Error-robust low-rank tensor approximation for multi-view clustering

Shuqin Wang [a,b], Yongyong Chen [c], Yi Jin [d], Yigang Cen [a,b,*], Yidong Li [d], Linna Zhang [e]

[a] *Institute of Information Science, Beijing Jiaotong University, Beijing 100044, China*
[b] *Beijing Key Laboratory of Advanced Information Science and Network Technology, Beijing 100044, China*
[c] *School of Computer Science and Technology, Harbin Institute of Technology, Shenzhen, Shenzhen 518055, China*
[d] *School of Computer and Information Technology, Beijing Jiaotong University, Beijing 100044, China*
[e] *College of Mechanical Engineering, Guizhou University, Guiyang 550025, China*

ABSTRACT

Multi-view clustering has been widely developed to improve the clustering performance over that of single-view clustering. However, the types of errors vary and behave inconsistently in each view, which results in performance degradation in real applications. To address this limitation, in this paper, we propose a novel Markov chain-based spectral clustering method for multi-view clustering to handle different types of errors. Unlike most of the existing self-representation-based subspace clustering methods, which process each view separately, ignoring complementary information among views, our method first computes the transition probability matrices of all views, then forms each transition probability matrix as a frontal slice of a third-order tensor to capture the multi-view information, and finally decomposes the tensor into an ideal tensor with the tensor nuclear norm constraint and an error term. Furthermore, the proposed method imposes the group l_1 and $l_{2,1}$ norms on the error matrix for error learning such that various errors can be clearly characterized and processed to improve performance. To solve the challenging optimization model, we propose an efficient algorithm using the augmented Lagrangian multiplier method. Experimental results on three real-world datasets show that the proposed method is superior to the state-of-the art methods in various evaluation metrics.

© 2021 Elsevier B.V. All rights reserved.

1. Introduction

Multi-view clustering has attracted great research interest in various areas, such as face clustering [1,2], human action recognition [3], image representation [4], and image co-segmentation [5]. Compared with single-view clustering, multi-view clustering can use relevant information from multiple views to divide the data into different clusters, so that the data within the same cluster are more likely to be similar to each other than those in other clusters. Considering that the feature data exist in several subspaces, multiple methods based on subspace clustering have been proposed, such as sparse subspace clustering (SSC) [6] and low-rank representation (LRR) [7], which use the self-representation matrix to perform clustering. However, the above methods only consider single-view information.

To handle the multi-view clustering task, the works in [4, 8,9] extended the single-view self-representation methods SSC and LRR into a multi-view setting. [8] represented multi-view data using the shared low-rank representation and a specific residual representation. The work in [9] used the sparse and

low-rank decomposition [10] of each view to form a common representation. [4] introduced a manifold constraint on the basis of low-rank subspace clustering to preserve the local geometrical structure of multiple features. Another effective multi-view extended method is feature concatenation multi-view subspace clustering [11], which uses low-rank representation to explore the consensus information of multi-view data. However, in real applications, the above matrix optimization-based methods have one common limitation, i.e., the high-order correlations among different views may not be fully used [12–14]. The reason is that all views are processed independently or concatenated as one vector [11].

To explore the high-order correlations among different views, tensor learning-based methods have been developed. Low-rank tensor constrained multi-view subspace clustering (LT-MSC) [15] was proposed to construct a third-order representation tensor to explore high-order correlations, fully preserving the spatial structure of the data. Xie et al. [16] introduced a new tensor decomposition scheme to ensure consistency between multiple views. However, the above tensor methods [12,15] may encounter high computational complexity. To address this, Wu et al. [13] developed an essential tensor learning method (ETLMSC) by extending robust multi-view spectral clustering (RMSC) [17]. Different from [12,15], which impose the low-rank constraint on the

* Corresponding author at: Institute of Information Science, Beijing Jiaotong University, Beijing 100044, China.
E-mail address: ygcen@bjtu.edu.cn (Y. Cen).

representation tensor, RMSC and ETLMSC assume the transition probability matrix or tensor to be low-rank, respectively. To eliminate the influence of errors, error robust multi-view clustering (EMVC) [18] was proposed to learn the shared transition probability matrix. Although the above tensor-based methods have achieved promising clustering results, when multi-view features are contaminated by various noise, the data points belonging to the same cluster may be misclassified. The reason is that LT-MSC, ETLMSC and [12] process the error of each view independently, which may cause the magnitude values of the error matrix to be inconsistent.

Inspired by the advantages of the recently proposed tensor-learning methods [12,13,15,19] and robust handling of various errors in multi-view data, we propose the error-robust low-rank tensor approximation model (ELRTA) for multi-view clustering, which constructs the probability transition matrices of all views into a third-order tensor. In addition, we decompose it into a noise-free tensor and an error term. The group l_1 and $l_{2,1}$ [20] norms are imposed on the error term to handle various types of errors. Furthermore, ELRTA vectorizes the error matrix of each view along its columns to make the magnitude values of the error matrix consistent. Fig. 1 shows the flowchart of the proposed ELRTA model. Given multi-view features $X^{(1)}, \ldots, X^{(M)}$, ELRTA first establishes the similarity matrix $S^{(i)}$ of each view. Then, ELRTA calculates the view-specific transition probability matrix $P^{(i)}$ via $P^{(i)} = (D^{(i)})^{-1}S^{(i)}$ and constructs the transition probability tensor $\mathcal{P}$ by stacking the transition probability matrix of each view together. Finally, ELRTA decomposes $\mathcal{P}$ as the sum of $\mathcal{Z}$ and $\mathcal{E}$, where $\mathcal{Z}$ is the noise-free transition probability tensor and $\mathcal{E}$ is the error tensor. $\mathcal{Z}$ is rotated to $\widetilde{\mathcal{Z}}$, which is imposed by tensor nuclear norm constraint based on tensor singular value decomposition (t-SVD). For $\mathcal{E}$, the group l_1 and $l_{2,1}$ norms are applied to eliminate various types of errors. Our contributions can be summarized as follows:

- We propose a novel multi-view clustering method that incorporates the tensor learning technique and the error robustness into a unified model, which is named error-robust low-rank tensor approximation (ELRTA).
- Instead of learning each view independently, ELRTA enforces nuclear norm constraint on tensor constructed by the transition probability matrices of multiple views. The constructed tensor is decomposed into a noise-free tensor and an error term, which are measured by the tensor nuclear norm and the group l_1 and $l_{2,1}$ norms, respectively.
- An effective optimization algorithm is proposed to solve the ELRTA model. Experimental results on three real-world datasets show that the proposed ELRTA is superior to several state-of-the-art methods.

The remainder of this paper is organized as follows. The related works for multi-view clustering are summarized in Section 2. Some preliminaries are shown in Section 3. Section 4 presents the proposed ELRTA model and designs an effective algorithm to solve the ELRTA model. Section 5 reports the results of extensive experiments and model analysis. The conclusion of this paper is summarized in Section 6.

2. Related work

Recent multi-view clustering methods are briefly reviewed in this section. They can be roughly classified into three categories.

Matrix factorization based methods: The main idea of these matrix factorization-based methods is that they decomposed multi-view data into a common representation by minimizing the overall loss terms of different views. Liu et al. [21] employed appropriate consensus coefficient matrices to balance all coefficient matrices on the basis of decomposing the original matrix. To maintain the local structures of multiple features, Wang et al. [22] proposed a multi-view conceptual clustering method, which provides a universal consensus representation for multiple views based on concept decomposition with local manifold regularization. Huang et al. [23] proposed a deep multi-view clustering model utilizing a collaborative deep matrix decomposition framework.

Graph-based methods: The main idea of graph-based methods is that they first construct the similarity matrices of all views from the original features and then fuse them into one graph to reveal the essential correlation among different views. For example, Tang et al. [24] used linked matrix factorization (LMF) to fuse multiple graph sources. Li et al. [25] proposed learning a bipartite graph based on spectral clustering, which uses local manifold fusion to integrate heterogeneous features and applies bipartite graphs to approximate similar graphs to improve the computational efficiency. Since the clustering performance depends heavily on the quality of the graph, Zhan et al. [26] proposed a graph learning-based method to improve the quality of a graph using rank constraints. The work in [27] automatically learned the optimal weight of each graph without additional parameters. Following this, the method in [28] incorporated clustering and adaptive neighbor learning in a unified model. Wang et al. [29] proposed learning each view graph matrix and unified matrix in a mutually reinforcing manner.

