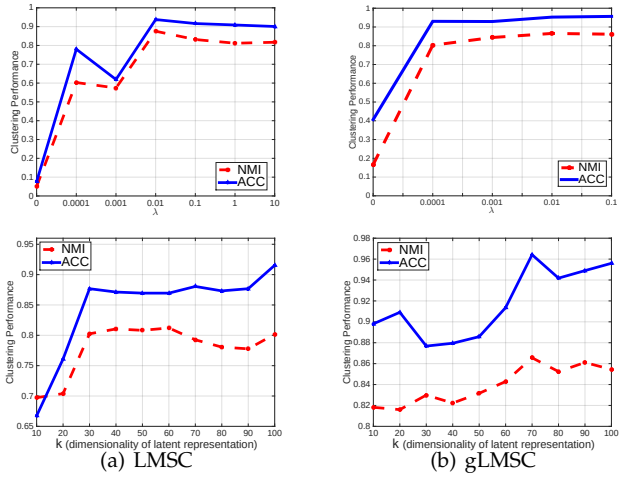
is relatively robust hence we set $k = 100$ for all datasets, which results in promising performance. Due to randomness, we run all algorithms 30 times and report the mean values and standard deviations.

4.2 Results on Synthetic Data

Firstly, we evaluate our algorithm in exploring multiple views on synthetic data. In our experiment, the randomly generated matrices are produced by independently sampling elements from a uniform distribution within the range $[0, 1]$. The synthetic data are from 6 subspaces/clusters with the sample numbers corresponding to these subspaces being $\{25, 30, 35, 40, 45, 50\}$, respectively. First, the latent representation matrix $\mathbf{H} \in \mathbb{R}^{k\times n}$ is generated randomly, with the number of dimensions $k = 90$ and the number of data points $n = 225$. These subspaces have 10, 12, 14, 16, 18, and 20 disjoint features, respectively. Then, based on the latent representation matrix $\mathbf{H}$ two views are produced with $\mathbf{X}^{(v)} = \mathbf{P}^{(v)}\mathbf{H} + \mathbf{E}^{(v)}$. Two types of noise are considered for $\mathbf{E}^{(v)}$: $\mathbf{E}^{(v)} = \mathbf{E}_s^{(v)} + \alpha\mathbf{E}_g^{(v)}$, where $\mathbf{E}_g^{(v)}$ and $\mathbf{E}_s^{(v)}$ are global and sample-specific noises, respectively. For $\mathbf{E}_s^{(v)}$, we randomly select a subset of columns (20 in our experiments), and set the other columns to zeros. For $\mathbf{E}_g^{(v)}$, we multiply it with a scalar $0 < \alpha < 1$ to tune the noise degree. In Fig. 3(a), benefiting from the complementarity of multiple views our approach obtains much better performance compared to that using a single view of features with different degrees of noise. In Fig. 3(b), we provide a visualization of affinity matrices for both single view and multiple views with $\alpha = 0.5$. The affinity matrix of multiple views reveals the underlying cluster structure much better than using a single view.

4.3 Results on Real Datasets

We next test our model on diverse real-world applications including medical image/general image clustering, community detection, and text clustering. Tables 2-3 present the clustering results of different clustering approaches. From Tables 2-3, the following observations can be made: (1) overall, lLMSC achieves very competitive and stable performance compared to most baselines. Taking the datasets Reuters and Politicsie for example, lLMSC outperforms all the traditional methods; (2) by exploring the general correlation with a neural network, gLMSC significantly improves lLMSC on 6 out of 8 datasets. For example, the NMI improvements of gLMSC over lLMSC are about 5.3% and 6.1% on Football and BBCSport, respectively; The potential reasons why gLMSC does not always outperform lLMSC may be: first, for some cases (e.g., Reuters: each document is associated with multiple types of languages), the linear model is enough to model the correlations among different views; second, although gLMSC is more general than lLMSC, there is no global optimal solution guaranteed for both gLMSC and lLMSC; (3) although the performance of our method is not always top, the performance is rather robust across different datasets, while the performance of some methods is very unpredictable and variable. For example, MLAP achieves the promising performance on ADNI and CALTECH. However, on Reuters and BBCSport, MLAP



