Adaptive Transition Probability Matrix Learning for Multiview Spectral Clustering

Yongyong Chen, Xiaolin Xiao, Zhongyun Hua, *Member, IEEE*, and Yicong Zhou, *Senior Member, IEEE*

***Abstract*— Multiview clustering as an important unsupervised method has been gathering a great deal of attention. However, most multiview clustering methods exploit the *self-representation property* to capture the relationship among data, resulting in high computation cost in calculating the self-representation coefficients. In addition, they usually employ different regularizers to learn the representation tensor or matrix from which a transition probability matrix is constructed in a separate step, such as the one proposed by Wu *et al.*. Thus, an optimal transition probability matrix cannot be guaranteed. To solve these issues, we propose a unified model for multiview spectral clustering by directly learning an adaptive transition probability matrix (MCA^{2}M), rather than an individual representation matrix of each view. Different from the one proposed by Wu *et al.*, MCA^{2}M utilizes the one-step strategy to directly learn the transition probability matrix under the robust principal component analysis framework. Unlike existing methods using the absolute symmetrization operation to guarantee the nonnegativity and symmetry of the affinity matrix, the transition probability matrix learned from MCA^{2}M is nonnegative and symmetric without any postprocessing. An alternating optimization algorithm is designed based on the efficient alternating direction method of multipliers. Extensive experiments on several real-world databases demonstrate that the proposed method outperforms the state-of-the-art methods.**

***Index Terms*— Adaptive learning, low-rank representation (LRR), Markov chain, multiview clustering, spectral clustering.**

Manuscript received 8 February 2020; revised 18 July 2020, 29 October 2020, and 25 January 2021; accepted 11 February 2021. Date of publication 2 March 2021; date of current version 2 September 2022. This work was supported in part by the Science and Technology Development Fund, Macau SAR (File no. 189/2017/A3), and by University of Macau (File no. MYRG2018-00136-FST), and in part by the National Natural Science Foundation of China under Grant 62071142. *(Corresponding author: Yicong Zhou.)*

Yongyong Chen is with the School of Computer Science and Technology, Harbin Institute of Technology, Shenzhen 518055, China, also with the Bio-Computing Research Center, Harbin Institute of Technology, Shenzhen 518055, China, and also with the Shenzhen Key Laboratory of Visual Object Detection and Recognition, Harbin Institute of Technology, Shenzhen 518055, China (e-mail: yongyongchen.cn@gmail.com).

Xiaolin Xiao is with the School of Computer Science and Engineering, South China University of Technology, Guangzhou 510006, China (e-mail: shellyxiaolin@gmail.com).

Zhongyun Hua is with the School of Computer Science and Technology, Harbin Institute of Technology, Shenzhen 518055, China (e-mail: huazhongyun@hit.edu.cn).

Yicong Zhou is with the Department of Computer and Information Science, University of Macau, Macau 999078, China (e-mail: yicongzhou@um.edu.mo).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TNNLS.2021.3059874.

Digital Object Identifier 10.1109/TNNLS.2021.3059874

I. Introduction

With the advance of data collection and storage, modern data are usually described by different features or modalities, i.e., multiviews. For example, face images usually are captured with different features, such as color, textures, and edges, for face recognition [2], [3]; in surveillance systems, the target person is monitored from multiple cameras for person identification [4], [5]. These multiview features can provide complementary and compatible information to each other, and thus, are beneficial for data analysis. Considering this fact, there is a rapidly growing interest in partitioning unlabeled data points with multiview features into several distinct groups with the assistance of multiview features, generating the class of multiview clustering [1], [6]–[8].

The goal of multiview clustering is to uncover the underlying data structures by using multiview features to overcome the deficiency from a single view [7], [9], [10]. A surge of approaches has been proposed for multiview clustering. The works in [11]–[14] may construct different graphs for multiple views or learn a fusion graph shared by all views and then apply graph cut algorithm to generate the final segmentation results. Specifically, Nie *et al.* [11] proposed to learn a Laplacian rank constrained graph. Tao *et al.* [12] learned a common similarity matrix. In view of the above-mentioned works, Zhan *et al.* [13], [14] proposed the robust and consensus graph learning methods. However, these multiview graph clustering methods may ignore some global priors, such as low rankness [7], [9], [15] since they usually assign nearest neighbors for each data point based on pairwise distances. From the perspective of the optimization models they employ, subspace clustering-based methods are usually based on the low-rank representation (LRR) model [16], while spectral clustering-based ones are under the robust principal component analysis (RPCA) framework [17]. Multiview subspace clustering methods usually extended traditional clustering methods, such as the sparse subspace clustering (SSC) [18] and LRR [16] to handle multiview features from the matrix and tensor aspects. For instance, the study in [19] performed the subspace clustering on each view and simultaneously learn a common indicator matrix for the representation consistency. Brbić and Kopriva [20] extended the low-rank SSC into multiview setting. Zhang *et al.* [21] developed a latent representation learning model to overcome potential noise and outliers from data contamination. In [22], the LRR model was used to learn a common representation, and a rank constraint [23] was used to promote the construction

of the affinity matrix. Unlike the above matrix-optimization methods, the works in [2], [7], [9], and [24] addressed the multiview clustering task from the tensor-optimization aspect and have achieved superior performance. The reason is that they stored all representation matrices into a third-order tensor to explore the high-order correlations underlying multiview data. However, they usually suffer from high computational complexity. This is mainly because at each iteration they are inevitable to perform the matrix inversion with $\mathcal{O}(n^3)$ complexity, where n is the number of data points.

To alleviate the high computational cost of subspace clustering methods, a large number of efficient methods based on SSC and LRR have been proposed. For example, Peng *et al*. [25] and Matsushima and Brbić [26] proposed a scalable SSC and a selective sampling-based scalable SSC, respectively. By factorizing the representation matrix into two small-scale factor matrices, the work in [27] developed an online low-rank subspace clustering method to yield the linear complexity. To handle the multiview clustering task, Kang *et al*. [28] proposed a linear large-scale multiview subspace clustering. Another line to reduce the computation cost is that the works in [1] and [29]–[31] proposed multiview spectral clustering methods under the RPCA framework. The main idea of them is that they separated the transition probability matrix or tensor into a low-rank part and a sparse term. The main differences are that the former two learned a shared low-rank matrix, while the last two pursued a low-rank tensor. Generally speaking, in these methods, the affinity matrix is constructed by two separated steps [32], [33], that is, the representation learning and affinity construction are deliberately separated for easy handling. However, they may suffer from two major limitations: 1) most existing multiview clustering methods first learn the representations $\{Z^{(v)}\}_{v=1}^{M}$ by different regularizers, such as the nuclear norm [7], [9], sparseness [29], and block-diagonal prior [34], and then constructs the transition probability matrix S by averaging all representations, i.e., $S = \sum_{v=1}^{M}(|Z^{(v)}| + |Z^{(v)^T}|)/2$. Since this two-step paradigm may lead to suboptimal results in both stages, the optimal transition probability matrix cannot be ensured for the subsequent clustering and 2) there are two inherent characteristics, i.e., the nonnegativity and symmetry of the transition probability matrix, which are fully ignored in the first step. Thus, it cannot avoid negative coefficients [35]. To the best of our knowledge, there are no exiting works to address the above-mentioned two limitations simultaneously.

In this article, we propose a novel multiview spectral clustering via learning an adaptive transition probability matrix (MCA^{2}M) based on two concerns: 1) high connection between the spectral clustering and Markov chain and 2) the one-step strategy. The first motivation can greatly reduce the computational cost since matrix inverse operation with the heavy cost is avoided at each iteration. While the second one aims to directly learn an adaptive transition probability matrix to better uncover the true similarity between data points instead of the two-step paradigm. This is quite different from those methods in [1], [7], [9], and [29] all of which followed the two-step paradigm to construct the final affinity matrix. In addition, MCA^{2}M learns the transition probability matrix learned with the nonnegativity and symmetry throughout the learning process. This is also different from [1], [7], [9], and [29] which use postprocessing to ensure nonnegativity and symmetry. In addition, our MCA^{2}M consumes low computation costs by inheriting the benefits of RPCA framework. The flowchart of the proposed MCA^{2}M is shown in Fig. 1.

The main contributions of this article are summarized as follows.

1) Under the RPCA framework, we proposed a unified model for the multiview spectral clustering method by directly learning an adaptive transition probability matrix (MCA^{2}M). The nonnegativity and symmetry of the transition probability matrix are explicitly preserved without performing any postprocessing as previous low-rank tensor approximation-based clustering methods have to do.
2) Our MCA^{2}M jointly optimizes the transition probability tensor and the transition probability matrix to better discover the true similarity between data points. The low-rank constraint was imposed on the transition probability tensor, which provides feedback to adjust the learned transition probability matrix.
3) An effective algorithm is designed to solve the MCA^{2}M model based on the alternating direction method of multipliers (ADMMs). Extensive experiments are conducted on eight real-world databases with different applications. These results have demonstrated the superiority of the proposed MCA^{2}M over many state-of-the-art clustering methods.

The rest of this article is structured as follows. Section II briefly reviews the related works for multiview clustering. Some preliminaries are summarized in Section III. The MCA^{2}M method is introduced and optimized based on ADMMs in Section IV. We evaluate the performance of the proposed method in Section V and conclude the whole article in Section VI.

II. Related Work

The related works can be roughly categorized into two groups: LRR-based multiview subspace clustering methods and RPCA-based multiview spectral clustering ones. Several representative examples of these two groups and the proposed MCA^{2}M are summarized in Table I.

A. LRR-Based Multiview Subspace Clustering Methods

The LRR-based clustering methods are based on the self-representation property, that is, each data point can be linearly represented by the other data points in the same subspace. SSC [18] and LRR [16] solved the single-view clustering (SVC) task by seeking a sparse representation and a low-rank one over the data itself, respectively. Instead of a two-step paradigm to construct the final similarity matrix, Yin *et al*. [33] proposed a unified model to learn the affinity matrix and representation simultaneously. Considering the high spatial structures of 2-D images, the study in [38] proposed a tensor LRR method for subspace clustering. To handle the multiview

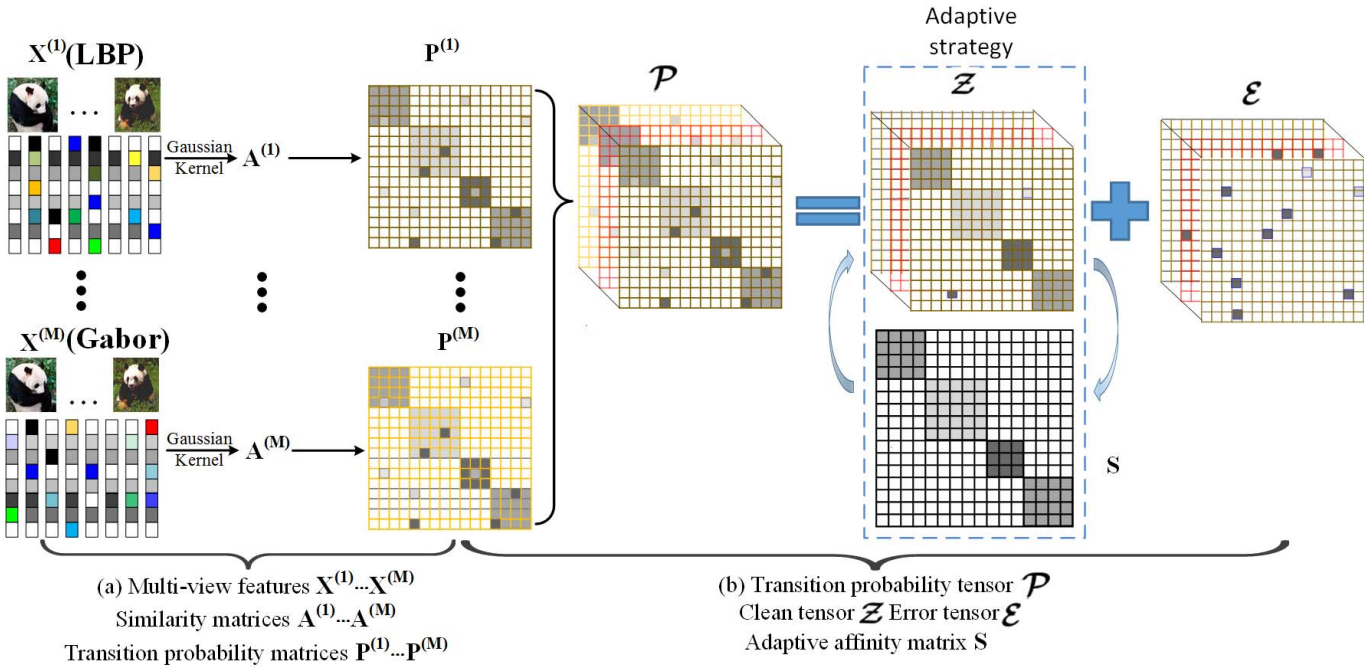

Fig. 1. Flowchart of the proposed MCA^{2}M method. Given multiview features $\{X^{(i)}\}$, MCA^{2}M first computes the similarity matrices $\{A^{(i)}\}$ and their transition probability matrices $\{P^{(i)}\}$, as shown in (a). By storing the ith $P^{(i)}$ as the ith frontal slice of a third-order *transition probability tensor* $\mathcal{P}$, as shown in (b), it is decomposed into a "clean" $\mathcal{Z}$ and an error tensor $\mathcal{E}$ to explore the high-order correlations among multiple views and overcome the noise interference. To well explore the correlations between $\mathcal{Z}$ and transition probability matrix S, MCA^{2}M learns an adaptive S for spectral clustering.

