Multiview Subspace Clustering via Co-Training Robust Data Representation

Jiyuan Liu, Xinwang Liu, *Senior Member, IEEE*, Yuexiang Yang, Xifeng Guo, Marius Kloft, *Senior Member, IEEE*, and Liangzhong He

***Abstract*— Taking the assumption that data samples are able to be reconstructed with the dictionary formed by themselves, recent multiview subspace clustering (MSC) algorithms aim to find a consensus reconstruction matrix via exploring complementary information across multiple views. Most of them directly operate on the original data observations without preprocessing, while others operate on the corresponding kernel matrices. However, they both ignore that the collected features may be designed arbitrarily and hard guaranteed to be independent and nonoverlapping. As a result, original data observations and kernel matrices would contain a large number of redundant details. To address this issue, we propose an MSC algorithm that groups samples and removes data redundancy concurrently. In specific, eigendecomposition is employed to obtain the robust data representation of low redundancy for later clustering. By utilizing the two processes into a unified model, clustering results will guide eigendecomposition to generate more discriminative data representation, which, as feedback, helps obtain better clustering results. In addition, an alternate and convergent algorithm is designed to solve the optimization problem. Extensive experiments are conducted on eight benchmarks, and the proposed algorithm outperforms comparative ones in recent literature by a large margin, verifying its superiority. At the same time, its effectiveness, computational efficiency, and robustness to noise are validated experimentally.**

***Index Terms*— Eigendecomposition, multiview clustering, robust representation, subspace clustering.**

I. Introduction

CLUSTERING is one of the most fundamental techniques and widely applied in numerous machine learning tasks, such as computer vision and bioinformatics [1]–[4]. Given the data drawn from a union of clusters, subspace clustering aims to reveal its intrinsic subspace structure [5]–[8]. Recent subspace clustering algorithms assume that each data sample is able to be reconstructed by a linear or affine combination of themselves [9]–[13]. Along with the reconstruction process, two critical components are produced, including the reconstruction error and the reconstruction matrix which is composed of the self-representation coefficients corresponding to each sample. Specifically, the representative sparse subspace clustering (SSC) [9], targeting at the high-dimensional data, imposes the l_1-norm on these two items so as to obtain the sparse data self-representations. Then, spectral clustering is applied on the learned self-representations to group the samples into different clusters [14], [15]. Compared with SSC, low-rank representation (LRR) [10], [16] holds that some samples are away from the underlying subspaces and therefore regularizes the columns of reconstruction error matrix to be sparse with $l_{2,1}$-norm. Other norms, such as Frobenius and kernel norm, are also used in the literature [17]–[23]. We adopt the Frobenius norm since it can group the highly correlated samples [17] and is able to be efficiently optimized.

In real applications, there are a large amount of multiview data and the aforementioned approaches are incapable of them. For instance, multiple semantic independent features, such as packet, TLS, and certificate features, are extracted in encrypted malware traffic detection [24]. Directly concatenating them is obviously the optimal way for further machine learning tasks. Therefore, multiview subspace clustering (MSC) algorithms are proposed to explore the complementary information among different views and achieve promising performances. Some methods produce the view-specific self-representations individually and then aggregate them into a consensus one or the final partition matrix [25]–[35]. For example, diversity-induced MSC (DiMSC) [25] employs the Hilbert–Schmidt independence criterion (HSIC) to measure the dependences between self-representations and minimize them to increase the diversity of underlying subspaces. Meanwhile, other approaches [36]–[40] reconstruct the data with a shared self-representation across all views. Wang *et al.* [36] notice that original data observations can be discomposed into two parts, including the shared latent representation which encodes the clustering details and view-specific deviations, such as noise. The proposed method follows the second type of method for its conciseness. The aforementioned approaches assume that data live on linear subspaces and directly adopt original data observations as input, while a few ones in literature adopt

Manuscript received August 1, 2020; revised December 27, 2020; accepted March 25, 2021. This work was supported in part by the National Key Research and Development Program of China under Project 2020AAA0107100; in part by the National Natural Science Foundation of China under Project 61922088, Project 61906020, and Project 61773392; in part by the Education Ministry-China Mobile Research Funding under Project MCM20170404; in part by the German Research Foundation (DFG) under Award KL 2698/2-1; and in part by the Federal Ministry of Science and Education (BMBF) under Award 01IS18051A and Award 031B0770E. *(Corresponding author: Xinwang Liu.)*

Jiyuan Liu, Xinwang Liu, Yuexiang Yang, and Xifeng Guo are with the College of Computer Science, National University of Defense Technology, Changsha 410072, China (e-mail: liujiyuan13@nudt.edu.cn; xinwangliu@nudt.edu.cn; yyx@nudt.edu.cn; guoxifeng1990@163.com).

Marius Kloft is with the Department of Computer Science, Technische Universität Kaiserslautern, 67653 Kaiserslautern, Germany (e-mail: kloft@cs.unikl.de).

Liangzhong He is with China Mobile (Su Zhou) Software Technology Company Ltd., Suzhou 215009, China (e-mail: heliangzhong@cmss.chinamobile.com).

This article has supplementary material provided by the authors and color versions of one or more figures available at https://doi.org/10.1109/TNNLS.2021.3069424.

Digital Object Identifier 10.1109/TNNLS.2021.3069424

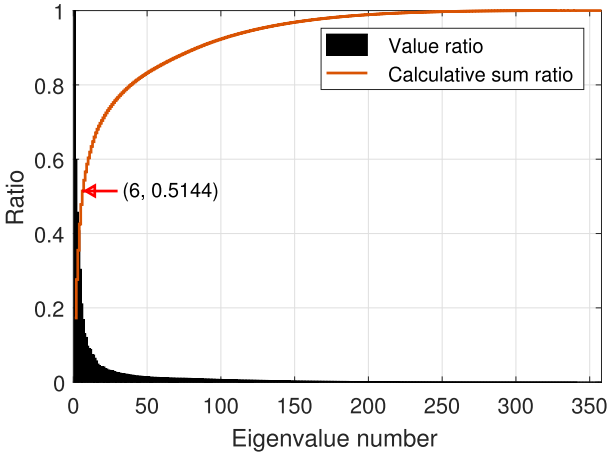

Fig. 1. Eigenvalue distribution of the inverse polynomial kernel matrix corresponding to the first view of Dermatology data set. The eigenvalues are sorted from large to small. The bar plot shows the times of each eigenvalue to the first one. Meanwhile, the curve plot presents the calculative sum of the sorted eigenvalues.

kernel trick to solve the nonlinear problem by producing the corresponding kernel matrices [13], [32], [34], [40]–[43]. For example, Patel and Vidal [41] motivated by the success of nonlinear representations in numerous machine learning tasks, first extend the SSC algorithm with kernel trick. Brbic and Kopriva [32] and Zhang *et al.* [34] kernelized their MSC algorithms by implicitly mapping data observations to reproducing kernel Hilbert spaces (RKHSs). Instead of directly using the primary kernels in MSC, Zhou *et al.* [40] constructed a neighbor kernel, which not only preserves the diagonal block structure but also enhances the robustness to noise and outliers, gaining a promising performance.

The two aforementioned types of inputs, including original data and corresponding kernel matrices, always contain redundant information, which is harmful to clustering performance. For original data observations, the collected features are designed arbitrarily in a large volume of applications and hard guaranteed to be independent or nonoverlapping. Sometimes, even the features designed by professionals do so. Meanwhile, the redundancy in original observations cannot be sufficiently removed by simply constructing corresponding kernel matrices. We perform eigendecomposition on the inverse polynomial kernel matrix generated from the first-view data observation of Dermatology data set (Dermatology is thoroughly described in Section IV). Resultant eigenvalues are plotted in Fig. 1. It can be seen that only a small number of eigenvalues are presented to be large and make the most of the eigenvalue sum, while the others are relatively small to zero, which indicates that kernel matrices consist of a large number of fruitless information.

In order to address this issue, we propose an elegant algorithm called MSC via co-training robust data representation (CoMSC). Its flowchart is presented in Fig. 2. Specifically, the data are first mapped by five kernel functions, including Gaussian, polynomial, linear, sigmoid, and inverse polynomial, into corresponding RKHSs. With the obtained kernel matrices, eigendecomposition technique is employed to remove redundant information in kernels and obtain robust data representations. Then, the MSC algorithm is adopted to construct the consensus self-representation via exploring complementary details in these learned representations. Nevertheless, we utilize these two processes into a single objective, where eigendecomposition provides MSC with robust representations; at the same time, MSC guides eigendecomposition to produce more suitable representations for clustering. With the robust view-specific representations and ideal consensus self-representation jointly optimized in this cyclic procedure, a satisfying clustering performance can be achieved. In addition, we design an alternate strategy to solve the resultant optimization problem efficiently. We also analyze its complexity and prove the convergence. Extensive experiments are conducted to evaluate its effectiveness, superiority, computational efficiency, and robustness to noise. The contributions are summarized as follows.

1) We provide a brief insight that original data observations contain a large number of redundant details, and simply preprocessing them into kernel matrices cannot remove the redundancy.
2) We propose an elegant MSC model by grouping data samples along with removing redundant information in inputs. Its effectiveness, superiority, and robustness to noise are validated experimentally.
3) We design an alternate algorithm to optimize the proposed model. This algorithm is validated to be efficient compared with recent MSC ones in the literature.

To the best of our knowledge, there are a few MSC methods concerning about data redundancy. Therefore, this article would encourage the community to consider the data quality when designing new multiview clustering algorithms. In addition, we propose to perform clustering and data preprocessing concurrently, which provides a new approach for researchers to improve the performance of their own clustering methods.

II. Related Work

A. Subspace Clustering

Given n data observations $\mathbf{X} \in \mathbb{R}^{d\times n}$ drawn from k clusters, subspace clustering algorithms aim to find reconstruction matrix $\mathbf{Z}$, which encodes data samples with the dictionary formed by themselves. Their general formulation can be presented as

$$\min_{\mathbf{Z}} \ \mathcal{L}(\mathbf{X}, \mathbf{XZ}) + \lambda \boldsymbol{\Omega}(\mathbf{Z}) \quad \text{s.t. } \mathbf{Z} \in \mathbb{R}^{n\times n} \tag{1}$$

where $\mathcal{L}(\cdot)$ and $\boldsymbol{\Omega}(\cdot)$ represent the regularization terms. Various norms are adopted in the literature and the most widely used ones are summarized in Table I. In real-world applications, noises and errors are often collected due to sensor failure or environment change. Therefore, an error matrix $\mathbf{E}$ is employed to capture them and the objective of least squares regression (LSR) presents

$$\begin{aligned} &\min_{\mathbf{Z}} \ \|\mathbf{E}\|_F + \lambda \|\mathbf{Z}\|_F \\ &\text{s.t. } \mathbf{X} = \mathbf{XZ} + \mathbf{E}, \quad \text{diag}(\mathbf{Z}) = \mathbf{0}, \ \mathbf{Z} \in \mathbb{R}^{n\times n}. \end{aligned} \tag{2}$$

The proposed algorithm is designed based on (2), for it can group the highly correlated samples and is able to be efficiently solved.

