Essential Tensor Learning for Multi-view Spectral Clustering

Jianlong Wu, Zhouchen Lin, *Fellow, IEEE,* and Hongbin Zha, *Member, IEEE*

***Abstract*—Multi-view clustering attracts much attention recently, which aims to take advantage of multi-view information to improve the performance of clustering. However, most recent work mainly focus on self-representation based subspace clustering, which is of high computation complexity. In this paper, we focus on the Markov chain based spectral clustering method and propose a novel essential tensor learning method to explore the high order correlations for multi-view representation. We first construct a tensor based on multi-view transition probability matrices of the Markov chain. By incorporating the idea from robust principle component analysis, tensor singular value decomposition (t-SVD) based tensor nuclear norm is imposed to preserve the low-rank property of the essential tensor, which can well capture the principle information from multiple views. We also employ the tensor rotation operator for this task to better investigate the relationship among views as well as reduce the computation complexity. The proposed method can be efficiently optimized by the alternating direction method of multipliers (ADMM). Extensive experiments on seven real world datasets corresponding to five different applications show that our method achieves superior performance over other state-of-the-art methods.**

***Index Terms*—Multi-view spectral clustering, essential tensor learning, tensor SVD**

I. Introduction

CLUSTERING is one of the fundamental tasks in computer vision and pattern recognition, which aims to divide samples into various groups based on their similarity without any prior information. It is very useful, especially when the label information is hard to acquire. There are many clustering based applications, such as image segmentation, dimension reduction, unsupervised classification, etc. During the past decades, a variety of methods [1]–[9] for clustering have been proposed. Among them, the standard spectral clustering (SPC) [1], sparse subspace clustering (SSC) [3], and low-rank representation (LRR) [4] are the most popular methods.

These single view clustering methods achieve good performance. In practice, we often acquire data from various domains or feature space. For example, one object can be described with text, images or videos, and different kinds of features can be extracted to represent each of them. In order to make full use of multi-view information to boost the performance, many multi-view clustering methods have been derived from these popular single view methods.

Due to the popularity of SSC [3] and LRR [4], many self-representation based subspace learning methods [5]–[8], [10] are proposed for multi-view clustering. They achieve promising performances. But they mainly focus on subspace learning and have high computation complexity. Another important issue is that they mainly investigate the correlations from the aspect of pairwise matrices, and it is more natural and effective to find comprehensive representation of multi-view from the tensor aspect. Motivated by the robust multi-view spectral clustering (RMSC) [7], there is a connection between the spectral clustering and Markov chain. So we mainly focus on the spectral clustering via Markov chain in this paper. However, RMSC [7] only learns the shared common information among all views. While multi-view representations also contain view-specific information, we hope to explore the high order correlation and find the principle components [11]–[16] of multi-view representations from the tensor aspect based on the Markov chain clustering.

As for tensor decomposition, we not only need to define the rank, but also find a tight convex relaxation of the tensor rank as nuclear norm. The CANDECOMP/PARAFAC (CP) [17], [18], Tucker [19] and tensor Singular Value Decomposition (t-SVD) [20] are three main tensor decomposition techniques. However, CP rank is generally NP-hard to compute and its convex relaxation is intractable. For Tucker decomposition, the commonly used Sum of Nuclear Norms (SNN) [21] is not a tight convex relaxation of the Tucker rank. Since t-SVD based tensor nuclear norm has been proven to be the tightest convex relaxation [22] to ℓ_1-norm of the tensor multi-rank, so we adopt it. With the t-SVD based tensor nuclear norm, our model can well capture both the consistent and view-specific information among multiple views, which will benefit the clustering.

In Fig. 1, we present the framework of our proposed method. We first construct a similarity matrix and a corresponding transition probability matrix for features of each view. Then, we propose to collect these transition probability matrices of multi-view into a 3-order tensor. In order to better investigate the correlations as well as reduce the computation complexity, we rotate the tensor. The essential tensor can be learnt via tensor low-rank and sparse decomposition based on tensor nuclear norm minimization defined by the t-SVD.

Main contributions are summarized as follows:

Manuscript received July 8, 2018; revised December 14, 2018; accepted May 2, 2019. The work of Z. Lin was supported by 973 Program of China (grant no. 2015CB352502), NSF of China (grant nos. 61625301 and 61731018), Qualcomm, and Microsoft Research Asia. The work of H. Zha was supported by the National Key Research and Development Program of China (grant no. 2017YFB1002601) and National Natural Science Foundation of China (grant nos. 61632003 and 61771026). The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Senching Samson Cheung. *(Corresponding author: Zhouchen Lin.)*

J. Wu, Z. Lin, and H. Zha are with the Key Laboratory of Machine Perception (MOE), School of Electronics Engineering and Computer Science, Peking University, Beijing 100871, China (e-mail: {jlwu1992, zlin}@pku.edu.cn; zha@cis.pku.edu.cn).

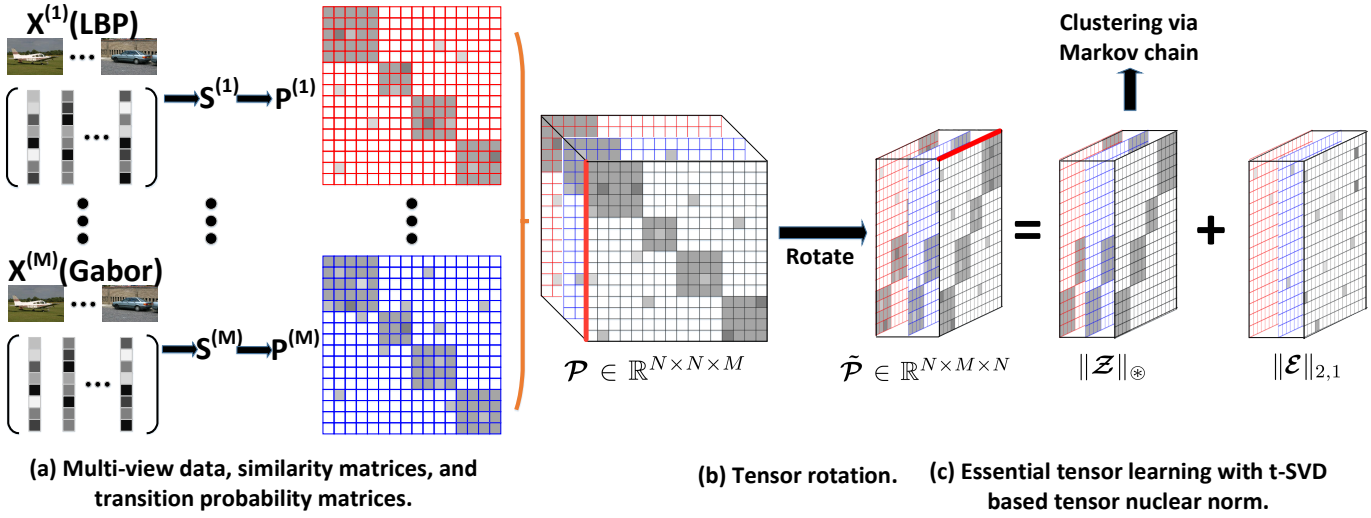


Fig. 1. The pipeline of our proposed essential tensor learning for multi-view spectral clustering. For multi-view data $\mathbf{X}^{(i)}(i = 1, \cdots, M)$, we first compute the view-specific similarity matrix $\mathbf{S}^i \in \mathbb{R}^{N\times N}$ and the corresponding transition probability matrix $\mathbf{P}^{(i)}$ by $\mathbf{P}^i = (\mathbf{D}^{(i)})^{-1}\mathbf{S}^{(i)} \in \mathbb{R}^{N\times N}$, where N is the total number of samples. Then we construct a transition probability matrix tensor $\mathcal{P}$ based on multi-view transition probability matrices. To better explore the high order correlations, we rotate the tensor $\mathcal{P}$ to $\tilde{\mathcal{P}}$ (please pay attention to the rotation of red edge). Under the assumption of low-rank and sparse, we learn the essential tensor $\mathcal{Z}$ based on t-SVD based tensor nuclear norm minimization. The learned low-rank tensor $\mathcal{Z}$ will be used as input to the standard Markov chain method for spectral clustering.

1) We propose a novel essential tensor learning method for the Markov chain based spectral clustering. With the t-SVD based tensor low-rank constraint and tensor rotation, our method is very effective to learn the principle information for clustering among multiple views.
2) We present an efficient algorithm based on ADMM to solve the proposed problem.
3) Our method achieves superior performance compared with the state-of-the-art methods on different datasets for various applications. In the meantime, it also has the lowest computation complexity.

II. Related Work

Multi-view clustering has been extensively studied during the past decade. The standard spectral clustering (SPC) [1] is the most classic method, which constructs the similarity matrix first, and then learns the affinity matrix by exploiting the properties of the Laplacian of graph. Most existing clustering methods are derived from SPC [1], and they mainly differ in the construction of affinity matrix, according to which, existing work can be mainly divided into two classes, including the graph based affinity matrix learning methods and the self-representation based subspace learning methods. We briefly review some related work.

The graph based methods learn affinity matrix based on the similarity matrix. For example, [23] proposes a co-training approach to search for the clusterings that agree across the views. [5] aims to find the complementary information across views based on a co-regularization method. [24] tries to find a universal Laplacian embedding for multi-view features using minimax optimization. The work in [25], [26] shows that there is a natural connection between the spectral clustering and the Markov random walk. Then, [27] constructs a transition probability matrix of Markov chain on each view, and then combines these matrices via a Markov mixture. Considering that multi-view data might be noisy, RMSC [7] hopes to recover a shared low-rank transition probability matrix for the Markov chain based spectral clustering. Recently, [28] proposes the structured low-rank matrix factorization methods for multi-view spectral clustering.

For the second class, multi-view subspace learning methods are derived from the popular SSC [3] and LRR [4], which aim to explore the relationships between samples based on self-representation. Most recent work of multi-view clustering mainly focus on self-representation based subspace learning. For example, [29] combines the advantages of both LRR and SSC. [30] extends the LRR into multi-view subspace clustering with generalized tensor nuclear norm. Then [31] adopts the t-SVD based tensor nuclear norm for better representation, and [32] proposes the tensorial t-product representation. Zhang et al. [33] jointly learns the underlying latent representation of features and the multi-view low-rank representation, and then generalize it to combine with deep neural network [34]. To explore the complementary property of multi-view representations, [6] utilizes the Hilbert Schmidt Independence Criterion (HSIC) as a diversity term between views, and [8] adds an exclusivity term to the structured sparse subspace clustering model [35] to preserve the complementary and consistent information.

Besides the above two classes of methods, there are also some other methods, such as the canonical correlation analysis (CCA) for multi-view clustering [36], multiple kernel learning [37], discriminative k-means [38], and so on.