TABLE I
REPRESENTATIVE EXAMPLES OF CLUSTERING METHODS AND OUR PROPOSED MCA^{2}M

Method	Model	Constrain	Optimization	Paradigm	Task	Affinity
SSC [18]	$\|Z\|_1+\lambda\|E\|_F^2$	$X=XZ+E$	LRR	2-D matrix	Single-view	Two-step
LRR [16]	$\|Z\|_*+\lambda\|E\|_{2,1}$	$X=XZ+E$	LRR	2-D matrix	Single-view	Two-step
RGC [36]	$\|Z\|_*+\alpha\|E\|_1+\beta tr(ZLZ^T)+\gamma\|S\|_F^2$	$X=Z+E$	RPCA	2-D matrix	Single-view	One-step
RMSC [29]	$\|Z\|_*+\lambda\sum_{v=1}^{M}\|E^{(v)}\|_1$	$P^{(v)}=P+E^{(v)}, P\geq 0, P\mathbf{1}=\mathbf{1}$	RPCA	2-D matrix	Multi-view	Two-step
LT-MSC [9]	$\|\mathcal{Z}\|_*+\lambda\|\mathcal{E}\|_{2,1}$	$X^{(v)}=X^{(v)}Z^{(v)}+E^{(v)}$	LRR	3-D tensor	Multi-view	Two-step
LMSC [37]	$\|Z\|_*+\lambda\|E\|_{2,1}$	$X=PH+E_h, H=HZ+E_r$	LRR	2-D matrix	Multi-view	Two-step
t-SVD-MSC [7]	$\|\mathcal{Z}\|_{\circledast}+\lambda\|\mathcal{E}\|_{2,1}$	$X^{(v)}=X^{(v)}Z^{(v)}+E^{(v)}$	LRR	3-D tensor	Multi-view	Two-step
ETLMSC [1]	$\|\mathcal{Z}\|_{\circledast}+\lambda\|\mathcal{E}\|_{2,1}$	$\mathcal{P}=\mathcal{Z}+\mathcal{E}$	RPCA	3-D tensor	Multi-view	Two-step
Our MCA^{2}M	$\|\mathcal{Z}\|_{\circledast}+\alpha\|E\|_{2,1}+\beta\sum_{v=1}^{M}\|\frac{\|\mathcal{Z}^{(v)}\|+\|\mathcal{Z}^{(v)^T}\|}{2}-S\|_F^2$	$\mathcal{P}=\mathcal{Z}+\mathcal{E}, S_i\geq 0, S_i^T\mathbf{1}=\mathbf{1}$	RPCA	3-D tensor	Multi-view	One-step

features, Zhang *et al*. [9] performed LRR on each view jointly and learned a representation tensor. Xie *et al*. [7], [39] used the tensor nuclear norm (TNN), rather than the sum nuclear norm in [9] to explore the low-rankness of the representation tensor. Yang *et al.* [40] took the view specification and consensus structure into account for multiview clustering, where the latter was regularized by block diagonal regularization [34]. Unlike the above-mentioned methods which learn the similarity matrix from the original multiview features, Zhang *et al*. [37] proposed to learn the latent representation of multiview features which are used to construct a common coefficient matrix by LRR. Other LRR-based methods include [41] and [42].

B. RPCA-Based Multiview Spectral Clustering Methods

Different from the above-mentioned LRR-based methods, the second class exploited the RPCA model (also known as *low-rank and sparse decomposition*) to separate the representation into a low-rank part and a sparse component [43]. For example, Kang *et al*. [36] proposed a robust graph learning method for SVC. It is intractable to extend [36] to handle the multiview features since the dimensions of multiple features are usually unequal. To solve this problem and relieve the high complexity of LRR-based clustering methods, the studies in [29] and [1] presented two robust multiview spectral clustering methods, as shown in the fourth and eighth lines of Table II, respectively. The study in [30] proposed an error-robust multiview spectral clustering method to address different types of errors in multiview features.

In summary, the proposed MCA^{2}M differs significantly from the aforementioned works. First, one common bottleneck of the above-mentioned LRR-based methods [7], [9], [37] is the high computational cost since they cannot avoid the matrix inversion operation with $\mathcal{O}(n^3)$ complexity, where n denotes

TABLE II
BASIC NOTATIONS AND THEIR DESCRIPTIONS

Notation	Meaning
$\mathcal{X}$, X, x	tensor, matrix, vector
$\mathcal{X}^{(k)}$	the k-th frontal slice of tensor $\mathcal{X}$
$\hat{\mathcal{X}} = \mathbf{fft}(\mathcal{X}, [], 3)$	fast Fourier transformation along tube fiber
$X^{(v)} \in \mathbb{R}^{d_v \times n}$	feature matrix of the v-th view
$\mathcal{Z} \in \mathbb{R}^{n \times n \times M}$	transition probability tensor
$S \in \mathbb{R}^{n \times n}$	affinity matrix
E	error matrix
$\|\cdot\|_{2,1}$, $\|\cdot\|_F$	$l_{2,1}$-norm, Frobenius norm
$\|\cdot\|_{\circledast}$, $\|\cdot\|_\infty$	t-SVD-nuclear norm, infinity norm

the number of all samples. They are not suitable for large-scale applications. Second, most of them followed the two-step paradigm to construct the affinity matrix, which may not be the optimal one for data clustering and lead to suboptimal results. In addition, two essential characteristics i.e., nonnegativity and symmetry are usually ignored in the construction of the affinity matrix. The proposed MCA^{2}M has three advantages: 1) the proposed MCA^{2}M inherits the advantage of the RPCA framework to avoid the matrix inversion; 2) unlike the two-step paradigm to construct the affinity matrix, MCA^{2}M learns the transition probability matrix directly; and 3) the proposed MCA^{2}M preserves the nonnegativity and symmetry of the transition probability matrix in the learning processing, rather than resorting to postprocessing.

III. PRELIMINARY

In this section, we aim to briefly review the background of the tensor singular value decomposition (t-SVD)-based TNN (see Definition 2) and the Markov chain-based spectral clustering.

A. T-SVD-Based Tensor Nuclear Norm

For easy representation, some basic notations are shown in Table II. In the following, we first introduce several operators that are used to define the t-SVD-TNN in (1). For a tensor $\mathcal{X} \in \mathbb{R}^{M \times N \times K}$, its block diagonal matrix $\mathbf{bdiag}(\mathcal{X})$ and block circular matrix $\mathbf{bcirc}(\mathcal{X})$ are defined as

$$\mathbf{bdiag}(\mathcal{X}) = \begin{bmatrix} \mathcal{X}^{(1)} & & & \\ & \mathcal{X}^{(2)} & & \\ & & \ddots & \\ & & & \mathcal{X}^{(K)} \end{bmatrix}$$

$$\mathbf{bcirc}(\mathcal{X}) = \begin{bmatrix} \mathcal{X}^{(1)} & \mathcal{X}^{(M)} & \cdots & \mathcal{X}^{(2)} \\ \mathcal{X}^{(2)} & \mathcal{X}^{(1)} & \cdots & \mathcal{X}^{(3)} \\ \vdots & \vdots & \ddots & \vdots \\ \mathcal{X}^{(M)} & \mathcal{X}^{(M-1)} & \cdots & \mathcal{X}^{(1)} \end{bmatrix}.$$

The block vectorization is defined as $\mathbf{bvec}(\mathcal{X}) = [\mathcal{X}^{(1)}; \ldots; \mathcal{X}^{(K)}]$. The inverse operations of **bvec** and **bdiag** are defined as $\mathbf{bvfold}(\mathbf{bvec}(\mathcal{X})) = \mathcal{X}$ and $\mathbf{bdfold}(\mathbf{bdiag}(\mathcal{X})) = \mathcal{X}$, respectively. Let $\mathcal{Y} \in \mathbb{R}^{N \times J \times K}$. The **t-product** $\mathcal{X} * \mathcal{Y}$ is an $M \times J \times K$ tensor, $\mathcal{X} * \mathcal{Y} = \mathbf{bvfold}(\mathbf{bcirc}(\mathcal{X}) * \mathbf{bvec}(\mathcal{Y}))$. The **transpose** of $\mathcal{X}$ is $\mathcal{X}^T \in \mathbb{R}^{N \times M \times K}$ by transposing each of the frontal slices and then reversing the order of transposed frontal slices 2 through K. The **identity tensor** $\mathcal{I} \in \mathbb{R}^{M \times M \times K}$ is a tensor whose first frontal slice is an $M \times M$ identity matrix and the rest frontal slices are zero. A tensor $\mathcal{X} \in \mathbb{R}^{M \times M \times K}$ is **orthogonal** if it satisfies $\mathcal{X}^T * \mathcal{X} = \mathcal{X} * \mathcal{X}^T = \mathcal{I}$.

Definition 1 (t-SVD): Given $\mathcal{X}$, its t-SVD is defined as

$$\mathcal{X} = \mathcal{U} * \mathcal{D} * \mathcal{V}^T$$

where $\mathcal{U} \in \mathbb{R}^{M \times M \times K}$ and $\mathcal{V} \in \mathbb{R}^{N \times N \times K}$ are orthogonal tensors, and $\mathcal{D} \in \mathbb{R}^{M \times N \times K}$ is an f-diagonal tensor. Each of its frontal slices is a diagonal matrix.

Definition 2 (t-SVD-TNN): The t-SVD-TNN of a tensor $\mathcal{X} \in \mathbb{R}^{M \times N \times K}$, denoted as $\|\mathcal{X}\|_{\circledast}$, is defined as the sum of singular values of all the frontal slices of $\hat{\mathcal{X}}$, that is

$$\|\mathcal{X}\|_{\circledast} = \sum_{i=1}^{\min\{M,N\}} \sum_{k=1}^{K} |\hat{\mathcal{D}}(i,i,k)|. \tag{1}$$

B. Markov Chain-Based Spectral Clustering

The main idea of spectral clustering is that each data point is represented as a vertex and pairwise similarities are measured by edges, and then the spectral clustering transforms the clustering task into a graph partition problem. Let $\mathcal{G} = (V, E, A)$ be a weighted graph where V is the vertex set, E is the edge set, and A is the similarity matrix. Suppose $X = [x_1, x_2, \ldots, x_n] \in \mathbb{R}^{d \times n}$ is the feature matrix, where d is the dimension of feature vector and n is the number of samples. One intuitive way to define the similarity $A_{i,j}$ between x_i and x_j is to use Gaussian kernels, i.e., $A_{i,j} = \exp(-((\|x_i - x_j\|_2^2)/\sigma^2))$, where σ^2 is the standard deviation. The overall procedure of the Markov chain-based spectral clustering method is summarized in Algorithm 1. For more details of the Markov chain-based spectral clustering theory, please refer to [1] and [44].

Algorithm 1 Markov Chain-Based Spectral Clustering

Input: Feature matrix $X = [x_1, x_2, \ldots, x_n]$;
Output: Assigned classes of all data points;

1: Compute the similarity matrix $A_{i,j} = \exp(-\frac{\|x_i - x_j\|_2^2}{\sigma^2})$;
2: Define a random walk over the weighted graph $\mathcal{G} = (V, E, A)$ with transition probability matrix $P = D^{-1}A \in \mathbb{R}^{n \times n}$ such that it has a unique stationary distribution π satisfying $\pi = P\pi$, D is a diagonal matrix with $D_{ii} = \Sigma_j A_{ij}$;
3: Construct the normalized Laplacian matrix $L = (\Pi^{\frac{1}{2}} P \Pi^{-\frac{1}{2}} + \Pi^{-\frac{1}{2}} P^T \Pi^{\frac{1}{2}})/2$, where Π is a diagonal matrix with $\Pi_{ii} = \pi(i)$;
4: Compute the r smallest generalized eigenvectors $U = [u_1, \ldots, u_r]$ satisfying $Lu = \lambda Du$;
5: Cluster U by k-means and assign each data point into a specific class.

IV. PROPOSED METHOD

In this section, we propose a novel multiview spectral clustering method by learning an adaptive affinity matrix

in Section IV-A. Then, we solve the optimization model by the ADMMs. The main idea of MCA^{2}M is that we want to learn a nonnegative and symmetric transition probability matrix without any postprocessing.

