IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received January 25, 2021, accepted February 1, 2021, date of publication February 8, 2021, date of current version July 12, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3057901

Multi-Metric Induced Robust Graph

ZENGMIN GENG[1], XIAODONG SUN[2], JIANXIA DU[2], JUJIAN ZHANG[1], AND YILING ZHOU[1]
[1]Division of Basic Courses, Beijing Institute of Fashion Technology, Beijing 100029, China
[2]School of Fashion, Beijing Institute of Fashion Technology, Beijing 100029, China

Corresponding author: Zengmin Geng (jsjgzm@bift.edu.cn)

This work was supported by the Scientific Research Project of Beijing Educational Committee under Grant KM202010012008.

ABSTRACT Graph-based learning model has a wide range of applications in machine learning and computer vision. The key issue of the graph-based applications is to construct an informative graph to effectively represent data correlations. In practice, real-world data is usually contaminated by complex noise beyond Gaussian noise and sparse noise, which degrades learning performance dramatically. To construct a robust graph that represents real-world data distribution well, we propose a novel graph construction method. The proposed method is designed to be robust to complex noise beyond sparse or Gaussian noise by combining different previously given metrics of measuring data errors, which robustly and correctly generates a set of edges. To demonstrate the superiority of the proposed method, we employed the clustering and classification tasks on the real-world datasets. The experimental results demonstrate that the proposed method is superior to the existing methods in both accuracy and robustness to complex noise.

INDEX TERMS Graph, robust, image processing, noise.

I. INTRODUCTION

Graph has been used as an effective relationship modelling tool to modulate data correlations for solving various problems in machine learning and image processing community [1]–[4]. A graph uses a set of the vertices to represent data samples and links two related samples for modulating pairwise relationship between two samples. Thus, compared with other learning models, a graph-based learning model can effectively model the data correlations, resulting in promising performance for real applications.

The key issue of the graph-based applications is to construct an informative graph that modulates data correlations. Most graph construction methods adopt a neighborhood-based approach to generate a set of the edges [1]–[4]. In detail, this approach takes each data sample as centroid vertex and links it to its K-Nearest-Neighbors (KNN) to generate a set of edges. The neighborhood-based approach has two main drawbacks that remain unsolved: (1) Learning performance is sensitive to the neighborhood size in the k-nearest neighbor selection. A small neighborhood size separates data samples from the same cluster; on the contrary, a large neighborhood size combines data samples from different clusters. (2) Various types of noises may easily contaminate the real-world data, which degrades the learning performance dramatically.

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

To solve the drawbacks of the neighborhood-based approach, the representation-based approach [5]–[23] has been proposed, where each edge is generated by solving a linear regression problem. These methods usually imposes different constraints, e.g., sparse constraint [5], ℓ_2-norm [12] and low-rank [15] constraints, on the regression coefficients to capture the desired data structures. The representation-based approach separates noise components from original data, which chooses the samples with non-zero coefficients to generate the edges, resulting in a noise-resistant graph.

Note that compared with the neighborhood-based approach, the representation-based approach has achieved more promising performance, specifically for handling the noisy datasets. However, the existing representation-based graph construction methods are usually suitable to handle the data contaminated by Gaussian or sparse noise [13]. When the data contains mixed noise, such as outliers and complex noise, or densely corrupted, the performance of the graph-based applications is degraded dramatically.

For many real-world problems, *on the one hand*, sampled data is often contaminated by complex noise beyond Gaussian noise or sparse noise, Specifically for the case when the data contains occlusion, densely corruption, and distortion; *On the other hand*, the prior knowledge on noise is usually unknown in advance, such that it is difficult to design a noise removing term in constructing a representation-based graph.

Inspired by the recent advances of graph construction, we propose a novel robust graph construction method, where data errors or noise are estimated via linearly combining multiple metrics of measuring the data errors.

The contributions of our work are two fold:

1) The proposed method uses multiple candidate metrics with their weights to construct a candidate metric pool, where each candidate metric is considered as the initial estimation of data errors. Then, these previously given metrics are linearly combined to estimate the noise or data errors.
2) The proposed error estimation term is incorporated into the sparse regression framework, where the weights of the candidate metrics are jointly learned with sparse coefficients. Then, the samples with non-zero coefficients are chosen to generate a set of edges.

The rest of this article is organized as follows. Section 2 introduces related work. Section 3 presents the proposed multi-metric induced robust graph construction method. Section 4 present the optimization scheme. Section 5 reports experimental settings, results and discussions. Finally, Section 6 concludes the article.

II. RELATED WORK

Most of the existing graph construction methods adopt the neighborhood-based approach [1]–[4] to generate a set of edges via linking the nearest neighbors. Because this approach relies on pair-wise distance to build data correlations, it is ineffective to capture the global data structures. As a result, the neighborhood-based methods tend to be sensitive to noise and data corruption. Various methods are designed to capture the underlying manifold structures of the data [30]–[34]. To construct a noise-resistant graph, the representation-based graph construction approach adopts the linear regression to generate a set of the edges and compute the corresponding weights. According to the types of constraints imposed on the coefficients, the existing representation based graph construction methods can be divided into the following four schemes:

ℓ_1-*Graph* [5], [6] adopts sparse regression to construct a graph. To modulate the data structures, ℓ_1-Graph [7], [8] embeds the underlying structures of the data into the sparse representation framework. To modulate high dimensional feature space, kernel ℓ_1-Graph [9] adopts a kernel trick to construct a graph. Manifold regularized ℓ_1-Graph discovers the underlying geometry structure of data distribution. For instance, Zheng *et al*. [10] propose a manifold regularized sparse representation. Jin *et al*. [11] propose multiple graph regularized sparse coding, which adopts multiple graphs regularized sparse coding for the clustering task.

ℓ_2-*Graph* [12] adopts ridge regression instead of sparse representation to construct a graph, which has the following two advantages: (1) Because the derivation of the coefficients has a closed-form solution, ℓ_2-Graph is computationally efficient. (2) ℓ_2-Graph discovers the grouping structures of the data, which is suitable for clustering task. Furthermore, correntropy induced ℓ_2-Graph [13] adopts the correntropy induced metric instead of ℓ_2-norm to measure data errors, which performs the better than ℓ_2-Graph. Jin *et al*. [14] propose locality preserving collaborative representation, which achieves promising classification performance on some image datasets.

Low-rank Graph [15] adopts a Low-Rank Representation (LRR) to construct a graph. Because the low-rank constraint is imposed on the entire coefficients matrix, Low-rank Graph modulates the global linear structure of the data. Multiple-view low-rank representation is designed to simultaneously capture the latent subspace of multiple-view features [16]. When the amount of observed data is insufficient, latent low-rank Graph regards the hidden data, after being transposed, as input data matrix [17]. Adaptive Low-rank Graph [18] jointly learns the affinity matrix and the representation coefficients in a unified framework. Kernel Low-rank representation [19] adopts a kernel projection to find high-dimensional space, where data have possible low-rank structure. Manifold regularized Low-rank Graph incorporates a hypergraph regularization term into the low-rank representation framework, which considers local consistence of the learned low-rank representation [20].

