

RESEARCH ARTICLE

Robust Graph Regularized Nonnegative Matrix Factorization

QI HUANG¹, GUODAO ZHANG², XUESONG YIN¹, AND YIGANG WANG²

¹School of Biological and Chemical Engineering, Zhejiang University of Science and Technology, Hangzhou 310023, China

²Department of Digital Media Technology, Hangzhou Dianzi University, Hangzhou 310018, China

Corresponding author: Xuesong Yin (yinxs@hdu.edu.cn)


This research was funded by Public-welfare Technology Application Research of Zhejiang under Grants LGG22F020032 and Zhejiang Provincial Science and Technology Program in China under Grant 2021C03137.

ABSTRACT Nonnegative Matrix Factorization (NMF) has become a popular technique for dimensionality reduction, and been widely used in machine learning, computer vision, and data mining. Existing unsupervised NMF methods impose the intrinsic geometric constraint on the encoding matrix, which only indirectly affects the base matrix. Moreover, they ignore the global structure of the data space. To address these issues, in this paper we propose a novel unsupervised NMF learning framework, called Robust Graph regularized Nonnegative Matrix Factorization (RGNMF). RGNMF constructs a sparse graph imposed on the basis matrix to catch the global structure and preserve the discriminative information. And it models the local structure by building a k -NN graph constrained on the encoding matrix, which gains the compact representation. Consequently, RGNMF not only respects the global structure, but also depicts the local structure. In addition, it employs such a $L_{2,1}$ -norm cost function to decompose the basis matrix and encoding matrix that its robustness can be improved. Further, it imposes the $L_{2,1}$ -norm constraint on the basis matrix to choose the discriminative feature. Hence, RGNMF can gain the robust discriminative representation by combining structure learning and $L_{2,1}$ -norm constraints imposed on the basis matrix and encoding matrix. Extensive experiments on real-world problems demonstrate that RGNMF achieves better clustering results than the state-of-the-art approaches.

INDEX TERMS Nonnegative matrix factorization, manifold learning, sparse representation, global structure, local structure, data representation.

I. INTRODUCTION

As a popular technique for data representation, nonnegative matrix factorization (NMF) has been successfully applied in computational intelligence [1], [2], machine learning [3], [4] and data mining [5], [6], [7]. Its power lies in its ability to give meaningful decompositions of data into two nonnegative matrices. Compared with other matrix factorization techniques, NMF can achieve relatively good performance. Moreover, the basis vectors obtained by NMF provide interpretability and clear physical meaning from the nonnegative view. Hence, more and more researchers have paid close attention to NMF. Specifically, given a nonnegative matrix $Y = [y_1, y_2, \dots, y_n] \in \mathbb{R}^{m \times n}$ consisting of n samples, NMF

The associate editor coordinating the review of this manuscript and approving it for publication was Berdakh Abibullaev .

focus on finding two low-dimensional nonnegative matrices A and X so that the product of A and X approximates Y . The objective function of NMF with the square of the Frobenius norm is defined as follows [8]:

$$\min_{A \geq 0, X \geq 0} \|Y - AX\|_F^2, \quad (1)$$

where A represents a basis matrix with m rows and l columns. $X \in \mathbb{R}^{l \times n}$ is usually a coefficient matrix (or an encoding matrix) and also considered as the new representation of data with respect to the new basis A . By alternately optimizing A and X in (1), one can gain their optimal solution. Although NMF can provide solid mathematical theory and encouraging performance, it suffers from three limitations in real-world applications. First of all, it ignores the intrinsic geometric structure. It has been shown that the intrinsic geometry of

the data distribution is essentially useful for many practical problems [9], [10], [11]. secondly, it fails to apply the supervised information of data. The supervised information can be used to learn more discriminative features [12], [13]. Finally, it is sensitive to noise and outliers [14]. To address these problems, recently, a number of improved NMF algorithms have been proposed and successfully applied in various fields. According to whether the supervised information is used, we divide existing NMF algorithms into three categories: supervised, semi-supervised and unsupervised.

Supervised NMF methods make use of labeled data to explore the discriminative representation that enforces the separability between classes and promotes the performance of NMF to some extent. Zafeiriou *et al.* [15] introduced well-known linear discriminant criterion into NMF by using the data labels, which increased the separability of samples from different classes and enforced the compactness of samples from the same class. Guan *et al.* [16] employed the label information of data to formulate between-class k-nearest neighbor (k-NN) and within-class k-NN scatters and then incorporated them into NMF. Similar to [16], An *et al.* [17] expanded the local regions of the between-class neighborhood and reduced the local regions of the within-class neighborhood to extract discriminative features of data. Li *et al.* [18] exploited the data labels to respectively construct a neighbor graph and a penalty graph for capturing both the within-class compactness and the between-class distinctness of data. Nikitidis *et al.* [19] sought the discriminant projection by incorporating subclass-based constraints into the loss function of NMF. Generally, it is difficult to obtain the class labels of the whole data, but relatively easy to gain a small number of data labels or pairwise constraints. Liu *et al.* [20] proposed a semi-supervised NMF, which explicitly imposes the label information on the encoding matrix as additional hard constraints to improve the discriminative ability of the representation. Different from [20] that uses a few data labels to guide matrix decomposition, Zhang *et al.* [21] integrated pairwise constraints in the form of must-link and cannot-link in the objective function of NMF to find an appropriate indicator matrix. Wang *et al.* [22] propagated both cannot-link and must-link constraints to unlabeled samples for constructing a new data weight matrix. Li *et al.* [23] applied the labeled and unlabeled data to gain the discriminative representations by investigating the block-diagonal structure learned by the label information. Jiao *et al.* [64] proposed a novel semi-supervised NMF method by merging the hypergraph regularizer and class label into NMF.

Generally, it is very expensive to gain the class label of data. For example, In the diagnosis of tuberculosis, it is difficult for a doctor to judge whether the tumor is negative or positive according to a medical image of the lung. Therefore, unsupervised NMFs are more suitable for solving real-world problems than supervised and semi-supervised NMF approaches [24]. Dirichlet matrix factorization [25] exploits matrix factorization to enhance the prediction and the reliability of recommender system, so as to achieve

a better performance for recommender system. Sparseness constraint-based NMF [26] incorporates sparseness constraints into NMF to explicitly control the sparsity of the factor matrices. In various applications, data are usually contaminated by noise and outliers. To address the issue, Kong *et al.* [14] minimized the $L_{2,1}$ -norm loss to factorize the original matrix into the two matrices which has been shown to handle noise and outliers. Studies [10], [28] have shown that manifold learning technique can be applied to improve the performance of NMF. Robust manifold NMF (RMNMF) [27] depicts the local structure and relaxes the nonnegativity of the basis matrix for seeking the proper factorization. NMF with Adaptive Neighbors (NMFAN) [29] alternately seeks the similar matrix, the basis matrix and the encoding matrix by constructing an adaptive k-NN graph. Low-rank matrix factorization [30] exploits the k-NN graph regularization and low-rank factorization to find these two factor matrices. General subspace constrained NMF [31] regularizes NMF with various subspace constraints formulated into a certain form. Graph regularized low-rank NMF (GNLMF) [32] incorporates the local structure into the nonnegative low-rank matrix factorization framework to get an effective low-rank data representation. Zhang *et al.* [33] exploited the manifold regularization and matrix factorization to simultaneously solve the affinity matrix and the encoding matrix. Yi *et al.* [34] constructed a sparse graph and a k-NN graph and merged them into NMF for seeking two factor matrices. Peng *et al.* [55] proposed a novel robust log-norm regularized sparse NMF method (RLS-NMF). RLS-NMF formulates $L_{2,\log}$ -shrink operator as the solution to the $L_{2,\log}$ -(pseudo) norm, which makes the data with noise subtraction nonnegative. With $L_{2,\log}$ -shrink operator, it develops multiplicative updating rules to gain a robust parts-based representation by sparser solutions. Yu *et al.* [65] exploited the correntropy measure in the loss function and constructed a hypergraph to preserve the high-order geometric information of the data, which improved the performance and robustness of NMF.

The above-mentioned NMF-based methods are linear, and effectively decompose the data located in the linear space. Recently, with the successful application of deep learning, many researchers have combined deep learning technique with NMF to handle the matrix factorization problem of nonlinear data. Trigeorgis *et al.* [56] proposed a deep semi-NMF via graph regularization technique, which can learn such hidden representation and interpret clustering according to different unknown attributes of a given data set. Deep NMF based on autoencoders (DNMF) [57] is applied to improve nonlinear data-driven fault detection by combining deep autoencoders into NMF. Zhao *et al.* [58] proposed a deep NMF framework based on underlying basis image learning, which is used to extract features that reflect the depth positioning features of samples. Sparse dual graph-regularized DNMF [59] respects the geometric structures of feature manifold and data manifold to seek the data information of hidden layers so that it learns a sparse and compact representation. Semi-supervised graph regularized DNMF (SGDNMF) [60]

employs a small number of labels to learn a representation from the hidden layer of the deep network. Furthermore, SGD-NMF introduces bi-orthogonal constraints on two factor matrices into NMF framework to make the solution unique. By revealing the hierarchical semantics of the input data, Huang *et al.* [61] proposed a new collaborative deep matrix factorization framework to seek the hidden representation of different attributes.

Obviously, supervised and semi-supervised NMF approaches can depict the global structure with the labeled data. Because of the lack of the supervised information, however, unsupervised ones pay more attention to the local structure of data or the sparsity of the factor matrix to gain the compact representation. Consequently, they fail to respect the global structure so that the discriminative power of the representation is limited to some extent. In addition, many of them constrain the geometrical information on the encoding matrix, which can find a proper encoding matrix. For many tasks, however, especially feature extraction, clustering and classification, the projection matrix constructed by the base matrix is used to project the original data into various subspaces [35], [36], [37]. Clearly, if the geometric constraint is imposed on the encoding matrix, the influence of data geometry on the base matrix is indirect. Recent studies have shown that the sparse graph constructed based on sparse representation technique can capture the discriminative information [38], [39]. For example, by constructing sparse graph, sparsity preserving projections (SPP) [40] not only naturally preserves the global structure of data, but also contains discriminative information. It has been shown that the performance of the learning model can be markedly enhanced if the discriminative information is exploited and the geometric structure is respected [16], [17], [41].

To address the above-mentioned problems, we propose a novel unsupervised NMF framework, called Robust Graph regularized NMF (RGNMF) which combines a sparse graph and a k -NN graph construction into matrix factorization to learn a discriminative representation. To be specific, RGNMF captures the global structure and preserves the discriminative information by constructing the sparse graph imposed on the basis matrix. And the new model constructs the k -NN graph constrained on the encoding matrix to depict the local geometric structure, which can learn the compact representation. Further, RGNMF exploits the $L_{2,1}$ -norm loss function to seek the basis matrix and encoding matrix so that it is insensitive to noise and outliers. Also, it imposes $L_{2,1}$ -norm constraint on the basis matrix to choose the discriminative feature. Hence, the proposed algorithm can gain a more discriminative representation for subsequent tasks by combining structure learning and $L_{2,1}$ -norm constraints imposed on the basis matrix and encoding matrix. In addition, an optimization scheme is developed to alternately solve such two metrics. Its convergence is proved theoretically and experimentally.

The proposed RGNMF has the following four contributions:

- (1) Different from existing approaches that ignore the geometric structure or only considers the local structure, our RGNMF algorithm not only respects the global structure, but also depicts the local structure. Moreover, it characterizes local and global structures as two regularization terms integrated into its loss function for respectively seeking the basis matrix and encoding matrix. Hence, RGNMF is particularly suitable for solving real-world problem via learning the intrinsic structure.
- (2) RGNMF constructs a sparse graph imposed on the basis matrix to catch the global structure and preserve the discriminative information. And it models the local structure by building a k -NN graph constrained on the encoding matrix, which gains the compact representation. Thus, RGNMF cannot only find more discriminative representations, but also project the new samples into the low-dimensional subspace by the learned basis.
- (3) RGNMF employs such a $L_{2,1}$ -norm cost function to decompose the basis matrix and encoding matrix that its robustness can be improved. Further, it imposes $L_{2,1}$ -norm constraint on the basis matrix to choose the discriminative feature. Obviously, the proposed algorithm can gain the robust data representation by combining structure learning and $L_{2,1}$ -norm constraints imposed on the basis matrix and encoding matrix. This indicates that RGNMF can naturally be used as a preprocessing technique for subsequent tasks, such as classification and clustering.
- (4) The power of our RGNMF algorithm lies in its ability to integrate feature learning, representation learning and $L_{2,1}$ -norm constraints into a general framework. Such a framework can be easily spread to supervised and semi-supervised scenarios. This naturally brings about wider application of RGNMF.