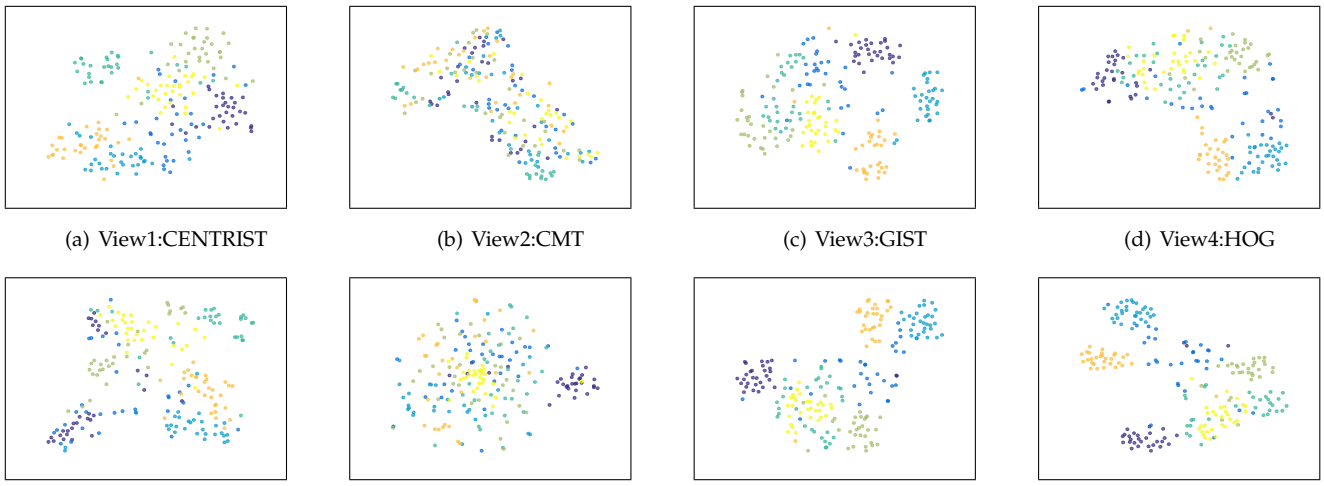
(a) LMSC (b) gLMSC

Fig. 5: Results of our method when using different parameters: λ (top row) and k (bottom row).

does not perform very well; (4) we also compared our algorithm with Deep Sparse Subspace Clustering (DSSC) [60] and t-SVD-MSC [59], while the performance of gLMSC is consistently better than them. For example, the performance improvements over t-SVD-MSC are about 6.1% and 3.3% on the two community datasets, i.e., Football and Politicsie, in terms of accuracy. The method t-SVD-MSC emphasizes the consistence over different views due to the low-rank constraint, while it is a challenge for it to balance the consistence and the complementarity. While our algorithm can handle this issue due to the flexible encoding of the intrinsic information from different views; (5) the performance of single-view methods with the best view is generally worse than multi-view methods, confirming that it is useful to incorporate multiple views.

Is the latent representation good? To investigate the improvement gains of our approach, we compare the latent representation of our algorithm, Generalized Canonical Correlational Analysis (GCCA) [63], Deep Canonical Correlation Analysis (DCCA) [43] and features of each single view by conducting k-means over them. As shown in Table 4, the performance using our latent representation is generally better than those using single-view features. This is empirical proof of the added value of the latent representation compared to the original features. Although nonlinear correlations are involved in the CCA-based algorithms, i.e., DCCA and GCCA, the performances are not promising compared with ours. One of the main possible reason is that the representation learning and clustering are separated for these algorithms, thus the learned representations are not guaranteed to be suitable for clustering. Furthermore, we visualize the features of each view and the latent representation using t-Distributed Stochastic Neighbor Embedding (t-SNE) [64] on MSRCV1. As shown in Fig. 4, the visualization is consistent with the clustering results shown in Table 4. Specifically, Fig. 4(c)(corresponding to view3) and (d) (corresponding to view4) more clearly reveal the underlying cluster structure, and the corresponding clustering performances are also much better than other views. Fig. 4(g) and (h) (corresponding to latent representation) further validate the advantage of our model, since the clusters are more

