Semi-supervised Affinity Matrix Learning via Dual-channel Information Recovery

Yuheng Jia, Hui Liu, Junhui Hou, *Senior Member, IEEE,* Sam Kwong, *Fellow, IEEE*, and Qingfu Zhang, *Fellow, IEEE*

***Abstract*—This paper explores the problem of semi-supervised affinity matrix learning, i.e., learning an affinity matrix of data samples under the supervision of a small number of pairwise constraints. By observing that both the matrix encoding pairwise constraints named pairwise constraint matrix (PCM) and the empirically constructed affinity matrix (EAM) express the similarity between samples, we assume that both of them are generated from a latent affinity matrix (LAM) that can depict the ideal pairwise relation between samples. Specifically, the PCM can be thought of as a partial observation of the LAM, while the EAM is a fully observed one but corrupted with noise/outliers. To this end, we innovatively cast the semi-supervised affinity matrix learning as the recovery of the LAM guided by the PCM and EAM, which is technically formulated as a convex optimization problem. We also provide an efficient algorithm for solving the resulting model numerically. Extensive experiments on benchmark data sets demonstrate the significant superiority of our method over state-of-the-art ones when used for constrained clustering and dimensionality reduction. The code is publicly available at https://github.com/jyh-learning/LAM.**

***Index Terms*—Semi-supervised learning, clustering, graph learning.**

I. Introduction

Clustering is a fundamental machine learning task with many applications [1], [2], such as image segmentation [3], anomaly detection [4], and community detection [5]. As a popular and well-known clustering method, spectral clustering (SC) has gained considerable attention due to the good performance and wide application domains. Specifically, SC only needs an affinity matrix that records the similarity relation between samples as input, and can output a spectral embedding for clustering. However, there exist many error connections in the empirically constructed affinity matrix due to the existence of noise in real world data, which degrades the performance of SC severely [6]. To solve this drawback, many constrained spectral clustering (CSC) methods were proposed [7], [8] by integrating some supervisory information in the form of pairwise constraints (PCs) into SC. PC is a natural manner to describe the available supervisory information, which indicates whether two samples belong to the same cluster or not.

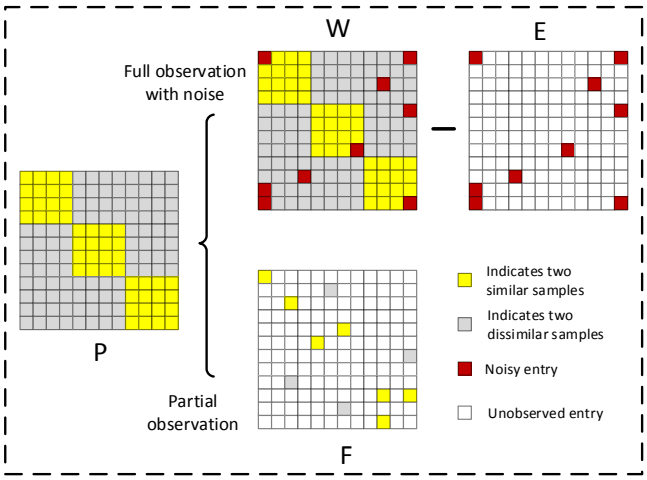

Fig. 1. Assumptions of the relation among the ideal latent affinity matrix $\mathbf{P}$, the empirical affinity matrix $\mathbf{W}$ and the pairwise constraint matrix $\mathbf{F}$. Specifically, $\mathbf{W}$ is an observation of $\mathbf{P}$ with some corruption $\mathbf{E}$, and $\mathbf{F}$ is a partial observation of $\mathbf{P}$. Based on these assumptions, the semi-supervised affinity matrix learning becomes the recovery of $\mathbf{P}$ with the guidance of $\mathbf{W}$ and $\mathbf{F}$.

Based on the manners of encoding the available PCs, the existing CSC methods can be roughly divided into two categories. The first kind of methods uses the PCs to restrict the feasible solution space of SC, i.e., finding a spectral embedding that is consistent with most of the available PCs [9], [10], [11]. For example, Jia *et al.* [12] first built a structured matrix according to the PCs and then used it to regularize the process of SC. Zhang *et al.* [13] formulated CSC as a symmetric non-negative matrix factorization problem and used the inner product of the factorized matrices to approximate the PCs. Methods in the second category aim to generate a more informative affinity matrix according to the PCs [14], [15], [16], which are more promising, as the affinity matrix is critical for SC. For example, [8], [17] used a graph structure to propagate the initial PCs to the whole data set and then used the propagated PC matrix to refine the original affinity matrix. Yang *et al.* [18] realized PC propagation with a global low-rank structure. Kulis *et al.* [19] constructed the semi-supervised affinity matrix by a kernel learning approach. Wu *et*

This work was supported by the Key Project of Science and Technology Innovation 2030 supported by the Ministry of Science and Technology of China under Grant 2018AAA0101301, in part by the Natural Science Foundation of China under Grants 61871342, 61772344, and 61672443, and in part by the Hong Kong RGC General Research Funds under Grants 9042820 (CityU 11219019), 9042489 (CityU 11206317), 9042322 (CityU 11200116), 9042816 (CityU 11209819), and 9048123 (CityU 21211518). (Corresponding author: Junhui Hou and Sam Kwong.)

Y. Jia is with the School of Computer Science and Engineering, Southeast University, Nanjing, 211189, China (e-mail: yhjia@seu.edu.cn).

H. Liu is with the Department of Computer Science, City University of Hong Kong, Kowloon, Hong Kong, (e-mail: hliu99-c@my.cityu.edu.hk).

J. Hou, S. Kwong and Q. Zhang are with the Department of Computer Science, City University of Hong Kong, Kowloon, Hong Kong and also with the City University of Hong Kong Shenzhen Research Institute, Shenzhen, 51800, China, (e-mail: jh.hou@cityu.edu.hk; cssamk@cityu.edu.hk; qingfu.zhang@cityu.edu.hk).

al. [20] performed PC propagation and CSC jointly to achieve mutual enhancement.

In this paper, we attempt to learn a more discriminative affinity matrix. We observe that the matrix constructed by encoding PCs, namely pairwise constraint matrix (PCM), can specify the relationship between two samples exactly; however only limited PCs are available in practice. On the contrary, an empirically constructed affinity matrix (EAM) is able to estimate the similarity between any two of the input data samples, but some of the estimations are incorrect due to the existence of noise in the real world data [1]. Based on these observations, we assume there exists an ideal latent affinity matrix (LAM) that is able to depict the pairwise relation between samples correctly, and both EAM and PCM are the observations of the LAM via different manners. The ideal LAM is a binary block-diagonal matrix, which can be thought of as a kind of structural sparse matrices. The structural sparse-based methods have enormous applications in machine learning, e.g., feature selection [21], multitask learning [22], and sparse Bayesian learning [23], to name a few. As shown in Fig. 1, the EAM is corrupted by severe noise/outliers, while the PCM is just a partial observation of the LAM. *Therefore, the semi-supervised affinity learning becomes an LAM recovery problem, i.e., recover the ideal LAM given EAM and PCM.* To this end, we formulate it as a convex optimization model, which is then solved efficiently by the inexact augmented Lagrange multiplier algorithm. Experimental results on 8 data sets show that our model significantly outperforms the state-of-the-art methods when used for constrained clustering and dimensionality reduction.

The remainder of this paper is organized as follows. We first introduce the background knowledge in Section II, and then present our semi-supervised latent affinity matrix learning model as well as the associated optimization algorithm in Section III, followed the experimental results and analyses in Section IV. Finally Section V concludes this paper.

II. Related Work

In this section, we first introduce the background knowledge of SC and CSC, and then discuss the applications of affinity matrix in machine learning. Table I summarizes the abbreviations used in this paper.

A. Spectral Clustering

Given an input data matrix with n samples, $\mathbf{X} = [\mathbf{x}_1, \cdots, \mathbf{x}_n] \in \mathbb{R}^{d\times n}$, where each sample $\mathbf{x}_i \in \mathbb{R}^{d\times 1}$ is represented with a d-dimensional vector, SC aims to partition $\mathbf{X}$ into different clusters. Generally, SC-based methods consist of the following three steps:

1) Build an affinity matrix $\mathbf{W} \in \mathbb{R}^{n\times n}$ empirically to measure the pairwise similarities among data samples.
2) Perform spectral decomposition on $\mathbf{W}$ or normalized $\mathbf{W}$ to obtain a lower-dimensional representation.
3) Perform K-means or its variants on the resulting lower-dimensional representation to generate the final clustering result.

TABLE I
List of Abbreviations

Abbrev.	Full Form
PCM	pairwise constraint matrix
EAM	empirically constructed affinity matrix
LAM	latent affinity matrix
SC	spectral clustering
CSC	constrained spectral clustering
PC	pairwise constraint
SNMF	symmetric nonnegative matrix factorization
PCP	pairwise constraint propagation
E2CP	exhaustive and efficient constraint propagation
SSLRR	semi-supervised low-rank representation
GPCA	graph regularized principal component analysis
NMF	nonnegative matrix factorization
IALM	inexact augmented Lagrange multiplier
SVD	singular value decomposition
ACC	clustering accuracy
NMI	normalized mutual information
RBF	radial basis function

For the commonly used methods for constructing $\mathbf{W}$, we refer the readers to [24]. Assuming that the highest (resp. lowest) similarity between two samples is 1 (resp. 0), we have $0 \leq \mathbf{W}_{ij} \leq 1, \forall i, j$.

B. Constrained Spectral Clustering

All traditional SC methods are unsupervised, while CSC aims to improve the unsupervised SC with the help of some supervisory information in the form of PC. The PCs are composed of a must-links set $\mathcal{M}$ and a cannot-links set $\mathcal{C}$, which record whether two samples belong to the same cluster or not, respectively. In this paper, the PCM is expressed by a partially observed matrix $\mathbf{F} \in \mathbb{R}^{n\times n}$:

$$\mathbf{F}_{ij} = \begin{cases} 1, & (\mathbf{x}_i, \mathbf{x}_j) \in \mathcal{M} \\ 0, & (\mathbf{x}_i, \mathbf{x}_j) \in \mathcal{C} \\ \text{unknown}, & \text{otherwise.} \end{cases} \tag{1}$$

Since the PCs indicate the cluster membership between two samples with high belief, it is clear that the performance of the supervised SC will be improved by exploiting the information from $\mathbf{F}$.