We organize this paper as follows: In Section 2, we briefly review several algorithms closely related to our work, such as KLS-NMF, GNLMF, and ENMF methods. In Section 3, our algorithm is introduced and the convergence proof of our optimization scheme is described in detail. Section 4 presents the experimental results and analysis. Finally, we conclude this paper in Section 5.

II. RELATED WORKS

In this section, we briefly review existing methods closely related to our work. These algorithms aim to respect the local geometric structure of data in the unsupervised scenario.

A. NMF WITH LOCAL LEARNING

Recent studies have shown that the local structure of data can be employed to enhance the quality of the learned representation [42], [43]. However, NMF fails to discover the intrinsic structure in its model. To this end, Cai *et al.* [10] proposed a graph regularized NMF (GNMF) to respect the intrinsic geometry of data. GNMF depicts the local structure by setting up a nearest neighbor graph and integrate it into NMF to seek

the compact representation. RMNMF [27] and MNMFL_{2,1} [28] extend GNMF by exploiting the L_{2,1}-norm loss function to replace the least square error function. Different from the above approaches that depict the local structure by building a nearest neighbor graph, a kernel local similarity-based NMF algorithm (KLS-NMF) [9] is proposed, which merges kernel local similarity learning and self-expressive property into matrix factorization for clustering. Specifically, KLS-NMF introduces self-expressive property to formulate a new basis matrix. Thus, the data matrix Y can be approximately expressed as the product of Y and A . KLS-NMF formulates the weight of the similarity between samples as the product of the basis matrix and encoding matrix. Moreover, it enforces an orthogonality constraint on the encoding matrix for enhancing the clustering performance. KLS-NMF solves the following problem:

$$\min_{A \geq 0, X \geq 0, X^T X = I} \|Y - YAX^T\|_F^2 + \text{Tr}(A^T M X), \quad (2)$$

where $\text{Tr}(\bullet)$ denotes the trace function of a matrix and $\text{Tr}(A^T M X)$ is a regularizer of the local similarity. M is a metric matrix with $S_{ij} = \|x_i - x_j\|_2^2$ and $Z = AX^T$ denotes a similarity matrix. To address the nonlinear problem, KLS-NMF introduces the kernel function into the model (2) and thus obtains the following loss function:

$$\min_{A \geq 0, X \geq 0, X^T X = I} \|\varphi(Y) - \varphi(Y)AX^T\|_F^2 + \text{Tr}(A^T M^\varphi X). \quad (3)$$

B. GRAPH REGULARIZED LOW-RANK NMF

Li *et al.* [31] incorporated NMF and the graph regularizer into a low-rank recovery algorithm for finding the essential representation of the data and thus proposed a graph regularized NLMF (GNLMF). Specifically, GNLMF decomposes its model into two subproblems: low-rank recovery and matrix factorization. It first applies the low-rank recovery technique to remove blur or noise in the original data, which optimizes the following objective subproblem:

$$\min_{G, B} \|Y - G - B\|_F^2, \quad (4)$$

where G is the low-rank part of Y and B denotes the blur or noise. After obtaining G , GNLMF encodes the geometric information by constructing the nearest neighbor graph and extends the objective function of NMF. Thus, the cost function of GNLMF is defined as the following optimization problem:

$$\min_{A \geq 0, X \geq 0} \|G - AX\|_F^2 + \alpha \text{Tr}(XLX^T) + \beta \text{Tr}(AA^T). \quad (5)$$

C. ELASTIC NMF

To address the problem of noise and outliers in the data, Xiong *et al.* [3] proposed a novel graph-regularized ENMF, which is adapted in Frobenius norm and L_{2,1}-norm. The elastic loss is used to fit matrix factorization for giving the

data and is defined as follows:

$$h(y_i, Ax_i) = \sum_i \frac{\delta \|y_i - Ax_i\|_F^2}{\delta + \|y_i - Ax_i\|} + \sum_i \frac{\|y_i - Ax_i\|_F^2}{\delta + \|y_i - Ax_i\|}, \quad (6)$$

where according to the scale parameter δ , the first term is called L₂ pseudo loss and the last term is called L₁ pseudo loss.

ENMF also considers the geometric structure of data by constructing the affinity graph and merges it into the elastic loss function as the regularization term. Thus, it optimizes the following objective function:

$$\min_{A \geq 0, X \geq 0} h(y_i, Ax_i) + \alpha \text{Tr}(XLX^T) + \beta \|X\|_{1,2}. \quad (7)$$

The above-mentioned algorithms improve the performance of NMF from different perspectives. However, these algorithms usually impose the intrinsic geometric information on the encoding matrix which indirectly affects the basis matrix. In addition, they fail to find the discriminative mapping, which is beneficial to obtain a better subspace representation. To address these issues, we propose a novel algorithm called RGNMF in this paper.

III. ROBUST GRAPH REGULARIZED NONNEGATIVE MATRIX FACTORIZATION (RGNMF)

A. MODEL FORMULATION

As analyzed above, the intrinsic geometric and discriminative information of the data space plays an essential role in finding such a more discriminative representation [12], [18], [23]. In fact, the quality of the representation obtained by most approaches is relatively poor due to the omission of the important information. Clearly, if the proposed algorithm satisfies the following three conditions, it can improve the quality of the data representation.

- (1) It should meet the local invariance. In other words, if two samples are neighbor in high-dimensional space, their corresponding representations are also neighbor in the low-dimensional space.
- (2) It should be able to discover the global structure, which is used to enhance the discriminative power of the representation.
- (3) It should choose the discriminative feature and make the model more robust.

When the data are projected into the low-dimensional space, one needs to preserve the local structure for finding the compact representation. To this end, samples from the high-dimensional space are close, their representations in the latent space should be close. For example, if two samples y_i and y_j are close, their corresponding low-dimensional representations x_i and x_j should be close. Therefore, this neighbor relationship is achieved by the following problem:

$$\begin{aligned} & \frac{1}{2} \sum_{i,j=1}^n \|x_i - x_j\|^2 S_{ij} \\ &= \sum_{i=1}^n x_i^T x_i D_{ii} - \sum_{i,j=1}^n x_i^T x_j S_{ij} \\ &= \text{Tr}(XDX^T) - \text{Tr}(XSX^T) = \text{Tr}(XL_S X^T) \end{aligned} \quad (8)$$

where the graph Laplacian L_S can be computed by $D-S$. Accordingly, any diagonal element of the diagonal matrix D is the sum of the corresponding column (or row) and the similarity matrix S can be obtained by Gaussian kernel function.

Clearly, the first condition can be realized by optimizing the problem (8). Because of the complexity of data structure in real-world applications, it is difficult to enhance the discriminative power of the representation by considering only one structure. For example, the face images of one person forms a subspace, and those of different persons consist of multiple subspaces. Because of the influence of illumination and posture, the data structures composed of these face images are very complex. If only the local structure is exploited, all images are not easy to be segmented into corresponding subspaces [44]. Hence, the global structure needs to be taken into account. Modified sparse representation (MSR) technique has been proved to be able to capture the global structure of data and find the discriminative mapping, which is helpful for the subsequent classification and clustering tasks [38], [40], [45]. Hence, we introduce MSR to respect the global structure and find the discriminative mapping by constructing the sparse graph. Specifically, we use as few samples as possible to reconstruct each sample y_i . We look for a sparse reconstructive weight z_i for each sample y_i by solving the following L_1 -norm optimization problem:

$$\begin{aligned} \min_{z_i} \|z_i\|_1 \\ \text{s.t. } y_i = Yz_i, \quad \mathbf{1}^T z_i = 1, \quad z_i \geq 0, \end{aligned} \quad (9)$$

where $z_i = [z_{i1}, \dots, z_{i,i-1}, 0, z_{i,i+1}, \dots, z_{in}]^T$ denotes an n -dimensional reconstructive vector where the i -th element is zero, and $\|\cdot\|_1$ is the L_1 -norm. $\mathbf{1} \in \mathbb{R}^n$ is a column vector whose entries are 1. The problem (9) is the widely used self-expression property of data, which represents each sample as a linear combination of other samples in the same group. It can reflect the intrinsic geometric characteristics of data and preserve potential discriminative information [38], [40], [68]. Lin et al. [69] exploited the self-expression property of data to seek a smooth node representation, so as to achieve multi-view graph clustering.

The alternating direction method of multipliers (ADMM) [49] has been proven to be effective in solving L_1 -norm optimization problems [50], [51], [52]. Thus, we exploit it to solve the problem (9) with regard to Z . Following methods [50], [51], [52], we make use of ADMM to solve the sparse weight Z . We first transform the problem (9) into the following equivalent problem:

$$\begin{aligned} \min_{Z, V} \|V\|_1 \\ \text{s.t. } V = Z, \quad Y = YZ, \quad \mathbf{1}^T Z = \mathbf{1}, \quad Z \geq 0. \end{aligned} \quad (10)$$

We define an augmented Lagrange formula to solve the problem (10):

$$\begin{aligned} \mathcal{L}(Z, V) = \|V\|_1 + \text{Tr}(C^T(Y - YZ)) + \text{Tr}(H^T(Z - V)) \\ + \frac{\mu}{2}(\|Y - YZ\|_F^2 + \|Z - V\|_F^2) \end{aligned} \quad (11)$$

where C and H are two Lagrange multipliers, $\mu > 0$ denotes a penalty parameter. Since there are two variables Z and V in the problem (11), we need to solve them alternately. Hence, we design the following three steps to deal with these two variables.

Step 1: Computing Z . When V is fixed, Z is updated by solving the problem

$$\begin{aligned} \min_{Z \geq 0} \frac{\mu}{2}(\|Y - YZ\|_F^2 + \|Z - V\|_F^2) \\ + \text{Tr}(C^T(Y - YZ)) + \text{Tr}(H^T(Z - V)). \end{aligned} \quad (12)$$

For each i ($i = 1 \dots n$), we define the following Lagrange function to obtain Z :

$$\begin{aligned} \mathcal{L}(z_i, \eta, \varsigma_i) = \frac{\mu}{2}(\|y_i - Yz_i\|_2^2 + \|z_i - v_i\|_2^2) \\ + C_i^T(y_i - Yz_i) + H_i^T(z_i - v_i) - \eta(\mathbf{1}^T z_i - 1) \\ - \varsigma_i^T z_i \end{aligned} \quad (13)$$

where η and ς_i are two nonnegative Lagrange multipliers.

We take the partial derivative of Eq. (13) with respect to z_i and set it to 0, thus obtaining:

$$\begin{aligned} \mu(Y^T Y + I)z_i - u(Y^T y_i + v_i) \\ - Y^T C_i + H_i - \eta \mathbf{1} - \varsigma_i = 0. \end{aligned} \quad (14)$$

We further simplify Eq. (14) and gain:

$$z_i + \delta_i - \eta \omega - v_i \varsigma_i = 0. \quad (15)$$

where $\delta_i = (Y^T Y + I)^{-1}(-Y^T y_i - v_i - (Y^T C_i - H_i)/\mu)$, $\omega = (Y^T Y)^{-1} \mathbf{1}/\mu$ and $v = (Y^T Y)^{-1}/\mu$. For the j -th element of z_i , we gain

$$z_{ij} + \delta_{ij} - \eta \omega_j - v_{ij} \varsigma_{ij} = 0. \quad (16)$$

According to KKT condition, we have $\varsigma_{ij} z_{ij} = 0$ and thus get

$$z_{ij} = (-\delta_{ij} + \eta \omega_j)_+, \quad (17)$$

where $(\tau)_+ = \max(0, \tau)$. Without loss of generality, we assume that $\delta_{i1}, \delta_{i2}, \dots, \delta_{in}$ are sorted from small to large. If k elements of the optimal z_i are nonzero, then we get $z_{ik} > 0$ and $z_{i,k+1} = 0$ in the light of Eq. (17). It follows that

$$-\delta_{ik} + \eta \omega_k > 0, \quad \text{and} \quad -\delta_{i,k+1} + \eta \omega_{k+1} \leq 0. \quad (18)$$

Combining the constraint $\mathbf{1}^T z_i = 1$ and Eq. (17), we arrive at

$$\sum_{r=1}^k (-\delta_{ir} + \eta \omega_r) = 1 \Rightarrow \eta = \frac{1 + \sum_{r=1}^k \delta_{ir}}{\sum_{r=1}^k \omega_r}. \quad (19)$$

Step 2: Computing V . When Z is fixed, V is updated by solving the problem

$$\min_V \|V\|_1 + \text{Tr}(H^T(Z - V)) + \frac{\mu}{2} \|Z - V\|_F^2. \quad (20)$$

We update V by solving the optimization problem

$$V = \arg \min \|V\|_1 + \frac{\mu}{2} \left\| Z - V + \frac{H}{\mu} \right\|_F^2. \quad (21)$$

Therefore, we can gain the closed solution of V :

$$V = \Omega_{1/\mu}(Z + H/\mu), \quad (22)$$

where Ω denotes the shrinkage operator [53].