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2018.2877660, IEEE Transactions on Pattern Analysis and Machine Intelligence



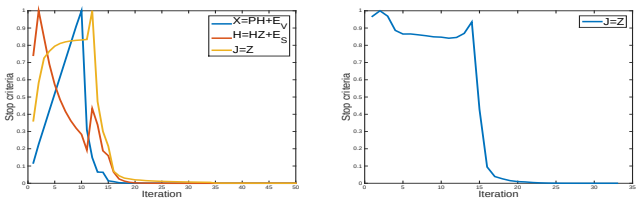
(a) View1:CENTRIST (b) View2:CMT (c) View3:GIST (d) View4:HOG

(e) View5:LBP (f) View6:SIFT (g) Latent Representation (lLMSC) (h) Latent Representation (gLMSC)

Fig. 4: Visualization of different views and latent representation with t-SNE on the MSRCV1 dataset.

compact and separable than those of the original features corresponding to different views.

Parameter tuning and convergence. Fig. 5 shows the results of our method using different parameters (taking BBCSport as an example). The performances of our linear and generalized models are both relatively stable and promising, as shown by the results achieved by setting λ in a relatively large range. The bottom of Fig. 5 presents model performance with respect to dimensionality (k) of the latent representation. Promising performance can be expected with relatively low dimensionality. Moreover, while gLMSC needs a latent representation of higher dimensionality than that of lLMSC, its performance is generally better because the more general correlation is addressed. Fig. 6 empirically shows that our algorithms converge within a small number of iterations.



(a) Linear LMSC (b) Generalized LMSC

Fig. 6: Convergence of our method. For a better view, the plots are normalized into the range [0, 1].

5 Conclusions and Discussion

Here we introduce the latent representation into multi-view subspace clustering. Our model effectively encodes complementarity in multiple views for subspace clustering under the assumption - each single feature view originates from one comprehensive latent representation. This is essentially different from existing multi-view subspace clustering approaches that perform self-representation directly within the single view or simply project each view of feature to a common space. The latent representation and the self-representation-based clustering complement each other. More importantly, by using a neural network-based approach to learn non-linear mappings, our model can handle more general correlations between the latent representation and each feature view. Experiments on both synthetic and benchmark datasets verify the clear advantages of the learned latent representation for multi-view subspace clustering compared to the state-of-the-art multi-view clustering methods.

Our model is able to flexibly explore the complementarity among multiple views for subspace clustering. However, there are several issues that require further clarifications and possible future investigations. Firstly, since graph (of the size $n \times n$) is involved for existing subspace clustering methods which leads to computational cost matrix operations. The time complexities of these subspace based clustering methods are generally in the same order. Specifically, SVD decomposition and matrix inversion are employed in our method which makes our algorithm with high computational cost. In the future, sampling technique and binary representations [65] will be considered to accelerate the clustering speed. Second, the quality differences for different views are not considered. The performance could be degraded, when low-quality views are more dominating.

References

[1] L. Parsons, E. Haque, and H. Liu, "Subspace clustering for high dimensional data: a review," *Acm Sigkdd Explorations Newsletter*, vol. 6, no. 1, pp. 90–105, 2004.

[2] R. Vidal, "Subspace clustering," *IEEE Signal Processing Magazine*, vol. 28, no. 2, pp. 52–68, 2011.

[3] P. S. Bradley and O. L. Mangasarian, "K-plane clustering," *Journal of Global Optimization*, vol. 16, no. 1, pp. 23–32, 2000.

[4] L. Lu and R. Vidal, "Combined central and subspace clustering for computer vision applications," in *ICML*. ACM, 2006, pp. 593–600.