Based on the way of utilizing $\mathbf{F}$, the existing CSC methods can be roughly classified into two categories. Methods in the first category directly learn a better lower-dimensional embedding according to the given PCs. For example, Li *et al.* [9] tried to learn an ideal embedding by forcing the inner product of the lower-dimensional representation and its transpose to close to the PCM. Wang *et al.* [25] proposed a model that satisfies both the PCM and EAM in an optimization framework. Jiang *et al.* [7] solved a generalized eigen-decomposition problem by incorporating the PCs. Jia *et al.* [12] used the PCs to construct a structured sparsity regularization term to regularize the process of SC. Zhang *et al.* [13] incorporated the PCs in symmetric nonnegative matrix factorization (SNMF) to realize CSC.

Since the affinity matrix plays a critical role in SC, methods in the second category attempt to construct a more informative affinity matrix according to the initial affinity matrix and the given PCs. For example, Kamvar *et al.*[14] proposed to directly use PCs to refine the EAM, which can only produce limited improvement, since the number of the given PCs is usually limited. Therefore, many PC propagation (PCP) methods were proposed. One kind of the methods first propagates the initial PCs to the whole data set, and then refines the EAM according to the propagated PCs [8]. Another kind of methods directly constructs a semi-supervised affinity graph according to the given PCs [20]. Recently, some affinity learning methods based on the self-representation property of data samples have gained a lot of attention, like low-rank representation [26] and sparse subspace clustering [27]. Those self-representation based methods have also been extended to a semi-supervised manner [28], [29]. See [30] for more works on affinity matrix learning.

In what follows, we will review two related semi-supervised affinity matrices learning models in detail.

1) Exhaustive and Efficient Constraint Propagation (E2CP) [8]: E2CP [8] learns an affinity matrix by first propagating the initial available PCs encoded by the PCM $\mathbf{F}$ to the whole data set, which is formulated as

$$\min_{\widehat{\mathbf{F}}} \|\widehat{\mathbf{F}} - \mathbf{F}\|_F^2 + \lambda \mathrm{tr}\left(\widehat{\mathbf{F}}^\mathsf{T}\mathbf{L}\widehat{\mathbf{F}} + \widehat{\mathbf{F}}\mathbf{L}\widehat{\mathbf{F}}^\mathsf{T}\right), \tag{2}$$

where $\widehat{\mathbf{F}} \in \mathbb{R}^{n\times n}$ is the propagated PCs matrix, $\mathbf{L} \in \mathbb{R}^{n\times n}$ is the graph Laplacian matrix for propagation, and $\lambda > 0$ is a hyper-parameter. After solving Eq. (2), the new affinity matrix $\widehat{\mathbf{W}}$ can be obtained by refining the EAM $\mathbf{W}$ through

$$\widehat{\mathbf{W}} = \begin{cases} 1 - (1 - \widehat{\mathbf{F}}_{ij})(1 - \mathbf{W}_{ij}), & \text{if } \widehat{\mathbf{F}}_{ij} \geq 0 \\ (1 + \widehat{\mathbf{F}}_{ij})\mathbf{W}_{ij}, & \text{if } \widehat{\mathbf{F}}_{ij} < 0. \end{cases} \tag{3}$$

E2CP has been successfully applied to many applications, like clustering. However, E2CP formulates the PCP and affinity learning in a separate manner, which overlooks the interaction between them.

2) Semi-supervised Low-Rank Representation (SSLRR) [28]: SSLRR [28] is a self-representation-based affinity matrix learning method, which is formulated as

$$\begin{aligned} &\min_{\widehat{\mathbf{W}},\mathbf{E}} \|\widehat{\mathbf{W}}\|_* + \lambda\|\mathbf{E}\|_{2,1} \\ &\text{s.t.}\mathbf{X} = \mathbf{X}\widehat{\mathbf{W}} + \mathbf{E}, \widehat{\mathbf{W}}^\mathsf{T}\mathbf{1} = \mathbf{1}, \widehat{\mathbf{W}}_{ij} = 0, \forall(\mathbf{x}_i, \mathbf{x}_j) \in \mathcal{C}, \end{aligned} \tag{4}$$

where $\|\cdot\|_*$ and $\|\cdot\|_{2,1}$ denote the trace norm and $\ell_{2,1}$ pseudo-norm of a matrix, respectively, and $\widehat{\mathbf{W}}$ and $\mathbf{E}$ denote the learned affinity matrix and the reconstruction error matrix, respectively. The supervisory information is introduced to the affinity matrix $\widehat{\mathbf{W}}$ by enforcing the elements belonging to the cannot-links to be zero. The effectiveness of SSLRR has been validated in classification [28]. However, SSLRR only incorporates the cannot-link information, while ignoring the must-link information.

C. Application of the Affinity Matrix

Apart from SC and CSC, the affinity matrix also plays an important role in many other fields of machine learning by acting as a graph to depict pairwise relations among samples. For example, by exploiting the pairwise relation between the labeled data and the unlabeled data, many graph-based semi-supervised classification models [31], [32] were proposed. Since the affinity matrix is usually sparse, which only connects two samples with high similarity, it could be used to describe the manifold structure of a data set. Therefore, the affinity matrix has been applied to many dimensionality reduction and data representation models to exploit the intrinsic manifold structure of input, like graph regularized principal component analysis (GPCA) [33], graph regularized NMF [34], [35], graph regularized concept factorization [36], Laplacian embedding [37], and graph regularized robust PCA [38], etc. Graph is also a powerful tool to represent the structured data [39]. There are also many specific machine learning applications that demand an affinity matrix like superpixel segmentation [3], community detection [5], feature selection [40], hashing [41], [42], dictionary learning [43], etc.

III. The Proposed Method

As mentioned earlier, our objective is "*learning a more discriminative affinity matrix for boosting subsequent tasks, like spectral clustering, by making full use of the information of an EAM and a PCM*". Considering that both the PCM and the EAM describe the pairwise similarity between data samples, we assume that there exists an LAM to ideally describe the pairwise relation between samples, and both the PCM and EAM are the two observations of the LAM via different manners.

Implicitly, the proposed model is formulated as

$$\min_{\mathbf{P}} \mathcal{L}_1(\mathbf{P}, \mathbf{F}) + \mathcal{L}_2(\mathbf{P}, \mathbf{W}) \tag{5}$$

where $\mathbf{P} \in \mathbb{R}^{n\times n}$ denotes the LAM to recover, and $\mathcal{L}_1(\cdot,\cdot)$ and $\mathcal{L}_2(\cdot,\cdot)$ denote two kinds of loss functions. To correctly recover $\mathbf{P}$, we take different characteristics of the EAM and the PCM into consideration. Specifically, the PCM is a partially observed matrix, where the value of the observed element is either 1 or 0, indicating two associated samples must belong to the same cluster or different clusters, respectively. We thus restrict

$$\mathbf{P}_{ij} = \mathbf{F}_{ij}, \forall(\mathbf{x}_i, \mathbf{x}_j) \in \Omega, \tag{6}$$

where Ω is the union of $\mathcal{M}$ and $\mathcal{C}$. On the other hand, due to the existence of noise in the input data, the EAM is not as reliable as the PCM. We, therefore, decompose the EAM into the summation of the LAM and a sparse error matrix, i.e.,

$$\mathbf{W} = \mathbf{P} + \mathbf{E}, \tag{7}$$

where $\mathbf{E} \in \mathbb{R}^{n\times n}$ represents the sparse error term.

Since recovering $\mathbf{P}$ from Eq. (7) is an inverse problem, i.e., there are numerous solutions (for $\mathbf{P}$) that can exactly satisfy the two conditions in Eqs. (6) and (7), we use the ideal appearance of an LAM to regularize the solution space of $\mathbf{P}$. Specifically, for an ideal LAM, we have

$$\mathbf{P}_{ij} = \begin{cases} 1, \text{if } (\mathbf{x}_i, \mathbf{x}_j) \text{ belong to the same cluster} \\ 0, \text{if } (\mathbf{x}_i, \mathbf{x}_j) \text{ belong to different clusters,} \end{cases} \tag{8}$$

which indicates that the ideal LAM should be a low-rank matrix with the rank equal to the number of the ground-truth clusters. Moreover, it also should be symmetric, i.e., $\mathbf{P}_{ij} = \mathbf{P}_{ji}$, and bounded, i.e, $0 \leq \mathbf{P}_{ij} \leq 1, \forall i, j$. Taking the ideal appearance of an LAM into account, we preliminarily formulate the learning of the LAM as a low-rank matrix recovery problem, i.e.,

$$\begin{aligned} &\min_{\mathbf{P},\mathbf{E}} \operatorname{rank}(\mathbf{P}) + \lambda\|\mathbf{E}\|_0 \\ &\text{s.t. } 0 \leq \mathbf{P}_{ij} \leq 1, \forall i, j, \mathbf{P}^\mathsf{T} = \mathbf{P}, \mathbf{W} = \mathbf{P} + \mathbf{E}, \\ &\qquad \mathbf{P}_{ij} = \mathbf{F}_{ij}, \forall (\mathbf{x}_i, \mathbf{x}_j) \in \Omega, \end{aligned} \tag{9}$$

where $\lambda > 0$ is the hyper-parameter to control the effect of the sparse error matrix. Previous works have shown that under quite general conditions, we can exactly recover a low-rank matrix by solving a problem like Eq. (9) [44], [45]. Moreover, inspired by the success of the graph structure in disclosing the latent representation [34], [33], we also introduce a graph regularization to map the local geometry structure of the input data to $\mathbf{P}$, i.e., $\min_{\mathbf{P}} \operatorname{Tr}\left(\mathbf{P}\mathbf{L}\mathbf{P}^\mathsf{T}\right)$, where $\mathbf{L} \in \mathbb{R}^{n\times n}$ is the Laplacian matrix[1]. Therefore, the proposed model becomes

$$\begin{aligned} &\min_{\mathbf{P},\mathbf{E}} \operatorname{rank}(\mathbf{P}) + \lambda\|\mathbf{E}\|_0 + \gamma \operatorname{Tr}\left(\mathbf{P}\mathbf{L}\mathbf{P}^\mathsf{T}\right) \\ &\text{s.t. } 0 \leq \mathbf{P}_{ij} \leq 1, \forall i, j, \mathbf{P}^\mathsf{T} = \mathbf{P}, \mathbf{W} = \mathbf{P} + \mathbf{E}, \\ &\qquad \mathbf{P}_{ij} = \mathbf{F}_{ij}, \forall (\mathbf{x}_i, \mathbf{x}_j) \in \Omega, \end{aligned} \tag{10}$$

where $\gamma > 0$ is the hyper-parameter introducing the local geometry structure.