Step 3: Computing two Lagrange multipliers and the penalty parameter. C , H and μ can be updated as follows:

$$C = C + \mu(Y - YZ) \quad (23)$$

$$H = H + \mu(Z - V) \quad (24)$$

$$\mu = \min(\rho\mu, \mu_{\max}) \quad (25)$$

where ρ and μ_{\max} are two nonnegative constants. We summarize the process of solving problem (9) in Algorithm 1.

Algorithm 1 Solving the Problem (9) by ADMM

Input: Data matrix Y .

Initialize: $Z = V = 0$, $C = H = 0$, $\mu = 10^{-2}$, $\rho = 1.2$, $\mu_{\max} = 10^8$.

while not converged do

1. Update Z by solving the problem (12).

2. Update V by exploiting (22).

3. Update C , H and μ by exploiting (23)-(25), respectively.
end while

Output: Z and V .

The theoretical results of ADMM [49], [50] have proved that the iterative process of solving two variables is convergent. As formulated in Algorithm 1, we exploit the ADMM method to iteratively solve two variables. Hence, Algorithm 1 converges.

After obtaining the optimal weight vector z_i , the sparse reconstructive weight matrix is denoted as $Z = [z_1, z_2, \dots, z_n]$. As demonstrated in [38], [40], and [45], the sparse graph has three important advantages: 1) Since z_i is constructed by exploiting all the samples, it characterizes the global structure. 2) Although there are no class labels available, the discriminative information can be naturally preserved in the weight matrix. 3) Because of its sparsity, the weight matrix Z is robust to noise and outliers in the data. Since NMF is a popular dimensionality reduction technique, A is also used as a projection matrix [10], [18], [22]. Our proposed RGNMF aims to project samples with the same structure into the same cluster and samples with different structures into different clusters. Hence, it can preserve the discriminative information through the basis matrix. To achieve this, after obtaining the reconstructive weight matrix Z , we can optimize the following problem to gain the basis matrix:

$$\min_A \sum_{i=1}^n \|A^T y_i - A^T Y z_i\|_2^2. \quad (26)$$

For the convenience of calculation, we simplify (26) as follows:

$$\begin{aligned} & \sum_{i=1}^n \|A^T y_i - A^T Y z_i\|_2^2 \\ &= \text{Tr}(A^T (\sum_{i=1}^n (y_i - Y z_i)(y_i - Y z_i)^T) A) \\ &= \text{Tr}(A^T (\sum_{i=1}^n (Y e_i - Y z_i)(Y e_i - Y z_i)^T) A) \end{aligned}$$

$$\begin{aligned} &= \text{Tr}(A^T Y (\sum_{i=1}^n (e_i e_i^T - e_i z_i^T - z_i e_i^T + z_i z_i^T)) Y^T A) \\ &= \text{Tr}(A^T Y (I - Z^T - Z + Z^T Z) Y^T A) \\ &= \text{Tr}(A^T Y L_Z Y^T A), \end{aligned} \quad (27)$$

where the i -th entry of the n -dimensional column vector e_i is 1, and the other entries are 0. $L_Z = I - Z^T - Z + Z^T Z$. By minimizing the problem (27), we respect the global structure and preserve the discriminative information. Therefore, we can fulfil the second property.

Discriminative Ridge Machine (DRM) [54] is a supervised classification method by introducing a discriminant ridge regression. DRM makes use of data labels to investigate the between-class scatter and within-class scatter, respectively. Therefore, it can accurately derive class information and obtain an appropriate representation model by taking into account the discriminativeness between classes. Although DRM and our RGNMF can learn a discriminative representation, they have two main differences:

- (1) With data labels, DRM can learn the global discriminativeness by maximizing between-class separability. Since our RGNMF is an unsupervised algorithm, we solve the problem (9) to respect the global structure. After arriving at the reconstructive weight matrix, our algorithm can learn the global discriminativeness by minimizing the problem (26). It is worth noting that, when DRM does not use data labels, it uses the same method as RGNMF to learn the global structure.
- (2) DRM seeks the local discriminativeness by maximizing within-class similarity. Different from DRM, our RGNMF considers the local geometrical structure by minimizing the problem (8). Actually, the difference between DRM and RGNMF in learning the local discriminativeness is a typical difference between supervised and unsupervised methods.

NMF and its variants usually employ the least squares loss function to minimize the two nonnegative factors. However, their performance declines when the data are contaminated by noise and outliers. On the other hand, the $L_{2,1}$ -norm loss function can effectively deal with contaminated data [27], [28], [32]. Therefore, we introduce a $L_{2,1}$ -norm loss to handle the problem of noise and outliers as

$$\min_{A \geq 0, X \geq 0} \|Y - AX\|_{2,1} + \lambda \|A\|_{2,1}. \quad (28)$$

where the $L_{2,1}$ -norm of the matrix A is equivalent to $\text{Tr}(A^T P A)$, that is, $\|A\|_{2,1} = \text{Tr}(A^T P A)$. The diagonal matrix P can be expressed as $[P_{ii}] = 1/(\|a_i\|_2)$. In (28), the first term is used to enhance the robustness of matrix factorization, and the second term is used to choose discriminative features [43], [46].

The advantage of using the $L_{2,1}$ -norm objective function is that our RGNMF can well address the problems of noise and outliers. Such an advantage has been proved by many NMF-based methods [7], [14], [27], [28]. To demonstrate the robustness of the $L_{2,1}$ -norm objective function, we display the

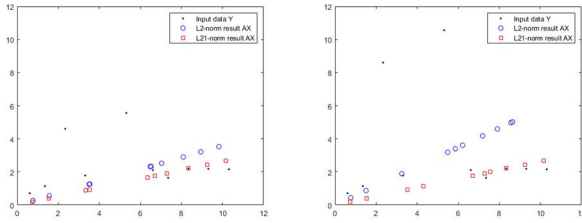


FIGURE 1. Illustration of the robustness of the $L_{2,1}$ -norm cost function.

experimental results of the $L_{2,1}$ -norm and L_2 -norm objective functions on the toy data set. The toy data set consists of ten data points, two of which are outliers. The experimental results are shown in Fig. 1. When the data set is contaminated by the outlier y_r , the residual $o_r = \|y_r - Ax_r\|_2^2$ is larger than those of other data points. Thus, the outlier can easily control the L_2 -norm objective function. As can be seen from the left panel of Fig. 1, the L_2 -norm results are closer to the outliers than the $L_{2,1}$ -norm results. Further, we assign larger values to two outliers. In other words, two outliers are farther away from other points. We can observe from the right panel of Fig. 1 that the L_2 -norm results are greatly affected. However, the $L_{2,1}$ -norm results seem to remain unchanged. This indicates that the $L_{2,1}$ -norm objective function is more robust than the L_2 -norm objective function.

According to the above formulation, the loss function of our RGNMF can be defined as

$$\begin{aligned} \min_{A, X} & \|Y - AX\|_{2,1} + \alpha \text{Tr}(A^T YL_Z Y^T A) + \beta \text{Tr}(XL_S X^T) \\ & + \lambda \|A\|_{2,1} \\ \text{s.t. } & A \geq 0, \quad X \geq 0, \end{aligned} \quad (29)$$

where α , β and λ are three nonnegative parameters. By optimizing the problem (29), the above three conditions can be met.

It is worth noting that although the k-NN graph and $L_{2,1}$ -norm were introduced into existing approaches [10], [27], [28], our algorithm has the following three differences from them:

- (1) Unlike other methods that use only a single graph, our RGNMF algorithm simultaneously constructs two graphs, namely, the sparse graph and the k-NN graph. The former graph is used to respect the global structure, and the latter one is used to preserve the local structure. Our algorithm characterizes local and global structures as two regularization terms integrated into our objective function. Hence, RGNMF takes advantage of complex data structure, which is particularly suitable for solving real-world problem.
- (2) Different from the existing methods of imposing the discriminative constraint on the coding matrix, our algorithm constructs the sparse graph imposed on the basis matrix to catch the global structure and preserve the discriminative information. And it models the local structure by building the k-NN graph constrained on the encoding matrix. Thus, RGNMF cannot only find more discriminative representations, but also project

the new samples into the low-dimensional subspace by the learned basis.

- (3) Our algorithm lays special stress on the joint $L_{2,1}$ -norm optimization of the cost function and the basis matrix. The $L_{2,1}$ -norm loss function is insensitive to noise and outliers. And the $L_{2,1}$ -norm optimization on the basis matrix is applied to choose the discriminative feature. In a word, the proposed algorithm can gain the robust data representation by combining structure learning and $L_{2,1}$ -norm minimization of the cost function and the basis matrix. This indicates that our algorithm can naturally be used as a preprocessing technique for subsequent tasks, such as classification and clustering.

B. OPTIMAL SOLUTION FOR TWO FACTOR MATRICES

To gain the optimal solution of the two matrices A and X , we update one matrix by fixing the other. To this end, the loss function of RGNMF in (29) can be expressed as

$$\begin{aligned} J &= \text{Tr}((Y - AX)P(Y - AX)^T) + \alpha \text{Tr}(A^T YL_Z Y^T A) \\ &+ \beta \text{Tr}(XL_S X^T) + \lambda \text{Tr}(A^T QA) \\ &= \text{Tr}(YPY^T) - 2\text{Tr}(YPX^T A^T) + \text{Tr}(AXPX^T A^T) \\ &+ \alpha \text{Tr}(A^T YL_Z Y^T A) + \beta \text{Tr}(XL_S X^T) + \lambda \text{Tr}(A^T QA) \end{aligned} \quad (30)$$

where we apply $\text{Tr}(U) = \text{Tr}(U^T)$ and $\text{Tr}(UH) = \text{Tr}(HU)$ to merge similar terms. The diagonal entries of two diagonal matrices P and Q are computed as

$$P_{bb} = 1/(2\sqrt{\sum_{j=1}^m (Y - AX)_{jb}^2}), \quad (31)$$

and

$$Q_{cc} = 1/(2\sqrt{\sum_{j=1}^m A_{jc}^2}). \quad (32)$$

Considering inequality constraints $A \geq 0$ and $X \geq 0$, two Lagrange multipliers ψ_{ik} and ϕ_{kj} need to be set for these two variables A_{ik} and X_{kj} . Therefore, we can introduce the following Lagrange problem with $\Psi = [\psi_{ik}]$ and $\Phi = [\phi_{kj}]$:

$$\begin{aligned} \mathcal{L} &= \text{Tr}(YPY^T) - 2\text{Tr}(YPX^T A^T) + \text{Tr}(AXPX^T A^T) \\ &+ \alpha \text{Tr}(A^T YL_Z Y^T A) + \beta \text{Tr}(XL_S X^T) + \lambda \text{Tr}(A^T QA) \\ &+ \text{Tr}(\Psi A^T) + \text{Tr}(\Phi X^T). \end{aligned} \quad (33)$$

We calculate the partial derivatives of (33) concerning two variables A and X as follows:

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial A} &= -2YPX^T + 2AXPX^T + 2\alpha YL_Z Y^T A \\ &+ 2\lambda QA + \Psi, \end{aligned} \quad (34)$$

$$\frac{\partial \mathcal{L}}{\partial X} = -2A^T YP + 2A^T AXP + 2\beta XL_S + \Phi. \quad (35)$$

We set $\psi_{ik} A_{ik} = 0$ and $\phi_{kj} X_{kj} = 0$ by applying the KKT conditions and arrive at two equations with regard to A_{ik} and X_{kj} .

$$\begin{aligned} -(YPX^T)_{ik} A_{ik} + (AXPX^T)_{ik} A_{ik} \\ + \alpha (YL_Z Y^T A)_{ik} A_{ik} + \lambda (QA)_{ik} A_{ik} = 0 \end{aligned} \quad (36)$$

$$\begin{aligned} -(A^T YP)_{kj} X_{kj} + (A^T AXP)_{kj} X_{kj} + \beta (XL_S)_{kj} X_{kj} = 0. \end{aligned} \quad (37)$$

Hence, we gain the following solutions of A and X :

$$A_{ik} \leftarrow A_{ik} \frac{(YPX^T + \alpha Y(Z + Z^T)Y^T A)_{ik}}{(AXPX^T + \alpha Y(I + ZZ^T)Y^T A + \lambda QA)_{ik}} \quad (38)$$

$$X_{kj} \leftarrow X_{kj} \frac{(A^T YP + \beta XS)_{kj}}{(A^T AXP + \beta XD)_{kj}}. \quad (39)$$

Algorithm 2 Our RGNMF

Input: Data matrix Y and parameters α, β, λ .

