IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received August 27, 2020, accepted September 12, 2020, date of publication September 24, 2020, date of current version October 6, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3025045

Discriminative Multiple Kernel Concept Factorization for Data Representation

LIN MU[1], **HAIYING ZHANG**[2], **LIANG DU**[2,3], **(Member, IEEE), JIE GUI**[1], **AIDAN LI**[1], **AND XI ZHANG**[1]

[1]Institute of Scientific and Technical Information of China, Beijing 100036, China
[2]School of Computer and Information Technology, Shanxi University, Taiyuan 030006, China
[3]Institute of Big Data Science and Industry, Shanxi University, Taiyuan 030006, China

Corresponding authors: Liang Du (csliangdu@gmail.com) and Jie Gui (guij@istic.ac.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61502289 and Grant 61806003, in part by the Shanxi Province Key Research and Development program under Grant 201803D31199, in part by the Natural Science Foundation of Shanxi Province, China, under Grant 201801D221163, and in part by the Scientific and Technologial Innovation Programs of Higher Education Institutions in Shanxi under Grant STIP 2016101.

ABSTRACT Concept Factorization (CF) improves Nonnegative matrix factorization (NMF), which can be only performed in the original data space, by conducting factorization within proper kernel space where the structure of data become much clear than the original data space. CF-based methods have been widely applied and yielded impressive results in optimal data representation and clustering tasks. However, CF methods still face with the problem of proper kernel function design or selection in practice. Most existing Multiple Kernel Clustering (MKC) algorithms do not sufficiently consider the intrinsic neighborhood structure of base kernels. In this paper, we propose a novel Discriminative Multiple Kernel Concept Factorization method for data representation and clustering. We first extend the original kernel concept factorization with the integration of multiple kernel clustering framework to alleviate the problem of kernel selection. For each base kernel, we extract the local discriminant structure of data via the local discriminant models with global integration. Moreover, we further linearly combine all these kernel-level local discriminant models to obtain an integrated consensus characterization of the intrinsic structure across base kernels. In this way, it is expected that our method can achieve better results by more compact data reconstruction and more faithful local structure preserving. An iterative algorithm with convergence guarantee is also developed to find the optimal solution. Extensive experiments on benchmark datasets further show that the proposed method outperforms many state-of-the-art algorithms.

INDEX TERMS Concept factorization, multiple kernel clustering, local discriminant regularization, data representation.

I. INTRODUCTION

Data representation is a fundamental topic in machine learning, pattern recognition and data mining. Previous studies have shown that the performance of many learning tasks, such as clustering and classification, can be largely improved with more faithful and compact representation. Matrix factorization techniques have been widely used to obtain low dimensional representations. Several methods have also been developed such as Singular Value Decomposition (SVD), Principle Component Analysis (PCA), Non-negative Matrix Factorization (NMF) [1]–[3]. By keeping the two latent factors be non-negative, NMF leads to the well known part-based representation, which not only provides better performance in face recognition and document clustering but also enables better semantic interpretation. However, NMF only works in the original non-negative space. As one of the most important extension of NMF, Concept Factorization (CF) [4] inherits the merit of non-negative representation and conducts factorization in any data space such as the Reproducing Kernel Hilbert space (RKHS). It has also been pointed out that the structure of data within proper kernel space may become much clear than in the original feature space [5]. Therefore, concept factorization can discover more meaningful concepts

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

and lead to better learning performance compared with NMF [4]. That is also the primary advantage of CF over NMF.

In recent years, various concept factorization methods have been further developed. Graph regularized concept factorization methods [6]–[11] extracts the concepts of data which are consistent with the manifold geometry by exploiting the graph Laplacian as additional regularization terms for smoothness. Sparse concepts can also be obtained with the locality-constraints [12], [13]. Semi-supervised concept factorization methods [14]–[18] have also been proposed by using the available supervised information to guild the factorization process. Most recently, multi-view concept factorization methods [19], [20] have also been proposed to handle the complementary information from multiple views. Most of existing works on CF only handle data with single kernel. However, CFs methods still face with the problem of the design or selection of proper kernel function in practice. By leveraging a predefined set of candidates kernels from different functions or views, the Multiple Kernel Clustering (MKC) methods are with great potential to alleviate the effort for kernel designing or integrating complementary information [21]. It is natural to extend existing single kernel clustering methods into multiple kernel scenario. The typical methods include K-means based [21]–[27], self-organizing map (SOM) [28], maximum margin clustering based [29]–[31], local learning-based [32], spectral clustering based [33]–[40] and subspace clustering based [41]–[45] algorithms. Compared with the single kernel counterpart, MKC should take special effort to handle the additional data problems such as noisy and incomplete kernels [24], [27], [44], [46]–[51]. Moreover, only a few efforts [45], [52], [53] have been taken to incorporate the local geometric structure of data for MKC. In addition, It has been shown that the disciminant information is also important for the learning tasks [54], [55].

To alleviate the effort for kernel designing and make full use of complementary information, it is imperative to learn an appropriate kernel efficiently to make the performance of concept factorization more stable or even better across multiple different kernels. In this paper, we present the novel Discriminative Multiple Kernel Concept Factorization (DMKCF) for data representation. To achieve this, we first combine multiple base kernels with linear weights to approximate the unknown proper kernel matrix. We then replace the data matrix in kernelized CF with the combined kernel matrix and get the multiple kernel concept factorization (MKCF). Specifically speaking, for each data point in each base kernel, we construct a local clique comprising this data point and its neighboring data points identified by the base kernel. We use a local discriminant model for each local clique from each base kernel to evaluate the representation performance of samples within the local clique. We then integrate the local models of all the local cliques from all the base kernels into a global model to approximate the underlying local and discriminant structure of data. We incorporate the induced Multiple Kernel Local Discriminative regularization on orthogonal non-negative low-dimensional representation into the above MKCF learning procedure. We then derive the corresponding multiplicative update rules to reduce the objective function monotonically and obtain the unique solution for the proposed DMKCF model. Extensive experimental results on benchmark data sets well demonstrate the effectiveness of the proposed method over state-of-the-art multiple kernel learning algorithms.

It is worthwhile to highlight several properties of the proposed DMKCF method.

- The proposed method avoids the problem of kernel selection in concept factorization by integrating multiple candidate kernels under the framework of multiple kernel clustering.
- The proposed method globally integrates the local discriminant models from all the local cliques and all the base kernels to approximate the underlying local and discriminant structure of multiple kernels, which is further used to regulate the procedure of concept factorization. The proposed method extracts the concepts with respect to the local structure and thus data samples associated with the same concept can be well clustered.
- We propose an effective iterative strategy with multiplicative updating rules to obtain the optimal unique solution, and provide the proof of rigorous convergence and correctness analysis of our method.

The rest of the paper is organized as follows. The preliminaries on non-negative matrix factorization and concept factorization are introduced in Section 2. Section 3 introduces the proposed Discriminative Multiple Kernel Concept Factorization method. The optimization algorithm is presented in Section 4. Extensive experimental results on clustering are presented in Section 5. Finally, we provide some concluding remarks and suggestions for future work in Section 6.

II. RELATED WORK

A. NON-NEGATIVE MATRIX FACTORIZATION

Given a data matrix $\mathbf{X} = [\mathbf{x}_1, \cdots, \mathbf{x}_n] \in \mathcal{R}^{d\times n}$, each column of $\mathbf{X}$ is a sample vector. By solving the following optimization problem, NMF [1], [2] aims to extract two non-negative matrices $\mathbf{W} \in \mathcal{R}^{d\times c}$ and $\mathbf{V} \in \mathcal{R}^{n\times c}$ whose product can well approximate the original matrix $\mathbf{X}$.

$$\min_{\mathbf{W},\mathbf{V}} \quad ||\mathbf{X} - \mathbf{W}\mathbf{V}^T||^2, \quad \text{s.t.} \quad \mathbf{W} \geq 0, \mathbf{V} \geq 0. \tag{1}$$

It can be seen that each data vector $\mathbf{x}_i$ is approximated by a linear combination of the columns of $\mathbf{W}$, weighted by the components of $\mathbf{V}$, i.e. $\mathbf{x}_i = \sum_j^c \mathbf{w}_j v_{ij}$. Thus, $\mathbf{W}$ can be regarded as a set of basis and $\mathbf{V}$ can be regarded as the new representation of each data point in the new basis $\mathbf{W}$.

B. CONCEPT FACTORIZATION

By replacing the basis vectors in NMF with the non-negative linear combination of the sample vectors, i.e., $\mathbf{w}_j = \sum_{j'=1}^n \mathbf{x}_{j'} u_{jj'}$, CF [4] performs factorization in linear space by solving:

$$\min_{\mathbf{U},\mathbf{V}} \quad ||\mathbf{X} - \mathbf{X}\mathbf{U}\mathbf{V}^T||^2, \quad \text{s.t.} \quad \mathbf{U} \geq 0, \mathbf{V} \geq 0. \tag{2}$$

Besides, it can be easily verified that the kernelized concept factorization can be written as

$$\begin{aligned} &\min_{\mathbf{U},\mathbf{V}} \ \mathrm{tr}(\mathbf{K}) - 2\mathrm{tr}(\mathbf{V}^T\mathbf{K}\mathbf{U}) + \mathrm{tr}(\mathbf{U}^T\mathbf{K}\mathbf{U}\mathbf{V}^T\mathbf{V}) \\ &\text{s.t. } \mathbf{U} \geq 0, \mathbf{V} \geq 0, \end{aligned} \tag{3}$$

where $\mathbf{K} \in \mathcal{R}^{n\times n}$ is the kernel matrix, and $\mathbf{K} = \mathbf{X}\mathbf{X}^T$ for the linear case. It has been shown that the optimal value of $\mathbf{U}$ and $\mathbf{V}$ in the kernel concept factorization model can be obtained by the following multiplicative update rules:

$$\mathbf{U}_{ij} = \mathbf{U}_{ij}\frac{(\mathbf{K}\mathbf{V})_{ij}}{(\mathbf{K}\mathbf{U}\mathbf{V}^T\mathbf{V})_{ij}} \tag{4}$$

$$\mathbf{V}_{ij} = \mathbf{V}_{ij}\frac{(\mathbf{K}\mathbf{U})_{ij}}{(\mathbf{V}\mathbf{U}^T\mathbf{K}\mathbf{U})_{ij}} \tag{5}$$

For the kernel matrix with negative entries, the multiplicative update rules become

$$\mathbf{U}_{ij} = \mathbf{U}_{ij}\frac{(\mathbf{K}\mathbf{V})_{ij} + \sqrt{(\mathbf{K}\mathbf{V})_{ij}^2 + 4\mathbf{P}_{ij}^+\mathbf{P}_{ij}^-}}{2\mathbf{P}_{ij}^+} \tag{6}$$

$$\mathbf{V}_{ij} = \mathbf{V}_{ij}\frac{(\mathbf{K}\mathbf{U})_{ij} + \sqrt{(\mathbf{K}\mathbf{U})_{ij}^2 + 4\mathbf{Q}_{ij}^+\mathbf{Q}_{ij}^-}}{2\mathbf{Q}_{ij}^+} \tag{7}$$

where $\mathbf{K}^+ = (|\mathbf{K}| + \mathbf{K})/2$, $\mathbf{K}^- = (|\mathbf{K}| - \mathbf{K})/2$, and we further denote $\mathbf{P}^+ = \mathbf{K}^+\mathbf{U}\mathbf{V}^T\mathbf{V}$, $\mathbf{P}^- = \mathbf{K}^-\mathbf{U}\mathbf{V}^T\mathbf{V}$, $\mathbf{Q}^+ = \mathbf{V}\mathbf{U}^T\mathbf{K}^+\mathbf{U}$, $\mathbf{Q}^- = \mathbf{V}\mathbf{U}^T\mathbf{K}^-\mathbf{U}$.