Subspace clustering-based methods: The main idea of the methods based on subspace clustering is different from the graph-based methods that learn the similarity matrix from the original data, subspace clustering-based methods learn the similarity matrix from the representation obtained by different regularizers. The most representative methods include SSC [6], LRR [7], and the least squares regression (LSR) [30]. Among them, SSC pursues the sparse representation of self-representing matrix using the l_1 norm; LRR pursues the low-rank representation by the nuclear norm; LSR learns the representation matrix through the least squares regression. Once the affinity matrix is obtained, all of them use the spectral clustering algorithm to yield the final clustering results. However, the above methods only handle the single-view clustering and cannot fully utilize multiple views information. In order to describe the data more completely, various multi-view subspace learning methods have been proposed, such as: diversity-induced multi-view subspace clustering (DiMSC) [31], exclusivity-consistency regularized multi-view subspace clustering (ECMSC) [32], latent multi-view subspace clustering (LMSC) [33], consistent and specific multi-view subspace clustering (CSMSC) [34]. Specifically, DiMSC and ECMSC use the diversity among different views to obtain a cluster structure with complementary information. LMSC uses latent representation to explore potential complementary information from multiple views. CSMSC jointly utilizes the consistency and specificity for subspace representation learning. However, consistency-based methods cannot fully use the diversity between different views and complementary-based methods cannot maintain a common clustering structure. Recently, [17] proposed a Markov chain method for learning a shared transition probability matrix based on low-rank and sparse decomposition. Most of the previous methods only considered the pairwise correlations of the samples and ignored high-order correlations among multiple features. Therefore, several tensor-based methods have been proposed to overcome the above limitations. Zhang et al. [15] proposed a multi-view subspace clustering method with the low-rank tensor constraint in which all subspace representation matrices are integrated to capture high-order correlations. The study in [12] employed the tensor nuclear norm based on t-SVD to perform the low-rank approximation [35]. Following this
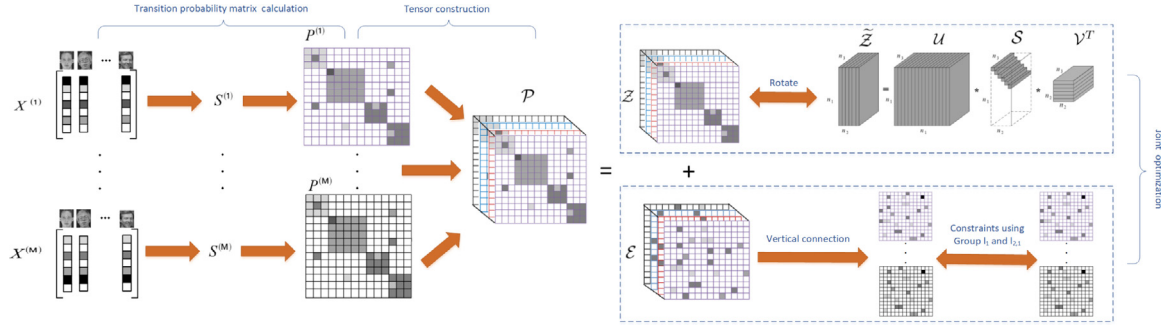
Fig. 1. The flowchart of the proposed ELRTA.

line, Chen et al. [14] proposed to learn the graph regularized low-rank representation tensor and affinity matrix (GLTA) simultaneously. Unlike the above self-representation-based subspace clustering methods, [13] studied essential tensor learning based on the Markov chain to explore the high-order correlations of multi-view features.

Our work is connected to the works mentioned above, but there are several significant differences between our work and them. Although the above works have achieved encouraging results, they mainly focus on the relevant information of single-view or multiple views, and ignore the impact of various types of errors on the clustering results. Besides, most existing methods cannot incorporate error removal in the multi-view clustering task. In this paper, we proposed an error learning method based on Markov chain to solve the above shortcomings.

3. Notations and preliminaries

In this section, we will introduce some notations and preliminaries used in the remainder of this paper. Consider an N-order tensor, which is represented using bold calligraphy letters (e.g., $\mathcal{X}$). The matrices and vectors are represented by upper case letters (e.g., X) and lower case letters (e.g., x) respectively. For an N-order tensor $\mathcal{X} \in \mathcal{R}^{n_1 \times n_2 \times \cdots \times n_N}$, a slice is a 2D matrix defined by fixing all but two indices, and a fiber is a 1D vector defined by fixing all but one [12]. For a third-order tensor $\mathcal{X} \in \mathcal{R}^{n_1 \times n_2 \times n_3}$, we denote its element (i, j, k) as $\mathcal{X}_{ijk}$. The Frobenius norm is $\|\mathcal{X}\|_F = \left(\sum_{i,j,k} |\mathcal{X}_{ijk}|^2 \right)^{\frac{1}{2}}$. We use the matlab notation $\mathcal{X}(i, :, :), \mathcal{X}(:, i, :), \mathcal{X}(:, :, i)$ to denote the ith horizontal, lateral and frontal slice, where the frontal slice $\mathcal{X}(:, :, i)$ is also denoted as $\mathcal{X}^{(i)}$. The Fourier transform alone the third dimension is denoted as $\bar{\mathcal{X}} = fft(\mathcal{X}, [], 3)$. Besides, the block circular matrix, block vectorization, block diag matrix and the corresponding opposite operations are defined as follows

$$bcirc(\mathcal{X}) = \begin{bmatrix} \mathcal{X}^{(1)} & \mathcal{X}^{(n_3)} & \cdots & \mathcal{X}^{(2)} \\ \mathcal{X}^{(2)} & \mathcal{X}^{(1)} & \cdots & \mathcal{X}^{(3)} \\ \vdots & \ddots & \ddots & \vdots \\ \mathcal{X}^{(n_3)} & \mathcal{X}^{(n_3-1)} & \cdots & \mathcal{X}^{(1)} \end{bmatrix},$$

$$bvec(\mathcal{X}) = \begin{bmatrix} \mathcal{X}^{(1)} \\ \mathcal{X}^{(2)} \\ \vdots \\ \mathcal{X}^{(n_3)} \end{bmatrix}, \quad bvfold(bvec(\mathcal{X})) = \mathcal{X},$$

$$bdiag(\mathcal{X}) = \begin{bmatrix} \mathcal{X}^{(1)} & & \\ & \ddots & \\ & & \mathcal{X}^{(n_3)} \end{bmatrix}, \quad bdfold(bdiag(\mathcal{X})) = \mathcal{X}.$$

We introduce the following related definitions and theorems [19] of t-SVD and the tensor nuclear norm in Eq. (5), which are involved in the solution of the proposed method.

Definition 1 (*t-Product*). Let $\mathcal{X} \in \mathcal{R}^{n_1 \times n_2 \times n_3}$, and $\mathcal{Y}$ be $n_2 \times n_4 \times n_3$. The t-product $\mathcal{X} * \mathcal{Y}$ is a tensor of size $n_1 \times n_4 \times n_3$

$$\mathcal{M} = \mathcal{X} * \mathcal{Y} =: bvfold\{bcirc(\mathcal{X})bvec(\mathcal{Y})\}. \tag{1}$$

Definition 2 (*Tensor Transpose*). If $\mathcal{X}$ is $n_1 \times n_2 \times n_3$, then the transpose tensor $\mathcal{X}^T$ is an $n_2 \times n_1 \times n_3$ tensor obtained by transposing frontal slice 2 through n_3.

Definition 3 (*Identity Tensor*). The identity tensor $\mathcal{I} \in \mathcal{R}^{n_1 \times n_2 \times n_3}$ is the tensor whose first frontal slice is the $n_1 \times n_1$ identity matrix and all other frontal slices are zero.

Definition 4 (*Orthogonal Tensor*). A tensor $\mathcal{Q} \in n_1 \times n_2 \times n_3$ is orthogonal if it satisfies

$$\mathcal{Q}^T * \mathcal{Q} = \mathcal{Q} * \mathcal{Q}^T = \mathcal{I}, \tag{2}$$

where $*$ is the t-product mentioned in Definition 1. $\mathcal{Q}^T \in \mathcal{R}^{n_2 \times n_1 \times n_3}$ and $\mathcal{I} \in \mathcal{R}^{n_1 \times n_2 \times n_3}$ are the transpose of tensor $\mathcal{Q}$ and the identity tensor, respectively. The detailed explanation is given in Definitions 2 and 3.