Set $t = 0$; Initialize A_0, X_0 .

Compute the similarity weight S by Gaussian kernel.

Compute the sparsity weight Z by using Algorithm 1.

while ($t < 300$ or $|J^{t-1} - J^t|/J^{t-1} > 10^{-3}$)

1. Update A by using (38).

2. Update X by using (39).

3. Update P and Q by using (31) and (32), respectively.

4. $t = t + 1$.

end while

Output: A and X .

Obviously, the solutions of A and X are an iterative updating process. When such an updating process stops, their optimal solutions can be obtained. We formulate the iterative updating process for A and X in Algorithm 2.

C. PROOF OF CONVERGENCE

We need to prove that the objective function in (30) is non-increasing under the updating rules in (38) and (39). The presented proof will adopt an auxiliary function similar to that used in NMF and its variants. Therefore, we introduce the definition of the auxiliary function.

Definition 1: Auxiliary Function Give any two function $G(b, b')$ and $H(b)$, if $G(b, b') \geq H(b)$ and $G(b, b) = H(b)$ hold, $G(b, b')$ denotes an auxiliary function of $H(b)$.

Proposition 1: Suppose $G(b, b')$ is the auxiliary function of $H(b)$, the function H is decreasing via the following optimization problem:

$$b^{t+1} = \arg \min_b G(b, b^t) \quad (40)$$

Proof of Proposition 1: $H(b^{t+1}) \leq G(b^{t+1}, b^t) \leq G(b^t, b^t) = H(b^t)$.

As seen from (38) and (39), the updating rules for A and X need to be executed alternately. We first demonstrate the convergence of the updating rule for A in (38). To this end, we fix three variables fix X, P , and Q , and define a proper auxiliary function for A . For any element A_{ik} in A , we make use of $F(A_{ik})$ to denote the part of (30) with regard to A . $F(A_{ik})$ is formulated as:

$$F(A) = Tr(-2YPX^T A^T + AXPX^T A^T) + \alpha Tr(A^T YL_Z Y^T A) + \lambda Tr(A^T QA). \quad (41)$$

Proposition 2: The function

$$\begin{aligned} G(A, A^t) &= F(A^t) + \sum_{i,k} \left(\frac{\partial F(A^t)}{\partial A^t} \right)_{ik} (A - A^t)_{ik} \\ &+ \sum_{i,k} \frac{(A^t XPX^T + \alpha Y(I + ZZ^T)Y^T A^t + \lambda QA^t)_{ik}}{A^t_{ik}} \\ &\times (A - A^t)_{ik}^2 \end{aligned} \quad (42)$$

is an auxiliary function of $F(A_{ik})$.

Proof of Proposition 2: It is easy to check that $G(A, A) = F(A)$. According to Definition 1, we need to verify $G(A, A^t) \geq F(A)$. Consequently, we compute the first-order and the second-order derivative of $F(A)$ about A , respectively:

$$F'(A_{ik}) = (-2YPX^T + 2AXPX^T + 2\alpha YL_Z Y^T A + 2\lambda QA)_{ik}, \quad (43)$$

$$F''(A_{ik}) = 2(XPX^T)_{kk} + 2\alpha(YL_Z Y^T)_{ii} + 2\lambda Q_{ii}. \quad (44)$$

With (43) and (44), it is easy to gain a Taylor series problem of $F(A)$,

$$\begin{aligned} F(A) &= F(A^t) + \sum_{i,k} F'(A^t)(A - A^t)_{ik} \\ &+ \sum_{i,k} [(XPX^T)_{kk} + \alpha(YL_Z Y^T)_{ii} + \lambda Q_{ii}](A - A^t)_{ik}^2. \end{aligned} \quad (45)$$

From (42) and (45), we can observe that $G(A, A^t) \geq F(A)$ is equivalent to

$$\begin{aligned} &\frac{(A^t XPX^T + \alpha Y(I + ZZ^T)Y^T A^t + \lambda QA^t)_{ik}}{A^t_{ik}} \\ &\geq (XPX^T)_{kk} + \alpha(YL_Z Y^T)_{ii} + \lambda Q_{ii}. \end{aligned} \quad (46)$$

We can gain the following inequality according to $A \geq 0$ and $X \geq 0$:

$$(A^t XPX^T)_{ik} = \sum_{r=1}^l A^t_{ir} (XPX^T)_{rk} \geq A^t_{ik} (XPX^T)_{kk} \quad (47)$$

$$\begin{aligned} \alpha(Y(I + ZZ^T)Y^T A^t)_{ik} &= \alpha \sum_{r=1}^m (Y(I + ZZ^T)Y^T)_{ir} A^t_{rk} \\ &\geq \alpha(Y(I + ZZ^T)Y^T)_{ii} A^t_{ik} \\ &\geq \alpha(Y(I + ZZ^T - Z - Z^T)Y^T)_{ii} A^t_{ik} \\ &\geq \alpha(YL_Z Y^T)_{ii} A^t_{ik} \end{aligned} \quad (48)$$

and

$$\lambda(QA^t)_{ik} = \lambda \sum_{r=1}^l Q_{ir} A^t_{rk} \geq \lambda Q_{ii} A^t_{ik}. \quad (49)$$

By combining (47)-(49), (42) holds and $G(A, A^t) \geq F(A)$. We finish the proof of Proposition 2.

Theorem 1: When the updating rule of (38) is used to solve the matrix A , the value of our objective function in (30) is decreasing.

Proof of Theorem 1: We substitute (42) into (40) to gain the following equation with the help of Propositions 1 and 2:

$$\begin{aligned} A_{ik}^{t+1} &= A_{ik}^t - A_{ik}^t \frac{\mathcal{F}'(A_{ik}^t)}{2(A^t X P X^T + \alpha Y(I + Z Z^T) Y^T A^t + \lambda Q A^t)_{ik}} \\ &= A_{ik}^t \frac{(Y P X^T + \alpha Y(Z + Z^T) Y^T A^t)_{ik}}{(A^t X P X^T + \alpha Y(I + Z Z^T) Y^T A^t + \lambda Q A^t)_{ik}} \end{aligned} \quad (50)$$

It is easy to see that (50) and (38) are consistent. We obtain that $G(A, A^t)$ is an auxiliary function of $F(A)$. Thus, the value of $F(A)$ is decreasing under (38) when the other variables are fixed.

Similar to the proof of the updating rule for A in (38), we prove that the value of $F(X)$ is decreasing, in which $F(X)$ is the objective function in (30) by fixing A, P and Q . $F(X)$ can be described as

$$F(X) = \text{Tr}(-2Y P X^T A^T + A X P X^T A^T) + \beta \text{Tr}(X L_S X^T). \quad (51)$$

Proposition 3: The function

$$\begin{aligned} G(X, X^t) &= \mathcal{F}(X^t) + \sum_{k,j} \mathcal{F}'(X^t)(X - X^t)_{kj} \\ &\quad + \sum_{k,j} \frac{(A^T A X^t P + \beta X^t D)_{kj}}{X_{kj}^t} (X - X^t)_{kj}^2 \end{aligned} \quad (52)$$

is an auxiliary function of $F(X)$ only related to X .

Proof of Proposition 3: Since $G(X, X) = F(X)$ is evident, we just need to prove $G(X, X^t) \geq F(X)$. Similar to the proof of Proposition 2, we formulate the first and the second derivative of $F(X)$:

$$F'(X_{kj}) = (-2A^T Y P + 2A^T A X P + 2\beta X L_S)_{kj}, \quad (53)$$

$$F''(X_{kj}) = 2(A^T A)_{kk} P_{jj} + 2\beta(L_S)_{jj}. \quad (54)$$

With (53) and (54), the function $F(X)$ can spread the following Taylor series problem:

$$\begin{aligned} F(X) &= F(X^t) + \sum_{k,j} F'(X^t)(X - X^t)_{kj} \\ &\quad + \sum_{k,j} [(A^T A)_{kk} P_{jj} + \beta(L_S)_{jj}](X - X^t)_{kj}^2. \end{aligned} \quad (55)$$

To verify $G(X, X^t) \geq F(X)$, we need to demonstrate that the following inequality holds:

$$\frac{(A^T A X^t P + \beta X^t D)_{kj}}{X_{kj}^t} \geq (A^T A)_{kk} P_{jj} + \beta(L_S)_{jj}. \quad (56)$$

According to $A \geq 0$ and $X \geq 0$, the following inequality holds:

$$\begin{aligned} (A^T A X^t P)_{kj} &= \sum_{r=1}^l (A^T A)_{kr} (X^t P)_{rj} \\ &= \sum_{r=1}^l (A^T A)_{kr} X_{rj}^t P_{jj} \geq (A^T A)_{kk} X_{kj}^t P_{jj} \end{aligned} \quad (57)$$

and

$$\begin{aligned} \beta(X^t D)_{kj} &= \beta \sum_{r=1}^n X_{kr}^t D_{rj} \geq \beta X_{kj}^t D_{jj} \\ &\geq \beta X_{kj}^t (D_{jj} - S_{jj}) = \beta X_{kj}^t (L_S)_{jj}. \end{aligned} \quad (58)$$

The sum of the left-hand side of (57) and (58) is greater than or equal to the sum of their right-hand side. Hence, we have $G(X, X^t) \geq F(X)$. Proposition 3 is proven.

Theorem 2: When the updating rule of (39) is used to solve the matrix X , the value of our objective function in (30) is decreasing.

Proof of Theorem 2: We substitute (52) into (40) to gain the following equation with the help of Propositions 1 and 3:

$$\begin{aligned} X_{kj}^{t+1} &= X_{kj}^t - X_{kj}^t \frac{F'(X_{kj}^t)}{2(A^T A X^t P + \beta X^t D)_{kj}} \\ &= X_{kj}^t \frac{(A^T Y P + \beta X^t S)_{kj}}{(A^T A X^t P + \beta X^t D)_{kj}}. \end{aligned} \quad (59)$$

Because of Proposition 3, the value of $F(X)$ is decreasing under (39) when the other variables are fixed. We have completed the proof of theorem 2.

Theorem 3: When the updating rules of (38) and (39) are used to solve the nonnegative matrices A and X , the value of our objective function in (30) is decreasing. RGNMF remains stable if and only if A and X reach the local optimal value.

Proof of Theorem 3: Assuming that Algorithm 1 is executed to the $(t+1)$ -th iteration, we gain the following inequality with Theorem 1:

$$J(A^{t+1}, X^t, P^t, Q^t) \leq J(A^t, X^t, P^t, Q^t), \quad (60)$$

where X^t, P^t and Q^t denote the values of the t -th iteration.

When the values of A^{t+1}, P^t and Q^t are fixed, we can gain the following in equality via Theorem 2:

$$J(A^{t+1}, X^{t+1}, P^t, Q^t) \leq J(A^{t+1}, X^t, P^t, Q^t). \quad (61)$$

According to (42) and (43), we have

$$J(A^{t+1}, X^{t+1}, P^t, Q^t) \leq J(A^t, X^t, P^t, Q^t), \quad (62)$$

that is

$$\sum_{r=1}^m \frac{\|u_r^{t+1}\|_2^2}{2\|u_r^t\|_2} \leq \sum_{r=1}^m \frac{\|u_r^t\|_2^2}{2\|u_r^t\|_2}, \quad (63)$$

where $U = Y - AX$ whose i -th column vector is u_i .

From Lemma 1 in [46], the following inequation holds:

$$\|u_r^{t+1}\|_2 - \frac{\|u_r^{t+1}\|_2^2}{2\|u_r^t\|_2} \leq \|u_r^t\|_2 - \frac{\|u_r^t\|_2^2}{2\|u_r^t\|_2}. \quad (64)$$

Further, we have

$$\sum_{r=1}^m \left(\|u_r^{t+1}\|_2 - \frac{\|u_r^{t+1}\|_2^2}{2\|u_r^t\|_2} \right) \leq \sum_{r=1}^m \left(\|u_r^t\|_2 - \frac{\|u_r^t\|_2^2}{2\|u_r^t\|_2} \right). \quad (65)$$

We get the following inequality by combining (63) and (65):

$$\sum_{r=1}^m \|u_r^{t+1}\|_2 \leq \sum_{r=1}^m \|u_r^t\|_2. \quad (66)$$

Therefore,

$$\|U^{t+1}\|_{2,1} \leq \|U^t\|_{2,1}. \quad (67)$$

According to the above analysis, we have

$$J(A^{t+1}, X^{t+1}, P^{t+1}, Q^t) \leq J(A^t, X^t, P^t, Q^t). \quad (68)$$

Similar to the proof of (67), we gain

$$\|A^{t+1}\|_{2,1} \leq \|A^t\|_{2,1}. \quad (69)$$

Hence,

$$J(A^{t+1}, X^{t+1}, P^{t+1}, Q^{t+1}) \leq J(A^t, X^t, P^t, Q^t). \quad (70)$$

Theorem 3 has proved and thus Algorithm 1 is convergent. When A and X are updated by exploiting (38) and (39), they remain stable if and only if they reach the local optimal value.