Due to the existence of the discrete and non-convex $\operatorname{rank}(\cdot)$ function and ℓ_0 matrix norm, the problem in Eq. (10) is highly non-convex and challenging to solve. To this end, we relax the two terms by using their convex envelopes, i.e., the nuclear norm $\|\cdot\|_*$ and the ℓ_1 norm, respectively. The proposed model is finally formulated as a convex problem, i.e.,

$$\begin{aligned} &\min_{\mathbf{P},\mathbf{E}} \|\mathbf{P}\|_* + \lambda\|\mathbf{E}\|_1 + \gamma \operatorname{Tr}\left(\mathbf{P}\mathbf{L}\mathbf{P}^\mathsf{T}\right) \\ &\text{s.t. } 0 \leq \mathbf{P}_{ij} \leq 1, \forall i, j, \mathbf{P}^\mathsf{T} = \mathbf{P}, \mathbf{W} = \mathbf{P} + \mathbf{E}, \\ &\qquad \mathbf{P}_{ij} = \mathbf{F}_{ij}, \forall (\mathbf{x}_i, \mathbf{x}_j) \in \Omega. \end{aligned} \tag{11}$$

After obtaining $\mathbf{P}$ via solving Eq. (11), we could perform spectral decomposition on $\mathbf{P}$ to obtain the final clustering result.

A. Optimization

Since both the feasible set and the objective function of Eq. (11) are convex, Eq. (11) is a convex optimization problem. To solve it, we adopt the inexact augmented Lagrange multiplier (IALM) algorithm [46], which solves a convex optimization problem by splitting it into smaller and easier-solvable sub-problems.

[1]Here, we simply use the EAM to build the Laplacian matrix $\mathbf{L}$, i.e., $\mathbf{L} = \mathbf{D} - \mathbf{W}$, where $\mathbf{D} \in \mathbb{R}^{n\times n}$ is the diagonal degree matrix of $\mathbf{W}$ with the ith diagonal element $\mathbf{D}_{ii} = \sum_j \mathbf{W}_{ij}$.

By introducing two auxiliary matrices $\mathbf{B} \in \mathbb{R}^{n\times n}$ and $\mathbf{C} \in \mathbb{R}^{n\times n}$, and setting $\mathbf{B} = \mathbf{P}$ and $\mathbf{C} = \mathbf{P}$, Eq. (11) is equivalently rewritten as

$$\begin{aligned} &\min_{\mathbf{P},\mathbf{E},\mathbf{B},\mathbf{C}} \|\mathbf{P}\|_* + \lambda\|\mathbf{E}\|_1 + \gamma \operatorname{Tr}\left(\mathbf{C}\mathbf{L}\mathbf{C}^\mathsf{T}\right) \\ &\text{s.t. } 0 \leq \mathbf{B}_{ij} \leq 1, \forall i, j, \mathbf{P}^\mathsf{T} = \mathbf{P}, \mathbf{W} = \mathbf{P} + \mathbf{E}, \\ &\qquad \mathbf{B}_{ij} = \mathbf{F}_{ij}, \forall (\mathbf{x}_i, \mathbf{x}_j) \in \Omega, \mathbf{B} = \mathbf{P}, \mathbf{C} = \mathbf{P}. \end{aligned} \tag{12}$$

The augmented Lagrangian form of Eq. (12) is,

$$\begin{aligned} &\operatorname*{argmin}_{\mathbf{P},\mathbf{E},\mathbf{B},\mathbf{C}} \|\mathbf{P}\|_* + \lambda\|\mathbf{E}\|_1 + \gamma \operatorname{Tr}\left(\mathbf{C}\mathbf{L}\mathbf{C}^\mathsf{T}\right) \\ &\quad + \langle \mathbf{Y}_1, \mathbf{W} - \mathbf{P} - \mathbf{E}\rangle + \frac{\mu}{2}\|\mathbf{W} - \mathbf{P} - \mathbf{E}\|_F^2 \\ &\quad + \langle \mathbf{Y}_2, \mathbf{P} - \mathbf{B}\rangle + \frac{\mu}{2}\|\mathbf{P} - \mathbf{B}\|_F^2 \\ &\quad + \langle \mathbf{Y}_3, \mathbf{P} - \mathbf{C}\rangle + \frac{\mu}{2}\|\mathbf{P} - \mathbf{C}\|_F^2 \\ &\text{s.t.} 0 \leq \mathbf{B}_{i,j} \leq 1, \forall ij, \mathbf{P}^\mathsf{T} = \mathbf{P}, \mathbf{B}_{ij} = \mathbf{F}_{ij}, \forall (\mathbf{x}_i, \mathbf{x}_j) \in \Omega, \end{aligned} \tag{13}$$

where $\langle \cdot, \cdot \rangle$ calculates the inner product of two matrices, $\|\cdot\|_F$ returns the Frobenius norm of a matrix, $\mathbf{Y}_1 \in \mathbb{R}^{n\times n}$, $\mathbf{Y}_2 \in \mathbb{R}^{n\times n}$ and $\mathbf{Y}_3 \in \mathbb{R}^{n\times n}$ are the Lagrangian multipliers, and μ is the positive scalar to introduce the augmented Lagrangian terms (i.e., $\|\mathbf{W} - \mathbf{P} - \mathbf{E}\|_F^2$, $\|\mathbf{P} - \mathbf{B}\|_F^2$ and $\|\mathbf{P} - \mathbf{C}\|_F^2$). To optimize Eq. (13) with the IALM, we separate it into the following four sub-problems.

1) The $\mathbf{P}$ subproblem: The $\mathbf{P}$ subproblem minimizes Eq. (13) over $\mathbf{P}$ with the other variables (i.e., $\mathbf{E}, \mathbf{B}, \mathbf{C}$) fixed, which is written as

$$\min_{\mathbf{P}} \frac{1}{3\mu}\|\mathbf{P}\|_* + \frac{1}{2}\left\|\mathbf{P} - \frac{1}{3}(\mathbf{T}_1 + \mathbf{T}_2 + \mathbf{T}_3)\right\|_F^2, \text{s.t. } \mathbf{P}^\mathsf{T} = \mathbf{P}, \tag{14}$$

where $\mathbf{T}_1 = \left(\mathbf{W} - \mathbf{E} + \frac{\mathbf{Y}_1}{\mu}\right)$, $\mathbf{T}_2 = \left(\mathbf{B} - \frac{\mathbf{Y}_2}{\mu}\right)$, and $\mathbf{T}_3 = \left(\mathbf{C} - \frac{\mathbf{Y}_3}{\mu}\right)$. Eq. (14) is a symmetric constrained nuclear norm minimization problem, which has a closed form solution given by Theorem 1.

Theorem 1 [47] *Given any square matrix $\mathbf{D} \in \mathbb{R}^{n\times n}$, the optimal solution to*

$$\min_{\mathbf{Q}} \tau\|\mathbf{Q}\|_* + \frac{1}{2}\|\mathbf{Q} - \mathbf{D}\|_F^2, \text{s.t. } \mathbf{Q} = \mathbf{Q}^\mathsf{T} \tag{15}$$

is identical to the solution of

$$\min_{\mathbf{Q}} \tau\|\mathbf{Q}\|_* + \frac{1}{2}\left\|\mathbf{Q} - \frac{1}{2}\left(\mathbf{D} + \mathbf{D}^\mathsf{T}\right)\right\|_F^2. \tag{16}$$

Particularly, Eq. (16) has a closed form solution, i.e., the singular value thresholding operator [26].

2) The $\mathbf{E}$ subproblem: Without the irrelevant variables, the $\mathbf{E}$ subproblem is written as

$$\min_{\mathbf{E}} \lambda\|\mathbf{E}\|_1 + \frac{\mu}{2}\left\|\mathbf{W} - \mathbf{P} - \mathbf{E} + \frac{\mathbf{Y}_1}{\mu}\right\|_F^2, \tag{17}$$

which is an ℓ_1 norm minimization problem with a closed form solution by soft-thresholding [46].

Algorithm 1 Solve the Problem in Eq. (11) Using IALM

Input: $\mathbf{W}$, $\mathbf{F}$, γ, λ, Ω;

Initialize: $\mathbf{P} = \mathbf{E} = \mathbf{B} = \mathbf{C} = \mathbf{Y}_1 = \mathbf{Y}_2 = \mathbf{Y}_3 = \mathbf{0}_{n\times n}$, $\mu = 10^{-4}$, $\mu_{max} = 10^8$;

while not converged **do**
- Update $\mathbf{P}$ by Eq. (14);
- Update $\mathbf{E}$ by Eq. (17);
- Update $\mathbf{B}$ by Eq. (19);
- Update $\mathbf{C}$ by Eq. (21);
- Update $\mathbf{Y}_1$, $\mathbf{Y}_2$, $\mathbf{Y}_3$, and μ by Eq. (22);
- Check the convergence conditions

$$\|\mathbf{P}-\mathbf{B}\|_\infty < 10^{-8}, \|\mathbf{P}-\mathbf{C}\|_\infty < 10^{-8} \text{ and } \|\mathbf{W}-\mathbf{P}-\mathbf{E}\|_\infty < 10^{-8}.$$

end while

3) The $\mathbf{B}$ *subproblem:* With fixed $\mathbf{P}$, $\mathbf{B}$, and $\mathbf{C}$, the $\mathbf{B}$ sub-problem is written as

$$\begin{aligned} &\min_{\mathbf{B}} \frac{\mu}{2}\left\|\mathbf{P}-\mathbf{B}+\frac{\mathbf{Y}_2}{\mu}\right\|_F^2 \\ &\text{s.t. } 0 \le \mathbf{B}_{ij} \le 1, \forall i,j, \mathbf{B}_{ij} = \mathbf{F}_{ij}, \forall(\mathbf{x}_i,\mathbf{x}_j) \in \Omega, \end{aligned} \tag{18}$$

which is a bounded quadratic equation. Eq. (18) can be solved in element-wise with a closed form solution:

$$\mathbf{B}_{ij} = \begin{cases} \mathbf{F}_{i,j}, & \text{if } (\mathbf{x}_i,\mathbf{x}_j) \in \Omega \\ 1, & \text{if } \left(\mathbf{P}+\frac{\mathbf{Y}_2}{\mu}\right)_{ij} \ge 1 \\ 0, & \text{if } \left(\mathbf{P}+\frac{\mathbf{Y}_2}{\mu}\right)_{ij} \le 0 \\ \left(\mathbf{P}+\frac{\mathbf{Y}_2}{\mu}\right)_{ij}, & \text{others.} \end{cases} \tag{19}$$

4) The $\mathbf{C}$ *subproblem:* Removing the irrelevant variables to $\mathbf{C}$, the $\mathbf{C}$ sub-problem is

$$\min_{\mathbf{C}} \gamma \mathrm{Tr}\left(\mathbf{C}\mathbf{L}\mathbf{C}^\mathsf{T}\right) + \frac{\mu}{2}\left\|\mathbf{P}-\mathbf{C}+\frac{\mathbf{Y}_3}{\mu}\right\|_F^2, \tag{20}$$

which is an unconstrained least-squares problem. The global optimum is obtained when the first order derivative of Eq. (20) with respect to $\mathbf{C}$ equals to 0, i.e.,

$$\mathbf{C} = (\mu\mathbf{P}+\mathbf{Y}_3)(2\gamma\mathbf{L}+\mu\mathbf{I})^{-1}, \tag{21}$$

where $\mathbf{I} \in \mathbb{R}^{n\times n}$ is the identity matrix of size $n \times n$.