Definition 5 (*f-Diagonal Tensor*). A tensor is called f-diagonal if each of its frontal slices is diagonal matrix.

Theorem 1 (*t-SVD*). *Let* $\mathcal{X} \in \mathcal{R}^{n_1 \times n_2 \times n_3}$, *it can be factored as*

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

where $\mathcal{U} \in \mathcal{R}^{n_1 \times n_1 \times n_3}$ and $\mathcal{V} \in \mathcal{R}^{n_2 \times n_2 \times n_3}$ are orthogonal tensors mentioned in Definition 4, $\mathcal{S} \in \mathcal{R}^{n_1 \times n_2 \times n_3}$ denotes an f-diagonal tensor mentioned in Definition 5. Fig. 1 illustrates the decomposition.

According to [19], if the t-SVD of $\mathcal{X} \in \mathcal{R}^{n_1 \times n_2 \times n_3}$ is given by $\mathcal{X} = \mathcal{U} * \mathcal{S} * \mathcal{V}^T$ in Eq. (3), then it is can be written as a finite sum of outer products of matrices:

$$\mathcal{X} = \sum_{i=1}^{min(n_1, n_2)} \mathcal{U}(:, i, :) * \mathcal{S}(i, i, :) * \mathcal{V}(:, i, :)^T. \tag{4}$$

Definition 6 (*Tensor Multi-rank*). The multi-rank of $\mathcal{X} \in \mathcal{R}^{n_1 \times n_2 \times n_3}$ is a vector $r \in \mathcal{R}^{n_3}$ with the ith element as the rank of the ith frontal slice of $\bar{\mathcal{S}}$.

Considering above definitions, the t-SVD-based tensor nuclear norm is defined as [19]:

$$\|\mathcal{X}\|_\circledast = \sum_{i=1}^{min(n_1, n_2)} \sum_{k=1}^{n_3} \bar{\mathcal{S}}(i, i, k), \tag{5}$$

where the t-SVD-based tensor nuclear norm is a valid norm and the tightest convex relaxation to l_1-norm of the tensor multi-rank mentioned in Definition 6.

4. The proposed ELRTA

Given the vth ($v = 1, \ldots, M$) feature matrix $X^{(v)} = \left[x_1^{(v)}, x_2^{(v)}, \ldots, x_N^{(v)} \right] \in R^{d^{(v)} \times N}$, where each column is a $d^{(v)}$ dimensional sample in the vth view and N is the number of samples. The key of the Markov chain multi-view clustering method [17] is to construct a transition probability matrix. Given a weighted graph $G = (V, E)$, vertex set V represents the data points, and edge set E denotes the similarity between two connecting data points. First, a similarity matrix $S \in R^{N \times N}$: $S_{ij} = exp(-\frac{\|x_i - x_j\|_2^2}{\sigma^2})$ is established, where S_{ij} denotes the similarity between data points x_i and x_j, σ^2 denotes the standard deviation. Then the corresponding transition probability matrix $P = (D^{-1}S)$ can be calculated, where D represents the degree matrix of the weighted graph G. For the stationary distribution π satisfying $\pi = P\pi$, define diagonal matrix Π with its ith diagonal elements as the stationary distribution $\pi(i)$. Then, a Laplacian matrix for Markov chain based spectral clustering can be constructed by $L = \Pi - \frac{\Pi P + P^T \Pi}{2}$. Finally, the k-means algorithm is used to cluster the eigenvectors corresponding to the r smallest generalized eigenvalues of the generalized eigenproblem $Lu = \lambda \Pi u$. Algorithm 1 summarizes the overall scheme of spectral clustering via Markov chain by computing the transition probability matrix [13,17].

Algorithm 1: Spectral clustering via Markov chain

Input: Graph $G = (V, E)$

1: Define a random walk over G with a transition probability matrix $P = (D^{-1}S)$ such that it has a stationary distribution π satisfying $\pi = P\pi$;

2: Define diagonal matrix Π with its i-th diagonal elements as the stationary distribution $\pi(i)$;

3: Construct the matrix $L = \Pi - \frac{\Pi P + P^T \Pi}{2}$;

4: Obtain the r smallest generalized eigenvectors $u_1, \cdots, u_r$ of the generalized eigenproblem $Lu = \lambda \Pi u$;

5: Run the k-means algorithm to cluster the row vectors of U, where U is the matrix consisting of the vectors $u_1, \ldots, u_r$;

Output: Clustering results.

In fact, the probability transition matrix may have more or less noise. To handle this, RMSC divides the probability transition matrix $P^{(v)}$ for each view into two parts using the low-rank and sparse decomposition: the shared transition probability matrix Z among all views and the error matrix. Each error matrix encodes noise of the transition matrix in the vth view. RMSC assumes that the shared transition probability matrix Z tends to be low-rank, while the error matrices of all views are sparse. The RMSC model is formulated as follows:

$$\min_{Z, E^{(v)}} \|Z\|_* + \lambda \sum_{v=1}^{M} \|E^{(v)}\|_1 \ s.t. \ P^{(v)} = Z + E^{(v)}. \tag{6}$$

Considering that RMSC only learns the common information between multiple views, and ignores the importance of the difference information contained in each view to clustering. Therefore, ETLMSC explores the high-order correlations between multiple views by constructing the transition probability matrix $\{P^{(v)}\}_{v=1}^{M}$ of all views as a third-order tensor $\mathcal{P} \in \mathcal{R}^{N \times N \times M}$. In addition, since the noise in the transition probability vectors is dense, and noisy samples should be sparse, ETLMSC uses $l_{2,1}$ norm instead of l_1 norm to encode the sparsity. ETLMSC is expressed as the following model:

$$\min_{\mathcal{Z}, \mathcal{E}} \|\mathcal{Z}\|_\circledast + \lambda \|\mathcal{E}\|_{2,1} \ s.t. \ \mathcal{P} = \mathcal{Z} + \mathcal{E}, \tag{7}$$

where $\mathcal{Z}$ is the clean low-rank tensor, constrained by the t-SVD-based tensor nuclear norm in Eq. (5) and $\mathcal{E}$ denotes the error term, constrained by the $l_{2,1}$ norm. Although ETLMSC can effectively learn the principle information among multiple views, it only eliminates the sample-specific noise [7].

4.1. The proposed ELRTA model

As stated in [7], there are different types of noise except for the sample-specific noise. One typical example is hyperspectral images, which are usually contaminated by different types of noise, including Gaussian noise, salt and pepper noise, stripes, deadlines, and so on [36]. In some cases, the features of a view are more or less discriminative for clustering, and errors in more discriminative features may significantly reduce clustering performance [18]. In this case, the Markov chain-based multi-view clustering methods RMSC and ETLMSC fail. Based on ETLMSC, we add the group l_1 norm to enhance the sparsity among different views, i.e., using the l_2 norm in each view and the l_1 norm among views. Benefitting from tensor learning that is able to encode the high-order correlations among multiple views and to eliminate various errors, we propose the ELRTA model as follows:

$$\min_{\mathcal{Z}, \mathcal{E}} \|\mathcal{Z}\|_\circledast + \lambda \|\mathcal{E}\|_{2,1} + \beta \|\mathcal{E}\|_{G1} \ s.t. \ \mathcal{P} = \mathcal{Z} + \mathcal{E}. \tag{8}$$

Inspired by the study in [13], the transition probability tensor $\mathcal{P}$ is also decomposed into two parts $\mathcal{Z}$ and $\mathcal{E}$. To make the magnitude of all error matrices as consistent as possible, ELRTA calculates the group l_1 and $l_{2,1}$ norms of $\mathcal{E}$ by vectorizing all error matrices. The group l_1 and $l_{2,1}$ norms are defined as follows:

$$\|\mathcal{E}\|_{G1} = \|E\|_{G1} = \sum_{i=1}^{N \times M} \|e^{(i)}\|_2,$$

$$\|\mathcal{E}\|_{2,1} = \|E\|_{2,1} = \sum_{j=1}^{N} \|e^{(j)}\|_2, \tag{9}$$

where $E = \left[E^{(1)}; E^{(2)}; \cdots; E^{(M)} \right]$. $e^{(i)}$ and $e^{(j)}$ denote the ith row and jth column of E. The group l_1 and $l_{2,1}$ norms are calculated by the sum of the l_2 norm of all $e^{(i)}$ and $e^{(j)}$ respectively, which encourage the rows and columns of E to be sparse. Note that Eq. (9) is the equivalent form of the group l_1 norm and $l_{2,1}$ norm of tensor. Since the group l_1 norm and $l_{2,1}$ norm are defined based on matrix elements and are local constraint of the matrix, ELRTA uses the group l_1 and $l_{2,1}$ norms constraint tensor $\mathcal{E}$ is equivalent to constraint matrix E.