D. COMPUTATIONAL COMPLEXITY

We use a big O notation to describe the complexity of the proposed algorithm. It is important to note that we need to compute two weight matrices Z and W . It cost $O(mn \log(m))$ to compute Z . In fact, Solving Z results in a sparse representation issue, which can be efficiently solved by off-the-shelf algorithms. RGNMF also needs $O(mn^2)$ to construct the nearest neighbor graph.

From Steps 4 and 5 in Algorithm 1, we need to calculate YP , AXP , XS and $ZY^T A$. Since G is a diagonal matrix, it cost $O(mn)$ and $O(lmn)$ to compute YP and AXP , respectively. RGNMF costs $O(\ln^2)$ and $O(lmn)$ to compute XS and $ZY^T A$, respectively. Thus, the complexities to update A and X are $O(mn + lmn)$ and $O(\ln^2 + lmn)$, respectively. In addition, since both P and Q are diagonal matrixes, they cost $O(mn + n)$ and $O(ml + l)$, respectively. Assuming that our proposed method converges after t iterations, the overall cost for RGNMF is $O(tmnl + t(mn + n) + t(ml + l) + mn \log(m) + mn^2)$.

IV. EXPERIMENTAL RESULT

Existing state-of-the-art algorithms can achieve relatively good performance. However, they depicted the graph structure constrained on the encoding matrix, which only indirectly affects the base matrix. In this section, we investigate our RGNMF by conducting extensive experiments.

A. DATA SETS

To make fair comparison with other methods, we use eight commonly-used data sets to evaluate the performance of our algorithm and other related methods. These data sets are composed of three biological data sets,¹ i.e., Carcinom,

¹<http://featureselection.asu.edu/datasets.php>

TABLE 1. Details of the data sets.

Data sets	Samples	Features	Classes
ORL	400	1024	40
CBCL	2000	1024	10
Face94	3040	4096	152
Face95	1440	4096	72
Georgia	750	4096	50
Carcinom	174	9182	11
LUNG	203	3312	5
TOX_171	171	5748	4
MNIST	10000	784	10

LUNG and TOX_171, five face image data sets, i.e., ORL,² CBCL [61], Face94,³ Face95³, Georgia [62]. In addition, we use a big data set MNIST⁴ to evaluate the performance of all algorithms. The data contained in the above-mentioned data sets are real-world and used by the state-of-the-art algorithms [7], [23], [28], [41], [46], [66], [67]. ORL and Georgia are two relatively new data sets. Details of these data sets are described in Table 1.

B. EVALUATION MEASURES

Two widely used measures, accuracy (ACC) and normalized mutual information (NMI) [10], [18], [20], are adopted to compare the performance of our RGNMF and other related approaches. The values of ACC and NMI are in the interval [0,1]. The closer their values are to 1, the better the learning performance of this algorithm. ACC is applied to measure the percentage of correct groups obtained by one algorithm and formulated as

$$ACC = \frac{1}{n} \sum_{i=1}^n \omega(r_i, \eta(c_i)), \quad (71)$$

where c_i denotes the clustering label and r_i is the ground truth label provided by the data set. Here, $\eta(c_i)$ aims at mapping the clustering labels to the ground truth labels. The indicator function $\omega(\alpha, \beta)$ equals 1 if $\alpha = \beta$ and equals 0 otherwise.

NMI is used to measure the similarity between two sets of clusters and is depicted as follows:

$$NMI = \frac{\sum_{i=1}^K \sum_{j=1}^K n_{ij} \log(\frac{n_{ij}}{n_i \hat{n}_j})}{\sqrt{(\sum_{i=1}^K n_i \log \frac{n_i}{n})(\sum_{j=1}^K \hat{n}_j \log \frac{\hat{n}_j}{n})}}, \quad (72)$$

where n_i denotes the sample number of the cluster C_i ($1 \leq i \leq K$) provided by the clustering algorithm and \hat{n}_j is the number of samples belonging to the j -th ground-truth class ($1 \leq j \leq K$), and n_{ij} denotes the number of samples that are in the intersection between the cluster C_i and the j -th class.

²<http://www.cad.zju.edu.cn/home/dengcai/Data/data.html>

³<https://cswww.essex.ac.uk/mv/allfaces/index.html>

⁴<http://yann.lecun.com/exdb/mnist/>

TABLE 2. Clustering results (ACC% \pm std%) of the compared methods.

Datasets	ORL	CBCL	Face94	Face95	Grimace	Carcinom	LUNG	TOX 171	MNIST
NMF	60.75 \pm 2.71	63.67 \pm 3.43	75.1 \pm 2.27	37.56 \pm 1.61	85.06 \pm 5.35	45.98 \pm 3.82	46.8 \pm 3.43	43.86 \pm 3.85	44.34 \pm 4.45
L ₂₁ NMF	60.54 \pm 2.81	69.43 \pm 3.71	75.81 \pm 1.56	40.26 \pm 1.85	87.81 \pm 5.42	50.2 \pm 3.67	45.81 \pm 3.9	43.28 \pm 4.23	46.18 \pm 5.21
GNNMF	66.72 \pm 3.22	64.79 \pm 4.15	75.22 \pm 0.61	41.66 \pm 1.41	87.58 \pm 5.20	59.2 \pm 3.14	79.8 \pm 2.86	44.14 \pm 3.66	61.06 \pm 3.34
MNMFL ₂₁	68.39 \pm 2.55	68.34 \pm 4.4	75.77 \pm 1.35	45.08 \pm 1.37	88.67 \pm 5.13	69.54 \pm 2.92	79.8 \pm 2.91	44.44 \pm 3.47	62.31 \pm 3.79
ONMFS	65.5 \pm 2.41	70.1 \pm 3.37	75.07 \pm 1.72	43.26 \pm 2.21	88.13 \pm 1.99	49.43 \pm 3.74	57.14 \pm 3.28	46.2 \pm 3.7	53.33 \pm 4.24
NMF-LCAG	64.76 \pm 3.18	64.82 \pm 5.55	74.66 \pm 1.45	38.71 \pm 1.52	85.99 \pm 4.64	54.02 \pm 4.25	76.84 \pm 3.34	45.61 \pm 4.39	58.25 \pm 3.29
ENMF	68.25 \pm 3.49	71.25 \pm 4.29	75.31 \pm 2.15	45.35 \pm 1.33	89.83 \pm 2.12	58.05 \pm 4.4	78.33 \pm 2.31	45.03 \pm 4.12	59.44 \pm 4.28
GLNMF	67.35 \pm 2.66	68.20 \pm 4.83	75.79 \pm 1.42	44.79 \pm 1.48	89.02 \pm 4.92	64.37 \pm 3.29	80.79 \pm 3.11	44.35 \pm 3.54	55.16 \pm 4.77
CHNMF	72.5 \pm 3.11	72.95 \pm 4.02	75.66 \pm 1.19	49.23 \pm 1.21	86.38 \pm 4.45	56.89 \pm 4.02	79.80 \pm 3.69	42.69 \pm 3.15	62.23 \pm 3.32
DSNMF	68.73 \pm 3.15	65.97 \pm 4.22	76.15 \pm 2.31	43.9 \pm 1.54	90.17 \pm 3.66	66.6 \pm 4.21	65.29 \pm 4.7	44.71 \pm 2.88	58.74 \pm 4.44
Ours	74.25\pm2.33	81.7\pm3.18	77.07\pm1.03	49.65\pm1.41	92.5\pm1.37	72.99\pm2.66	86.21\pm2.5	47.37\pm3.32	67.07\pm3.19

TABLE 3. Clustering results (NMI% \pm std%) of the compared methods.

Datasets	ORL	CBCL	Face94	Face95	Grimace	Carcinom	LUNG	TOX 171	MNIST
NMF	77.21 \pm 1.65	68.59 \pm 1.64	92.46 \pm 0.59	61.49 \pm 1.04	91.00 \pm 2.08	42.57 \pm 2.56	28.17 \pm 4.13	13.61 \pm 5.89	41.62 \pm 4.11
L ₂₁ NMF	79.25 \pm 1.23	71.96 \pm 1.90	93.08 \pm 0.56	64.03 \pm 1.14	94.47 \pm 1.92	43.43 \pm 2.43	26.07 \pm 4.2	14.91 \pm 5.65	40.34 \pm 4.47
GNNMF	81.93 \pm 1.68	69.76 \pm 1.35	92.74 \pm 0.25	64.54 \pm 1.03	94.59 \pm 2.09	66.0 \pm 2.1	56.27 \pm 3.79	15.64 \pm 6.11	62.20 \pm 3.98
MNMFL ₂₁	82.78 \pm 0.87	72.19 \pm 2.51	93.17 \pm 0.63	66.22 \pm 0.69	94.60 \pm 1.87	69.05 \pm 1.93	56.27 \pm 3.88	16.59 \pm 6.76	63.15 \pm 4.05
ONMFS	81.49 \pm 1.44	72.76 \pm 2.04	92.49 \pm 0.75	65.3 \pm 1.26	94.29 \pm 1.4	49.05 \pm 2.78	40.78 \pm 5.39	16.49 \pm 6.55	46.47 \pm 4.27
NMF-LCAG	80.55 \pm 1.42	69.53 \pm 2.69	92.08 \pm 0.72	62.72 \pm 0.93	93.46 \pm 2.32	55.55 \pm 2.51	52.58 \pm 4.41	16.37 \pm 6.43	56.36 \pm 4.36
ENMF	83.43 \pm 1.34	72.21 \pm 2.32	92.75 \pm 0.86	67.38 \pm 1.17	94.62 \pm 1.13	59.32 \pm 2.77	56.34 \pm 3.26	15.45 \pm 6.84	59.54 \pm 3.54
GLNMF	82.24 \pm 1.14	70.96 \pm 2.74	93.24 \pm 0.57	65.26 \pm 1.12	94.77 \pm 2.01	67.21 \pm 2.18	56.9 \pm 3.51	14.58 \pm 6.21	57.22 \pm 3.80
CHNMF	86.4 \pm 1.65	74.15 \pm 2.66	93.17 \pm 0.58	68.06 \pm 0.54	93.07 \pm 2.77	61.67 \pm 2.89	51.78 \pm 4.43	17.02 \pm 4.33	62.88 \pm 3.7
DSNMF	85.54 \pm 1.46	71.57 \pm 2.21	93.33 \pm 1.65	64.21 \pm 1.33	94.49 \pm 2.43	68.09 \pm 2.72	64.59 \pm 3.88	16.44 \pm 5.65	55.57 \pm 4.09
Ours	87.15\pm1.18	78.48\pm1.62	93.83\pm0.42	68.41\pm0.84	96.98\pm0.57	74.16\pm1.89	67.48\pm3.44	26.66\pm5.31	64.28\pm3.24

C. COMPARED METHODS

A baseline and state-of-the-art NMF approaches are exploited to compare with our RGNMF. Here, we briefly summarize them as follows:

- (1) NMF [8]: Standard NMF is considered as a baseline.
- (2) L₂₁NMF [14]: A L_{2,1}-norm loss function in NMF is exploited to decompose the input matrix into two nonnegative factor matrices.
- (3) GNNMF [10]: It takes advantage of the local structure of data to guide the process of matrix factorization.
- (4) MNMFL₂₁ [28]: It merges the local structure into L₂₁NMF as a regularizer to enhance the robustness of GNNMF.
- (5) ONMFS [47]: It presents nonnegative PCA algorithm to solve the orthogonal NMF with global approximation guarantees.
- (6) NMF-LCAG [34]: It integrates graph construction into the processes of matrix factorization and thus the graph structure changes during the NMF procedure.
- (7) ENMF [3]: Its loss function is intercalated between Frobenius norm and L_{2,1}-norm and adds the local structure of data as the regularization term.
- (8) GLNMF [31]: It uses low-rank recovery technique to obtain the low-rank part of the raw data and then incorporates the local structure into NMF for factorizing the low-rank data.
- (9) CHNMF [65]: It exploits the correntropy measure in the loss function and constructed a hypergraph to preserve the high-order geometric information of the data.