III. Notations and Preliminaries

A. Notations

For convenience, we summarize the frequently used notations in Table I. In this paper, we mainly consider the 3-

TABLE I
SUMMARY OF NOTATIONS IN THIS PAPER.

a	A scalar.	$\mathbf{A}$	A matrix.
$\mathbf{a}$	A vector.	$\boldsymbol{\mathcal{A}}$	A tensor.
$\|\mathbf{A}\|_F$	$\|\mathbf{A}\|_F = \sqrt{\sum_{ij} A_{ij}^2}$.	$\|\mathbf{A}\|_*$	Sum of the singular values.
$\|\mathbf{A}\|_1$	$\|\mathbf{A}\|_1 = \sum_{ij} \|A_{ij}\|$.	$\|\mathbf{A}\|_{2,1}$	$\|\mathbf{A}\|_{2,1} = \sum_j \|A(:,j)\|_2$.
$\boldsymbol{\mathcal{A}}_{ijk}$	The (i,j,k)-th entry of $\boldsymbol{\mathcal{A}}$.	$\|\boldsymbol{\mathcal{A}}\|_F$	$\|\boldsymbol{\mathcal{A}}\|_F = \sqrt{\sum_{ijk} \|\boldsymbol{\mathcal{A}}_{ijk}\|^2}$.
$\boldsymbol{\mathcal{A}}(i,:,:)$	The i-th horizontal slice of $\boldsymbol{\mathcal{A}}$.	$\|\boldsymbol{\mathcal{A}}\|_1$	$\|\boldsymbol{\mathcal{A}}\|_1 = \sum_{ijk} \|\boldsymbol{\mathcal{A}}_{ijk}\|$.
$\boldsymbol{\mathcal{A}}(:,i,:)$	The i-th lateral slice of $\boldsymbol{\mathcal{A}}$.	$\|\boldsymbol{\mathcal{A}}\|_{2,1}$	$\|\boldsymbol{\mathcal{A}}\|_{2,1} = \sum_{i,j} \|\boldsymbol{\mathcal{A}}(i,j,:)\|_2$.
$\boldsymbol{\mathcal{A}}(:,:,i)$	The i-th frontal slice of $\boldsymbol{\mathcal{A}}$.	$\|\boldsymbol{\mathcal{A}}\|_\infty$	$\|\boldsymbol{\mathcal{A}}\|_\infty = \max_{ijk} \|\boldsymbol{\mathcal{A}}_{ijk}\|$.
$\boldsymbol{\mathcal{A}}_f$	$\boldsymbol{\mathcal{A}}_f = \text{fft}(\boldsymbol{\mathcal{A}}, [\,], 3)$.	$\|\boldsymbol{\mathcal{A}}\|_\circledast$	t-SVD based tensor nuclear norm.
$\mathbf{A}^{(i)}$	$\mathbf{A}^{(i)} = \boldsymbol{\mathcal{A}}(:,:,i)$.	$\boldsymbol{\mathcal{A}}^T$	The transpose of $\boldsymbol{\mathcal{A}}$.
$\mathbf{A}_{(i)}$	Mode-i matricization of $\boldsymbol{\mathcal{A}}$.		

order tensor $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$. Vector along the i-th mode is called the mode-i fiber. Here, we define the $\ell_{2,1}$-norm of a tensor as the sum of ℓ_2-norm of each mode-3 fiber. $\mathbf{A}_{(i)}$ denote the matricization of $\boldsymbol{\mathcal{A}}$ along the i-th mode. It can be constructed by arranging the mode-i fibers to be the columns of the resulting matrix. The transpose $\boldsymbol{\mathcal{A}}^T \in \mathbb{R}^{n_2 \times n_1 \times n_3}$ is obtained by transposing each frontal slice and then reversing the order of transposed frontal slices 2 through n_3. $\boldsymbol{\mathcal{A}}_f = \text{fft}(\boldsymbol{\mathcal{A}}, [\,], 3)$ denotes the fast Fourier transformation (FFT) of a tensor $\boldsymbol{\mathcal{A}}$ along the 3rd dimension, and we also have $\boldsymbol{\mathcal{A}} = \text{ifft}(\boldsymbol{\mathcal{A}}_f, [\,], 3)$.

Besides, for a tensor $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, we also define the block vectorizing and its inverse operation as $\text{bvec}(\boldsymbol{\mathcal{A}}) = [\mathbf{A}^{(1)}; \mathbf{A}^{(2)}; \cdots; \mathbf{A}^{(n_3)}] \in \mathbb{R}^{n_1 n_3 \times n_2}$ and $\text{fold}(\text{bvec}(\boldsymbol{\mathcal{A}})) = \boldsymbol{\mathcal{A}}$, respectively. The block diagonal matrix $\text{bdiag}(\boldsymbol{\mathcal{A}}) \in \mathbb{R}^{n_1 n_3 \times n_2 n_3}$ and the block circulant matrix $\text{bcirc}(\boldsymbol{\mathcal{A}}) \in \mathbb{R}^{n_1 n_3 \times n_2 n_3}$ are defined by:

$$\text{bdiag}(\boldsymbol{\mathcal{A}}) := \begin{bmatrix} \mathbf{A}^{(1)} & & & \\ & \mathbf{A}^{(2)} & & \\ & & \ddots & \\ & & & \mathbf{A}^{(n_3)} \end{bmatrix},$$

$$\text{bcirc}(\boldsymbol{\mathcal{A}}) := \begin{bmatrix} \mathbf{A}^{(1)} & \mathbf{A}^{(n_3)} & \cdots & \mathbf{A}^{(2)} \\ \mathbf{A}^{(2)} & \mathbf{A}^{(1)} & \cdots & \mathbf{A}^{(3)} \\ \vdots & \ddots & \ddots & \vdots \\ \mathbf{A}^{(n_3)} & \mathbf{A}^{(n_3-1)} & \cdots & \mathbf{A}^{(1)} \end{bmatrix}.$$

B. Preliminaries

To help understand the definition of tensor nuclear norm, we first introduce some related definitions [20].

*Definition 1 (**t-product**):* Let $\boldsymbol{\mathcal{A}}$ be $n_1 \times n_2 \times n_3$, and $\boldsymbol{\mathcal{B}}$ be $n_2 \times n_4 \times n_3$. Then the t-product $\boldsymbol{\mathcal{A}} * \boldsymbol{\mathcal{B}}$ is the $n_1 \times n_4 \times n_3$ tensor

$$\boldsymbol{\mathcal{A}} * \boldsymbol{\mathcal{B}} = \text{fold}(\text{bcirc}(\boldsymbol{\mathcal{A}})\text{bvec}(\boldsymbol{\mathcal{B}})). \tag{1}$$

*Definition 2 (**f-diagonal tensor**):* A tensor is called f-diagonal if each of its frontal slices is diagonal matrix.

*Definition 3 (**Identity tensor**):* For the identity tensor $\boldsymbol{\mathcal{I}} \in \mathbb{R}^{n \times n \times n_3}$, its first frontal slice is the identity matrix with size $n \times n$, and all other frontal slices are zero.

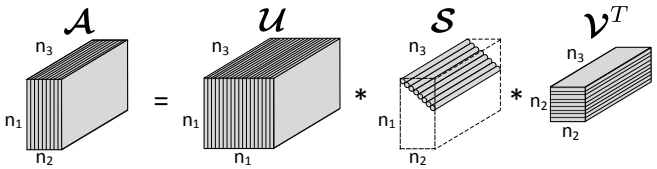


Fig. 2. Illustration of the t-SVD decomposition of an $n_1 \times n_2 \times n_3$ tensor.

*Definition 4 (**Orthogonal tensor**):* A tensor $\boldsymbol{\mathcal{Q}} \in \mathbb{R}^{n \times n \times n_3}$ is orthogonal if it satisfies

$$\boldsymbol{\mathcal{Q}}^{\mathrm{T}} * \boldsymbol{\mathcal{Q}} = \boldsymbol{\mathcal{Q}} * \boldsymbol{\mathcal{Q}}^{\mathrm{T}} = \boldsymbol{\mathcal{I}}. \tag{2}$$

*Definition 5 (**t-SVD**):* For a tensor $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, it can be factorized by t-SVD as

$$\boldsymbol{\mathcal{A}} = \boldsymbol{\mathcal{U}} * \boldsymbol{\mathcal{S}} * \boldsymbol{\mathcal{V}}^T, \tag{3}$$

where $\boldsymbol{\mathcal{U}} \in \mathbb{R}^{n_1 \times n_1 \times n_3}$ and $\boldsymbol{\mathcal{V}} \in \mathbb{R}^{n_2 \times n_2 \times n_3}$ are orthogonal, and $\boldsymbol{\mathcal{S}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ is f-diagonal.

*Definition 6 (**t-SVD based tensor nuclear norm**):* The t-SVD based tensor nuclear norm $\|\boldsymbol{\mathcal{A}}\|_\circledast$ of a tensor $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ is defined by the sum of singular values of all the frontal slices of $\boldsymbol{\mathcal{A}}_f$:

$$\|\boldsymbol{\mathcal{A}}\|_\circledast = \sum_{k=1}^{n_3} \|\boldsymbol{\mathcal{A}}_f^{(k)}\|_* = \sum_{i=1}^{\min(n_1,n_2)} \sum_{k=1}^{n_3} |\boldsymbol{\mathcal{S}}_f^{(k)}(i,i)|, \tag{4}$$

where $\boldsymbol{\mathcal{S}}_f^{(k)}$ is computed by the SVD $\boldsymbol{\mathcal{A}}_f^{(k)} = \boldsymbol{\mathcal{U}}_f^{(k)} \boldsymbol{\mathcal{S}}_f^{(k)} \boldsymbol{\mathcal{V}}_f^{(k)T}$ of frontal slices of $\boldsymbol{\mathcal{A}}_f$.

IV. Essential Tensor Learning for Multi-view Spectral Clustering

In this section, we first introduce the overview of spectral clustering by Markov chain. Then we present the details and analysis of our proposed essential tensor learning for multi-view spectral clustering (ETLMSC).

A. Markov Chain based Spectral Clustering

Denote $\mathbf{X} = [\mathbf{x}_1, \cdots, \mathbf{x}_N] \in \mathbb{R}^{d \times N}$ as the the matrix of data vectors, where N is the number of data points and d is the dimension of feature vectors. We first compute the similarity

Algorithm 1 Spectral Clustering by Markov Chain

Input: Data points $\{\mathbf{x}_1, \cdots, \mathbf{x}_N\}$.