4.2. Optimization of ELRTA

The above optimization problem can be solved by the alternating direction method of multipliers (ADMM) [37]. The main idea of ADMM is to reformulate the original constrained problem as the unconstrained problem using an augmented Lagrangian function, and then iteratively update each variable by fixing all the other variables [38]. The augmented Lagrangian function is defined as the sum of the objective function and the penalty term under the Frobenius norm. The corresponding augmented Lagrangian function of Model (8) is:

$$\mathcal{L}(\mathcal{Z}, \mathcal{E}; \Lambda, \rho) = \|\mathcal{Z}\|_\circledast + \lambda \|\mathcal{E}\|_{2,1} + \beta \|\mathcal{E}\|_{G1} +$$
$$\langle \Lambda, \mathcal{P} - \mathcal{Z} - \mathcal{E} \rangle + \frac{\rho}{2} \|\mathcal{P} - \mathcal{Z} - \mathcal{E}\|_F^2, \tag{10}$$

where Λ represents the Lagrange multiplier, and ρ is an adaptive penalty parameter. $\langle \cdot \rangle$ denotes the standard trace inner product, i.e., $\langle A, B \rangle = Tr(A^T B)$. Following ADMM, the steps of the iterative minimization scheme are as follows:

$\mathcal{Z}$-**Subproblem:** By fixing other variables to constant, we update $\mathcal{Z}$ by solving

$$
\begin{aligned}
\mathcal{Z}_{k+1} &= \arg\min_{\mathcal{Z}} \mathcal{L}(\mathcal{Z}, \mathcal{E}_k; \Lambda_k, \rho_k) \\
&= \arg\min_{\mathcal{Z}} \|\mathcal{Z}\|_{\circledast} + \frac{\rho_k}{2} \|\mathcal{Z} - (\mathcal{P} - \mathcal{E}_k + \frac{\Lambda_k}{\rho_k})\|_F^2.
\end{aligned} \tag{11}
$$

In order to facilitate calculation, we need to rotate $\mathcal{Z}$ to obtain $\widetilde{\mathcal{Z}} \in \mathcal{R}^{N \times M \times N}$. Then, problem (11) can be solved by the tensor tubal-shrinkage operator [16]:

$$
\widetilde{\mathcal{Z}}_{k+1} = \mathcal{C}_{\frac{N}{\rho_k}}(\mathcal{F}) = \mathcal{U} * \mathcal{C}_{\frac{N}{\rho_k}}(\mathcal{S}) * \mathcal{V}^T, \tag{12}
$$

where $\mathcal{F} = \mathcal{P} - \mathcal{E}_k + \frac{\Lambda_k}{\rho_k} = \mathcal{U} * \mathcal{S} * \mathcal{V}^T$ and $\mathcal{C}_{\frac{N}{\rho_k}}(\mathcal{S}) = \mathcal{S} * \mathcal{J}$. Herein, $\mathcal{J}$ is an $N \times M \times N$ f-diagonal tensor whose diagonal element in the Fourier domain is $\mathcal{J}_f(i, i, j) = \max\{1 - [N/\rho_k(\bar{\mathcal{S}}^{(j)}(i, i))], 0\}$.

$\mathcal{E}$-**Subproblem:** The error term $\mathcal{E}$ can be updated by solving

$$
\begin{aligned}
\mathcal{E}_{k+1} &= \arg\min_{\mathcal{E}} \mathcal{L}(\mathcal{Z}_{k+1}, \mathcal{E}; \Lambda_k, \rho_k) \\
&= \arg\min_{\mathcal{E}} \lambda \|\mathcal{E}\|_{2,1} + \beta \|\mathcal{E}\|_{G1} + \frac{\rho_k}{2} \|\mathcal{E} - \mathcal{B}\|_F^2.
\end{aligned} \tag{13}
$$

where $\mathcal{B} = \mathcal{P} - \mathcal{Z}_{k+1} + \frac{\Lambda_k}{\rho_k}$.

The above optimization problem (13) can be equivalently written as the following form:

$$
\min_{\mathcal{E}} \frac{\lambda}{\rho_k} \|E\|_{2,1} + \frac{\beta}{\rho_k} \|E\|_{G1} + \frac{1}{2} \|E - B\|_F^2, \tag{14}
$$

where B is constructed by vertically concatenating each slice of tensor $\mathcal{B}$. Taking the derivative of Eq. (14) with respect to E and making it equal to zero. The optimization problem in (14) can be stated equivalently as follows [18]:

$$
\frac{\lambda}{\rho_k} \widehat{D} e^{(j)} + \frac{\beta}{\rho_k} D e^{(j)} + (e^{(j)} - B^{(j)}) = 0, \tag{15}
$$

where $B^{(j)}$ denotes the jth column of matrix B. $\widehat{D}$ is a block diagonal matrix with the jth diagonal block as $(1/(2\|e^{(j)}\|_2)) \times I$, and I is an identity matrix with a size of $N \times M$. D is a diagonal matrix and the ith diagonal element is $(1/(2\|e^{(i)}\|_2))$. $e^{(j)}$ can be obtained by:

$$
e^{(j)} = (\frac{\lambda}{\rho_k} \widehat{D} + \frac{\beta}{\rho_k} D + 1)^{-1} B^{(j)}. \tag{16}
$$

Λ **and** ρ **Subproblems:** Λ and ρ are updated by

$$
\begin{aligned}
\Lambda_{k+1} &= \Lambda_k + \rho_k(\mathcal{P} - \mathcal{Z}_{k+1} - \mathcal{E}_{k+1}); \\
\rho_{k+1} &= \min\{\gamma * \rho_k, \rho_{max}\},
\end{aligned} \tag{17}
$$

where γ facilitates the convergence speed. ρ_{max} is the maximum value of the penalty parameter ρ. Algorithm 2 summarizes the whole procedure of the proposed ELRTA. We sum the frontal slices of the transition probability tensor $\mathcal{Z}$ to calculate the transition probability matrix $Z = \sum_{i=1}^{M} \mathcal{Z}(:, :, i)$, and then put Z into the second step of Algorithm 1 as the transition probability matrix to obtain the final clustering results.

4.3. Complexity analysis

The complexity of our proposed method mainly depends on the calculation of subproblems: $\mathcal{Z}$ and $\mathcal{E}$. For subproblem $\mathcal{Z}$, the rotated $\widetilde{\mathcal{Z}}$ takes $\mathcal{O}(2N^2 M \log(N) + N^2 M^2) \approx \mathcal{O}(2N^2 M \log(N))$ per iteration, where $N \gg M$ and $\log(N) > M$. The first term $\mathcal{O}(2N^2 M \log(N))$ represents the computational cost of the FFT and inverse FFT of the third-order tensor, and the second term $\mathcal{O}(N^2 M^2)$ denotes the cost of SVD of $N \ N \times M$ matrices. For the unrotated $\mathcal{Z}$, the cost of SVD for $M \ N \times N$ matrices is $\mathcal{O}(N^3 M)$. Therefore, when the number of samples N is relatively large, the calculation after rotating $\mathcal{Z}$ will significantly reduce the

---

Algorithm 2 : Error-robust Low-Rank Tensor Approximation (ELRTA) for Multi-view Clustering

---

Input: multi-view features: $\{X^{(v)}\}$; parameters: λ, β;
Initialize: $\mathcal{P}$, $\mathcal{Z}$, Λ initialized to $\mathbf{0}$; $E^{(i)} = rand$;
1: **while** not converged **do**
2: Update $\widetilde{\mathcal{Z}}_{k+1}$ by Eq. (12);
3: **for** $j = 1, \cdots, N$ **do**
4:    Update $e^{(j)}$ by Eq. (16);
5: **end for**
6: Update Λ_{k+1} and ρ_{k+1} by Eq. (17);
7: Check the convergence condition:
8: $\|\mathcal{Z}_{k+1} - \mathcal{Z}_k\|_\infty \le tol$, $\|\mathcal{E}_{k+1} - \mathcal{E}_k\|_\infty \le tol$;
9: **end while**
10: Calculate the clean transition probability matrix $Z = \sum_{i=1}^{M} \mathcal{Z}(:, :, i)$;
11: Take Z as the transition probability matrix of Algorithm 1 to obtain the clustering results.