III. DISCRIMINATIVE MULTIPLE KERNEL CONCEPT FACTORIZATION

In this section, we extend kernel concept factorization to automatically learn an appropriate kernel from the convex linear combination of several pre-computed kernel matrices within the multiple kernel learning framework. We also present the multiple kernel local discriminative regularization to capture the local structure of multiple base kernels. Then, we have the Discriminative Multiple Kernel Concept Factorization method for data representation.

A. MULTIPLE KERNEL CONCEPT FACTORIZATION

Suppose there are altogether m different kernel functions $\{\mathcal{K}^i\}_{i=1}^m$ available for the clustering task in hand. Accordingly, there are m different associated feature spaces denoted as $\{\mathcal{H}\}_i^m$. To combine these kernels and also ensure that the resulted kernel still satisfies Mercer condition, we construct an augmented Hilbert space $\tilde{\mathcal{H}} = \oplus_{i=1}^m \mathcal{H}^i$ by concatenating all feature spaces $\phi_{\boldsymbol{\mu}}(\mathbf{x}) = [\mu_1\phi_1(\mathbf{x}); \mu_2\phi_2(\mathbf{x}); \ldots; \mu_m\phi_m(\mathbf{x})]^T$ with different weight μ_i ($\mu_i \geq 0$), or equivalently the importance factor for kernel function $\mathcal{K}^i$. It can be verified that clustering in feature space $\tilde{\mathcal{H}}$ is equivalent to employing the following combined kernel function [32]

$$\tilde{\mathcal{K}}(\mathbf{x}, \mathbf{x}') = \sum_{i=1}^m \mu_i^2\mathcal{K}^i(\mathbf{x}, \mathbf{x}'). \tag{8}$$

It is known that the convex combination, with $\boldsymbol{\mu}$ ($\mu_i \geq 0$), of the positive semi-definite kernel matrices $\{\mathbf{K}^i\}_{i=1}^m$ is still a positive semi-definite kernel matrix. By replacing the single kernel in Eq. (3) with the combined kernel, we present the multiple kernel concept factorization by solving:

$$\begin{aligned} &\min_{\mathbf{U},\mathbf{V},\boldsymbol{\mu}} \ \mathrm{tr}(\mathbf{K}_{\boldsymbol{\mu}}) - 2\mathrm{tr}(\mathbf{V}^T\mathbf{K}_{\boldsymbol{\mu}}\mathbf{U}) + \mathrm{tr}(\mathbf{U}^T\mathbf{K}_{\boldsymbol{\mu}}\mathbf{U}\mathbf{V}^T\mathbf{V}) \\ &\text{s.t. } \mathbf{U} \geq 0, \quad \mathbf{V} \geq 0, \ \boldsymbol{\mu} \geq 0, \ \sum_{i=1}^m \mu_i = 1, \end{aligned} \tag{9}$$

where $(\mathbf{K}^i)_{ab} = \mathcal{K}^i(\mathbf{x}_a, \mathbf{x}_b)$ is the kernel Gram matrix of the i-th predefined kernel function over the unlabeled dataset $\mathbf{X}$, and $(\mathbf{K}_{\boldsymbol{\mu}})_{ab} = \mathcal{K}_{\boldsymbol{\mu}}(\mathbf{x}_a, \mathbf{x}_b)$ is the kernel matrix of the consensus kernel function $\mathcal{K}_{\boldsymbol{\mu}}(\cdot, \cdot)$.

B. LOCALIZED DISCRIMINATIVE MULTIPLE KERNEL REGULARIZATION

In this subsection, we propose a new Local Discriminant Multiple Kernel regularization to utilize both manifold information and discriminant information for multiple kernel clustering. We extract a local clique, for each data point from each base kernel, comprising of this data point and its neighboring points. We build a local discriminant model such local clique for better data separation and representation. We integrate all the local discriminant models for each point and each base kernel and get the localized Discriminative Multiple Kernel regularization.

Given a centered data set consisting of n data points $\{\mathbf{x}_i\}_{i=1}^n \in \mathcal{R}^d$, the goal of clustering is to find a disjoint partitioning $\{\pi_j\}_{j=1}^c$ of the data where π_j is the j-th cluster. We define the cluster indicator matrix defined as $\mathbf{P} = [\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_c] = [\mathbf{P}_{ij}] \in \{0, 1\}^{n\times c}$. Specifically,

$$\mathbf{P}_{ij} = \begin{cases} 1 & \text{if } \mathbf{x}_i \in \pi_j, \\ 0 & \text{if } \mathbf{x}_i \notin \pi_j, \end{cases} \tag{10}$$

We then introduce the scaled cluster indicator matrix as $\mathbf{Y} = \mathbf{P}(\mathbf{P}^T\mathbf{P})^{-1}$, where $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_c] = [\mathbf{Y}_{ij}] \in \mathcal{R}^{n\times c}$. Specifically,

$$\mathbf{Y}_{ij} = \begin{cases} \dfrac{1}{|\pi_j|} & \text{if } \mathbf{x}_i \in \pi_j, \\ 0 & \text{if } \mathbf{x}_i \notin \pi_j, \end{cases} \tag{11}$$

where $|\pi_j|$ is the sample size of the j-th cluster π_j. Denote $\mathbf{g}_j = \sum_{\mathbf{x}\in\pi_j} \frac{\mathbf{x}}{|\pi_j|}$ as the mean of the j-th cluster. Therefore, we can define the within-cluster scatter, between-cluster scatter, and total scatter matrices as $\mathbf{S}_w = \sum_{j=1}^c \sum_{\mathbf{x}\in\pi_j}(\mathbf{x} - \mathbf{g}_j)(\mathbf{x} - \mathbf{g}_j)^T$, $\mathbf{S}_b = \mathbf{X}\mathbf{Y}\mathbf{Y}^T\mathbf{X}^T$, $\mathbf{S}_t = \mathbf{X}\mathbf{X}^T$. It has been pointed out that $\mathrm{tr}(\mathbf{S}_w)$ captures the intra-cluster distance, and $\mathrm{tr}(\mathbf{S}_b)$ captures the inter-cluster distance. And we have $\mathbf{S}_t = \mathbf{S}_w + \mathbf{S}_b$. For high-dimensional data, a reliable estimation of the total scatter (covariance) matrix can be obtained by adding additional regularization and we have $\widetilde{\mathbf{S}}_t = \mathbf{X}\mathbf{X}^T + \gamma \mathbf{I}_d$, where $\mathbf{I}_d$ is the identity matrix of size d and $\gamma > 0$ is a regularization parameter.

Intuitively, to better cluster the data, the distance between data from different clusters should be as large as possible while the distance between data from the same cluster should be as small as possible [56]. Inspired by Fisher criterion and the discriminant clustering [57], the optimal scaled cluster assignment matrix $\mathbf{Y}^*$ can be obtained by minimize the following linear discriminant model

$$\mathbf{Y}^* = \arg\min_{\mathbf{Y}} \operatorname{tr}(\mathbf{Y}^T\mathbf{Y} - \mathbf{Y}^T\mathbf{X}^T(\mathbf{X}\mathbf{X}^T + \gamma\mathbf{I})^{-1}\mathbf{X}\mathbf{Y}).$$

By using the Woodbury identity, the above problem can be equivalently reformulated as [57],

$$\mathbf{Y}^* = \arg\min_{\mathbf{Y}} \quad \operatorname{tr}\left(\mathbf{Y}^T(\mathbf{HKH} + \gamma\mathbf{I}_n)^{-1}\mathbf{Y}\right), \tag{12}$$

where $\mathbf{H} = \mathbf{I} - \frac{1}{n}\mathbf{1}_n\mathbf{1}_n^T$ is the centering matrix and the equation $\mathbf{H} = \mathbf{H}^T = \mathbf{HH}$ holds, the kernel matrix $\mathbf{K} = \mathbf{X}^T\mathbf{X}$ for linear kernel function.

Given the p-th kernel candidate matrix, we consider a local clique $\mathcal{N}_i^p$ comprising τ data points including the i-th sample and its $\tau - 1$ nearest neighbors determined by the kernel matrix $\mathbf{K}^p$, and employ a local kernel discriminant model to evaluate the clustering results for the data points. Let denote $\mathbf{Y}^p$ be the scaled partition matrix determined by kernel matrix $\mathbf{K}^p$ and $\mathbf{Y}_{(i)}^p \in \mathcal{R}^{\tau\times c}$ be the local scaled cluster assignment matrix for the i-th clique with $\mathbf{K}^p$. The localized discriminant model can be written as

$$\begin{aligned}\mathbf{Y}_{(i)}^{p*} &= \arg\min_{\mathbf{Y}_{(i)}^p} \operatorname{tr}\left((\mathbf{Y}_{(i)}^p)^T(\mathbf{H}_\tau\mathbf{K}_i^p\mathbf{H}_\tau + \gamma\mathbf{I}_\tau)^{-1}\mathbf{Y}_{(i)}^p\right)\\ &= \arg\min_{\mathbf{Y}_{(i)}^p} \operatorname{tr}\left((\mathbf{Y}_{(i)}^p)^T\mathbf{L}_i^p\mathbf{Y}_{(i)}^p\right),\end{aligned} \tag{13}$$

where $\mathbf{L}_i^p = (\mathbf{H}_\tau\mathbf{K}_i^p\mathbf{H}_\tau + \gamma\mathbf{I}_\tau)^{-1}$ is the local Laplacian matrix. It can be seen that a larger local discriminant score indicates that the samples in the local clique from different clusters are better separated.