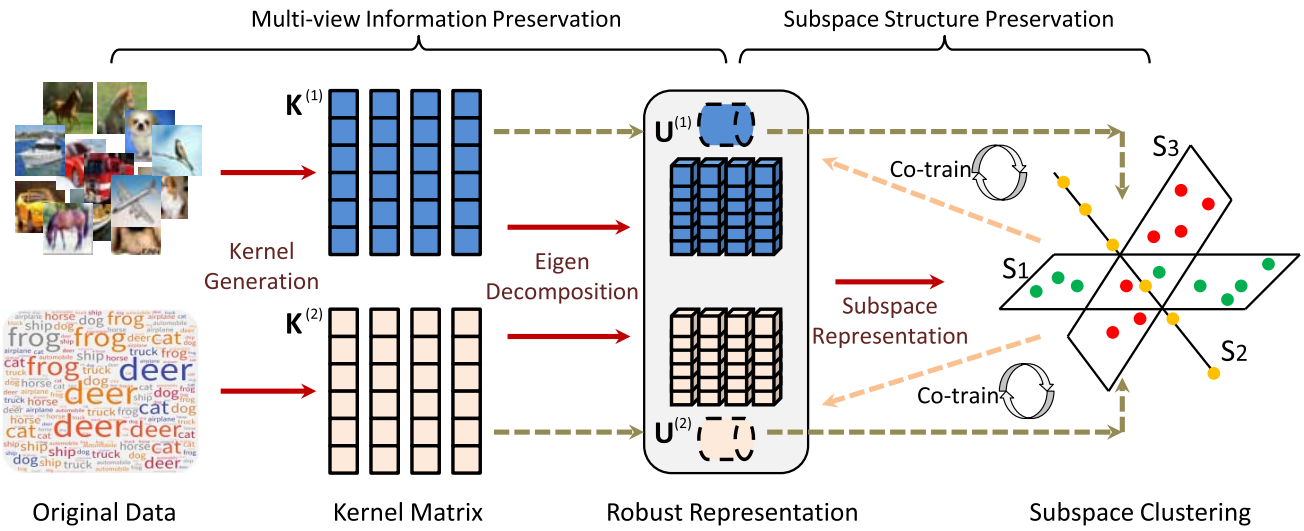

Fig. 2. Overview of the proposed method (taking the data of two views as an example). Two semantic parts are concerned, including the multiview information preservation and subspace structure preservation. Following the solid arrows, it can be observed that kernel matrices are first generated from the original data. Then, eigendecomposition is employed to obtain robust representations. Furthermore, the unified subspace representation is computed via utilizing the complementary information of multiple views. Following the dash arrows, the clustering details are delivered from kernel matrices to the robust representations and then to the consensus subspace structure. Next, the subspace structure guides the generation of purposive robust representations as feedback. Better self-representations are obtained along with the loop.

TABLE I
COMMON REGULARIZATIONS IN SUBSPACE CLUSTERING

Algorithm	$\mathcal{L}(\cdot)$	$\mathbf{\Omega}(\cdot)$
SSC [9]	$\{\mathbf{Z}\|\mathbf{X}=\mathbf{XZ}, diag(\mathbf{Z})=0\}$	$\|\mathbf{Z}\|_0$ or $\|\mathbf{Z}\|_1$
LRR [16]	$\{\mathbf{Z}\|\mathbf{X}=\mathbf{XZ}\}$	$\|\mathbf{Z}\|_*$
MSR [18]	$\{\mathbf{Z}\|\mathbf{X}=\mathbf{XZ}, diag(\mathbf{Z})=0\}$	$\|\mathbf{Z}\|_1+\sigma\|\mathbf{Z}\|_*$
LSR [17]	$\{\mathbf{Z}\|\mathbf{X}=\mathbf{XZ}, diag(\mathbf{Z})=0\}$	$\|\mathbf{Z}\|_F$

B. Multiview Subspace Clustering

Given data from V views $\{\mathbf{X}_v\}_{v=1}^V$, where $\mathbf{X}_v$ is drawn from $\mathbb{R}^{d_v\times n}$ and d_p is the feature dimension of pth view, MSC algorithms aim to find consensus reconstruction matrix $\mathbf{Z}$, which can be presented as

$$\min_{\{\mathbf{Z}_v\}_{i=1}^V,\mathbf{Z}} \mathcal{L}\big(\{\mathbf{X}_v,\ \ \mathbf{X}_v\mathbf{Z}_v\}_{v=1}^V\big)+\lambda\mathbf{\Omega}\big(\{\mathbf{Z}_v\}_{v=1}^V,\mathbf{Z}\big)$$
$$\text{s.t. } \mathbf{Z}_v\in\mathbb{R}^{n\times n}\quad \forall v\in\{1,2,\ldots,V\},\ \ \mathbf{Z}\in\mathbb{R}^{n\times n} \tag{3}$$

where $\mathbf{Z}_v$ is the vth reconstruction matrix, also termed data self-representation. Some MSC approaches assume that data observations of each view lie on the same subspaces and find $\mathbf{Z}$ via

$$\min_{\mathbf{Z}}\ \mathcal{L}\big(\{\mathbf{X}_v,\mathbf{X}_v\mathbf{Z}\}_{v=1}^V\big)+\lambda\mathbf{\Omega}(\mathbf{Z})\quad \text{s.t. } \mathbf{Z}\in\mathbb{R}^{n\times n}. \tag{4}$$

It is obvious that the model inputs in (1)–(4) are the original data observations, which are sometimes centered or normalized [30]. However, the quality of these observations is hard guaranteed in real-world data sets. For instance, the data features are designed by nonprofessionals or even arbitrarily, leading to information redundancy. These data of poor quality severely affect the performances of MSC algorithms. Kernel matrix is another natural form of data observation and can be directly adopted as input by simply substituting $\mathbf{X}$ in the aforementioned models. However, this does not remove the information redundancy and improve data quality, as shown in Fig. 1.

III. PROPOSED ALGORITHM

A. Objective

In order to remove the redundancy in the two types of inputs, i.e., original data observations and corresponding kernel matrices, we first define several kernel mappings $\{\phi_s(\cdot)\}_{s=1}^S$. For the vth view, the kernel matrices are computed as

$$\mathbf{K}_s^{(v)}(i,j)=\phi_s\left(\mathbf{x}_i^{(v)}\right)^\top\phi_s\left(\mathbf{x}_j^{(v)}\right) \tag{5}$$

in which $i,j\in\{1,2,\ldots,n\}$ represents the sample indexes. In this way, m corresponding kernel matrices are obtained as $\{\mathbf{K}_p\}_{p=1}^m$, s.t. $m=S*V$. However, the generated kernel matrices contain a large volume of redundant details. Fig. 1 shows the eigenvalue distribution of the inverse polynomial kernel matrix corresponding to first view of Dermatology. As claimed in [44, Sec. 4.2], the eigenvector corresponding to a larger eigenvalue carries more discriminative information. If taking the eigenvalue to roughly measure the volume of discriminative information in a corresponding eigenvector, we can see that top-50 eigenvectors keep more than 80% kernel details. Nevertheless, there are six classes in Dermatology. However, top-6 eigenvectors only contain 51.44% kernel details. In sum, two observations can be concluded:

1) The relationships among data samples are only contained in a small proportion of eigenvectors, while most of eigenvectors are redundant and should be removed.

2) It is not ideal to fix the size of the robust data representations as $\mathbb{R}^{k\times n}$. Instead, matrices of size $\mathbb{R}^{c\times n}$, where $c > k$ should be employed.

Therefore, we employ the eigenvectors corresponding to c largest eigenvalues as the robust data representation, contributing to

$$\begin{aligned}\mathbf{U}^* = \arg\max_{\mathbf{U}}\ & \mathrm{Tr}\big(\mathbf{U}\mathbf{K}\mathbf{U}^\top\big)\\ \text{s.t. }& \mathbf{U}\mathbf{U}^\top = \mathbf{I},\quad \mathbf{U}\in\mathbb{R}^{c\times n}\end{aligned}\tag{6}$$

which exhibits two merits.

1) $\mathbf{U}^*$ keeps the most profitable details in kernel matrix.
2) The orthogonal constraint on $\mathbf{U}^*$ ensures the representations living in low-rank spaces, which benefits the afterward subspace clustering process.

Then, we extend the LSR algorithm [17] in (2) into multiview setting following the framework in (4). The multiview LSR objective is given as

$$\begin{aligned}\min_{\mathbf{Z},\ \boldsymbol{\beta}}\ & \lambda\|\mathbf{Z}\|_F^2 + \sum_{p=1}^{m}\beta_p\|\mathbf{X}_p - \mathbf{X}_p\mathbf{Z}\|_F^2\\ \text{s.t. }& \mathrm{diag}(\mathbf{Z}) = \mathbf{0},\quad \mathbf{Z}\in\mathbb{R}^{n\times n},\ \boldsymbol{\beta}^{\frac{1}{2}\top}\mathbf{1} = 1,\ \boldsymbol{\beta}\in\mathbb{R}_+^m.\end{aligned}\tag{7}$$

Nevertheless, (7) can be efficiently optimized and theoretically guaranteed to be convergent since a closed-form solution with respect to $\mathbf{Z}$ can be obtained. By substituting the input of (7), i.e., $\{\mathbf{X}_p\}_{p=1}^m$, with the preprocessed robust data representations in (6), i.e., $\{\mathbf{U}_p\}_{p=1}^m$, and utilizing these two processes into one framework, the proposed objective of CoMSC algorithm is obtained as

$$\begin{aligned}\min_{\{\mathbf{U}_p\}_{p=1}^m,\ \mathbf{Z},\ \boldsymbol{\beta},\ \boldsymbol{\gamma}}\ & \lambda\|\mathbf{Z}\|_F^2 + \sum_{p=1}^{m}\beta_p\|\mathbf{U}_p - \mathbf{U}_p\mathbf{Z}\|_F^2\\ & - \sum_{p=1}^{m}\gamma_p\mathrm{Tr}\big(\mathbf{U}_p\mathbf{K}_p\mathbf{U}_p^\top\big)\\ \text{s.t. }& \mathrm{diag}(\mathbf{Z}) = \mathbf{0},\quad \mathbf{Z}\in\mathbb{R}^{n\times n},\ \mathbf{U}_p\mathbf{U}_p^\top = \mathbf{I},\ \mathbf{U}_p\in\mathbb{R}^{c\times n}\\ & \boldsymbol{\beta}^{\frac{1}{2}\top}\mathbf{1} = 1,\quad \boldsymbol{\beta}\in\mathbb{R}_+^m,\ \boldsymbol{\gamma}^\top\boldsymbol{\gamma} = 1,\ \boldsymbol{\gamma}\in\mathbb{R}_+^m\end{aligned}\tag{8}$$

in which the tradeoff between data representation learning and MSC is set to 1, for that they are considered to be equally important. The coefficients $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$ indicate the importance of each view and are imposed on different norms to ensure the convexity [45]. In sum, once the robust data representations $\{\mathbf{U}_p\}_{p=1}^m$ are built from the corresponding kernel matrices $\{\mathbf{K}_p\}_{p=1}^m$, they are adopted in clustering to build a consensus reconstruction matrix $\mathbf{Z}$. As feedback, the clustering process guides the data representation learning to produce more purposive ones. With the close collaboration of these two processes, a promising performance can be achieved.