1: Compute the similarity matrix $\mathbf{S} \in \mathbb{R}^{N\times N}$ with Gaussian kernel $S_{ij} = \exp(-\frac{\|\mathbf{x}_i-\mathbf{x}_j\|_2^2}{\sigma^2})$.
2: Construct the weighted graph $\mathbf{G} = (\mathbf{V}, \mathbf{E}, \mathbf{S})$ and define a random walk over $\mathbf{G}$ with transition probability matrix $\mathbf{P} = \mathbf{D}^{-1}\mathbf{S} \in \mathbb{R}^{N\times N}$ such that it has a unique stationary distribution π satisfying $\pi = \mathbf{P}^T\pi$.
3: Compute eigenvalues decomposition of the normalized Laplacian matrix $\mathbf{L}^{'} = (\mathbf{\Pi}^{\frac{1}{2}}\mathbf{P}\mathbf{\Pi}^{-\frac{1}{2}} + \mathbf{\Pi}^{-\frac{1}{2}}\mathbf{P}^T\mathbf{\Pi}^{\frac{1}{2}})/2$, where $\mathbf{\Pi}$ is a diagonal matrix with $\Pi_{ii} = \pi(i)$.
4: Adopt the k-means to cluster row vectors of $\mathbf{U} \in \mathbb{R}^{N\times C}$, which consists of C eigenvectors corresponding to the C largest eigenvalues of $\mathbf{L}^{'}$ in the last step, and assign each data point into the corresponding class.

Output: Assigned class of each data point.

matrix $\mathbf{S}$, where S_{ij} denotes the similarity between data points $\mathbf{x}_i$ and $\mathbf{x}_j$. Gaussian kernel is commonly used to define their similarity. We have $S_{ij} = \exp(-\frac{\|\mathbf{x}_i-\mathbf{x}_j\|_2^2}{\sigma^2})$, where the ℓ_2 distance is adopted and σ is the standard deviation. Then we can construct a weighted graph $\mathbf{G} = (\mathbf{V}, \mathbf{E}, \mathbf{S})$, where the vertices set $\mathbf{V}$ consists of the sample points, the edges set $\mathbf{E}$ denotes the connection between data points, and the similarity $\mathbf{S}$ defines the weight of each edge. For spectral clustering [1], it tries to find an optimal partition in the weighted graph $\mathbf{G}$. According to [25], [26], there is a natural connection between spectral clustering and random walkers on the weighted graph. We first define the transition probability matrix by $\mathbf{P} = \mathbf{D}^{-1}\mathbf{S}$, where P_{ij} denotes the probability of random walk from node i to node j, and $\mathbf{D}$ is a diagonal matrix with elements $D_{ii} = \sum_j S_{ij}$. For this Markov chain, we hope the random walk over the graph converges to a unique and positive stationary distribution π, that is $\pi = \mathbf{P}^T\pi$. Let $\mathbf{\Pi}$ denote the diagonal matrix with $\Pi_{ii} = \pi(i)$, then the Laplacian matrix for the Markov chain based spectral clustering can be computed by $\mathbf{L} = \mathbf{\Pi} - (\mathbf{\Pi}\mathbf{P} + \mathbf{P}^T\mathbf{\Pi})/2$. Denote C as the number of clusters, the indicator function $\mathbf{f}$ for clustering can be solved by computing the eigenvectors corresponding to the C smallest eigenvalues of the generalized eigenvalue decomposition problem $\mathbf{L}\mathbf{f} = \lambda\mathbf{\Pi}\mathbf{f}$, which is equivalent to the eigenvectors corresponding to the C largest eigenvalues of the normalized Laplacian matrix $\mathbf{L}^{'} = (\mathbf{\Pi}^{\frac{1}{2}}\mathbf{P}\mathbf{\Pi}^{-\frac{1}{2}} + \mathbf{\Pi}^{-\frac{1}{2}}\mathbf{P}^T\mathbf{\Pi}^{\frac{1}{2}})/2$. Finally, k-means algorithm [39] is adopted to cluster based on these indicator vectors. In Algorithm 1, we briefly summarize the outline for spectral clustering by Markov chains. For more details, please refer to [7], [26].

B. The Proposed Method

Assume that there are M different views in total. Let $\mathbf{X}^{(i)} = [\mathbf{x}_1^{(i)}, \cdots, \mathbf{x}_N^{(i)}] \in \mathbb{R}^{d^{(i)}\times N}$ denote the data matrix of the i-th view, where N is the number of samples, $d^{(i)}$ is the dimension of feature vectors in the i-th view, and i ranges from 1 to M. For multi-view spectral clustering via Markov chain, we first compute the similarity matrix $\mathbf{S}^{(i)} \in \mathbb{R}^{N\times N}$, construct the weighted graph $G^{(i)}$, and compute the transition probability matrix $\mathbf{P}^{(i)}$ for each view. According to Algorithm 1, we can see that the transition probability matrix $\mathbf{P}$ plays a very important role in the clustering by Markov chain. So we mainly focus on how to learn an essential transition probability matrix for spectral clustering based on the multi-view $\mathbf{P}^{(i)}, i = 1, \cdots, M$.

RMSC [7] hopes to capture the shared information among multi-view transition probability matrices. It divides each $\mathbf{P}^i$ into two parts: a shared probability matrix $\mathbf{Z}$ describing important information for clustering, and view-specific deviation error matrix $\mathbf{E}^{(i)}$. As the number of clusters is much smaller than the sample number, RMSC imposes low-rank constraint on $\mathbf{Z}$. It also assumes that the error matrix should be sparse. Then the objective function for RMSC [7] is formulated as

$$\min_{\mathbf{Z},\mathbf{E}^{(i)}} \|\mathbf{Z}\|_* + \lambda\sum_{i=1}^{M}\|\mathbf{E}^{(i)}\|_1 \; s.t. \; \mathbf{P}^{(i)} = \mathbf{Z} + \mathbf{E}^{(i)}, i = 1, \cdots, M, \tag{5}$$

where λ is a balance parameter.

RMSC only learns the shared common information among multiple views. However, each view also contains unique information that is useful for clustering. Motivated by this, we hope to explore high order correlations among multiple views based on tensor representation.

We divide each $\mathbf{P}^i$ into two parts $\mathbf{P}^{(i)} = \mathbf{Z}^{(i)} + \mathbf{E}^{(i)}$. Then we construct a 3-order tensor $\boldsymbol{\mathcal{Z}}$ by collecting all $\mathbf{Z}^{(i)}$. As multi-view features are extracted from the same objects, different $\mathbf{Z}^{(i)}$ also contains some similar information. In the meantime, the number of clusters is much smaller than the sample number. So the tensor $\boldsymbol{\mathcal{Z}}$ should be low-rank. We use the t-SVD based tensor nuclear norm $\|\cdot\|_{\circledast}$ to regularize $\boldsymbol{\mathcal{Z}}$ and get the primary objective function for our model:

$$\min_{\mathbf{Z},\mathbf{E}^{(i)}} \|\boldsymbol{\mathcal{Z}}\|_{\circledast} + \lambda\sum_{i=1}^{M}\|\mathbf{E}^{(i)}\|_1 \; s.t. \; \mathbf{P}^{(i)} = \mathbf{Z}^{(i)} + \mathbf{E}^{(i)}, i = 1, \cdots, M. \tag{6}$$

The minimization of low-rank tensor can help us find the essential information among different views. Specifically, the consistent information among multiple views may be represented by several principle components of the t-SVD, and view-specific information can be preserved in other singular values of the corresponding slice of the f-diagonal tensor $\boldsymbol{\mathcal{S}}$, which is computed by the t-SVD.. By constructing a 3-order transition probability tensor $\boldsymbol{\mathcal{P}} \in \mathbb{R}^{N\times N\times M}$, where $\mathbf{P}^{(i)}$ is the i-th frontal slice of the tensor $\boldsymbol{\mathcal{P}}$, the above problem can be reformulated as the tensor form:

$$\min_{\boldsymbol{\mathcal{Z}},\boldsymbol{\mathcal{E}}} \|\boldsymbol{\mathcal{Z}}\|_{\circledast} + \lambda\|\boldsymbol{\mathcal{E}}\|_1, \; s.t. \; \boldsymbol{\mathcal{P}} = \boldsymbol{\mathcal{Z}} + \boldsymbol{\mathcal{E}}. \tag{7}$$

Instead of optimizing the above problem, we first rotate the original transition probability tensor $\boldsymbol{\mathcal{P}} \in \mathbb{R}^{N\times N\times M}$ into $\tilde{\boldsymbol{\mathcal{P}}} \in \mathbb{R}^{N\times M\times N}$, which can be seen in the middle part of Fig. 1 (please pay attention to the rotation of the red edge of the tensor). This tensor rotation can be easily achieved by the $shiftdim$ function in Matlab. There are mainly two advantages for this operation. First, according to the definition of t-SVD, FFT operates along the third dimension of the tensor and then we perform SVD in each frontal slice. As

Algorithm 2 Essential Transition Probability Tensor Learning for Multi-view Spectral Clustering

Input: Multi-view data $\mathbf{X}^{(i)} \in \mathbb{R}^{d_i \times N},\ i = 1, 2, \cdots, M$.

1: Compute $\mathbf{S}^i$ and $\mathbf{P}^i$, and construct the rotated tensor $\tilde{\boldsymbol{\mathcal{P}}}$. Set $k = 0$, $\boldsymbol{\mathcal{L}}^0 = \boldsymbol{\mathcal{E}}^0 = \boldsymbol{\mathcal{Y}}^0 = \mathbf{0}$, $\mu^0 = 10^{-3}$, $\rho = 2$, $\mu^{max} = 10^8$, and $\epsilon = 10^{-6}$.
2: **while** not converged **do**
3: Fix $\boldsymbol{\mathcal{E}}^k$. Update $\boldsymbol{\mathcal{Z}}^{k+1}$ by Eq. 11.
4: Fix $\boldsymbol{\mathcal{Z}}^{k+1}$. Update $\boldsymbol{\mathcal{E}}^{k+1}$ by Eq. 12.
5: Update $\boldsymbol{\mathcal{Y}}^{k+1}$ by Eq. 15.
6: $\mu^{k+1} = \min(\rho\mu^k, \mu^{max})$.
7: Check the convergence conditions: $\|\boldsymbol{\mathcal{Z}}^{k+1} - \boldsymbol{\mathcal{Z}}^k\|_\infty \leq \epsilon$, $\|\boldsymbol{\mathcal{E}}^{k+1} - \boldsymbol{\mathcal{E}}^k\|_\infty \leq \epsilon$, $\|\tilde{\boldsymbol{\mathcal{P}}} - \boldsymbol{\mathcal{Z}}^{k+1} - \boldsymbol{\mathcal{E}}^{k+1}\|_\infty \leq \epsilon$.
8: $k = k + 1$.
9: **end while**

Output: $\boldsymbol{\mathcal{Z}}^{k+1}$ and $\boldsymbol{\mathcal{E}}^{k+1}$.

we hope to capture the essential information among all views, SVD in each slice with the information of multi-view and all samples is more meaningful. Moreover, FFT along the feature dimension can preserve the relationship among views. Second, this rotation can largely reduce the computation complexity in optimization, which will be analysed in the subsection IV-D.