---

computational complexity. For subproblem $\mathcal{E}$, it needs $\mathcal{O}(2N^2 M)$ cost in each iteration. To summarize, the total complexity of our proposed ELRTA algorithm is $\mathcal{O}(K(2N^2 M \log(N))) + \mathcal{O}(N^3)$, where K is the total number of iterations, and $\mathcal{O}(N^3)$ denotes the general computational complexity of the spectral clustering.

5. Experimental results

In this section, we conduct experiments on three real-world datasets to investigate the performance of the proposed ELRTA. All experiments are implemented in MATLAB 2016a on workstation with 3.50 GHz CPU and 16 GB RAM.

5.1. Experimental settings

5.1.1. Datasets
We select three widely used datasets to evaluate the clustering performance.

- The BBC4view dataset consists of 685 images of 5 object categories, each with 137 images and associated with 4 views;
- The BBCSport dataset contains files from the BBC Sports website, which corresponds to sports news in 5 subject areas (2 views);
- The Still-DB dataset is an action image dataset that contains 467 images of 6 object categories and associated with 2 views.

5.1.2. Compared methods
We compare the proposed ELRTA with three single-view and twelve multi-view state-of-the-art clustering methods: SSC**best** [6]: the best results of sparse subspace clustering method using the l_1 norm to learn a representation matrix; LRR**best** [7]: the best results of the low-rank representation method using the nuclear norm to learn a representation matrix; RSS**best** [39]: the best results of the robust subspace segmentation method using the least square regression; RMSC [17]: multi-view clustering using the low-rank and sparse matrix decomposition to learn the shared transition probability matrix as the input to the standard Markov chain; DiMSC [31]: multi-view subspace clustering exploiting the diversity among different views; LT-MSC [15]: tensor-based multi-view clustering using the low-rank tensor constraint to learn a representation tensor; MVCC [22]: multi-view concept clustering based on the concept factorization with local manifold regularization, which drives a common

Table 1
Clustering results (mean ± standard deviation) on *BBC4view*.

Data	Type	Method	ACC	NMI	AR	F-score	Precision	Recall
BBC4view	SVC	SSC$_{best}$ [6]	0.660 ± 0.002	0.494 ± 0.005	0.470 ± 0.001	0.599 ± 0.001	0.578 ± 0.001	0.622 ± 0.001
		LRR$_{best}$ [7]	0.802 ± 0.000	0.568 ± 0.000	0.621 ± 0.000	0.712 ± 0.000	0.697 ± 0.000	0.727 ± 0.000
		RSS$_{best}$ [39]	0.837 ± 0.000	0.621 ± 0.000	0.665 ± 0.000	0.747 ± 0.000	0.720 ± 0.000	0.775 ± 0.000
	MVC	RMSC [17]	0.775 ± 0.003	0.616 ± 0.004	0.560 ± 0.002	0.656 ± 0.002	0.703 ± 0.003	0.616 ± 0.001
		DiMSC [31]	0.892 ± 0.001	0.728 ± 0.002	0.752 ± 0.002	0.810 ± 0.002	0.811 ± 0.002	0.810 ± 0.002
		LT-MSC [15]	0.591 ± 0.000	0.442 ± 0.005	0.400 ± 0.001	0.546 ± 0.000	0.525 ± 0.000	0.570 ± 0.001
		MVCC [22]	0.745 ± 0.001	0.587 ± 0.001	0.550 ± 0.000	0.656 ± 0.001	0.654 ± 0.001	0.658 ± 0.000
		MLAN [28]	0.853 ± 0.007	0.698 ± 0.010	0.716 ± 0.005	0.783 ± 0.004	0.776 ± 0.003	0.790 ± 0.004
		ECMSC [32]	0.308 ± 0.028	0.047 ± 0.009	0.008 ± 0.018	0.322 ± 0.017	0.239 ± 0.009	0.497 ± 0.064
		t-SVD [12]	0.858 ± 0.001	0.685 ± 0.002	0.725 ± 0.002	0.789 ± 0.001	0.800 ± 0.001	0.778 ± 0.002
		GMC [29]	0.693 ± 0.000	0.563 ± 0.000	0.479 ± 0.000	0.633 ± 0.000	0.501 ± 0.000	0.860 ± 0.000
		LMSC [33]	0.883 ± 0.000	0.699 ± 0.000	0.746 ± 0.000	0.806 ± 0.000	0.797 ± 0.000	0.816 ± 0.000
		GLTA [14]	0.910 ± 0.000	0.771 ± 0.000	0.810 ± 0.000	0.854 ± 0.000	0.864 ± 0.000	0.845 ± 0.000
		GLSR [4]	0.917 ± 0.000	0.780 ± 0.000	0.811 ± 0.000	0.856 ± 0.000	0.835 ± 0.000	0.878 ± 0.000
		ETLMSC [13]	0.745 ± 0.057	0.661 ± 0.039	0.562 ± 0.047	0.660 ± 0.038	0.693 ± 0.032	0.630 ± 0.047
	Proposed	ELRTA	**0.938 ± 0.070**	**0.936 ± 0.047**	**0.915 ± 0.073**	**0.935 ± 0.087**	**0.943 ± 0.088**	**0.927 ± 0.086**

consensus representation for multiple views; MLAN [28]: multi-view clustering with adaptive neighbors; ECMSC [32]: multi-view clustering exploiting supplementary information among different views; t-SVD [12]: tensor-based multi-view subspace clustering via the tensor multi-rank constraint; GMC [29]: graph-based multi-view clustering method; LMSC [33]: multi-view clustering via latent representation learning; GLTA [14]: multi-view clustering via simultaneously learning the graph regularized low-rank tensor representation and affinity matrix; GLSR [4]: multi-view clustering with the graph-regularized least squares regression; ETLMSC [13]: multi-view clustering based on the Markov chain.

5.1.3. Evaluation metrics

We adopt six widely used evaluation metrics to quantitatively evaluate the performance of all clustering methods. The metrics include ACC (accuracy), NMI (normalized mutual information), Precision, Recall, F-score, AR (adjusted rand index). For all clustering methods, larger values of the metrics are expected to achieve better clustering performance [40].

5.2. Results

The clustering results of all datasets are shown in Tables 1, 2, and 3. The best clustering results are highlighted in boldface and the second best clustering results are underlined. Due to randomness, we run all algorithms 10 times and report the mean values with their standard deviations.

For the BBC4view dataset, the clustering results are shown in Table 1. Our method, GLSR and GLTA achieved excellent clustering results, where GLTA learned the affinity matrix and the low-rank tensor representation. However, the impact of various noise in the dataset is also critical. Based on this, the proposed ELRTA mainly focuses on eliminating various types of errors Among all methods, the proposed ELRTA has yielded the best performance on the BBC4view dataset. Our method achieves significant improvement of approximately 2.1%, 15.6%, 10.4%, 7.9%, 10.8% and 4.9% over the second-best GLSR method with respect to the six evaluation metrics.

Tables 2 and 3 recorded the clustering results of all compared methods on the BBCSport and Still-DB datasets. In Table 2, the ELRTA, ETLMSC, and GLTA methods show clustering advantages on the BBCSport dataset. In Table 3, the performance of GLTA on the Still-DB dataset is poor, while ELRTA and ETLMSC still have better performance, which also proves that the noise removal technology is stable and effective. It is worth noting that, as the most competitive method, ETLMSC is also based on the Markov chain method for low-rank tensor and noise learning. However,

ETLMSC only focuses on eliminating sample-specific noise. As shown in Table 2, for all metrics, ELRTA improves the results by approximately 3.5%, 1.2%, 4.1%, 3.1%, 3.1% and 3% over ETLMSC, while the improvement of ELRTA is 2.4%, 1.7%, 1.9%, 1.4%, 1.7% and 1.2% in Table 3. This means that ETLMSC is not robust to various noise in real applications. In summary, ELRTA has the best clustering performance among all methods on the BBCSport and Still-DB dataset.