Finally, the Lagrangian multiplier matrices and μ are updated by

$$\begin{cases} \mathbf{Y}_1 \leftarrow \mathbf{Y}_1 + \mathbf{W} - \mathbf{P} - \mathbf{E} \\ \mathbf{Y}_2 \leftarrow \mathbf{Y}_2 + \mathbf{P} - \mathbf{B} \\ \mathbf{Y}_3 \leftarrow \mathbf{Y}_3 + \mathbf{P} - \mathbf{C} \\ \mu = \min(1.1\mu, \mu_{max}). \end{cases} \tag{22}$$

The overall optimization procedure is summarized in Algorithm 1.

B. Complexity and Convergence of Algorithm 1

The computational complexity of Algorithm 1 is mainly determined by *steps 2* and *3*. Specifically, *step 2* needs to solve the singular value decomposition (SVD) of an $n \times n$ matrix with the complexity of $\mathcal{O}(n^3)$. When we use the partial SVD [48], the complexity of this step can be reduced to $\mathcal{O}(rn^2)$ with r being the lowest rank of $\mathbf{P}$. For *step 3*, we need to calculate the inverse of a matrix of size $n \times n$ and the matrix product between two $n \times n$ matrices. Therefore, the computational complexity of the *step 3* is $\mathcal{O}(n^3)$. In summary, the overall computational complexity of the Algorithm 1 is $\mathcal{O}(rn^2 + n^3)$ for each iteration.

Remark: it is worth pointing out that as the matrix $2\gamma\mathbf{L}+\mu\mathbf{I}$ in *step 3* is very sparse, its inverse can be obtained fast in practice.

Algorithm 1 solves the original problem with 4 blocks of variables; however, to the best of our knowledge, there is no general convergence proof for the IALM algorithm with more than 2 block of variables. Fortunately, owing to the convexity of the proposed model and the fact that each subproblem has a closed form solution, Algorithm 1 empirically converges well. See the corresponding results in Section IV.

IV. Experimental Results

We compared the proposed model with the following state-of-the-art CSC methods:

- E2CP [8] first propagates the initial PCs to the whole data set, then refines the original affinity matrix based on the propagated PCs to generate a more informative affinity matrix;
- GSM [7] incorporates the PCs into two Laplacian matrices and then minimizes a Rayleigh quotient via solving a generalized eigen-decomposition problem;
- SNMFCC 2016 [13] is a CSC method based on SNMF;
- PCPSNMF [20] performs PCP and SNMF simultaneously to realize CSC;
- S3C [12] is a CSC method with a structured sparsity regularization term to encode the PCs;
- SL [14] realizes CSC by modifying the EAM according to the PCM; and
- LRPCP [18] is a PCP method based on low-rank matrix learning;

and the following semi-supervised self-representation-based affinity matrix learning methods:

- SSCPCP [49] performs sparse subspace clustering and PCP simultaneously in a semi-supervised fashion;
- StruLRR [50] is a semi-supervised low-rank representation method with a structured regularizer constructed according to the PCM; and
- SSLRR [28] is a semi-supervised low rank representation method for affinity matrix construction.

The experiments were conducted on eight commonly used data sets with various data types, including one face data set, two handwritten digit data sets and five UCI data sets. See Table II for the details of these data sets.

To evaluate the clustering performance, we adopted two commonly used metrics, i.e., clustering accuracy (ACC) and normalized mutual information (NMI). See [34] for the detailed definitions of ACC and NMI. Both ACC and NMI lie in the range of $[0, 1]$, and the larger, the better.

TABLE II
DATA SETS DESCRIPTION

Dataset	# Sample (n)	# Dimension (d)	# Cluster (c)	Source
UMIST	575	644	20	Face
USPS	400	256	10	Digital
MNIST	500	784	10	Digital
CHART	600	60	6	UCI
COTTON	356	21	6	UCI
LIBRAS	360	90	15	UCI
USER	403	5	4	UCI
SOLAR	323	12	6	UCI

For fair comparisons, the EAM for all the methods was constructed by a k-nearest neighbours (k-nn) graph with a radial basis function (RBF) kernel. Considering that the EAM affects performance severely, for all the methods we exhaustively searched k and the bandwidth from the ranges of $\{3, 6, 9, 12\}$ and $\{0.01, 0.1, 1\}$, respectively, and used the ones producing the best performance. After obtaining the spectral decomposition of each method, K-means was performed on those decompositions to generate the final clustering result. There are two hyper-parameters λ and γ in our model, which were determined by exhaustive search from the ranges of $\{0.0001, 0.001, 0.01, 0.1, 1, 10\}$ and $\{0.001, 0.01, 0.1, 1, 10, 100\}$, respectively. Note that, for fair comparisons, the hyper-parameters of all the methods under comparison were also exhaustively searched in the ranges suggested by the original papers, and the ones producing the best performance were adopted. The PCs were randomly generated from each data set according to the ground-truth label. To investigate the influence of the number of PCs on performance, the number of PCs varies from 500 to 1000 with a step of 100 for each data set. Considering that the random selection of PCs may severely affect the performance of different methods, we repeated the experiments 10 times, and reported the average results for all the methods.

A. Comparison of Clustering Performance

Figs. 2 and 3 show the values of ACC and NMI of different methods with different numbers of PCs on all the eight data sets, where we can draw the following conclusions.

- With the increase of the number of PCs, all the constrained clustering models generally perform better, which suggests the importance and effectiveness of PCs.
- Our model consistently performs better than the compared methods on all the data sets with different numbers of PCs in terms of both ACC and NMI. Especially, on LIBRAS, our method improves the ACC 10.3%, compared with the second best method.
- The compared methods highly depend on the characteristics of data sets. For example, E2CP produces high ACC on COTTON, but very low ACC on UMIST. The value of ACC of GSM on SOLAR is high, while on USPS, GSM obtains the low values of NMI with different numbers of PCs. The self-representation-based affinity matrix learning methods like SSCPCP works well on USPS and LIBRAS, but poorly on the remaining data sets. The reason is that the self-representation-based affinity matrix learning methods are built on a strong assumption that each sample can be linearly reconstructed from other samples from the same cluster, and the performance will degrade if the data do not satisfy this assumption. On the contrary, our model is able to produce both high values of ACC and NMI on all the eight data sets, indicating the robustness of our method to different data sets.

Fig. 4 shows the distributions of NMI values of all the CSC methods with 1000 PCs on all the data, from which we can see that the proposed method usually has the highest median value compared with the other methods, which further confirms the advantage of our method in clustering. Moreover, the length of the boxes corresponding to our method is relatively short on all data sets, which proves that our model is robust to the selection of PCs.

B. Visual Comparison of the Learned Affinity Matrix

Fig. 5 visually compares the EAM constructed by the k-nn graph with a RBF kernel, the typical learned affinity matrices of the proposed method and the compared methods with 1000 PCs, and the ideal affinity matrix on USPS. Obviously, the affinity matrices produced by E2CP and PCPCSNMF are sparse and close to the EAM. On the contrary, both the affinity matrices by SSLRR and our method are denser with a salient block diagonal structure. Moreover, there are less error connections in the affinity matrix by our method than SSLRR. These visual results demonstrate that our model is able to learn a more informative affinity matrix than the compared methods, and also explain why the proposed method can achieve better clustering performance as shown in Figs. 2 and 3.

C. Parameter Sensitivity

In this section, we investigated the sensitivity of the two hyper-parameters involved in our method, i.e., λ and γ[2]. Fig. 6 shows the ACC values with respect to λ and γ on all the eight data sets with 1000 PCs. It can be observed that, when λ or γ gets close to 0, the value of ACC drops sharply. On the other hand, when both λ and γ become larger, our model tends to be stable and produces high value of ACC on all the eight data sets, which validates the robustness of our model to those two hyper-parameters. We suggest to set λ and γ to 0.01 and 100, respectively, in the practical applications.

D. Convergence Behaviour

Although there is no general theoretical convergence guarantee for the IALM-like algorithms with more than two blocks of variables, we empirically found that the proposed algorithm converges well. Fig. 7 shows the convergence curves of the proposed algorithm on all the eight data sets with 1000 PCs. As we can observe, the proposed algorithm gets converged, i.e., $\max\{\|\mathbf{P}-\mathbf{B}\|_\infty, \|\mathbf{P}-\mathbf{C}\|_\infty, \|\mathbf{W}-\mathbf{P}-\mathbf{E}\|_\infty\} < 10^{-8}$ in around 250 iterations on all the data sets, which demonstrates the efficiency of the proposed optimization method.

[2]For the EAM, k was empirically set to $\lfloor \log_2(n) + 1 \rfloor$ with $\lfloor \cdot \rfloor$ being the round toward negative infinity operator, and the bandwidth of the RBF kernel was set to mean distance of the current sample to its k nearest neighbours.