A. Objective Function of the Proposed MCA^{2}M

Given a database with n samples and M features (views) $\{X^{(v)}\}_{v=1}^{M}$, $X^{(v)} = [x_1^{(v)}, \ldots, x_n^{(v)}] \in \mathbb{R}^{d_v \times n}$ denotes the feature matrix and d_v is the feature dimension of the vth view. For each view, we first compute the similarity matrix $A^{(v)} \in \mathbb{R}^{n\times n}$ by $A_{i,j}^{(v)} = \exp(-((\|x_i^{(v)} - x_j^{(v)}\|_2^2)/\sigma^2))$ and then construct the transition probability matrix $P^{(v)}$, $v = 1, 2, \ldots, M$. RMSC [29] learns only the shared common transition probability matrix $\hat{P}$ among all views. One obvious shortcoming is that some specific information underlying different views may be ignored. Unlike RMSC, we hope to pursue high-order correlation among all views. As shown in Fig. 1(b), we store all $P^{(v)}$ as a frontal slice of the third-order tensor $\mathcal{P} \in \mathbb{R}^{n\times n\times M}$. Since the original samples usually are corrupted by noises and outliers, the multiview features are also erroneous. The transition probability matrices are perturbed. To alleviate the noise perturbation, we consider to separate the tensor $\mathcal{P}$ into a clean tensor $\mathcal{Z}$ and an error tensor $\mathcal{E}$, i.e., $\mathcal{P} = \mathcal{Z} + \mathcal{E}$. Considering that the sample number is usually larger than the cluster number, the tensor $\mathcal{Z}$ has the low-rank property and we adopt the t-SVD-TNN defined in (1) to depict the low rankness. Different from the existing methods [1], [7], [9], which construct the transition probability matrix S by two separate steps, we select to directly learn an adaptive S. In addition, two essential characteristics of the transition probability matrix are nonnegativity and symmetry since each entry is to measure the similarity between samples x_i and x_j. In this article, we pursue to learn an adaptive transition probability matrix with nonnegativity and symmetry. Incorporating the above-mentioned concerns into a unified model, the proposed MCA^{2}M model is formulated as follows:

$$\begin{aligned}\min_{\mathcal{Z},E,S} \|\mathcal{Z}\|_{\circledast} + \alpha\|E\|_{2,1} + \beta\sum_{v=1}^{M}\left\|\frac{|\mathcal{Z}^{(v)}| + |\mathcal{Z}^{(v)^T}|}{2} - S\right\|_F^2 \\ \text{s.t. } \mathcal{P} = \mathcal{Z} + \mathcal{E} \quad \forall i \ S_i \geq 0, \quad S_i^T\mathbf{1} = \mathbf{1} \\ E = [\mathcal{E}^{(1)}; \mathcal{E}^{(2)}; \ldots; \mathcal{E}^{(M)}] \end{aligned} \tag{2}$$

where α and β are nonnegative tradeoff parameters to balance the effects of all terms. The last constraint of (2) is to transform the error tensor $\mathcal{E} \in \mathbb{R}^{n\times n\times M}$ into an error matrix $E \in \mathbb{R}^{n^2\times M}$ by vertically concatenating all frontal slices along the column direction.

Remarks.

1) Different from the LRR-based clustering methods [7], [9], [24], [37], which are usually based on the self-representation property, the proposed MCA^{2}M model considers $\mathcal{P} = \mathcal{Z} + \mathcal{E}$ such that the matrix inversion operation with high computational cost can be fully avoided.
2) The constraint $S_i \geq 0$ has two implications: when samples x_i and x_j locate different groups, $S_{i,j} = 0$; when they belong to the same group, $S_{i,j} > 0$. $S_i^T\mathbf{1} = \mathbf{1}$ aims to guarantee the probability property of S_i.
3) Borrowing the last term of Eq. (2) and the second remark, we can yield a nonnegative and symmetric S to better uncover the true affinity between data points.
4) The proposed MCA^{2}M adopts the $l_{2,1}$-norm to alleviate the noise perturbation.
5) The proposed MCA^{2}M model is highly inspired by [1], which learns the transition probability tensor $\mathcal{Z}$ first and then learns the transition probability matrix S sequentially. This two-step paradigm is prone to yield a suboptimal S. To overcome this limitation, (2) incorporates the one-step strategy to jointly learn $\mathcal{Z}$ and S in a mutual reinforcement manner such that they can help each other.

B. Optimization of the Proposed MCA^{2}M

It may be difficult to solve (2) directly by ADMM since variable $\mathcal{Z}$ is coupled by the first term, the third item, and the first constraint of (2). To solve this challenging optimization model, we first split the variable $\mathcal{Z}$ by introducing a new auxiliary variable $\mathcal{H}$ and equivalently reformulate (2) as the following optimization problem:

$$\begin{aligned}\min_{\mathcal{H},\mathcal{Z},E,S} \|\mathcal{H}\|_{\circledast} + \alpha\|E\|_{2,1} + \beta\sum_{v=1}^{M}\left\|\frac{|\mathcal{Z}^{(v)}| + |\mathcal{Z}^{(v)^T}|}{2} - S\right\|_F^2 \\ \text{s.t. } \mathcal{P} = \mathcal{Z} + \mathcal{E} \quad \forall i \ S_i \geq 0, \ S_i^T\mathbf{1} = \mathbf{1} \\ E = [\mathcal{E}^{(1)}; \mathcal{E}^{(2)}; \ldots; \mathcal{E}^{(M)}], \quad \mathcal{H} = \mathcal{Z}. \end{aligned} \tag{3}$$

The main idea of the ADMM technique is to solve a constrained problem by its unconstrained augmented Lagrangian function and then iteratively update each variable by keeping other variables fixed at their latest values [45]. The unconstrained augmented Lagrangian function is given by

$$\begin{aligned}&\mathcal{L}_\mu(\mathcal{H}, \mathcal{Z}, E, S; \Theta, \Lambda) \\ &= \beta\sum_{v=1}^{M}\left\|\frac{|\mathcal{Z}^{(v)}| + |\mathcal{Z}^{(v)^T}|}{2} - S\right\|_F^2 \\ &\quad + \|\mathcal{H}\|_{\circledast} + \alpha\|E\|_{2,1} + \langle\Lambda, \mathcal{H} - \mathcal{Z}\rangle + \frac{\mu}{2}\|\mathcal{H} - \mathcal{Z}\|_F^2 \\ &\quad + \langle\Theta, \mathcal{P} - \mathcal{Z} - \mathcal{E}\rangle + \frac{\mu}{2}\|\mathcal{P} - \mathcal{Z} - \mathcal{E}\|_F^2 \end{aligned} \tag{4}$$

where Θ and Λ are two Lagrangian multipliers of size $N \times N \times M$. μ is the penalty parameter. $\langle\cdot\rangle$ denotes the standard trace inner product. Following ADMM, all variables are iteratively updated as follows:

$$\mathcal{H}_{k+1} = \arg\min_{\mathcal{H}} \ \mathcal{L}_{\mu_k}(\mathcal{H}, \mathcal{Z}_k, E_k, S_k; \Theta_k, \Lambda_k) \tag{5}$$

$$\mathcal{Z}_{k+1} = \arg\min_{\mathcal{Z}} \ \mathcal{L}_{\mu_k}(\mathcal{H}_{k+1}, \mathcal{Z}, E_k, S_k; \Theta_k, \Lambda_k) \tag{6}$$

$$E_{k+1} = \arg\min_{E} \ \mathcal{L}_{\mu_k}(\mathcal{H}_{k+1}, \mathcal{Z}_{k+1}, E, S_k; \Theta_k, \Lambda_k) \tag{7}$$

$$S_{k+1} = \arg\min_{S} \ \mathcal{L}_{\mu_k}(\mathcal{H}_{k+1}, \mathcal{Z}_{k+1}, E_{k+1}, S; \Theta_k, \Lambda_k). \tag{8}$$

Specifically, given the kth update, the $k+1$th iteration of each subproblem is presented as follows.

1) *Subproblem $\mathcal{H}$:* The optimization of (5) with respect to $\mathcal{H}$ is

$$\arg\min_{\mathcal{H}} \|\mathcal{H}\|_{\circledast} + \frac{\mu_k}{2}\left\|\mathcal{H} - \left(\mathcal{Z}_k - \frac{\Lambda_k}{\mu_k}\right)\right\|_F^2. \tag{9}$$

Denoting $\mathcal{F}_k = \mathcal{Z}_k - (\Lambda_k/\mu_k)$, the closed-form solution of (9) can be obtained by the tensor tubal-shrinkage operator [7]

$$\mathcal{H}_{k+1} = \mathcal{C}_{\frac{V}{\rho_k}}(\mathcal{F}_k) = \mathcal{U} * \mathcal{C}_{\frac{V}{\rho_k}}(\mathcal{D}) * \mathcal{V}^T \tag{10}$$

where $\mathcal{F}_k = \mathcal{U} * \mathcal{D} * \mathcal{V}^T$ is the t-SVD of $\mathcal{F}_k$, and $\mathcal{C}_{(V/\rho_k)}(\mathcal{D}) = \mathcal{D} * \mathcal{J}$, in which $\mathcal{J}$ is an f-diagonal tensor whose diagonal element in the Fourier domain is $\mathcal{J}(i,i,k) = \max\{1 - ((V/\rho_k)/(\mathcal{D}(i,i,k))), 0\}$.

2) *Subproblem $\mathcal{Z}$:* The optimization of (6) with respect to $\mathcal{Z}$ is written as

$$\begin{aligned}\arg\min_{\mathcal{Z}} \ & \beta \sum_{v=1}^{M} \left\| \frac{|\mathcal{Z}^{(v)}| + |\mathcal{Z}^{(v)^T}|}{2} - S_k \right\|_F^2 \\ & + \frac{\mu_k}{2}\|\mathcal{Z} - \mathcal{A}_k\|_F^2 + \frac{\mu_k}{2}\|\mathcal{Z} - \mathcal{B}_k\|_F^2 \\ = \arg\min_{\mathcal{Z}} & \beta \sum_{v=1}^{M} \left\| \frac{|\mathcal{Z}^{(v)}| + |\mathcal{Z}^{(v)^T}|}{2} - S_k \right\|_F^2 \\ & + \mu_k \left\| \mathcal{Z} - \frac{\mathcal{A}_k + \mathcal{B}_k}{2} \right\|_F^2 \end{aligned} \tag{11}$$

where $\mathcal{A}_k = \mathcal{P} - \mathcal{E}_k + (\Theta_k/\mu_k)$ and $\mathcal{B}_k = \mathcal{H}_{k+1} + (\Lambda_k/\mu_k)$. It is intractable to directly solve (11) due to the absolute value operator as shown in the first term. However, one important observation is that the other of (11) is free to the absolute value operator [46]. Thus, the elements of $\mathcal{Z}_{k+1}$ must have the same sign as the ones in $((\mathcal{A}_k + \mathcal{B}_k)/2)$. Inspired by this observation, $\mathcal{Z}_{k+1}$ can be obtained by

$$\mathcal{Z}_{k+1} = \hat{\mathcal{Z}}_{k+1} \otimes \operatorname{sign}\left(\frac{\mathcal{A}_k + \mathcal{B}_k}{2}\right) \tag{12}$$

and $\otimes$ represents the elementwise multiplication; $\hat{\mathcal{Z}}_{k+1}$ is the closed-form solution of the following optimization problem:

$$\min_{\hat{\mathcal{Z}}} \ \beta \sum_{v=1}^{M} \left\| \hat{Z}^{(v)} - \frac{S_k + S_k^T}{2} \right\|_F^2 + \mu_k \left\| \hat{\mathcal{Z}} - \left| \frac{\mathcal{A}_k + \mathcal{B}_k}{2} \right| \right\|_F^2. \tag{13}$$

Specifically, (13) can be separated into M independent minimization problems and the vth problem is

$$\begin{aligned}\min_{\hat{Z}} \ & \beta \left\| \hat{Z}^{(v)} - \frac{S_k + S_k^T}{2} \right\|_F^2 + \mu_k \left\| \hat{Z} - \left| \frac{A_k^{(v)} + B_k^{(v)}}{2} \right| \right\|_F^2 \\ = \min_{\hat{Z}} & (\beta + \mu_k) \left\| \hat{Z}_k^{(v)} - \frac{\beta\left(S_k + S_k^T\right) + \mu_k |\mathcal{A}_k + \mathcal{B}_k|}{2 * (\beta + \mu_k)} \right\|_F^2. \end{aligned}$$

The closed-form solution $\hat{Z}_{k+1}^{(v)}$ is

$$\hat{Z}_{k+1}^{(v)} = \frac{\beta\left(S_k + S_k^T\right) + \mu_k |\mathcal{A}_k + \mathcal{B}_k|}{2 * (\beta + \mu_k)}. \tag{14}$$

3) *Subproblem E:* The optimization of (7) with respect to E is written as

$$\arg\min_{E} \ \frac{\alpha}{\mu_k}\|E\|_{2,1} + \frac{1}{2}\|\mathcal{E} - \mathcal{C}_k\|_F^2, \tag{15}$$