It can be found that most multi-view clustering methods have better clustering performance than all single-view clustering methods including SSC$_{best}$, LRR$_{best}$, and RSS$_{best}$. For example, the proposed ELRTA obtained at least 20% improvement over all single-view clustering methods except for the ACC metric. The advantage of the proposed ELRTA on the Still-DB dataset is more prominent. The main reason is that the multi-view clustering methods can simultaneously use the complementary information among multiple views. This conclusion has been proven in many literatures, including GLTA [14] and GLSR [4].

The multi-view clustering methods can be roughly divided into two groups: matrix-based methods and tensor-based methods. The experimental results show that the tensor-based methods explore the high-order correlation information, resulting in clustering performance better than that of the matrix-based methods. For example, our method improves each metric by at least 10% compared with RMSC and GMC. Among them, GMC uses the original multi-view features to learn the similarity matrix, while the original data is usually corrupted by different noise and outliers.

Furthermore, the proposed ELRTA achieves an average improvement of at least 10% compared to the tensor-based methods LT-MSC and t-SVD. This shows that our method can make full use of the relationship between spectral clustering and Markov chain to obtain effective clustering information.

In summary, the proposed ELRTA achieves the best performance over all competing methods. Furthermore, ELRTA is superior to the Markov chain-based methods RMSC and ETLMSC, which are most relevant to our method, indicating that the ELRTA method is effective by introducing the group l_1 norm to remove various types of noise. To make the magnitude values consistent, the columns of the error matrices are arranged vertically in the calculation process, which is also necessary to obtain superior clustering results.

5.3. Model analysis

(1) Numerical Convergence: This subsection studies the numerical convergence of the proposed ELRTA model. Fig. 2 shows
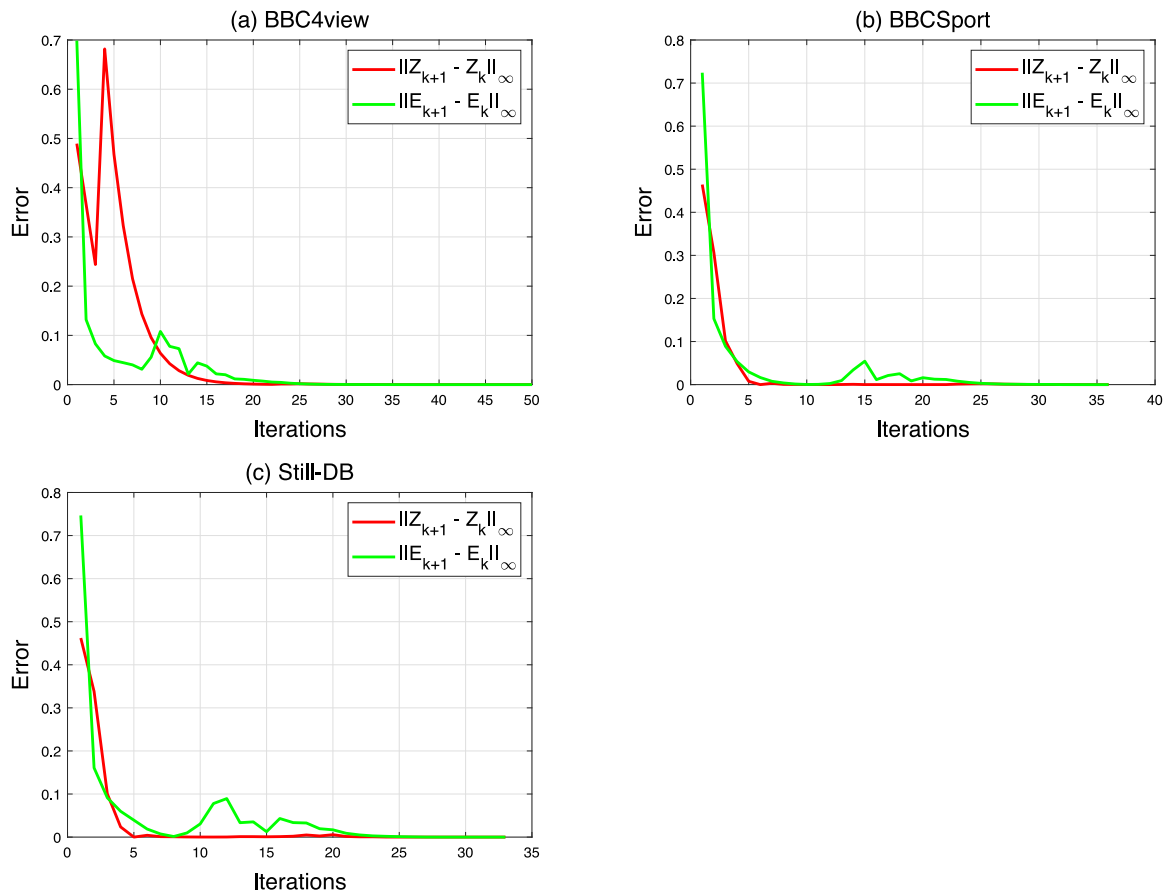
Fig. 2. Errors versus iterations on (a) BBC4view, (b) BBCSport, and (c) Still-DB databases.

Table 2
Clustering results (mean $\pm$ standard deviation) on *BBCSport*.

Data	Type	Method	ACC	NMI	AR	F-score	Precision	Recall
		SSC**best** [6]	0.627 $\pm$ 0.003	0.534 $\pm$ 0.008	0.364 $\pm$ 0.007	0.565 $\pm$ 0.005	0.427 $\pm$ 0.004	0.834 $\pm$ 0.004
	SVC	LRR**best** [7]	0.836 $\pm$ 0.001	0.698 $\pm$ 0.002	0.705 $\pm$ 0.001	0.776 $\pm$ 0.001	0.768 $\pm$ 0.001	0.784 $\pm$ 0.001
		RSS**best** [39]	0.878 $\pm$ 0.000	0.714 $\pm$ 0.000	0.717 $\pm$ 0.000	0.784 $\pm$ 0.000	0.787 $\pm$ 0.000	0.782 $\pm$ 0.000
		RMSC [17]	0.826 $\pm$ 0.001	0.666 $\pm$ 0.001	0.637 $\pm$ 0.001	0.719 $\pm$ 0.001	0.766 $\pm$ 0.001	0.677 $\pm$ 0.001
		DiMSC [31]	0.922 $\pm$ 0.000	0.785 $\pm$ 0.000	0.813 $\pm$ 0.000	0.858 $\pm$ 0.000	0.846 $\pm$ 0.000	0.872 $\pm$ 0.000
		LT-MSC [15]	0.460 $\pm$ 0.046	0.222 $\pm$ 0.028	0.167 $\pm$ 0.043	0.428 $\pm$ 0.014	0.328 $\pm$ 0.028	0.629 $\pm$ 0.053
		MVCC [22]	0.928 $\pm$ 0.000	0.816 $\pm$ 0.000	0.831 $\pm$ 0.000	0.870 $\pm$ 0.000	0.889 $\pm$ 0.000	0.853 $\pm$ 0.000
		MLAN [28]	0.721 $\pm$ 0.000	0.779 $\pm$ 0.000	0.591 $\pm$ 0.000	0.714 $\pm$ 0.000	0.567 $\pm$ 0.000	0.962 $\pm$ 0.000
BBCSport	MVC	ECMSC [32]	0.285 $\pm$ 0.014	0.027 $\pm$ 0.013	0.009 $\pm$ 0.011	0.267 $\pm$ 0.020	0.244 $\pm$ 0.007	0.297 $\pm$ 0.045
		t-SVD [12]	0.879 $\pm$ 0.000	0.765 $\pm$ 0.000	0.784 $\pm$ 0.000	0.834 $\pm$ 0.000	0.863 $\pm$ 0.000	0.807 $\pm$ 0.000
		GMC [29]	0.807 $\pm$ 0.000	0.760 $\pm$ 0.000	0.722 $\pm$ 0.000	0.794 $\pm$ 0.000	0.727 $\pm$ 0.000	0.875 $\pm$ 0.000
		LMSC [33]	0.847 $\pm$ 0.003	0.739 $\pm$ 0.001	0.749 $\pm$ 0.001	0.810 $\pm$ 0.001	0.799 $\pm$ 0.001	0.822 $\pm$ 0.001
		GLTA [14]	0.939 $\pm$ 0.000	0.825 $\pm$ 0.000	0.849 $\pm$ 0.000	0.885 $\pm$ 0.000	0.890 $\pm$ 0.000	0.880 $\pm$ 0.000
		GLSR [4]	0.873 $\pm$ 0.000	0.781 $\pm$ 0.000	0.803 $\pm$ 0.000	0.851 $\pm$ 0.000	0.837 $\pm$ 0.000	0.865 $\pm$ 0.000
		ETLMSC [13]	0.959 $\pm$ 0.086	0.972 $\pm$ 0.058	0.949 $\pm$ 0.107	0.961 $\pm$ 0.081	0.963 $\pm$ 0.078	0.960 $\pm$ 0.085
	Proposed	ELRTA	**0.994 $\pm$ 0.053**	**0.984 $\pm$ 0.030**	**0.990 $\pm$ 0.052**	**0.992 $\pm$ 0.040**	**0.994 $\pm$ 0.034**	**0.990 $\pm$ 0.045**

the iterative error curves on the BBC4view, BBCSport, and Still-DB datasets. The horizontal ordinate represents the number of iterations while the vertical coordinate denotes the errors. It can be seen that the error of the proposed ELRTA method are close to 0 after 25 iterations. This means that our algorithm can converge steadily after a certain number of iterations. In the optimization process, the error curves gradually decrease as the number of iterations increases and tend to stabilize after several fluctuations. The above conclusions show that our algorithm has strong numerical convergence.