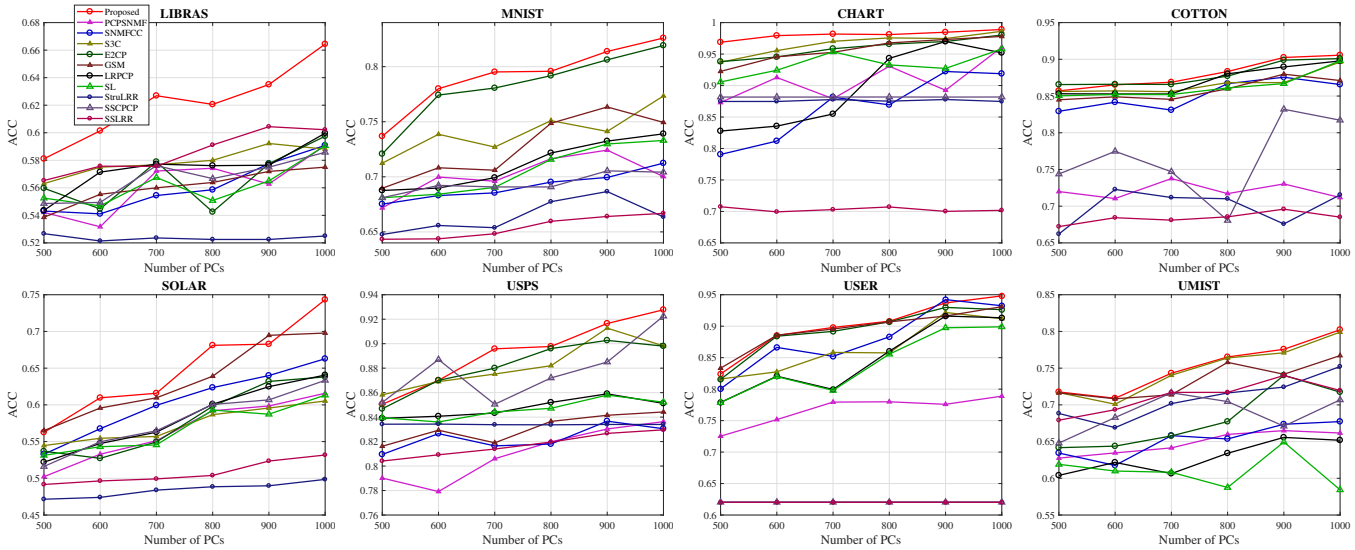

Fig. 2. Comparisons of ACCs by different methods on eight data sets with different numbers of PCs. The randomly selected PCs account for a very small propagation of all the potential PCs, e.g., 500 PCs are only 0.13% of all the potential PCs in CHART. All the sub-figures share the same legend.

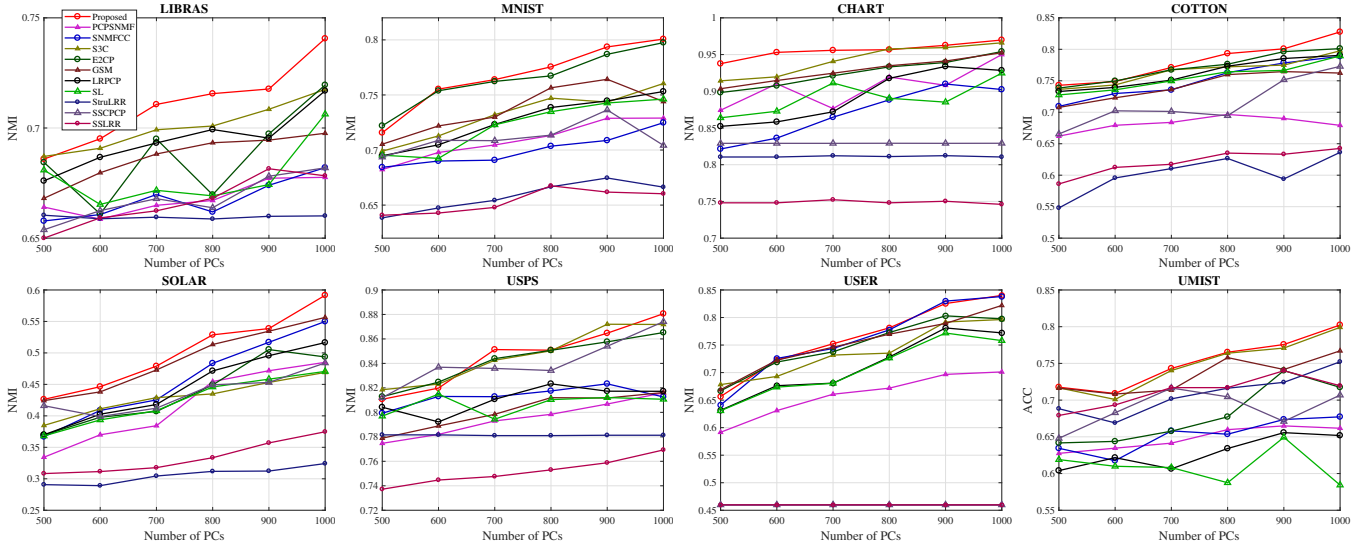

Fig. 3. Comparisons of NMIs by different methods on eight data sets with different numbers of PCs. All the sub-figures share the same legend.

E. Comparison of Running Time

In this subsection, we compared the actual running time of different methods. Specifically, all the methods were implemented with MATLAB on a Windows computer with a 3.7GHz CPU and 32.0 GB memory. As we can see from Fig. 8, E2CP and SL are the most efficient methods, due to the simplicity. S3C, LRPCP and the proposed method have the comparable running time on all the data sets, since they have a similar computational complexity. The self-representation-based methods including SSLRR, SSCPCP and StruLRR always consume more times on all the data sets. The reason is that the computational complexities of S3C, LRPCP and our method are only related to the number of samples, while those of the self-representation-based methods are also related to the dimension of features, making it inefficient for the input with high-dimensional features.

F. Ablation Study

In this subsection, we investigated how the prior terms affect the performance of the proposed model. Table III shows the results of ablation studies. Specifically, w/oL and w/oS, denote the proposed model without $\mathrm{Tr}\left(\mathbf{PLP}^{\mathsf{T}}\right)$ and without the symmetric constraint (i.e., $\mathbf{P}=\mathbf{P}^{\mathsf{T}}$), respectively. w/oS-Post denotes the case of w/oS with a post symmetric processing, i.e., symmetrize $\mathbf{P}$ by $\mathbf{P}=\frac{1}{2}(\mathbf{P}+\mathbf{P}^{\mathsf{T}})$ after solving w/oS. As we can see from Table III, the proposed model almost always outperforms all the compared cases, indicating that all the adopted prior terms are useful and contribute to our model. Especially, according to the last column, the proposed method performs much better than the compared cases averagely with respect to both ACC and NMI. Moreover, the w/oL usually performs the worst compared to other cases, suggesting the importance of the local graph term. It is also worth pointing

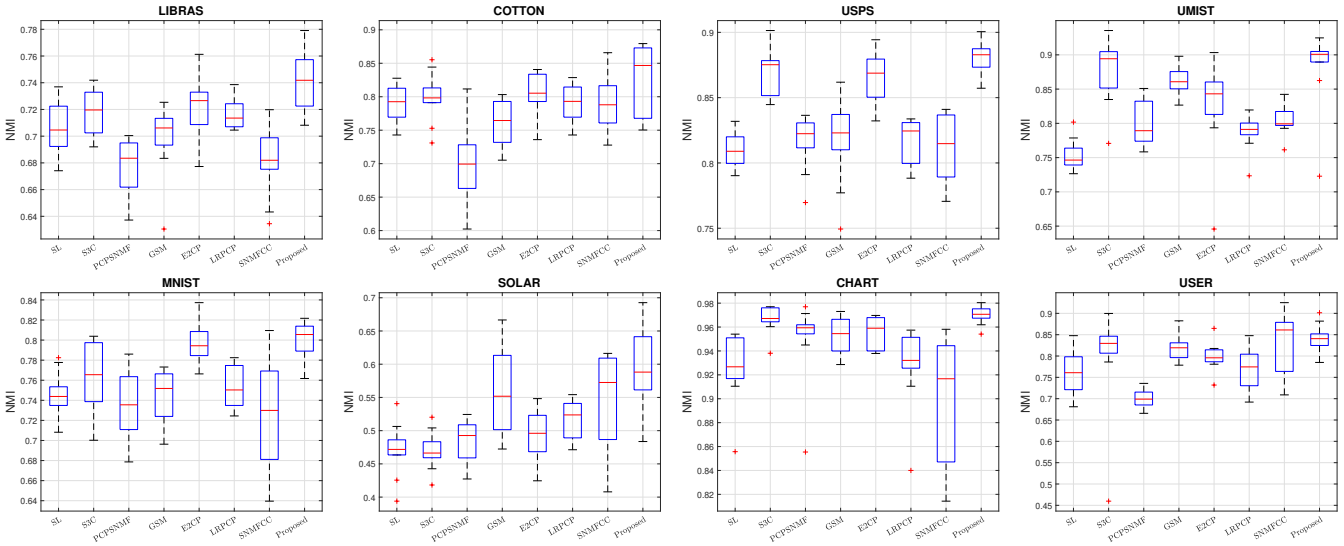

Fig. 4. The comparison of the boxplots of the NMI of all the methods with 1000 PCs on all the datasets.

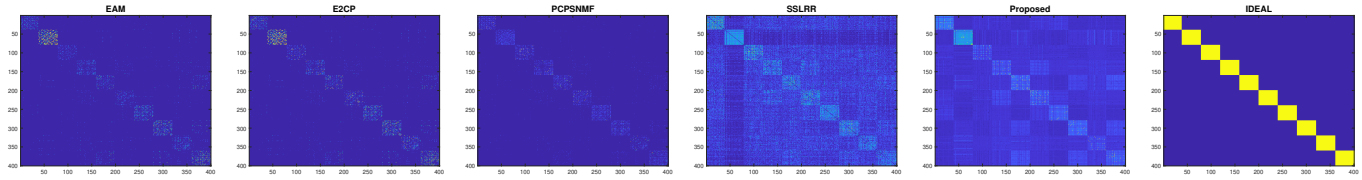

Fig. 5. Visual comparison of the learned affinity matrices for different methods.

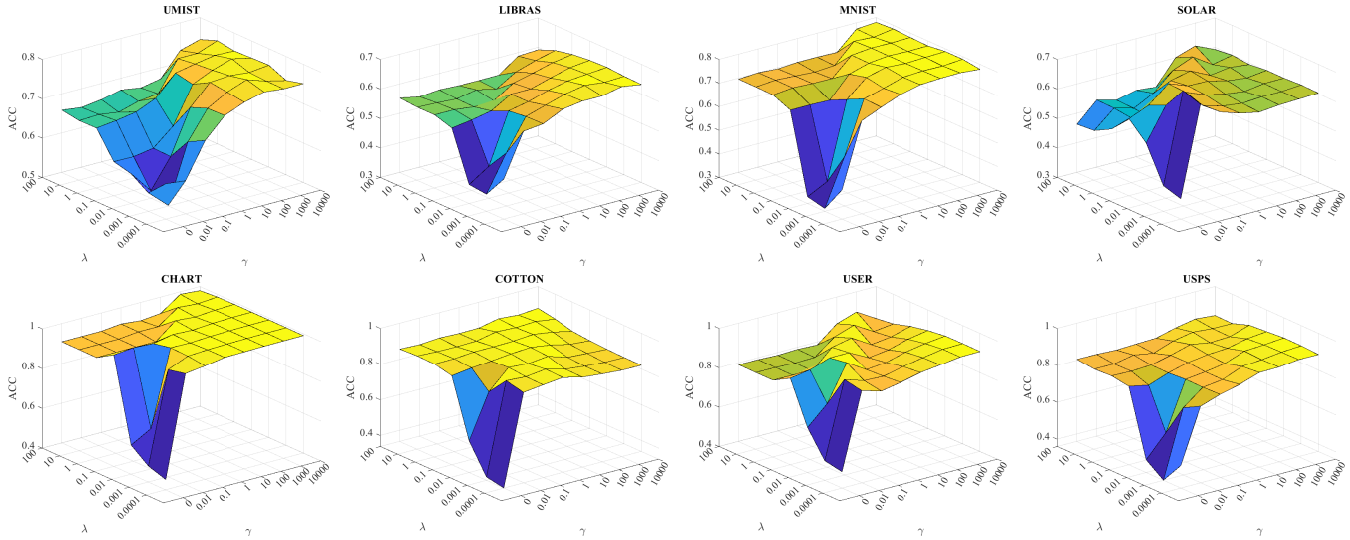

Fig. 6. ACC with respect to λ and γ on all the data sets with 1000 PCs.

out that w/oS-Post performs better than w/oS but worse than the proposed one. This suggests that incorporating the symmetric constraint in the overall objective is superior to the one with a post processing.