Besides, for the error term, if one sample contains much noise and outliers, transition probability vectors in the tensor related to this sample will be influenced. Noises in these vectors are not sparse, so ℓ_2-norm regularization on vectors is more proper. As noisy samples should be sparse, tensor $\ell_{2,1}$-norm works. It is more robust to outliers and noises. So we use $\ell_{2,1}$-norm to characterize the sparsity property. Then the final objective function of our proposed ETLMSC method can be reformulated as follows:

$$\min_{\boldsymbol{\mathcal{Z}}, \boldsymbol{\mathcal{E}}} \|\boldsymbol{\mathcal{Z}}\|_\circledast + \lambda\|\boldsymbol{\mathcal{E}}\|_{2,1},\ s.t.\ \tilde{\boldsymbol{\mathcal{P}}} = \boldsymbol{\mathcal{Z}} + \boldsymbol{\mathcal{E}}, \tag{8}$$

where $\tilde{\boldsymbol{\mathcal{P}}}$ denotes the rotated transition probability tensor. For the tensor $\boldsymbol{\mathcal{E}}$ after rotation, the $\ell_{2,1}$-norm is defined as the sum of ℓ_2-norm of each fiber along the coefficient dimension. According to the definition of $\ell_{2,1}$-norm and matricization in Table I, we have $\|\boldsymbol{\mathcal{E}}\|_{2,1} = \|\mathbf{E}_{(3)}\|_{2,1}$, which is helpful to the optimization of $\boldsymbol{\mathcal{E}}$.

C. Optimization

We adopt the alternating direction method of multipliers (ADMM) [40] to solve Eq. (8). The augmented Lagrangian function can be formulated as follows:

$$\begin{aligned}\mathcal{L}(\boldsymbol{\mathcal{Z}}, \boldsymbol{\mathcal{E}}) &= \|\boldsymbol{\mathcal{Z}}\|_\circledast + \lambda\|\boldsymbol{\mathcal{E}}\|_{2,1} \\ &+ \langle \boldsymbol{\mathcal{Y}}, \tilde{\boldsymbol{\mathcal{P}}} - \boldsymbol{\mathcal{Z}} - \boldsymbol{\mathcal{E}}\rangle + \frac{\mu^k}{2}\|\tilde{\boldsymbol{\mathcal{P}}} - \boldsymbol{\mathcal{Z}} - \boldsymbol{\mathcal{E}}\|_F^2 \\ &= \|\boldsymbol{\mathcal{Z}}\|_\circledast + \lambda\|\boldsymbol{\mathcal{E}}\|_{2,1} + \frac{\mu^k}{2}\|\tilde{\boldsymbol{\mathcal{P}}} - \boldsymbol{\mathcal{Z}} - \boldsymbol{\mathcal{E}} + \boldsymbol{\mathcal{Y}}/\mu^k\|_F^2,\end{aligned} \tag{9}$$

where $\mu^k > 0$ is a penalty parameter at k-th iteration and $\boldsymbol{\mathcal{Y}}$ is a Lagrange multiplier. ADMM alternately updates each variable as follows.

$\boldsymbol{\mathcal{Z}}$ sub-problem:

$$\boldsymbol{\mathcal{Z}}^{k+1} = \arg\min_{\boldsymbol{\mathcal{Z}}} \|\boldsymbol{\mathcal{Z}}\|_\circledast + \frac{\mu^k}{2}\|\boldsymbol{\mathcal{Z}} - \left(\tilde{\boldsymbol{\mathcal{P}}} - \boldsymbol{\mathcal{E}}^k + \boldsymbol{\mathcal{Y}}^k/\mu^k\right)\|_F^2, \tag{10}$$

which is a t-SVD based tensor nuclear norm minimization problem. According to [41], it has the following close-form solution with the tensor tubal-shrinkage operator:

$$\boldsymbol{\mathcal{Z}}^{k+1} = \mathcal{C}_{\mu'}(\tilde{\boldsymbol{\mathcal{P}}} - \boldsymbol{\mathcal{E}}^k + \boldsymbol{\mathcal{Y}}^k/\mu^k) = \boldsymbol{\mathcal{U}} * \mathcal{C}_{\mu'}(\boldsymbol{\mathcal{S}}) * \boldsymbol{\mathcal{V}}^T, \tag{11}$$

where $\mu' = N \cdot \mu^k$, $\tilde{\boldsymbol{\mathcal{P}}} - \boldsymbol{\mathcal{E}}^k + \boldsymbol{\mathcal{Y}}^k/\mu^k = \boldsymbol{\mathcal{U}} * \boldsymbol{\mathcal{S}} * \boldsymbol{\mathcal{V}}^{\boldsymbol{T}}$ and $\mathcal{C}_{\mu'}(\boldsymbol{\mathcal{S}}) = \boldsymbol{\mathcal{S}} * \boldsymbol{\mathcal{J}}$. $\boldsymbol{\mathcal{J}} \in \mathbb{R}^{N \times M \times N}$ is an f-diagonal tensor whose diagonal element in the Fourier domain is $\boldsymbol{\mathcal{J}}_f(i, i, j) = \max(1 - \frac{\mu'}{\boldsymbol{\mathcal{S}}_f^{(j)}(i,i)}, 0)$.

$\boldsymbol{\mathcal{E}}$ sub-problem:

$$\boldsymbol{\mathcal{E}}^{k+1} = \arg\min_{\boldsymbol{\mathcal{E}}} \lambda\|\boldsymbol{\mathcal{E}}\|_{2,1} + \frac{\mu^k}{2}\|\boldsymbol{\mathcal{E}} - \left(\tilde{\boldsymbol{\mathcal{P}}} - \boldsymbol{\mathcal{Z}}^{k+1} + \boldsymbol{\mathcal{Y}}^k/\mu^k\right)\|_F^2. \tag{12}$$

As the $\ell_{2,1}$-norm of the tensor $\boldsymbol{\mathcal{E}}$ is defined as the sum of ℓ_2-norm of each mode-3 fiber, we matricize each tensor along the 3rd mode. So we have $\|\mathbf{E}_{(3)}^{k+1}\|_{2,1} = \|\boldsymbol{\mathcal{E}}^{k+1}\|_{2,1}$. It can be transformed into the matrix form:

$$\begin{aligned}\mathbf{E}_{(3)}^{k+1} &= \arg\min_{\mathbf{E}_{(3)}} \lambda\|\mathbf{E}_{(3)}\|_{2,1} \\ &+ \frac{\mu^k}{2}\|\mathbf{E}_{(3)} - \left(\tilde{\mathbf{P}}_{(3)} - \mathbf{Z}_{(3)}^{k+1} + \mathbf{Y}_{(3)}^k/\mu^k\right)\|_F^2.\end{aligned} \tag{13}$$

Let $\mathbf{D} = \tilde{\mathbf{P}}_{(3)} - \mathbf{Z}_{(3)}^{k+1} + \mathbf{Y}_{(3)}^k/\mu^k$, and according to [4], the problem in Eq. (13) has the following close-form solution:

$$\mathbf{E}_{(3):,i}^{k+1} = \begin{cases} \dfrac{||\mathbf{D}_{:,i}||_2 - \frac{\lambda}{\mu^k}}{||\mathbf{D}_{:,i}||_2}\mathbf{D}_{:,i}, & \text{if } ||\mathbf{D}_{:,i}||_2 > \dfrac{\lambda}{\mu^k} \\ \mathbf{0}, & \text{otherwise.} \end{cases} \tag{14}$$

where $\mathbf{D}_{:,i}$ represents the i-th column of the matrix $\mathbf{D}$. After we get $\mathbf{E}_{(3)}^{k+1}$, we transform it into the tensor form.

Update multipliers:

$$\boldsymbol{\mathcal{Y}}^{k+1} = \boldsymbol{\mathcal{Y}}^k + \mu^k\left(\tilde{\boldsymbol{\mathcal{P}}} - \boldsymbol{\mathcal{Z}}^{k+1} - \boldsymbol{\mathcal{E}}^{k+1}\right). \tag{15}$$

The whole optimization process is summarized in Algorithm 2. After we learn the essential transition probability tensor $\boldsymbol{\mathcal{Z}} \in \mathbb{R}^{N \times M \times N}$, we compute the essential transition probability matrix $\mathbf{Z}^* \in \mathbb{R}^{N \times N}$ by summing its lateral slices as $\mathbf{Z}^* = \sum_{i=1}^M \boldsymbol{\mathcal{Z}}(:, i, :)$. Then we put $\mathbf{Z}^*$ into the second step of Algorithm 1 to replace the transition probability matrix $\mathbf{P}$, and we can get the final clustering result.

D. Convergence and Complexity

At each iteration, we can get the close-form solution of $\boldsymbol{\mathcal{Z}}^{k+1}$ and $\boldsymbol{\mathcal{E}}^{k+1}$. In [40], the convergence of ADMM with two blocks of variables has already been proved. Accordingly, our algorithm will converge to an optimal solution.

For the computation complexity, at each iteration, it takes $\mathcal{O}(MN^2)$ to compute the close-form solution of $\boldsymbol{\mathcal{E}}$. As for updating $\boldsymbol{\mathcal{Z}}$, on the one hand, we need to calculate the FFT and inverse FFT of a $N \times M \times N$ tensor along the third dimension, which takes $\mathcal{O}(MN^2 \log(N))$. On the other hand,

Fig. 3. Some sample images of these image datasets for various applications. (a) The UCI-Digits dataset; (b) The COIL-20 dataset; (c) The Notting-Hill dataset; (d) The Scene-15 dataset; (e) The MITIndoor-67 dataset.

in the Fourier domain, we need to compute the SVD of each frontal slice of a tensor with size $N \times M \times N$, which takes $\mathcal{O}(M^2N^2)$. So we need $\mathcal{O}(M^2N^2 + MN^2\log(N))$ in total to compute the close-form solution of $\mathcal{Z}$ under tensor rotation operation. However, if we do not rotate the tensor, we need $\mathcal{O}(MN^3 + MN^2\log(M))$. As the number of views M is much smaller than the number of samples N in multi-view setting, that is $M \ll N$ and $M \leq \log(N)$. Therefore, we can see that the computation complexity is largely reduced by the tensor rotation. Denote K as the number of iterations, the complexity to learn the essential tensor in Algorithm 2 is $\mathcal{O}(KMN^2(M + \log(N)))$, which is relatively efficient.

After we get the essential transition probability matrix, we adopt the Markov chain based spectral clustering to get the final result, which usually cost $\mathcal{O}(N^3)$. Therefore, the overall complexity is $\mathcal{O}(N^3 + KMN^2(M + \log(N)))$.

V. Experiments

A. Experimental Settings

1) Datasets: We adopt seven commonly used real world datasets, which cover five different applications, including news article clustering, digit clustering, generic object clustering, face clustering, and scene clustering. In Table II, we summarize the statistic information of these seven datasets. Some samples of these image datasets are presented in Fig 3. We briefly introduce these datasets as follows.

TABLE II
Statistics of different datasets.