To derive the simultaneous effects of different regularization, multiple constraints are imposed on the regression coefficients of linear regression, resulting in more resistant edges. For instance, Non-Negative Sparse Low-Rank Graph (NNSLR) [21] requires the data representation to be a non-negative low-rank and sparse matrix. Furthermore, a weighted ℓ_1-norm regularization term is incorporated into low-rank representation to model the local linear subspace structure of the data [22]. Low-rank Tensor Graph [23] enforces the representation not only sparse but also explores the complementary information of multi-view data.

In summary, the existing methods are usually suitable to cope with the Gaussian noise or sparse noise. In this article, we make an attempt to handle complex noise beyond the Gaussian and sparse noises for constructing a robust graph.

III. METHOD

Suppose there is a collection of N data samples drawn from a union of linear sub-spaces. For convenience, a collection of data samples is represented as $\mathbf{X} = [\mathbf{x}_1, \cdots, \mathbf{x}_i, \cdots, \mathbf{x}_N] \in \mathbb{R}^{M\times N}$, where $\mathbf{x}_i \in \mathbb{R}^M$ is the i-th sample.

A. MULTI-METRIC INDUCED ERROR ESTIMATION

For data lying on or near to the union of the linear subspaces, we adopt the "data self-expression" property to represent each sample as the linear combination of the data by itself, defined as follows:

$$\mathbf{X} = \mathbf{XC} + \sum_{i=1}^{P} \mathbf{E}_i, \tag{1}$$

where $\mathbf{C} = [\mathbf{c}_1\mathbf{c}_2 \cdots \mathbf{c}_N] \in \mathbb{R}^{N\times N}$ is the coefficients matrix, and $\mathbf{c}_i$ is the coefficients vector of the i-th sample;

$\mathbf{E}_i = [\mathbf{e}_1 \mathbf{e}_2 \cdots \mathbf{e}_N] \in \mathbb{R}^{M \times N}$ is the i-th component of the data errors matrix and $\mathbf{e}_i \in \mathbb{R}^M$ is the data errors vector of the i-th sample. The entire data errors $\mathbf{E} = \sum_{i=1}^{P} \mathbf{E}_i$ and P denotes the number of metrics of meaturing the data errors.

Because of noise and data corruption, the sampled real-world data usually deviate from the underlying subspace, which makes the data errors not equal to zeros. Thus, the key issue to represent the data correlations is the correct estimation of data errors.

The existing metrics are usually only effective to measure the specific data error. For instance, the ℓ_2-norm based metric is suitable to measure the Gaussian noise [6], the ℓ_1-norm based metric is suitable to measure the sparse noise [14] and the ℓ_21-norm based metric is suitable to estimate the sample-specific corruption [17]. When the data contains the complex noise in any distribution, or data noise is unknown in advance, it is difficult to estimate the corresponding noise components of the data.

To estimate the noise beyond the Gaussian noise or sparse noise, we would like to estimate the data errors by using the linear combination of multiple candidate metrics, defined as follows:

$$Error(E, \boldsymbol{\lambda}) = \min_{E,\boldsymbol{\lambda}} \sum_{i=1}^{P} \lambda_i M_i(E_i), \tag{2}$$

where $M_i(\cdot)$ denotes the i-th candidate metric to measure the data errors; $\boldsymbol{\lambda} = [\lambda_1, \lambda_2, \cdots, \lambda_i, \cdots, \lambda_P]$ is the weights vector of the candidate metrics, λ_i is the weight of the i-th candidate metric and P is the number of the candidate metrics. The λ_i with a larger value indicates the i-th candidate metric plays a more important role in estimating the data errors.

Furthermore, two constraints $\sum_{i=1}^{P} \lambda_i = 1, \boldsymbol{\lambda} \geq \mathbf{0}$, are imposed on weights of the candidate metrics for avoiding the negative contributions and assigning a natural probability interpretation. Finally, to avoid the error estimation over-fitting into the single candidate metric, the Shannon entropy term $\sum_{i=1}^{P} \lambda_i \ln \lambda_i$ is incorporated into the error estimation term, resulting in the following:

$$\begin{aligned} &\min_{\mathbf{E},\boldsymbol{\lambda}} \sum_{i=1}^{P} \lambda_i M_i(\mathbf{E}_i) + \gamma \sum_{i=1}^{P} \lambda_i \ln \lambda_i \\ &\text{s.t.} \sum_{i=1}^{P} \lambda_i = 1, \ \boldsymbol{\lambda} \geq \mathbf{0} \end{aligned} \tag{3}$$

where γ is the tradeoff parameter to balance the two terms.

In practice, different metrics are used to construct a candidate metrics pool, where each candidate metric is considered as the initial estimation of data errors. Then, the previously given candidate metrics in the metrics pool are combined linearly to estimate the real data errors; Finally, the error estimation term is defined in the enlarged parameter space.

For computational efficiency, we adopt only the following three metrics: the ℓ_1-norm based metric, the *Frobenius*-norm (ℓ_2-norm) based metric and the ℓ_21-norm based metric to construct a candidate metrics pool. Then, these three metrics are combined to estimate the real data errors, defined as follows:

$$\begin{aligned} &\min_{\mathbf{E},\boldsymbol{\lambda}} \lambda_1 \|\mathbf{E}_1\|_1 + \lambda_2 \|\mathbf{E}_2\|_{21} + \lambda_3 \|\mathbf{E}_3\|_F^2 + \gamma \sum_{\mathrm{i}=1}^{3} \lambda_{\mathrm{i}} \ln \lambda_{\mathrm{i}} \\ &\text{s.t.} \sum_{\mathrm{i}=1}^{3} \lambda_{\mathrm{i}} = 1, \ \boldsymbol{\lambda} \geq \mathbf{0} \end{aligned} \tag{4}$$

where the Frobenius-norm (ℓ_2-norm) based metric is used to measure the Gaussian noise; the ℓ_1-norm based metric is used to measure the sparse noise and the ℓ_21-norm based metric is used to measure the sample-specific corruption.

B. MULTI-METRIC INDUCED DATA RECONSTRUCTION

In the following, we incorporate the proposed multi-metric induced error estimation term into the sparse representation framework. Based on the coefficients, a novel robust graph construction method is proposed.

$$\begin{aligned} \min_{\mathbf{E}_1,\mathbf{E}_2,\mathbf{E}_3,\mathbf{C},\boldsymbol{\lambda}} & \lambda_1 \|\mathbf{E}_1\|_1 + \lambda_2 \|\mathbf{E}_2\|_{21} + \lambda_3 \|\mathbf{E}_3\|_F^2 \\ & + \mu_1 \|\mathbf{C}\|_1 + \gamma \sum_{i=1}^{3} \lambda_i \ln \lambda_i \\ \text{s.t. } & \mathbf{E}_1 + \mathbf{E}_2 + \mathbf{E}_3 = \mathbf{X} - \mathbf{X}\mathbf{C}, \ \mathrm{diag}(\mathbf{C}) = \mathbf{0}, \\ & \sum_{k=1}^{3} \lambda_k = 1, \ \ \boldsymbol{\lambda} \geq \mathbf{0}. \end{aligned} \tag{5}$$

The objective function contains three components:

1) ERROR ESTIMATION TERMS

These terms are used to estimate the real data errors, where three norm-based candidate metrics are linearly combined to measure the data errors. Except for three metrics, a Shannon entropy term is used to avoid the error estimation over-fitting into the single candidate metric.