[5] R. Vidal, Y. Ma, and S. Sastry, "Generalized principal component analysis (gpca)," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 12, pp. 1945–1959, 2005.

[6] J. P. Costeira and T. Kanade, "A multibody factorization method for independently moving objects," *International Journal of Computer Vision*, vol. 29, no. 3, pp. 159–179, 1998.

[7] A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," in *NIPS*, 2002, pp. 849–856.

[8] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888–905, 2000.

[9] E. Elhamifar and R. Vidal, "Sparse subspace clustering: Algorithm, theory, and applications," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 11, pp. 2765–2781, 2013.

[10] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, and Y. Ma, "Robust recovery of subspace structures by low-rank representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 1, pp. 171–184, 2013.

[11] H. Hu, Z. Lin, J. Feng, and J. Zhou, "Smooth representation clustering," in *CVPR*, 2014, pp. 3834–3841.

[12] J. Feng, Z. Lin, H. Xu, and S. Yan, "Robust subspace segmentation with block-diagonal prior," in *CVPR*, 2014, pp. 3818–3825.

[13] G. Liu, Z. Lin, and Y. Yu, "Robust subspace segmentation by low-rank representation," in *ICML*, 2010, pp. 663–670.

[14] M. Yin, Y. Guo, J. Gao, Z. He, and S. Xie, "Kernel sparse subspace clustering on symmetric positive definite manifolds," in *CVPR*, 2016, pp. 5157–5164.

[15] X. Cao, C. Zhang, H. Fu, S. Liu, and H. Zhang, "Diversity-induced multi-view subspace clustering," in *CVPR*, 2015, pp. 586–594.

[16] C. Zhang, H. Fu, S. Liu, G. Liu, and X. Cao, "Low-rank tensor constrained multiview subspace clustering," in *ICCV*, 2015, pp. 1582–1590.

[17] H. Gao, F. Nie, X. Li, and H. Huang, "Multi-view subspace clustering," in *ICCV*, 2015, pp. 4238–4246.

[18] Y. Guo, "Convex subspace representation learning from multi-view data," in *AAAI*, 2013.

[19] M. White, X. Zhang, D. Schuurmans, and Y.-l. Yu, "Convex multi-view subspace learning," in *NIPS*, 2012, pp. 1673–1681.

[20] C. Zhang, Q. Hu, H. Fu, P. Zhu, and X. Cao, "Latent multi-view subspace clustering," in *CVPR*, 2017, pp. 4279–4287.

[21] C. Xu, D. Tao, and C. Xu, "Large-margin multi-viewinformation bottleneck," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 8, pp. 1559–1572, 2014.

[22] W. Wang, R. Arora, K. Livescu, and J. Bilmes, "On deep multi-view representation learning," in *ICML*, 2015, pp. 1083–1092.

[23] R. Arora and K. Livescu, "Multi-view learning with supervision for transformed bottleneck features," in *ICASSP*, 2014, pp. 2499–2503.

[24] Y. Luo, D. Tao, K. Ramamohanarao, C. Xu, and Y. Wen, "Tensor canonical correlation analysis for multi-view dimension reduction," *IEEE Transactions on Knowledge and Data Engineering*, vol. 27, no. 11, pp. 3111–3124, 2015.

[25] J. He and R. Lawrence, "A graph-based framework for multi-task multi-view learning," in *ICML*, 2011, pp. 25–32.

[26] V. R. de Sa, "Spectral clustering with two views," in *ICML workshop on learning with multiple views*, 2005, pp. 20–27.

[27] J. Liu, C. Wang, J. Gao, and J. Han, "Multi-view clustering via joint nonnegative matrix factorization," in *SDM*, vol. 13. SIAM, 2013, pp. 252–260.

[28] W. Tang, Z. Lu, and I. S. Dhillon, "Clustering with multiple graphs," in *ICDM*, 2009, pp. 1016–1021.