Moreover, we denote $\mathbf{L}^p$ as the aggregated Laplacian matrix induced from $\mathbf{K}^p$, which can be obtained by

$$\mathbf{L}^p = \sum_{i=1}^{n} \mathbf{S}_{(i)}^p\mathbf{L}_i^p(\mathbf{S}_{(i)}^p)^T, \tag{14}$$

where $\mathbf{S}_{(i)}^p \in \mathcal{R}^{n\times k}$ is the local selection matrix with its element $(\mathbf{S}_{(i)}^p)_{jj'} = 1$ if the j-th sample is the j'-th neighbor of the i-th sample determined by $\mathbf{K}^p$; $(\mathbf{S}_{(i)}^p)_{jj'} = 0$, otherwise.

The overall clustering results can then be obtained by globally optimizing the local discriminant models of all the local cliques.

$$\begin{aligned}\mathbf{Y}^{p*} &= \arg\min_{\mathbf{Y}^p} \sum_{i=1}^{n} \operatorname{tr}\left((\mathbf{Y}_{(i)}^p)^T(\mathbf{H}_\tau\mathbf{K}_i^p\mathbf{H}_\tau + \gamma\mathbf{I}_\tau)^{-1}\mathbf{Y}_{(i)}^p\right)\\ &= \arg\min_{\mathbf{Y}_p} \sum_{i=1}^{n} \operatorname{tr}\left((\mathbf{Y}_{(i)}^p)^T\mathbf{L}_i^p\mathbf{Y}_{(i)}^p\right)\\ &= \arg\min_{\mathbf{Y}^p} \sum_{i=1}^{n} \operatorname{tr}\left(\mathbf{Y}\mathbf{S}_{(i)}^p\mathbf{L}_i^p(\mathbf{S}_{(i)}^p)^T\mathbf{Y}\right)\\ &= \arg\min_{\mathbf{Y}^p} \operatorname{tr}((\mathbf{Y}^p)^T\sum_{i=1}^{n}\mathbf{S}_{(i)}^p\mathbf{L}_i^p(\mathbf{S}_{(i)}^p)^T\mathbf{Y})\\ &= \arg\min_{\mathbf{Y}^p} \operatorname{tr}((\mathbf{Y}^p)^T\mathbf{L}^p\mathbf{Y}^p).\end{aligned} \tag{15}$$

Considering the fact that different kernels have different local neighborhoods, it is desired to aggregated these aggregated local discriminant models. Inspired by the linear combination of multiple kernel learning, we also introduce the multiple kernel aggregated Laplacian by the linear combination of these kernel-specific Laplacian matrices

$$\mathbf{L}_{\boldsymbol{\mu}} = \sum_{p=1}^{m} \mu_p^2\mathbf{L}^p. \tag{16}$$

It is believed that the above Laplacian matrix well capture the local information and discriminant information in multiple kernels. To further improve the performance for the task of clustering and the learning of concept factorization, we further replace the unknown scaled partition matrix with the non-negative low-dimensional representation and propose the novel Local Discriminative Multiple Kernel Regularization, which can be formulated as

$$\begin{aligned}&\min_{\mathbf{V}} \ \operatorname{tr}(\mathbf{V}^T\mathbf{L}_{\boldsymbol{\mu}}\mathbf{V})\\ &\text{s.t. } \mathbf{V}^T\mathbf{V} = \mathbf{I}, \mathbf{V} \geq 0.\end{aligned} \tag{17}$$

To efficiently address the constraint $\mathbf{V}^T\mathbf{V} = \mathbf{I}$, we relax the equation condition by integrating a penalty term into optimization problem and get

$$\begin{aligned}&\min_{\mathbf{V}} \ \operatorname{tr}(\mathbf{V}^T\mathbf{L}_{\boldsymbol{\mu}}\mathbf{V}) + \xi||\mathbf{V}^T\mathbf{V} - \mathbf{I}||^2\\ &\text{s.t. } \mathbf{V} \geq 0,\end{aligned} \tag{18}$$

where ξ the a regularization parameter.

C. LOCALIZED DISCRIMINATIVE MULTIPLE KERNEL CONCEPT FACTORIZATION

Based on the multiple kernel concept factorization in Eq. (9) and the localized discriminative regularization in Eq. (18), we propose the novel Discriminative Multiple Kernel Concept Factorization (DMKCF) method for data representation and clustering, which can be formulated as follows.

$$\begin{aligned}\min_{\mathbf{U},\mathbf{V},\boldsymbol{\mu}} \ &\operatorname{tr}(\mathbf{K}_{\boldsymbol{\mu}}) - 2\operatorname{tr}(\mathbf{V}^T\mathbf{K}_{\boldsymbol{\mu}}\mathbf{U}) + \operatorname{tr}(\mathbf{U}^T\mathbf{K}_{\boldsymbol{\mu}}\mathbf{U}\mathbf{V}^T\mathbf{V})\\ &+ \lambda\operatorname{tr}(\mathbf{V}^T\mathbf{L}_{\boldsymbol{\mu}}\mathbf{V}) + \xi||\mathbf{V}^T\mathbf{V} - \mathbf{I}||^2\\ \text{s.t. } &\mathbf{U} \geq 0, \mathbf{V} \geq 0, \boldsymbol{\mu} \geq 0, \sum_{i=1}^{m}\mu_i = 1.\end{aligned} \tag{19}$$

The objection function in Eq. (19) contains five terms, where the first three terms are the kernel concept factorization, the fourth term is the local discriminative multiple kernel regularization and the last term is the orthogonal constraint for unique solution.

It can be seen that the consensus kernel $\mathbf{K}_{\boldsymbol{\mu}}$ is generated from the linear combination of base kernels. Instead of using the consensus kernel $\mathbf{K}_{\boldsymbol{\mu}}$ to extract the local structure, we also use each base kernel to capture the kernel-level local discriminant structure, where the discrete neighborhood structure of base kernel will not be changed during the learning procedure. Finally, the consensus local structure $\mathbf{L}_{\boldsymbol{\mu}}$ is also generated from the linear combination of base graph Laplacian $\{\mathbf{L}^i\}_{i=1}^m$. As a result, the integrated graph Laplacian $\mathbf{L}_{\boldsymbol{\mu}}$ will not be affected by the discrete neighborhood structure change of $\mathbf{K}_{\boldsymbol{\mu}}$.

IV. OPTIMIZATION

Because the optimization problem in Eq. (19) comprises three different variables, it is hard to derive its closed solution directly. Thus we derive an alternative iterative algorithm to solve the problem, which converts the problem with a couple of variables $(\mathbf{U}, \mathbf{V}, \boldsymbol{\mu})$ into a series of sub problems where only one variable is involved. The convergence and complexity analysis are further presented.

A. UPDATE U

When other variables are fixed, the rest optimization problem with respect to the variable $\mathbf{U}$ can be formulated as follows

$$\min_{\mathbf{U}} \ \mathrm{tr}(\mathbf{U}^T\mathbf{K}_{\boldsymbol{\mu}}\mathbf{U}\mathbf{V}^T\mathbf{V}) - 2\mathrm{tr}(\mathbf{V}^T\mathbf{K}_{\boldsymbol{\mu}}\mathbf{U})$$
$$\text{s.t. } \mathbf{U} \geq 0. \tag{20}$$

It can be seen that Eq. (20) is similar with Eq. (3). Therefore, Eq. (20) can be updated by the following multiplicative update rule for non-negative $\mathbf{K}_{\boldsymbol{\mu}}$

$$\mathbf{U}_{ij} = \mathbf{U}_{ij}\frac{(\mathbf{K}_{\boldsymbol{\mu}}\mathbf{V})_{ij}}{(\mathbf{K}_{\boldsymbol{\mu}}\mathbf{U}\mathbf{V}^T\mathbf{V})_{ij}}. \tag{21}$$

For the kernel matrix $\mathbf{K}_{\boldsymbol{\mu}}$ with negative entries, the multiplicative update rules become

$$\mathbf{U}_{ij} = \mathbf{U}_{ij}\frac{(\mathbf{K}_{\boldsymbol{\mu}}\mathbf{V})_{ij} + \sqrt{(\mathbf{K}_{\boldsymbol{\mu}}\mathbf{V})_{ij}^2 + 4\mathbf{B}_{ij}^+\mathbf{B}_{ij}^-}}{2\mathbf{B}_{ij}^+}, \tag{22}$$

where $\mathbf{K}_{\boldsymbol{\mu}}^+ = (|\mathbf{K}_{\boldsymbol{\mu}}| + \mathbf{K}_{\boldsymbol{\mu}})/2$, $\mathbf{K}_{\boldsymbol{\mu}}^- = (|\mathbf{K}_{\boldsymbol{\mu}}| - \mathbf{K}_{\boldsymbol{\mu}})/2$, and we further denote $\mathbf{B}^+ = \mathbf{K}_{\boldsymbol{\mu}}^+\mathbf{U}\mathbf{V}^T\mathbf{V}$, $\mathbf{B}^- = \mathbf{K}_{\boldsymbol{\mu}}^-\mathbf{U}\mathbf{V}^T\mathbf{V}$.

B. UPDATE μ

The optimization problem with respect to the variable $\boldsymbol{\mu}$ can be formulated as follows

$$\min_{\boldsymbol{\mu}} \ \boldsymbol{\mu}^T\mathbf{A}\boldsymbol{\mu}$$
$$\text{s.t. } \boldsymbol{\mu} \geq 0, \sum_{i=1}^{m}\mu_i = 1, \tag{23}$$

where $\mathbf{A}$ is a diagonal matrix with its diagonal element $\mathbf{A}_{ii} = \mathrm{tr}(\mathbf{K}^i) - 2\mathrm{tr}(\mathbf{V}^T\mathbf{K}^i\mathbf{U}) + \mathrm{tr}(\mathbf{U}^T\mathbf{K}^i\mathbf{U}\mathbf{V}^T\mathbf{V}) + \lambda\mathrm{tr}(\mathbf{V}^T\mathbf{L}^i\mathbf{V})$. It can be seen that Eq. (23) is a quadratic programming problem with linear constraints which can be solved by existing off-the-shelf packages.