2) SPARSE REGULARIZATION TERM

To link a small number of neighbors to construct a sparse graph, the sparse regularization term is used to make few coefficients be non-zeros.

3) CONSTRAINTS

Three types of constraints are used: (a) Data reconstruction constraint, which represents each sample as the linear combination of the data. (b) The $\mathrm{diag}(\mathbf{C}) = \mathbf{0}$ constraint is imposed on the coefficients to eliminate the trivial solution of reconstructing each sample as itself. (c) The constraints are imposed on the weights of the candidate metrics to avoid negative contributions and give a natural probability interpretation for each weight.

The coefficients matrix characterizes how data samples contribute to data reconstruction. Such information is crucial to discover the underlying structures of the data.

The proposed error estimation term makes the derivation of coefficients robust to complex noise and corruption. Furthermore, the sparse regularization ensures that each sample is associated with only a few samples, which is suitable to construct a sparse graph structure. The constraints discover the underlying linear subspace of the data by imposing the constraints on the coefficients and weights of the candidate metrics.

C. EDGE GENERATION

The coefficients matrix derived by solving Eq. (5) is an asymmetric matrix. To leverage the existing graph-based frameworks, we employ the simple symmetrization operator to derive a symmetrical matrix: $C^* = (\mathbf{C} + \mathbf{C}^T)/2$.

For each vertex x_i, we link it to the samples with non-zero coefficients for generating a set of the edges. Then, the weight of each edge is defined as follows:

$$\mathbf{A}_{ij} = \left|\mathbf{c}_{ij}\right|, \tag{6}$$

where $\mathbf{A}_{ij}$ is the weight of an edge linking the sample x_i and x_j, $\mathbf{c}_{ij}$ is the j-th coefficient of $\mathbf{x}_i$.

Because the proposed graph construction method uses the error estimation based sparse representation for adaptive estimation of data errors, we refer to the proposed method as Ada-ℓ_1-Gr.

1) RELATIONSHIP BETWEEN THE PROPOSED METHOD AND ℓ_1-GRAPH

Elhamifar *et al.* proposed ℓ_1-Graph [6]. Different from ℓ_1-Graph, the proposed method fuses multiple candidate metrics to estimate the data errors; whereas, ℓ_1-Graph uses the *Frobenius*-norm based metric to estimate the Gaussian noise. Only when the *Frobenius*-norm based metric is within the candidate metrics pool to estimate the data errors, does the proposed method degenerate into ℓ_1-Graph. Thus, ℓ_1-Graph is referred to as a special case of the proposed model.

IV. OPTIMIZATION

Since the objective function is non-convex, it is a challenge to derive a global optimization solution. Instead, to derive local optimal solution, a number of optimization methods have been proposed. Because Alternating Direction Multiplier (ADM) [24] is suitable to handle the large-scale data, we adopt the ADM approach to optimize each variable separately.

A. THE ADM-BASED OPTIMIZATION

By removing $\mathbf{E}_3$, Eq. (5) can be rewritten as follows:

$$\begin{aligned}\min_{\mathbf{C},\mathbf{E}_1,\mathbf{E}_2,\boldsymbol{\lambda}} \lambda_1\|\mathbf{E}_1\|_1+\lambda_2\|\mathbf{E}_2\|_{21} + \lambda_3 \|\mathbf{X}-\mathbf{X}\mathbf{C}-\mathbf{E}_1 - \mathbf{E}_2\|_F^2 \\ +\mu\|\mathbf{C}\|_1 + \gamma\sum_{i=1}^{3}\lambda_i \ln\lambda_i \\ \text{s.t. } \operatorname{diag}(\mathbf{C}) = \mathbf{0}, \quad \sum_{k=1}^{3}\lambda_k = 1,\ \boldsymbol{\lambda} \geq \mathbf{0}.\end{aligned} \tag{7}$$

By introducing an auxiliary matrix $\mathbf{P} \in \mathbb{R}^{N\times N}$, we rewrite Eq. (7) as

$$\begin{aligned}\min_{\mathbf{P},\mathbf{C},\mathbf{E}_1,\mathbf{E}_2,\boldsymbol{\lambda}} \lambda_1\|\mathbf{E}_1\|_1 + \lambda_2\|\mathbf{E}_2\|_{21} + \lambda_3 \|\mathbf{X} - \mathbf{X}\mathbf{P} - \mathbf{E}_1 \\ -\mathbf{E}_2\|_F^2 + \mu\|\mathbf{C}\|_1 + \gamma\sum_{i=1}^{3}\lambda_i \ln\lambda_i \\ \text{s.t. } \mathbf{P} = \mathbf{C} - \operatorname{diag}(\mathbf{C}), \ \sum_{k=1}^{3}\lambda_k = 1,\ \boldsymbol{\lambda} \geq \mathbf{0}.\end{aligned} \tag{8}$$

To make the objective function strictly convex in terms of the optimization variables $(\mathbf{P}, \mathbf{C}, \mathbf{E_1}, \mathbf{E_2})$, one penalty term of the constraint is incorporated into the objective function of Eq. (8), resulting in

$$\begin{aligned}\min_{\mathbf{P},\mathbf{C},\mathbf{E}_1,\mathbf{E}_2,\boldsymbol{\lambda}} \lambda_1\|\mathbf{E}_1\|_1+\lambda_2\|\mathbf{E}_2\|_{21} + \lambda_3 \|\mathbf{X}-\mathbf{X}\mathbf{P}-\mathbf{E}_1-\mathbf{E}_2\|_F^2 \\ +\mu\|\mathbf{C}\|_1 + \gamma\sum_{i=1}^{3}\lambda_i \ln\lambda_i + \frac{\rho}{2}\|\mathbf{P} - (\mathbf{C} - \operatorname{diag}(\mathbf{C}))\|_F^2 \\ \text{s.t. } \mathbf{P} = \mathbf{C} - \operatorname{diag}(\mathbf{C}), \ \sum_{k=1}^{3}\lambda_k = 1,\ \boldsymbol{\lambda} \geq \mathbf{0},\end{aligned} \tag{9}$$

where $\rho > 0$ is the augmented Lagrange parameter.

By adding the Lagrange multiplier $\mathbf{M}$ for the equality constraint in Eq. (9), Eq. (9) is reformulated as

$$\begin{aligned}\min_{\mathbf{P},\mathbf{C},\mathbf{E}_1,\mathbf{E}_2,\boldsymbol{\lambda}} \lambda_1\|\mathbf{E}_1\|_1+\lambda_2\|\mathbf{E}_2\|_{21}+\lambda_3 \|\mathbf{X}-\mathbf{X}\mathbf{P} - \mathbf{E}_1-\mathbf{E}_2\|_F^2 \\ +\mu\|\mathbf{C}\|_1 + \gamma\sum_{i=1}^{3}\lambda_i \ln\lambda_i + \frac{\rho}{2}\|\mathbf{P} - (\mathbf{C} - \operatorname{diag}(\mathbf{C}))\|_F^2 \\ + tr(\mathbf{M}^T(\mathbf{P} - (\mathbf{C} - \operatorname{diag}(\mathbf{C})))) \\ \text{s.t. } \sum_{k=1}^{3}\lambda_k = 1,\ \boldsymbol{\lambda} \geq \mathbf{0},\end{aligned} \tag{10}$$

where $tr(\cdot)$ is the trace operator of a given matrix.