[29] A. Kumar, P. Rai, and H. Daume, "Co-regularized multi-view spectral clustering," in *NIPS*, 2011, pp. 1413–1421.

[30] A. Kumar and H. Daumé, "A co-training approach for multi-view spectral clustering," in *ICML*, 2011, pp. 393–400.

[31] X. Cai, F. Nie, and H. Huang, "Multi-view k-means clustering on big data," in *IJCAI*, 2013, pp. 2598–2604.

[32] C. Cortes, M. Mohri, and A. Rostamizadeh, "Learning non-linear combinations of kernels," in *NIPS*, 2009, pp. 396–404.

[33] G. Tzortzis and A. Likas, "Kernel-based weighted multi-view clustering," in *ICDM*, 2012, pp. 675–684.

[34] C. Zhang, H. Fu, Q. Hu, P. Zhu, and X. Cao, "Flexible multi-view dimensionality co-reduction," *IEEE Transactions on Image Processing*, vol. 26, no. 2, pp. 648–659, 2017.

[35] J. Tang, X. Hu, H. Gao, and H. Liu, "Unsupervised feature selection for multi-view data in social media," in *SDM*, 2013, pp. 270–278.

[36] C. K, K. S. M, and L. K, "Multi-view clustering via canonical correlation analysis," in *ICML*, 2009, pp. 129–136.

[37] K. Chaudhuri, S. M. Kakade, K. Livescu, and K. Sridharan, "Multi-view clustering via canonical correlation analysis," in *ICML*, 2009, pp. 129–136.

[38] V. M. Patel, H. Van Nguyen, and R. Vidal, "Latent space sparse subspace clustering," in *ICCV*, 2013, pp. 225–232.

[39] G. Liu and S. Yan, "Latent low-rank representation for subspace segmentation and feature extraction," in *ICCV*, 2011, pp. 1615–1622.

[40] M. Abavisani and V. Patel, "Domain adaptive subspace clustering." in *BMVC*, 2016.

[41] A. Sharma, A. Kumar, H. Daume, and D. W. Jacobs, "Generalized multiview analysis: A discriminative latent space," in *CVPR*, 2012, pp. 2160–2167.

[42] P. Rastogi, B. Van Durme, and R. Arora, "Multiview lsa: Representation learning via generalized cca," in *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2015, pp. 556–566.

[43] G. Andrew, R. Arora, J. Bilmes, and K. Livescu, "Deep canonical correlation analysis," in *ICML*, 2013, pp. 1247–1255.

[44] X. Peng, S. Xiao, J. Feng, W.-Y. Yau, and Z. Yi, "Deep subspace clustering with sparsity prior." in *IJCAI*, 2016, pp. 1925–1931.

[45] P. Ji, T. Zhang, H. Li, M. Salzmann, and I. Reid, "Deep subspace clustering networks," in *NIPS*, 2017.

[46] C. Lang, G. Liu, J. Yu, and S. Yan, "Saliency detection by multitask sparsity pursuit," *IEEE Transactions on Image Processing*, vol. 21, no. 3, pp. 1327–1338, 2012.

[47] Z. Lin, R. Liu, and Z. Su, "Linearized alternating direction method with adaptive penalty for low-rank representation," in *NIPS*, 2011, pp. 612–620.

[48] G. Wahba, "A least squares estimate of satellite attitude," *SIAM review*, vol. 7, no. 3, pp. 409–409, 1965.

[49] R. H. Bartels and G. Stewart, "Solution of the matrix equation AX+ XB= C," *Communications of the ACM*, vol. 15, no. 9, pp. 820–826, 1972.

[50] J.-F. Cai, E. J. Candès, and Z. Shen, "A singular value thresholding algorithm for matrix completion," *SIAM Journal on Optimization*, vol. 20, no. 4, pp. 1956–1982, 2010.

[51] S. Gu, L. Zhang, W. Zuo, and X. Feng, "Projective dictionary pair learning for pattern classification," in *NIPS*, 2014.

[52] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath *et al.*, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82–97, 2012.