B. Optimization

To solve the proposed objective in (8), we design an alternate optimization strategy. In specific, each unknown variable is solved while fixing the others fixed in each step. By cyclically optimizing every variable, the procedure will converge to a local minimum. We present the optimization strategy in detail as follows.

1) $\mathbf{Z}$-Subproblem: Fixing $\{\mathbf{U}_p\}_{p=1}^m$, $\boldsymbol{\beta}$, and $\boldsymbol{\gamma}$, the optimal $\mathbf{Z}^*$ can be solved via solving the following optimization problem:

$$\begin{aligned}\min_{\mathbf{Z}}\ & \lambda\|\mathbf{Z}\|_F^2 + \sum_{p=1}^{m}\beta_p\|\mathbf{U}_p - \mathbf{U}_p\mathbf{Z}\|_F^2\\ \text{s.t. }& \mathrm{diag}(\mathbf{Z}) = \mathbf{0},\quad \mathbf{Z}\in\mathbb{R}^{n\times n}.\end{aligned}\tag{9}$$

Observing that the diagonal of $\mathbf{Z}$ is compulsively constrained to zeros, we remove the ith column of $\mathbf{U}_p = \{\mathbf{u}_p^{(i)}\}_{i=1}^n \in \mathbb{R}^{c\times n}$ to obtain $\mathbf{H}_p^{(i)} = \{\mathbf{u}_p^{(1)},\ldots,\mathbf{u}_p^{(i-1)},\mathbf{u}_p^{(i+1)},\ldots,\mathbf{u}_p^{(n)}\} \in \mathbb{R}^{c\times(n-1)}$ and optimize each column of $\mathbf{Z}$ separately as

$$\begin{aligned}\min_{\mathbf{z}_i}\ & \lambda\|\mathbf{z}_i\|_F^2 + \sum_{p=1}^{m}\beta_p\big\|\mathbf{u}_p^{(i)} - \mathbf{H}_p^{(i)}\mathbf{z}_i\big\|_F^2\\ \text{s.t. }& \mathbf{z}_i\in\mathbb{R}^{n-1}\end{aligned}\tag{10}$$

which can be transformed to

$$\begin{aligned}\min_{\mathbf{z}_i}\ & \mathrm{Tr}\big(\mathbf{E}_i\mathbf{z}_i\mathbf{z}_i^\top\big) - 2\left(\sum_{p=1}^{m}\beta_p\mathbf{u}_p^{(i)\top}\mathbf{H}_p^{(i)}\right)\mathbf{z}_i\\ \text{s.t. }& \mathbf{E}_i = \lambda\mathbf{I} + \sum_{p=1}^{m}\beta_p\mathbf{H}_p^{(i)\top}\mathbf{H}_p^{(i)},\quad \mathbf{z}_i\in\mathbb{R}^{n-1}.\end{aligned}\tag{11}$$

It is easy to prove that $\mathbf{E}_i$ is positively defined, and thus, (11) is convex and has a global minimum. By setting its deviation to zero, the optimal $\mathbf{z}_i^*$ is obtained as

$$\begin{aligned}\mathbf{z}_i^* &= \mathbf{E}_i^{-1}\left(\sum_{p=1}^{m}\beta_p\mathbf{H}_p^{(i)\top}\mathbf{u}_p^{(i)}\right)\\ \text{s.t. }\mathbf{E}_i &= \lambda\mathbf{I} + \sum_{p=1}^{m}\beta_p\mathbf{H}_p^{(i)\top}\mathbf{H}_p^{(i)}.\end{aligned}\tag{12}$$

However, it is of high computation complexity to obtain the optimal solution via (12) since an inverse matrix is required to be computed for every column of $\mathbf{Z}$. Defining $\mathbf{D} = (\lambda\mathbf{I} + \sum_{p=1}^m\beta_p\mathbf{U}_p^\top\mathbf{U}_p)^{-1}$ and $\mathbf{U}_p\mathbf{P} = [\mathbf{H}_p^{(i)}, \mathbf{u}_p^{(i)}]$ where $\mathbf{P}$ is a permutation matrix, $\mathbf{P}^\top\mathbf{P} = \mathbf{P}\mathbf{P}^\top = \mathbf{I}$, we have

$$\begin{aligned}\mathbf{P}^\top\mathbf{D}\mathbf{P} &= \left[\mathbf{P}^\top\left(\lambda_1\mathbf{I} + \sum_{p=1}^{m}\beta_p\mathbf{U}_p^\top\mathbf{U}_p\right)\mathbf{P}\right]^{-1}\\ &= \begin{bmatrix}\lambda\mathbf{I} + \sum_{p=1}^{m}\beta_p\mathbf{H}_p^{(i)\top}\mathbf{H}_p^{(i)} & \sum_{p=1}^{m}\beta_p\mathbf{H}_p^\top\mathbf{u}_p^{(i)}\\ \sum_{p=1}^{m}\beta_p\mathbf{u}_p^{(i)\top}\mathbf{H}_p & \lambda + \sum_{p=1}^{m}\beta_p\mathbf{u}_p^{(i)\top}\mathbf{u}_p^{(i)}\end{bmatrix}^{-1}\\ &= \begin{bmatrix}\mathbf{E}_i^{-1} & 0\\ 0 & 0\end{bmatrix} + \sigma_i\begin{bmatrix}\mathbf{b}_i\mathbf{b}_i^\top & \mathbf{b}_i\\ \mathbf{b}_i^\top & 1\end{bmatrix}\end{aligned}\tag{13}$$

in which

$$\begin{aligned}\mathbf{b}_i &= -\mathbf{z}_i^*\\ \sigma_i &= \lambda + \sum_{p=1}^{m}\beta_p\mathbf{u}_p^{(i)\top}\mathbf{u}_p^{(i)} - \sum_{p=1}^{m}\beta_p\mathbf{u}_p^{(i)\top}\big(\mathbf{H}_p^{(i)}\mathbf{E}_i^{-1}\mathbf{H}_p^{(i)\top}\big)\mathbf{u}_p^{(i)}.\end{aligned}\tag{14}$$

The last step holds for Woodbury formula [46]. It can be seen that $\mathbf{z}_i^* = -\mathbf{b}_i$ from (14). At the same time, we can obtain from the definition of $\mathbf{P}$ that

$$\mathbf{Z}^*(j,i) = \begin{cases} -\mathbf{D}(j,i)/\mathbf{D}(i,i), & j \neq i \\ 0, & j = i \end{cases} \tag{15}$$

which can be rewritten as

$$\mathbf{Z}^* = -\mathbf{D}(\mathrm{diag}(\mathbf{D}))^{-1}, \quad \mathrm{diag}(\mathbf{Z}^*) = \mathbf{0}. \tag{16}$$

2) $\mathbf{U}$*-Subproblem:* It is obvious that the data representations $\{\mathbf{U}_p\}_{p=1}^m$ are independent of each other. Thus, they are able to be optimized separately. With fixing $\{\mathbf{U}_q\}_{q=1,q\neq p}^m$, $\mathbf{Z}$, $\boldsymbol{\beta}$, and $\boldsymbol{\gamma}$, the proposed objective can be reduced to

$$\begin{aligned} &\min_{\mathbf{U}_p} \ \beta_p \|\mathbf{U}_p - \mathbf{U}_p\mathbf{Z}\|_F^2 - \gamma_p \mathrm{Tr}\big(\mathbf{U}_p\mathbf{K}_p\mathbf{U}_p^\top\big) \\ &\text{s.t. } \mathbf{U}_p\mathbf{U}_p^\top = \mathbf{I}, \quad \mathbf{U}_p \in \mathbb{R}^{c\times n} \end{aligned} \tag{17}$$

which can be further transformed into

$$\begin{aligned} &\max_{\mathbf{U}_p} \ \mathrm{Tr}\big(\mathbf{U}_p\mathbf{M}_p\mathbf{U}_p^\top\big) \\ &\text{s.t. } \mathbf{M}_p = 2\beta_p\mathbf{Z}^\top - \beta_p\mathbf{Z}\mathbf{Z}^\top + \gamma_p\mathbf{K}_p \\ &\quad \mathbf{U}_p\mathbf{U}_p^\top = \mathbf{I}, \quad \mathbf{U}_p \in \mathbb{R}^{c\times n}. \end{aligned} \tag{18}$$

Equation (18) can be efficiently solved via eigendecomposition where $\mathbf{U}_p^*$ is the matrix of eigenvectors corresponding to c largest eigenvalues [37], [47], [48].

3) $\boldsymbol{\beta}$*-Subproblem:* Fixing $\{\mathbf{U}_p\}_{p=1}^m$, $\mathbf{Z}$ and $\boldsymbol{\gamma}$, the objective with respect to $\boldsymbol{\beta}$ can be reduced to

$$\begin{aligned} &\min_{\boldsymbol{\beta}} \ \boldsymbol{\beta}^\top \boldsymbol{\nu} \\ &\text{s.t. } \nu_p = \|\mathbf{U}_p - \mathbf{U}_p\mathbf{Z}\|_F^2, \quad \boldsymbol{\beta}^{\frac{1}{2}\top}\mathbf{1} = 1, \ \boldsymbol{\beta} \in \mathbb{R}_+^m. \end{aligned} \tag{19}$$

According to the Cauchy–Schwarz inequality

$$\begin{aligned} \big(\boldsymbol{\beta}^\top\boldsymbol{\nu}\big)\left(\sum_{p=1}^m \frac{1}{\nu_p}\right) &= \left(\sum_{p=1}^m \Big(\sqrt{\beta_p}\sqrt{\nu_p}\Big)^2\right)\left(\sum_{p=1}^m \left(\frac{1}{\sqrt{\nu_p}}\right)^2\right) \\ &\geq \left(\sum_{p=1}^m \sqrt{\beta_p}\right)^2 = 1 \end{aligned} \tag{20}$$

in which the equality holds when

$$\nu_1\sqrt{\beta_1} = \nu_2\sqrt{\beta_2} = \cdots = \nu_m\sqrt{\beta_m}. \tag{21}$$

Considering the extra regularization on $\boldsymbol{\beta}$, i.e., $\boldsymbol{\beta}^{(1/2)\top}\mathbf{1} = 1$, we solve the optimization problem as

$$\beta_p^* = 1 \Big/ \left(\nu_p \sum_{q=1}^m \frac{1}{\nu_q}\right)^2. \tag{22}$$

4) $\boldsymbol{\gamma}$*-Subproblem:* Fixing $\{\mathbf{U}_p\}_{p=1}^m$, $\mathbf{Z}$ and $\boldsymbol{\beta}$, the objective with respect to γ can be reduced to

$$\begin{aligned} &\max_{\boldsymbol{\gamma}} \ \boldsymbol{\gamma}^\top\boldsymbol{\nu} \\ &\text{s.t. } \nu_p = \mathrm{Tr}\big(\mathbf{U}_p\mathbf{K}_p\mathbf{U}_p^\top\big), \quad \boldsymbol{\gamma}^\top\boldsymbol{\gamma} = 1, \ \boldsymbol{\gamma} \in \mathbb{R}_+^m. \end{aligned} \tag{23}$$

According to the Cauchy–Schwarz inequality

$$\begin{aligned} \big(\boldsymbol{\gamma}^\top\boldsymbol{\nu}\big)^2 &= \left(\sum_{p=1}^m \gamma_p\nu_p\right)^2 \leq \left(\sum_{p=1}^m \gamma_p^2\right)\left(\sum_{p=1}^m \nu_p^2\right) \\ &= \big(\boldsymbol{\gamma}^\top\boldsymbol{\gamma}\big)\big(\boldsymbol{\nu}^\top\boldsymbol{\nu}\big) = \boldsymbol{\nu}^\top\boldsymbol{\nu} \end{aligned} \tag{24}$$

in which the equality holds when

$$\gamma_1/\nu_1 = \gamma_2/\nu_2 = \cdots = \gamma_m/\nu_m. \tag{25}$$

Considering the extra regularization on $\boldsymbol{\gamma}$, i.e., $\boldsymbol{\gamma}^\top\boldsymbol{\gamma} = 1$, we solve the optimization problem as

$$\gamma_p^* = \nu_p \Big/ \left(\sum_{q=1}^m \nu_q^2\right)^{1/2}. \tag{26}$$

An overview of the alternate optimization strategy is outlined in Algorithm 1.