C. UPDATE V

When other variables are fixed, the rest optimization problem with respect to the variable $\mathbf{V}$ can be formulated as follows

$$\min_{\mathbf{V}} \ \mathrm{tr}(\mathbf{V}\mathbf{U}^T\mathbf{K}_{\boldsymbol{\mu}}\mathbf{U}\mathbf{V}^T) - 2\mathrm{tr}(\mathbf{V}^T\mathbf{K}_{\boldsymbol{\mu}}\mathbf{U}) + \lambda\mathrm{tr}(\mathbf{V}^T\mathbf{L}_{\boldsymbol{\mu}}\mathbf{V})$$
$$+\xi\mathrm{tr}(\mathbf{V}^T\mathbf{V}\mathbf{V}^T\mathbf{V}) - 2\xi\mathrm{tr}(\mathbf{V}^T\mathbf{V})$$
$$\text{s.t. } \mathbf{V} \geq 0. \tag{24}$$

Eq. (24) is a quadratic programming problem with non-negative and orthogonal constraints. We can derive the similar multiplicative update rule for the positive only kernel matrix $\mathbf{K}_{\boldsymbol{\mu}}$ and Laplacian matrix $\mathbf{L}_{\boldsymbol{\mu}}$ as [6]

$$\mathbf{V}_{ij} = \mathbf{V}_{ij}\frac{(\mathbf{K}_{\boldsymbol{\mu}}\mathbf{U} + 2\xi\mathbf{V})_{ij}}{(\mathbf{V}\mathbf{U}^T\mathbf{K}_{\boldsymbol{\mu}}\mathbf{U} + 2\xi\mathbf{V}\mathbf{V}^T\mathbf{V})_{ij}}. \tag{25}$$

For the kernel matrix $\mathbf{K}_{\boldsymbol{\mu}}$ or Laplacian matrix $\mathbf{L}_{\boldsymbol{\mu}}$ with negative entries, we first introduce the following notations

$$\begin{aligned}
\mathbf{K}_{\boldsymbol{\mu}}^+ &= (|\mathbf{K}_{\boldsymbol{\mu}}| + \mathbf{K}_{\boldsymbol{\mu}})/2,\\
\mathbf{K}_{\boldsymbol{\mu}}^- &= (|\mathbf{K}_{\boldsymbol{\mu}}| - \mathbf{K}_{\boldsymbol{\mu}})/2,\\
\mathbf{L}_{\boldsymbol{\mu}}^+ &= (|\mathbf{L}_{\boldsymbol{\mu}}| + \mathbf{L}_{\boldsymbol{\mu}})/2,\\
\mathbf{L}_{\boldsymbol{\mu}}^- &= (|\mathbf{L}_{\boldsymbol{\mu}}| - \mathbf{L}_{\boldsymbol{\mu}})/2,\\
\mathbf{Q}^+ &= \mathbf{V}\mathbf{U}^T\mathbf{K}_{\boldsymbol{\mu}}^+\mathbf{U},\\
\mathbf{Q}^- &= \mathbf{V}\mathbf{U}^T\mathbf{K}_{\boldsymbol{\mu}}^-\mathbf{U},\\
\mathbf{T}_1 &= 2\xi(\mathbf{V}'\mathbf{V}'^T\mathbf{V}')_{ip},\\
\mathbf{T}_2 &= (\mathbf{V}'\mathbf{Q}^+ + \mathbf{K}_{\boldsymbol{\mu}}^-\mathbf{U} + \lambda\mathbf{L}_{\boldsymbol{\mu}}^+\mathbf{V}')_{ip}\mathbf{V}'^2_{ip},\\
\mathbf{T}_3 &= -(\mathbf{V}'\mathbf{Q}^- + (\mathbf{K}_{\boldsymbol{\mu}}^+\mathbf{U})_{ip} + \lambda\mathbf{L}_{\boldsymbol{\mu}}^-\mathbf{V}' + 2\xi\mathbf{V}')_{ip}\mathbf{V}'^4_{ip},
\end{aligned}$$

and we then get the following multiplicative rule to update $\mathbf{V}$

$$\mathbf{V}_{ip} = \sqrt{\frac{-\mathbf{T}_2 + \sqrt{\mathbf{T}_2^2 - 4\mathbf{T}_1\mathbf{T}_3}}{2\mathbf{T}_1}}. \tag{26}$$

In summary, we present the iterative updating algorithm of optimizing Eq. (19) in Algorithm 1.

Algorithm 1 The Algorithm to Solve Eq. (19)

```
Input: ${\mathbf{K}^i}_{i=1}^m$, ${\mathbf{L}^i}_{i=1}^m$, $\mathbf{U}$, $\mathbf{V}$, $\lambda$, $\xi$
1: Initialize $\boldsymbol{\mu}$;
2: repeat
3:     Update $\mathbf{K}_{\boldsymbol{\mu}}$ according to Eq. (8);
4:     Update $\mathbf{L}_{\boldsymbol{\mu}}$ according to Eq. (16);
5:     Update $\mathbf{U}$ according to Eq. (21) or Eq. (22);
6:     Update $\mathbf{V}$ according to Eq. (25) or Eq. (26);
7:     Update $\boldsymbol{\mu}$ by solving Eq. (23);
8: until Converges
Output: $\mathbf{U}$, $\mathbf{V}$, $\boldsymbol{\mu}$
```

D. CONVERGENCE

In this subsection, we will investigate the convergence of Algorithm 1. Here, we first use the auxiliary function approach [1] to show that the objective function in Eq. (24) with respect to $\mathbf{V}$ can be reduced monotonically.

TABLE 1. Description of the data sets.

Dataset	# instances	# features	# classes
USPS49	1673	256	2
PIE	2856	1024	68
COIL20	1440	1024	20
RELATHE	1427	4322	2
BBC	737	1000	5
K1b	2340	21839	6
Prostate	102	5966	2
ALLAML	72	7129	2
SMKCAN	187	19993	2
CLLSUB	111	11340	3

Definition IV-D1: [1] $\mathcal{J}(h, h')$ is an auxiliary function for $\mathcal{L}(h)$ if the following conditions holds

$$\mathcal{J}(h, h') \geq \mathcal{L}(h), \mathcal{J}(h, h) = \mathcal{L}(h). \tag{27}$$

Lemma IV-D2: [1] If $\mathcal{J}(h, h')$ is an auxiliary function for $\mathcal{J}(.)$, then $\mathcal{J}(.)$ is non-increasing under the update

$$h^{t+1} = \arg\min_h \mathcal{J}(h, h^t) \tag{28}$$

Proof: $\mathcal{L}(h^{t+1}) \leq \mathcal{J}(h^{t+1}, h^t) \leq \mathcal{J}(h^t, h^t) = \mathcal{L}(h^t)$. □

In the following, we will present 2 theorems, which guarantee the convergence of Algorithm 1.

Theorem IV-D3: Let

$$\begin{aligned}\mathcal{L}(\mathbf{V}) = {} & \mathrm{tr}(\mathbf{V}\mathbf{Q}^{+}\mathbf{V}^{T}) - \mathrm{tr}(\mathbf{V}\mathbf{Q}^{-}\mathbf{V}^{T}) - 2\mathrm{tr}(\mathbf{V}^{T}\mathbf{K}_{\boldsymbol{\mu}}^{+}\mathbf{U}) \\ & + 2\mathrm{tr}(\mathbf{V}^{T}\mathbf{K}_{\boldsymbol{\mu}}^{-}\mathbf{U}) + \lambda\mathrm{tr}(\mathbf{V}^{T}\mathbf{L}_{\boldsymbol{\mu}}^{+}\mathbf{V}) - \lambda\mathrm{tr}(\mathbf{V}^{T}\mathbf{L}_{\boldsymbol{\mu}}^{-}\mathbf{V}) \\ & + \xi\mathrm{tr}(\mathbf{V}^{T}\mathbf{V}\mathbf{V}^{T}\mathbf{V}) - 2\xi\mathrm{tr}(\mathbf{V}^{T}\mathbf{V})\end{aligned} \tag{29}$$

Then the following function

$$\begin{aligned}&\mathcal{J}(\mathbf{V}, \mathbf{V}') \\ &= \sum_{i=1}^{n}\sum_{p=1}^{k}\frac{(\mathbf{V}'\mathbf{Q}^{+})_{ip}\mathbf{V}_{ip}^{2}}{\mathbf{V}'_{ip}} \\ &\quad - \sum_{i=1}^{n}\sum_{p=1}^{k}\sum_{q=1}^{k}\mathbf{Q}_{pq}^{-}\mathbf{V}'_{ip}\mathbf{V}'_{iq}(1+\log\frac{\mathbf{V}_{ip}\mathbf{V}_{iq}}{\mathbf{V}'_{ip}\mathbf{V}'_{iq}}) \\ &\quad - 2\sum_{i=1}^{n}\sum_{p=1}^{k}(\mathbf{K}_{\boldsymbol{\mu}}^{+}\mathbf{U})_{ip}\mathbf{V}'_{ip}(1+\log\frac{\mathbf{V}_{ip}}{\mathbf{V}'_{ip}}) \\ &\quad + 2\sum_{i=1}^{n}\sum_{p=1}^{k}(\mathbf{K}_{\boldsymbol{\mu}}^{-}\mathbf{U})_{ip}\frac{\mathbf{V}_{ip}^{2}+\mathbf{V'}_{ip}^{2}}{2\mathbf{V}'_{ip}} \\ &\quad + \lambda\sum_{i=1}^{n}\sum_{p=1}^{k}\frac{(\mathbf{L}_{\boldsymbol{\mu}}^{+}\mathbf{V}')_{ip}\mathbf{V}_{ip}^{2}}{\mathbf{V}'_{ip}} \\ &\quad - \lambda\sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{p=1}^{k}(\mathbf{L}_{\boldsymbol{\mu}}^{-})_{ij}\mathbf{V}'_{ip}\mathbf{V}'_{jp}(1+\log\frac{\mathbf{V}_{ip}\mathbf{V}_{jp}}{\mathbf{V}'_{ip}\mathbf{V}'_{jp}}) \\ &\quad + \xi\sum_{i=1}^{n}\sum_{p=1}^{k}\frac{(\mathbf{V}'\mathbf{V}'^{T}\mathbf{V}')_{ip}\mathbf{V}_{ip}^{4}}{\mathbf{V'}_{ip}^{3}} \\ &\quad - \xi\sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{p=1}^{n}\mathbf{I}_{ij}\mathbf{V}'_{ip}\mathbf{V}'_{jp}(1+\log\frac{\mathbf{V}_{ip}\mathbf{V}_{jp}}{\mathbf{V}'_{ip}\mathbf{V}'_{jp}})\end{aligned} \tag{30}$$

is an auxiliary function for $\mathcal{L}(\mathbf{V})$. Furthermore, it is a convex function in $\mathbf{V}$ and its global minimum is

$$\mathbf{V}_{ip} = \sqrt{\frac{-\mathbf{T}_2 + \sqrt{\mathbf{T}_2^2 - 4\mathbf{T}_1\mathbf{T}_3}}{2\mathbf{T}_1}}. \tag{31}$$

Proof: See Appendix. □

Theorem IV-D4: The objective function in Eq. (24) will be non-increasing under the update rule in Eq. (26).