[53] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *NIPS*, 2012, pp. 1097–1105.

[54] M.-R. Amini, N. Usunier, and C. Goutte, "Learning from multiple partially observed views - an application to multilingual text categorization," in *NIPS*, 2009.

[55] J. Xu, J. Han, and F. Nie, "Discriminatively embedded k-means for multi-view clustering," in *CVPR*, 2016, pp. 5356–5364.

[56] R. Xia, Y. Pan, L. Du, and J. Yin, "Robust multi-view spectral clustering via low-rank and sparse decomposition," in *AAAI*, 2014, pp. 2149–2155.

[57] C. H. Lampert, H. Nickisch, and S. Harmeling, "Attribute-based classification for zero-shot visual object categorization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 3, pp. 453–465, 2014.

[58] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[59] Y. Xie, D. Tao, W. Zhang, L. Zhang, Y. Liu, and Y. Qu, "On unifying multi-view self-representations for clustering by tensor multi-rank minimization," *arXiv preprint arXiv:1610.07126*, 2016.

[60] X. Peng, J. Feng, S. Xiao, J. Lu, Z. Yi, and S. Yan, "Deep sparse subspace clustering," *arXiv preprint arXiv:1709.08374*, 2017.

[61] B. Cheng, G. Liu, J. Wang, Z. Huang, and S. Yan, "Multi-task low-rank affinity pursuit for image segmentation," in *ICCV*, 2011, pp. 2439–2446.

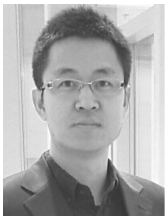
[62] M. Abavisani and V. M. Patel, "Multimodal sparse and low-rank subspace clustering," *Information Fusion*, vol. 39, pp. 168–177, 2018.

[63] J. R. Kettenring, "Canonical analysis of several sets of variables," *Biometrika*, vol. 58, no. 3, pp. 433–451, 1971.

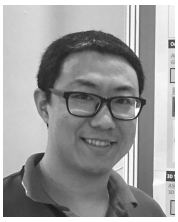
[64] L. v. d. Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of Machine Learning Research*, vol. 9, no. Nov, pp. 2579–2605, 2008.

[65] Z. Zhang, L. Liu, F. Shen, H. T. Shen, and L. Shao, "Binary multi-view clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018.

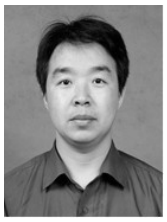
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2018.2877660, IEEE Transactions on Pattern Analysis and Machine Intelligence



Changqing Zhang received his B.S. and M.S. degrees from the College of Computer Science, Sichuan University, Chengdu, China, in 2005 and 2008, respectively, and the Ph.D. degree in Computer Science from Tianjin University, China, in 2016. He is an assistant professor at the School of Computer Science and Technology, Tianjin University. His current research interests include machine learning, computer vision and medical image analysis.



Huazhu Fu Huazhu Fu received the B.S. degree in Mathematical Sciences from Nankai University in 2006, the M.E. degree in Mechatronics Engineering from Tianjin University of Technology in 2010, and the Ph.D. degree in Computer Science from Tianjin University, China, in 2013. He was a research fellow with Nanyang Technological University (NTU), Singapore, for two years. Currently, he is a Research Scientist with Institute for Infocomm Research, at Agency for Science, Technology and Research, Singapore. His research interests include computer vision, image processing, and medical image analysis. He is the Associate Editor of BMC Medical Imaging.