G. Application to Semi-Supervised Dimensionality Reduction

In this subsection, we investigated the performance of the proposed method on dimensionality reduction. Specifically, we applied the learned affinity matrix to GPCA [33], i.e.,

$$\min_{\mathbf{U},\mathbf{Y}} \|\mathbf{X}-\mathbf{U}^{\mathsf{T}}\mathbf{Y}\|_F^2 + \eta \mathrm{tr}\left(\mathbf{Y}\mathbf{L}\mathbf{Y}^{\mathsf{T}}\right), \quad \text{s.t.} \quad \mathbf{U}\mathbf{U}^{\mathsf{T}} = \mathbf{I}, \tag{23}$$

where $\mathbf{U} \in \mathbb{R}^{k\times d}$ is the basis matrix, $\mathbf{Y} \in \mathbb{R}^{k\times n}$ is the lower-dimensional embedding matrix, k is the dimension of $\mathbf{Y}$, $\eta \geq 0$ introduces the graph Laplacian regularization to PCA. $\mathbf{L} = \mathbf{D} - \mathbf{W}$ is the graph Laplacian matrix, where $\mathbf{D}$ is a diagonal

TABLE III
ABLATION STUDY

# PCs	**ACC**	CHART	COTTON	LIBRAS	SOLAR	MNIST	UMIST	USER	USPS	**Average**
500	w/oS	0.9465	0.8331	0.5142	0.3560	0.6728	0.6614	0.7504	0.8342	0.6961
	w/oS-Post	0.9722	0.8669	0.5481	0.5427	0.6904	0.6614	0.8089	0.8450	0.7419
	w/oL	0.9422	0.8511	0.5400	0.5229	0.6784	0.5843	0.7784	0.8380	0.7169
	Proposed	0.9688	0.8570	0.5811	0.5625	0.7368	0.7176	0.8241	0.8505	**0.7623**
700	w/oS	0.6650	0.7674	0.5706	0.6115	0.5538	0.6294	0.6754	0.8160	0.6611
	w/oS-Post	0.9762	0.8677	0.5825	0.6167	0.7298	0.7172	0.8618	0.8547	0.7758
	w/oL	0.9550	0.8654	0.5769	0.5505	0.7002	0.6042	0.8228	0.8432	0.7397
	Proposed	0.9818	0.8688	0.6269	0.6158	0.7952	0.7431	0.8980	0.8958	**0.8032**
1000	w/oS	0.9858	0.9034	0.5867	0.5731	0.6136	0.6311	0.9020	0.6910	0.7358
	w/oS-Post	0.9873	0.9034	0.5997	0.7257	0.7414	0.6776	0.9176	0.8660	0.8023
	w/oL	0.9712	0.8972	0.5869	0.6675	0.7330	0.6440	0.8990	0.8453	0.7805
	Proposed	0.9890	0.9056	0.6644	0.7433	0.826	0.8023	0.9481	0.9277	**0.8508**
# PCs	**NMI**	CHART	COTTON	LIBRAS	SOLAR	MNIST	UMIST	USER	USPS	**Average**
500	w/oS	0.9240	0.7234	0.6459	0.2498	0.6567	0.7893	0.4651	0.8022	0.6570
	w/oS-Post	0.9425	0.7459	0.6740	0.3937	0.6974	0.7921	0.6327	0.8110	0.7111
	w/oL	0.8956	0.7321	0.6739	0.3652	0.6959	0.7469	0.6315	0.8088	0.6937
	Proposed	0.9376	0.7428	0.6859	0.4262	0.7157	0.824	0.6565	0.8107	**0.7249**
700	w/oS	0.9371	0.6203	0.6726	0.4614	0.5413	0.7769	0.7178	0.8314	0.6948
	w/oS-Post	0.9499	0.7712	0.6898	0.4614	0.7245	0.8099	0.7202	0.8314	0.7447
	w/oL	0.9177	0.7711	0.6813	0.4196	0.7242	0.7606	0.6806	0.8062	0.7201
	Proposed	0.9558	0.7711	0.7107	0.4790	0.7639	0.8363	0.7523	0.8513	**0.7651**
1000	w/oS	0.9635	0.8045	0.6961	0.1703	0.5167	0.7672	0.7945	0.6903	0.6753
	w/oS-Post	0.9669	0.8045	0.7111	0.5739	0.7517	0.8112	0.8012	0.8398	0.7825
	w/oL	0.9394	0.7893	0.7039	0.5163	0.7474	0.7727	0.7585	0.8140	0.7551
	Proposed	0.9698	0.8276	0.7406	0.5916	0.8007	0.8821	0.8401	0.8806	**0.8166**

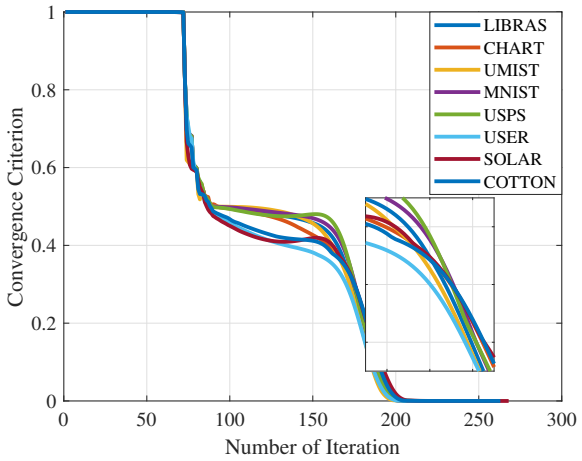

Fig. 7. Illustration of the convergence behavior of the proposed optimization algorithm.

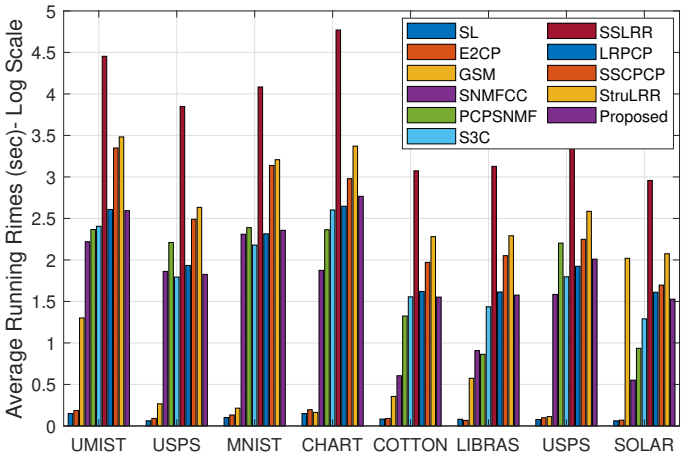

Fig. 8. Running time comparison of different methods on all the data sets.

matrix with the diagonal element equal to the sum of each row of $\mathbf{W}$.

To apply the proposed method and the compared methods to GPCA, we could replace the $\mathbf{W}$ with the learned affinity matrices by different methods. To evaluate the qualities of the affinity matrices, we first used GPCA to reduce the dimension of the input data set, and then applied the nearest neighbors classifier and compared the classification accuracy. Specifically, for each data set, we randomly selected $1/2/3$ samples per class as the training set, and reported the classification accuracy on the remaining samples. To reduce the influence of the random selection of the training set, we repeated the selection for 20 times and reported the average results. For all the data sets, the number of PCs was set to 1000, the dimension of $\mathbf{Y}$ was searched from $\{0.5c, c, 2c\}$, where c denotes the number of clusters[3], and η was searched from the range of $\{0.001, 0.01, 0.1, 1, 10, 100, 1000\}$.

The experimental results are shown in Table IV, where we could draw the following conclusions. GPCA always outperforms PCA, validating the importance of the affinity relation in dimensionality reduction. Moreover, since PCs can provide supervisory information, the affinity matrices constructed with the guidance of PCs surpass the one with EAM. GPCA with the proposed method achieves the highest classification accu-

[3]Since the dimension of USER is 5, and cluster number of it is 4, the lower dimension of USER ranges from $\{2, 3, 4\}$.

racy in most cases ($21/24$), and in the remaining cases ($3/24$), the proposed method achieves the second highest classification accuracy. Besides, the improvements of our method over the second best ones on all the data sets are significant. For example, on MNIST with 1 labeled sample per class, UMIST with 1 labeled samples per class, and USER with 1 labeled sample per class, the classification accuracy increases 25.8%, 13.7%, and 9.1%, respectively, compared with the second best methods. These results substantiate the excellent performance of the learned LAM in dimensionality reduction.

V. Conclusion and Future Work

We have presented a novel semi-supervised affinity learning model. Different from the existing methods, we assumed that there is an LAM to depict the ideal pairwise relationships among the data samples, and the PCM and EAM play different roles in recovering the LAM. Technically, we formulated the LAM learning as a convex symmetric constrained matrix completion problem, and solved it with the IALM method. We applied proposed model on CSC and dimensionality reduction, and extensive experimental comparisons on 8 data sets demonstrate that our model outperforms the state-of-the-art methods.

There are several valuable working directions to be investigated. First, the proposed model can be extended naturally to solve the multi-view clustering problem given multiple initial EAMs, where each affinity matrix is constructed by a typical view. Second, due to the matrix inverse and the SVD operation involved in the optimization algorithm, the computational complexity of the proposed optimization method is quite high. How to reduce the computational complexity is important. Third, the ideal LAM is block diagonal, and thus the recently proposed block diagonal prior [51] could be studied here. Last but not least, some theoretical analyses, like convergence conditions, could be investigated.

References

[1] Y. Jia, W. Wu, R. Wang, J. Hou, and S. Kwong, "Joint optimization for pairwise constraint propagation," *IEEE Trans. Neural Netw. Learn. Syst.*, pp. 1–13, 2020.