Algorithm 1 MSC via CoMSC

Require: data $\{\mathbf{X}_v\}_{v=1}^V$, size of robust data representation c and parameter λ.
Ensure: consensus reconstruction matrix $\mathbf{Z}$.
1: Generate the kernel matrices $\{\mathbf{K}_p\}_{p=1}^m$ from $\{\mathbf{X}_v\}_{v=1}^V$.
2: Initialize $\{\mathbf{U}_p\}_{p=1}^m$, $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$.
3: **while** $(obj^{t-1} - obj^t)/obj^t \leq \sigma$ **do**
4: Update $\mathbf{Z}$ by solving Eq. (16).
5: Update $\{\mathbf{U}_p\}_{p=1}^m$ with Eq. (18).
6: Update $\boldsymbol{\beta}$ with Eq. (22).
7: Update $\boldsymbol{\gamma}$ with Eq. (26).
8: $t = t + 1$.
9: Calculate objective value obj^t with Eq. (8).
10: **end while**

C. Convergence and Complexity

Most subspace clustering methods, such as [49], cannot be proved to converge, while the convergence of our proposed algorithm is able to be theoretically guaranteed. For the ease of expression, we reformulate the objective into

$$\min_{\mathbf{Z},\ \{\mathbf{U}_p\}_{p=1}^m,\ \boldsymbol{\beta},\ \boldsymbol{\gamma}} \mathcal{J}\Big(\mathbf{Z},\ \big\{\mathbf{U}_p\big\}_{p=1}^m,\ \boldsymbol{\beta},\ \boldsymbol{\gamma}\Big). \tag{27}$$

As shown in Algorithm 1, the optimization strategy consists of four iterative parts, i.e., $\mathbf{U}$, $\mathbf{Z}$, $\boldsymbol{\beta}$, and $\boldsymbol{\gamma}$ subproblems. Correspondingly, the analysis of each subproblem on convergence is listed as follows. Note that superscript t represents the optimization at round t.

1) $\mathbf{Z}$-*Subproblem:* Given $\{\mathbf{U}_p\}_{p=1}^{m\,(t)}$, $\boldsymbol{\beta}^{(t)}$, and $\boldsymbol{\gamma}^{(t)}$, we can obtain $\mathbf{Z}^{(t+1)}$ via optimizing (16), resulting in

$$\mathcal{J}\left(\mathbf{Z}^{(t)}, \{\mathbf{U}_p\}_{p=1}^{m(t)}, \boldsymbol{\beta}^{(t)}, \boldsymbol{\gamma}^{(t)}\right) \geq \mathcal{J}\left(\mathbf{Z}^{(t+1)}, \{\mathbf{U}_p\}_{p=1}^{m\,(t)}, \boldsymbol{\beta}^{(t)}, \boldsymbol{\gamma}^{(t)}\right). \quad (28)$$

2) $\mathbf{U}$-*Subproblem:* Given $\mathbf{Z}^{(t+1)}$, $\boldsymbol{\beta}^{(t)}$, and $\boldsymbol{\gamma}^{(t)}$, we can obtain $\{\mathbf{U}_p\}_{p=1}^{m\,(t+1)}$ via optimizing (18), resulting in

$$\mathcal{J}\left(\mathbf{Z}^{(t+1)}, \{\mathbf{U}_p\}_{p=1}^{m\,(t)}, \boldsymbol{\beta}^{(t)}, \boldsymbol{\gamma}^{(t)}\right) \geq \mathcal{J}\left(\mathbf{Z}^{(t+1)}, \{\mathbf{U}_p\}_{p=1}^{m(t+1)}, \boldsymbol{\beta}^{(t)}, \boldsymbol{\gamma}^{(t)}\right). \quad (29)$$

3) $\boldsymbol{\beta}$-*Subproblem:* Given $\mathbf{Z}^{(t+1)}$, $\{\mathbf{U}_p\}_{p=1}^{m\,(t+1)}$, and $\boldsymbol{\gamma}^{(t)}$, we can obtain $\boldsymbol{\beta}^{(t+1)}$ via optimizing (22), resulting in

$$\mathcal{J}\left(\mathbf{Z}^{(t+1)}, \{\mathbf{U}_p\}_{p=1}^{m(t+1)}, \boldsymbol{\beta}^{(t)}, \boldsymbol{\gamma}^{(t)}\right) \geq \mathcal{J}\left(\mathbf{Z}^{(t+1)}, \{\mathbf{U}_p\}_{p=1}^{m\,(t+1)}, \boldsymbol{\beta}^{(t+1)}, \boldsymbol{\gamma}^{(t)}\right). \quad (30)$$

4) $\boldsymbol{\gamma}$-*Subproblem:* Given $\mathbf{Z}^{(t+1)}$, $\{\mathbf{U}_p\}_{p=1}^{m\,(t+1)}$, and $\boldsymbol{\beta}^{(t+1)}$, we can obtain $\boldsymbol{\gamma}^{(t+1)}$ via optimizing (26), resulting in

$$\mathcal{J}\left(\mathbf{Z}^{(t+1)}, \{\mathbf{U}_p\}_{p=1}^{m\,(t+1)}, \boldsymbol{\beta}^{(t+1)}, \boldsymbol{\gamma}^{(t)}\right) \geq \mathcal{J}\left(\mathbf{Z}^{(t+1)}, \{\mathbf{U}_p\}_{p=1}^{m\,(t+1)}, \boldsymbol{\beta}^{(t+1)}, \boldsymbol{\gamma}^{(t+1)}\right). \quad (31)$$

To sum up (28)–(31), the following inequality holds that:

$$\mathcal{J}\left(\mathbf{Z}^{(t)}, \{\mathbf{U}_p\}_{p=1}^{m\,(t)}, \boldsymbol{\beta}^{(t)}, \boldsymbol{\gamma}^{(t)}\right) \geq \mathcal{J}\left(\mathbf{Z}^{(t+1)}, \{\mathbf{U}_p\}_{p=1}^{m\,(t+1)}, \boldsymbol{\beta}^{(t+1)}, \boldsymbol{\gamma}^{(t+1)}\right) \quad (32)$$

which indicates that the objective value monotonically decreases along with iterations. Meanwhile,

$$\mathcal{J} \geq 0 + 0 - \sum_{p=1}^{m} \gamma_p \sum_{i=1}^{n} \sigma_{pi} \geq -\sum_{p,i=1}^{m,n} \sigma_{pi} \quad (33)$$

in which $\{\sigma_{pi}\}_{p,i=1}^{m,n}$ are the eigenvalues of the kernel matrices $\{\mathbf{K}_p\}_{p=1}^{m}$. Equation (33) illustrates that the objective is lower bounded. Therefore, the proposed algorithm is theoretically convergent.

Complexity analysis is conducted corresponding to the four subproblems. In $\mathbf{U}$-subproblem, an eigendecomposition is performed on each view, and thus, the complexity is $O(mn^3)$. For updating $\mathbf{Z}$, the LU decomposition is employed to compute the inverse of $\lambda\mathbf{I} + \sum_{p=1}^{m} \beta_p \mathbf{U}_p^\top \mathbf{U}_p$, which has a complexity of $O(n^3)$. While solving $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$, their complexities are $O(cn^2)$. Assuming that t iterations are needed to converge, the overall complexity is $O(tmn^3)$.

IV. Experiment

A. Experiment Setting

We employ eight data sets to evaluate the effectiveness, superiority, and efficiency of the proposed algorithm, including the following.

TABLE II
Specifications of the Used Data Sets

Dataset	Number of Samples	Views	Clusters
Dermatology	358	2	6
BBCSport	282	3	5
WebKB	265	4	5
Prokaryotic	551	3	4
Reuters	1200	5	6
Wiki	2866	2	10
Caltech7	1474	6	7
HandWritten	2000	6	10

1) Dermatology[1] is used for the diagnosis of erythemato-squamous diseases, including psoriasis, seborrheic dermatitis, lichen planus, pityriasis rosea, chronic dermatitis, and pityriasis rubra pilaris.
2) WebKB[2] consists of webpages that are described from two aspects, i.e., contents and links. They are collected from four universities and Wisconsin is selected.
3) BBCSport[3] is constructed from single-view sport corpora by splitting news articles into segments.
4) Prokaryotic[4] contains multiple prokaryotic species described with heterogeneous multiview data, including textual data and different genomic representations.
5) Reuters[5] contains 2000 documents each described with five languages, including English, French, German, Italian, and Spanish.
6) Wiki[6] contains 2866 selected sections from the Wikipedia's featured article collection where word and SIFT histogram are used for text and image, respectively.
7) Caltech7 is a subset of Caltech101,[7] which collects a large number of object pictures belonging to 101 categories. Insides, seven popular classes, including face, motorbike, dollar bill, garfield, snoopy, stop sign and windsor chair, are selected.
8) HandWritten[8] consists of features of handwritten numerals (0–9) extracted from a collection of Dutch utility maps.

Their specifications are summarized in Table II.

Meanwhile, the proposed algorithm is compared with MSC algorithms in recent literature. In specific, two baselines and another ten algorithms are given in the following.

1) LSRb [17] (*baseline*) performs subspace clustering for each view and the best result is reported.
2) LSRc [17] (*baseline*) performs subspace clustering by simply contacting all views into a single one.