Algorithm 2 MCA^{2}M for Multiview Spectral Clustering

Input: multiview features: $X^{(v)}$ $(v = 1, 2, \cdots, V)$; parameters: α, β;

Initialize: $\mathcal{H}_0$, $\mathcal{Z}_0$, E_0, Θ_0, Λ_0 initialized to $\mathbf{0}$; $\rho_0 = 10^{-3}$, $\lambda = 2$, $tol = 10^{-7}$, $k = 0$

1: Compute $L^{(i)}$ and $P^{(i)}$ via Steps 2 and 3 in Algorithm 1;
2: **while** not converged **do**
3: Update $\mathcal{H}_{k+1}$ via Eq. (10);
4: Update $\mathcal{Z}_{k+1}$ via Eq. (12);
5: Update E_{k+1} via Eq. (17);
6: Update S_{k+1} via Eq. (18);
7: Update Θ_{k+1}, Λ_{k+1} and μ_{k+1} via Eqs. (19), (20), and (21), respectively;
8: Check the convergence conditions
9:

$$\max\left\{ \begin{matrix} \|\mathcal{P} - \mathcal{Z}_{k+1} - \mathcal{E}_{k+1}\|_\infty \\ \|\mathcal{H}_{k+1} - \mathcal{Z}_{k+1}\|_\infty \end{matrix} \right\} \le tol,$$

10: **end while**

Output: Adaptive transition probability matrix S_k.

where $\mathcal{C}_k = \mathcal{P} - \mathcal{Z}_{k+1} + (\Theta_k/\mu_k)$. After vertically concatenating together along the column of $\mathcal{C}_k$ into a matrix $C_k = [\mathcal{C}_k^{(1)}; \mathcal{C}_k^{(2)}; \ldots; \mathcal{C}_k^{(M)}]$, (15) is reformulated as

$$\arg\min_{E} \ \frac{\alpha}{\mu_k}\|E\|_{2,1} + \frac{1}{2}\|E - C_k\|_F^2, \tag{16}$$

and the jth column of E_{k+1} is

$$\begin{cases} \dfrac{\|C_k(:,j)\|_2 - \frac{\alpha}{\mu_k}}{\|C_k(:,j)\|_2} C_k(:,j), & \text{if } \dfrac{\alpha}{\mu_k} < \|C_k(:,j)\|_2 \\ 0, & \text{otherwise.} \end{cases} \tag{17}$$

4) *Subproblem S:* The optimization of (8) with respect to S is written as

$$\begin{aligned}\arg\min_{S} \ & \sum_{v=1}^{M} \left\| S - \frac{\left|\mathcal{Z}_{k+1}^{(v)}\right| + \left|\mathcal{Z}_{k+1}^{(v)^T}\right|}{2} \right\|_F^2 \\ \text{s.t. } & \forall i \ S_i \ge 0, \quad S_i^T \mathbf{1} = \mathbf{1}. \end{aligned} \tag{18}$$

It is easy to see that (18) can be separated into n independent problems and each of them is a proximal operator problem with a probabilistic simplex constraint. The projection algorithm [47], [48] was adopted to obtain the closed-form solution.

5) *Subproblem Θ, Λ and μ:* Two Lagrangian multipliers Θ_{k+1}, Λ_{k+1} and the penalty parameter μ_{k+1} are updated by

$$\Theta_{k+1} = \Theta_k + \mu_k(\mathcal{P} - \mathcal{Z}_{k+1} - \mathcal{E}_{k+1}) \tag{19}$$

$$\Lambda_{k+1} = \Lambda_k + \mu_k(\mathcal{H}_{k+1} - \mathcal{Z}_{k+1}) \tag{20}$$

$$\mu_{k+1} = \min\{\lambda * \mu_k, \mu_{\max}\} \tag{21}$$

where λ is set to 2. Equation (21) is known as a continuation scheme [49]. The details of the MCA^{2}M

algorithm are given in Algorithm 2. The convergence conditions of Algorithm 2 are defined as

$$\max\left\{\begin{array}{l}\|\mathcal{P}-\mathcal{Z}_{k+1}-\mathcal{E}_{k+1}\|_\infty\\ \|\mathcal{H}_{k+1}-\mathcal{Z}_{k+1}\|_\infty\end{array}\right\}\leq \text{tol} \tag{22}$$

where tol $>$ 0 is a predefined tolerance. Once the adaptive transition probability matrix S is yielded by Algorithm 2, the final clustering results are obtained by replacing the transition probability matrix P in the second step of Algorithm 1 with the learned S.

V. Experiments

To empirically investigate the performance of the proposed MCA^{2}M, extensive experiments on eight real-world and challenging multiview databases over three SVC methods and seventeen multiview clustering ones. All experiments are conducted on an Intel Core i3 3.50-GHz Win7 workstation with 16-GB memory.

A. Experimental Settings

1) Databases: The details of all test databases are described as follows.

1) *Still-DB [50]:* It is a still image database for action recognition. There are 467 images corresponding to six different actions. Three types of features, including 200d sift bow, 200d color sift bow, and 200d shape context bow, are exploited in still-DB.
2) *BBCSport:*[1] It is a news story database which has 544 documents from the BBC sport website. This database has two different types of features with five classes.
3) *UCI Digits:*[2] It is a handwritten digit database from the UCI repository. There are 2000 examples in total and ten classes (handwritten digits 0–9). Three feature sets, including 76d Fourier coefficients of the character shapes, 216d profile correlations, and 6d Karhunen–Love coefficients are extracted.
4) *100leaves:*[3] It is a plant species leaves database with 1600 samples in total (100 categories). Three features, including shape descriptor, fine-scale margin, and texture histogram, are exploited.
5) *COIL-20:*[4] It is an object database which contains 1440 images with 20 object categories. We employed three different types of features including 1024d intensity, 3304d LBP, and 6750d Gabor.
6) *Flowers:*[5] It is a flower database which contains 1360 samples in total with 17 flower categories. We extracted three different visual features, including color, texture, and shape, with the same size 1360d.
7) *MITIndoor:* It is a scene database which includes 5360 images. All images belong to 67 categories and three types of handcrafted visual features are extracted.
8) *Caltech101:* It is an object database which has 8677 object images with 101 categories. The same to [7], four types of features are selected.

[1]http://mlg.ucd.ie/datasets/segment.html
[2]http://archive.ics.uci.edu/ml/datasets/Multiple+Features
[3]https://archive.ics.uci.edu/ml/datasets/One-hundred+plant+species+leaves+data+set
[4]http://www.cs.columbia.edu/CAVE/software/softlib/
[5]http://www.robots.ox.ac.uk/vgg/data/flowers/

TABLE III
Summary of All Real-World Multiview Databases

Database	Instance	Cluster	View1	View2	View3	View4
Sitll-DB	467	6	200d	200d	200d	-
BBCSport	544	5	3183d	3203d	-	-
UCI-Digits	2000	10	240d	76d	6d	-
100leaves	1600	100	64d	64d	64d	-
COIL-20	1440	20	1024d	3304d	6750d	-
Flowers	1360	17	1360d	1360d	1360d	-
MITIndoor	5360	67	4096d	3600d	1770d	-
Caltech101	8677	101	4096d	3600d	1770d	1240d

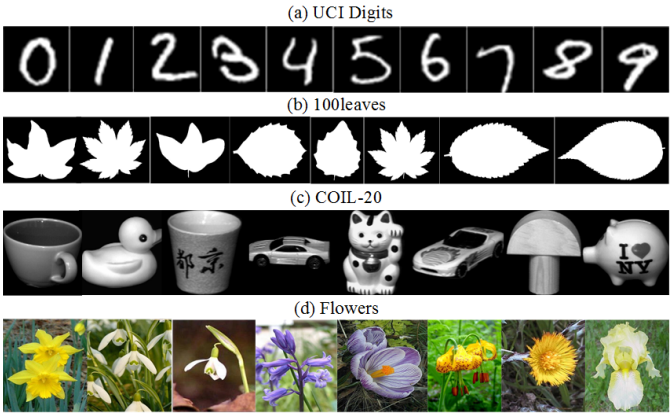

Fig. 2. Samples of (a) UCI digits, (b) 100leaves, (c) COIL-20, and (d) Flowers.

All databases are briefly summarized in Table III and some examples are shown in Fig. 2.

2) Competitors: The proposed MCA^{2}M method is compared with the following state-of-the-art clustering methods:

1) *Single-View Competitors:* SPC [51] achieves the best result among all views via standard spectral clustering; SSC [18] achieves the best result among all views via SSC; LRR [16] achieves the best result among all views via LRR.
2) *Multiview Competitors:* RMSC [29] recovers a shared low-rank transition probability matrix via low-rank and sparse matrix decomposition; DiMSC [52] utilizes the Hilbert Schmidt independence criterion as a diversity term to explore the complementarity of multiview features. LT-MSC [9] exploits the low-rank tensor constraint for multiview clustering. MVCC [53] uses the concept of factorization with local manifold regularization. DSemi-NMF [54] adopts the deep nonnegative matrix factorization to capture the hidden information. MLAN [55] selects the adaptive neighbors for multiview clustering. ECMSC [56] takes the exclusivity and consistency of multiview features into consideration for multiview clustering. t-SVD-MSC [7] uses the multirank tensor minimization to capture the high-order correlation. ETLMSC [1] learns an essential

TABLE IV
CLUSTERING RESULTS (MEAN ± STANDARD DEVIATION) ON STILL-DB DATABASE ($\alpha = 0.0007,\ \beta = 0.007$)

Type	Method	ACC↑	NMI↑	AR↑	F-score↑	Precision↑	Recall↑
SVC	$\mathrm{SPC_{best}}$	0.294±0.009	0.105±0.008	0.061±0.000	0.219±0.006	0.222±0.001	0.216±0.002
	$\mathrm{SSC_{best}}$	0.328±0.002	0.138±0.003	0.087±0.001	0.295±0.001	0.222±0.001	0.440±0.007
	$\mathrm{LRR_{best}}$	0.306±0.004	0.109±0.003	0.061±0.004	0.219±0.000	0.223±0.003	0.212±0.004
MVC (Graph)	DSemi-NMF	0.264±0.001	0.055±0.000	0.026±0.001	0.198±0.000	0.192±0.001	0.203±0.001
	MLAN	0.349±0.000	0.138±0.000	0.098±0.000	0.272±0.000	0.242±0.000	0.310±0.000
	GMC	0.251±0.000	0.078±0.000	0.005±0.000	0.278±0.000	0.174±0.000	0.701±0.000
	MCGC	0.308±0.000	0.073±0.000	0.062±0.000	0.231±0.000	0.220±0.000	0.244±0.000
	LMVSC	0.317±0.000	0.189±0.000	0.066±0.000	0.239±0.000	0.221±0.000	0.259±0.000
MVC (LRR)	DiMSC	0.323±0.002	0.122±0.008	0.083±0.001	0.249±0.000	0.235±0.004	0.256±0.002
	LT-MSC	0.342±0.002	0.136±0.002	0.090±0.001	0.252±0.002	0.243±0.001	0.261±0.003
	MVCC	0.251±0.000	0.078±0.000	0.005±0.000	0.278±0.000	0.174±0.000	**0.702±0.000**
	ECMSC	0.320±0.008	0.111±0.007	0.090±0.001	0.264±0.010	0.237±0.006	0.300±0.031
	t-SVD-MSC	0.347±0.010	0.130±0.004	0.088±0.003	0.255±0.004	0.239±0.002	0.273±0.006
	LMSC	0.327±0.003	0.136±0.003	0.084±0.011	0.269±0.005	0.235±0.007	0.247±0.012
	GLTA	0.366±0.007	0.126±0.005	0.102±0.005	0.262±0.003	0.251±0.004	0.275±0.003
	$\mathrm{SM^2SC}$	0.452±0.002	0.336±0.002	0.312±0.002	0.370±0.002	0.381±0.003	0.378±0.002
	SCMV-3DT	0.330±0.000	0.119±0.001	0.069±0.001	0.228±0.001	0.229±0.001	0.227±0.001
MVC (RPCA)	RMSC	0.305±0.010	0.089±0.009	0.073±0.011	0.221±0.002	0.231±0.004	0.219±0.002
	ETLMSC	0.604±0.043	0.520±0.015	0.423±0.029	0.523±0.024	0.518±0.022	0.528±0.027
	EMVC	0.303±0.014	0.121±0.003	0.071±0.002	0.231±0.001	0.230±0.002	0.231±0.002
Ours	$\mathrm{MCA^2M}$-$\mathcal{Z}$	0.634±0.060	0.593±0.043	0.487±0.068	0.576±0.057	0.573±0.055	0.578±0.060
	$\mathrm{MCA^2M}$-S	**0.648±0.052**	**0.597±0.036**	**0.491±0.062**	**0.578±0.051**	**0.579±0.049**	0.579±0.054

transition probability matrix tensor for multiview clustering. LMSC [37] uses a latent representation for multiview clustering. GMC [57] is a graph-based multiview clustering. GLTA [24] learns the representation tensor and affinity matrix simultaneously. $\mathrm{SM^2SC}$ [40] refers to split multiplicative multiview subspace clustering. SCMV-3DT [2] uses the tensorial t-product representation. MCGC [14] learns a consensus graph with minimizing disagreement between different views. EMVC [30] incorporates the error learning into RMSC. LMVSC [28] refers to the large-scale multiview subspace clustering with linear complexity.