[2] Y. Jia, J. Hou, and S. Kwong, "Constrained clustering with dissimilarity propagation-guided graph-laplacian pca," *IEEE Trans. Neural Netw. Learn. Syst.*, pp. 1–13, 2020.

[3] H. Li, S. Kwong, C. Chen, Y. Jia, and R. Cong, "Superpixel segmentation based on square-wise asymmetric partition and structural approximation," *IEEE Trans. Multimedia*, 2019.

[4] Z. Fu, W. Hu, and T. Tan, "Similarity based vehicle trajectory clustering and anomaly detection," in *Proc. IEEE ICIP*, vol. 2. IEEE, 2005, pp. II–602.

[5] W. Wu, S. Kwong, Y. Zhou, Y. Jia, and W. Gao, "Nonnegative matrix factorization with mixed hypergraph regularization for community detection," *Information Sciences*, vol. 435, pp. 263–281, 2018.

[6] Y. Jia, H. Liu, J. Hou, and S. Kwong, "Clustering-aware graph construction: A joint learning perspective," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 6, pp. 357–370, 2020.

[7] C. Jiang, H. Xie, and Z. Bai, "Robust and efficient computation of eigenvectors in a generalized spectral method for constrained clustering," in *Artificial Intelligence and Statistics*, 2017, pp. 757–766.

[8] Z. Lu and Y. Peng, "Exhaustive and efficient constraint propagation: A graph-based learning approach and its applications," *International Journal of Computer Vision*, vol. 103, no. 3, pp. 306–325, 2013.

[9] Z. Li, J. Liu, and X. Tang, "Constrained clustering via spectral regularization," in *Proc. IEEE CVPR*. IEEE, 2009, pp. 421–428.

[10] Y. Jia, H. Liu, J. Hou, and S. Kwong, "Semisupervised adaptive symmetric non-negative matrix factorization," *IEEE Trans. Cybern.*, pp. 1–1, 2020.

[11] Y. Jia, S. Kwong, J. Hou, and W. Wu, "Semi-supervised non-negative matrix factorization with dissimilarity and similarity regularization," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 7, pp. 2510–2521, 2020.

[12] Y. Jia, S. Kwong, and J. Hou, "Semi-supervised spectral clustering with structured sparsity regularization," *IEEE Signal Process. Lett.*, vol. 25, no. 3, pp. 403–407, March 2018.

[13] 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, July 2016.

[14] K. Kamvar, S. Sepandar, K. Klein, D. Dan, M. Manning, and C. Christopher, "Spectral learning," in *Proc. IJCAI*. Stanford InfoLab, 2003.

[15] Y. Jia, H. Liu, J. Hou, and S. Kwong, "Pairwise constraint propagation with dual adversarial manifold regularization," *IEEE Trans. Neural Netw. Learn. Syst.*, pp. 1–13, 2020.

[16] H. Liu, Y. Jia, J. Hou, and Q. Zhang, "Imbalance-aware pairwise constraint propagation," in *Proc. ACM MM*, 2019, pp. 1605–1613.

[17] Z. Fu, H. H. Ip, H. Lu, and Z. Lu, "Multi-modal constraint propagation for heterogeneous image clustering," in *Proc. ACM MM*. ACM, 2011, pp. 143–152.

[18] Z. Yang, Y. Hu, H. Liu, H. Chen, and Z. Wu, "Matrix completion for cross-view pairwise constraint propagation," in *Proc. ACM MM*. ACM, 2014, pp. 897–900.

[19] B. Kulis, S. Basu, I. Dhillon, and R. Mooney, "Semi-supervised graph clustering: a kernel approach," *Machine learning*, vol. 74, no. 1, pp. 1–22, 2009.

[20] W. Wu, Y. Jia, S. Kwong, and J. Hou, "Pairwise constraint propagation-induced symmetric nonnegative matrix factorization," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 12, pp. 6348–6361, Dec 2018.

[21] Z. Li, J. Liu, Y. Yang, X. Zhou, and H. Lu, "Clustering-guided sparse structural learning for unsupervised feature selection," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 9, pp. 2138–2150, 2014.

[22] D. Luo, C. Ding, and H. Huang, "Towards structural sparsity: An explicit l2/l0 approach," in *Proc. IEEE ICDM*, 2010, pp. 344–353.

[23] Y. Jia, S. Kwong, W. Wu, R. Wang, and W. Gao, "Sparse bayesian learning-based kernel poisson regression," *IEEE Trans. Cybern.*, vol. 49, no. 1, pp. 56–68, 2019.

[24] U. Von Luxburg, "A tutorial on spectral clustering," *Statistics and computing*, vol. 17, no. 4, pp. 395–416, 2007.

[25] X. Wang, B. Qian, and I. Davidson, "On constrained spectral clustering and its applications," *Data Mining and Knowledge Discovery*, vol. 28, no. 1, pp. 1–30, 2014.

[26] 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, 2012.

[27] C. Li, Z. Lin, H. Zhang, and J. Guo, "Learning semi-supervised representation towards a unified optimization framework for semi-supervised learning," in *Proc. ICCV*, 2015, pp. 2767–2775.

[28] L. Zhuang, Z. Zhou, S. Gao, J. Yin, Z. Lin, and Y. Ma, "Label information guided graph construction for semi-supervised learning," *IEEE Trans. Image Process.*, vol. 26, no. 9, pp. 4182–4192, Sep. 2017.

[29] 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, 2017.

[30] X. Fang, N. Han, W. K. Wong, S. Teng, J. Wu, S. Xie, and X. Li, "Flexible affinity matrix learning for unsupervised and semisupervised classification," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 4, pp. 1133–1149, April 2019.

[31] W. Wu, Y. Jia, S. Wang, R. Wang, H. Fan, and S. Kwong, "Positive and negative label-driven nonnegative matrix factorization," *IEEE Trans. Circuits Syst. Video Technol.*, pp. 1–1, 2020.

[32] W. Wu, S. Kwong, J. Hou, Y. Jia, and H. H. S. Ip, "Simultaneous dimensionality reduction and classification via dual embedding regularized nonnegative matrix factorization," *IEEE Trans. Image Process.*, vol. 28, no. 8, pp. 3836–3847, 2019.

[33] 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, July 2013.

[34] 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.

[35] X. Wang, T. Zhang, and X. Gao, "Multiview clustering based on non-negative matrix factorization and pairwise measurements," *IEEE Trans. Cybern.*, vol. 49, no. 9, pp. 3333–3346, 2019.

[36] D. Cai, X. He, and J. Han, "Locally consistent concept factorization for document clustering," *IEEE Trans. Knowl. Data Eng.*, vol. 23, no. 6, pp. 902–913, 2010.

TABLE IV
CLASSIFICATION ACCURACY COMPARISON FOR DIMENSIONALITY REDUCTION

Data Sets	# LS	PCA	GPCA+EAM	GPCA+initial PCs	GPCA+E2CP	GPCA+PCPSNMF	GPCA+SSLRR	GPCA+Proposed
	1	0.3794 ± 0.0824	0.3794 ± 0.0895	0.3913 ± 0.0829	0.4178 ± 0.0844	0.4283 ± 0.0867	0.3643 ± 0.0736	$\mathbf{0.4529 \pm 0.0784}$
SOLAR	2	0.4091 ± 0.0505	0.4117 ± 0.0591	0.4689 ± 0.0507	0.4876 ± 0.0500	0.4926 ± 0.0472	0.4234 ± 0.0659	$\mathbf{0.5088 \pm 0.0634}$
	3	0.4572 ± 0.0489	0.4672 ± 0.0520	0.5105 ± 0.0423	0.5203 ± 0.0386	0.5230 ± 0.0346	0.4570 ± 0.0460	$\mathbf{0.5439 \pm 0.0640}$
	# LS	PCA	GPCA+EAM	GPCA+initial PCs	GPCA+E2CP	GPCA+PCPSNMF	GPCA+SSLRR	GPCA+Proposed
	1	0.3999 ± 0.0637	0.4527 ± 0.0679	0.4554 ± 0.0678	0.4564 ± 0.0685	0.4744 ± 0.0662	0.4050 ± 0.0600	$\mathbf{0.5968 \pm 0.0837}$
MNIST	2	0.5169 ± 0.0397	0.5577 ± 0.0553	0.5595 ± 0.0536	0.5601 ± 0.0537	0.6091 ± 0.0405	0.5433 ± 0.0353	$\mathbf{0.6859 \pm 0.0556}$
	3	0.5577 ± 0.0270	0.6063 ± 0.0296	0.6080 ± 0.0288	0.6086 ± 0.0288	0.6422 ± 0.0308	0.5803 ± 0.0348	$\mathbf{0.7133 \pm 0.0427}$
	# LS	PCA	GPCA+EAM	GPCA+initial PCs	GPCA+E2CP	GPCA+PCPSNMF	GPCA+SSLRR	GPCA+Proposed
	1	0.6102 ± 0.0613	0.7795 ± 0.0648	0.7880 ± 0.0607	$\mathbf{0.8031 \pm 0.0550}$	0.7779 ± 0.0617	0.6154 ± 0.0653	0.7947 ± 0.0627
CHART	2	0.7387 ± 0.0423	0.8982 ± 0.0333	0.9139 ± 0.0336	$\mathbf{0.9196 \pm 0.0349}$	0.8807 ± 0.0765	0.7542 ± 0.0526	0.9150 ± 0.0328
	3	0.7375 ± 0.0587	0.8761 ± 0.0512	0.8934 ± 0.0531	$\mathbf{0.9029 \pm 0.0540}$	0.8670 ± 0.0847	0.7505 ± 0.0562	0.8934 ± 0.0531
	# LS	PCA	GPCA+EAM	GPCA+initial PCs	GPCA+E2CP	GPCA+PCPSNMF	GPCA+SSLRR	GPCA+Proposed
	1	0.4997 ± 0.0584	0.6109 ± 0.0668	0.6489 ± 0.0677	0.6524 ± 0.0620	0.6624 ± 0.0650	0.5539 ± 0.0582	$\mathbf{0.6974 \pm 0.0668}$
USPS	2	0.6028 ± 0.0563	0.6937 ± 0.0540	0.7231 ± 0.0645	0.7289 ± 0.0457	0.7339 ± 0.0630	0.6405 ± 0.0481	$\mathbf{0.7684 \pm 0.0654}$
	3	0.6584 ± 0.0286	0.7369 ± 0.0338	0.7682 ± 0.0324	0.7696 ± 0.0339	0.7697 ± 0.0301	0.7019 ± 0.0304	$\mathbf{0.8045 \pm 0.0385}$
	# LS	PCA	GPCA+EAM	GPCA+initial PCs	GPCA+E2CP	GPCA+PCPSNMF	GPCA+SSLRR	GPCA+Proposed
	1	0.4516 ± 0.0210	0.4891 ± 0.0358	0.5031 ± 0.0372	0.5105 ± 0.0372	0.5308 ± 0.0308	0.4452 ± 0.0265	$\mathbf{0.6034 \pm 0.0502}$
UMIST	2	0.6118 ± 0.0370	0.6418 ± 0.0337	0.6512 ± 0.0338	0.6572 ± 0.0340	0.6752 ± 0.0374	0.6013 ± 0.0312	$\mathbf{0.7224 \pm 0.0407}$
	3	0.7151 ± 0.0406	0.7366 ± 0.0446	0.7437 ± 0.0433	0.7475 ± 0.0428	0.7647 ± 0.0376	0.7076 ± 0.0407	$\mathbf{0.7919 \pm 0.0497}$
	# LS	PCA	GPCA+EAM	GPCA+initial PCs	GPCA+E2CP	GPCA+PCPSNMF	GPCA+SSLRR	GPCA+Proposed
	1	0.4630 ± 0.0841	0.5228 ± 0.1003	0.5983 ± 0.0996	0.6155 ± 0.1014	0.5574 ± 0.0861	0.5349 ± 0.0914	$\mathbf{0.6362 \pm 0.1150}$
COTTON	2	0.5766 ± 0.0730	0.5491 ± 0.0982	0.6574 ± 0.0741	0.6644 ± 0.0735	0.6367 ± 0.0791	0.6329 ± 0.0671	$\mathbf{0.6945 \pm 0.0814}$
	3	0.5981 ± 0.0644	0.5578 ± 0.1016	0.6863 ± 0.0565	0.6954 ± 0.0567	0.6603 ± 0.0669	0.6836 ± 0.0590	$\mathbf{0.7097 \pm 0.0641}$
	# LS	PCA	GPCA+EAM	GPCA+initial PCs	GPCA+E2CP	GPCA+PCPSNMF	GPCA+SSLRR	GPCA+Proposed
	1	0.3881 ± 0.0409	0.4493 ± 0.0369	0.4494 ± 0.0366	0.4518 ± 0.0370	0.4277 ± 0.0379	0.4019 ± 0.0305	$\mathbf{0.4825 \pm 0.0547}$
LIBRAS	2	0.5033 ± 0.0495	0.5488 ± 0.0397	0.5557 ± 0.0440	0.5584 ± 0.0448	0.5409 ± 0.0446	0.5208 ± 0.0445	$\mathbf{0.5919 \pm 0.0492}$
	3	0.5894 ± 0.0315	0.5999 ± 0.0384	0.6141 ± 0.0302	0.6156 ± 0.0299	0.6223 ± 0.0294	0.6042 ± 0.0315	$\mathbf{0.6576 \pm 0.0377}$
	# LS	PCA	GPCA+EAM	GPCA+initial PCs	GPCA+E2CP	GPCA+PCPSNMF	GPCA+SSLRR	GPCA+Proposed
	1	0.4375 ± 0.1070	0.4702 ± 0.1021	0.4795 ± 0.0823	0.5026 ± 0.0851	0.5204 ± 0.1055	0.4728 ± 0.0874	$\mathbf{0.5675 \pm 0.1382}$
USER	2	0.5427 ± 0.0580	0.5457 ± 0.0684	0.5813 ± 0.0707	0.5918 ± 0.0779	0.6155 ± 0.0846	0.5069 ± 0.0695	$\mathbf{0.6266 \pm 0.0933}$
	3	0.5783 ± 0.0772	0.5899 ± 0.0631	0.6338 ± 0.0649	0.6514 ± 0.0603	0.6874 ± 0.0523	0.5592 ± 0.0524	$\mathbf{0.7038 \pm 0.0572}$