Dataset	Images	Objective	Clusters	Views
BBC-Sport	773	Text	5	2
UCI-Digits	2000	Digit	10	3
COIL-20	1440	Object	20	3
Notting-Hill	4660	Video Face	5	3
Scene-15	4485	Scene	15	3
MITIndoor-67	5360	Scene	67	4
Caltech-101	8677	Object	101	4

BBC-Sport [42] [1] contains 737 documents from the BBC Sport website corresponding to sports news in five topical areas, including the athletics, cricket, football, rugby, and tennis. There are two different views in total.

UCI-Digits [43] consists of $2,000$ digits images corresponding to 10 classes. Same to [7], we extract three different features to represent these digit images, including Fourier coefficients, pixel averages and morphological features.

COIL-20 [2] is the abbreviation of the Columbia object image library dataset, which contains $1,440$ images of 20 object categories. Each category contains 72 images and all images are normalized to size 32×32. For this datasets, we also extract

[1] http://mlg.ucd.ie/datasets
[2] http://www.cs.columbia.edu/CAVE/software/softlib/

TABLE III
EXPERIMENTAL RESULTS ON THE BBC-SPORT AND THE UCI-DIGIT DATASETS. FOR ETLMSC, WE SET $\lambda = 0.03$ AND $\lambda = 0.007$ FOR THESE TWO DATASETS, RESPECTIVELY.

Datasets	BBC-Sport						UCI-Digits					
Methods	NMI	ACC	AR	F-score	Precision	Recall	NMI	ACC	AR	F-score	Precision	Recall
SPC$_{best}$	0.735	0.853	0.744	0.798	0.804	0.792	0.642	0.731	0.545	0.591	0.582	0.601
LRR$_{best}$	0.747	0.886	0.725	0.789	0.803	0.776	0.768	0.871	0.736	0.763	0.759	0.767
Co-reg	0.771	0.849	0.783	0.829	0.836	0.822	0.804	0.780	0.755	0.780	0.764	0.798
RMSC	0.808	0.912	0.837	0.871	0.879	0.864	0.822	0.915	0.789	0.811	0.797	0.826
DiMSC	0.814	0.901	0.843	0.880	0.875	0.882	0.772	0.703	0.652	0.695	0.673	0.718
LTMSC	0.066	0.379	0.005	0.383	0.239	0.953	0.775	0.803	0.725	0.753	0.739	0.767
ECMSC	0.090	0.408	0.060	0.391	0.267	0.942	0.780	0.718	0.672	0.707	0.660	0.760
UR-ETLMSC	0.808	0.879	0.823	0.865	0.859	0.873	0.782	0.841	0.719	0.747	0.739	0.756
t-SVD-MSC	0.830	0.941	0.853	0.888	0.881	0.896	0.932	0.955	0.924	0.932	0.930	0.934
ETLMSC	**0.984**	**0.978**	**0.967**	**0.977**	**0.963**	**0.998**	**0.977**	**0.958**	**0.953**	**0.958**	**0.940**	**0.980**

TABLE IV
EXPERIMENTAL RESULTS ON THE COIL-20 AND THE NOTTING-HILL DATASETS. FOR ETLMSC, WE SET $\lambda = 0.003$ AND $\lambda = 0.0008$ FOR THESE TWO DATASETS, RESPECTIVELY.

Datasets	COIL-20						Notting-Hill					
Methods	NMI	ACC	AR	F-score	Precision	Recall	NMI	ACC	AR	F-score	Precision	Recall
SPC$_{best}$	0.806	0.672	0.619	0.640	0.596	0.692	0.723	0.816	0.712	0.775	0.780	0.776
LRR$_{best}$	0.829	0.761	0.720	0.734	0.717	0.751	0.579	0.794	0.558	0.653	0.672	0.636
Co-reg	0.774	0.659	0.592	0.613	0.590	0.640	0.703	0.805	0.686	0.754	0.766	0.743
RMSC	0.800	0.685	0.637	0.656	0.620	0.698	0.585	0.807	0.496	0.603	0.621	0.586
DiMSC	0.846	0.778	0.732	0.745	0.739	0.751	0.799	0.837	0.787	0.834	0.822	0.847
LTMSC	0.860	0.804	0.748	0.760	0.741	0.479	0.779	0.868	0.777	0.825	0.830	0.814
ECMSC	0.942	0.782	0.781	0.794	0.695	**0.925**	0.817	0.767	0.679	0.764	0.637	0.954
UR-ETLMSC	0.829	0.750	0.696	0.711	0.692	0.732	0.794	0.835	0.787	0.834	0.828	0.840
t-SVD-MSC	0.884	0.830	0.786	0.800	0.785	0.808	0.900	**0.957**	**0.900**	0.922	0.937	0.907
ETLMSC	**0.947**	**0.877**	**0.862**	**0.869**	**0.830**	0.914	**0.911**	0.951	0.898	**0.924**	**0.940**	**0.908**

three types of features (intensity, LBP [44] and Gabor [45] features), which is same to [30], [31].

Notting-Hill [46] is a video based face dataset, which is collected from the movie "Notting-Hill". It contains $4,660$ faces of 5 main casts in 76 tracks. All face images are with size 50×40. Intensity, LBP [44] and Gabor [45] features are extracted for representation.

Scene-15 [47] has 15 natural scene categories with both indoor and outdoor environments, including industrial, store, bedroom, kitchen, and etc. There are $4,485$ images in total. Similar to [31], we extract three kinds of image features for representation, including PHOW [48], LBP [44], and CENTRIST [49].

MITIndoor-67 [50] contains $15K$ indoor images of 67 categories. Same to [31], the training subset which has $5,360$ images is adopted for clustering. Besides the three kinds of features for Scene-15, we also extract deep features based on pretrained VGG-VD [51] network to improve the performance.

Caltech-101 [52] includes $8,677$ object images of 101 categories. For each category, it has about 40 to 800 images. This dataset is the largest dataset used in all these related multi-view clustering methods. We adopt all these images of 101 classes to test the performance of clustering, which is same to [31]. Besides the three kinds of features for Scene-15, the Inception V3 [53] network is used to extract deep features.

2) Compared Methods: We compare our proposed approach ETLMSC and UR-ETLMSC (the proposed method without tensor rotation) with the following state-of-the-art methods, including two single view and six multi-view methods.

SPC$_{best}$ achieves the best result among all views with standard spectral clustering [1].

LRR$_{best}$ achieves the best result among all views with the low-rank representation [4].

Co-reg [5] is the co-regularization method for spectral clustering, which co-regularizes the clustering hypothesis to explore the complementary information.

RMSC [7] recovers a shared low-rank transition probability matrix as input to the Markov chain based spectral clustering.

DiMSC [6] employs the HSIC as a diversity term to explore the complementarity of multi-view representations.

LTMSC [30] adopts the low-rank tensor constraint for multi-view subspace clustering.

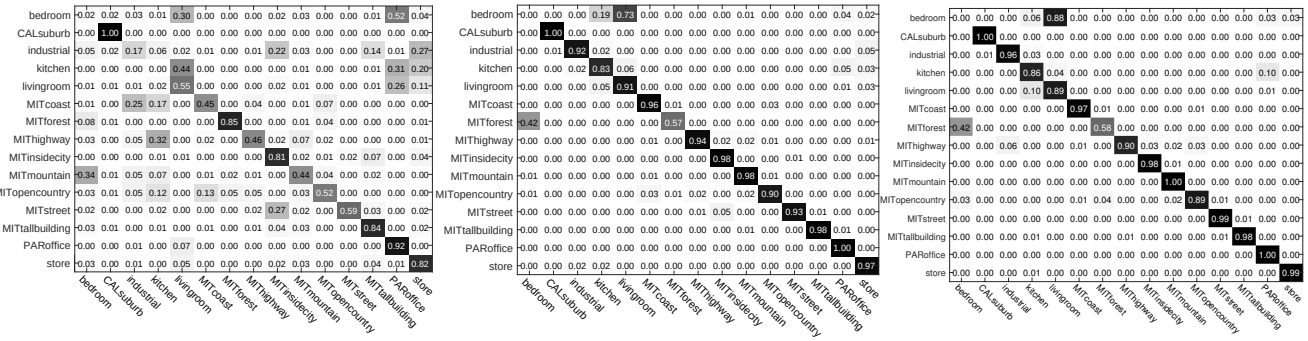
ECMSC [8] consists of position-aware exclusivity term and consistency term for regularization.

t-SVD-MSC [31] uses the t-SVD based tensor nuclear norm to learn optimal subspace.

Among all above methods, only SPC$_{best}$, Co-reg, and RMSC are spectral clustering methods, and other methods are self-representation based subspace clustering methods.

TABLE V
EXPERIMENTAL RESULTS ON THE SCENE-15 AND THE MITINDOOR-67 DATASETS. FOR ETLMSC, WE SET $\lambda = 0.003$ FOR BOTH TWO DATASETS.

Datasets	Scene-15						MITIndoor-67					
Methods	NMI	ACC	AR	F-score	Precision	Recall	NMI	ACC	AR	F-score	Precision	Recall
SPC$_{best}$	0.421	0.437	0.270	0.321	0.314	0.329	0.559	0.443	0.304	0.315	0.294	0.340
LRR$_{best}$	0.426	0.445	0.272	0.324	0.316	0.333	0.226	0.120	0.031	0.045	0.044	0.047
Co-reg	0.470	0.503	0.334	0.380	0.382	0.378	0.270	0.149	0.054	0.067	0.066	0.070
RMSC	0.564	0.507	0.394	0.437	0.425	0.450	0.342	0.232	0.110	0.123	0.121	0.125
DiMSC	0.269	0.300	0.117	0.181	0.173	0.190	0.383	0.246	0.128	0.141	0.138	0.144
LTMSC	0.571	0.574	0.424	0.465	0.452	0.479	0.226	0.120	0.031	0.045	0.044	0.047
ECMSC	0.463	0.457	0.303	0.357	0.318	0.408	0.590	0.469	0.323	0.333	0.314	0.355
UR-ETLMSC	0.536	0.534	0.369	0.419	0.420	0.419	0.467	0.335	0.204	0.216	0.211	0.220
t-SVD-MSC	0.858	0.812	0.771	0.788	0.743	0.839	0.750	0.684	0.555	0.562	0.543	0.582
ETLMSC	**0.902**	**0.878**	**0.851**	**0.862**	**0.848**	**0.877**	**0.899**	**0.775**	**0.729**	**0.733**	**0.709**	**0.758**



(a) LTMSC (b) t-SVD-MSC (c) ETLMSC

Fig. 4. The confusion matrices comparison among three tensor based methods, including the proposed LTMSC, t-SVD-MSC, and ETLMSC on the Scene-15 dataset.

TABLE VI
EXPERIMENTAL RESULTS ON THE CALTECH-101 DATASETS. FOR ETLMSC, WE SET $\lambda = 0.003$.

