Reciprocal Multi-Layer Subspace Learning for Multi-View Clustering

Ruihuang Li[1] Changqing Zhang[1*] Huazhu Fu[2] Xi Peng[3] Tianyi Zhou[4] Qinghua Hu[1]
[1]Tianjin University [2]Inception Institute of Artificial Intelligence [3]Sichuan University
[4]Institute of High Performance Computing, A*STAR
{liruihuang, zhangchangqing, huqinghua}@tju.edu.cn
{huazhufu, pengx.gm, joey.tianyi.zhou}@gmail.com

Abstract

Multi-view clustering is a long-standing important research topic, however, remains challenging when handling high-dimensional data and simultaneously exploring the consistency and complementarity of different views. In this work, we present a novel Reciprocal Multi-layer Subspace Learning (RMSL) algorithm for multi-view clustering, which is composed of two main components: Hierarchical Self-Representative Layers (HSRL), and Backward Encoding Networks (BEN). Specifically, HSRL constructs reciprocal multi-layer subspace representations linked with a latent representation to hierarchically recover the underlying low-dimensional subspaces in which the high-dimensional data lie; BEN explores complex relationships among different views and implicitly enforces the subspaces of all views to be consistent with each other and more separable. The latent representation flexibly encodes complementary information from multiple views and depicts data more comprehensively. Our model can be efficiently optimized by an alternating optimization scheme. Extensive experiments on benchmark datasets show the superiority of RMSL over other state-of-the-art clustering methods.

1. Introduction

Multi-view clustering, which aims to obtain a consensus partition of data across multiple views, has become a fundamental technique in the computer vision and machine learning communities. It is common in many practical applications that data are described using high-dimensional and highly heterogeneous features from multiple views. For example, one image can be represented by different descriptors such as Gabor [16], SIFT [20], and HOG [8], *etc*. Compared to single-view approaches, multi-view clustering can access to more comprehensive characteristics and structural information hidden in the data. However, most of conventional methods [4, 7, 32] directly project multiple raw features into a common space, while neglecting the high-dimensionality of data and large imbalances between different views, which will degrade the clustering performance.

Under the assumption that high-dimensional data can be well characterized by low-dimensional subspaces, subspace clustering aims to recover the underlying subspace structure of data. The effectiveness and robustness of existing self-representation-based subspace clustering methods [10, 19, 12, 21] have been validated. The key of these methods is to find an affinity matrix, each entry of which reveals the degree of similarity of two samples.

Recently, several multi-view subspace clustering methods have been proposed [6, 31, 32, 28, 22], which can be roughly divided into two main groups. The first category [6, 31] conducts self-representation within each individual view to learn an affinity matrix. By combining all view-specific affinity matrices together, an comprehensive similarity matrix which reflects intrinsic relationships among data is resulted. Although these methods have achieved promising performances, there are still some limitations: first, these methods reconstruct data within each single view, thus can not well extract comprehensive information; second, they focus on exploiting linear subspaces of data, while many real-world datasets are not necessarily subject to linear subspaces. The second category [32] aims to search for a latent representation shared by different views and then conducts self-representation on it. Despite the comprehensiveness of latent representation, these approaches can not explore the consistency of different views. In addition, these methods integrate multiple views in the raw-feature level, thus they are easily affected by the high-dimensionality of original features and possible noise.

To address above limitations, we propose the Reciprocal Multi-Layer Subspace Learning (RMSL) algorithm to cluster data from multiple sources. There is a basic assumption for multi-view clustering problem that different views

*Corresponding Author

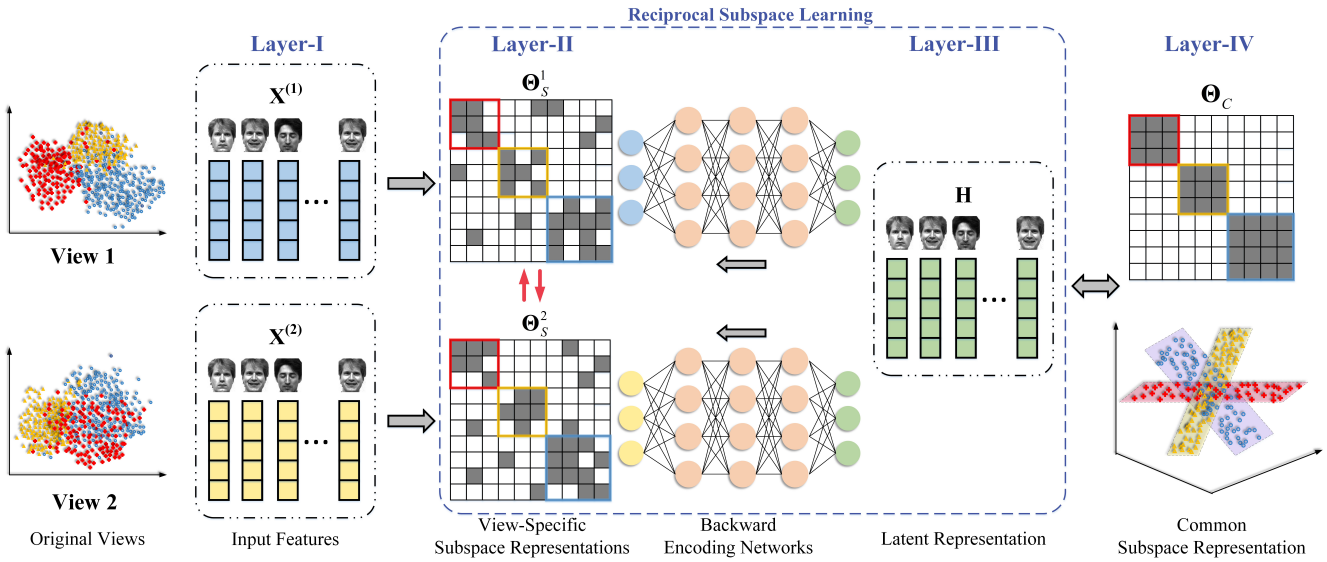

Figure 1. Illustration of the Reciprocal Multi-Layer Subspace Learning (RMSL) for multi-view clustering. We simultaneously construct view-specific $\{\mathbf{\Theta}_S^v\}_{v=1}^V$ and common $\mathbf{\Theta}_C$ subspace representations to reciprocally recover the subspace structure of data; the latent representation $\mathbf{H}$ is learned by enforcing it to be similar to different view-specific subspace representations through BEN, which implicitly drives subspaces of different views to be consistent with each other.

would share a common underlying cluster structure [15], thus we co-regularize view-specific subspace representations into a consensus one to enhance the structural consistency of different views. As shown in Figure 1, we construct reciprocal multi-layer subspace representations linked with the latent representation $\mathbf{H}$ to hierarchically recover the underlying cluster structure of data. Specifically, multi-layer subspace representations reciprocally improve each other in a joint framework; BEN reconstructs view-specific self-representations from the common representation $\mathbf{H}$, so that $\mathbf{H}$ will flexibly integrate comprehensive information from multiple views and reflect the intrinsic relationships among data points. Note that, instead of integrating multiple views in the raw-feature level like LMSC [32], we encode multiple view-specific subspace representations $\{\mathbf{\Theta}_S^v\}_{v=1}^V$ into the latent representation $\mathbf{H}$ through BEN, which is of vital importance because subspace representations can reflect underlying cluster structures of data. The contributions of this paper include:

- We propose the Reciprocal Multi-Layer Subspace Learning (RMSL) method, which constructs reciprocal multi-layer subspace representations linked with a latent representation to hierarchically identify the underlying cluster structure of high-dimensional data.
- Based on reconstruction, we learn the latent representation by enforcing it to be close to different view-specific representations, which implicitly co-regularizes subspace structures of all views to be consistent with each other.
- With the introduction of neural networks, more general relationships among different views can be explored, and the latent representation will flexibly encode complementary information from multiple views.
- Our model is optimized by alternating optimization algorithm and shows superior performances on real-world datasets in comparison with other state-of-the-art methods.