- (10) DSNMF [56]: it outlines a deep framework to learn such hidden representation and to interpret clustering according to different unknown attributes of a given data set.

D. COMPARISON OF CLUSTERING PERFORMANCE

Because nine compared approaches are unsupervised, clustering comparison is performed on the whole sample space. For fair comparison, we exploit a random scheme to initialize two nonnegative factors A and X . Following [3], [23], and [28], the clustering number K is set to the real number of classes belonging to one data set and the classical K -means is applied to cluster the learned representation X for obtaining the clustering results. MNMFL₂₁ and GNNMF have the regularization parameter λ and the nearest neighbor size p . As described in their experiments, λ changes within $\{10^{-3}, 10^{-2}, 10^{-1}, 10^0, 10^1, 10^2, 10^3\}$ and the best values of ACC and NMI are shown. In addition to the above two parameters, GNLMF has α and r that need to be assigned in advance. According to [31], α and r are assigned 10^{-4} and $0.1 \times \min(m, n)$, respectively. There are four regularization parameters, α , β , λ , and γ in NMF-LCAG. The values of α , β , λ , and γ are searched from $\{10^{-3}, 10^{-2}, 10^{-1}, 10^0, 10^1, 10^2, 10^3\}$ and the best values of ACC and NMI are shown. For ENMF, its three parameters δ , α , and β are searched from $\{10^{-3}, 10^{-2}, 10^{-1}, 10^0, 10^1, 10^2, 10^3\}$. In addition to p , the proposed RGNMF has three regularization parameters, α , β , and λ . Three parameters are selected from the grid $\{10^{-3}, 10^{-2}, 10^{-1}, 10^0, 10^1, 10^2, 10^3\}$. For GNNMF, MNMFL₂₁, NMF-LCAG, ENMF, GNLMF and RGNMF, the

neighborhood size p is set to 5 in all experiments. Experiments are performed 20 runs with different initial points on each data set. We show the average clustering result for each algorithm. Finally, following state-of-the-art methods [3], [28], [47], [48], we set the dimension l to the number of clusters in the experiments.

The values of ACC and NMI on eight data sets used in this paper are demonstrated in Tables 2-3. From Tables 2-3, a few interesting observations can be gained.

- (1) Obviously, our RGNMF performs differently on different data sets. For example, on Face94 data set, the ACC value of RGNMF is 1.28% higher than that of the second best GLNMF. On LUNG data set, the NMI value of RGNMF is 10.58% higher than that of the second best GLNMF. For larger MNIST data set, our RGNMF is superior to these methods compared in this paper. Obviously, it can be seen that our algorithm always achieves relatively better performance than other ten compared approaches. Thus, RGNMF can find more discriminative representations for the real-world data. Intuitively, the constructed k-NN graph and sparse graph that are imposed on the basis matrix and the encoding matrix play an essential role in finding the discriminative representation.
- (2) CHNMF is inferior to our RGNMF. The reason is that CHNMF constructs the Hypergraph to preserve the local structure of the data, while ignoring the global structure. It performs better than other methods on ORL, CBCL, and Face95 data sets. CHNMF can also provide the comparative performance on Face94, LUNG, TOX_171 and MNIS. However, its performance is lower than that of MNMFL₂₁ on Grimace, and Carcinom data sets. In a word, although CHNMF is slightly worse than our algorithm, it is a very good method. DSNMF also achieves good clustering performance. For example, it outperforms other methods on Face94 and Grimace data sets. For ORL, Face95, LUNG and MNIST data sets, however, CHNMF is superior to DSNMF. Generally, deep methods suffer from two limitations: one is that they usually involve a large number of parameters and are easy to fit; the other is that they require very high training cost in running time and space [69], [70].
- (3) Although MNMFL₂₁, ENMF and GNLMF form the remaining five algorithms including NMF, L₂₁NMF, ONMFS, NMF-LCAG and GNMF, they are inferior to the proposed method. The reason is that three algorithms pay attention to the local structure and ignore the global structure. In addition, their regularizers are imposed on the encoding matrix, which fails to directly affect the base matrix. We can see that the three algorithms perform almost well on most data sets. However, both MNMFL₂₁ and GNLMF perform better than ENMF on the Carcinom data set.
- (4) GNMF outperforms NMF, L₂₁NMF, and NMF-LCAG, because it constructs a nearest neighbor

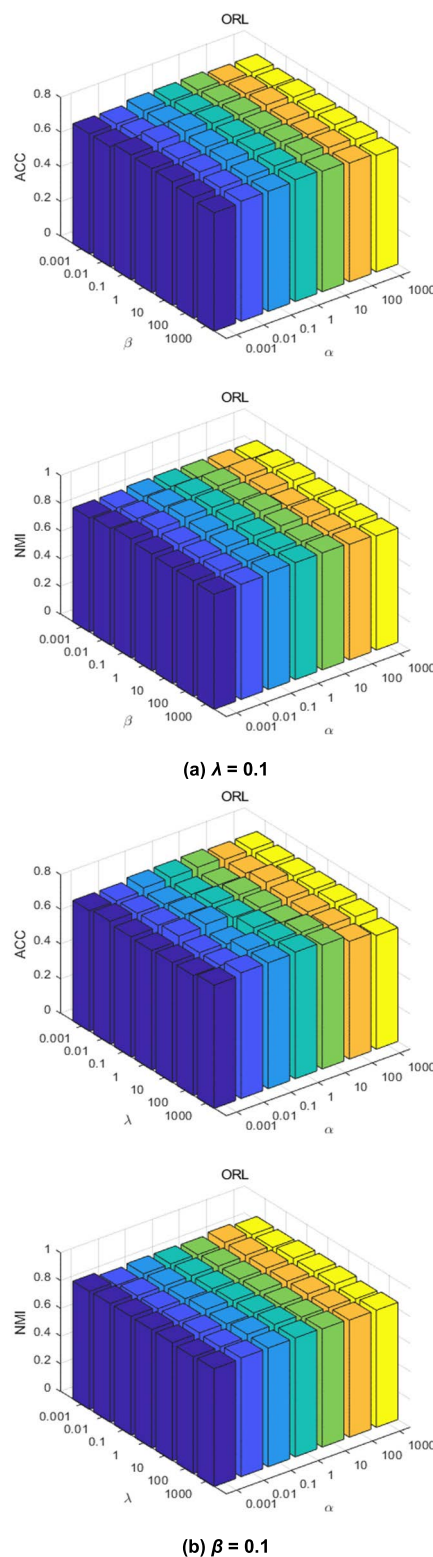
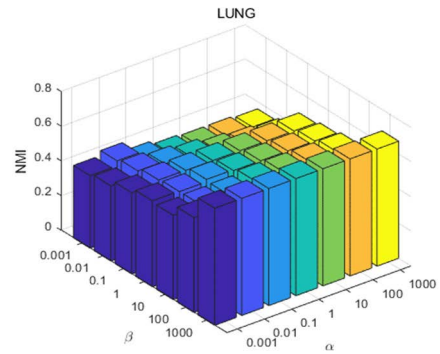
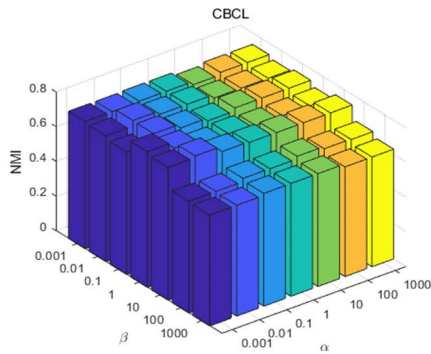
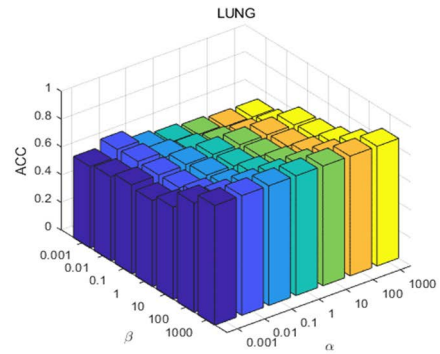
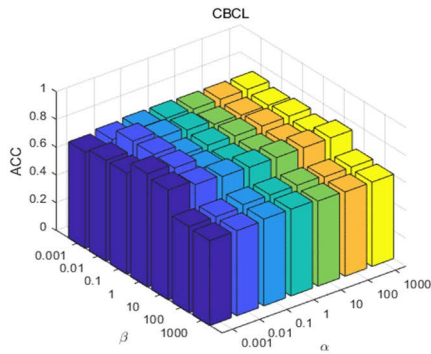


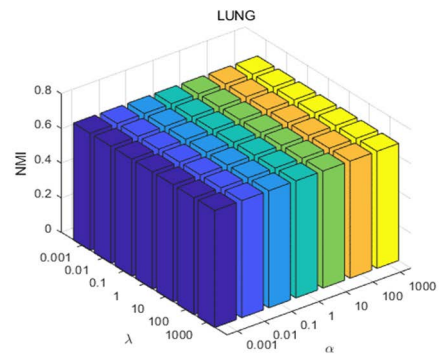
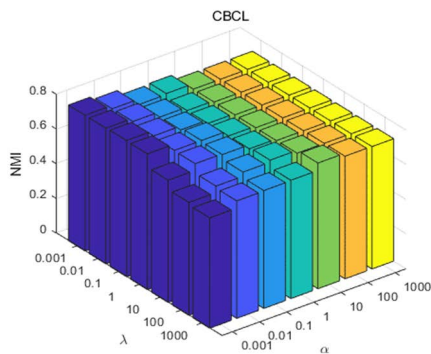
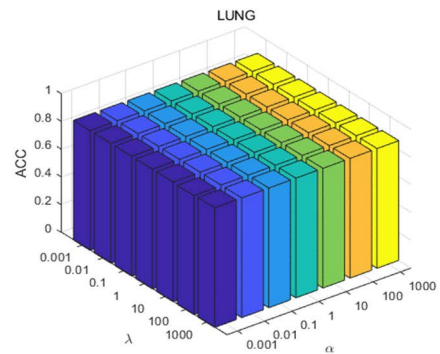
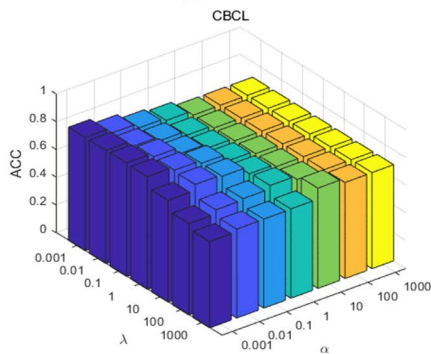
FIGURE 2. Performance of RGNMF with different α , β and λ on ORL data set.

graph to encode the geometric information of the data space. However, its performance is relatively worse than MNMFL₂₁, ENMF, GNLMF and RGNMF.



(a) $\lambda = 0.1$

(a) $\lambda = 0.1$



(b) $\beta = 0.1$

(b) $\beta = 1000$

FIGURE 3. Clustering performance of RGNMF with different α , β and λ on CBCL data set.

FIGURE 4. Clustering performance of RGNMF with different α , β and λ on LUNG data set.

As demonstrated in [28], the $L_{2,1}$ -norm is helpful to distinguish noise and outliers around clusters and enhance the clustering accuracy. GNMF incorporates

the local structure into the least square loss function of NMF. Thus, its performance is limited to some extent.