Note that DiMSC, LT-MSC, MVCC, ECMSC, t-SVD-MSC, LMSC, GLTA, $\mathrm{SM^2SC}$, and SCMV-3DT are LRR-based multiview clustering methods. RMSC and ETLMSC are representatives of RPCA-based multiview clustering methods.

For the parameter setting, we follow the original articles. Specifically, the details are summarized as follows: SSC and LRR search parameters in the range $[0.01, 10]$. One tradeoff parameter of RMSC is selected from $[0.005, 1]$. Two parameters of DiMSC range in $[0.01, 0.03]$ and [20:20:180]. LT-MSC selects the parameter in $[0.01, 100]$. In MVCC, the first two parameters are set to 5 and 100, while the other two ones range in $[0.001, 1000]$, respectively. The layer sizes of DSemi-NMF are set to {$[100, 50]$, [500, 50], $[500, 200]$} and other parameters are defaulted. MLAN has one random parameter ranging in $[1, 30]$. We set three parameters of ECMSC as $[0.1, 1]$, $[0.1, 1]$, and 1.2, respectively. T-SVD-MSC selects one parameter from the interval $[0.1, 2]$. ETLMSC varies in the range of $[0.008, 0.1]$. The dimensions of the latent representation and the regularization parameter of LMSC are searched from the range [10:10:100] and $[0.01, 0.1, 1, 10, 100]$, respectively. GMC sets the tradeoff parameter as 1. Two free parameters of GLTA are selected from the interval $[0.001, 1]$ and $[0.01, 100]$ respectively. $\mathrm{SM^2SC}$ tunes three parameters in the range [0.1, 0.15, 0.2, 0.3, 0.4, 0.5, 1, 10, 40, 100], [0.1, 0.5, 1, 1.5, 2], and [0.05, 0.1, 0.4, 1, 5]. There are three free parameters in SCMV-3DT are set to different values from $[10^{-5}, 10^{-4}, \ldots, 10]$. MCGC first learns each viewgraph by setting k neighbor and then introduces a tradeoff parameter to learn the consensus graph. Two parameters are selected from [3:1:15] and [0.6:5:50]. EMVC selects two parameters from $[10^{-3}, 10^{-2}, \ldots, 10^{4}]$. LMVSC searches anchor number in the range $[k, 50, 100]$ and one parameter in $[0.001, 0.01, 0.1, 1, 10]$, where k denotes the number of clusters.

3) Evaluation Measures: Following [1], [7], and [24], we use six common evaluation measures to estimate the clustering performance of all methods. They are accuracy (ACC), normalized mutual information (NMI), adjusted rank index (AR), F-score, Precision, and Recall. For each evaluation measure, a higher value stands for better clustering performance. Besides, they measure different properties of clustering, and thus, give a comprehensive evaluation of all methods. For all competing methods, we follow their parameter settings and record their best results. As for the parameter setting of the proposed $\mathrm{MCA^2M}$, one can refer to Section V-B.

B. Clustering Results and Analysis

In this section, the performance comparisons of all clustering methods are first given. Several aspects of the proposed $\mathrm{MCA^2M}$, including parameter selection, convergence analysis, and running time, are also reported.

1) Performance Comparison: Tables IV–XI report the detailed clustering performance comparisons of all methods. For a fair evaluation, we run ten repetitions for each

TABLE V
CLUSTERING RESULTS (MEAN ± STANDARD DEVIATION) ON BBCSPORT DATABASE ($\alpha = 0.02,\ \beta = 0.005$)

Type	Method	ACC↑	NMI↑	AR↑	F-score↑	Precision↑	Recall↑
SVC	$\text{SPC}_{\mathbf{best}}$	0.719±0.000	0.482±0.002	0.460±0.001	0.581±0.001	0.616±0.002	0.550±0.001
	$\text{SSC}_{\mathbf{best}}$	0.627±0.003	0.534±0.008	0.364±0.007	0.565±0.005	0.427±0.004	0.834±0.004
	$\text{LRR}_{\mathbf{best}}$	0.836±0.001	0.698±0.002	0.705±0.001	0.776±0.001	0.768±0.001	0.784±0.001
MVC (Graph)	DSemi-NMF	0.320±0.002	0.061±0.001	0.035±0.001	0.258±0.001	0.266±0.001	0.250±0.002
	MLAN	0.721±0.000	0.779±0.000	0.591±0.000	0.714±0.000	0.567±0.000	0.962±0.000
	GMC	0.807±0.000	0.760±0.000	0.722±0.000	0.794±0.000	0.727±0.000	0.875±0.000
	MCGC	0.426±0.000	0.131±0.000	0.087±0.000	0.353±0.000	0.289±0.000	0.454±0.000
	LMVSC	0.517±0.000	0.382±0.000	0.151±0.000	0.394±0.000	0.329±0.000	0.491±0.000
MVC (LRR)	DiMSC	0.922±0.000	0.785±0.000	0.813±0.000	0.858±0.000	0.846±0.000	0.872±0.000
	LT-MSC	0.460±0.046	0.222±0.028	0.167±0.043	0.428±0.014	0.328±0.028	0.629±0.053
	MVCC	0.928±0.000	0.816±0.000	0.831±0.000	0.870±0.000	0.889±0.000	0.853±0.000
	ECMSC	0.285±0.014	0.027±0.013	0.009±0.011	0.267±0.020	0.244±0.007	0.297±0.045
	t-SVD-MSC	0.879±0.000	0.765±0.000	0.784±0.000	0.834±0.000	0.863±0.000	0.807±0.000
	LMSC	0.847±0.003	0.739±0.001	0.749±0.001	0.810±0.001	0.799±0.001	0.822±0.001
	GLTA	0.939±0.000	0.825±0.000	0.849±0.000	0.885±0.000	0.890±0.000	0.880±0.000
	SM^2SC	0.982±0.000	0.937±0.000	0.952±0.000	0.963±0.000	0.970±0.000	0.957±0.000
	SCMV-3DT	0.980±0.000	0.929±0.000	0.935±0.000	0.950±0.000	0.959±0.000	0.942±0.000
MVC (RPCA)	RMSC	0.826±0.001	0.666±0.001	0.637±0.001	0.719±0.001	0.766±0.001	0.677±0.001
	ETLMSC	0.959±0.086	0.972±0.058	0.949±0.107	0.961±0.081	0.963±0.078	0.960±0.085
	EMVC	0.826±0.005	0.666±0.011	0.635±0.009	0.717±0.007	0.764±0.007	0.675±0.008
Ours	$\text{MCA}^2\text{M-}\mathcal{Z}$	0.963±0.074	0.972±0.045	0.962±0.075	0.971±0.057	0.970±0.060	0.972±0.052
	$\text{MCA}^2\text{M-}S$	**0.998±0.000**	**0.993±0.000**	**0.997±0.000**	**0.997±0.000**	**0.998±0.000**	**0.997±0.000**

TABLE VI
CLUSTERING RESULTS (MEAN ± STANDARD DEVIATION) ON UCI-DIGITS DATABASE ($\alpha = 0.007,\ \beta = 0.005$)

Type	Method	ACC↑	NMI↑	AR↑	F-score↑	Precision↑	Recall↑
SVC	$\text{SPC}_{\mathbf{best}}$	0.732±0.011	0.642±0.005	0.545±0.012	0.591±0.011	0.582±0.013	0.601±0.011
	$\text{SSC}_{\mathbf{best}}$	0.815±0.011	0.840±0.001	0.770±0.005	0.794±0.004	0.747±0.010	0.848±0.004
	$\text{LRR}_{\mathbf{best}}$	0.871±0.001	0.768±0.002	0.736±0.002	0.763±0.002	0.759±0.002	0.767±0.002
MVC (Graph)	MLAN	0.874±0.000	0.910±0.000	0.847±0.000	0.864±0.000	0.797±0.000	0.943±0.000
	GMC	0.736±0.000	0.815±0.000	0.678±0.000	0.713±0.000	0.644±0.000	0.799±0.000
	MCGC	0.913±0.000	0.854±0.000	0.830±0.000	0.847±0.000	0.847±0.000	0.848±0.000
	LMVSC	0.790±0.000	0.756±0.000	0.643±0.000	0.681±0.000	0.643±0.000	0.722±0.000
MVC (LRR)	DiMSC	0.703±0.010	0.772±0.006	0.652±0.006	0.695±0.006	0.673±0.005	0.718±0.007
	LT-MSC	0.803±0.001	0.775±0.001	0.725±0.001	0.753±0.001	0.739±0.001	0.767±0.001
	MVCC	0.914±0.001	0.871±0.001	0.832±0.001	0.849±0.001	0.839±0.001	0.858±0.001
	ECMSC	0.718±0.001	0.780±0.001	0.672±0.001	0.707±0.001	0.660±0.001	0.760±0.001
	t-SVD-MSC	0.955±0.000	0.932±0.000	0.924±0.000	0.932±0.000	0.930±0.000	0.934±0.000
	LMSC	0.893±0.000	0.815±0.000	0.783±0.000	0.805±0.000	0.798±0.000	0.812±0.000
	GLTA	0.815±0.001	0.768±0.001	0.707±0.001	0.737±0.001	0.731±0.001	0.743±0.001
	SM^2SC	0.961±0.001	0.914±0.001	0.914±0.001	0.923±0.001	0.922±0.001	0.924±0.001
	SCMV-3DT	0.930±0.001	0.861±0.001	0.846±0.001	0.861±0.001	0.859±0.001	0.864±0.001
MVC (RPCA)	RMSC	0.915±0.024	0.822±0.008	0.789±0.014	0.811±0.012	0.797±0.017	0.826±0.006
	ETLMSC	0.958±0.078	0.977±0.028	0.953±0.069	0.958±0.062	0.940±0.088	0.980±0.029
	EMVC	0.793±0.044	0.757±0.021	0.680±0.041	0.713±0.036	0.699±0.044	0.728±0.030
Ours	$\text{MCA}^2\text{M-}\mathcal{Z}$	0.956±0.069	0.979±0.027	0.956±0.066	0.960±0.059	0.945±0.084	0.978±0.030
	$\text{MCA}^2\text{M-}S$	**0.972±0.051**	**0.982±0.019**	**0.967±0.052**	**0.971±0.046**	**0.959±0.072**	**0.985±0.017**

experiment of all methods and report their average values with standard deviations. The bold and underlined numbers denote the best and second-best results, respectively. In Tables IV–X, all competing methods can be roughly categorized into four groups: SVC, multiview graph clustering, LRR-based multiview subspace clustering, and RPCA-based multiview spectral clustering. Note that $\text{MCA}^2\text{M-}\mathcal{Z}$ and $\text{MCA}^2\text{M-}S$ denote the results by variables $\mathcal{Z}$ and S in Eq. (2), respectively.