2. Related Work

Subspace Clustering. Self-representation-based subspace clustering is quite effective for high-dimensional data. Given a set of data points $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N]$ drawn from multiple subspaces, each one can be expressed as a linear combination of all the data points, *i.e.*, $\mathbf{X} = \mathbf{XZ}$, where $\mathbf{Z}$ is the learned self-representative coefficient matrix. The underlying subspace structure can be revealed by optimizing the following objective function:

$$\min_{\mathbf{Z}} \mathcal{L}(\mathbf{X}; \mathbf{Z}) + \beta \mathcal{R}(\mathbf{Z}), \tag{1}$$

where $\mathcal{L}(\cdot\,;\cdot)$ and $\mathcal{R}(\cdot)$ are the self-representation term and the regularizer on $\mathbf{Z}$, respectively. Then the similarity matrix $\mathbf{S}$ is further obtained by $\mathbf{S} = |\mathbf{Z}| + |\mathbf{Z}^T|$ for spectral clustering. Existing methods mainly differ in the choices of norms for these two terms as summarized in Table 1.

Multi-View Clustering. Multi-view clustering has inspired a surge of research interest in machine learning. Kumar *et al*. imposed co-regularized strategy on spectral clustering [15]. Xia *et al*. obtained a shared low-rank transition probability matrix as an input to the Markov chain for spectral clustering [29]. Tao *et al*. conducted multi-view clustering in ensemble clustering way [25], which constructs a consensus partition of data across different views based on all view-specific Basic partitions (BP). Nonnegative matrix

factorization-based methods decompose each feature matrix into a centroid one and a cluster assignment to preserve the local information. For example, Zhao *et al*. combined deep matrix factorization into the multi-view clustering framework to search for a factorization associated with the common partition of data [34]. Based on multiple kernel learning, Tzortzis *et al*. integrated heterogeneous features represented in terms of kernel matrices [26]. For large-scale data, Zhang *et al*. presented a Binary Multi-View Clustering (BMVC) framework [33], which significantly reduces the computation and memory footprint, while obtaining superior performance.

There are two main categories for Multi-view Subspace Clustering (MSC) methods. One category conducts self-representation within each view [6, 31] and simultaneously explores correlations among different views. Diversity-induced Multi-view Subspace Clustering (DiMSC) [6] proposes to enhance the complementarity of different subspace representations by reducing redundancy; Low-rank Tensor Constrained Multi-view Subspace Clustering (LT-MSC) [31] models inter-view high-order correlations using tensor. Let $\mathbf{X}^v$ and $\mathbf{Z}^v$ denote feature matrix and subspace representation corresponding to the vth view, respectively, then we obtain the following general formulation:

$$\min_{\{\mathbf{Z}^v\}_{v=1}^V} \mathcal{L}(\{\mathbf{X}^v\}_{v=1}^V; \{\mathbf{Z}^v\}_{v=1}^V) + \lambda \mathcal{R}(\{\mathbf{Z}^v\}_{v=1}^V), \quad (2)$$

where $\mathcal{L}(\cdot\,;\cdot)$ and $\mathcal{R}(\cdot)$ are the loss function for data reconstruction and the regularizer on $\mathbf{Z}^v$, respectively.

The second category conducts subspace representation based on the common latent representation rather than original features. Latent Multi-view Subspace Clustering (LMSC) [32] explores complementary information from different views and simultaneously constructs a latent representation. The objective function can be written as:

$$\min_{\mathbf{H},\mathbf{\Theta},\mathbf{Z}} \mathcal{L}_1(\{\mathbf{X}^v\}_{v=1}^V, \mathbf{H}; \mathbf{\Theta}) + \lambda_1 \mathcal{L}_2(\mathbf{H}; \mathbf{Z}) + \lambda_2 \mathcal{R}(\mathbf{Z}), \quad (3)$$

where $\mathcal{L}_1(\cdot\,;\cdot)$ and $\mathcal{L}_2(\cdot\,;\cdot)$ represent loss functions for multi-view data reconstruction and subspace representation, respectively. $\mathbf{\Theta}$ is the parameter to learn the latent representation.

Multi-View Representation Learning. The growing amount of data collected from multiple information sources presents an opportunity to learn better representations. There are two main training criteria that have been applied for recently proposed Deep Neural Networks-based multi-view representation learning methods. One is based on auto-encoder [23], which learns a shared representation between modalities for better reconstructing inputs; the other is based on Canonical Correlation Analysis (CCA) [11], which projects different views into a common space by maximizing their correlations, such as Deep Canonical Correlation Analysis (DCCA) [2]. In addition, Wang *et al*. combined the criteria of CCA and auto-encoder, and proposed the Deep Canonically Correlated Auto-Encoder (DCCAE) [27]. In addition, there are also many methods [30, 9] for handling heterogeneous data from multiple sources.

Table 1. The choices of norms for subspace clustering

Algorithm	$\mathcal{L}(\mathbf{X};\mathbf{Z})$	$\mathcal{R}(\mathbf{Z})$
SSC [10]	$\|\|\mathbf{X}-\mathbf{XZ}\|\|_1$	$\|\|\mathbf{Z}\|\|_1$
$\text{LRR}_{2,1}$ [19]	$\|\|\mathbf{X}-\mathbf{XZ}\|\|_{2,1}$	$\|\|\mathbf{Z}\|\|_*$
LRR_1 [19]	$\|\|\mathbf{X}-\mathbf{XZ}\|\|_1$	$\|\|\mathbf{Z}\|\|_*$
LRR_2 [19]	$\|\|\mathbf{X}-\mathbf{XZ}\|\|_F^2$	$\|\|\mathbf{Z}\|\|_*$
LSR [21]	$\|\|\mathbf{X}-\mathbf{XZ}\|\|_F^2$	$\|\|\mathbf{Z}\|\|_F^2$
SMR [12]	$\|\|\mathbf{X}-\mathbf{XZ}\|\|_F^2$	$tr(\mathbf{ZLZ}^T)$

3. The Proposed Approach

3.1. Hierarchical Self-Representative Layers

The self-representation term $\mathcal{L}(\mathbf{X};\mathbf{Z})$ in Eq. (1) can be taken as a linear fully connected layer without activations termed Self-Representative Layer (SRL) [13]. Specifically, $\mathbf{x}_i$ and $\mathbf{Z}$ denote a node in the network and the weighting parameters of SRL, respectively. Moreover, $\mathcal{R}(\mathbf{Z})$ imposes regularization on the weights of SRL.