Eq. (10) can be solved by two separate steps: primal variable updating and dual ascending, i.e., the ADM approach iteratively updates the primal variables $(\mathbf{P}, \mathbf{C}, \mathbf{E_1}, \mathbf{E_2}, \boldsymbol{\lambda})$,and the Lagrange multiplier $\mathbf{M}$, to obtain the optimal solution. These variables are updated as follows:

- $(\mathbf{P} - \mathbf{Update})$: Updating $\mathbf{P}$ relies on the following problem by fixing the other variables and removing irrelevant terms:

$$\begin{aligned}\min_{\mathbf{P}} \lambda_3 \|\mathbf{X} - \mathbf{X}\mathbf{P} - \mathbf{E}_1 - \mathbf{E}_2\|_F^2 \\ + \frac{\rho}{2}\|\mathbf{P} - (\mathbf{C} - (diag(\mathbf{C}))\|_F^2 \\ + tr(\mathbf{M}^T(\mathbf{P} - (\mathbf{C} - \operatorname{diag}(\mathbf{C})))).\end{aligned} \tag{11}$$

By setting the derivative of the objective function in Eq. (11) with respect to P equal to zero, we have

$$\begin{aligned}(\rho\mathbf{I} + 2\lambda_3\mathbf{X}^T\mathbf{X})\mathbf{P} = \rho(\mathbf{C}) \\ + 2\lambda_3\mathbf{X}^T(\mathbf{X} - \mathbf{E}_1 - \mathbf{E}_2) - \mathbf{M}.\end{aligned} \tag{12}$$

Eq. (12) is a $N \times N$ system of the linear equations, which drives the solution of $\mathbf{P}^*$.

- ($\mathbf{C}-$**Update**): Updating $\mathbf{C}$ relies on the following problem by fixing the other variables and removing irrelevant terms:

$$\min_{\mathbf{C}} \mu\|\mathbf{C}\|_1+\frac{\rho}{2}\|\mathbf{P}-(\mathbf{C}-\text{diag}(\mathbf{C}))\|_F^2 +tr(\mathbf{M}^T(\mathbf{P}-(\mathbf{C}-\text{diag}(\mathbf{C})))) \quad (13)$$

This optimization problem of C-update has the closed-form solution

$$\mathbf{C}^* = Q - diag(Q). \quad (14)$$

where $\mathbf{Q} = \tau_{\mu\rho^{-1}}(\mathbf{P}+M/\rho)$ and $\tau_\beta(\cdot)$ is the shrinkage-thresholding operator[25].

- ($\mathbf{E_1}-$**Update**): Updating $\mathbf{E_1}$ relies on the following problem by fixing the other variables and removing irrelevant terms:

$$\min_{\mathbf{E}_1} \lambda_3\|\mathbf{X}-\mathbf{XP}-\mathbf{E}_1-\mathbf{E}_2\|_F^2+\lambda_1\|\mathbf{E}_1\|_1. \quad (15)$$

This optimization problem of E_1-update has the closed-form solution:

$$\mathbf{E}_1^* = \tau_{\lambda_1(2\lambda_3)^{-1}}(\mathbf{X}-\mathbf{XP}-\mathbf{E}_2), \quad (16)$$

where $\tau_\beta(\cdot)$ is the shrinkage-thresholding operator [25].

- ($\mathbf{E_2}-$**Update**): Updating $\mathbf{E_2}$ relies on the following problem by fixing the other variables and removing irrelevant terms:

$$\min_{\mathbf{E}_2} \lambda_3\|\mathbf{X}-\mathbf{XP}-\mathbf{E}_1-\mathbf{E}_2\|_F^2+\lambda_2\|\mathbf{E}_2\|_{21}. \quad (17)$$

This optimization problem of E_2-update has the closed-form solution:

$$\mathbf{E}_2^* = \Omega_{\lambda_2(2\lambda_3)^{-1}}(\mathbf{X}-\mathbf{XP}-\mathbf{E}_1), \quad (18)$$

where $\Omega_\beta(\cdot)$ is the $\ell_2 1$ minimization operator [15].

- **On Optimizing Weights $\boldsymbol{\lambda}$**

Fixing the other variables, the optimization problem in Eq. (5) is transformed into

$$\min_{\boldsymbol{\lambda}} \lambda_1\|\mathbf{E}_1\|_1+\lambda_2\|\mathbf{E}_2\|_{21}+\lambda_3\|\mathbf{E}_3\|_F^2+\gamma\sum_{i=1}^{3}\lambda_i\ln\lambda_i$$
$$\text{s.t.}\ \sum_{i=1}^{C}\lambda_i=1,\ \boldsymbol{\lambda}\geq\mathbf{0}, \quad (19)$$

If we define $metric(1) = \|\mathbf{E}_1\|_1$, $metric(2) = \|\mathbf{E}_2\|_{21}$ and $metric(3) = \|\mathbf{E}_3\|_F^2$, then Eq. (19) is rewritten as

$$\min_{\boldsymbol{\lambda}} \sum_{i=1}^{3}\lambda_i(metric(i))+\sum_{i=1}^{3}\lambda_i\ln\lambda_i$$
$$\text{s.t.}\ \sum_{i=1}^{C}\lambda_i=1,\lambda\geq\mathbf{0}. \quad (20)$$

Let the Lagrangian function of Eq. (20) be

$$Q(\boldsymbol{\lambda},\varphi)=\sum_{i=1}^{3}\lambda_i(metric(i))+\gamma\sum_{i=1}^{3}\lambda_i\ln\lambda_i+\varphi(\sum_{i=1}^{3}\lambda_i-1) \quad (21)$$

By setting the partial derivative of Eq. (20) with respect to λ_i equal to zeros: $\frac{\partial Q}{\partial\lambda_i}=\lambda_i(metric(i))+\gamma(\ln\lambda_i+1)-\varphi=0$, we have

$$\lambda_i=\exp(-metric(i)/\gamma)\cdot\exp((\gamma-\varphi)/\gamma) \quad (22)$$

Considering $\sum_{i=1}^{3}\lambda_i=1$, the closed-form solution is derived as

$$\lambda_i=\frac{\exp(-(metric(i)/\gamma)}{\sum_{i=1}^{3}\exp(-(metric(i)/\gamma)},\quad i=1,2,3. \quad (23)$$

When updating the variables $(\mathbf{P},\mathbf{C},\mathbf{E_1},\mathbf{E_2},\boldsymbol{\lambda})$, ADM performs a gradient ascent update with the step size of ρ on the Lagrange multiplier $\mathbf{M}$ until convergence is achieved or the number of iterations exceeds a maximum iteration number to obtain the optimal solution.