One can see that the proposed MCA^2M method almost consistently outperforms other methods including the recently proposed t-SVD-MSC, ETLMSC, GLTA, and SM^2SC, especially for COIL-20 and Still-DB databases. For example, MCA^2M improve around 6.5% and 3.0% ACC over the runner-up on COIL-20 and Still-DB databases, respectively. This indicates that the transition probability matrix i.e., S learned from the proposed MCA^2M is effective for the multiview clustering task. ***Single-view versus Multiview:*** In most cases, most of the multiview clustering methods including the proposed MCA^2M perform better than the three SVC methods. Specifically, all multiview clustering methods achieve at least

TABLE VII

CLUSTERING RESULTS (MEAN ± STANDARD DEVIATION) ON 100LEAVES DATABASE ($\alpha = 0.005,\ \beta = 0.009$)

Type	Method	ACC↑	NMI↑	AR↑	F-score↑	Precision↑	Recall↑
SVC	$\text{SPC}_{\mathbf{best}}$	0.535±0.011	0.748±0.004	0.388±0.011	0.394±0.011	0.379±0.012	0.411±0.011
	$\text{SSC}_{\mathbf{best}}$	0.627±0.012	0.814±0.005	0.508±0.008	0.513±0.008	0.479±0.010	0.552±0.009
	$\text{LRR}_{\mathbf{best}}$	0.528±0.011	0.752±0.004	0.398±0.009	0.404±0.009	0.387±0.008	0.423±0.012
MVC (Graph)	DSemi-NMF	0.234±0.004	0.541±0.002	0.088±0.003	0.097±0.003	0.090±0.003	0.105±0.003
	MLAN	0.883±0.001	0.950±0.001	0.830±0.001	0.823±0.012	0.791±0.018	0.858±0.007
	GMC	0.824±0.000	0.929±0.000	0.497±0.000	0.504±0.000	0.352±0.000	0.887±0.000
	MCGC	0.810±0.000	0.887±0.000	0.706±0.000	0.709±0.000	0.696±0.000	0.724±0.000
	LMVSC	0.559±0.000	0.786±0.000	0.447±0.000	0.453±0.000	0.408±0.000	0.510±0.000
MVC (LRR)	DiMSC	0.721±0.024	0.873±0.007	0.638±0.021	0.641±0.021	0.607±0.025	0.680±0.018
	LT-MSC	0.736±0.007	0.870±0.006	0.641±0.012	0.644±0.012	0.615±0.012	0.678±0.013
	MVCC	0.498±0.011	0.740±0.004	0.356±0.009	0.362±0.008	0.332±0.009	0.398±0.008
	ECMSC	0.733±0.005	0.863±0.004	0.628±0.009	0.632±0.009	0.602±0.009	0.665±0.011
	t-SVD-MSC	**0.923±0.014**	**0.983±0.002**	**0.921±0.012**	**0.922±0.012**	**0.883±0.018**	**0.964±0.005**
	LMSC	0.766±0.015	0.892±0.004	0.686±0.011	0.689±0.011	0.655±0.013	0.725±0.010
	GLTA	0.826±0.012	0.926±0.004	0.772±0.011	0.775±0.011	0.740±0.014	0.813±0.010
	SM^2SC	0.804±0.011	0.815±0.005	0.707±0.021	0.710±0.020	0.651±0.030	0.782±0.013
	SCMV-3DT	0.720±0.010	0.874±0.004	0.632±0.011	0.636±0.011	0.588±0.013	0.692±0.009
MVC (RPCA)	RMSC	0.711±0.026	0.875±0.008	0.630±0.025	0.634±0.025	0.595±0.027	0.679±0.022
	ETLMSC	0.836±0.022	0.962±0.005	0.838±0.020	0.839±0.020	0.782±0.028	0.905±0.014
	EMVC	0.681±0.014	0.851±0.007	0.584±0.014	0.589±0.014	0.554±0.016	0.629±0.015
Ours	$\text{MCA}^2\text{M-}\mathcal{Z}$	0.872±0.023	0.961±0.007	0.855±0.026	0.856±0.025	0.822±0.032	0.893±0.019
	$\text{MCA}^2\text{M-}S$	0.873±0.012	0.965±0.003	0.859±0.011	0.861±0.012	0.820±0.014	0.906±0.010

TABLE VIII

CLUSTERING RESULTS (MEAN ± STANDARD DEVIATION) ON COIL-20 DATABASE ($\alpha = 0.005,\ \beta = 0.013$)

Type	Method	ACC↑	NMI↑	AR↑	F-score↑	Precision↑	Recall↑
SVC	$\text{SPC}_{\mathbf{best}}$	0.672±0.063	0.806±0.008	0.619±0.018	0.640±0.017	0.596±0.021	0.692±0.013
	$\text{SSC}_{\mathbf{best}}$	0.803±0.022	0.935±0.009	0.798±0.022	0.809±0.013	0.734±0.027	0.804±0.028
	$\text{LRR}_{\mathbf{best}}$	0.761±0.003	0.829±0.006	0.720±0.020	0.734±0.006	0.717±0.003	0.751±0.002
MVC (Graph)	DSemi-NMF	0.839±0.009	0.843±0.009	**0.951±0.001**	0.852±0.008	0.786±0.015	0.843±0.001
	MLAN	0.862±0.011	0.961±0.004	0.835±0.006	0.844±0.013	0.758±0.008	0.953±0.007
	GMC	0.791±0.001	0.941±0.000	0.782±0.000	0.794±0.000	0.694±0.000	0.929±0.000
	MCGC	0.931±0.000	0.955±0.000	0.939±0.000	0.942±0.000	**0.969±0.000**	0.916±0.000
	LMVSC	0.664±0.000	0.765±0.000	0.539±0.000	0.565±0.000	0.500±0.000	0.649±0.000
MVC (LRR)	DiMSC	0.778±0.022	0.846±0.002	0.732±0.005	0.745±0.005	0.739±0.007	0.751±0.003
	LT-MSC	0.804±0.011	0.860±0.002	0.748±0.004	0.760±0.007	0.741±0.009	0.776±0.006
	MVCC	0.732±0.018	0.845±0.007	0.675±0.022	0.692±0.021	0.647±0.034	0.744±0.013
	ECMSC	0.782±0.001	0.942±0.001	0.781±0.001	0.794±0.001	0.695±0.002	0.925±0.001
	t-SVD-MSC	0.830±0.000	0.884±0.005	0.786±0.003	0.800±0.004	0.785±0.007	0.808±0.001
	LMSC	0.749±0.018	0.866±0.006	0.699±0.025	0.715±0.023	0.655±0.041	0.790±0.017
	GLTA	0.878±0.008	0.945±0.001	0.869±0.007	0.875±0.007	0.856±0.013	0.895±0.001
	SM^2SC	0.859±0.034	0.955±0.013	0.844±0.046	0.852±0.044	0.784±0.064	0.936±0.020
	SCMV-3DT	0.701±0.028	0.810±0.009	0.635±0.003	0.654±0.029	0.614±0.039	0.702±0.018
MVC (RPCA)	RMSC	0.685±0.045	0.800±0.017	0.637±0.044	0.656±0.042	0.620±0.057	0.698±0.026
	ETLMSC	0.877±0.065	0.947±0.024	0.862±0.057	0.869±0.054	0.830±0.065	0.914±0.045
	EMVC	0.750±0.028	0.840±0.012	0.711±0.024	0.726±0.023	0.707±0.028	0.745±0.023
Ours	$\text{MCA}^2\text{M-}\mathcal{Z}$	0.927±0.047	0.974±0.012	0.924±0.040	0.928±0.038	0.897±0.057	0.962±0.018
	$\text{MCA}^2\text{M-}S$	**0.943±0.051**	**0.980±0.013**	0.942±0.045	**0.945±0.043**	0.919±0.066	**0.974±0.018**

1.2% ACC values over $\text{SPC}_{\mathbf{best}}$ on the COIL-20 database. The reason is that most SVC methods explore only the pairwise correlation or latent representation from the matrix aspect. This means that high-order correlations among multiple views such as the view-specific information cannot be well exploited. This observation is consistent with the conclusion in [2] and [7]. However, there exists a performance fluctuation of some multiview clustering methods. For instance, at least 4 multiview clustering methods yield inferior performance than SVC methods on BBCSport, Flowers, Still-DB, and MITIndoor databases. ***LRR-based Multiview Clustering vs RPCA-based Multiview Clustering:*** In general, these methods based on tensor optimization may have superior clustering performance than ones based on matrix optimization. Compared with ETLMSC, our MCA^2M yields better results on most databases. This demonstrates that the one-step strategy can improve the quality of the transition probability matrix. In addition, $\text{MCA}^2\text{M-}S$ performs better than $\text{MCA}^2\text{M-}\mathcal{Z}$.

TABLE IX
CLUSTERING RESULTS (MEAN ± STANDARD DEVIATION) ON FLOWERS DATABASE ($\alpha = 0.003,\ \beta = 0.009$)

Type	Method	ACC↑	NMI↑	AR↑	F-score↑	Precision↑	Recall↑
SVC	$\text{SPC}_{\mathbf{best}}$	0.392±0.007	0.416±0.005	0.236±0.002	0.281±0.002	0.276±0.002	0.287±0.002
	$\text{SSC}_{\mathbf{best}}$	0.398±0.004	0.439±0.003	0.240±0.004	0.285±0.004	0.280±0.004	0.290±0.004
	$\text{LRR}_{\mathbf{best}}$	0.396±0.009	0.419±0.006	0.239±0.003	0.284±0.003	0.279±0.004	0.290±0.003
MVC (Graph)	DSemi-NMF	0.240±0.011	0.223±0.006	0.080±0.005	0.139±0.004	0.126±0.004	0.155±0.005
	MLAN	0.501±0.008	0.532±0.003	0.331±0.010	0.373±0.009	0.345±0.010	0.404±0.006
	GMC	0.177±0.000	0.247±0.000	0.020±0.000	0.125±0.000	0.068±0.000	0.822±0.000
	MCGC	0.363±0.000	0.329±0.000	0.176±0.000	0.227±0.000	0.215±0.000	0.240±0.000
	LMVSC	0.360±0.000	0.385±0.000	0.198±0.000	0.246±0.000	0.241±0.000	0.250±0.000
MVC (LRR)	DiMSC	0.434±0.014	0.442±0.011	0.266±0.009	0.310±0.008	0.302±0.007	0.318±0.010
	LT-MSC	0.476±0.012	0.478±0.008	0.313±0.009	0.354±0.008	0.347±0.009	0.361±0.008
	MVCC	0.236±0.003	0.256±0.007	0.094±0.004	0.148±0.004	0.144±0.004	0.153±0.005
	ECMSC	0.446±0.002	0.423±0.001	0.260±0.001	0.304±0.001	0.298±0.001	0.311±0.001
	t-SVD-MSC	0.836±0.005	0.852±0.002	0.766±0.002	0.780±0.002	0.772±0.002	0.789±0.002
	LMSC	0.442±0.009	0.444±0.009	0.275±0.007	0.318±0.012	0.312±0.011	0.325±0.011
	GLTA	0.524±0.018	0.530±0.011	0.369±0.014	0.407±0.013	0.395±0.013	0.419±0.013
	SM^2SC	0.442±0.008	0.453±0.005	0.276±0.007	0.319±0.007	0.313±0.007	0.325±0.006
MVC (RPCA)	RMSC	0.385±0.016	0.396±0.014	0.231±0.019	0.249±0.011	0.234±0.012	0.256±0.010
	ETLMSC	0.811±0.066	0.874±0.025	0.763±0.057	0.778±0.054	0.748±0.064	0.810±0.041
	EMVC	0.433±0.011	0.440±0.009	0.257±0.009	0.301±0.009	0.295±0.008	0.307±0.011
Ours	$\text{MCA}^2\text{M-}\mathcal{Z}$	0.846±0.056	0.909±0.021	0.817±0.051	0.828±0.048	0.796±0.060	0.864±0.035
	$\text{MCA}^2\text{M-}S$	**0.857±0.037**	**0.915±0.019**	**0.826±0.041**	**0.837±0.039**	**0.806±0.045**	**0.870±0.033**

TABLE X
CLUSTERING RESULTS (MEAN ± STANDARD DEVIATION) ON MITINDOOR DATABASE ($\alpha = 0.004,\ \beta = 0.001$)

Type	Method	ACC↑	NMI↑	AR↑	F-score↑	Precision↑	Recall↑
SVC	$\text{SPC}_{\mathbf{best}}$	0.443±0.011	0.559±0.009	0.304±0.011	0.315±0.013	0.294±0.010	0.340±0.014
	$\text{SSC}_{\mathbf{best}}$	0.472±0.006	0.615±0.003	0.332±0.005	0.343±0.005	0.314±0.005	0.378±0.009
	$\text{LRR}_{\mathbf{best}}$	0.495±0.008	0.612±0.004	0.345±0.005	0.355±0.005	0.345±0.005	0.366±0.006
MVC (Graph)	DSemi-NMF	0.101±0.002	0.201±0.001	0.020±0.001	0.035±0.001	0.034±0.001	0.038±0.001
	MLAN	0.492±0.010	0.622±0.003	0.328±0.007	0.339±0.007	0.316±0.008	0.366±0.007
	GMC	0.100±0.000	0.206±0.000	0.003±0.000	0.032±0.000	0.016±0.000	0.834±0.000
	MCGC	0.254±0.000	0.340±0.000	0.040±0.000	0.064±0.000	0.038±0.000	0.208±0.000
	LMVSC	0.371±0.000	0.522±0.000	0.112±0.000	0.132±0.000	0.087±0.000	0.278±0.000
MVC (LRR)	LT-MSC	0.446±0.011	0.564±0.004	0.293±0.006	0.304±0.006	0.289±0.006	0.322±0.007
	MVCC	0.119±0.003	0.262±0.002	0.036±0.001	0.051±0.001	0.047±0.001	0.056±0.002
	ECMSC	0.387±0.008	0.508±0.007	0.239±0.001	0.251±0.009	0.237±0.006	0.267±0.005
	t-SVD-MSC	0.870±0.018	0.905±0.005	0.810±0.016	0.813±0.016	0.793±0.022	0.833±0.010
	LMSC	0.414±0.013	0.532±0.005	0.257±0.008	0.268±0.008	0.258±0.008	0.279±0.007
	SM^2SC	0.477±0.009	0.583±0.006	0.321±0.006	0.332±0.006	0.319±0.006	0.346±0.006
MVC (RPCA)	RMSC	0.289±0.011	0.445±0.007	0.163±0.006	0.176±0.006	0.171±0.006	0.181±0.006
	ETLMSC	0.864±0.032	0.950±0.009	0.852±0.030	0.854±0.030	0.824±0.035	0.888±0.023
Ours	$\text{MCA}^2\text{M-}\mathcal{Z}$	0.862±0.023	0.951±0.007	0.853±0.022	0.856±0.022	0.813±0.030	0.903±0.014
	$\text{MCA}^2\text{M-}S$	**0.876±0.034**	**0.955±0.009**	**0.867±0.031**	**0.869±0.031**	**0.832±0.040**	**0.910±0.020**

DiMSC and GLTA run out of memory in the current platform.