Assuming that $\{\mathbf{X}^1, \ldots, \mathbf{X}^V\}$ come from V different views, and $\mathbf{H}$ represents the latent representation, we aim to simultaneously construct view-specific and common subspace representations denoted as $\{\mathbf{\Theta}_S^v\}_{v=1}^V$ and $\mathbf{\Theta}_C$, respectively, using Hierarchical Self-Representative Layers (HSRL). Specifically, view-specific SRL maps original features into subspace representations, and the common SRL further reveals the subspace structure of latent representation $\mathbf{H}$. Both of them simultaneously explore structural information of data, handle possible noise, and improve the clustering performance. We update the weighting parameters of HSRL using the objective function below:

$$\begin{aligned}\min_{\{\mathbf{\Theta}_S^v\}_{v=1}^V, \mathbf{\Theta}_C} & \mathcal{L}_S(\{\mathbf{X}^v\}_{v=1}^V, \mathbf{H}; \{\mathbf{\Theta}_S^v\}_{v=1}^V, \mathbf{\Theta}_C) \\ & + \beta \mathcal{R}(\{\mathbf{\Theta}_S^v\}_{v=1}^V, \mathbf{\Theta}_C),\end{aligned} \quad (4)$$

where $\mathcal{L}_S(\cdot\,;\cdot)$ denotes the loss function associated with self-representation. In this work, we consider applying Frobenius norm on reconstruction loss to alleviate noise effect, and choosing nuclear norm for regularization term to guarantee the high within-class homogeneity [19]. Then we rewrite Eq. (4) as:

$$\begin{aligned}\min_{\{\mathbf{\Theta}_S^v\}_{v=1}^V, \mathbf{\Theta}_C} & \frac{1}{2}\sum_{v=1}^V ||\mathbf{X}^v - \mathbf{X}^v\mathbf{\Theta}_S^v||_F^2 + \frac{1}{2}||\mathbf{H} - \mathbf{H}\mathbf{\Theta}_C||_F^2 \\ & + \beta(\sum_{v=1}^V ||\mathbf{\Theta}_S^v||_* + ||\mathbf{\Theta}_C||_*).\end{aligned} \quad (5)$$

3.2. Backward Encoding Networks

Considering the complementarity of different view-specific subspace representations, we introduce the Back-

ward Encoding Networks (BEN) to explore complex relationships among them and simultaneously construct a latent representation $\mathbf{H}$. Note that, instead of forward projecting diverse views into a common low-dimensional space like CCA-based methods [7, 2], we try to learn a common latent representation $\mathbf{H}$ by using it to reconstruct all view-specific representations $\{\mathbf{\Theta}_S^v\}_{v=1}^V$ through nonlinear mappings $\{g_{\mathbf{\Theta}_E^v}(\mathbf{H})\}_{v=1}^V$, where $\mathbf{\Theta}_E^v$ is the weighting parameter of BEN corresponding to the vth view. For example, the latent vector $\mathbf{h}_i$ is mapped to the ith vector $\mathbf{\Theta}_{S,i}^v$ in the vth view, *i.e.*, $\mathbf{\Theta}_{S,i}^v = g_{\mathbf{\Theta}_E^v}(\mathbf{h}_i)$. By enforcing the latent representation to be close to each view-specific subspace representation, subspace structures of all views will be consistent with each other. We update BEN parameters $\{\mathbf{\Theta}_E^v\}_{v=1}^V$ and infer the latent representation $\mathbf{H}$ with the following loss function:

$$\begin{aligned}&\min_{\{\mathbf{\Theta}_E^v\}_{v=1}^V,\mathbf{H}} \mathcal{L}_E(\{\mathbf{\Theta}_S^v\}_{v=1}^V,\mathbf{H};\{\mathbf{\Theta}_E^v\}_{v=1}^V)+\gamma\mathcal{R}(\{\mathbf{\Theta}_E^v\}_{v=1}^V)\\ =&\min_{\{\mathbf{\Theta}_E^v\}_{v=1}^V,\mathbf{H}} \frac{1}{2}\sum_{v=1}^V||\mathbf{\Theta}_S^v-g_{\mathbf{\Theta}_E^v}(\mathbf{H})||_F^2+\gamma\sum_{v=1}^V||\mathbf{\Theta}_E^v||_F^2 \quad (6)\\ &\text{with}\quad g_{\mathbf{\Theta}_E^v}(\mathbf{H})=\mathbf{W}_M^v f(\mathbf{W}_{M-1}^v\cdots f(\mathbf{W}_1^v\mathbf{H})),\end{aligned}$$

where $\mathcal{L}_E(\cdot;\cdot)$ denotes the reconstruction loss for updating $\mathbf{H}$. BEN consists of M fully connected layers, which are able to nonlinearly encode complementary information from different views into a common latent representation $\mathbf{H}$. Besides, we introduce the regularization $\mathcal{R}(\mathbf{\Theta}_E^v)$ on the networks to raise the generalization ability of our model. Specifically, $\mathbf{W}_M^v$ is the weight matrix between the Mth and $(M-1)$th layer corresponding to the vth view, and $f(\cdot)$ is the activation function.

Consequently, the model parameters $\mathbf{\Theta}$ of RMSL, including BEN parameters $\mathbf{\Theta}_E^v$ and HSRL parameters $\mathbf{\Theta}_S^v$ (view-specific), $\mathbf{\Theta}_C$ (common), can be jointly optimized by the following general objective function:

$$\begin{aligned}&\min_{\mathbf{H},\mathbf{\Theta}} \alpha\mathcal{L}_E(\{\mathbf{\Theta}_S^v\}_{v=1}^V,\mathbf{H};\{\mathbf{\Theta}_E^v\}_{v=1}^V)+\gamma\mathcal{R}(\{\mathbf{\Theta}_E^v\}_{v=1}^V) \quad (7)\\ &+\mathcal{L}_S(\{\mathbf{X}^v\}_{v=1}^V,\mathbf{H};\{\mathbf{\Theta}_S^v\}_{v=1}^V,\mathbf{\Theta}_C)+\beta\mathcal{R}(\{\mathbf{\Theta}_S^v\}_{v=1}^V,\mathbf{\Theta}_C).\end{aligned}$$

To summarize, our model constructs reciprocal multi-layer subspace representations linked with a latent representation, to hierarchically recover the cluster structure of data and seek for a common partition of data shared by all the views. LMSC [32] learns a latent representation based on original feature matrices $\{\mathbf{X}^1,\ldots,\mathbf{X}^V\}$, while cannot explore the consistency of different views, and it is easily affected by the high-dimensionality and possible noise of raw data. Considering subspace representation can reveal the underlying low-dimensional subspace structure of high-dimensional data, our model drives the latent representation $\mathbf{H}$ to be similar to different view-specific subspace representations $\{\mathbf{\Theta}_S^v\}_{v=1}^V$, which implicitly facilitates subspace structures of all views to be consistent with each other.

Similar to ours, DCCAE [27] also constructs a common space based on view-specific features extracted from original views with DNNs, but it is quite different from ours: (1) we learn view-specific subspace representations using SRL, which is quite effective for high-dimensional data, and basically subspace representation itself is also high-dimensional, which inspires us to construct multi-layer self-representations to hierarchically identify the underlying cluster structure of data; (2) DCCAE integrates multiple views by maximizing their correlations according to Canonical Correlation Analysis (CCA), but neglects the complementarity of different views. Different from DCCAE, we learn a shared latent representation by reconstructing each view-specific subspace representation from it using BEN, which enforces the latent representation to flexibly encode complementary information from all views.