(2) Parameter Selection: We use all datasets to analyze the parameter sensitivity. λ represents the effect of sample-specific noise. β represents the impact of various types of noise and their combinations. We tune two parameters λ and β from [0.001, 0.005, 0.01, 0.05, 0.1, 0.2, 0.4, 0.5] and [0.01, 0.05, 0.1, 0.5, 1, 10], respectively. Fig. 3 shows the ACC results obtained by different combinations of λ and β. For the BBCSport dataset, most combinations can achieve relatively high ACC values, which shows the partial stability of the proposed ELRTA method. Overall, we find that the optimal values of λ and β are within the range of [0.009, 0.005] and [0.003, 0.01], respectively. Among the optimal values, the value of parameter β is relatively large, which indicates that various types of noise and their combinations have a greater impact on the datasets, highlighting the necessity of our proposed ELRTA method.

Table 3
Clustering results (mean ± standard deviation) on *Still-DB*.

Data	Type	Method	ACC	NMI	AR	F-score	Precision	Recall
Still-DB	SVC	SSC$_{best}$ [6]	0.328 ± 0.002	0.138 ± 0.003	0.087 ± 0.001	0.295 ± 0.001	0.222 ± 0.001	0.440 ± 0.007
		LRR$_{best}$ [7]	0.306 ± 0.004	0.109 ± 0.003	0.061 ± 0.004	0.219 ± 0.000	0.223 ± 0.003	0.212 ± 0.004
		RSS$_{best}$ [39]	0.342 ± 0.000	0.122 ± 0.000	0.088 ± 0.000	0.258 ± 0.000	0.237 ± 0.000	0.282 ± 0.000
	MVC	RMSC [17]	0.305 ± 0.010	0.089 ± 0.009	0.073 ± 0.011	0.221 ± 0.002	0.231 ± 0.004	0.219 ± 0.002
		DiMSC [31]	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 [15]	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 [22]	0.251 ± 0.000	0.078 ± 0.000	0.005 ± 0.000	0.278 ± 0.000	0.174 ± 0.000	0.702 ± 0.000
		MLAN [28]	0.349 ± 0.000	0.138 ± 0.000	0.098 ± 0.000	0.272 ± 0.000	0.242 ± 0.000	0.310 ± 0.000
		ECMSC [32]	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 [12]	0.347 ± 0.010	0.130 ± 0.004	0.088 ± 0.003	0.255 ± 0.004	0.239 ± 0.002	0.273 ± 0.006
		GMC [29]	0.251 ± 0.000	0.078 ± 0.000	0.005 ± 0.000	0.278 ± 0.000	0.174 ± 0.000	0.701 ± 0.000
		LMSC [33]	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 [14]	0.366 ± 0.007	0.126 ± 0.005	0.102 ± 0.005	0.262 ± 0.003	0.251 ± 0.004	0.275 ± 0.003
		GLSR [4]	0.368 ± 0.003	0.142 ± 0.002	0.105 ± 0.003	0.279 ± 0.003	0.246 ± 0.002	0.321 ± 0.006
		ETLMSC [13]	0.604 ± 0.043	0.520 ± 0.015	0.423 ± 0.029	0.523 ± 0.024	0.518 ± 0.022	0.528 ± 0.027
	Proposed	ELRTA	**0.628 ± 0.023**	**0.537 ± 0.039**	**0.442 ± 0.022**	**0.537 ± 0.018**	**0.535 ± 0.018**	**0.540 ± 0.019**

Table 4
Average running time (in seconds) on all datasets.

Method	DiMSC	LT-MSC	ECMSC	t-SVD	GLSR	LMSC	GLTA	ETLMSC	ELRTA
BBC4view	207.21	335.51	1238.70	97.99	68.73	180.38	204.52	48.06	1101.51
BBCSport	38.15	77.23	266.86	19.59	24.92	59.28	54.63	4.21	195.09
Still-DB	6.23	18.68	8.82	6.12	9.03	12.30	17.62	2.29	187.21
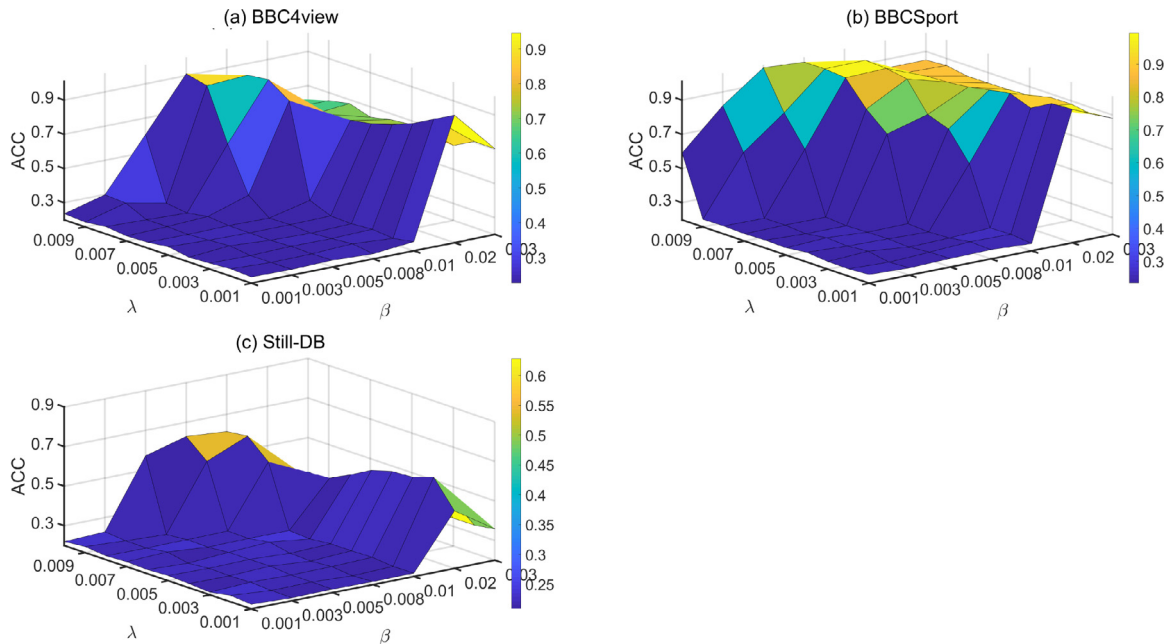
Fig. 3. Parameter selection with respect to λ and β on (a) BBC4view, (b) BBCSport, and (c) Still-DB databases.

(3) Running Time: Table 4 shows the average running time (in seconds) of the proposed ELRTA method and eight multi-view clustering methods on all datasets. We can observe that the recently proposed ETLMSC requires the least running time on all datasets, but its clustering performance is worse than that of the proposed ELRTA method, especially on the BBC4view dataset (as shown in Table 1). Since ELRTA imposes two constraints on the error term to remove different combinations of noise, ELRTA costs more running times than the other methods. In summary, although the proposed ELRTA is time-consuming, it significantly improves the clustering performance.