Proof: By Lemma IV-D2 and Theorem IV-D3, we can get $\mathcal{L}(\mathbf{V}^0) = \mathcal{J}(\mathbf{V}^0, \mathbf{V}^0) \geq \mathcal{J}(\mathbf{V}^1, \mathbf{V}^0) \geq \mathcal{L}(\mathbf{V}^1) \ldots$. So $\mathcal{L}(\mathbf{V})$ is non-creasing. □

The convergence of DMKCF under the update rules in Algorithm 1 can be summarized as follows. For fixed $\boldsymbol{\mu}^t$ and $\mathbf{U}^t$ in the t-iteration, the objective function of the rest sub-problem w.r.t $\mathbf{V}$ in Eq. (24) will be non-increasing under rules in Eq. (25) or Eq. (26). The proof can be found in Theorem IV-D4. For fixed $\boldsymbol{\mu}^t$ and $\mathbf{V}^t$ in the t-iteration, the sub problem w.r.t the variable $\mathbf{U}$ in Eq. (20) is exactly the same as the standard concept factorization model in Eq. (3). Thus, the multiplicative update rules in Eq. (21) or Eq. (22) are also the same as the standard concept factorization model in Eq. (3), and will reduce the objective function in Eq. (20). Please see [58] for details. For fixed $\mathbf{U}^t$ and $\mathbf{V}^t$, the rest objective function w.r.t $\boldsymbol{\mu}$ in Eq. (23) will also be decreased by the quadratic optimization tools. In summary, the objective function in Eq. (19) will be non-increasing under the alternative optimization step w.r.t. $\mathbf{U}$, $\mathbf{V}$ and $\boldsymbol{\mu}^t$. Since the objective function in Eq. (19) is obviously lower bounded, the overall optimization problem in Eq. (19) converges.

E. ALGORITHM COMPLEXITY ANALYSIS

In this subsection, we discuss the computational complexity of our proposed algorithm, and use the big $\mathcal{O}$ notation to express the complexity. The computation cost of finding τ-nearest neighbors of all sample points in all the base kernels is $\mathcal{O}(mn\log\tau)$; $\mathcal{O}(\tau^3)$ is the computation cost of single local discriminant model; The computation cost of computing the multiple Laplacian matrices is $\mathcal{O}(mn\tau^3)$. The computation cost of computing $\mathbf{K}_{\boldsymbol{\mu}}$ and $\mathbf{L}_{\boldsymbol{\mu}}$ is $\mathcal{O}(mn^2)$; The computation cost of the multiplicative updating in Eq. (21) or Eq. (22) is $\mathcal{O}(n^2k)$; The computation cost of the multiplicative updating in Eq. (25) or Eq. (26) is also $\mathcal{O}(n^2k)$; The computation cost of solving Eq. (23) is $\mathcal{O}(n^2k + m^3)$. If the updating procedure stops after t iterations, the overall cost of the multiplicative updating is $\mathcal{O}(tn^2(m+k) + tm^3)$. Because $n \gg m$ and $n \gg \tau$, the total cost of DMKCF is $\mathcal{O}(mn + n^2mt)$. It can be seen that the computational complexity of DMKCF is linear with the number of kernels and iterations, quadratic with the number of samples.

V. EXPERIMENT

In this section, to evaluate the effectiveness of our proposed MKC algorithm, especially the effectiveness, four experiments are designed. In the first experiment, we construct a synthetic data set to test the robustness against noise and

TABLE 2. Clustering comparison of all these 10 algorithms. We report the best results in terms of ACC/NMI/Purity respectively, from multiple random initializations and parameters.

Data Sets	Metrics	CTSC	Coreg	RMSC	RMKKM	MKKMMR	LKAMKC	ONMKC	LKGr	JMKSC	DMKCF
USPS49	ACC	0.7095	0.7400	0.7466	0.7854	0.7657	0.7651	0.8213	0.7884	0.8452	**0.9032**
	NMI	0.1328	0.1731	0.1832	0.2532	0.2165	0.2147	0.3241	0.2567	0.3783	**0.5413**
	Purity	0.7095	0.7400	0.7466	0.7854	0.7657	0.7651	0.8213	0.7884	0.8452	**0.9032**
PIE	ACC	0.6418	0.4716	0.5924	0.2682	0.6558	0.6380	0.6215	0.6078	0.7160	**0.7878**
	NMI	0.8349	0.7618	0.8086	0.5483	0.8376	0.8200	0.8210	0.8272	0.8770	**0.9093**
	Purity	0.6558	0.5084	0.6204	0.3186	0.6859	0.6516	0.6492	0.6359	0.7482	**0.8043**
COIL20	ACC	0.6861	0.6563	0.6806	0.6479	0.6597	0.6861	0.6924	0.6854	0.7688	**0.8632**
	NMI	0.7846	0.7725	0.7722	0.7476	0.7731	0.7868	0.7775	0.7605	0.8534	**0.9280**
	Purity	0.6979	0.6771	0.6840	0.6667	0.6951	0.7090	0.7090	0.6903	0.7813	**0.8847**
RELATHE	ACC	0.6412	0.6741	0.5809	0.5683	0.6300	0.7169	0.6370	0.5830	0.5718	**0.8150**
	NMI	0.0990	0.1272	0.0548	0.0090	0.0908	0.1537	0.0910	0.0379	0.0295	**0.3269**
	Purity	0.6412	0.6741	0.5809	0.5683	0.6300	0.7169	0.6370	0.5830	0.5718	**0.8150**
BBC	ACC	0.5577	0.5414	0.5400	0.7843	0.7436	0.7517	0.7463	0.5265	0.5631	**0.8440**
	NMI	0.3807	0.3655	0.3708	0.5839	0.4751	0.5216	0.5051	0.3947	0.3310	**0.6202**
	Purity	0.6024	0.5943	0.6431	0.7843	0.7436	0.7517	0.7463	0.6160	0.5984	**0.8440**
K1B	ACC	0.6962	0.6821	0.7821	0.8850	0.7688	0.7765	0.7598	0.7564	0.8598	**0.9256**
	NMI	0.5396	0.5663	0.5745	0.7406	0.5878	0.5953	0.5577	0.5825	0.7380	**0.7658**
	Purity	0.8410	0.8607	0.8714	0.8979	0.8880	0.8889	0.8714	0.8761	0.8957	**0.9504**
Prostate	ACC	0.5784	0.5784	0.5784	0.5784	0.6176	0.6176	0.5784	0.6275	0.6373	**0.6471**
	NMI	0.0178	0.0178	0.0176	0.0178	0.0426	0.0492	0.0176	0.0574	0.0551	**0.0800**
	Purity	0.5784	0.5784	0.5784	0.5784	0.6176	0.6176	0.5784	0.6275	0.6373	**0.6471**
ALLAML	ACC	0.7361	0.6667	0.5694	0.7361	0.6667	0.7083	0.7222	0.7639	0.7639	**0.8472**
	NMI	0.1509	0.0862	0.0385	0.1509	0.0862	0.2090	0.1461	0.1863	0.1900	**0.3416**
	Purity	0.7361	0.6667	0.6528	0.7361	0.6667	0.7083	0.7222	0.7639	0.7639	**0.8472**
SMKCAN	ACC	0.5989	0.5668	0.6096	0.6043	0.6310	0.6471	0.6524	0.6417	0.5989	**0.6845**
	NMI	0.0275	0.0119	0.0340	0.0307	0.0524	0.0672	0.0726	0.0620	0.0274	**0.1044**
	Purity	0.5989	0.5668	0.6096	0.6043	0.6310	0.6471	0.6524	0.6417	0.5989	**0.6845**
CLLSUB	ACC	0.5495	0.5495	0.5405	0.5315	0.5495	0.5495	0.5405	0.5676	0.5676	**0.5946**
	NMI	0.2631	0.2631	0.2154	0.1807	0.2631	0.2631	0.1987	0.2608	0.2661	**0.2952**
	Purity	0.5495	0.5495	0.5405	0.5315	0.5495	0.5495	0.5405	0.5676	0.5676	**0.6036**
Average	mACC	0.6395	0.6127	0.6221	0.6389	0.6688	0.6867	0.6772	0.6548	0.6892	**0.7912**
	mNMI	0.3231	0.3145	0.3070	0.3263	0.3425	0.3681	0.3511	0.3426	0.3746	**0.4913**
	mPurity	0.6611	0.6416	0.6528	0.6472	0.6873	0.7006	0.6928	0.6790	0.7008	**0.7984**

outliers of the proposed neighbor kernel. Second, we compare our proposed algorithm with nine state-of-the-art MKC algorithms on real-world data sets to evaluate its performance. Then, we test the sensitivity of the algorithm against the main hyperparameters. Finally, we apply neighbor kernels to the existing MKC algorithms and test the capacity of the proposed kernel on enhancing the performance of these methods.

A. DATA SETS

We perform experiments on 10 different public datasets, including 3 image ones (USPS49, PIE, COIL20),3 text corpora ones (RELATHE,BBC,K1b) and 4 biological ones (Prostate,ALLAML,SMKCAN,CLLSUB). They have been widely used to evaluate the performance of different clustering methods. The detailed statistics information and data dimensionality of these datasets are summarized in Table 1.

B. COMPARED ALGORITHMS

To demonstrate how the clustering performance can be improved by the proposed approach, we compared the results of the following state-of-the-art multiple kernel clustering algorithms, which include:

- **CTSC**.[1] It is a co-training multiview spectral clustering proposed by [33].
- **Coreg**.[2] It is a co-regularized multiview spectral clustering proposed by [34].
- **RMSC**.[3] The RMSC (Robust Multiview Spectral Clustering) is proposed by [39]. We first transform the kernels into probabilistic transition matrices following [39], and then apply RMSC to get the final clustering results.
- **RMKKM**.[4] The robust multiple kernel k-means method with $\ell_{2,1}-$norm for data clustering [24].
- **MKKMMR**.[5] It improves the multiple kernel k-means with matrix-induced regularization [25].
- **LKAMKC**.[6] The LKAMKC algorithm is proposed in [52] by introducing the local kernel alignment for multiple kernel clustering.

[1] We use the code at http://users.umiacs.umd.edu/~abhishek/code_cospectral.zip.