Datasets	Caltech-101					
Methods	NMI	ACC	AR	F-score	Precision	Recall
SPC$_{best}$	0.723	0.484	0.319	0.340	0.597	0.235
LRR$_{best}$	0.728	0.510	0.304	0.339	0.627	0.231
Co-reg	0.824	0.582	0.401	0.412	0.661	0.301
RMSC	0.573	0.346	0.246	0.258	0.457	0.182
DiMSC	0.589	0.351	0.226	0.253	0.362	0.191
LTMSC	0.788	0.559	0.393	0.403	0.670	0.288
ECMSC	0.662	0.419	0.312	0.326	0.465	0.251
UR-ETLMSC	0.740	0.463	0.342	0.352	0.638	0.243
t-SVD-MSC	0.858	0.607	0.430	0.440	0.742	0.323
ETLMSC	**0.899**	**0.639**	**0.456**	**0.465**	**0.825**	**0.324**

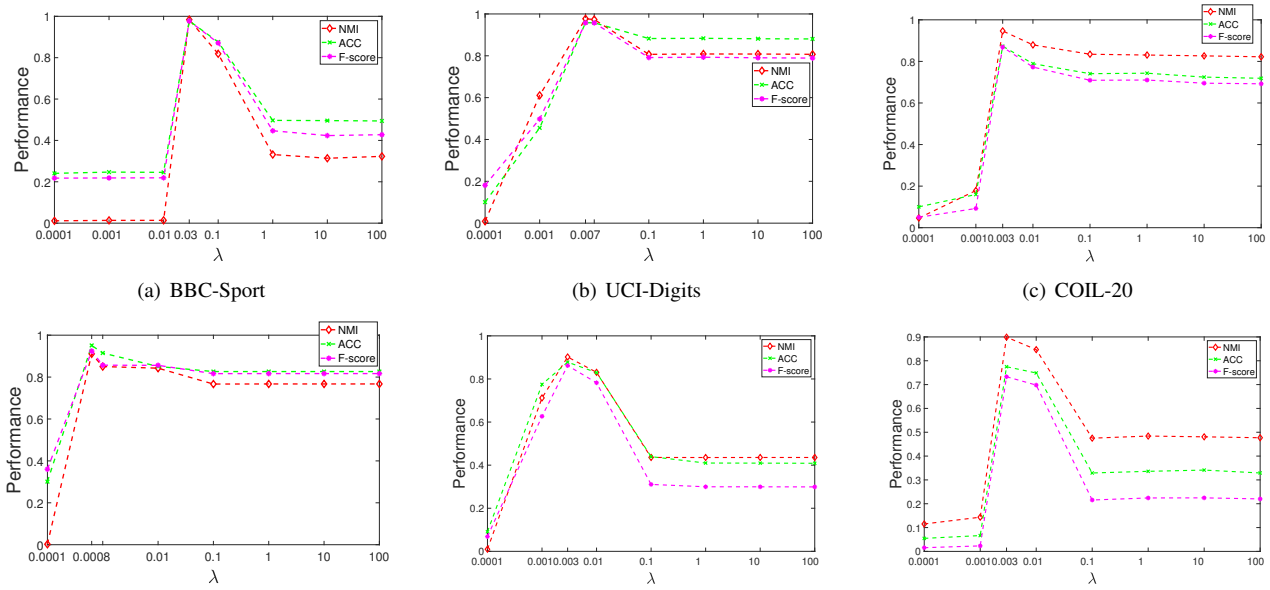
3) Evaluation Metrics: To comprehensively evaluate the performance of clustering, we adopt all six commonly used metrics including normalized mutual information (NMI), accuracy (ACC), adjusted rand index (AR), F-score , precision and recall. These six metrics favour different properties in clustering task. For all metrics, the higher value indicates the better performance.

B. Experimental Results and Analysis

1) Performance Comparison: We present the detailed clustering results on seven datasets in Tables III-VI. All results are measured by the average of 20 runs. In each table, the bold values represent the best performance. To better compare the performance of different methods, we divide all methods into four subclasses in the table, including single view methods, spectral clustering methods, subspace learning methods, and tensor based methods. The optimal parameters for these methods are fine-tuned by grid searching.

On all datasets, t-SVD-MSC and the proposed ETLMSC achieve the top two best results under nearly all these different metrics. From Tables III-VI, we can easily see that our proposed ETLMSC achieves the best performance on the BBC-Sport, UCI-Digits, COIL-20, Scene-15, MITIndoor-67, and Caltech-101 datasets under all six evaluation metrics. Especially on the BBC-Sport and MITIndoor-67 datasets, our results are more than 10% higher than the second best results achieved by t-SVD-MSC. There are also 2%, 2%, 6% and 3% improvement compared with the second best performance of t-SVD-MSC on the UCI-Digits, COIL-20, Scene-15, and Caltech-101 datasets, respectively. The Notting-Hill dataset is a video based face dataset. According to [54], [55], facial images have the subspace structure, and self-representation based subspace learning method is more suitable for this task. While t-SVD-MSC is based on subspace learning, the performance of our method is still comparable to that achieved by t-SVD-MSC, and much higher than those of all other methods, which is shown in the right part of Table IV.

For single view methods, they obtain good performance. But



(a) BBC-Sport (b) UCI-Digits (c) COIL-20

(d) Notting-Hill (e) Scene-15 (f) MITIndoor-67

Fig. 5. Parameter tuning with respect to λ on the first six datasets. Please note that the x-axis is in log scale.

in general, multi-view methods work better than single view methods. Moreover, both ECMSC and DiMSC work very well for this task. As they both try to investigate complementary information, it shows that it is necessary to learn view-specific information.

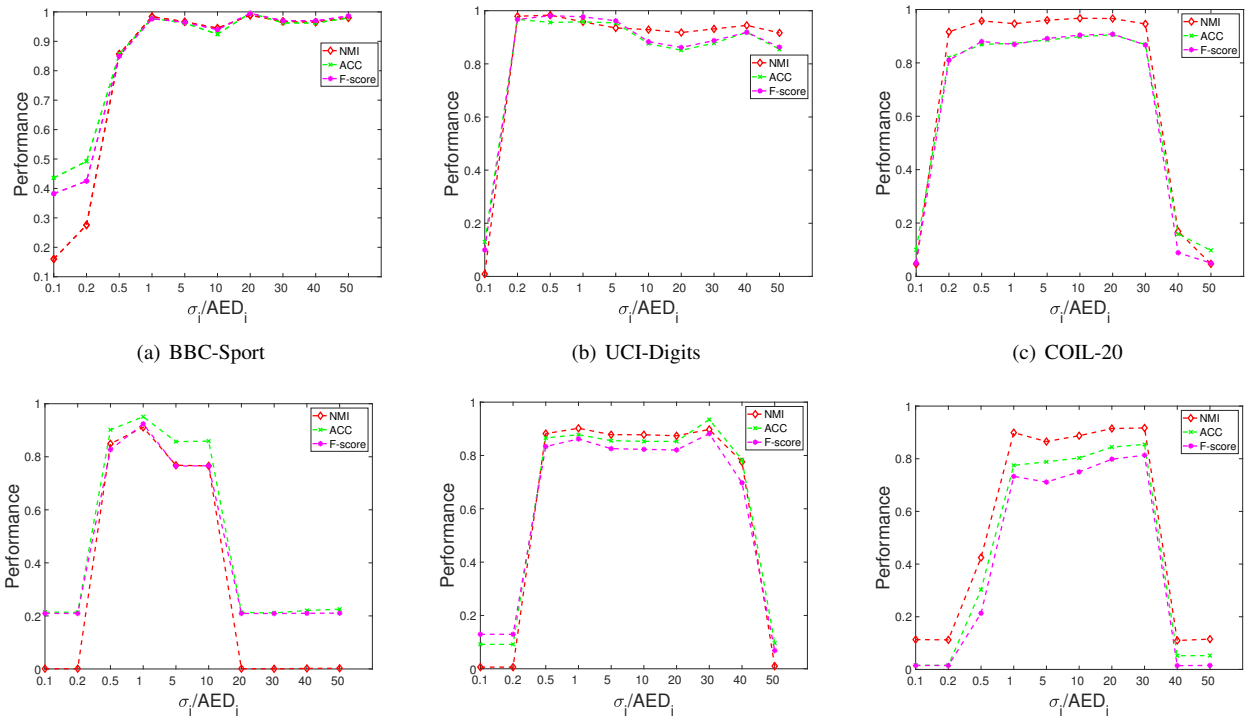
Tensor based methods, including ETLMSC and t-SVD-MSC, achieve significant improvement compared with all other state-of-the-art methods in most cases. There is a huge gap between tensor based methods and other methods, which can be attributed to the effectiveness of tensor based correlations exploration. In Fig. 4, we also present the confusion matrices of these three tensor based methods on the Scene-15 dataset. The row and column names correspond to the ground-truth and predicted labels, respectively. We can see that compared with LTMSC, our proposed ETLMSC and t-SVD-MSC achieve much better results in almost all classes in terms of accuracy, which can be attribute to the effectiveness of t-SVD decomposition based tensor nuclear norm. Compared with t-SVD-MSC, our ETLMSC improves slightly in many categories, which can also be verified by the accuracy.

Compared with RMSC, which is also a Markov chain based method, our proposed ETLMSC gains significant improvement. The main reason is that RMSC only captures the shared information among different view, while ETLMSC incorporates view-specific information that is useful for clustering. Based on the t-SVD based tensor nuclear norm to regularize the essential tensor, our method can well preserve these principle components among multi-view representations.

Tensor rotation plays an important role in our methods. Besides the complexity reduction, it can also largely improve the performance, which has already been validated by t-SVD-MSC [31]. We can see that ETLMSC achieves much better results than UR-ETLMSC on all datasets. The main reason is that after rotation, we can throughly investigate the complementary information among different views as the SVD is performed on each matrix composed of different view features after FFT. However, without rotation, the arrangement of similarity coefficients could be destroyed in Fourier domain, so that complementary information cannot be effectively explored. Therefore, UR-ETLMSC only sometime shows comparable performance with the state-of-the-art methods.

2) Parameter Sensitivity Analysis: There are mainly two parameters in our model, including the balance parameter λ and the standard deviation σ of Gaussian kernel to compute the similarity. In experiments, we find the optimal value for λ by grid searching. As for σ_i for the i-th view, we directly set it to the average Euclidean distance (AED_i) between all i-th view features, which is same to RMSC. We present the evaluation results of our proposed ETLMSC method on the first six datasets with respect to different λ and ratio of σ_i/AED_i in Figs. 5 and 6, respectively. From Fig. 5, we can observe that on these datasets, the performance of our proposed ETLMSC is relatively stable when λ varies in the range of $[0.0008, 0.01]$. λ plays an important role in balancing the contributions of these two parts. When it is very small (close to 0), the $\ell_{2,1}$ norm regularization on $\boldsymbol{\mathcal{E}}$ will not work. $\|\boldsymbol{\mathcal{Z}}\|_{\circledast}$ will be minimized as much as possible, which leads to $rank(\mathbf{Z}^{(i)}) \leq 1$. So the result is very bad. Moreover, the optimal parameter for each dataset is reported in their corresponding table.