(5) As we can see, the performance of NMF-LCAG is lower than that of other manifold learning approaches,

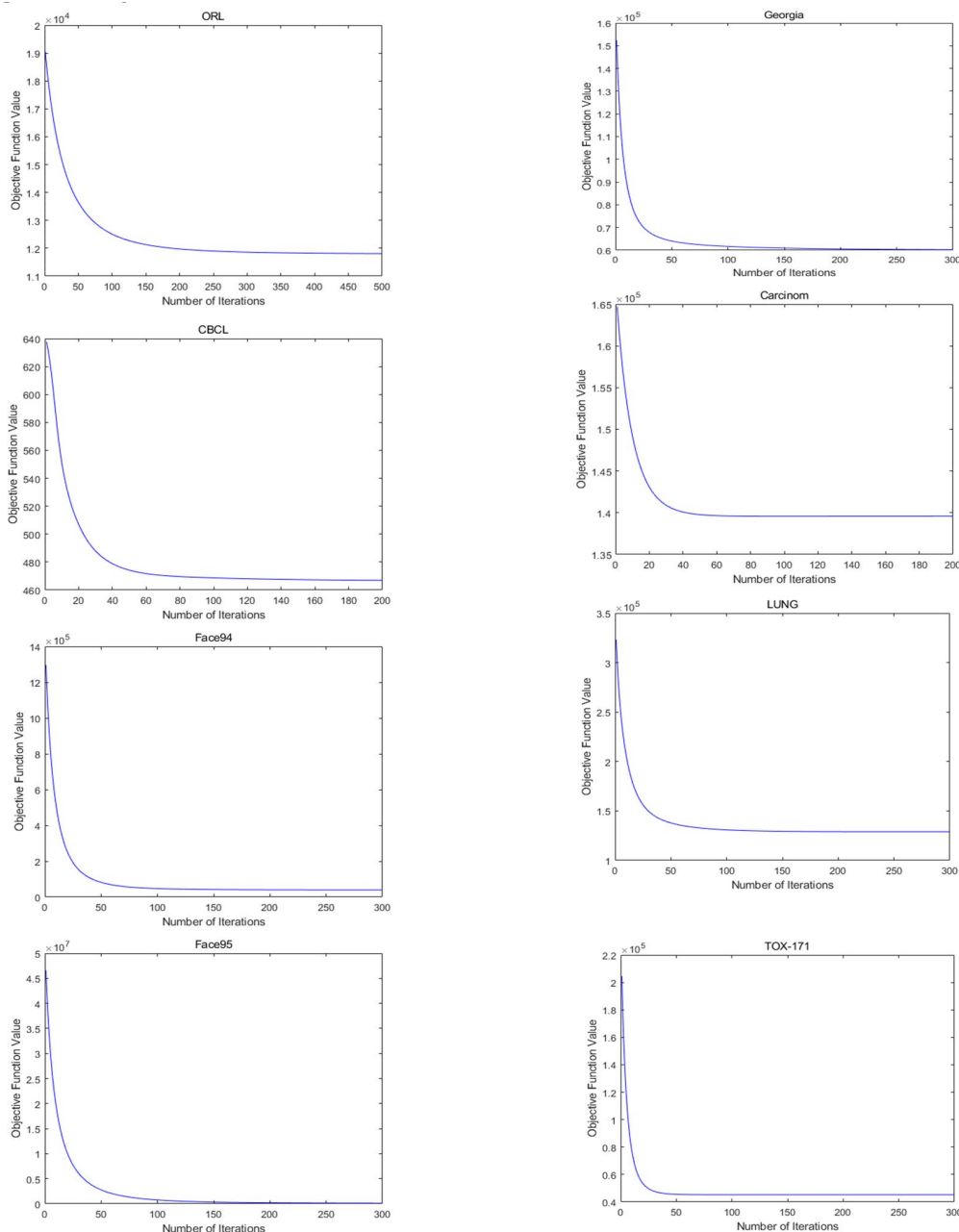


FIGURE 5. Convergence curves of our RGNMF algorithm on eight data sets.

even though it constructs the graph to consider the intrinsic geometric structure. The reason is that NMF-LCAG exploits self-expressive coefficients to depict the local structure of data, which may not be conducive to improving clustering performance. In addition, the optimal solution cannot be gained, since two regularization terms are controlled by one parameter λ [18].

- (6) As a baseline, NMF is simple to perform, but it usually performs worse than other eight methods. Thus, it is necessary to introduce different regularization terms into NMF to improve the performance.

The performance of ONMF is better than that of L21NMF in six data sets and NMF in all data sets, which validates that orthogonal constraint in NMF performs well for clustering tasks [3].

- (7) It is worth noting that NMF performs almost as well as manifold learning-based algorithms on the Face94 and TOX_171 data sets, but it is much worse than RGNMF, especially on the TOX_171 data set. This indicates that is not enough to improve the performance of NMF only by encoding the local structure, but also by considering other information, such as the global structure.

E. PARAMETER SENSITIVENESS

As mentioned in the previous section, in addition to p , there are three regularization parameters α , β , and λ in RGNMF. Clearly, if all three parameters are set to 0, our RGNMF is converted to L_{21} NMF [14]. If α and λ are set to 0, RGNMF is changed to MNMFL $_{21}$ [28]. Consequently, our RGNMF is more general than L_{21} NMF and MNMFL $_{21}$. In other words, MNMFL $_{21}$ and L_{21} NMF are two special cases of our RGNMF. To demonstrate the influence of three parameters α , β and λ on the performance of our algorithm, we perform the sensitivity experiments on ORL, CBCL and LUNG data sets. We can gain similar insight on the remaining data sets and thus leave out them.

We can observe from Figs. 2-4 that these three parameters have a significant impact on the performance of RGNMF. Actually, such a significant impact also exists in other methods. As seen in experiments, the impact of parameters on the performance of the algorithm is different on different data sets. For example, α , β and λ have a relatively little impact on RGNMF on the ORL data set. When three parameters vary from 0.001 to 1000, the performance of RGNMF has little change. Clearly, the performance of the proposed method is relatively stable with respect to three parameters. It can be seen from Fig. 3 that RGNMF can achieve consistently good performance when α and β vary respectively in $[10^{-3}, 10^2]$ and $[10^{-2}, 10^1]$ on the CBCL data set. Compared with α and β , λ has a relatively little influence on RGNMF. RGNMF becomes stable and obtains the good performance when λ is less than 100. As we can see from Fig. 4, β has a relatively large influence on RGNMF on the LUNG data set. RGNMF performs relatively poorly when β is less than 900. However, when β is greater 900, RGNMF becomes very stable and outperforms other algorithms. Similarly, others manifold learning methods, including GMNF, NMF-LCAG, MNMFL $_{21}$, ENMF and GNLNF, perform relatively worse when the regularization parameter of the affinity graph in their objective function is less than 900. The five methods can get the best performance when the value of this parameter is set to 1000. For example, on the LUNG data set, the best ACC and NMI values of GNLNF are 80.79% and 56.9%, respectively. It is worth noting that GMNF and MNMFL $_{21}$ have the same clustering accuracy. The best ACC and NMI of the two algorithms are 79.8% and 56.27%, respectively. If the regularization parameter of the local structure in our objective function is set to 1000, the best ACC and NMI of RGNMF are 86.21% and 67.48%, respectively. Obviously, NMI of RGNMF is about 11% higher than that of GNLNF.

F. CONVERGENCE ANALYSIS

To intuitively understand the convergence of our RGNMF algorithm, we discuss the empirical results on its convergence. The convergence curves of RGNMF are shown in Fig. 5 on eight data sets. In each subfigure of Fig. 5, the y-axis denotes the value of the objective function and the x-axis represents the number of iterations. We can observe

that RGNMF usually converges within 150 iterations. This indicates that our algorithm converges relatively fast.

V. CONCLUSION

In this paper, we proposed a novel unsupervised NMF algorithm, called RGNMF. Different from existing approaches that ignore the geometric structure or only considers the local structure, first of all, a sparse graph is constructed to model the global structure imposed on the basis matrix, and a nearest neighbor graph is constructed to respect the local structure constrained on the encoding matrix. Secondly, RGNMF exploits the $L_{2,1}$ -norm loss function to seek the basis matrix and encoding matrix, which can avoid the interference of noise and outliers. Thirdly, it enforces a $L_{2,1}$ -norm regularization on the basis matrix to choose the important features of samples. Hence, the discriminative representations of data are simultaneously learned by explicitly exploiting the intrinsic structure and $L_{2,1}$ -norm minimization. Finally, the optimization approach is developed to seek two factor matrices. And its convergence is proven. Experiments on eight real-world data sets demonstrate that RGNMF is better than the other eight algorithms.

In the next work, we will focus on the following questions: 1) There are three parameters α , β and λ which control the smoothness and the sparsity of the new model. Obviously, the proper values of the three parameters are very important to our algorithm. However, it is not clear how to select parameters effectively in theory. 2) Although it is difficult to obtain all the class labels of data, some of the class labels are available. We will consider representation learning from labeled and unlabeled data and naturally extend our RGNMF to semi-supervised scenarios.

ACKNOWLEDGMENT

(Qi Huang and Guodao Zhang are co-first authors.)

REFERENCES

- [1] B. Gao, W. L. Woo, and B. W.-K. Ling, "Machine learning source separation using maximum a posteriori nonnegative matrix factorization," *IEEE Trans. Cybern.*, vol. 44, no. 7, pp. 1169–1179, Jul. 2014.
- [2] D. Wang, F. Nie, and H. Huang, "Fast robust non-negative matrix factorization for largescale human action data clustering," in *Proc. 26th Int. Joint Conf. Artif. Intell. (IJCAI)*, Jul. 2016, pp. 2104–2110.
- [3] H. Xiong and D. Kong, "Elastic nonnegative matrix factorization," *Pattern Recognit.*, vol. 90, pp. 464–475, Jun. 2019.
- [4] V. Gligorijevic, Y. Panagakis, and S. P. Zafeiriou, "Non-negative matrix factorizations for multiplex network analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 4, pp. 928–940, Apr. 2019.
- [5] X. Pei, T. Wu, and C. Chen, "Automated graph regularized projective nonnegative matrix factorization for document clustering," *IEEE Trans. Cybern.*, vol. 44, no. 10, pp. 1821–1831, Oct. 2014.
- [6] N. Guan, D. Tao, Z. Luo, and B. Yuan, "NENMF: An optimal gradient method for nonnegative matrix factorization," *IEEE Trans. Signal Process.*, vol. 60, no. 6, pp. 2882–2898, Jun. 2012.
- [7] Q. Huang, X. Yin, S. Chen, Y. Wang, and B. Chen, "Robust nonnegative matrix factorization with structure regularization," *Neurocomputing*, vol. 412, pp. 72–90, Oct. 2020.
- [8] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, no. 6755, pp. 788–791, Oct. 1999.
- [9] C. Peng, Z. Zhang, Z. Kang, C. Chen, and Q. Cheng, "Nonnegative matrix factorization with local similarity learning," *Inf. Sci.*, vol. 562, pp. 325–346, Jul. 2021.