6. Conclusions and future work

In this paper, we proposed a tensor model based on Markov chain: error-robust low-rank tensor approximation for multi-view clustering (ELRTA). ELRTA constructs the transition probability matrices into a third-order tensor and decomposes it into a noise-free transition probability tensor and an error term. Our method has two main advantages: (1) The noise of the transition probability tensor is filtered out by decomposition, and the t-SVD-based tensor nuclear norm constraint is used to perform low-rank approximation to capture the high-order correlations among multiple views. Compared with matrix-based clustering

methods such as RMSC [4,17,33], our ELRTA fully uses the relevant information among views; (2) The clustering performance is effectively improved by applying the group l_1 and $l_{2,1}$ norms on the error term to eliminate the various types of errors. In addition, an iterative algorithm is proposed. Experimental results on three real-world datasets show that ELRTA has superior performance over the existing multi-view clustering methods. In future work, considering the limitations of large datasets, we will focus on developing fast and scalable multi-view clustering methods.

CRediT authorship contribution statement

Shuqin Wang: Writing - original draft, Writing code. **Yongyong Chen:** Writing - review & editing. **Yi Jin:** Formal analysis. **Yigang Cen:** Writing - review & editing, Funding acquisition. **Yidong Li:** Investigation. **Linna Zhang:** Resources.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

References

[1] Y. Chen, Z. Yi, Locality-constrained least squares regression for subspace clustering, Knowl.-Based Syst. 163 (2019) 51–56.

[2] G. Zhang, Y. Zhou, X. He, C. Wang, D. Huang, One-step kernel multi-view subspace clustering, Knowl.-Based Syst. 189 (2020) 105–126.

[3] L. Wang, Z. Ding, Z. Tao, Y. Liu, Y. Fu, Generative multi-view human action recognition, in: Proc. IEEE Int. Conf. Comput. Vis., 2019, pp. 6212–6221.

[4] Y. Chen, S. Wang, F. Zheng, Y. Cen, Graph-regularized least squares regression for multi-view subspace clustering, Knowl.-Based Syst. 194 (2020) 105482.

[5] Z. Tao, H. Liu, H. Fu, Y. Fu, Multi-view saliency-guided clustering for image cosegmentation, IEEE Trans. Image Process. 28 (9) (2019) 4634–4645.

[6] E. Elhamifar, R. Vidal, Sparse subspace clustering: Algorithm, theory, and applications, IEEE Trans. Pattern Anal. Mach. Intell. 35 (11) (2013) 2765–2781.

[7] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, Y. Ma, Robust recovery of subspace structures by low-rank representation, IEEE Trans. Pattern Anal. Mach. Intell. 35 (1) (2013) 171–184.

[8] K. Li, S. Li, Z. Ding, Latent discriminant subspace representations for multi-view outlier detection, in: Proc. AAAI Conf. Artif. Intell., 2018, pp. 3522–3529.

[9] M. Brbic, I. Kopriva, Multi-view low-rank sparse subspace clustering, Pattern Recognit. 73 (2018) 247–258.

[10] H. Wang, Y. Cen, Z. He, R. Zhao, F. Zhang, Robust generalized low-rank decomposition of multi-matrices for image recovery, IEEE Trans. Multimedia 9 (5) (2017) 969–983.

[11] Q. Zheng, J. Zhu, Z. Li, S. Pang, J. Wang, Y. Li, Feature concatenation multi-view subspace clustering, Neurocomputing 379 (2020) 89–102.

[12] Y. Xie, D. Tao, W. Zhang, Y. Liu, L. Zhang, Y. Qu, On unifying multi-view self-representations for clustering by tensor multi-rank minimization, Int. J. Comput. Vis. 126 (11) (2018) 1157–1179.

[13] J. Wu, Z. Lin, H. Zha, Essential tensor learning for multi-view spectral clustering, IEEE Trans. Image Process. 28 (12) (2019) 5910–5922.

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

[15] C. Zhang, H. Fu, S. Liu, G. Liu, X. Cao, Low-rank tensor constrained multiview subspace clustering, in: Proc. IEEE Int. Conf. Comput. Vis., 2015, pp. 1582–1590.

[16] Y. Xie, W. Zhang, Y. Qu, Hyper-Laplacian regularized multilinear multiview self-representations for clustering and semisupervised learning, IEEE Trans. Cybern. 93 (2) (2020) 572–586.

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

[18] M. Najafi, L. He, P.S. Yu, Error-robust multi-view clustering, in: Proc. IEEE BigData, 2017, pp. 736–745.

[19] Y. Chen, X. Xiao, Y. Zhou, Multi-view subspace clustering via simultaneously learning the representation tensor and affinity matrix, Pattern Recognit. 106 (2020) 107441.

[20] A. Li, Z. Miao, Y. Cen, X. Zhang, L. Zhang, S. Chen, Abnormal event detection in surveillance videos based on low-rank and compact coefficient dictionary learning, Pattern Recognit. 108 (12) (2020) 107355.

[21] J. Liu, C. Wang, J. Gao, J. Han, Multi-view clustering via joint non negative matrix factorization, in: Proc. IEEE Int. Conf. Data Min., 2013, pp. 252–260.

[22] H. Wang, Y. Yang, T. Li, Multi-view clustering via concept factorization with local manifold regularization, in: Proc. IEEE Int. Conf. Data Min., 2016, pp. 1245–1250.

[23] S. Huang, Z. Kang, Z. Xu, Auto-weighted multi-view clustering via deep matrix decomposition, Pattern Recognit. 97 (2020) 107015.

[24] W. Tang, Z. Lu, I.S. Dhillon, Clustering with Multiple Graphs, in: Proc. IEEE Int. Conf. Data Min., 2009, pp. 1016–1021.

[25] Y. Li, F. Nie, H. Huang, J. Huang, Large-scale multi-view spectral clustering via bipartite graph, in: Proc. AAAI Conf. Artif. Intell., 2015, pp. 2750–2756.

[26] K. Zhan, C. Zhang, J. Guan, Graph learning for multiview clustering, IEEE Trans. Cybern. 48 (2017) 2887–2895.

[27] F. Nie, J. Li, X. Li, et al., Parameter-free auto-weighted multiple graph learning: a framework for multiview clustering and semi-supervised classification, in: Proc. Int. Joint Conf. Artif. Intell., 2016, pp. 1881–1887.

[28] F. Nie, G. Cai, J. Li, X. Li, Auto-weighted multi-view learning for image clustering and semi-supervised classification, IEEE Trans. Image Process. 27 (3) (2018) 1501–1511.

[29] H. Wang, Y. Yang, B. Liu, GMC: Graph-based multi-view clustering, IEEE Trans. Knowl. Data Eng. 32 (6) (2020) 1116–1129.

[30] C. Lu, H. Min, Z. Zhao, Robust and efficient subspace segmentation via least squares regression, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2012, pp. 347–360.

[31] X. Cao, C. Zhang, H. Fu, S. Liu, H. Zhang, Diversity-induced multi-view subspace clustering, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2015, pp. 586–594.

[32] X. Wang, X. Guo, Z. Lei, C. Zhang, S.Z. Li, Exclusivity-consistency regularized multi-view subspace clustering, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2017, pp. 923–931.

[33] C. Zhang, H. Fu, Q. Hu, X. Cao, Y. Xie, D. Tao, D. Xu, Generalized latent multi-view subspace clustering, IEEE Trans. Pattern Anal. Machine Intell. 42 (1) (2020) 86–99.

[34] S. Luo, C. Zhang, W. Zhang, X. Cao, Consistent and specific multi-view subspace clustering, in: Proc. AAAI Conf. Artif. Intell., 2018, pp. 3730–3737.

[35] Y. Chen, X. Xiao, Y. Zhou, Low-rank quaternion approximation for color image processing, IEEE Trans. Image Process. 29 (2019) 1426–1439.

[36] Y. Chen, Y. Guo, Y. Wang, D. Wang, C. Peng, G. He, Denoising of hyperspectral images using nonconvex low rank matrix approximation, IEEE Trans. Geosci. Remote Sens. 55 (9) (2017) 5366–5380.

[37] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein, et al., Distributed optimization and statistical learning via the alternating direction method of multipliers, Found. Trends® Mach. Learn. 3 (1) (2011) 1–122.

[38] S. Wang, Y. Wang, Y. Chen, P. Pan, Z. Sun, G. He, Robust PCA using matrix factorization for background/foreground separation, IEEE Access 6 (2018) 18945–18953.

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

[40] Y. Chen, X. Xiao, Y. Zhou, Jointly learning kernel representation tensor and affinity matrix for multi-view clustering, IEEE Trans. Multimedia 22 (8) (2020) 1985–1997.