[1] https://archive.ics.uci.edu/ml/datasets/dermatology
[2] http://lig-membres.imag.fr/grimal/data.html
[3] http://mlg.ucd.ie/datasets/segment.html
[4] https://github.com/mbrbic/MultiViewLRSSC/tree/master/datasets
[5] http://lig-membres.imag.fr/grimal/data.html
[6] http://www.svcl.ucsd.edu/projects/crossmodal/
[7] http://www.vision.caltech.edu/Image_Datasets/Caltech101/
[8] https://archive.ics.uci.edu/ml/datasets/Multiple+Features

TABLE III
PERFORMANCE COMPARISON OF XLRS, KLRS, RLRS, AND COMSC

	Algorithm	Dermatology	WebKB	BBCSport	Prokaryotic	Reuters	Wiki	Caltech7	HandWritten
ACC	XLRS	74.30	49.06	79.79	54.63	45.58	53.25	39.62	72.45
	KLRS	86.59	52.08	81.92	70.96	52.83	54.71	49.66	64.25
	RLRS	96.65	54.34	**88.30**	81.85	51.42	54.75	64.72	93.00
	CoMSC	**97.21**	**55.85**	**88.30**	**84.57**	**54.42**	**58.51**	**69.88**	**94.25**
NMI	XLRS	61.33	11.23	68.76	27.80	23.96	52.03	36.09	67.05
	KLRS	81.55	33.09	68.48	38.53	**33.21**	53.30	44.62	67.48
	RLRS	92.57	**36.09**	**78.20**	49.94	31.43	53.39	55.92	86.70
	CoMSC	**93.76**	**36.09**	**78.20**	**53.86**	32.81	**54.69**	**56.30**	**90.05**
Purity	XLRS	74.30	55.47	86.53	65.15	47.33	60.78	82.63	72.45
	KLRS	86.59	70.94	85.11	77.86	53.08	61.79	83.72	66.65
	RLRS	96.65	**73.96**	**88.65**	84.39	53.17	61.83	87.72	93.00
	CoMSC	**97.21**	72.83	**88.65**	**85.84**	**54.58**	**62.98**	**87.65**	**94.25**

3) RMSC [50] first builds a transition probability matrix corresponding to each view and adopts them to recover a shared low-rank transition probability matrix that is used as an input to the standard Markov chain method for clustering.
4) DiMSC [25] employs HSIC to measure the dependences between self-representations and minimize them to increase the diversity of underlying subspaces.
5) LT-MSC [26] regards the subspace representation matrices of multiple views as a tensor and regularizes it with low-rank regularization for the affinity matrix.
6) MSSC [38] first exploits the self-expressiveness in each data view and then enforces the common representation across all views.
7) ECMSC [30] harnesses the complementary information between different representations by introducing a novel position-aware exclusivity term and a consistency term.
8) LMSC [49] assumes that all data views can be reconstructed by the affine transformations of one latent representation and co-train these parameterized transformations with the afterward subspace clustering.
9) LRSSC [32] balances the agreement across different views while encouraging sparsity and low rankness of the solution.
10) CSMSC [51] assumes that the self-representations consist of view-consistent part and view-specific parts concurrently. By separately regularizing the two parts, the algorithm obtains a satisfying performance.
11) FMR [52] utilizes complementary information by exploring nonlinear and high-order correlations among different views with HSIC.
12) PMSC [53] fuses multiview information in partition level and assigns larger weights to the partitions close to the consensus one.

We directly adopt the codes of comparative methods from authors' websites, perform grid search in the parameter sets recommended in their papers, and report the best results. For the proposed algorithm, we apply five kernel mappings, including Gaussian, polynomial, linear, sigmoid, and inverse polynomial, on the original data observations to obtain the corresponding kernel matrices. In addition, the tradeoff λ and the size of robust data representations, c, are, respectively, searched from $2.\hat{}\{-10, -8, \ldots, 10\}$ and $\{k, 2k, \ldots, 20k\}$, where k is the number of clusters. In the same way, the best results are reported. Furthermore, we open the code on Github.[9]

B. Experiment Results

In the following experiments, three evaluation metrics, including accuracy (ACC), normalized mutual information (NMI), and purity, are adopted to measure the performances of the proposed algorithm.

1) Effectiveness: In order to show the effectiveness of the proposed robust representation, we first conduct experiments on four objectives. They are given in the following.

1) *Multiview Subspace Clustering via Least Squares Regression (XLSR):* Its objective is presented in (7) and the original data observations are adopted as input.
2) *Multiview Subspace Clustering via Least Squares Regression With Kernel Matrices (KLSR):* It has the same objective as XLSR in (7), but the kernel matrices of five types are adopted as input.
3) *Multiview Subspace Clustering via Least Squares Regression With Robust Data Representations (RLSR):* The objective is the same as XLSR in (7), but the robust data representations generated from the eigendecomposition of corresponding kernel matrices are adopted as input.
4) *MSC via CoMSC:* It is the proposed algorithm in this article and the objective is shown in (8).

The results are presented in Table III where the best results are marked in bold. We have the following observations.

1) KLRS outperforms XLRS on seven data sets except Handwritten by 12.29%, 2.13%, 3.02%, 16.33%, 7.25%,

[9]https://github.com/liujiyuan13/CoMSC-code_release

1.47%, and 10.04% in ACC. In NMI, 20.22% on Dermatology, 21.86% on WebKB, 10.73% on Prokaryotic, 9.25% on Reuters, and 8.53% on Caltech7 are observed, with the others similar or slightly worse than XLRS by less than 1.00%. In purity, 12.29% on dermatology, 15.47% on WebKB, 12.71% on Prokaryotic, and 5.76% on Reuters are presented. These observations illustrate that most of the chosen data sets live on the nonlinear subspaces and better performances can be obtained by simply adopting the kernel matrices as input. At the same time, XLRS exceeds KLRS on HandWritten by 8.20% in ACC and 5.80% in purity, which indicates that the data set is of linear subspace structure. Therefore, it is not always the optimal choice to adopt kernel trick on original data arbitrarily without sufficiently preinvestigation.

2) RLRS outperforms KLRS on seven data sets, including Dermatology, WebKB, BBCSport, Prokaryotic, Wiki, Caltech7, and Handwritten, by 10.06%, 2.26%, 6.38%, 10.89%, 0.03%, 15.06%, and 28.75% in ACC, respectively. Although the performances of RLRS are lower than those of KLRS for Reuters, the gaps are relatively small, i.e., 1.41% in ACC. We can conclude that the eigenvectors corresponding to larger eigenvalues are better representations for subspace clustering, which proves our claim that eigendecomposition is able to remove the redundancy in the original data observations and kernel matrices.
3) Meanwhile, it can be observed that CoMSC outperforms RLSR by 0.56%, 1.51%, 0.00%, 2.72%, 3.00%, 3.77%, 5.16%, and 1.25% in ACC. The results of Dermatology and BBCSport are not so significant. This would be caused by the simplicity of the two small data sets on which high clustering accuracies have been obtained and less room is left for further improvement. Nevertheless, CoMSC shows satisfying improvements on big data sets, such as Rueters, Wiki, and so on. This indicates that the subspace clustering can guide the proposed model to obtain more purposive and profitable data representations, leading to a better performance at last.
4) Comparing CoMSC, XLRS, and KLRS together, it can be seen that CoMSC outperforms the other two by large margins. In specific, it exceeds the second best by 10.62%, 6.38%, 3.77%, 13.62%, 1.59%, 3.80%, 20.22%, and 21.80% in ACC. Consistent improvements are also shown in purity, i.e., 10.62%, 2.12%, 1.89%, 7.98%, 1.50%, 1.19%, 3.93%, and 31.80%. In NMI, 12.21% on Dermatology, 9.44% on BBCSport, 3.00% on WebKB, 15.33% on Prokaryotic, 1.39% on Wiki, 11.68% on Caltech7, and 22.57% on HandWritten are observed, with only 0.40% decrease on Reuters. The observations illustrate that the proposed robust and purposive representations can boost the clustering performance to a large extent.

Overall, we can conclude that adopting kernel matrices on original data observations is helpful to the clustering task on most data sets, but the proposed robust and purposive representations, generated by co-training eigendecomposition and subspace clustering, can consistently improve the clustering performance by a large margin.

2) Superiority Over Recent MSC Algorithms: By jointly optimizing data representation and performing subspace clustering, the proposed algorithm outperforms MSC algorithms in recent literature. In order to validate this point, we conduct extensive experiments on 12 MSC algorithms and compare their performances in Table IV. We mark the best in bold and the second best with underline. Note that "–" indicates the corresponding values unavailable for long execution time. It can be seen that the proposed algorithm consistently and significantly outperforms the comparative ones.

1) It exceeds the baselines to a large extent over all metrics on seven data sets except Caltech7, i.e., 24.48%, 12.41%, 6.42%, 31.32%, 23.34%, 5.00%, and 14.00% in ACC; 43.28%, 22.93%, 31.57%, 41.42%, 21.78%, 2.06%, and 15.87% in NMI; and 11.39%, 12.05%, 22.26%, 27.76%, 23.16%, 1.85%, and 14.00% in purity. Meanwhile, some algorithms achieve worse performances than the two baselines, such as RMSC, DiMSC, and PMSC on Dermatology in ACC, which conversely supports the superiority of the proposed method.
2) Compared with the other MSC algorithms in recent literature, the proposed method outperforms them consistently and significantly. Except Caltech7, it exceeds the second best by 3.08%, 6.74%, 1.89%, 19.42%, 0.84%, 1.48%, and 2.00% in ACC; 7.12%, 10.95%, 3.60%, 20.12%, 0.43%, 3.76%, and 4.38% in NMI; and 3.08%, 2.83%, 2.64%, 13.43%, 0.41%, 1.64%, and 2.00% in purity. At the same time, the proposed algorithm improves the clustering performance by 4.41% in ACC and 1.53% in NMI with only 0.21% decrease in purity. Although relatively small improvements are observed on some data sets in one or more metrics over the second best results, it obviously achieves much better results than any one of the comparative methods alone.
3) We can observe that some algorithms fail on specific data sets. For example, LRSSC only achieves 3.25%, 6.92%, 7.61%, and 2.73% in NMI on BBCSport, WebKB, Prokaryotic, and Reuters, respectively. Poor performances deviated from averages largely can also be observed on PMSC, LT-MSC, LLRb, LLRc, and so on. However, the proposed method obtains promising results over eight data sets in all metrics, verifying its superiority.

Overall, the proposed CoMSC establishes its superiority over the recent MSC algorithms, as reported in Table IV.