B. CONVERGENCE PROPERTIES

Because the Lagrange function contains more than two block variables $(\mathbf{P},\mathbf{C},\mathbf{E_1},\mathbf{E_2},\boldsymbol{\lambda})$, it is difficult to give a strict mathematical proof for its convergence. However, recent studies [15] show that when the optimal gap of each iteration monotonically decreases, the ADM approach has good convergence properties. The convexity of the objective function with respect to each of the block variables guarantees, that in practice, the ADM approach has good convergence properties.

C. COMPUTATIONAL COMPLEXITY

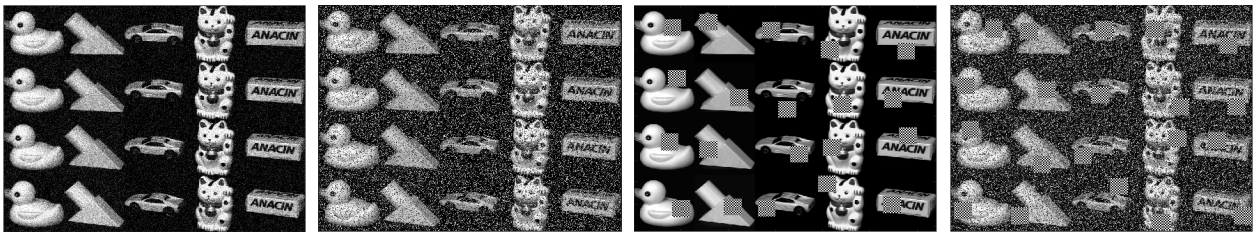
The computational cost of the proposed is mainly determined by the derivation of coefficients matrix (See Eq. (5)), where the ADM approach is used to obtain the coefficients matrix. From the procedure of ADM approach, the computation cost of ADM approach is mainly determined by the $\ell_{2,1}$ minimization solver. For convenience, let k denotes the number of iterations. Then, the computational complexity of the $\ell_{2,1}$ minimization solver is $O(kM^2N)$, where M is the dimensionality of the data sample and N is the number of the samples. Thus, the computational complexity of the proposed method is $O(kM^2N)$.

V. EXPERIMENTS

A. EXPERIMENTAL SETTING

In our experiments, we compared the proposed method (Ada-ℓ_1-Gr) to the following methods: KNN-Gr, ℓ_1-Gr [6], ℓ_2-Gr [12], ℓ_2-cGr [13], MLRR-Gr [20] and LRR-Gr [15].

For ℓ_1-Gr, we adopt its variant, where the combination of ℓ_1-norm based metric and ℓ_1-norm based metric is used to measure the reconstruction error and the sum of the



(a) (b) (c) (d)

FIGURE 1. The examples of corrupted images. (a) Gaussian noise. (b) Random pixel corruption. (c) Contiguous occlusion corruption. (d) Mixed noise. Compared to the single noise, the mixed noise is more difficult to handle.

regularization parameter is set to 1. The sparse constraint is imposed on the coefficient matrix.

For cLRR-Gr, it has two variants: cLRR-Gr1 (using the correntropy induced metric) and cLRR-Gr2 (using the columnwise based correntropy). Both methods impose the low-rank constraint on the coefficient matrix.

For ℓ_2-cGr, it adopts the correntropy induced metric and imposes the ℓ_2-norm constraint on the coefficient matrix.

The optimal parameter values of all the compared methods are chosen by cross-validation tests. Then, we conduct the experiments on the following: the Extended Yale B face dataset [27], the Coil20 object dataset [28], [35]–[37] and the USPS digit dataset [29]. For the experiments on the datasets, all the data points from each dataset were scaled to be unitary in the Euclidean norm.

B. IMAGE CLUSTERING EXPERIMENTS

Graph-based clustering is a spectral clustering-based model [26], where the K-means clustering algorithm is employed on the rows of the eigenvector matrix to obtain the final clustering results. The clustering number is set to be the number of the classes on the dataset.

We conducted clustering experiments on clean data and corrupted data, where complex noise is added to each image in the dataset by mixing three types of noise and data corruptions: (1) *Gaussian noise*. Gaussian noise is added to each image, x; that is, $\widetilde{\mathbf{x}} = \mathbf{x} + \alpha n$, where α is the corruption ratio ranging from 0% to 60% with an interval of 15%, and n is the noise following a standard normal distribution. (2) *Random pixel corruption*. To simulate sparse noise, we randomly choose a set of pixels from each image and set their pixel values to follow a uniform distribution over [0, pmax], where pmax is the largest pixel value of the current image. (3) *Contiguous occlusion corruption*. To simulate the contiguous occlusions, we replaced α percent of randomly chosen pixels from each selected image with the black-white squares.

For each dataset, we randomly selected eighty percent of the images and added the same corruption ratio of the above three types of noises to each image, resulting in three corrupted datasets. Figure 1 shows the examples of the corrupted images.

TABLE 1. Clustering on Yale B dataset.

Mixed Noise	AC(%)				NMI(%)			
Corrupted ratio	0%	30%	50%	80%	0%	30%	50%	80%
KNN-Gr	33.6	20.2	17.5	15.3	39.5	37.9	28.3	19.2
ℓ_1-Gr[6]	39.4	37.2	34.6	31.4	42.2	41.4	39.9	38.2
ℓ_2-Gr[12]	37.6	35.2	33.9	30.5	40.8	39.9	38.5	36.3
LRR-Gr[15]	38.9	38.4	37.3	35.4	42.3	41.6	40.3	38.4
MLRR-Gr[20]	39.2	38.9	37.8	35.8	43.6	42.8	40.8	38.9
Ada-ℓ_1-Gr	**41.3**	**40.2**	**39.4**	**38.1**	**48.2**	**46.7**	**45.1**	**43.7**

TABLE 2. Clustering on Coil 20 dataset.

Mixed Noise	AC(%)				NMI(%)			
Corrupted ratio	0%	30%	50%	80%	0%	30%	50%	80%
KNN-Gr	79.6	74.2	66.5	55.4	81.1	82.2	74.1	70.8
ℓ_1-Gr[6]	81.6	79.3	76.4	74.1	83.5	82.3	81.9	79.3
ℓ_2-Gr[12]	80.3	77.6	74.8	71.9	80.8	80.6	78.9	75.6
LRR-Gr[15]	80.7	78.7	77.6	75.5	81.2	84.3	83.9	82.4
MLRR-Gr[20]	82.1	81.2	79.3	76.4	84.2	82.8	80.3	78.7
Ada-ℓ_1-Gr	**84.7**	**82.9**	**81.3**	**79.6**	**87.7**	**86.1**	**85.4**	**83.3**

TABLE 3. Clusering on USPS dataset.