This is because the nonnegativity and symmetry of S can be guaranteed in (2).

2) Parameter Selection: The standard deviation σ^2 of Gaussian kernels for each view is set to the median of the pairwise Euclidean distances. There are two regularized parameters α and β affecting the performance of the proposed MCA^2M. To investigate the influence of α and β, we select α and β from $[0.0005, 0.01]$ and $[0.001, 0.01]$, respectively. We report the ACC values with respect to different combinations of α and β on 100leaves, COIL-20, Flowers, and Still-DB databases in Fig. 3. We can conclude that the proposed MCA^2M can achieve promising ACC values by selecting $\alpha \in [0.0005, 0.01]$ and $\beta \in [0.001, 0.01]$.characterize Another parameter is σ which aims to characterize the correlations among data points. Thus, we conducted new experiments with different σ and reported results in Fig. 4. One can see that when we select the best σ, the performance of MCA^2M may be further improved. For simplicity, we set σ as the average Euclidean distance of each view feature.

3) Convergence Analysis: It is intractable to guarantee the theoretical convergence of the proposed MCA^2M. This is because there are three variables in (2), i.e., $\mathcal{Z}, E, S$. In addition, the objective function of (2) is nonconvex due to the absolute operation. To the best of our knowledge, the convergence theory is still an open problem. Fig. 5 reports the relative error of the stopping criterion defined in (22), ACC,

TABLE XI

CLUSTERING RESULTS (MEAN ± STANDARD DEVIATION) ON CALTECH101 DATABASE ($\alpha = 0.005,\ \beta = 0.002$)

Type	Method	ACC↑	NMI↑	AR↑	F-score↑	Precision↑	Recall↑
SVC	SPC_{best}	0.484±0.019	0.723±0.032	0.319±0.014	0.340±0.025	0.597±0.018	0.235±0.020
	SSC_{best}	0.420±0.015	0.723±0.005	0.303±0.011	0.317±0.012	0.441±0.025	0.248±0.010
	LRR_{best}	0.510±0.009	0.728±0.014	0.304±0.017	0.339±0.008	0.627±0.012	0.231±0.010
MVC (Graph)	DSemi-NMF	0.218±0.003	0.404±0.003	0.111±0.004	0.126±0.004	0.206±0.009	0.091±0.002
	MLAN	0.579±0.024	0.748±0.020	0.222±0.015	0.265±0.015	0.173±0.009	**0.560±0.016**
	GMC	0.331±0.000	0.544±0.000	0.031±0.000	0.081±0.000	0.044±0.000	0.470±0.000
	MCGC	0.393±0.000	0.612±0.000	0.221±0.000	0.238±0.000	0.315±0.000	0.191±0.000
	LMVSC	0.513±0.000	0.782±0.000	0.338±0.000	0.352±0.000	0.464±0.000	0.284±0.000
MVC (LRR)	LT-MSC	0.559±0.012	0.788±0.005	0.393±0.007	0.403±0.003	0.670±0.009	0.288±0.012
	ECMSC	0.359±0.004	0.606±0.001	0.273±0.003	0.286±0.006	0.433±0.020	0.214±0.030
	t-SVD-MSC	0.607±0.005	0.858±0.003	0.430±0.005	0.440±0.010	0.742±0.007	0.323±0.009
	LMSC	0.566±0.012	0.818±0.004	0.383±0.010	0.392±0.010	0.710±0.014	0.271±0.008
	SM^2SC	0.568±0.021	0.825±0.008	0.380±0.015	0.391±0.015	0.671±0.022	0.275±0.013
MVC (RPCA)	RMSC	0.346±0.036	0.573±0.047	0.246±0.031	0.258±0.027	0.457±0.033	0.182±0.031
	ETLMSC	**0.639±0.019**	**0.899±0.007**	**0.456±0.017**	**0.465±0.017**	**0.825±0.029**	0.324±0.012
Ours	$\text{MCA}^2\text{M-}\mathcal{Z}$	0.589±0.008	0,875±0.005	0.428±0.007	0.437±0.007	0.771±0.013	0.305±0.006
	$\text{MCA}^2\text{M-}S$	0.606±0.022	0.884±0.006	0.439±0.019	0.449±0.019	0.782±0.034	0.315±0.013

DiMSC, MVCC, GLTA and EMVC run out of memory in the current platform.

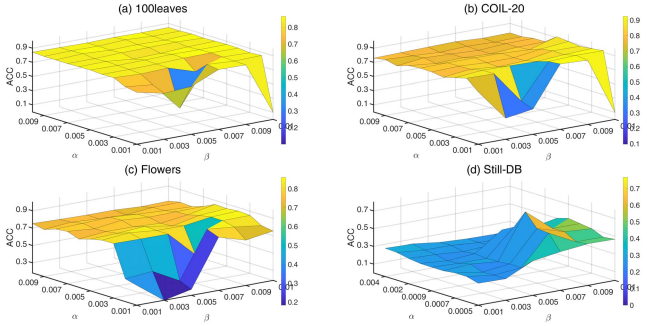

Fig. 3. Parameter tuning with respect to α and β on (a) 100leaves, (b) COIL-20, (c) Flowers, and (d) Still-DB databases.

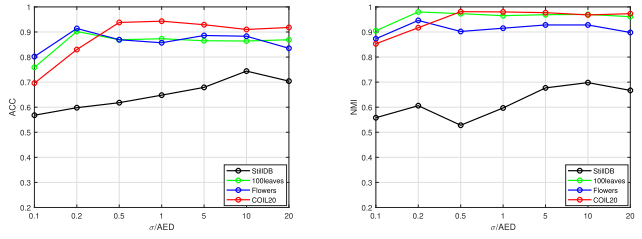

Fig. 4. Influence of σ for Gaussian kernel (ACC and NMI) on four databases.

and NMI values of MCA^2M versus iterations on 100leaves, COIL-20, Flowers, and Still-DB databases. One can see that with the iteration increasing, the relative errors of MCA^2M decrease and MCA^2M converges steadily within 25 iterations. We also observe that at the first several iterations, ACC and NMI values of MCA^2M are much lower and after 10 iterations, MCA^2M achieves promising ACC and NMI values. This also demonstrates the good empirical convergence of the proposed MCA^2M.

4) Complexity and Running Time: It is tractable to update variables $\mathcal{Z}$ and S. The main computation bottleneck of the proposed MCA^2M is to update variables $\mathcal{H}$ and E. Specifically, the fast Fourier transformation (FFT), inverse FFT, and singular value decomposition are performed for updating $\mathcal{H}$,

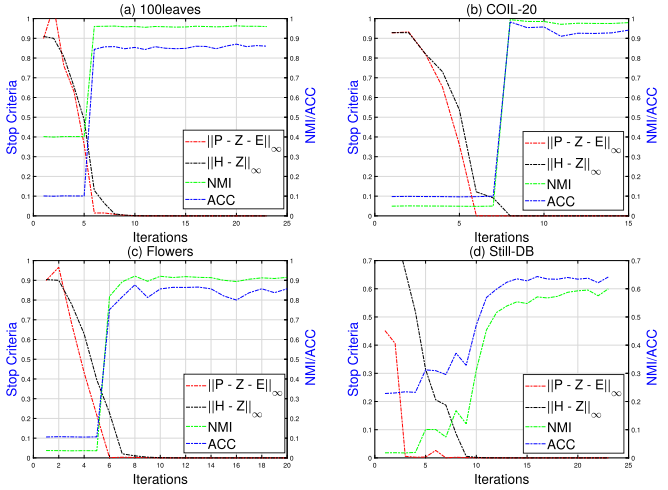

Fig. 5. Relative errors, ACC, and NMI versus iterations on (a) 100leaves, (b) COIL-20, (c) Flowers, and (d) Still-DB databases.

and thus, it costs $\mathcal{O}(Mn^2\log(n) + M^2n^2)$. When updating E, the computation cost is $\mathcal{O}(Mn^2)$. Therefore, the whole complexity of MCA^2M is $\mathcal{O}(Mn^2\log(n) + M^2n^2)$ which is the same as that of ETLMSC [1]. Table XII displays the average running times and complexities of eight representative multiview clustering methods on all databases. On the MITIndoor database, DiMSC and GLTA run out of memory in the current platform since they both need to solve the Sylvester equation and perform matrix inversion. Thus, we do not report their running times on the MITIndoor database in Table XII. Generally speaking, ETLMSC costs the shortest running time and our MCA^2M performs the second-best but achieves better performance than all other methods. As aforementioned, the LRR-based multiview clustering methods, such as LT-MSC, t-SVD-MSC, LMSC, and GLTA are highly time consuming. This is mainly because they are inevitable to perform the matrix inversion with $\mathcal{O}(n^3)$ cost.

TABLE XII
AVERAGE RUNNING TIME (IN SECONDS) ON ALL DATABASES

Method	DiMSC	LT-MSC	ECMSC	t-SVD-MSC	LMSC	GLTA	ETLMSC	EMVC	MCA^{2}M
Complexity	$\mathcal{O}(KMn^3)$	$\mathcal{O}(KMn^3)$	$\mathcal{O}((K+M)n^3)$	$\mathcal{O}(Mn^3+KMn^2\log(n))$	$\mathcal{O}(KMn^3)$	$\mathcal{O}(Mn^3+KMn^2\log(n))$	$(KMn^2(\log(n)+M))$	$\mathcal{O}(KMn^3)$	$\mathcal{O}(KMn^2(\log(n)+M))$
Still-DB	6.23	18.68	8.82	6.12	12.30	17.62	2.29	94.30	4.95
BBCSport	38.15	77.23	266.86	19.59	59.28	54.63	4.21	99.26	4.80
UCI-Digit	296.66	725.5	305.59	158.26	634.3	598.68	58.58	5213.52	92.01
COIL-20	617.29	874.91	1305.32	169.10	321.05	324.14	35.59	3129.95	36.40
Leaves	152.36	349.56	129.64	95.24	471.36	300.19	41.62	6210.93	55.56
Flowers	110.21	322.19	193.66	118.32	330.22	218.06	26.89	4253.12	35.46
MITIndoor	-	117720.7	10642.25	3404.86	11581.6	-	1245.12	-	1935.7

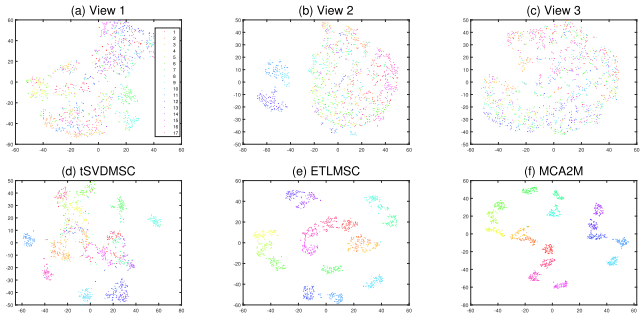

Fig. 6. Visualization of the embedding results on Flowers database (a) view 1, (b) view 2, (c) view 3, (d) t-SVD-MSC, (e) ETLMSC, and (f) MCA^{2}M.

5) Visualization: In this section, we aim to investigate the quality of the learned graph by t-SVD-MSC, ETLMSC, our MCA^{2}M, and multiview features. Following [40], we perform the Laplacian embedding method on the transition probability matrix S learned by our MCA^{2}M to yield the leading l eigenvectors $U \in \mathbb{R}^{n\times l}$, where n and l denote the numbers of samples and clusters, respectively. Then, we show 2-D visualizations of all features and U by using the t-distributed stochastic neighbor embedding (t-SNE) algorithm [58] in Fig. 6. We can see that compared with Fig. 6(a)–(e), which are obtained by performing t-SNE on each view, t-SVD-MSC and ETLMSC, it is easier to separate all data points by MCA^{2}M. This observation is also consistent with the clustering results, as shown in Tables IV–XI.

VI. CONCLUSION

In this article, we put forward a novel multiview spectral clustering method by learning an adaptive transition probability matrix (MCA^{2}M). Based on the Markov chain theory, the multiview features were used to construct multiview transition probability matrices. These matrices were adopted to yield a transition probability tensor to explore the high-order correlations of multiple features. Unlike most existing methods which learned the final affinity matrix in two separate steps, MCA^{2}M learns an adaptive transition probability matrix directly to better uncover the true relationship between data points. To evaluate the effectiveness and efficiency of the proposed MCA^{2}M, eight real-world databases with different applications were used and the experimental results have demonstrated the superiority and efficiency of MCA^{2}M over many state-of-the-art clustering approaches. In the future, we will extend the proposed method for multiview learning and classification tasks.