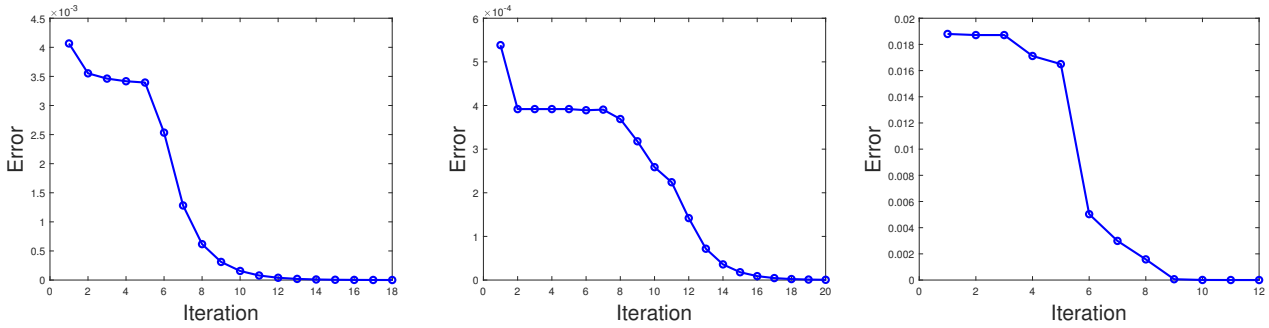
As for σ, all results of ETLMSC presented in Tables III-VI are based on the ratio $\sigma_i/AED_i = 1$. From Fig. 6, we can see that our method is not sensitive to this parameter when it varies in a certain large range. σ controls the discrimination of similarity. When σ is too small (or too large), all similarities will be close to 0 (or 1). It will be hard to distinguish the difference, which leads to bad results. For all the results reported in the manuscript, they are achieved with $\sigma_i/AED_i = 1$. We



(a) BBC-Sport (b) UCI-Digits (c) COIL-20

(d) Notting-Hill (e) Scene-15 (f) MITIndoor-67

Fig. 6. Influence of σ for Gaussian kernel based similarity on the first six datasets.



(a) COIL-20 (b) Notting-Hill (c) MITIndoor-67

Fig. 7. Convergence results on the COIL-20, Notting-Hill, and MITIndoor-67 datasets.

can see that with proper ratio, the performance can be further improved, especially on the BBCSport, UCI-Digit, COIL-20, Scene-15, and MITIndoor-67 datasets.

For the parameters μ and ρ of ADMM, we directly adopt the suggestion of [40] and fix them as 10^{-5} and 1.9, respectively. These two parameters mainly influence the number of iteration for convergence.

3) Convergence Analysis: The theoretical convergence of our algorithm has already been proved in [40]. In Fig. 7, we show the total error of our algorithm in each iteration on the COIL-20, Notting-Hill, and Caltech-101 datasets. Here, the total error is defined as the maximum value of changes in each iteration $\|\boldsymbol{\mathcal{Z}}^{k+1}-\boldsymbol{\mathcal{Z}}^{k}\|_\infty$, $\|\boldsymbol{\mathcal{E}}^{k+1}-\boldsymbol{\mathcal{E}}^{k}\|_\infty$, and reconstruction error $\|\tilde{\boldsymbol{\mathcal{P}}}-\boldsymbol{\mathcal{Z}}^{k+1}-\boldsymbol{\mathcal{E}}^{k+1}\|_\infty$:

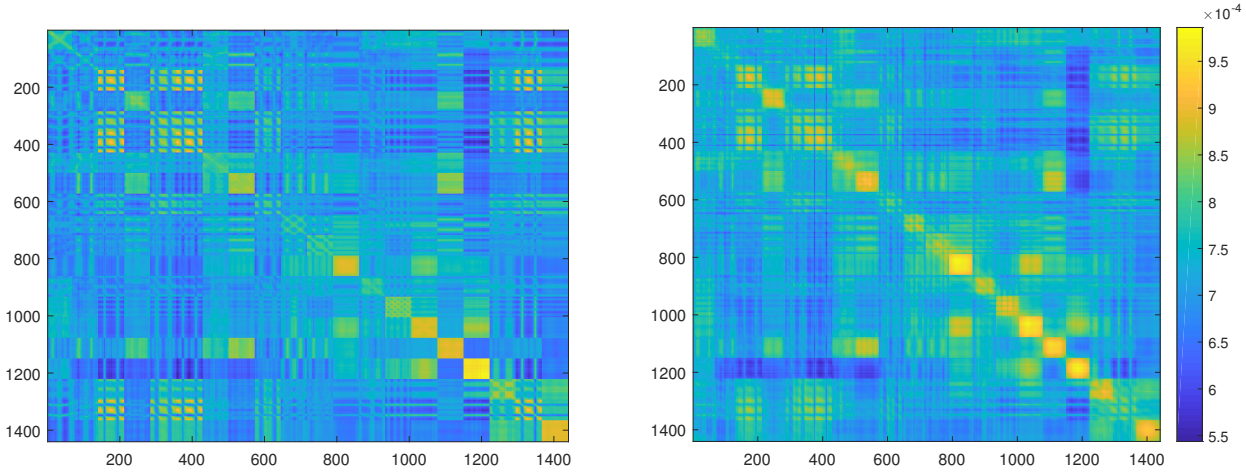
$$\text{Error}=\max(\|\Delta\boldsymbol{\mathcal{Z}}\|_\infty,\|\Delta\boldsymbol{\mathcal{E}}\|_\infty,\|\tilde{\boldsymbol{\mathcal{P}}}-\boldsymbol{\mathcal{Z}}^{k+1}-\boldsymbol{\mathcal{E}}^{k+1}\|_\infty).$$

According to Fig. 7, we can see that the error decreases with the increasing of iteration number. Our algorithm converges within 20 iterations, which is also true on other datasets. As we can compute the close-form solution in each iteration with relatively low computation complexity, our algorithm is very efficient.

4) Complexity Comparison: In Table VII, we present computation complexity and running time of the state-of-the-art methods on all these datasets. Since all these methods share the similar post-processing procedure that has the same complexity, we only report the computational complexity and running time for learning the affinity matrix. We need to mention that the number of iteration K has an obvious affect on the running time, and parameter selection will influence K. So we can see that the running time of ECMSC on the UCI-Digit dataset could be shorter than that on the COIL-20 dataset. We

TABLE VII
TIME COMPLEXITY AND RUNNING TIME TO COMPUTE AFFINITY MATRIX ON THESE DATASETS OF DIFFERENT METHODS. K, M, N ARE THE NUMBER OF ITERATIONS, VIEWS, AND SAMPLES, RESPECTIVELY. ALL THE TIME ARE MEASURED BY SECONDS.

Methods	RMSC	DiMSC	LTMSC	ECMSC	t-SVD-MSC	ETLMSC(Ours)
Complexity	$\mathcal{O}(KN^3)$	$\mathcal{O}(KMN^3)$	$\mathcal{O}(KMN^3)$	$\mathcal{O}((K+M)N^3)$	$\mathcal{O}(MN^3+KMN^2\log(N))$	$\mathcal{O}(KMN^2\log(N))$
Time on BBC-Sport	4.5	35.8	23.4	78.7	10.6	**2.1**
Time on COIL-20	74.8	1075.1	375.9	954.2	103.4	**19.6**
Time on UCI-Digit	214.6	2706.4	959.3	468.5	225.7	**54.6**
Time on Notting-Hill	2531.3	43813.6	10408.7	6319.3	3373.3	**562.8**
Time on Scene-15	2407.9	38904.7	9270.6	5663.9	2627.8	**489.7**
Time on MITIndoor-67	3796.5	66274.3	15759.2	9673.2	5957.5	**930.5**
Time on Caltech-101	15710.9	218825.5	76833.2	41558.6	18929.7	**5395.7**



(a) RMSC (b) ETLMSC

Fig. 8. Visualization of learned transition probability matrices of two spectral clustering based methods on the COIL-20 dataset.

can see that our method has the lowest complexity and the shortest processing time among these related approaches on all datasets, which demonstrates the efficiency of our proposed method. For example, on the COIL-20 dataset, our algorithm can finish within 20 seconds, while the second best method RMSC needs more than 70 seconds, and t-SVD-MSC costs more than 100 seconds. On the largest Caltech-101 dataset, our method can save much time compared with t-SVD-MSC.

5) Representation Visualization: In Fig. 8, we show the visualization of the learned optimal transition probability matrix. Due to the limition of space, we only present the results of two Markov chain based spectral clustering methods (RMSC and our proposed ETLMSC) on the COIL-20 dataset. For ETLMSC, the transition probability matrix is computed by the average of lateral slices of the optimal essential tensor $\mathcal{Z}$. The yellow color represents the large value. Compared with the result of RMSC in Fig. 8(a), we can easily see that the result of ETLMSC in Fig. 8(b) is much better as most large values concentrate on the diagonal blocks. This can also be verified by comparing the experimental results in Tables III-V. While RMSC only captures shared information among different views, it is more meaningful for our ETLMSC method to explore high order multi-view correlations based on tensor formulation.

6) Comparison with t-SVD-MSC: t-SVD-MSC [31] achieves very good performance for the task of multi-view clustering. Both the proposed ETLMSC and t-SVD-MSC [31] are based on the tensor nuclear norm defined by the t-SVD for multi-view clustering. But there are many differences. First, construction of affinity matrix and tensor is totally different. We adopt the Markov chain to compute the transition probability matrix, while t-SVD-MSC is based on self-representation, which is of high computation complexity and under the assumption of subspace structure. Second, the model and optimization process are much different. We directly divide the transition probability tensor into two parts with low-rank and sparse constraints, while their method need to optimize the self-representation coefficients. So the optimization process is also different. Most importantly, compared with t-SVD-MSC, based on the experimental results presented above, our method achieves better performance with much lower complexity and less processing time.

VI. CONCLUSION AND FUTURE WORK

In this paper, we propose a novel essential tensor learning method for Markov chain based multi-view spectral clustering. Based on multi-view transition probability matrices, we construct a 3-order tensor. We explore the high order correlations

among multiple views by learning the essential tensor with low-rank constraint based on t-SVD based tensor nuclear norm. With tensor rotation operation, the proposed algorithm can be optimized efficiently and the principle components can be well preserved. We evaluate the performance of our method on seven datasets with respect to different applications, and it achieves superior performance compared with the state-of-the-art methods.

For future work, we would like to focus on the fast and scalable algorithms, such as the sampling technique or recover the subspace of the whole tensor with a much smaller seed tensor. So that the computation complexity of the proposed model can be further reduced, which will make ETLMSC much suitable for large-scale applications.