3.3. Optimization

To optimize our objective function in Eq. (7), we employ the Alternating Direction Minimization (ADM) strategy. In order to make objective function separable, we replace $\mathbf{\Theta}_S^v$ and $\mathbf{\Theta}_C$ with the newly introduced auxiliary variables $\mathbf{R}^v$ and $\mathbf{J}$, respectively, and then obtain the following equivalent objective function:

$$\begin{aligned}&\min_{\mathbf{\Theta},\mathbf{H},\mathbf{J},\{\mathbf{R}\}_{v=1}^V} \frac{1}{2}\sum_{v=1}^V||\mathbf{X}^v-\mathbf{X}^v\mathbf{\Theta}_S^v||_F^2+\frac{1}{2}||\mathbf{H}-\mathbf{H}\mathbf{\Theta}_C||_F^2\\ &+\beta(\sum_{v=1}^V||\mathbf{R}^v||_*+||\mathbf{J}||_*)+\sum_{v=1}^V\frac{\alpha^v}{2}||\mathbf{\Theta}_S^v-g_{\mathbf{\Theta}_E^v}(\mathbf{H})||_F^2 \quad (8)\\ &+\gamma\sum_{v=1}^V||\mathbf{\Theta}_E^v||_F^2 \qquad s.t.\ \mathbf{\Theta}_C=\mathbf{J},\quad \mathbf{\Theta}_S^v=\mathbf{R}^v.\end{aligned}$$

We adopt the Augmented Lagrange Multiplier (ALM) [18] method to solve this problem by minimizing the following function:

$$\begin{aligned}\mathcal{L}(\mathbf{\Theta},\mathbf{H},\mathbf{J},\{\mathbf{R}^v\}_{v=1}^V)&=\frac{1}{2}\sum_{v=1}^V||\mathbf{X}^v-\mathbf{X}^v\mathbf{\Theta}_S^v||_F^2\\ &+\frac{1}{2}||\mathbf{H}-\mathbf{H}\mathbf{\Theta}_C||_F^2+\sum_{v=1}^V\frac{\alpha^v}{2}||\mathbf{\Theta}_S^v-g_{\mathbf{\Theta}_E^v}(\mathbf{H})||_F^2\\ &+\beta(\sum_{v=1}^V||\mathbf{R}^v||_*+||\mathbf{J}||_*)+\gamma\sum_{v=1}^V||\mathbf{\Theta}_E^v||_F^2\\ &+\sum_{v=1}^V\Phi(\mathbf{Y}_1^v,\mathbf{\Theta}_S^v-\mathbf{R}^v)+\Phi(\mathbf{Y}_2,\mathbf{\Theta}_C-\mathbf{J}).\end{aligned} \quad (9)$$

We define $\Phi(\mathbf{Y},\mathbf{D})=\frac{\mu}{2}\|\mathbf{D}\|_F^2+\langle\mathbf{Y},\mathbf{D}\rangle$, where $\langle\cdot,\cdot\rangle$ is the Frobenius inner product defined as $\langle\mathbf{A},\mathbf{B}\rangle=\mathrm{tr}\left(\mathbf{A}^T\mathbf{B}\right)$. $\mu>0$ and $\mathbf{Y}$ are the penalty factor and Lagrange multiplier, respectively. According to ADM strategy, we divide our objective function into the following subproblems:

• Update the HSRL parameters $\mathbf{\Theta}_S^v$ and $\mathbf{\Theta}_C$: fixing the

other variables, we update $\mathbf{\Theta}_S^v$ by solving the following subproblem:

$$\begin{aligned}\mathbf{\Theta}_S^{v*} = \arg\min_{\mathbf{\Theta}_S^v} \frac{\alpha^v}{2}||\mathbf{\Theta}_S^v - g_{\mathbf{\Theta}_E^v}(\mathbf{H})||_F^2 \\ +\frac{1}{2}||\mathbf{X}^v - \mathbf{X}^v\mathbf{\Theta}_S^v||_{\mathrm{F}}^2 + \Phi(\mathbf{Y}_1^v, \mathbf{\Theta}_S^v - \mathbf{R}^v).\end{aligned} \tag{10}$$

Taking the derivative with respect to $\mathbf{\Theta}_S^v$ and setting it to zero, we can get the closed-form solution:

$$\begin{aligned}\mathbf{\Theta}_S^{v*} = [(\mathbf{X}^v)^T\mathbf{X}^v + (\alpha^v + \mu)\mathbf{I}]^{-1} \\ \cdot[(\mathbf{X}^v)^T\mathbf{X}^v + \mu\mathbf{R}^v - \mathbf{Y}_1^v + \alpha^v g_{\mathbf{\Theta}_E^v}(\mathbf{H})].\end{aligned} \tag{11}$$

Similarly, the subproblem with respect to $\mathbf{\Theta}_C$ is:

$$\mathbf{\Theta}_C^* = \arg\min_{\mathbf{\Theta}_C} \frac{1}{2}||\mathbf{H} - \mathbf{H}\mathbf{\Theta}_C||_F^2 + \Phi(\mathbf{Y}_2, \mathbf{\Theta}_C - \mathbf{J}). \tag{12}$$

The solution associated with this subproblem is:

$$\mathbf{\Theta}_C^* = (\mathbf{H}^T\mathbf{H} + \mu\mathbf{I})^{-1}(\mathbf{H}^T\mathbf{H} + \mu\mathbf{J} - \mathbf{Y}_2). \tag{13}$$

• Update the BEN parameters $\mathbf{\Theta}_E^v$, *i.e.*, $\mathbf{W}_1^v$ and $\mathbf{W}_2^v$: in this paper, we define $g_{\mathbf{\Theta}_E^v}(\mathbf{H})$ as a two-layer fully connected network and $\mathbf{W}_1^v$, $\mathbf{W}_2^v$ are the weight matrices between adjacent layers. In addition, we adopt the '*tanh*' activation function whose derivative is: $\tanh'(z) = 1 - \tanh^2(z)$. Then we rewrite Eq. (6) as:

$$\begin{aligned}\mathcal{L}(\mathbf{W}^v) = \frac{\alpha^v}{2}||\mathbf{\Theta}_S^v - \mathbf{W}_2^v f(\mathbf{W}_1^v\mathbf{H})||_F^2 \\ + \frac{\gamma}{2}(||\mathbf{W}_1^v||_F^2 + ||\mathbf{W}_2^v||_F^2).\end{aligned} \tag{14}$$

The rules to update $\mathbf{W}_1^v$ and $\mathbf{W}_2^v$ are as follows:

$$\begin{aligned}&\mathbf{W}_2^{v*} = \mathbf{\Theta}_S^v(\mathbf{F}^v)^T[\mathbf{F}^v(\mathbf{F}^v)^T + \frac{\gamma}{\alpha_v}\mathbf{I}]^{-1}, \\ \text{and}\quad &\frac{\partial\mathcal{L}(\mathbf{W})}{\partial\mathbf{W}_1^v} = \alpha^v(\mathbf{W}_2^v)^T(\mathbf{W}_2^v\mathbf{F}^v - \mathbf{\Theta}_S^v) \\ &\qquad\circ(\mathbf{1} - \mathbf{F}^v \circ \mathbf{F}^v)\mathbf{H}^T + \gamma\mathbf{W}_1^v,\end{aligned} \tag{15}$$

where $\mathbf{F}^v = f(\mathbf{W}_1^v\mathbf{H}) = \tanh(\mathbf{W}_1^v\mathbf{H})$, $\mathbf{1}$ denotes a matrix whose elements are all ones, and $\circ$ represents element-wise multiplication. We adopt the Gradient Descent (GD) algorithm to update $\mathbf{W}_1^v$.