REFERENCES

[1] J. Wu, Z. Lin, and H. Zha, "Essential tensor learning for multi-view spectral clustering," *IEEE Trans. Image Process.*, vol. 28, no. 12, pp. 5910–5922, Dec. 2019.
[2] M. Yin, J. Gao, S. Xie, and Y. Guo, "Multiview subspace clustering via tensorial T-product representation," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 3, pp. 851–864, Mar. 2019.
[3] G. Gao, J. Yang, X.-Y. Jing, F. Shen, W. Yang, and D. Yue, "Learning robust and discriminative low-rank representations for face recognition with occlusion," *Pattern Recognit.*, vol. 66, pp. 129–143, Jun. 2017.
[4] Z. Zhang, Y. Xie, W. Zhang, Y. Tang, and Q. Tian, "Tensor multi-task learning for person re-identification," *IEEE Trans. Image Process.*, vol. 29, pp. 2463–2477, 2020.
[5] Y. Chen, S. Wang, F. Zheng, and Y. Cen, "Graph-regularized least squares regression for multi-view subspace clustering," *Knowl.-Based Syst.*, vol. 194, Apr. 2020, Art. no. 105482.
[6] Y. Chen, X. Xiao, C. Peng, G. Lu, and Y. Zhou, "Low-rank tensor graph learning for multi-view subspace clustering," *IEEE Trans. Circuits Syst. Video Technol.*, early access, Feb. 3, 2021, doi: 10.1109/TCSVT.2021.3055625.
[7] Y. Xie, D. Tao, W. Zhang, Y. Liu, L. Zhang, and Y. Qu, "On unifying multi-view self-representations for clustering by tensor multi-rank minimization," *Int. J. Comput. Vis.*, vol. 126, no. 11, pp. 1157–1179, Nov. 2018.
[8] C. Zhang, H. Fu, J. Wang, W. Li, X. Cao, and Q. Hu, "Tensorized multi-view subspace representation learning," *Int. J. Comput. Vis.*, vol. 128, pp. 1–18, Sep. 2020.
[9] 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)*, Dec. 2015, pp. 1582–1590.
[10] Y. Chen, X. Xiao, and Y. Zhou, "Jointly learning kernel representation tensor and affinity matrix for multi-view clustering," *IEEE Trans. Multimedia*, vol. 22, no. 8, pp. 1985–1997, Aug. 2020.
[11] F. Nie, J. Li, and X. Li, "Self-weighted multiview clustering with multiple graphs," in *Proc. 26th Int. Joint Conf. Artif. Intell.*, Aug. 2017, pp. 2564–2570.
[12] H. Tao, C. Hou, J. Zhu, and D. Yi, "Multi-view clustering with adaptively learned graph," in *Proc. Asian Conf. Mach. Learn.*, 2017, pp. 113–128.
[13] K. Zhan, C. Zhang, J. Guan, and J. Wang, "Graph learning for multiview clustering," *IEEE Trans. Cybern.*, vol. 48, no. 10, pp. 2887–2895, Oct. 2018.
[14] 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.
[15] Y. Chen, X. Xiao, and Y. Zhou, "Multi-view subspace clustering via simultaneously learning the representation tensor and affinity matrix," *Pattern Recognit.*, vol. 106, Oct. 2020, Art. no. 107441.
[16] 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.
[17] E. J. Candès, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?" *J. ACM*, vol. 58, no. 3, p. 11, May 2011.

[18] 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.

[19] H. Gao, F. Nie, X. Li, and H. Huang, "Multi-view subspace clustering," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 4238–4246.

[20] M. Brbi and I. Kopriva, "Multi-view low-rank sparse subspace clustering," *Pattern Recognit.*, vol. 73, pp. 247–258, Jan. 2018.

[21] C. Zhang *et al.*, "Generalized latent multi-view subspace clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 1, pp. 86–99, Jan. 2020.

[22] 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.

[23] X. Fang *et al.*, "Flexible affinity matrix learning for unsupervised and semisupervised classification," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 4, pp. 1133–1149, Apr. 2019.

[24] Y. Chen, X. Xiao, and Y. Zhou, "Multi-view clustering via simultaneously learning graph regularized low-rank tensor representation and affinity matrix," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Jul. 2019, pp. 1348–1353.

[25] X. Peng, L. Zhang, and Z. Yi, "Scalable sparse subspace clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 430–437.

[26] S. Matsushima and M. Brbic, "Selective sampling-based scalable sparse subspace clustering," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2019, pp. 12416–12425.

[27] J. Shen, P. Li, and H. Xu, "Online low-rank subspace clustering by basis dictionary pursuit," in *Proc. Int. Conf. Mach. Learn.*, 2016, pp. 622–631.

[28] Z. Kang, W. Zhou, Z. Zhao, J. Shao, M. Han, and Z. Xu, "Large-scale multi-view subspace clustering in linear time," in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 4412–4419.

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

[30] M. Najafi, L. He, and P. S. Yu, "Error-robust multi-view clustering," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Dec. 2017, pp. 736–745.

[31] Y. Zhang, W. Yang, B. Liu, G. Ke, Y. Pan, and J. Yin, "Multi-view spectral clustering via tensor-SVD decomposition," in *Proc. IEEE 29th Int. Conf. Tools with Artif. Intell. (ICTAI)*, Nov. 2017, pp. 493–497.

[32] X. Guo, "Robust subspace segmentation by simultaneously learning data representations and their affinity matrix," in *Proc. Joint Conf. Artif. Intell.*, 2015, pp. 3547–3553.

[33] M. Yin, S. Xie, Z. Wu, Y. Zhang, and J. Gao, "Subspace clustering via learning an adaptive low-rank graph," *IEEE Trans. Image Process.*, vol. 27, no. 8, pp. 3716–3728, Aug. 2018.

[34] C. Lu, J. Feng, Z. Lin, T. Mei, and S. Yan, "Subspace clustering by block diagonal representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 2, pp. 487–501, Feb. 2019.

[35] J. Xu *et al.*, "Scaled simplex representation for subspace clustering," *IEEE Trans. Cybern.*, early access, Oct. 16, 2019, doi: 10.1109/TCYB.2019.2943691.

[36] Z. Kang, H. Pan, S. C. H. Hoi, and Z. Xu, "Robust graph learning from noisy data," *IEEE Trans. Cybern.*, vol. 50, no. 5, pp. 1833–1843, May 2020.

[37] 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)*, Jul. 2017, pp. 4279–4287.

[38] Y. Fu, J. Gao, D. Tien, Z. Lin, and X. Hong, "Tensor LRR and sparse coding-based subspace clustering," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 10, pp. 2120–2133, Apr. 2016.

[39] Y. Xie, W. Zhang, Y. Qu, L. Dai, and D. Tao, "Hyper-laplacian regularized multilinear multiview self-representations for clustering and semi-supervised learning," *IEEE Trans. Cybern.*, vol. 50, no. 2, pp. 572–586, Feb. 2020.

[40] Z. Yang, Q. Xu, W. Zhang, X. Cao, and Q. Huang, "Split multiplicative multi-view subspace clustering," *IEEE Trans. Image Process.*, vol. 28, no. 10, pp. 5147–5160, Oct. 2019.

[41] Q. Gao, W. Xia, Z. Wan, D. Xie, and P. Zhang, "Tensor-svd based graph learning for multi-view subspace clustering," in *Proc. AAAI Conf. Artif. Intell.*, 2020, vol. 34, no. 4, pp. 3930–3937.

[42] J. Wen, Y. Xu, and H. Liu, "Incomplete multiview spectral clustering with adaptive graph learning," *IEEE Trans. Cybern.*, vol. 50, no. 4, pp. 1418–1429, Apr. 2020.

[43] Q. Gao, P. Zhang, W. Xia, D. Xie, X. Gao, and D. Tao, "Enhanced tensor RPCA and its application," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, Aug. 18, 2020, doi: 10.1109/TPAMI.2020.3017672.

[44] D. Zhou, J. Huang, and B. Schölkopf, "Learning from labeled and unlabeled data on a directed graph," in *Proc. 22nd Int. Conf. Mach. Learn. (ICML)*, 2005, pp. 1036–1043.

[45] Y. Chen, X. Xiao, and Y. Zhou, "Low-rank quaternion approximation for color image processing," *IEEE Trans. Image Process.*, vol. 29, pp. 1426–1439, Sep. 2019.

[46] J. Feng, Z. Lin, H. Xu, and S. Yan, "Robust subspace segmentation with block-diagonal prior," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 3818–3825.

[47] F. Nie, X. Wang, and H. Huang, "Clustering and projected clustering with adaptive neighbors," in *Proc. 20th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2014, pp. 977–986.

[48] J. Duchi, S. Shalev-Shwartz, Y. Singer, and T. Chandra, "Efficient projections onto the l_1-ball for learning in high dimensions," in *Proc. 25th Int. Conf. Mach. Learn. (ICML)*, 2008, pp. 272–279.

[49] S. Boyd, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, 2010.

[50] N. Ikizler, R. G. Cinbis, S. Pehlivan, and P. Duygulu, "Recognizing actions from still images," in *Proc. 19th Int. Conf. Pattern Recognit.*, Dec. 2008, pp. 1–4.

[51] A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," in *Proc. Neural Inf. Process. Syst.*, 2002, pp. 849–856.

[52] 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)*, Jun. 2015, pp. 586–594.

[53] H. Wang, Y. Yang, and T. Li, "Multi-view clustering via concept factorization with local manifold regularization," in *Proc. IEEE 16th Int. Conf. Data Mining (ICDM)*, Dec. 2016, pp. 1245–1250.

[54] H. Zhao, Z. Ding, and Y. Fu, "Multi-view clustering via deep matrix factorization," in *Proc. AAAI Conf. Artif. Intell.*, 2017, pp. 2921–2927.

[55] F. Nie, G. Cai, J. Li, and X. Li, "Auto-weighted multi-view learning for image clustering and semi-supervised classification," *IEEE Trans. Image Process.*, vol. 27, no. 3, pp. 1501–1511, Mar. 2018.

[56] 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)*, Jul. 2017, pp. 923–931.

[57] H. Wang, Y. Yang, and B. Liu, "GMC: Graph-based multi-view clustering," *IEEE Trans. Knowl. Data Eng.*, vol. 32, no. 6, pp. 1116–1129, Mar. 2019.

[58] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, pp. 2579–2605, Nov. 2008.

Yongyong Chen received the B.S. and M.S. degrees from the Shandong University of Science and Technology, Qingdao, China, in 2014 and 2017, respectively, and the Ph.D. degree from the University of Macau, Zhuhai, Macau, in 2020.

He is currently an Assistant Professor with the School of Computer Science and Technology, Harbin Institute of Technology, Shenzhen, China. His research interests include image processing, data mining, and computer vision.

Xiaolin Xiao received the B.E. degree from Wuhan University, Wuhan, China, in 2013, and the Ph.D. degree from the University of Macau, Zhuhai, Macau, in 2019.

She is currently a Post-Doctoral Fellow with the School of Computer Science and Engineering, South China University of Technology, Guangzhou, China. Her research interests include superpixel segmentation, saliency detection, and color image processing and understanding.

Zhongyun Hua (Member, IEEE) received the B.S. degree from Chongqing University, Chongqing, China, in 2011, and the M.S. and Ph.D. degrees from the University of Macau, Zhuhai, Macau, in 2013 and 2016, respectively, all in software engineering.

He is currently an Associate Professor with the School of Computer Science and Technology, Harbin Institute of Technology (Shenzhen), Shenzhen, China. He has authored or coauthored over 30 technical articles at prestigious international journals and conferences. His current research interests include chaotic systems, multimedia security, and image processing.

Yicong Zhou (Senior Member, IEEE) received the B.S. degree in electrical engineering from Hunan University, Changsha, China, and the M.S. and Ph.D. degrees in electrical engineering from Tufts University, Medford, MA, USA.

In 2011, he joined the University of Macau, Zhuhai, Macau, as an Assistant Professor with the Department of Computer and Information Science, where he is currently an Associate Professor and the Director of the Vision and Image Processing Laboratory. His research interests include image processing, computer vision, machine learning, and multimedia security.

Dr. Zhou is a fellow of the Society of Photo-Optical Instrumentation Engineers (SPIE). He received the Third Price of the Macao Natural Science Award as a sole winner in 2020 and a co-recipient in 2014. He received the Best Editor Award to recognize his contribution to the *Journal of Visual Communication and Image Representation* in 2020. Since 2015, he has been the leading Co-Chair of the Technical Committee on Cognitive Computing in the IEEE Systems, Man, and Cybernetics Society. He serves as an Associate Editor for the IEEE TRANSACTIONS ON NEUTRAL NETWORKS AND LEARNING SYSTEMS, the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, the IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING, and four other journals. He was recognized as the "Highly Cited Researcher" in Web of Science in 2020.