Mixed Noise	AC(%)				NMI(%)			
Corrupted ratio	0%	30%	50%	80%	0%	30%	50%	80%
KNN-Gr	83.6	79.2	70.5	68.5	87.2	86.7	81.5	74.1
ℓ_1-Gr[6]	82.7	81.5	79.5	78.6	85.2	84.3	83.3	81.2
ℓ_2-Gr[12]	83.5	82.2	80.4	79.3	87.6	85.6	84.6	83.3
LRR-Gr[15]	83.6	81.8	80.5	79.6	87.9	85.8	83.9	81.6
MLRR-Gr[20]	84.2	82.9	81.1	80.4	89.2	88.4	86.3	84.4
Ada-ℓ_1-Gr	**85.3**	**84.2**	**82.6**	**81.3**	**90.6**	**87.5**	**86.3**	**85.4**

In our clustering experiments, two popular benchmarks -Accuracy of Clustering (AC) and Normalized Mutual Information (NMI) [11] - are used to measure clustering performance. We independently repeated clustering experiments five times and reported the average clustering results for comparison. Tables 1-3 show the clustering results.

From Tables 1-3, we observe the following:

Our method consistently outperforms the other methods. The results are attributed that the fact our method effectively estimates the mixed noise components, further resulting in the noise-resistant edges. Specifically, when the level of corruption is large, the proposed method *significantly* outperforms the neighborhood-based methods.

C. IMAGE CLASSIFICATION EXPERIMENTS

The graph-based classification model adopts the combination of classification loss function and the graph regularizer for classification task. The commonly used graph-based classification framework [21] is defined as follows:

$$\mathbf{F}^* = \arg\min_{\mathbf{F}} \left\{ \sum_{i=1}^{c} \mathbf{F}_i^T \Phi \mathbf{F}_i + \nu \|\mathbf{F}-\mathbf{Y}\|^2 \right\}. \quad (24)$$

where Φ is graph Laplacian, c is the number of data classes; $\mathbf{F}$ signifies the classification relevant scores; $\mathbf{Y}$ is the initial label matrix, in which $\mathbf{Y}_{ij} = 1$ if the i-th data sample is labeled as the j-th class; otherwise, it is $\mathbf{Y}_{ij} = 0$ and ν is the tradeoff parameter that simply set to 1. When obtaining the closed-form solution of $\mathbf{F}$, the label of the i-th sample is recognized as $j^* = \arg\max_j \mathbf{F}_{ij}$, $(j = 1, 2, \cdots c)$.

We added the mixed noise to each image in the datasets in the same way as in the clustering experiments. Classification performance is measured by the Classification Accuracy for Classification (ACC), defined as the ratio between the number of correct classification and the size of test dataset. We selected twenty percent of the labeled samples from each subject to form a training set to conduct the classification experiments on the resultant corrupted datasets. Tab. 4-6 list the classification results.

TABLE 4. Classification on Yale B dataset.

Mixed Noise	ACC(%)				
Corrupted ratio	0%	15%	30%	50%	80%
KNN-Gr	54.3	52.2	46.3	43.6	40.4
ℓ_1-Gr[6]	58.4	55.3	54.6	51.5	48.7
ℓ_2-Gr[12]	57.3	54.2	53.4	50.3	47.5
LRR-Gr[15]	59.3	58.4	56.3	55.3	54.5
MLRR-Gr[20]	61.2	60.6	58.3	56.4	54.2
Ada-ℓ_1-Gr	**64.3**	**63.2**	**61.6**	**60.3**	**58.5**

TABLE 5. Classification on Coil 20 dataset.

Mixed Noise	ACC(%)				
Corrupted ratio	0%	15%	30%	50%	80%
KNN-Gr	93.3	89.2	85.5	75.5	60.3
ℓ_1-Gr[6]	91.2	89.3	87.6	84.4	80.9
ℓ_2-Gr[12]	89.6	87.0	85.7	82.6	78.3
LRR-Gr[15]	90.9	89.3	87.3	85.1	82.2
MLRR-Gr[20]	92.2	90.9	89.3	87.4	84.2
Ada-ℓ_1-Gr	**94.8**	**93.2**	**91.7**	**89.3**	**87.4**

TABLE 6. Classification on USPS dataset.

Mixed Noise	ACC(%)				
Corrupted ratio	0%	15%	30%	50%	80%
KNN-Gr	91.6	89.2	80.5	73.5	66.5
ℓ_1-Gr[6]	91.4	90.3	88.6	86.4	84.5
ℓ_2-Gr[12]	92.3	91.0	87.4	85.6	83.8
LRR-Gr[15]	93.3	92.5	91.6	88.4	86.6
MLRR-Gr[20]	94.9	93.9	92.3	91.4	90.2
Ada-ℓ_1-Gr	**95.3**	**94.8**	**94.0**	**93.3**	**92.6**

From Tables 4-6, we observe the following:

The proposed method achieves the highest classification accuracy. With an increasing in data corruption ratios, the proposed method outperforms corresponding neighborhood-based methods by a larger performance margin.

D. EFFECTIVENESS OF ERROR ESTIMATION

To know how error estimation affects the performance, we compute the reconstruction errors for each method

$$DataError = \frac{error(\mathbf{E})}{\|\mathbf{E}\|_F^2} \quad (25)$$

where $error(\mathbf{E})$ is the error estimation term. e.g. for the ℓ_1-Graph, the error estimation term is $\|\mathbf{E}\|_F^2$ and for the proposed method, the error estimation term is $\lambda_1\|\mathbf{E}_1\|_1 + \lambda_2\|\mathbf{E}_2\|_{21} + \lambda_3\|\mathbf{E}_3\|_F^2$. We calculated the relative values of the error estimation term and list these results in Table 7.

TABLE 7. The reconstruction errors of different methods.

Method	YaleB	Coil20	USPS
ℓ_1-Gr[6]	0.842	0.967	0.958
ℓ_2-Gr[12]	1.000	1.000	1.000
LRR-Gr[15]	0.803	0.902	0.896
MLRR-Gr[20]	0.764	0.887	0.856
Ada-ℓ_1-Gr	**0.574**	**0.758**	**0.779**

From Table 7, we observe that the proposed method has the smallest reconstruction error. The experimental results show that the proposed method effectively estimates the data errors, which partially explains why our method is superior to the compared methods.

E. PARAMETER SETTINGS

The proposed method has two essential parameters: μ and γ, where μ is the sparse regularization parameter, and γ is the Shannon entropy regularization parameter. To observe how the performance depends on the parameters, we conducted the parameter tuning experiments on a Coil 20 dataset by varying one parameter values while fixing the others. Figures 2-3 show the experimental results.

From Figures 2-3, we observe the following:

As the parameter value of μ increases, the performance of the proposed method increases accordingly. Until the parameter value is greater than 10, the performance is degraded a little. These experiments results indicate that sparse regularization is crucial for representing the data correlations. When the parameter value is sufficiently small, the derived coefficients are not discriminate enough to generate the informative edges.

As the parameter value of γ increases, the proposed method performs better. When the parameter value is greater than 1, the performance slightly degenerates. The experimental results show that when parameter is very small, the error estimation may over-fit few candidate metrics, indicating that only very few candidate metrics play important roles in estimating data errors; when the parameter value is very large, the error estimation may fit all the metrics, indicating that all candidate metrics play important roles.