Qinghua Hu (SM'13) received the B.S., M.S., and Ph.D. degrees from the Harbin Institute of Technology, Harbin, China, in 1999, 2002, and 2008, respectively. He was a Post-Doctoral Fellow with the Department of Computing, Hong Kong Polytechnic University, Hong Kong, from 2009 to 2011. He is currently a Full Professor there. He has authored over 150 journal and conference papers in the areas of granular computing-based machine learning, reasoning with uncertainty, pattern recognition, and fault diagnosis. His current research interests include multi-modality learning, metric learning, uncertainty modeling and reasoning with fuzzy sets, rough sets and probability theory. Prof. Hu was the Program Committee Co-Chair of the International Conference on Rough Sets and Current Trends in Computing in 2010, the Chinese Rough Set and Soft Computing Society in 2012 and 2014, and the International Conference on Rough Sets and Knowledge Technology and the International Conference on Machine Learning and Cybernetics in 2014, and the General Co-Chair of IJCRS 2015. Now he is the PC Co-chairs of CCML 2017 and CCCV 2017.

Xiaochun Cao (SM'14) received the B.E. and M.E. degrees in computer science from Beihang University, Beijing, China, and the Ph.D. degree in computer science from the University of Central Florida, Orlando, FL, USA. He has been a Professor with the Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China, since 2012. After graduation, he spent about three years at ObjectVideo Inc. as a Research Scientist. From 2008 to 2012, he was a Professor with Tianjin University, Tianjin, China. He has authored and coauthored more than 120 journal and conference papers. Prof. Cao is a Fellow of the IET. He is on the Editorial Board of the IEEE TRANSACTIONS OF IMAGE PROCESSING. His dissertation was nominated for the University of Central Floridas university-level Outstanding Dissertation Award. In 2004 and 2010, he was the recipient of the Piero Zamperoni Best Student Paper Award at the International Conference on Pattern Recognition.

Yuan Xie (M'12) received the Ph.D. degree in Pattern Recognition and Intelligent Systems from the Institute of Automation, Chinese Academy of Sciences (CAS), in 2013. He is currently an associated professor with the Research Center of Precision Sensing and Control, Institute of Automation, CAS. His research interests include image processing, computer vision, machine learning and pattern recognition. He has published around 30 papers in major international journals including the IJCV, IEEE TIP, TNNLS, TCYB, TCSVT, TGRS, TMM, etc. He also has served as a reviewer for more than 15 journals and conferences. Dr. Xie received the Hong Kong Scholar Award from the Society of Hong Kong Scholars and the China National Postdoctoral Council in 2014.

Dacheng Tao (F'15) is Professor of Computer Science with the Centre for Quantum Computation & Intelligent Systems, and the Faculty of Engineering and Information Technology in the University of Technology Sydney. He mainly applies statistics and mathematics to data analytics problems and his research interests spread across computer vision, data science, image processing, machine learning, and video surveillance. His research results have expounded in one monograph and 200+ publications at prestigious journals and prominent conferences, such as IEEE T-PAMI, T-NNLS, T-IP, JMLR, IJCV, NIPS, ICML, CVPR, ICCV, ECCV, AISTATS, ICDM; and ACM SIGKDD, with several best paper awards, such as the best theory/algorithm paper runner up award in IEEE ICDM'07, the best student paper award in IEEE ICDM'13, and the 2014 ICDM 10-year highest-impact paper award. He received the 2015 Australian Scopus-Eureka Prize, the 2015 ACS Gold Disruptor Award and the 2015 UTS Vice-Chancellor's Medal for Exceptional Research. He is a Fellow of the IEEE, OSA, IAPR and SPIE.

Dong Xu (F'17) received the B.E. and Ph.D. degrees from University of Science and Technology of China, in 2001 and 2005, respectively. While pursuing his Ph.D., he was with Microsoft Research Asia, Beijing, China, and the Chinese University of Hong Kong, Shatin, Hong Kong, for more than two years. He was a PostDoctoral Research Scientist with Columbia University, New York, NY, for one year. He worked as a faculty member with Nanyang Technological University, Singapore. Currently, he is a Professor and Chair in Computer Engineering with the School of Electrical and Information Engineering, the University of Sydney, Australia. His current research interests include computer vision, statistical learning, and multimedia content analysis. Dr. Xu was the co-author of a paper that won the Best Student Paper Award in the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR) in 2010, and a paper that won the Prize Paper Award in IEEE Transactions on Multimedia (TMM) in 2014.