• Update $\mathbf{H}$: similarly, $\mathbf{H}$ also can be effectively optimized by GD, where the gradient with respect to $\mathbf{H}$ is:

$$\begin{aligned}\frac{\partial\mathcal{L}(\mathbf{H})}{\partial\mathbf{H}} = \sum_{v=1}^{V}\alpha^v(\mathbf{W}_1^v)^T[(\mathbf{W}_2^v)^T(\mathbf{W}_2^v\mathbf{F}^v - \mathbf{\Theta}_S^v) \\ \circ(1 - \mathbf{F}^v \circ \mathbf{F}^v)] + \mathbf{H}(\mathbf{I} - \mathbf{\Theta}_C - \mathbf{\Theta}_C^T + \mathbf{\Theta}_C\mathbf{\Theta}_C^T).\end{aligned} \tag{16}$$

• Update auxiliary variables $\mathbf{J}$, $\mathbf{R}^v$, and multipliers $\mathbf{Y}_1^v$, $\mathbf{Y}_2$:

$$\begin{aligned}&\mathbf{J}^* = \arg\min_{\mathbf{J}} \frac{\beta}{\mu}||\mathbf{J}||_* + \frac{1}{2}||\mathbf{J} - (\mathbf{\Theta}_C + \frac{\mathbf{Y}_2}{\mu})||_F^2, \\ &\mathbf{R}^{v*} = \arg\min_{\mathbf{R}^v} \frac{\beta}{\mu}||\mathbf{R}^v||_* + \frac{1}{2}||\mathbf{R}^v - (\mathbf{\Theta}_S^v + \frac{\mathbf{Y}_1^v}{\mu})||_F^2, \\ &\mathbf{Y}_1^{v*} = \mathbf{Y}_1^v + \mu(\mathbf{\Theta}_S^v - \mathbf{R}^v), \\ &\mathbf{Y}_2^* = \mathbf{Y}_2 + \mu(\mathbf{\Theta}_C - \mathbf{J}).\end{aligned} \tag{17}$$

Algorithm 1: Optimization of our method

Input: Multi-view data: $\{\mathbf{X}^{(1)}, ..., \mathbf{X}^{(V)}\}$, hyperparameters $\{\alpha^v\}_{v=1}^V$, β and γ, the dimensionality of latent representation K.

Initialize: Randomly initialize latent representation $\mathbf{H}$, HSRL parameters $\mathbf{\Theta}_C$, and BEN parameters $\{\mathbf{\Theta}_E^v\}_{v=1}^V$; Generate $\{\mathbf{\Theta}_S^v\}_{v=1}^V$ by Eq. (1); $\mu = 10^{-5}$, $\rho = 1.5$, $\epsilon = 10^{-4}$, $\max_\mu = 10^6$.

while *not converged* **do**

- Update the HSRL parameters $\{\mathbf{\Theta}_S^v\}_{v=1}^V, \mathbf{\Theta}_C$ according to Eq. (11) and Eq. (13);
- Update BEN parameters $\{\mathbf{\Theta}_E^v\}_{v=1}^V$ by Eq. (15);
- Update the latent representation $\mathbf{H}$ by Eq. (16);
- Update auxiliary variables $\mathbf{J}$, $\{\mathbf{R}^v\}_{v=1}^V$ and multipliers $\{\mathbf{Y}_1^v\}_{v=1}^V$, $\mathbf{Y}_2$ according to Eq. (17);
- Update the parameter μ by $\mu = \min(\max_\mu, \rho\mu)$;
- Check the convergence conditions: $||\mathbf{\Theta}_C - \mathbf{J}||_\infty < \epsilon$ and $||\mathbf{\Theta}_S^v - \mathbf{R}^v||_\infty < \epsilon$.

end

Output: $\mathbf{H}$, $\{\mathbf{\Theta}_S^v\}_{v=1}^V$, $\mathbf{\Theta}_C$.

The subproblems corresponding to $\mathbf{J}$ and $\mathbf{R}^v$ can be solved by the Singular Value Thresholding (SVT) [5] algorithm. For clarification, the optimization procedure is summarized in Algorithm 1.

3.4. Complexity Analysis

The computational cost of our method is mainly composed of two parts, *i.e.*, affinity matrix learning and spectral clustering. For clarification, we define N, T, V, K as the number of data points, iterations, views, and the dimensionality of latent space, respectively. L_1 and L_2 denote the iterations of GD for updating $\mathbf{W}_1$ and $\mathbf{H}$, and D_1 is the dimensionality of middle layer of the network. The complexity of spectral clustering is $O(N^3)$ from singular value decomposition. As for affinity matrix learning, the complexities of updating $\mathbf{\Theta}_S^v$, $\mathbf{\Theta}_C$, $\mathbf{J}$, and $\mathbf{R}^v$ are $O(N^3)$. Updating BEN parameters and latent representation $\mathbf{H}$ consume $O(L_1D_1VN^2)$ and $O(L_2(N^3+D_1VN^2+KN^2))$, respectively. The total complexity of our model is $O(T(L_2N^3 + (L_1+L_2)D_1VN^2+L_2KN^2))$, where $L_1, L_2,\ D_1,$ and V can be seen as constants. In general, the computational complexity is $O(N^3)$.

The main complexities of self-representative subspace clustering methods [6, 31, 32] are from the graph (with the size $N \times N$) involved, which leads to high time-complexity matrix operations, such as SVD decomposition. Generally, the total complexities of these methods on large-scale data are $O(N^3)$.

Table 2. Performance comparisons of different methods

Datasets	Metrics	Co-Reg	RMSC	DMF	MVEC	BMVC	DiMSC	LT-MSC	LMSC	Ours
Football	ACC	62.96±1.21	78.55±3.84	78.98±1.29	77.08±2.55	77.42±0.00	75.40±2.26	79.03±2.01	86.25±1.45	**91.57±0.93**
	NMI	76.58±1.47	84.34±2.04	83.38±0.79	83.36±1.11	80.22±0.00	82.16±1.45	84.22±1.17	89.31±2.22	**92.29±0.42**
	F-score	60.19±2.81	70.97±4.01	70.35±0.87	67.08±3.70	63.72±0.00	67.13±1.19	71.32±1.37	79.40±1.40	**83.87±1.57**
	RI	93.33±0.21	97.08±0.44	96.51±0.44	96.22±0.54	96.69±0.00	96.74±0.59	97.19±0.55	97.97±0.73	**98.40±0.16**
Reuters	ACC	26.38±1.75	39.46±1.29	40.02±2.40	50.10±0.00	44.10±0.01	43.70±1.13	34.30±0.63	48.50±1.06	**54.04±0.56**
	NMI	28.74±1.13	19.00±0.75	22.88±1.15	30.17±0.02	25.41±0.00	23.31±0.33	17.93±1.32	31.23±0.99	**37.49±0.76**
	F-score	36.45±1.67	31.86±1.40	34.63±1.37	39.59±0.02	37.93±0.00	33.01±0.39	28.29±0.95	40.69±1.21	**44.20±0.45**
	RI	68.60±0.29	68.05±0.92	58.71±1.29	**74.17±0.62**	69.15±0.00	67.49±0.28	68.16±0.53	68.37±0.63	71.37±0.35
BBCSport	ACC	73.31±0.58	73.72±0.37	76.84±0.00	96.88±0.00	80.70±0.00	95.10±2.17	90.26±0.73	96.32±0.78	**97.61±0.18**
	NMI	71.76±0.05	60.84±0.75	53.39±0.08	90.32±0.02	70.80±0.00	85.11±0.13	77.54±0.46	88.66±0.46	**91.73±0.52**
	F-score	76.64±0.14	65.51±0.20	62.46±0.01	93.72±0.02	80.81±0.00	91.02±0.14	80.16±0.59	92.54±0.26	**95.35±0.41**
	RI	89.14±0.03	92.29±0.33	82.45±0.00	97.02±0.01	90.23±0.00	95.72±0.10	90.36±0.27	96.49±0.11	**97.81±0.19**
ORL	ACC	60.89±1.85	75.20±2.84	74.45±2.29	71.06±2.02	72.75±0.00	83.84±1.16	81.94±2.54	82.75±1.20	**88.10±1.27**
	NMI	84.56±1.41	89.76±1.98	86.27±1.00	80.23±1.16	85.20±0.00	94.02±1.35	93.10±1.06	92.81±0.57	**94.96±0.47**
	F-score	63.38±2.38	70.12±4.34	64.61±2.56	61.94±2.31	62.95±0.00	80.71±1.38	75.83±2.62	77.53±1.14	**84.22±1.43**
	RI	97.30±0.25	97.53±0.34	98.31±0.14	97.16±0.65	98.23±0.00	98.73±0.17	98.37±0.40	98.80±0.09	**99.15±0.07**
COIL-20	ACC	56.02±0.07	68.59±4.50	72.97±0.20	70.45±2.24	73.33±0.00	72.78±1.44	80.43±1.12	74.72±1.53	**82.19±1.39**
	NMI	76.54±1.24	80.11±1.88	85.80±0.26	87.76±0.89	80.07±0.00	84.61±1.75	86.27±0.25	86.63±1.40	**94.10±1.32**
	F-score	59.37±0.13	65.62±4.26	63.22±1.03	64.05±2.12	67.62±0.00	71.99±0.50	76.13±0.75	71.26±1.72	**81.20±1.72**
	RI	95.51±0.38	96.64±0.32	95.44±0.19	95.35±0.49	96.66±0.00	97.14±0.11	97.23±0.28	96.94±0.37	**97.90±0.41**
ANIMAL	ACC	21.87±1.63	61.58±4.50	45.92±1.80	57.96±1.33	50.15±0.00	32.61±1.81	33.65±0.67	64.47±0.44	**66.16±0.54**
	NMI	45.46±0.71	70.46±1.84	56.64±1.31	68.72±0.50	66.76±0.00	44.62±0.89	41.29±0.40	72.66±0.35	**73.19±0.60**
	F-score	21.19±1.14	54.30±4.16	32.86±2.17	49.31±2.49	41.47±0.00	20.66±1.10	21.65±0.49	54.54±0.37	**57.29±1.14**
	RI	96.55±0.16	97.95±0.35	97.09±0.22	97.12±0.25	96.98±0.00	96.30±0.23	96.53±0.16	97.97±0.08	**98.12±0.05**