3) Computational Efficiency: In most cases, MSC algorithms are of high computation load for the ADMM or ALM optimization strategy is adopted, while the proposed algorithm employs a simple alternate strategy that has a closed-form solution in each step, making it more efficient compared with most MSC algorithms in recent literature. The theoretical analysis of computation complexity is thoroughly analyzed in Section III-C. In addition, we validate its efficiency experimentally by comparing it with the others respect to the execution times on eight chosen data sets. The experiments are conducted on an Ubuntu 18.04 server with 4 Intel Xeon (Cascade Lake)

TABLE IV
PERFORMANCE COMPARISON OF MSC ALGORITHMS IN RECENT LITERATURE

	Algorithm	Dermatology	BBCSport	WebKB	Prokaryotic	Reuters	Wiki	Caltech7	HandWritten
ACC	LSRb [17]	72.73	48.23	49.43	53.36	31.08	53.59	**72.73**	72.10
	LSRc [17]	63.84	75.89	47.92	42.29	22.75	53.98	63.84	80.25
	RMSC [50]	63.13	70.21	45.66	52.63	53.50	57.50	46.88	78.40
	DiMSC [25]	50.00	76.24	50.94	49.00	42.33	52.51	43.69	34.55
	LT-MSC [26]	90.22	71.99	44.15	41.20	38.00	51.71	60.24	91.45
	MSSC [38]	94.13	65.60	53.96	50.64	49.25	45.36	47.56	91.90
	ECMSC [30]	87.71	80.85	48.68	43.19	-	-	-	-
	LMSC [49]	85.20	76.60	48.68	45.74	49.67	56.94	54.75	84.40
	LRSSC [32]	94.13	34.40	40.00	38.48	24.08	34.30	46.27	-
	CSMSC [51]	90.50	78.72	50.57	65.15	39.83	34.23	63.16	92.25
	FMR [52]	88.55	81.56	41.13	57.35	53.58	58.86	47.02	80.05
	PMSC [53]	70.95	37.94	46.04	58.26	20.25	15.35	54.34	46.40
	CoMSC	**97.21**	**88.30**	**55.85**	**84.57**	**54.42**	**58.98**	69.88	**94.25**
NMI	LSRb [17]	50.48	20.26	4.52	12.44	11.03	52.47	50.48	70.86
	LSRc [17]	46.24	55.27	3.73	12.06	6.52	52.63	26.24	74.18
	RMSC [50]	59.23	53.43	11.49	28.91	32.38	50.01	37.93	74.73
	DiMSC [25]	42.34	63.96	18.20	10.65	21.03	46.01	37.07	24.80
	LT-MSC [26]	80.28	58.18	8.73	11.65	18.69	44.50	54.72	84.82
	MSSC [38]	85.16	50.28	32.49	16.63	28.11	37.95	34.98	85.37
	ECMSC [30]	76.72	58.89	11.69	18.11	-	-	-	-
	LMSC [49]	86.57	67.12	16.59	16.49	29.93	50.39	49.49	78.58
	LRSSC [32]	86.64	3.25	6.92	7.61	2.73	17.39	27.79	-
	CSMSC [51]	83.71	67.25	11.52	32.89	18.43	23.58	54.77	85.67
	FMR [52]	85.24	64.87	15.28	33.74	31.43	50.93	38.76	70.72
	PMSC [53]	60.64	11.80	4.27	5.16	8.16	3.36	1.08	44.48
	CoMSC	**93.76**	**78.20**	**36.09**	**53.86**	**32.81**	**54.69**	**56.30**	**90.05**
Purity	LSRb [17]	85.82	50.00	50.57	57.71	31.42	60.89	85.82	76.15
	LSRc [17]	83.58	76.60	49.81	58.08	23.00	61.13	83.58	80.25
	RMSC [50]	71.51	79.08	53.59	68.42	54.17	60.05	81.34	78.95
	DiMSC [25]	58.94	84.04	60.80	58.44	44.08	56.11	80.46	36.25
	LT-MSC [26]	90.22	81.21	54.34	58.62	42.92	54.40	**88.81**	91.45
	MSSC [38]	94.13	74.82	70.19	64.43	51.42	49.20	81.95	91.90
	ECMSC [30]	87.71	83.33	56.60	59.35	-	-	-	-
	LMSC [49]	86.59	85.11	57.74	61.53	53.17	60.43	87.86	84.40
	LRSSC [32]	94.13	36.53	52.08	56.81	24.50	36.71	77.61	-
	CSMSC [51]	90.50	85.82	55.09	72.41	43.25	37.16	88.13	92.25
	FMR [52]	88.55	84.04	58.11	71.87	53.58	60.99	83.45	80.05
	PMSC [53]	72.35	43.97	48.68	59.17	25.17	18.28	54.55	53.55
	CoMSC	**97.21**	**88.65**	**72.83**	**85.84**	**54.58**	**62.98**	87.65	**94.25**

Platinum 8269CY. Table V reports the results and we mark the best in bold and the second-best with underline. It can be seen that RMSC is the most efficient, for it requires the shortest times on BBCSport, WebKB, Prokaryotic, Caltech7, and HandWritten and the second shortest times on Reuters and Wiki. Meanwhile, the second most efficient algorithm is the proposed CoMSC, with the shortest times on Reuters and Wiki and the second shorted times on BBCSport, WebKB and Prokaryotic. By the way, MSSC shows comparable results with CoMSC. The other algorithms are less efficient, for which they require more than 1000 s on one or more data sets. Therefore, we can conclude that CoMSC is more efficient than most MSC algorithms in recent literature, making it feasible in practical applications.

4) Robustness to Noise: Another merit of the proposed algorithm is its robustness to noise. We validate this by conducting experiments on Dermatology and BBCSport with noises that are generated by following the method in [49]. In specific, two types of noise are considered, including sample-specific $\mathbf{N}_s^{(v)}$ and global $\mathbf{N}_g^{(v)}$, where v refers to the vth view. For $\mathbf{N}_s^{(v)}$, we generate a random matrix with the same size of the vth data observation and keep some columns

TABLE V
EXECUTION TIME COMPARISON (IN SECONDS) OF MSC ALGORITHMS IN RECENT LITERATURE

Algorithm	Dermatology	BBCSport	WebKB	Prokaryotic	Reuters	Wiki	Caltech7	HandWritten
RMSC [50]	1.22	**0.95**	**0.98**	**4.09**	66.34	394.93	**65.34**	**203.09**
DiMSC [25]	3.42	3.71	12.27	18.85	917.85	2627.20	1407.90	5243.50
LT-MSC [26]	4.41	13.34	7.84	19.23	1672.50	2048.20	814.52	1500.90
MSSC [38]	**0.86**	25.30	25.81	35.48	972.56	472.20	182.01	292.83
ECMSC [30]	0.92	314.85	22.60	8.37	-	-	-	-
LMSC [49]	9.87	22.96	10.72	29.66	305.92	2464.70	254.43	593.73
LRSSC [32]	1.23	38.26	29.93	115.37	2535.20	7283.90	2846.60	-
CSMSC [51]	4.56	27.36	11.31	21.87	1765.00	1655.20	448.00	953.13
FMR [52]	24.96	24.07	17.34	72.34	1069.00	9745.10	902.88	3378.30
PMSC [53]	30.67	24.41	18.00	79.16	2223.90	5869.40	906.40	3815.50
CoMSC	8.45	3.19	3.78	7.49	**22.77**	**46.79**	485.49	362.59

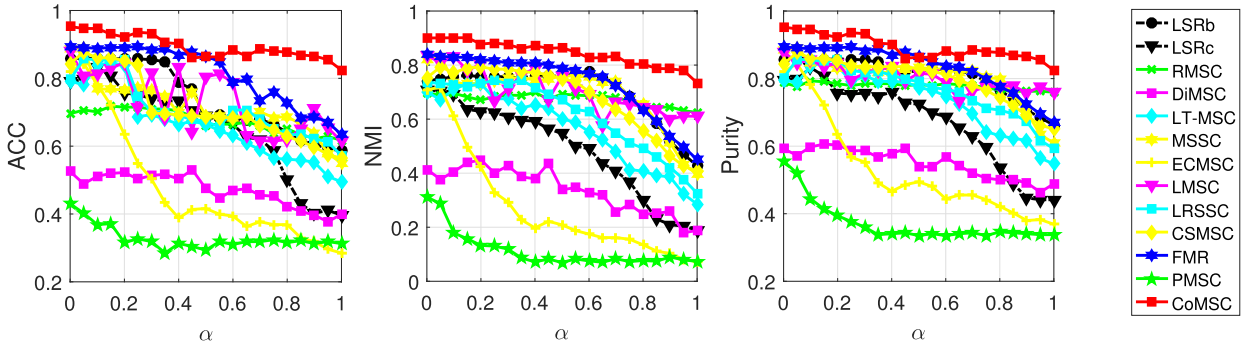

Fig. 3. Performance comparison on dermatology with different magnitudes of noise. Twelve recent MSC methods, including two baselines, i.e., LSRb and LSRc, are concerned.

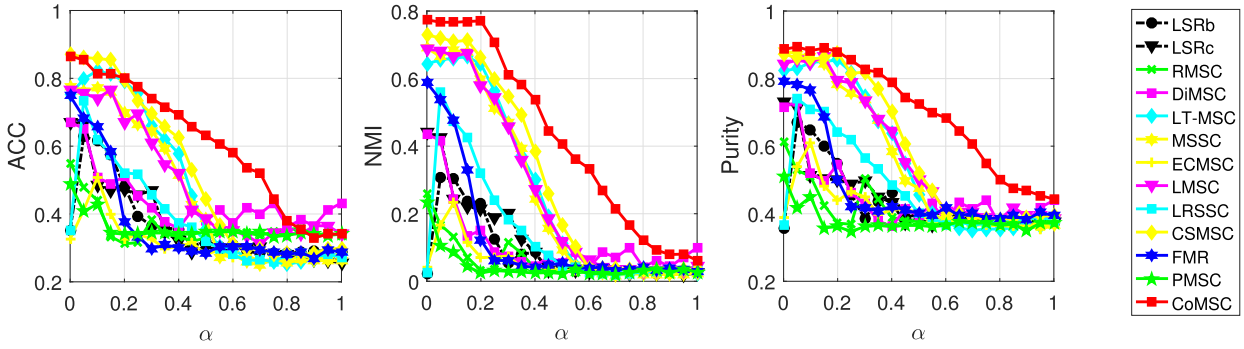

Fig. 4. Performance comparison on BBCSport with different magnitudes of noise. Twelve recent MSC methods, including two baselines, i.e., LSRb and LSRc, are concerned.

(20 columns in our experiments) while setting the others to zero. For $\mathbf{N}_g^{(v)}$, a coefficient α is multiplied on a randomly generated matrix to control noise magnitude. The overall noise can be obtained as $\mathbf{N}^{(v)} = \mathbf{N}_s^{(v)} + \alpha\mathbf{N}_g^{(v)}$. We compare CoMSC and recent MSC algorithms under different magnitudes of noise, and the results are reported in Figs. 3 and 4. It can be observed that all algorithms have different degrees of performance decrease when increasing noise, but CoMSC keeps the top-1 performances over all noise volumes on the two data sets. On BBCSport, CoMSC shows the smallest decrease and even keeps stable in ACC and purity when $\alpha \in [0.4, 0.8]$, while some algorithms, such as LT-MSC and FMR, drop quickly in this range. At the same time, DiMSC, ECMSC, and PMSC fail in this noise setting, presenting poor performances far away from average. We can see from Fig. 4 that BBCSport is more delicate to noise. All algorithms decrease to random guess at last. In ACC, LLRb and LLRc first drop to the bottom and is followed by RMSC, PMSC, FMR, DiMSC, LRSSC, LMSC, MSSC, LT-MSC, and CSMSC in order before $\alpha = 0.6$. On the contrary, CoMSC reaches

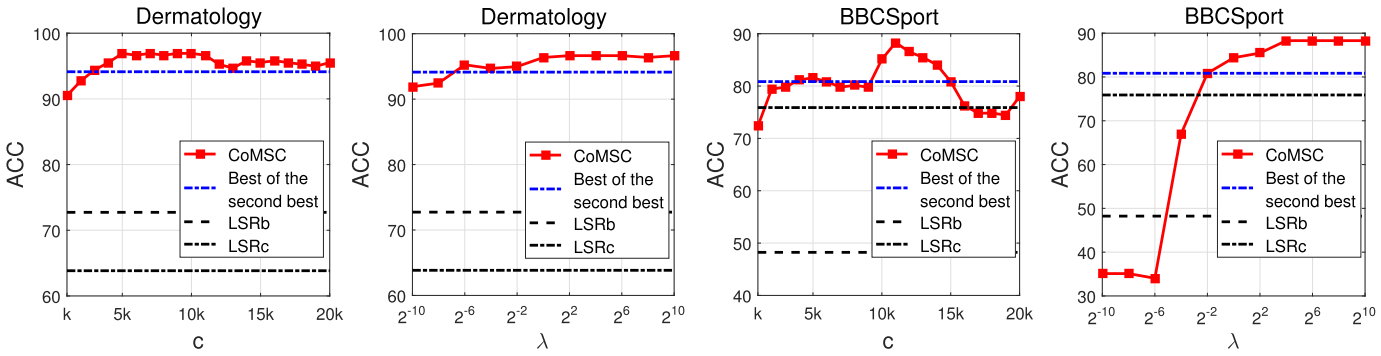

Fig. 5. Parameter sensitivity study. The left two plots show the results on dermatology, while the right two are tested on BBCSport. Best of the second best refers to the best result of the second best comparative methods when performing grid search. The other metrics, including NMI and purity, share the similar trend with ACC and are shown in the Appendix.