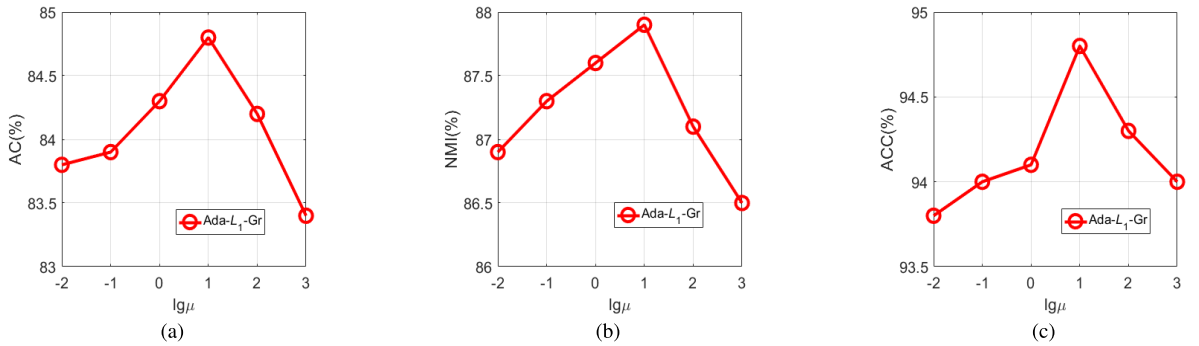


FIGURE 2. The experimental results versus tradeoff parameter μ: (a) Accuracy of clustering versus μ. (b) Normalized mutual information of clustering versus μ. (c) Accuracy of classification versus μ.

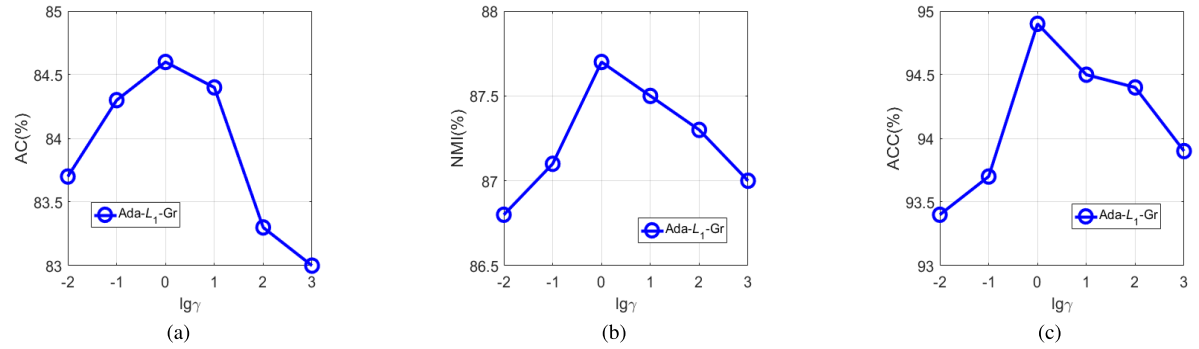


FIGURE 3. The experimental results versus tradeoff parameter γ: (a) Accuracy of clustering versus γ. (b) Normalized mutual information of clustering versus γ. (c) Accuracy of classification versus γ.

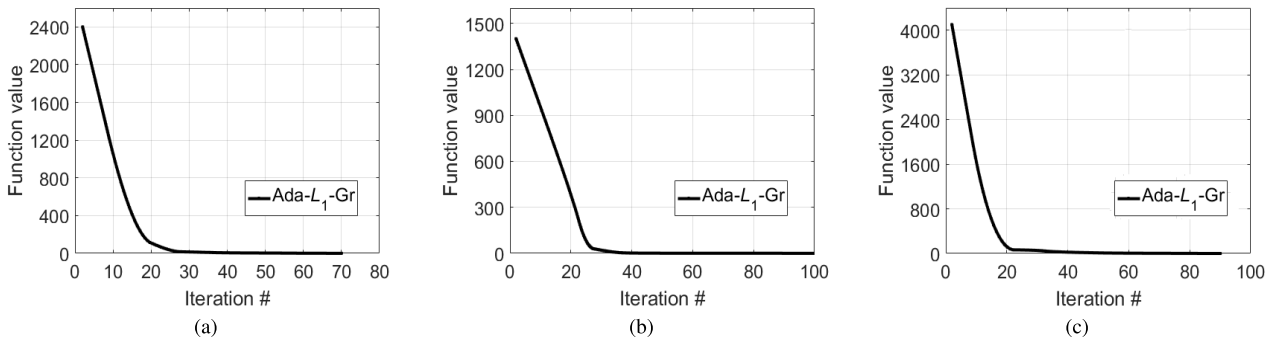


FIGURE 4. Convergence curves of the proposed method: (a) Yale B dataset, (b) Coil 20 dataset and (c) USPS dataset.

F. CONVERGENCE OF THE ADM-BASED OPTIMIZATION

We adopt the ADM approach to optimize the objective functions of the proposed method. Therefore, the convergence properties of the ADM optimization approach is crucial. In this subsection, we present the convergence curves of the proposed method. The changes of the objective function value are reported in Figure 4.

As shown in Figure 4, the objective function value decreases rapidly at the outset, and then, decrease slowly. After fewer than 40 iterations, the objective function values are stable.

VI. CONCLUSION

In this article, we have proposed a novel multi-metric induced robust graph construction method. Different from the existing methods, the proposed method effectively estimates complex data errors beyond Gaussian noise or sparse noise. Therefore, a robust graph are constructed, resulting in the

promising clustering or classification results on real-world image datasets. Note that the proposed method is designed to handle the single-view data. When the real-world data contains multiple view features, it requires to concatenate the different view features for employing the proposed method. For future work, we plan to extend the proposed method to handle the multi-view data.