[1] The top value is highlighted in red bold font and the second best in blue.

4. Experiments

4.1. Datasets

We evaluate the clustering performance of our model over three different types of benchmark datasets, including image, text, and community networks.

• **Football**[1] consists of 248 English football players and clubs active on Twitter, which are described from 9 different views. The disjoint communities are associated with 20 clubs in the league.

• **Reuters** [1] is a multilingual dataset including 1000 newswire articles of 6 classes written in 5 languages.

• **BBCSport**[2] is a collection of 544 documents associated with 2 views taken from sports articles in 5 topical areas.

• **ORL**[3] contains 400 face images of 40 distinct subjects, from which 3 types of features are extracted.

• **COIL-20**[4] consists of 1440 images of 20 objects taken by a camera from varying angles. This dataset is characterized from 3 different views.

• **ANIMAL** [17] contains 30475 images of 50 animal classes, which are composed of 2 types of deep features (extracted with DECAF [14] and VGG19 [24]). We select 10158 samples with fixed interval to generate a subset.

[1] http://mlg.ucd.ie/aggregation/

[2] http://mlg.ucd.ie/datasets/

[3] http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html

[4] http://www.cs.columbia.edu/CAVE/software/softlib/

4.2. Experimental Setup

Compared Methods. We compare our method against the following 8 baselines: **Co-Reg** [15] co-regularizes clustering hypotheses of different views to be consistent with each other; **RMSC** [29] constructs a joint low-rank transition probability matrix as an input for spectral clustering; **DMF** [34] conducts multi-view clustering in the deep matrix factorization framework, which maximizes mutual information of different views by enforcing the final-layer nonnegative representation of each view to be the same; **MVEC** [25] extends ensemble clustering to multi-view cases, which generates a group of view-specific basic partitions to extract a consensus one shared by multiple views; **BMVC** [33] realizes large-scale clustering by integrating compact collaborative discrete representation learning and binary clustering structure learning into a unified framework; **DiMSC** [6] explores the diversity of different subspace representations for better incorporating complementary information from multiple views; **LT-MSC** [31] adopts low-rank tensor to exploit high-oder relationships among different views; **LMSC** [32] learns a comprehensive latent representation from different views for subspace clustering. For all above methods, the parameters are tuned to achieve the best performance.

Parameter Setting. In this work, we employ the grid search strategy to find the optimal hyper-parameters. For simplicity, we set the trade-off parameters $\alpha^1 = \ldots = \alpha^V = \alpha$ and choose α and β from $\{0.1, 0.2, \cdots, 1\}$. The dimensionality K of latent representation and the regularization

Table 3. Performance comparisons on different representations

Datasets	Metrics	Best View	KCCA	DCCA	DCCAE	Layer-I	Layer-II	Layer-III	Layer-IV
Football	ACC	49.36±2.36	48.11±4.76	63.31±3.17	64.19±2.14	38.54±2.46	69.26±2.35	71.17±4.48	**78.37±4.06**
	NMI	61.26±2.56	58.84±3.27	73.45±1.99	79.56±1.99	48.29±4.03	80.92±1.65	83.34±4.00	**86.05±3.17**
	F-score	27.94±3.57	23.95±2.24	44.39±2.39	54.08±2.46	21.34±2.82	58.55±3.01	61.44±3.61	**69.44±4.17**
	RI	82.97±2.15	80.26±3.15	91.40±1.89	94.35±0.73	81.24±2.14	95.64±0.43	95.19±2.06	**96.46±1.65**
Reuters	ACC	33.30±1.59	34.19±3.17	40.33±3.04	31.40±1.86	32.75±2.48	40.13±2.78	32.48±1.95	**42.40±2.55**
	NMI	10.38±3.35	12.11±1.61	22.08±2.33	15.20±2.92	9.12±1.81	18.96±3.38	15.89±2.16	**23.33±1.69**
	F-score	35.58±1.87	36.30±3.04	31.56±3.48	28.01±1.42	35.52±2.36	**37.62±2.34**	35.72±1.78	36.87±2.15
	RI	34.15±1.18	38.23±1.44	53.46±2.44	56.96±1.60	33.50±2.10	54.03±2.86	38.14±1.72	**57.78±2.70**
BBCSport	ACC	41.94±3.26	39.51±2.36	69.39±1.80	72.98±3.13	51.29±3.65	41.31±3.09	62.28±3.25	**77.13±1.63**
	NMI	15.94±2.19	12.45±1.88	50.36±1.83	54.55±4.03	35.11±2.44	15.57±1.57	54.42±3.49	**71.47±3.12**
	F-score	41.53±2.23	40.42±1.88	61.20±2.49	66.46±1.89	46.56±1.81	41.16±2.19	58.39±3.34	**73.45±2.78**
	RI	37.45±1.13	34.08±3.26	80.61±0.39	83.35±1.51	49.37±2.27	36.49±2.50	69.76±42.85	**84.35±1.98**
ORL	ACC	56.48±2.34	57.39±3.90	58.40±2.87	54.66±1.90	51.85±3.05	68.91±3.32	74.50±2.70	**76.95±1.95**
	NMI	77.43±3.60	77.62±1.96	76.46±1.22	74.70±0.93	74.30±1.60	86.91±1.53	90.27±1.50	**91.29±1.30**
	F-score	44.68±2.47	45.26±2.23	47.44±2.60	41.31±1.98	38.37±3.10	58.27±2.48	56.92±1.94	**65.30±2.17**
	RI	96.94±2.36	96.97±0.42	97.27±0.17	97.02±0.16	96.44±0.31	97.60±0.51	97.00±1.13	**97.96±0.63**
COIL-20	ACC	58.60±3.07	56.07±2.42	60.35±3.34	58.81±3.02	57.71±2.12	46.26±1.87	55.36±1.81	**63.04±2.20**
	NMI	76.26±4.24	77.41±2.59	78.43±1.24	72.79±1.21	76.91±1.95	70.81±3.10	74.44±1.79	**79.24±1.91**
	F-score	58.00±3.13	44.15±2.07	59.51±2.71	53.14±2.37	55.13±2.92	35.76±3.39	46.08±3.83	**61.05±2.79**
	RI	95.25±1.38	90.50±2.31	95.09±0.35	94.86±0.33	94.59±0.55	87.00±4.21	92.48±2.05	**95.67±0.61**
ANIMAL	ACC	28.21±1.30	30.00±0.98	39.18±1.45	28.85±0.85	45.25±2.50	21.81±3.57	46.38±2.15	**49.89±1.63**
	NMI	42.56±0.73	43.62±0.38	51.50±0.63	45.30±0.45	59.59±0.94	31.27±4.18	62.81±1.52	**65.68±0.80**
	F-score	16.99±0.93	17.28±0.68	25.40±0.95	17.42±0.66	33.00±2.48	7.59±1.89	30.63±4.15	**37.26±2.46**
	RI	95.99±0.93	96.21±0.06	**96.55±0.10**	96.30±0.07	96.16±2.41	73.71±1.64	94.82±1.10	96.32±0.35