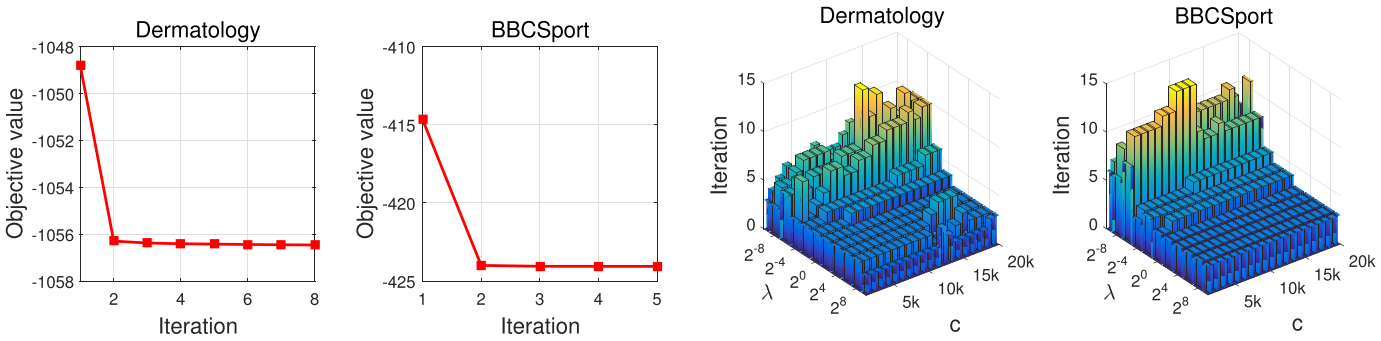

Fig. 6. Convergence validation on dermatology and BBCSport. The left two plots show the objective values along with iterations, whereas the right two present the iteration numbers required to stop.

the bottom at $\alpha = 0.8$. Similar observations can be obtained in NMI and purity, showing the robustness of the proposed method.

C. Parameter Study and Convergence

In order to investigate parameter stability of the proposed algorithm, we perform grid search on the size of robust data representation c when fixing λ to 2^{10}. Then, λ is tested with $c = 10k$. The results on dermatology and BBCSport are presented in Fig. 5. It can be observed that CoMSC largely exceeds the baselines, i.e., LLRb and LLRc. Meanwhile, we choose the best results of the other 12 algorithms over their own parameter ranges as best of the second best. The plots show that CoMSC stably outperforms them across a large range of both parameters, making it practical in real-world applications. We recommend to select c from $5k$ to $10k$ and λ from 2^0 to 2^{10}.

Furthermore, the left two plots in Fig. 6 show that the objective value monotonically decreases along with iterations and reaches the bottom on both dermatology and BBCSport, which proves the convergence of CoMSC experimentally. The right two plots in Fig. 6 present the number of iterations, which CoMSC requires to meet the stop criteria on Dermatology and BBCSport. The proposed algorithm quickly stops within 15 iterations.

V. Conclusion

Most MSC algorithms adopt the primary data observations or corresponding kernel matrices as input, but ignore their redundancies, leading to unsatisfactory performances. To address this issue, we propose an elegant method named MSC via CoMSC. It employs eigendecomposition technique to obtain robust data representations for the afterward subspace clustering. Meanwhile, the clustering result guides to generate more purposive data representations conversely. The proposed algorithm achieves state-of-the-art performance and is validated to be convergent, effective, efficient, and robust to noise. We will explore the relationship between eigenvalue distribution of kernel matrix and size of robust representations in the future work.

References

[1] X. Liu, E. Zhu, J. Liu, T. Hospedales, Y. Wang, and M. Wang, "SimpleMKKM: Simple multiple kernel K-means," 2020, *arXiv:2005.04975*. [Online]. Available: http://arxiv.org/abs/2005.04975

[2] X. Liu *et al.*, "Multiple kernel K-means with incomplete kernels," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 5, pp. 1191–1204, May 2019, doi: 10.1109/TPAMI.2019.2892416.

[3] X. Liu *et al.*, "Late fusion incomplete multi-view clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 10, pp. 2410–2423, Oct. 2019, doi: 10.1109/TPAMI.2018.2879108.

[4] X. Peng, H. Zhu, J. Feng, C. Shen, H. Zhang, and J. T. Zhou, "Deep clustering with sample-assignment invariance prior," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 11, pp. 4857–4868, Nov. 2020, doi: 10.1109/TNNLS.2019.2958324.

[5] L. Parsons, E. Haque, and H. Liu, "Subspace clustering for high dimensional data: A review," *ACM SIGKDD Explor. Newslett.*, vol. 6, no. 1, pp. 90–105, Jun. 2004, doi: 10.1145/1007730.1007731.

[6] E. Elhamifar and R. Vidal, "Sparse subspace clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Miami, FL, USA, Jun. 2009, pp. 2790–2797, doi: 10.1109/CVPR.2009.5206547.

[7] R. Vidal, "Subspace clustering," *IEEE Signal Process. Mag.*, vol. 28, no. 2, pp. 52–68, Mar. 2011, doi: 10.1109/MSP.2010.939739.

[8] Y. Wang, L. Wu, X. Lin, and J. Gao, "Multiview spectral clustering via structured low-rank matrix factorization," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 10, pp. 4833–4843, Oct. 2018, doi: 10.1109/TNNLS.2017.2777489.

[9] E. Elhamifar and R. Vidal, "Sparse subspace clustering: Algorithm, theory, and applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 11, pp. 2765–2781, Nov. 2013, doi: 10.1109/TPAMI.2013.57.

[10] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, and Y. Ma, "Robust recovery of subspace structures by low-rank representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 171–184, Jan. 2013, doi: 10.1109/TPAMI.2012.88.

[11] H. Hu, Z. Lin, J. Feng, and J. Zhou, "Smooth representation clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Columbus, OH, USA, Jun. 2014, pp. 3834–3841, doi: 10.1109/CVPR.2014.484.

[12] J. Feng, Z. Lin, H. Xu, and S. Yan, "Robust subspace segmentation with block-diagonal prior," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Columbus, OH, USA, Jun. 2014, pp. 3818–3825, doi: 10.1109/CVPR.2014.482.

[13] M. Yin, Y. Guo, J. Gao, Z. He, and S. Xie, "Kernel sparse subspace clustering on symmetric positive definite manifolds," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 5157–5164, doi: 10.1109/CVPR.2016.557.

[14] C. Tang *et al.*, "Learning a joint affinity graph for multiview subspace clustering," *IEEE Trans. Multimedia*, vol. 21, no. 7, pp. 1724–1736, Jul. 2019, doi: 10.1109/TMM.2018.2889560.

[15] S. Zhou *et al.*, "Subspace segmentation-based robust multiple kernel clustering," *Inf. Fusion*, vol. 53, pp. 145–154, Jan. 2020, doi: 10.1016/j.inffus.2019.06.017.

[16] G. Liu, Z. Lin, and Y. Yu, "Robust subspace segmentation by low-rank representation," in *Proc. 27th Int. Conf. Mach. Learn. (ICML)*, Haifa, Israel, 2010, pp. 663–670. [Online]. Available: https://icml.cc/Conferences/2010/papers/521.pdf

[17] C.-Y. Lu, H. Min, Z.-Q. Zhao, L. Zhu, D.-S. Huang, and S. Yan, "Robust and efficient subspace segmentation via least squares regression," in *Computer Vision—ECCV 2012*, A. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato, and C. Schmid, Eds. Berlin, Germany: Springer, 2012, pp. 347–360.

[18] D. Luo, F. Nie, C. Ding, and H. Huang, "Multi-subspace representation and discovery," in *Machine Learning and Knowledge Discovery in Databases*, D. Gunopulos, T. Hofmann, D. Malerba, and M. Vazirgiannis, Eds. Berlin, Germany: Springer, 2011, pp. 405–420.

[19] C. Lu, J. Tang, M. Lin, L. Lin, S. Yan, and Z. Lin, "Correntropy induced l2 graph for robust subspace clustering," in *Proc. IEEE Int. Conf. Comput. Vis.*, Sydney, NSW, Australia, Dec. 2013, pp. 1801–1808, doi: 10.1109/ICCV.2013.226.

[20] C.-D. Wang, J.-H. Lai, and P. S. Yu, "Multi-view clustering based on belief propagation," *IEEE Trans. Knowl. Data Eng.*, vol. 28, no. 4, pp. 1007–1021, Apr. 2016, doi: 10.1109/TKDE.2015.2503743.

[21] K. Zhan, C. Niu, C. Chen, F. Nie, C. Zhang, and Y. Yang, "Graph structure fusion for multiview clustering," *IEEE Trans. Knowl. Data Eng.*, vol. 31, no. 10, pp. 1984–1993, Oct. 2019, doi: 10.1109/TKDE.2018.2872061.

[22] C. Tang *et al.*, "Feature selective projection with low-rank embedding and dual Laplacian regularization," *IEEE Trans. Knowl. Data Eng.*, vol. 32, no. 9, pp. 1747–1760, Sep. 2020.

[23] X. Peng, J. Feng, J. T. Zhou, Y. Lei, and S. Yan, "Deep subspace clustering," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 12, pp. 5509–5521, Dec. 2020, doi: 10.1109/TNNLS.2020.2968848.

[24] J. Liu, Y. Zeng, J. Shi, Y. Yang, R. Wang, and L. He, "MalDetect: A structure of encrypted malware traffic detection," *Comput., Mater. Continua*, vol. 60, no. 2, pp. 721–739, 2019, doi: 10.32604/cmc.2019.05610.

[25] X. Cao, C. Zhang, H. Fu, S. Liu, and H. Zhang, "Diversity-induced multi-view subspace clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, Jun. 2015, pp. 586–594, doi: 10.1109/CVPR.2015.7298657.

[26] C. Zhang, H. Fu, S. Liu, G. Liu, and X. Cao, "Low-rank tensor constrained multiview subspace clustering," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Santiago, Chile, Dec. 2015, pp. 1582–1590, doi: 10.1109/ICCV.2015.185.

[27] H. Gao, F. Nie, X. Li, and H. Huang, "Multi-view subspace clustering," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Santiago, Chile, Dec. 2015, pp. 4238–4246, doi: 10.1109/ICCV.2015.482.