[2] We use the code at http://users.umiacs.umd.edu/~abhishek/code_coregspectral.zip.

[3] We use the code at http://ss.sysu.edu.cn/~py/RMSC.zip.

[4] We use the code at https://github.com/csliangdu/RMKKM.

[5] We use the code provided by the authors.

[6] We use the code provided by the authors.

TABLE 3. Clustering comparison on (mean ACC)/(standard derivation)/(*p*-value). The results shown in boldface are significant better than the others, with a significant level of 0.05.

Data Sets	CTSC	Coreg	RMSC	RMKKM	MKKMMR	LKAMKC	ONMKC	LKGr	JMKSC	DMKCF
USPS49	70.95 ± 0.00e+00 0.00e+00	73.93 ± 6.10e-02 7.85e-48	74.61 ± 2.66e-02 2.39e-54	78.51 ± 3.05e-02 7.55e-51	76.61 ± 2.66e-02 3.20e-53	76.55 ± 2.81e-02 8.60e-53	80.65 ± 6.36e+00 1.73e-06	78.84 ± 0.00e+00 0.00e+00	84.52 ± 1.14e-14 0.00e+00	**90.32** ± **0.00e+00** **1.00e+00**
PIE	57.32 ± 3.45e+00 5.66e-13	48.24 ± 1.78e+00 3.62e-17	57.41 ± 2.50e+00 5.99e-12	25.96 ± 1.30e+00 1.30e-12	61.24 ± 2.17e+00 6.67e-14	58.10 ± 2.11e+00 8.48e-27	57.78 ± 2.77e+00 8.49e-27	55.33 ± 2.06e+00 5.97e-27	63.12 ± 2.29e+00 4.86e-23	**71.86** ± **3.00e+00** **1.00e+00**
COIL20	60.57 ± 4.06e+00 5.41e-07	62.13 ± 3.60e+00 1.09e-07	59.92 ± 4.16e+00 8.79e-07	61.65 ± 3.24e+00 1.02e-07	60.97 ± 3.76e+00 6.89e-08	60.31 ± 3.07e+00 2.54e-08	59.27 ± 4.39e+00 1.78e-07	58.68 ± 3.55e+00 8.12e-09	62.20 ± 5.21e+00 7.10e-06	**72.57** ± **5.71e+00** **1.00e+00**
RELATHE	64.54 ± 2.97e-01 4.75e-35	67.41 ± 2.28e-14 0.00e+00	59.69 ± 2.73e+00 6.69e-19	56.31 ± 9.18e-01 5.17e-29	63.03 ± 3.59e-02 3.49e-53	71.58 ± 1.41e-01 8.65e-37	63.70 ± 0.00e+00 4.39e-296	58.30 ± 1.14e-14 1.04e-290	57.18 ± 1.57e-02 2.62e-62	**81.50** ± **2.28e-14** **1.00e+00**
BBC	54.99 ± 4.43e+00 3.47e-10	55.53 ± 3.94e+00 1.19e-10	55.79 ± 3.28e+00 2.81e-10	**77.44** ± **1.22e+00** **1.38e-01**	69.04 ± 7.40e+00 1.39e-04	74.23 ± 3.87e+00 1.03e-02	69.37 ± 4.88e+00 5.99e-05	54.57 ± 4.86e+00 2.67e-10	47.48 ± 3.62e+00 2.33e-13	**80.16** ± **7.97e+00** **1.00e+00**
K1b	69.75 ± 2.83e+00 4.54e-17	64.62 ± 4.16e+00 8.98e-17	71.72 ± 4.52e+00 8.44e-14	81.25 ± 5.77e+00 3.30e-07	69.54 ± 7.15e+00 4.18e-11	69.49 ± 6.51e+00 1.49e-11	78.25 ± 1.56e+00 2.82e-16	72.86 ± 1.89e+00 7.91e-18	69.70 ± 1.81e+00 3.21e-21	**91.26** ± **1.20e+00** **1.00e+00**
Prostate	57.84 ± 1.14e-14 0.00e+00	57.84 ± 1.14e-14 0.00e+00	57.84 ± 1.14e-14 0.00e+00	57.75 ± 3.02e-01 2.46e-26	61.32 ± 5.00e-01 8.77e-15	61.37 ± 4.93e-01 9.65e-15	56.08 ± 1.70e+00 2.90e-14	62.75 ± 2.28e-14 0.00e+00	62.84 ± 1.00e+00 8.73e-04	**63.73** ± **2.28e-14** **1.00e+00**
ALLAML	73.61 ± 2.28e-14 1.81e-293	67.01 ± 8.87e-01 1.01e-25	56.39 ± 6.98e-01 7.91e-32	71.04 ± 2.64e+00 1.51e-14	66.67 ± 2.28e-14 3.39e-292	68.19 ± 4.18e+00 1.39e-12	67.78 ± 6.40e+00 1.34e-09	76.39 ± 2.28e-14 4.89e-294	71.81 ± 5.88e+00 4.15e-08	**83.33** ± **2.28e-14** **1.00e+00**
SMKCAN	59.49 ± 7.13e-01 1.37e-22	56.10 ± 5.46e-01 1.99e-27	60.96 ± 1.14e-14 1.17e-294	60.37 ± 1.65e-01 8.32e-34	63.10 ± 0.00e+00 0.00e+00	64.71 ± 1.14e-14 0.00e+00	64.55 ± 7.78e-01 3.88e-15	64.17 ± 2.28e-14 0.00e+00	59.89 ± 1.14e-14 2.06e-292	**68.45** ± **3.42e-14** **1.00e+00**
CLLSUB	53.02 ± 4.79e+00 2.66e-02	**54.37** ± **2.62e+00** **6.12e-02**	53.02 ± 4.90e+00 2.89e-02	53.15 ± 0.00e+00 1.51e-04	**54.41** ± **2.42e+00** **5.89e-02**	53.87 ± 3.33e+00 3.83e-02	50.50 ± 5.13e+00 7.07e-04	52.57 ± 5.28e+00 2.30e-02	54.23 ± 2.22e+00 3.47e-02	**56.13** ± **2.82e+00** **1.00e+00**
Average	62.21	60.72	60.74	62.34	64.59	65.84	64.79	63.45	63.30	**75.93**

- **ONMKC**.[7] It learns an optimal neighborhood kernel directly for multiple kernel clustering [26].
- **LKGr**.[8] It learns a low-rank kernel matrix from the neighborhood of candidate kernels. [59].
- **JMKSC**.[9] It uses the correntropy metric weighting and block diagonal regularizer for robust multiple kernel subspace clustering [60].

C. EXPERIMENTAL SETTINGS

Following the similar strategy of other multiple kernel learning approaches, we apply 12 different kernel functions as basis for multiple kernel clustering. These kernels include, seven Gaussian kernels $\mathcal{K}(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp(-||\boldsymbol{x}_i - \boldsymbol{x}_j||^2/2\delta^2)$ with $\delta = \sqrt{t}{*}D_0$, where D_0 is the maximum distance between samples and t varies in the range of $\{-8, -4, -2, 1, 2, 4, 8\}$, four polynomial kernels $\mathcal{K}(\boldsymbol{x}_i, \boldsymbol{x}_j) = (a + \boldsymbol{x}_i^T \boldsymbol{x}_j)^b$ with $a = \{0, 1\}$ and $b = \{2, 4\}$ and a linear kernel $\mathcal{K}(\boldsymbol{x}_i, \boldsymbol{x}_j) = \boldsymbol{x}_i^T \boldsymbol{x}_j$. Finally, all the kernels have been normalized through $\mathcal{K}(\boldsymbol{x}_i, \boldsymbol{x}_j) = \mathcal{K}(\boldsymbol{x}_i, \boldsymbol{x}_j)/\sqrt{\mathcal{K}(\boldsymbol{x}_i, \boldsymbol{x}_i)\mathcal{K}(\boldsymbol{x}_j, \boldsymbol{x}_j)}$ and then rescaled to [0, 1].

There are some parameters to be set in advance. The number of clusters is set to the true number of classes for all the data sets and clustering algorithms. The regularization parameter γ in RMKKM [24] is searched over $\{0.1, 0.2, \cdots, 0.9\}$ as suggested. We search the regularization parameters τ in the range of $\{0.01, 0.1, \cdots, 0.95\} * n$, λ in the range of $\{2^{-15}, 2^{-14}, \cdots, 2^{15}\}$ for the method of LKAMKC [52] as suggested. We search the regularization parameters ρ in the range of $\{2^{-15}, 2^{-13}, \cdots, 2^{15}\}$, λ in the range of $\{2^{-15}, 2^{-13}, \cdots, 2^{15}\}$ for the method of ONMKC [26] as suggested. Since the original search range of α, β, γ is not specified [59]. We search these regularization parameters α, β, γ in the wide range of $\{10^{-5}, 10^{-3}, \cdots, 10^{5}\}$ for the method of LKGr [59] to make a fair comparison. We search the regularization parameters α in the range of $\{10^{-4}, 10^{-3}, \cdots, 10^{1}\}$, β in the range of $\{1, 5, 10, 15, 20, 25, 30\}$, and γ in the range of $\{10^{-1}, 1, 5, 10, 15, 20, 25, 30\}$ for the method of JMKSC [60] as suggested. For our method DMKCF, the neighborhood size k is set to $k = 5$ in all the experiments, the regularization parameters γ, λ, ξ are searched over the range of $\{10^{-5}, 10^{-4}, \cdots, 10^{5}\}$.

In this paper, three measures, i.e., ACC, NMI and Purity, are used to evaluate the clustering results.

D. EXPERIMENTAL RESULTS

The results of all clustering algorithms depend upon the initialization [56]. For all the clustering algorithms,

[7] We use the code provided by the authors.

[8] We use the code at https://github.com/sckangz/KBS18.