- [10] D. Cai, X. He, J. Han, and T. S. Huang, "Graph regularized nonnegative matrix factorization for data representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 8, pp. 1548–1560, Aug. 2011.
- [11] R. Zhi, M. Flierl, Q. Ruan, and W. B. Kleijn, "Graph-preserving sparse nonnegative matrix factorization with application to facial expression recognition," *IEEE Trans. Syst., Man, Cybern., B, Cybern.*, vol. 41, no. 1, pp. 38–52, Jan. 2011.
- [12] Y. Lu, Z. Lai, Y. Xu, X. Li, D. Zhang, and C. Yuan, "Nonnegative discriminant matrix factorization," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 27, no. 7, pp. 1392–1405, Jul. 2017.
- [13] P. Li, J. Bu, Y. Yang, R. Ji, C. Chen, and D. Cai, "Discriminative orthogonal nonnegative matrix factorization with flexibility for data representation," *Expert Syst. Appl.*, vol. 41, no. 4, pp. 1283–1293, Apr. 2014.
- [14] D. Kong, C. Ding, and H. Huang, "Robust nonnegative matrix factorization using L_{21} -norm," in *Proc. 20th ACM Int. Conf. Inf. Knowl. Manage. (CIKM)*, Oct. 2011, pp. 673–682.
- [15] S. Zafeiriou, A. Tefas, I. Buciu, and I. Pitas, "Exploiting discriminant information in nonnegative matrix factorization with application to frontal face verification," *IEEE Trans. Neural Netw.*, vol. 17, no. 3, pp. 683–695, May 2006.
- [16] N. Guan, D. Tao, Z. Luo, and B. Yuan, "Manifold regularized discriminative nonnegative matrix factorization with fast gradient descent," *IEEE Trans. Image Process.*, vol. 20, no. 7, pp. 2030–2048, Jul. 2011.
- [17] S. An, J. Yoo, and S. Choi, "Manifold-respecting discriminant nonnegative matrix factorization," *Pattern Recognit. Lett.*, vol. 32, no. 6, pp. 832–837, Apr. 2011.
- [18] X. Li, G. Cui, and Y. Dong, "Discriminative and orthogonal subspace constraints-based nonnegative matrix factorization," *ACM Trans. Intell. Syst. Technol.*, vol. 9, no. 6, pp. 1–24, Nov. 2018.
- [19] S. Nikitidis, A. Tefas, N. Nikolaidis, and I. Pitas, "Subclass discriminant nonnegative matrix factorization for facial image analysis," *Pattern Recognit.*, vol. 45, no. 12, pp. 4080–4091, Dec. 2012.
- [20] H. Liu, Z. Wu, X. Li, D. Cai, and T. S. Huang, "Constrained nonnegative matrix factorization for image representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 7, pp. 1299–1311, Jul. 2012.
- [21] X. Zhang, L. Zong, X. Liu, and J. Luo, "Constrained clustering with nonnegative matrix factorization," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 7, pp. 1514–1526, Jul. 2016.
- [22] D. Wang, X. Gao, and X. Wang, "Semi-supervised nonnegative matrix factorization via constraint propagation," *IEEE Trans. Cybern.*, vol. 46, no. 1, pp. 233–244, Jan. 2016.
- [23] Z. Li, J. Tang, and X. He, "Robust structured nonnegative matrix factorization for image representation," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 5, pp. 1947–1960, May 2018.
- [24] M. Gong, X. Jiang, H. Li, and K. C. Tan, "Multiobjective sparse nonnegative matrix factorization," *IEEE Trans. Cybern.*, vol. 49, no. 8, pp. 2941–2954, Aug. 2019.
- [25] C. Peng, Z. Zhang, C. Chen, Z. Kang, and Q. Cheng, "Two-dimensional semi-nonnegative matrix factorization for clustering," *Inf. Sci.*, vol. 590, pp. 106–141, Apr. 2022.
- [26] P. O. Hoyer, "Non-negative matrix factorization with sparseness constraints," *J. Mach. Learn. Res.*, vol. 5, pp. 1457–1469, May 2004.
- [27] J. Huang, F. Nie, H. Huang, and C. Ding, "Robust manifold nonnegative matrix factorization," *ACM Trans. Knowl. Discovery Data*, vol. 8, no. 3, pp. 11–31, Jun. 2014.
- [28] B. Wu, E. Wang, Z. Zhu, W. Chen, and P. Xiao, "Manifold NMF with L_{21} norm for clustering," *Neurocomputing*, vol. 273, pp. 78–88, Jan. 2018.
- [29] Z. Zhang and K. Zhao, "Low-rank matrix approximation with manifold regularization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 7, pp. 1717–1729, Jul. 2013.
- [30] Y. Liu, Y. Liao, L. Tang, F. Tang, and W. Liu, "General subspace constrained non-negative matrix factorization for data representation," *Neurocomputing*, vol. 173, pp. 224–232, Jan. 2016.
- [31] X. Li, G. Cui, and Y. Dong, "Graph regularized non-negative low-rank matrix factorization for image clustering," *IEEE Trans. Cybern.*, vol. 47, no. 11, pp. 3840–3853, May 2017.
- [32] S. Huang, Z. Xu, and F. Wang, "Nonnegative matrix factorization with adaptive neighbors," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, May 2017, pp. 486–493.
- [33] L. Zhang, Q. Zhang, B. Du, J. You, and D. Tao, "Adaptive manifold regularized matrix factorization for data clustering," in *Proc. 26th Int. Joint Conf. Artif. Intell.*, Aug. 2017, pp. 3399–3405.
- [34] Y. Yi, J. Wang, W. Zhou, C. Zheng, J. Kong, and S. Qiao, "Non-negative matrix factorization with locality constrained adaptive graph," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 2, pp. 427–441, Feb. 2020.
- [35] Y. Lu, Z. Lai, Y. Xu, J. You, X. Li, and C. Yuan, "Projective robust nonnegative factorization," *Inf. Sci.*, vols. 364–365, pp. 16–32, Oct. 2016.
- [36] K. Wang, R. He, L. Wang, W. Wang, and T. Tan, "Joint feature selection and subspace learning for cross-modal retrieval," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 10, pp. 2010–2023, Oct. 2016.
- [37] N. Zhou, Y. Xu, H. Cheng, J. Fang, and W. Pedrycz, "Global and local structure preserving sparse subspace learning: An iterative approach to unsupervised feature selection," *Pattern Recognit.*, vol. 53, pp. 87–101, May 2016.
- [38] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 2, pp. 210–227, Feb. 2009.
- [39] B. Cheng, J. Yang, S. Yan, Y. Fu, and T. S. Huang, "Learning with ℓ^1 -graph for image analysis," *IEEE Trans. Image Process.*, vol. 19, no. 4, pp. 858–866, Apr. 2010.
- [40] L. Qiao, S. Chen, and X. Tan, "Sparsity preserving projections with applications to face recognition," *Pattern Recognit.*, vol. 43, no. 1, pp. 331–341, 2010.
- [41] Q. Yang, X. Yin, S. Kou, and Y. Wang, "Robust structured convex nonnegative matrix factorization for data representation," *IEEE Access*, vol. 9, pp. 155087–155102, 2021.
- [42] X. He and P. Niyogi, "Locality preserving projections," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, Dec. 2003, pp. 153–160.
- [43] Z. Li, J. Liu, J. Tang, and H. Lu, "Robust structured subspace learning for data representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 10, pp. 2085–2098, Aug. 2015.
- [44] J. Yang, D. Zhang, J.-Y. Yang, and B. Niu, "Globally maximizing, locally minimizing: Unsupervised discriminant projection with applications to face and palm biometrics," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 4, pp. 650–664, Apr. 2007.
- [45] Y. Yi, J. Wang, W. Zhou, Y. Fang, J. Kong, and Y. Lu, "Joint graph optimization and projection learning for dimensionality reduction," *Pattern Recognit.*, vol. 92, pp. 258–273, Aug. 2019.
- [46] F. Nie, H. Huang, X. Cai, and C. Ding, "Efficient and robust feature selection via joint $\ell^2, 1$ -norms minimization," in *Proc. Adv. Neural Inf. Process. Syst.*, Vancouver, BC, Canada, Dec. 2010, pp. 1813–1821.
- [47] M. Asteris, D. Papailiopoulos, and A. G. Dimakis, "Orthogonal NMF through subspace exploration," in *Proc. Adv. Neural Inf. Process. Syst.*, Dec. 2015, pp. 343–351.
- [48] W.-S. Chen, Q. Zeng, and B. Pan, "A survey of deep nonnegative matrix factorization," *Neurocomputing*, vol. 491, pp. 305–320, Jun. 2022.
- [49] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, Nov. 2010.
- [50] E. Elhamifar and R. Vidal, "Sparse subspace clustering: Algorithm, theory, and applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 11, pp. 2765–2781, Nov. 2013.
- [51] J. Wen, X. Fang, Y. Xu, C. Tian, and L. Fei, "Low-rank representation with adaptive graph regularization," *Neural Netw.*, vol. 108, pp. 83–96, Aug. 2018.
- [52] C.-G. Li, C. You, and R. Vidal, "Structured sparse subspace clustering: A joint affinity learning and subspace clustering framework," *IEEE Trans. Image Process.*, vol. 26, no. 6, pp. 2988–3001, Jun. 2017.
- [53] Z. Lin, M. Chen, and Y. Ma, "The augmented Lagrange multiplier method for exact recovery of corrupted low-rank matrices," 2010, *arXiv:1009.5055*.
- [54] C. Peng and Q. Cheng, "Discriminative ridge machine: A classifier for high-dimensional data or imbalanced data," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 6, pp. 2595–2609, Jun. 2021.
- [55] C. Peng, Y. Zhang, Y. Chen, Z. Kang, C. Chen, and Q. Cheng, "Log-based sparse nonnegative matrix factorization for data representation," *Knowl.-Based Syst.*, vol. 251, Sep. 2022, Art. no. 109127.
- [56] G. Trigeorgis, K. Bousmalis, S. Zafeiriou, and B. W. Schuller, "A deep matrix factorization method for learning attribute representations," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 3, pp. 417–429, Mar. 2017, doi: [10.1109/TPAMI.2016.2554555](https://doi.org/10.1109/TPAMI.2016.2554555).
- [57] Z. Ren, W. Zhang, and Z. Zhang, "A deep nonnegative matrix factorization approach via autoencoder for nonlinear fault detection," *IEEE Trans. Ind. Informat.*, vol. 16, no. 8, pp. 5042–5052, Aug. 2020.

- [58] Y. Zhao, H. Wang, and J. Pei, "Deep non-negative matrix factorization architecture based on underlying basis images learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 6, pp. 1897–1913, Jun. 2021.
- [59] W. Guo, "Sparse dual graph-regularized deep non-negative matrix factorization for image clustering," *IEEE Access*, vol. 9, pp. 39926–39938, 2021.
- [60] Y. Meng, R. Shang, F. Shang, L. Jiao, and R. Stolkin, "Semi-supervised graph regularized deep NMF with bi-orthogonal constraints for data representation," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 3, no. 1, pp. 13–28, Dec. 2019.
- [61] S. Huang, Z. Kang, and Z. Xu, "Auto-weighted multi-view clustering via deep matrix decomposition," *Pattern Recognit.*, vol. 97, Jan. 2020, Art. no. 107015.
- [62] B. Weyrauch, B. Heisele, J. Huang, and V. Blanz, "Component-based face recognition with 3D morphable models," in *Proc. Conf. Comput. Vis. Pattern Recognit. Workshop*, Washington, DC, USA, Jun. 2004, p. 85.
- [63] L. Chen, H. Man, and A. V. Nefian, "Face recognition based on multi-class mapping of Fisher scores," *Pattern Recognit.*, vol. 38, no. 6, pp. 799–811, Jun. 2005.
- [64] C.-N. Jiao, Y.-L. Gao, N. Yu, J.-X. Liu, and L.-Y. Qi, "Hyper-graph regularized constrained NMF for selecting differentially expressed genes and tumor classification," *IEEE J. Biomed. Health Informat.*, vol. 24, no. 10, pp. 3002–3011, Feb. 2020.
- [65] N. Yu, M.-J. Wu, J.-X. Liu, C.-H. Zheng, and Y. Xu, "Correntropy-based hypergraph regularized NMF for clustering and feature selection on multi-cancer integrated data," *IEEE Trans. Cybern.*, vol. 51, no. 8, pp. 3952–3963, Aug. 2021.
- [66] Z. Kang, Z. Lin, X. Zhu, and W. Xu, "Structured graph learning for scalable subspace clustering: From single view to multiview," *IEEE Trans. Cybern.*, early access, Mar. 17, 2021, doi: [10.1109/TCYB.2021.3061660](https://doi.org/10.1109/TCYB.2021.3061660).
- [67] Z. Kang, W. Zhou, Z. Zhao, J. Shao, M. Han, and Z. Xu, "Large-scale multi-view subspace clustering in linear time," in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 4412–4419.
- [68] Y. Chen, C.-G. Li, and C. You, "Stochastic sparse subspace clustering," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 4154–4163.
- [69] Z. Lin, Z. Kang, L. Zhang, and L. Tian, "Multi-view attributed graph clustering," *IEEE Trans. Knowl. Data Eng.*, early access, Aug. 6, 2021, doi: [10.1109/TKDE.2021.3101227](https://doi.org/10.1109/TKDE.2021.3101227).
- [70] R. Liu, G. Zhang, J. Wang, and S. Zhao, "Cross-modal 360° depth completion and reconstruction for large-scale indoor environment," *IEEE Trans. Intell. Transp. Syst.*, early access, Mar. 14, 2022, doi: [10.1109/TITS.2022.3155925](https://doi.org/10.1109/TITS.2022.3155925).



GUODAO ZHANG received the B.S. degree from Wenzhou University, Wenzhou, China, in 2010, and the Ph.D. degree from the Zhejiang University of Technology, Hangzhou, China, in 2022. He is currently a Lecturer with the Department of Digital Media Technology, Hangzhou Dianzi University, Hangzhou. His main research interests include computational intelligence, process modeling, and data mining.



XUESONG YIN received the Ph.D. degree in computer science from the Nanjing University of Aeronautics and Astronautics, Nanjing, China, in 2010. He is currently a Professor with the Department of Digital Media Technology, Hangzhou Dianzi University, Hangzhou, China. His main research interests include machine learning, data mining, image processing, and pattern recognition.



QI HUANG received the B.S. and M.S. degrees in pharmaceutical analysis from China Pharmaceutical University, Nanjing, China. She is currently a Lecturer with the School of Biological and Chemical Engineering, Zhejiang University of Science and Technology, Zhejiang, China. Her research interests include biology computing, machine learning, and computation pharmacy.



YIGANG WANG received the M.S. and Ph.D. degrees in applied mathematics from Zhejiang University, Hangzhou, China. He is currently a Professor with the Department of Digital Media Technology, Hangzhou Dianzi University, Hangzhou. His research interests include image processing, computer vision, pattern recognition, and computer graphics.

...