[1] The top value is highlighted in red bold font and the second best in blue.

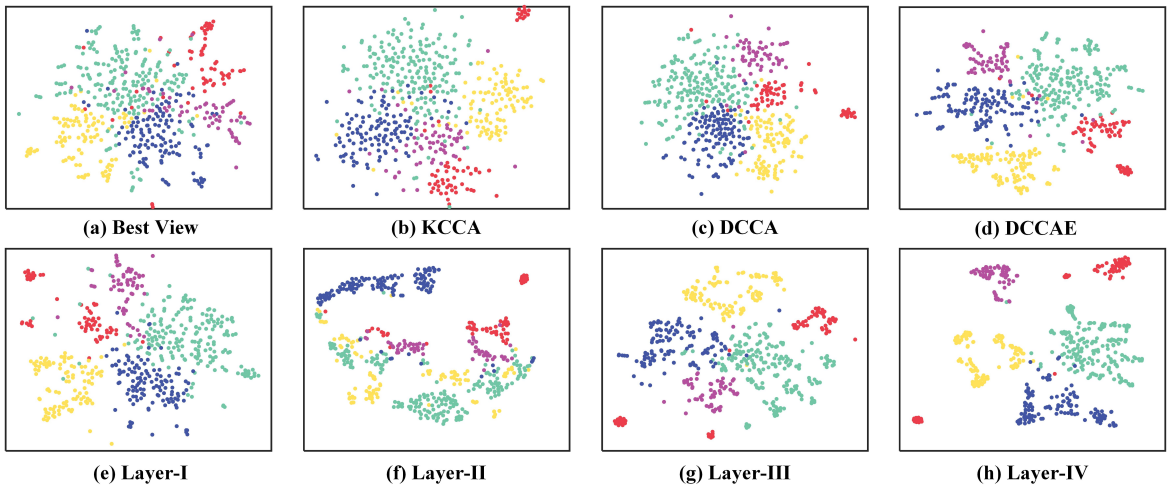

Figure 2. t-SNE visualizations of different layers and representations learned by various methods on BBCSport.

parameter γ for updating network weights are tuned from $\{20, 40, \cdots, 200\}$ and $\{0.001, 0.005, \cdots, 1\}$, respectively. Figure 3 (a)(b)(c) show the effect of parameters varying on the clustering performance on Football. We can see that our model is quite insensitive to hyper-parameters α and β. When the values of K and γ are respectively 120 and 0.1, the optimal clustering performance is obtained.

Evaluation Measures. We utilize four popular metrics to evaluate clustering performance, each of which favors different property of clustering, including Normalized Mutual Information (NMI), Accuracy (ACC), F-score, and Rand Index (RI). The higher value means better performance. Note that, we report the mean values and standard derivations of 30 independent trials over each dataset to avoid the randomicity.

4.3. Experimental Results

We compare RMSL with 8 state-of-the-art multi-view clustering algorithms on 6 datasets. The results are reported in Table 2, from which we have the following observations: (1) our algorithm consistently obtains the best performances on all datasets in terms of ACC and NMI, which shows its robustness to different types of data; (2) although BMVC is easily scale to large data, it does not

produce very competitive performances on all datasets compared with ours; (3) our method boosts the clustering performance by a large margin over DiMSC and LT-MSC that conduct self-representation within each single view, which indicates that learning a shared latent representation is of vital importance because it contains complementary information from multiple views and can depict data more comprehensively; (4) RMSL significantly outperforms LMSC on all datasets, which demonstrates that integrating multiple views based on subspace representations will be superior to that using original features.

Ablation Study. To further investigate the effectiveness of diverse components of our model, we conduct k-means on four different layers of RMSL. As shown in Figure 1, **Layer-I** to **Layer-IV** respectively represent the concatenation of multiple input features, the combination of view-specific subspace representations, the latent representation, and the common subspace representation. **Best View** denotes the clustering result within the best view. In addition, we compare our method with several multi-view representation learning algorithms: **KCCA** [3] and **DCCA** [2] are nonlinear extensions of CCA, which extract both nonlinear features for each view and the canonical correlations between different views using kernel technique and DNNs, respectively; **DCCAE** [27] adds the auto-encoder regularization term to DCCA for better reconstructing inputs.

From the results reported in Table 3, we have the following observations: (1) generally, from Layer-I to Layer-IV, the deeper layer recovers more clear subspace structure and generates better clustering performance; (2) Layer-I (feature concatenation) performs even worse than Best View on most datasets, since the dimensionality of feature concatenation is too high to reflect the intrinsic structure of data; (3) although KCCA, DCCA, and DCCAE have considered nonlinear correlations between different views, they are not as competitive as RMSL. The possible reasons include: first, they project multiple original views into a common space by maximizing their correlations, but cannot well explore the complementarity of different views; second, the multi-view fusion process is separated from clustering, which causes the latent representation not well-adapted to clustering; third, CCA-based methods directly integrate noisy high-dimensional raw features, which will degrade the quality of learned common representation.

We also qualitatively investigate the improvement of cluster structure of different representations. The resulting t-SNE visualizations on BBCSport are shown in Figure 2. In general, the visualizations agree with the clustering results in Table 3. We can observe that projections by nonlinear CCA algorithms (KCCA, DCCA, DCCAE) manage to map data points of the same identity to similar locations, but the class separation distances are too small. According to Figure 2 (e)-(h), the cluster structure revealed by deeper

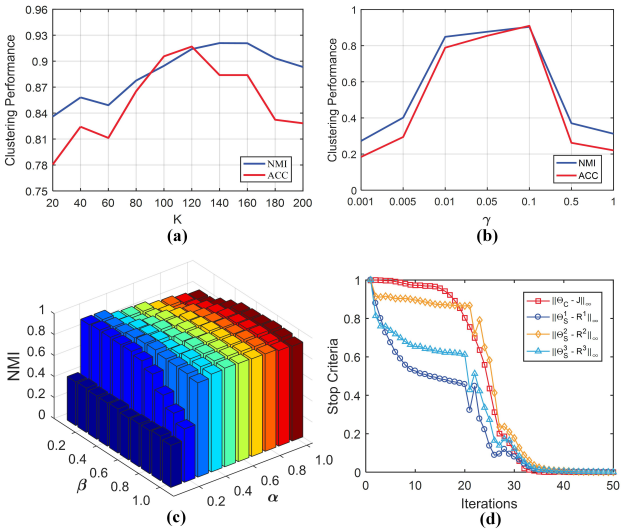

Figure 3. Parameter sensitivity analysis of (a) K, (b) γ, (c) α and β. (d) The convergence curves (the ordinate values are normalized into the range $[0, 1]$).

layer is more clear. Overall, Layer-IV (common subspace representation) gives the most compact subspace structure, with different identities pushed far apart.