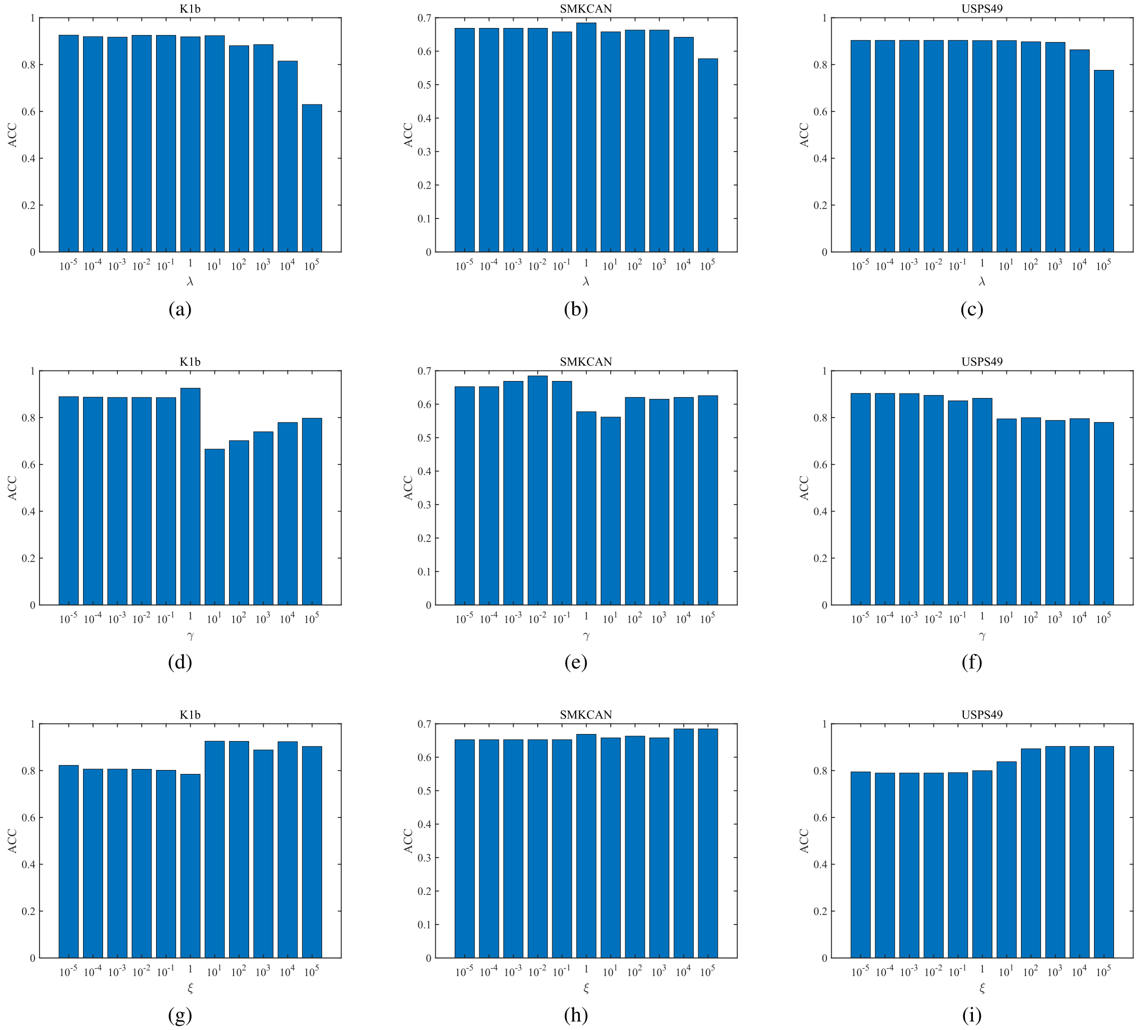
[9] We use the code provided by the authors.

TABLE 4. Clustering comparison on (mean NMI)/(standard derivation)/(*p*-value). The results shown in boldface are significant better than the others, with a significant level of 0.05.

Data Sets	CTSC	Coreg	RMSC	RMKKM	MKKMMR	LKAMKC	ONMKC	LKGr	JMKSC	DMKCF
USPS49	13.28 ± 2.85e-15 0.00e+00	17.22 ± 8.39e-02 6.65e-52	18.25 ± 4.18e-02 2.01e-57	25.27 ± 6.28e-02 2.93e-52	21.72 ± 4.61e-02 8.83e-56	21.54 ± 4.63e-02 8.67e-56	32.34 ± 1.18e-01 9.78e-45	25.67 ± 5.70e-15 5.88e-300	37.83 ± 1.14e-14 4.45e-301	**54.13** **± 0.00e+00** **1.00e+00**
PIE	79.24 ± 2.45e+00 8.48e-12	75.41 ± 0.88e+00 1.18e-16	79.51 ± 1.62e+00 1.70e-11	54.40 ± 0.67e+00 7.49e-10	82.03 ± 0.79e+00 9.86e-16	80.40 ± 0.67e+00 1.13e-34	80.53 ± 0.97e+00 1.13e-34	80.31 ± 0.82e+00 3.95e-33	84.26 ± 1.00e+00 2.63e-18	**71.86** **± 5.06e+00** **1.00e+00**
COIL20	74.46 ± 1.70e+00 4.91e-07	75.04 ± 1.59e+00 8.84e-07	74.05 ± 2.09e+00 2.79e-07	73.36 ± 1.93e+00 2.04e-08	73.98 ± 2.12e+00 1.36e-06	74.27 ± 1.99e+00 1.55e-09	72.70 ± 1.86e+00 2.34e-08	70.07 ± 2.09e+00 2.78e-11	77.95 ± 2.72e+00 1.96e-03	**81.24** **± 3.42e+00** **1.00e+00**
RELATHE	10.29 ± 2.82e-01 8.60e-38	12.72 ± 5.70e-15 4.92e-297	5.96 ± 7.77e-01 6.99e-31	2.53 ± 2.65e+00 9.09e-22	9.13 ± 2.60e-02 7.35e-58	15.25 ± 1.52e-01 7.95e-41	9.10 ± 1.42e-15 1.09e-292	3.76 ± 1.27e-01 1.71e-46	2.95 ± 0.00e+00 1.33e-294	**32.69** **± 1.14e-14** **1.00e+00**
BBC	38.20 ± 1.90e+00 2.53e-13	38.26 ± 2.66e+00 4.56e-15	37.80 ± 1.50e+00 5.17e-14	56.11 ± 3.69e+00 1.53e-02	46.32 ± 3.09e+00 2.37e-10	50.66 ± 3.29e+00 1.75e-07	45.42 ± 5.49e+00 5.95e-08	38.74 ± 1.69e+00 1.73e-14	25.42 ± 5.19e+00 1.32e-14	**60.11** **± 4.67e+00** **1.00e+00**
K1b	53.36 ± 1.41e+00 5.03e-24	53.54 ± 3.28e+00 4.17e-17	53.55 ± 2.42e+00 1.45e-19	61.84 ± 6.48e+00 3.96e-08	53.64 ± 5.17e+00 1.73e-13	54.50 ± 5.70e+00 2.23e-12	55.11 ± 1.70e+00 6.33e-22	58.65 ± 4.75e+00 5.82e-12	47.18 ± 6.17e-01 8.24e-33	**74.65** **± 8.52e-02** **1.00e+00**
Prostate	1.78 ± 0.00e+00 0.00e+00	1.78 ± 0.00e+00 0.00e+00	1.76 ± 0.00e+00 1.57e-295	1.66 ± 4.13e-01 3.15e-24	4.04 ± 2.77e-01 1.18e-23	3.57 ± 1.04e+00 8.43e-14	1.27 ± 4.66e-01 9.91e-24	5.74 ± 7.12e-16 6.33e-302	5.51 ± 2.14e-15 1.14e-293	**8.00** **± 0.00e+00** **1.00e+00**
ALLAML	15.09 ± 2.85e-15 0.00e+00	8.81 ± 8.68e-01 1.58e-29	3.61 ± 3.08e-01 1.31e-39	14.07 ± 3.76e+00 1.21e-15	8.62 ± 0.00e+00 4.59e-299	15.82 ± 8.02e+00 3.71e-09	14.57 ± 3.67e-02 1.66e-53	18.63 ± 5.70e-15 5.84e-295	17.75 ± 1.82e+00 6.80e-20	**34.16** **± 0.00e+00** **1.00e+00**
SMKCAN	2.53 ± 3.79e-01 9.31e-27	0.98 ± 1.95e-01 1.07e-33	3.40 ± 7.12e-16 7.16e-300	3.00 ± 1.69e-01 6.67e-33	5.20 ± 2.46e-02 6.01e-46	6.72 ± 1.42e-15 0.00e+00	6.45 ± 1.35e+00 4.66e-11	6.20 ± 0.00e+00 0.00e+00	2.74 ± 7.12e-16 6.90e-295	**10.44** **± 1.42e-15** **1.00e+00**
CLLSUB	22.62 ± 9.01e+00 3.89e-02	**25.07** **± 5.55e+00** **1.09e-01**	22.35 ± 6.85e+00 8.61e-03	18.07 ± 5.70e-15 3.15e-07	**25.27** **± 4.63e+00** **1.06e-01**	22.67 ± 8.89e+00 3.89e-02	11.49 ± 6.97e+00 3.06e-08	23.81 ± 3.30e+00 1.36e-02	22.82 ± 4.83e+00 5.28e-03	**28.20** **± 5.91e+00** **1.00e+00**
Average	31.09	30.88	30.02	31.03	32.99	34.54	30.90	33.16	32.44	**45.54**

TABLE 5. Clustering comparison on (mean Purity)/(standard derivation)/(*p*-value). The results shown in boldface are significant better than the others, with a significant level of 0.05.