REFERENCES

[1] S. Huang, Z. Kang, I. W. Tsang, and Z. Xu, "Auto-weighted multi-view clustering via kernelized graph learning," *Pattern Recognit.*, vol. 88, pp. 174–184, Apr. 2019.
[2] T. Liu, C. K. L. Lekamalage, G.-B. Huang, and Z. Lin, "An adaptive graph learning method based on dual data representations for clustering," *Pattern Recognit.*, vol. 77, pp. 126–139, May 2018.
[3] W. Wang, Y. Yan, F. Nie, S. Yan, and N. Sebe, "Flexible manifold learning with optimal graph for image and video representation," *IEEE Trans. Image Process.*, vol. 27, no. 6, pp. 2664–2675, Jun. 2018.
[4] Z. Li, F. Nie, X. Chang, Y. Yang, C. Zhang, and N. Sebe, "Dynamic affinity graph construction for spectral clustering using multiple features," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 12, pp. 6323–6332, Dec. 2018.
[5] J. Wright, Y. Ma, J. Mairal, G. Sapiro, T. S. Huang, and S. Yan, "Sparse representation for computer vision and pattern recognition," *Proc. IEEE*, vol. 98, no. 6, pp. 1031–1044, Jun. 2010.
[6] 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.
[7] X. Sun, Q. Qu, N. M. Nasrabadi, and T. D. Tran, "Structured priors for sparse-representation-based hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 11, no. 7, pp. 1235–1239, Jul. 2014.
[8] Y. Wang, Y. Y. Tang, L. Li, and X. Zheng, "Block sparse representation for pattern classification: Theory, extensions and applications," *Pattern Recognit.*, vol. 88, pp. 198–209, Apr. 2019.
[9] S. Gao, I. W. Tsang, and L.-T. Chia, "Sparse representation with kernels," *IEEE Trans. Image Process.*, vol. 22, no. 2, pp. 423–434, Feb. 2013.
[10] M. Zheng, J. Bu, C. Chen, C. Wang, L. Zhang, G. Qiu, and D. Cai, "Graph regularized sparse coding for image representation," *IEEE Trans. Image Process.*, vol. 20, no. 5, pp. 1327–1336, May 2011.
[11] T. Jin, Z. Yu, L. Li, and C. Li, "Multiple graph regularized sparse coding and multiple hypergraph regularized sparse coding for image representation," *Neurocomputing*, vol. 154, pp. 245–256, Apr. 2015.
[12] X. Peng, Z. Yu, Z. Yi, and H. Tang, "Constructing the L2-graph for robust subspace learning and subspace clustering," *IEEE Trans. Cybern.*, vol. 47, no. 4, pp. 1053–1066, Apr. 2017.
[13] C. Lu, J. Tang, M. Lin, L. Lin, S. Yan, and Z. Lin, "Correntropy induced ℓ_2-graph for robust subspace clustering," in *Proc. CVPR*, 2013, pp. 1801–1808.
[14] T. Jin, Z. Liu, Z. Yu, X. Min, and L. Li, "Locality preserving collaborative representation for face recognition," *Neural Process. Lett.*, vol. 45, no. 3, pp. 967–979, Jun. 2017.
[15] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, and Y. Ma, "Robust recovery of subspace structures by low-rank representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 171–184, Jan. 2013.
[16] A. Li, Y. Ding, D. Chen, G. Sun, H. Jiang, and Q. Wu, "Cross-view feature learning via structures unlocking based on robust low-rank constraint," *IEEE Access*, vol. 8, pp. 46851–46860, 2020.
[17] G. Liu and S. Yan, "Latent low-rank representation for subspace segmentation and feature extraction," in *Proc. CVPR*, 2011, pp. 1615–1622.
[18] M. Yin, S. Xie, Z. Wu, Y. Zhang, and J. Gao, "Subspace clustering via learning an adaptive low-rank graph," *IEEE Trans. Image Process.*, vol. 27, no. 8, pp. 3716–3728, Aug. 2018.
[19] S. Yang, Z. Feng, Y. Ren, H. Liu, and L. Jiao, "Semi-supervised classification via kernel low-rank representation graph," *Knowl. Based Syst.*, vol. 69, pp. 150–158, Oct. 2014.
[20] M. Yin, J. Gao, and Z. Lin, "Laplacian regularized low-rank representation and its applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 3, pp. 504–517, Mar. 2016.
[21] W. Zhu and B. Peng, "Sparse and low-rank regularized deep subspace clustering," *Knowl.-Based Syst.*, vol. 204, Sep. 2020, Art. no. 106199.
[22] Y. Zheng, X. Zhang, S. Yang, and L. Jiao, "Low-rank representation with local constraint for graph construction," *Neurocomputing*, vol. 122, pp. 398–405, Dec. 2013.
[23] C. Zhang, H. Fu, S. Liu, G. Liu, and X. Cao, "Low-rank tensor constrained multiview subspace clustering," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1582–1590.
[24] S. W. Fung and L. Ruthotto, "An uncertainty-weighted asynchronous ADMM method for parallel PDE parameter estimation," *SIAM J. Sci. Comput.*, vol. 41, no. 5, pp. S129–S148, Jan. 2019.
[25] Z. Lin, M. Chen, L. Wu, and Y. Ma, "The augmented Lagrange multiplier method for exact recovery of corrupted low-rank matrices," Tech. Rep. UILU-ENG-09-2215, 2009.
[26] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 888–905, Aug. 2000.
[27] K.-C. Lee, J. Ho, and D. J. Kriegman, "Acquiring linear subspaces for face recognition under variable lighting," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 5, pp. 684–698, May 2005.
[28] S. A. Nene, S. K. Nayar, and H. Murase, "Columbia object image library (COIL-20)," Tech. Rep. CUCS-005-96, Feb. 1996.
[29] J. J. Hull, "A database for handwritten text recognition research," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 16, no. 5, pp. 550–554, May 1994.
[30] C. Tang, M. Bian, X. Liu, M. Li, H. Zhou, P. Wang, and H. Yin, "Unsupervised feature selection via latent representation learning and manifold regularization," *Neural Netw.*, vol. 117, pp. 163–178, Sep. 2019.
[31] C. Tang, X. Liu, X. Zhu, J. Xiong, M. Li, J. Xia, X. Wang, and L. Wang, "Feature selective projection with low-rank embedding and dual Laplacian regularization," *IEEE Trans. Knowl. Data Eng.*, vol. 32, no. 9, pp. 1747–1760, Sep. 2020.
[32] C. Tang, X. Liu, P. Wang, C. Zhang, M. Li, and L. Wang, "Adaptive hypergraph embedded semi-supervised multi-label image annotation," *IEEE Trans. Multimedia*, vol. 21, no. 11, pp. 2837–2849, Nov. 2019.
[33] C. Tang, X. Zhu, X. Liu, M. Li, P. Wang, C. Zhang, and L. Wang, "Learning a joint affinity graph for multiview subspace clustering," *IEEE Trans. Multimedia*, vol. 21, no. 7, pp. 1724–1736, Jul. 2019.
[34] C. Tang, X. Liu, X. Zhu, E. Zhu, Z. Luo, L. Wang, and W. Gao, "CGD: Multi-view clustering via cross-view graph diffusion," in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 5924–5931.
[35] C. L. Chowdhary, K. Muatjitjeja, and D. S. Jat, "Three-dimensional object recognition based intelligence system for identification," in *Proc. Int. Conf. Emerg. Trends Netw. Comput. Commun. (ETNCC)*, May 2015, pp. 162–166.
[36] D. P. Acharjya and V. Santhi, *Bio-Inspired Computing for Image and Video Processing*, 1st ed. Boca Raton, FL, USA: Chapman & Hall, Dec. 2017, doi: 10.1201/9781315153797.
[37] G. Ye, Z. Tang, H. Wang, D. Fang, J. Fang, S. Huang, and Z. Wang, "Deep program structure modeling through multi-relational graph-based learning," in *Proc. ACM Int. Conf. Parallel Architectures Compilation Techn.*, Sep. 2020, pp. 111–123.

ZENGMIN GENG is currently a Professor with the Beijing Institute of Fashion Technology, China. His main research interests include computer vision and pattern recognition.

XIAODONG SUN is currently an Associate Professor with the Beijing Institute of Fashion Technology, China. Her main research interests include computer vision and pattern recognition.

JUJIAN ZHANG is currently a Professor with the Beijing Institute of Fashion Technology, China. His main research interests include computer vision and pattern recognition.

JIANXIA DU is currently an Associate Professor with the Beijing Institute of Fashion Technology, China. Her main research interests include computer vision and pattern recognition.

YILING ZHOU is currently an Associate Professor with the Beijing Institute of Fashion Technology, China. Her main research interests include computer vision and pattern recognition.