Convergence Analysis. The convergence of inexact Augmented Lagrange Multiplier (ALM) approach with three or more variable blocks is still difficult to prove in theory [19]. Fortunately, our algorithm could empirically converge within a number of iterations on all datasets as shown in Figure 3 (d).

5. Conclusions

In this paper, a novel Reciprocal Multi-layer Subspace learning (RMSL) algorithm is proposed to cluster high-dimensional and noisy data from diverse sources. In RMSL, Hierarchical Self-Representative Layers (HSRL) recover the subspace structure of data. Moreover, the Backward Encoding Networks (BEN) simultaneously explore complementary and consistent structural information from different views and integrate multiple view-specific subspace representations together into a common latent representation. Extensive experiments conducted on real-world datasets show the superiority of RMSL over other state-of-the-art clustering methods.

6. Acknowledgement

This work was supported by the National Natural Science Foundation of China (No.61602337, 61806135, 61625204, and 61836006), and the Opening Project of State Key Laboratory of Digital Publishing Technology.

References

[1] Massih-Reza Amini, Nicolas Usunier, and Cyril Goutte. Learning from multiple partially observed views - an application to multilingual text categorization. In *Neural Information Processing Systems (NIPS)*, pages 28–36, 2009.

[2] Galen Andrew, Raman Arora, Jeff A. Bilmes, and Karen Livescu. Deep canonical correlation analysis. In *ICML*, pages 1247–1255, 2013.

[3] Francis R. Bach and Michael I. Jordan. Kernel independent component analysis. In *JMLR*, 2002.

[4] Matthew B. Blaschko and Christoph H. Lampert. Correlational spectral clustering. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, 2008.

[5] Jian-Feng Cai, Emmanuel J. Cands, and Zuowei Shen. A singular value thresholding algorithm for matrix completion. *Siam Journal on Optimization*, 20(4):1956–1982, 2010.

[6] Xiaochun Cao, Changqing Zhang, Huazhu Fu, Si Liu, and Hua Zhang. Diversity-induced multi-view subspace clustering. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 586–594, 2015.

[7] Kamalika Chaudhuri, Sham M. Kakade, Karen Livescu, and Karthik Sridharan. Multi-view clustering via canonical correlation analysis. In *International Conference on Machine Learning (ICML)*, pages 129–136, 2009.

[8] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 886–893, 2005.

[9] Cheng Deng, Xianglong Liu, Chao Li, and Dacheng Tao. Active multi-kernel domain adaptation for hyperspectral image classification. *Pattern Recognition*, 77:306–315, 2018.

[10] Ehsan Elhamifar and Rene Vidal. Sparse subspace clustering: Algorithm, theory, and applications. *IEEE TPAMI*, 35(11):2765–2781, 2013.

[11] Harold Hotelling. Relations between two sets of variates. *Biometrika*, 28(3/4):321–377, 1936.

[12] Han Hu, Zhouchen Lin, Jianjiang Feng, and Jie Zhou. Smooth representation clustering. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3834–3841, 2014.

[13] Pan Ji, Tong Zhang, Hongdong Li, Mathieu Salzmann, and Ian D. Reid. Deep subspace clustering networks. *Neural Information Processing Systems (NIPS)*, pages 24–33, 2017.

[14] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In *Neural Information Processing Systems (NIPS)*, pages 1097–1105, 2012.

[15] Abhishek Kumar, Piyush Rai, and Hal Daume. Co-regularized multi-view spectral clustering. In *Neural Information Processing Systems (NIPS)*, pages 1413–1421, 2011.

[16] M. Lades, J. C. Vorbruggen, J. Buhmann, J. Lange, C. von der Malsburg, R. P. Wurtz, and W. Konen. Distortion invariant object recognition in the dynamic link architecture. *IEEE Transactions on Computers*, 42(3):300–311, 1993.

[17] Christoph H Lampert, Nickisch Hannes, and Harmeling Stefan. Attribute-based classification for zero-shot visual object categorization. *IEEE TPAMI*, 36(3):453–465, 2014.

[18] Zhouchen Lin, Risheng Liu, and Zhixun Su. Linearized alternating direction method with adaptive penalty for low-rank representation. In *Neural Information Processing Systems (NIPS)*, pages 612–620, 2011.

[19] Guangcan Liu, Zhouchen Lin, Shuicheng Yan, Ju Sun, Yong Yu, and Yi Ma. Robust recovery of subspace structures by low-rank representation. *IEEE TPAMI*, 35(1):171–184, 2013.

[20] David G. Lowe. Object recognition from local scale-invariant features. In *IEEE International Conference on Computer Vision (ICCV)*, volume 2, pages 1150–1157, 1999.

[21] Can-Yi Lu, Hai Min, Zhong-Qiu Zhao, Lin Zhu, De-Shuang Huang, and Shuicheng Yan. Robust and efficient subspace segmentation via least squares regression. In *European Conference on Computer Vision (ECCV)*, pages 347–360, 2012.

[22] Shirui Luo, Changqing Zhang, Wei Zhang, and Xiaochun Cao. Consistent and specific multi-view subspace clustering. In *AAAI*, pages 3730–3737, 2018.

[23] Jiquan Ngiam, Aditya Khosla, Mingyu Kim, Juhan Nam, Honglak Lee, and Andrew Y. Ng. Multimodal deep learning. In *International Conference on Machine Learning (ICML)*, pages 689–696, 2011.

[24] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *ICLR*, 2015.

[25] Zhiqiang Tao, Hongfu Liu, Sheng Li, Zhengming Ding, and Yun Fu. From ensemble clustering to multi-view clustering. In *IJCAI*, pages 2843–2849, 2017.

[26] Grigorios Tzortzis and Aristidis Likas. Kernel-based weighted multi-view clustering. In *IEEE International Conference on Data Mining (ICDM)*, pages 675–684, 2012.

[27] Weiran Wang, Raman Arora, Karen Livescu, and Jeff A. Bilmes. On deep multi-view representation learning. In *International Conference on Machine Learning (ICML)*, pages 1083–1092, 2015.

[28] Xiaobo Wang, Xiaojie Guo, Zhen Lei, Changqing Zhang, and Stan Z. Li. Exclusivity-consistency regularized multi-view subspace clustering. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–9, 2017.

[29] Rongkai Xia, Yan Pan, Lei Du, and Jian Yin. Robust multi-view spectral clustering via low-rank and decomposition. In *AAAI*, pages 2149–2155, 2014.

[30] Erkun Yang, Cheng Deng, Chao Li, Wei Liu, Jie Li, and Dacheng Tao. Shared predictive cross-modal deep quantization. *IEEE TNNLS*, 29.

[31] Changqing Zhang, Huazhu Fu, Si Liu, Guangcan Liu, and Xiaochun Cao. Low-rank tensor constrained multiview subspace clustering. In *IEEE International Conference on Computer Vision (ICCV)*, pages 1582–1590, 2015.

[32] Changqing Zhang, Qinghua Hu, Huazhu Fu, Pengfei Zhu, and Xiaochun Cao. Latent multi-view subspace clustering. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4333–4341, 2017.

[33] Zheng Zhang, Li Liu, Fumin Shen, Heng Tao Shen, and Ling Shao. Binary multi-view clustering. *IEEE TPAMI*, 2018.

[34] Handong Zhao, Zhengming Ding, and Yun Fu. Multi-view clustering via deep matrix factorization. In *AAAI*, pages 2921–2927, 2017.