ACKNOWLEDGMENT

We would like to thank Dr. Yuan Xie for his selfless support in sharing codes and datasets as well as the valuable suggestions.

REFERENCES

[1] A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," in *Proceedings of the Neural Information Processing Systems*, pp. 849–856, 2002.

[2] G. Liu, Z. Lin, and Y. Yu, "Robust subspace segmentation by low-rank representation," in *Proceedings of the International Conference on Machine Learning*, pp. 663–670, 2010.

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

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

[5] A. Kumar, P. Rai, and H. Daume, "Co-regularized multi-view spectral clustering," in *Proceedings of the Neural Information Processing Systems*, pp. 1413–1421, 2011.

[6] X. Cao, C. Zhang, H. Fu, S. Liu, and H. Zhang, "Diversity-induced multi-view subspace clustering," in *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 586–594, 2015.

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

[8] X. Wang, X. Guo, Z. Lei, C. Zhang, and S. Z. Li, "Exclusivity-consistency regularized multi-view subspace clustering," in *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 923–931, 2017.

[9] J. Wu, K. Long, F. Wang, C. Qian, C. Li, Z. Lin, and H. Zha, "Deep comprehensive correlation mining for image clustering," *arXiv preprint arXiv:1904.06925*, 2019.

[10] X. Xie, X. Guo, G. Liu, and J. Wang, "Implicit block diagonal low-rank representation," *IEEE Transactions on Image Processing*, vol. 27, no. 1, pp. 477–489, 2018.

[11] J. Wright, A. Ganesh, S. Rao, Y. Peng, and Y. Ma, "Robust principal component analysis: Exact recovery of corrupted low-rank matrices via convex optimization," in *Proceedings of the Neural Information Processing Systems*, pp. 2080–2088, 2009.

[12] X. Xie, J. Wu, G. Liu, and J. Wang, "Matrix recovery with implicitly low-rank data," *Neurocomputing*, vol. 334, pp. 219–226, 2019.

[13] C. Lu, J. Feng, Y. Chen, W. Liu, Z. Lin, and S. Yan, "Tensor robust principal component analysis: Exact recovery of corrupted low-rank tensors via convex optimization," in *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 5249–5257, 2016.

[14] P. Zhou and J. Feng, "Outlier-robust tensor pca," in *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 3938–3946, 2017.

[15] P. Zhou, C. Lu, Z. Lin, and C. Zhang, "Tensor factorization for low-rank tensor completion," *IEEE Transactions on Image Processing*, vol. 27, no. 3, pp. 1152–1163, 2018.

[16] H. Kong, X. Xie, and Z. Lin, "t-schatten-p norm for low-rank tensor recovery," *IEEE Journal of Selected Topics in Signal Processing*, 2018.

[17] J. D. Carroll and J.-J. Chang, "Analysis of individual differences in multidimensional scaling via an n-way generalization of eckart-young decomposition," *Psychometrika*, vol. 35, no. 3, pp. 283–319, 1970.

[18] R. A. Harshman, "Foundations of the parafac procedure: Models and conditions for an" explanatory" multimodal factor analysis," 1970.

[19] L. R. Tucker, "Some mathematical notes on three-mode factor analysis," *Psychometrika*, vol. 31, no. 3, pp. 279–311, 1966.

[20] M. E. Kilmer, K. Braman, N. Hao, and R. C. Hoover, "Third-order tensors as operators on matrices: A theoretical and computational framework with applications in imaging," *SIAM Journal on Matrix Analysis and Applications*, vol. 34, no. 1, pp. 148–172, 2013.

[21] B. Huang, C. Mu, D. Goldfarb, and J. Wright, "Provable low-rank tensor recovery," *Optimization-Online*, vol. 4252, p. 2, 2014.

[22] Z. Zhang, G. Ely, S. Aeron, N. Hao, and M. Kilmer, "Novel methods for multilinear data completion and de-noising based on tensor-svd," in *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 3842–3849, 2014.

[23] A. Kumar and H. Daumé, "A co-training approach for multi-view spectral clustering," in *Proceedings of the International Conference on Machine Learning*, pp. 393–400, 2011.

[24] H. Wang, C. Weng, and J. Yuan, "Multi-feature spectral clustering with minimax optimization," in *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 4106–4113, 2014.

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

[26] D. Zhou, J. Huang, and B. Schölkopf, "Learning from labeled and unlabeled data on a directed graph," in *Proceedings of the International Conference on Machine Learning*, pp. 1036–1043, 2005.

[27] D. Zhou and C. J. Burges, "Spectral clustering and transductive learning with multiple views," in *Proceedings of the International Conference on Machine Learning*, pp. 1159–1166, 2007.

[28] Y. Wang, L. Wu, X. Lin, and J. Gao, "Multiview spectral clustering via structured low-rank matrix factorization," *IEEE Transactions on Neural Networks and Learning Systems*, 2018.

[29] M. Brbić and I. Kopriva, "Multi-view low-rank sparse subspace clustering," *Pattern Recognition*, vol. 73, pp. 247–258, 2018.

[30] C. Zhang, H. Fu, S. Liu, G. Liu, and X. Cao, "Low-rank tensor constrained multiview subspace clustering," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1582–1590, 2015.

[31] 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," *International Journal of Computer Vision*, pp. 1–23, 2018.

[32] M. Yin, J. Gao, S. Xie, and Y. Guo, "Multiview subspace clustering via tensorial t-product representation," *IEEE Transactions on Neural Networks and Learning Systems*, no. 99, pp. 1–14, 2018.

[33] C. Zhang, Q. Hu, H. Fu, P. Zhu, and X. Cao, "Latent multi-view subspace clustering," in *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, vol. 30, pp. 4279–4287, 2017.

[34] C. Zhang, H. Fu, Q. Hu, X. Cao, Y. Xie, D. Tao, and D. Xu, "Generalized latent multi-view subspace clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018.

[35] C.-G. Li and R. Vidal, "Structured sparse subspace clustering: A unified optimization framework," in *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 277–286, 2015.

[36] K. Chaudhuri, S. M. Kakade, K. Livescu, and K. Sridharan, "Multi-view clustering via canonical correlation analysis," in *Proceedings of the International Conference on Machine Learning*, pp. 129–136, 2009.

[37] C. Cortes, M. Mohri, and A. Rostamizadeh, "Learning non-linear combinations of kernels," in *Proceedings of the Neural Information Processing Systems*, pp. 396–404, 2009.

[38] J. Xu, J. Han, and F. Nie, "Discriminatively embedded k-means for multi-view clustering," in *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 5356–5364, 2016.

[39] J. A. Hartigan and M. A. Wong, "Algorithm as 136: A k-means clustering algorithm," *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, vol. 28, no. 1, pp. 100–108, 1979.

[40] Z. Lin, R. Liu, and Z. Su, "Linearized alternating direction method with adaptive penalty for low-rank representation," in *Proceedings of the Neural Information Processing Systems*, pp. 612–620, 2011.
[41] W. Hu, D. Tao, W. Zhang, Y. Xie, and Y. Yang, "The twist tensor nuclear norm for video completion," *IEEE Transactions on Nuural Networks and Learning Systems*, pp. 1–13, 2017.
[42] D. Greene and P. Cunningham, "Practical solutions to the problem of diagonal dominance in kernel document clustering," in *Proceedings of the International Conference on Machine Learning*, pp. 377–384, 2006.
[43] A. Asuncion and D. Newman, "UCI machine learning repository," 2007.
[44] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 971–987, 2002.
[45] M. Lades, J. C. Vorbruggen, J. Buhmann, J. Lange, C. Von Der Malsburg, R. P. Wurtz, and W. Konen, "Distortion invariant object recognition in the dynamic link architecture," *IEEE Transactions on Computers*, vol. 42, no. 3, pp. 300–311, 1993.
[46] Y.-F. Zhang, C. Xu, H. Lu, and Y.-M. Huang, "Character identification in feature-length films using global face-name matching," *IEEE Transactions on Multimedia*, vol. 11, no. 7, pp. 1276–1288, 2009.
[47] L. Fei-Fei and P. Perona, "A bayesian hierarchical model for learning natural scene categories," in *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 524–531, 2005.
[48] A. Bosch, A. Zisserman, and X. Munoz, "Image classification using random forests and ferns," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1–8, 2007.
[49] J. Wu and J. M. Rehg, "Centrist: A visual descriptor for scene categorization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 8, pp. 1489–1501, 2011.
[50] A. Quattoni and A. Torralba, "Recognizing indoor scenes," in *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 413–420, 2009.
[51] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.
[52] L. Fei-Fei, R. Fergus, and P. Perona, "Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories," *Computer Vision and Image Understanding*, vol. 106, no. 1, pp. 59–70, 2007.
[53] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 2818–2826, 2016.
[54] D. Arpit, I. Nwogu, and V. Govindaraju, "Dimensionality reduction with subspace structure preservation," in *Proceedings of the Neural Information Processing Systems*, pp. 712–720, 2014.
[55] G. Zhang, R. He, and L. S. Davis, "Jointly learning dictionaries and subspace structure for video-based face recognition," in *Proceedings of the IEEE Asian Conference on Computer Vision*, pp. 97–111, 2014.

Jianlong Wu received the B.E. degree in electronics and information engineering from Huazhong University of Science and Technology in 2014. He is currently pursuing the Ph.D. degree with the School of Electronics Engineering and Computer Science, Peking University. His research interests include computer vision, pattern recognition and machine learning.

Zhouchen Lin (M'00-SM'08-F'18) received the Ph.D. degree in applied mathematics from Peking University in 2000. He is currently a Professor with the Key Laboratory of Machine Perception (MOE), School of Electronics Engineering and Computer Science, Peking University. His research interests include computer vision, image processing, machine learning, pattern recognition, and numerical optimization. He is a fellow of the IAPR and the IEEE. He is an Associate Editor of the IEEE Transactions on Pattern Analysis and Machine Intelligence and the International Journal of Computer Vision, an Area Chair of ACCV 2009/2018, CVPR 2014/2016/2019, ICCV 2015, NIPS 2015/2018/2019, and AAAI 2019, and a Senior Program Committee Member of AAAI 2016/2017/2018 and IJCAI 2016/2018/2019.

Hongbin Zha (M'06) received the M.S. and Ph.D. degrees in electrical engineering from Kyushu University, Fukuoka, Japan, in 1987 and 1990, respectively. He joined Kyushu University in 1991 as an associate professor. He was a Research Associate with the Kyushu Institute of Technology. He was also a Visiting Professor with the Center for Vision, Speech, and Signal Processing, Surrey University, U.K., in 1999. Since 2000, he has been a Professor with the Key Laboratory of Machine Perception, Peking University, Beijing, China. He has authored more than 300 technical publications in journals, books, and international conference proceedings. His research interests include computer vision, digital geometry processing, and robotics. He received the Franklin V. Taylor Award from the IEEE Systems, Man, and Cybernetics Society in 1999. He is a member of the IEEE Computer Society.