The highest value is highlighted by **bold**, and the second highest value is underlined. # LS denotes the number of labeled samples per class. The results of PCA were obtained by setting η of Eq. (23) to 0 .

[37] Y. Jia, S. Kwong, J. Hou, and W. Wu, "Convex constrained clustering with graph-laplacian pca," in *Proc. ICME*, 2018, pp. 1–6.

[38] N. Shahid, V. Kalofolias, X. Bresson, M. Bronstein, and P. Vandergheynst, "Robust principal component analysis on graphs," in *Proc. ICCV*, 2015, pp. 2812–2820.

[39] X. Dong, D. Thanou, M. Rabbat, and P. Frossard, "Learning graphs from data: A signal representation perspective," *IEEE Signal Process. Mag.*, vol. 36, no. 3, pp. 44–63, 2019.

[40] Y. Peng, X. Zhai, Y. Zhao, and X. Huang, "Semi-supervised cross-media feature learning with unified patch graph regularization," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 3, pp. 583–596, 2016.

[41] J. Zhang, Y. Peng, and M. Yuan, "Sch-gan: Semi-supervised cross-modal hashing by generative adversarial network," *IEEE Trans. Cybern.*, vol. 50, no. 2, pp. 489–502, 2020.

[42] J. Zhang and Y. Peng, "Ssdh: Semi-supervised deep hashing for large scale image retrieval," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 1, pp. 212–225, 2019.

[43] X. Zhang, Q. Liu, D. Wang, L. Zhao, N. Gu, and S. Maybank, "Self-taught semisupervised dictionary learning with nonnegative constraint," *IEEE Trans. Ind. Informat.*, vol. 16, no. 1, pp. 532–543, 2020.

[44] E. J. Candès, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?" *Journal of the ACM (JACM)*, vol. 58, no. 3, pp. 1–37, 2011.

[45] J. Wright, A. Ganesh, S. Rao, Y. Peng, and Y. Ma, "Robust principal component analysis: Exact recovery of corrupted low-rank matrices via convex optimization," in *Proc. NIPS*, 2009, pp. 2080–2088.

[46] Z. Lin, M. Chen, and Y. Ma, "The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices," *arXiv preprint arXiv:1009.5055*, 2010.

[47] J. Chen, H. Mao, Y. Sang, and Z. Yi, "Subspace clustering using a symmetric low-rank representation," *Knowledge-Based Systems*, vol. 127, pp. 46–57, 2017.

[48] L. Zhuang, Z. Zhou, S. Gao, J. Yin, Z. Lin, and Y. Ma, "Label information guided graph construction for semi-supervised learning," *IEEE Trans. Image Process.*, vol. 26, no. 9, pp. 4182–4192, 2017.

[49] K. Somandepalli and S. Narayanan, "Reinforcing self-expressive representation with constraint propagation for face clustering in movies," in *Proc. ICASSP*. IEEE, 2019, pp. 4065–4069.

[50] K. Tang, R. Liu, Z. Su, and J. Zhang, "Structure-constrained low-rank representation," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 12, pp. 2167–2179, 2014.

[51] C. Lu, J. Feng, Z. Lin, T. Mei, and S. Yan, "Subspace clustering by block diagonal representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 2, pp. 487–501, 2018.

Yuheng Jia received the B.S. degree in automation and the M.S. degree in control theory and engineering from Zhengzhou University, Zhengzhou, China, in 2012 and 2015, respectively, and the Ph.D. degree in computer science from the City University of Hong Kong, SAR, China, in 2019.

He is currently an associate professor with the School of Computer Science and Engineering, Southeast University, China. His research interests include machine learning, Bayesian method, spectral clustering and low-rank modeling.

Hui Liu received the B.S. degree from Central South University, Changsha, China and M.S. degree from Nanyang Technological University, Singapore. From 2014 to 2017, she was a research associate in Maritime Institute of Nanyang Technological University. She is currently pursuing the Ph.D. degree in department of computer science from City University of HongKong, SAR, China.

Junhui Hou (M'16-SM'20) received the B.Eng. degree in information engineering (Talented Students Program) from the South China University of Technology, Guangzhou, China, in 2009, the M.Eng. degree in signal and information processing from Northwestern Polytechnical University, Xian, China, in 2012, and the Ph.D. degree in electrical and electronic engineering from the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore, in 2016. He has been an Assistant Professor with the Department of Computer Science, City University of Hong Kong, since 2017. His research interests fall into the general areas of visual computing, such as image/video/3D geometry data representation, processing and analysis, semi/un-supervised data modeling, and data compression and adaptive transmission.

Dr. Hou was the recipient of several prestigious awards, including the Chinese Government Award for Outstanding Students Study Abroad from China Scholarship Council in 2015, and the Early Career Award (3/381) from the Hong Kong Research Grants Council in 2018. He is a member of Multimedia Systems & Applications Technical Committee (MSA-TC), IEEE CAS. He is currently serving as an Associate Editor for IEEE Transactions on Circuits and Systems for Video Technology, The Visual Computer, and Signal Processing: Image Communication, and the Guest Editor for the IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing. He also served an Area Chair of ACM MM 2019 and 2020, IEEE ICME 2020, and WACV 2021. He is a senior member of IEEE.

Sam Kwong (SM'04-F'14) received the B.Sc. degree in electrical engineering from the State University of New York, Buffalo, NY, USA, in 1983 and the M.Sc. degree in electrical engineering from the University of Waterloo, Waterloo, ON, Canada, in 1985, and the Ph.D. degree from the University of Hagen, North Rhine-Westphalia, Germany, in 1996. From 1985 to 1987, he was a Diagnostic Engineer with the Control Data Canada, Mississauga, ON, Canada. He later joined Bell Northern Research Canada, Ottawa, ON, Canada, as a Member of Scientific Staff, and the City University of Hong Kong (CityU), Hong Kong, as a Lecturer with the Department of Electronic Engineering in 1990. He is currently a Chair Professor with the Department of Computer Science, CityU. His research interests include video coding, pattern recognition, and evolutionary algorithms. He is currently the Vice-President of Conferences and Meetings with the IEEE Systems, Man and Cybernetics. He also serves as an Associate Editor of the IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION, the IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS, and the IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, and the Journal of Information Science.

Qingfu Zhang (M'01-SM'06-F'17) received the B.Sc. degree in mathematics from Shanxi University, China in 1984, the M.Sc. degree in applied mathematics and the Ph.D. degree in information engineering from Xidian University, China, in 1991 and 1994, respectively. He is a Chair Professor of Computational Intelligence at the Department of Computer Science, City University of Hong Kong. His main research interests include evolutionary computation, optimization, neural networks, data analysis, and their applications.

Dr. Zhang is an Associate Editor of the IEEE Transactions on Evolutionary Computation and the IEEE Transactions on Cybernetics. He is a Web of Science highly cited researcher in Computer Science for five consecutive years since 2016.