[28] Q. Yin, S. Wu, R. He, and L. Wang, "Multi-view clustering via pairwise sparse subspace representation," *Neurocomputing*, vol. 156, pp. 12–21, May 2015, doi: 10.1016/j.neucom.2015.01.017.

[29] L. Wang, D. Li, T. He, and Z. Xue, "Manifold regularized multi-view subspace clustering for image representation," in *Proc. 23rd Int. Conf. Pattern Recognit. (ICPR)*, Cancún, Mexico, Dec. 2016, pp. 283–288, doi: 10.1109/ICPR.2016.7899647.

[30] X. Wang, X. Guo, Z. Lei, C. Zhang, and S. Z. Li, "Exclusivity-consistency regularized multi-view subspace clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 1–9, doi: 10.1109/CVPR.2017.8.

[31] Y. Fan, J. Liang, R. He, B. Hu, and S. Lyu, "Robust localized multi-view subspace clustering," May 2017, *arXiv:1705.07777*. [Online]. Available: http://arxiv.org/abs/1705.07777

[32] M. Brbić and I. Kopriva, "Multi-view low-rank sparse subspace clustering," *Pattern Recognit.*, vol. 73, pp. 247–258, Jan. 2018, doi: 10.1016/j.patcog.2017.08.024.

[33] H. Yu, T. Zhang, Y. Lian, and Y. Cai, "Co-regularized multi-view subspace clustering," in *Proc. 10th Asian Conf. Mach. Learn. (ACML)*, Beijing, China, Nov. 2018, pp. 17–32. [Online]. Available: http://proceedings.mlr.press/v95/yu18a.html

[34] G.-Y. Zhang, Y.-R. Zhou, X.-Y. He, C.-D. Wang, and D. Huang, "One-step kernel multi-view subspace clustering," *Knowl.-Based Syst.*, vol. 189, Feb. 2020, Art. no. 105126, doi: 10.1016/j.knosys.2019.105126.

[35] K. Zhan, F. Nie, J. Wang, and Y. Yang, "Multiview consensus graph clustering," *IEEE Trans. Image Process.*, vol. 28, no. 3, pp. 1261–1270, Mar. 2019, doi: 10.1109/TIP.2018.2877335.

[36] Y. Wang, X. Lin, L. Wu, W. Zhang, Q. Zhang, and X. Huang, "Robust subspace clustering for multi-view data by exploiting correlation consensus," *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 3939–3949, Nov. 2015, doi: 10.1109/TIP.2015.2457339.

[37] J. Liu, X. Liu, Y. Yang, S. Wang, and S. Zhou, "Hierarchical multiple kernel clustering," in *Proc. 35th AAAI Conf. Artif. Intell. (AAAI)*, 2021.

[38] M. Abavisani and V. M. Patel, "Multimodal sparse and low-rank subspace clustering," *Inf. Fusion*, vol. 39, pp. 168–177, Jan. 2018, doi: 10.1016/j.inffus.2017.05.002.

[39] D. Xie, Q. Gao, Q. Wang, X. Zhang, and X. Gao, "Adaptive latent similarity learning for multi-view clustering," *Neural Netw.*, vol. 121, pp. 409–418, Jan. 2020, doi: 10.1016/j.neunet.2019.09.013.

[40] S. Zhou *et al.*, "Multiple kernel clustering with neighbor-kernel subspace segmentation," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 4, pp. 1351–1362, Apr. 2020, doi: 10.1109/TNNLS.2019.2919900.

[41] V. M. Patel and R. Vidal, "Kernel sparse subspace clustering," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Paris, France, Oct. 2014, pp. 2849–2853, doi: 10.1109/ICIP.2014.7025576.

[42] S. Hechmi, A. Gallas, and E. Zagrouba, "Multi-kernel sparse subspace clustering on the Riemannian manifold of symmetric positive definite matrices," *Pattern Recognit. Lett.*, vol. 125, pp. 21–27, Jul. 2019, doi: 10.1016/j.patrec.2019.03.019.

[43] S. Zhou *et al.*, "Multi-view spectral clustering with optimal neighborhood Laplacian matrix," in *Proc. 34th AAAI Conf. Artif. Intell. (AAAI)*. New York, NY, USA: AAAI Press, 2020, pp. 6965–6972. [Online]. Available: https://aaai.org/ojs/index.php/AAAI/article/view/6180

[44] B. Schölkopf, A. Smola, and K.-R. Müller, "Nonlinear component analysis as a kernel eigenvalue problem," *Neural Comput.*, vol. 10, no. 5, pp. 1299–1319, Jul. 1998.

[45] M. Kloft, U. Brefeld, S. Sonnenburg, P. Laskov, K. Müller, and A. Zien, "Efficient and accurate lp-norm multiple kernel learning," in *Proc. Adv. Neural Inf. Process. Syst.* Vancouver, BC, Canada: Curran Associates, 2009, pp. 997–1005. [Online]. Available: http://papers.nips.cc/paper/3675-efficient-and-accurate-lp-norm-multiple-kernel-learning

[46] G. H. Golub and C. F. V. Loan, *Matrix Computations*, vol. 3. Baltimore, MD, USA: Johns Hopkins Univ. Press, 1983.

[47] S. Wang *et al.*, "Multi-view clustering via late fusion alignment maximization," in *Proc. 28th Int. Joint Conf. Artif. Intell.*, Macao, China, Aug. 2019, pp. 3778–3784, doi: 10.24963/ijcai.2019/524.

[48] J. Liu *et al.*, "Optimal neighborhood multiple kernel clustering with adaptive local kernels," *IEEE Trans. Knowl. Data Eng.*, early access, Aug. 4, 2021, doi: 10.1109/TKDE.2020.3014104.

[49] C. Zhang, Q. Hu, H. Fu, P. Zhu, and X. Cao, "Latent multi-view subspace clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*. Honolulu, HI, USA, Jul. 2017, pp. 4333–4341, doi: 10.1109/CVPR.2017.461.

[50] R. Xia, Y. Pan, L. Du, and J. Yin, "Robust multi-view spectral clustering via low-rank and sparse decomposition," in *Proc. 28th AAAI Conf. Artif. Intell.*, Québec City, QC, Canada, 2014, pp. 2149–2155. [Online]. Available: http://www.aaai.org/ocs/index.php/AAAI/AAAI14/paper/view/8135

[51] S. Luo, C. Zhang, W. Zhang, and X. Cao, "Consistent and specific multi-view subspace clustering," in *Proc. 32nd AAAI Conf. Artif. Intell.*, New Orleans, LA, USA, 2018, pp. 3730–3737. [Online]. Available: https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/16212

[52] R. Li, C. Zhang, Q. Hu, P. Zhu, and Z. Wang, "Flexible multi-view representation learning for subspace clustering," in *Proc. 28th Int. Joint Conf. Artif. Intell.*, Macao, China, Aug. 2019, pp. 2916–2922, doi: 10.24963/ijcai.2019/404.

[53] Z. Kang *et al.*, "Partition level multiview subspace clustering," *Neural Netw.*, vol. 122, pp. 279–288, Feb. 2020, doi: 10.1016/j.neunet.2019.10.010.

Jiyuan Liu is currently pursuing the Ph.D. degree with the National University of Defense Technology (NUDT), Changsha, China.

He has published papers in journals and conferences, such as IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, AAAI Conference on Artificial Intelligence (AAAI), and International Joint Conference on Artificial Intelligence (IJCAI). His current research interests include multiview clustering, deep clustering, and anomaly detection.

Mr. Liu serves as a Program Committee Member and a Reviewer for IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, the 35th AAAI Conference on Artificial Intelligence 2021 (AAAI-21), the 30th International Joint Conference on Artificial Intelligence 2021 (IJCAI-21), and so on.

Xinwang Liu (Senior Member, IEEE) received the Ph.D. degree from the National University of Defense Technology (NUDT), Changsha, China, in 2013.

He is currently a Professor at the School of Computer, NUDT. He has published more than 60 peer-reviewed papers, including those in highly regarded journals and conferences such as IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, IEEE TRANSACTIONS ON IMAGE PROCESSING, IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, IEEE TRANSACTIONS ON MULTIMEDIA, IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, NeurIPS, Computer Vision and Pattern Recognition (CVPR), International Conference on Computer Vision (ICCV), AAAI Conference on Artificial Intelligence (AAAI), and International Joint Conference on Artificial Intelligence (IJCAI). His current research interests include kernel learning and unsupervised feature learning. More information can be found at https://xinwangliu.github.io/.

Yuexiang Yang received the B.S. degree in mathematics from Xiangtan University, Xiangtan, China, in 1986, and the M.S. degree in computer application and the Ph.D. degree in computer science and technology from the National University of Defense Technology, Changsha, China, in 1989 and 2008, respectively.

He is currently the Executive Director of the Information Branch of China Higher Education Association. He has coauthored more than 100 papers in international journals and conference or workshop proceedings. His research interests include information retrieval, network security, and data analysis.

Dr. Yang has been serving as a reviewer and a program committee member of various conferences and journals.

Xifeng Guo received the M.S. and Ph.D. degrees in computer science from the National University of Defense Technology, Changsha, China, in 2016 and 2020, respectively.

He has published papers in highly regarded journals and conferences, such as IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING (TKDE), IEEE MultiMedia, *Pattern Recognition*, AAAI Conference on Artificial Intelligence (AAAI), and International Joint Conference on Artificial Intelligence (IJCAI). His research interests include deep clustering, unsupervised learning, transfer learning, and computer vision.

Dr. Guo served on the Technical Program Committee of IJCAI 2020 and 2021 and AAAI 2020 and 2021. He serves as a Reviewer for IEEE TKDE, IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, *Pattern Recognition*, and *Neural Networks*.

Marius Kloft (Senior Member, IEEE) received the Ph.D. degree from the Technical University of Berlin, Berlin, Germany, and University of California at Berkeley, Berkeley, CA, USA, in 2011.

He was an Assistant Professor at the Humboldt University of Berlin, Berlin, from 2014 to 2017, and a joint Post-Doctoral Fellow at the Courant Institute of Mathematical Sciences, New York University, New York, NY, USA, and the Memorial Sloan-Kettering Cancer Center, New York. He has been a Professor of machine learning with the Department of Computer Science, Technische Universität Kaiserslautern, Kaiserslautern, Germany, since 2017. He is interested in theory and algorithms of statistical machine learning and its applications. His research covers a broad range of topics and applications, where he tries to unify theoretically proven approaches (e.g., based on learning theory) with recent advances (e.g., in deep learning and reinforcement learning). He has been working on, e.g., multimodal learning, anomaly detection, extreme classification, and adversarial learning for computer security.

Dr. Kloft received the Google Most Influential Papers Award in 2014. He has served as a senior AC for AISTATS 2020 and AAAI Conference on Artificial Intelligence (AAAI) 2020 and is an Associate Editor of IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS.

Liangzhong He received the B.S. degree in information security from the University of Electronic Science and Technology of China, Chengdu, China, in 2011.

He is currently the Research Manager of the Security Department, China Mobile (Su Zhou) Software Technology Company Ltd., Suzhou, China. His focus is the development of cloud workload protection platforms, cloud security posture management products, cloud security web application firewall, and so on.