Data Sets	CTSC	Coreg	RMSC	RMKKM	MKKMMR	LKAMKC	ONMKC	LKGr	JMKSC	DMKCF
USPS49	70.95 ± 0.00e+00 0.00e+00	73.93 ± 6.10e-02 7.85e-48	74.61 ± 2.66e-02 2.39e-54	78.52 ± 2.93e-02 3.46e-51	76.60 ± 3.00e-02 3.29e-52	76.55 ± 2.93e-02 1.83e-52	82.08 ± 6.01e-02 2.77e-42	78.84 ± 0.00e+00 0.00e+00	84.52 ± 1.14e-14 0.00e+00	**90.32** **± 0.00e+00** **1.00e+00**
PIE	59.64 ± 3.42e+00 1.88e-13	51.19 ± 1.80e+00 9.31e-17	59.79 ± 2.32e+00 2.01e-12	30.37 ± 0.95e+00 1.22e-12	64.45 ± 2.17e+00 2.02e-12	61.08 ± 1.81e+00 1.16e-27	61.02 ± 2.31e+00 1.16e-27	59.08 ± 1.68e+00 5.72e-28	67.11 ± 1.71e+00 8.02e-25	**71.86** **± 3.75e+00** **1.00e+00**
COIL20	63.64 ± 3.03e+00 7.38e-07	64.98 ± 2.75e+00 6.69e-07	63.55 ± 3.49e+00 1.31e-06	63.56 ± 2.31e+00 1.23e-08	62.63 ± 3.43e+00 8.28e-07	63.51 ± 3.18e+00 1.63e-07	64.70 ± 2.65e+00 9.17e-08	60.90 ± 3.20e+00 3.79e-09	68.37 ± 4.06e+00 8.71e-05	**73.87** **± 4.75e+00** **1.00e+00**
RELATHE	64.54 ± 2.97e-01 4.75e-35	67.41 ± 2.28e-14 0.00e+00	59.69 ± 2.73e+00 6.69e-19	56.15 ± 9.83e-01 1.69e-28	63.05 ± 3.29e-02 6.75e-54	71.53 ± 1.43e-01 1.07e-36	63.70 ± 0.00e+00 4.39e-296	58.28 ± 1.10e-01 7.19e-46	57.18 ± 0.00e+00 2.59e-295	**81.50** **± 2.28e-14** **1.00e+00**
BBC	60.60 ± 2.41e+00 1.54e-10	61.63 ± 2.85e+00 4.45e-11	64.15 ± 2.85e-01 3.07e-10	76.72 ± 2.52e+00 2.10e-02	72.69 ± 3.83e+00 1.48e-04	74.04 ± 3.44e+00 7.35e-05	71.15 ± 4.63e+00 6.48e-05	60.80 ± 2.57e+00 2.51e-11	52.86 ± 4.39e+00 1.72e-11	**80.35** **± 6.13e+00** **1.00e+00**
K1b	82.66 ± 2.08e+00 8.79e-14	84.00 ± 2.77e+00 6.94e-10	84.06 ± 1.52e+00 2.05e-14	85.00 ± 2.98e+00 1.68e-08	84.08 ± 1.51e+00 1.69e-13	84.77 ± 2.87e+00 1.96e-09	85.86 ± 8.05e-01 4.96e-15	86.53 ± 9.41e-01 1.10e-15	81.47 ± 5.00e-01 9.74e-23	**91.34** **± 5.16e-01** **1.00e+00**
Prostate	57.84 ± 1.14e-14 0.00e+00	57.84 ± 1.14e-14 0.00e+00	57.84 ± 1.14e-14 0.00e+00	57.50 ± 1.53e+00 1.31e-14	61.32 ± 5.00e-01 1.57e-17	61.47 ± 4.61e-01 7.78e-18	56.52 ± 1.28e+00 4.61e-17	62.75 ± 2.28e-14 0.00e+00	62.84 ± 1.00e+00 9.25e-08	**64.71** **± 1.14e-14** **1.00e+00**
ALLAML	73.61 ± 2.28e-14 1.81e-293	67.01 ± 8.87e-01 1.01e-25	65.28 ± 2.28e-14 7.42e-293	73.33 ± 2.32e+00 6.42e-14	66.67 ± 2.28e-14 3.39e-292	69.31 ± 3.09e+00 2.37e-14	70.35 ± 3.07e+00 8.50e-14	76.39 ± 2.28e-14 4.89e-294	71.94 ± 5.58e+00 2.27e-08	**83.33** **± 2.28e-14** **1.00e+00**
SMKCAN	59.49 ± 7.13e-01 1.37e-22	56.10 ± 5.46e-01 1.99e-27	60.96 ± 1.14e-14 1.17e-294	60.51 ± 2.62e-01 7.61e-30	63.10 ± 0.00e+00 0.00e+00	64.71 ± 1.14e-14 0.00e+00	64.33 ± 7.97e-01 2.27e-15	64.17 ± 2.28e-14 0.00e+00	59.89 ± 1.14e-14 2.06e-292	**68.45** **± 3.42e-14** **1.00e+00**
CLLSUB	53.74 ± 2.97e+00 1.43e-08	54.55 ± 1.81e+00 5.99e-11	53.92 ± 2.15e+00 1.82e-10	53.15 ± 0.00e+00 5.26e-22	53.74 ± 2.98e+00 1.68e-09	53.74 ± 2.97e+00 1.43e-08	50.59 ± 3.07e+00 5.56e-11	52.75 ± 5.04e+00 1.55e-06	54.32 ± 1.78e+00 2.71e-11	**60.23** **± 6.04e-01** **1.00e+00**
Average	64.67	63.86	64.39	63.48	66.83	68.07	67.03	66.05	66.05	**76.60**



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

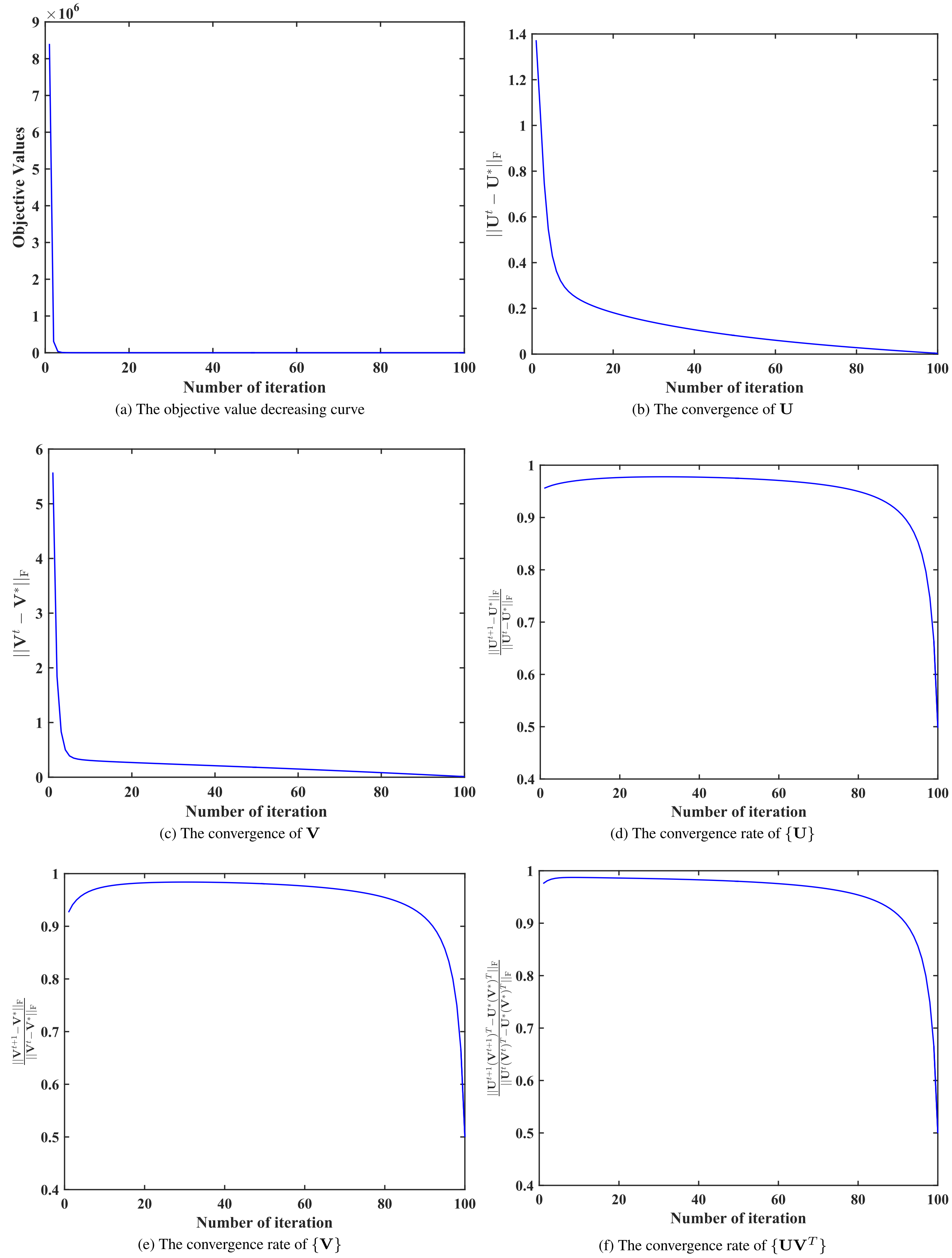
FIGURE 1. ACC variants of λ, γ and ξ on K1b, SMKCAN, USPS49, respectively. (a-c) ACC variants of λ on K1b, SMKCAN, USPS49. (d-f) ACC variants of γ on K1b, SMKCAN, USPS49. (g-i) ACC variants of ξ on K1b, SMKCAN, USPS49.

we independently repeat the experiments 20 times with random initializations to reduce the statistical variation.

For each clustering algorithm, we report the best results for each parameter corresponding to the best objective values in terms of ACC/NMI/Purity, respectively, from twenty rounds of random initializations in Table 2. We also report the averaged results over all these 10 data sets in the last row of Table 2. It can be seen that our method consistently outperform other state-of-the-art multipl kernel clustering algorithms. Moreover, it can be seen that our method achieves 14.79%, 31.15% and 13.93% improvement in terms of ACC/NMI/Purity, respectively on the averaged results. These results can well demonstrate the effectiveness of the proposed method.

For each clustering algorithm, we also calculate the the mean ACC/NMI/Purity from twenty rounds of random initializations for each parameter and then we additionally report the best mean ACC/NMI/Purity together with the standard deviation corresponding to the optimal parameter and the p-value of the paired t-test against the best results in Table 3, 4, 5. Thus, each cell in Table 3, 4, 5 include the best mean ACC/NMI/Purity, the standard deviation and the p-value. The best one and those having no significant difference ($p > 0.05$) from the best one are marked in bold. Again, we can observe that our method outperforms better than other MKC algorithms in most cases. And the improvements in most cases are also significant.

For all these compared multiple kernel clustering algorithms, we can observe that the ACC/NMI/Purity in Table 2 corresponding to the best objective values are generally higher than the mean ACC/NMI/Purity in Table 3, 4, 5. Since it is still a nontrivial task to obtain the globally optimal solutions for the clustering algorithms, it is reasonable to choose the clustering results from the optimal initialization corresponding to the best objective values in practical clustering applications [56].



(a) The objective value decreasing curve

(b) The convergence of $\mathbf{U}$

(c) The convergence of $\mathbf{V}$

(d) The convergence rate of $\{\mathbf{U}\}$

(e) The convergence rate of $\{\mathbf{V}\}$

(f) The convergence rate of $\{\mathbf{U}\mathbf{V}^T\}$

FIGURE 2. **The convergence behavior of our method on BBC data set.**

E. PARAMETER SENSITIVITY

In the subsection, we investigate the sensitivities of three parameters λ, γ, and ξ in our method. Figure 1 plots the clustering accuracy(ACC) with different values of these parameters on K1b, SMKCAN, USPS49 respectively. These figures show that our proposed algorithm is not very sensitive to λ, γ and ξ within relative wide ranges.

F. CONVERGENCE ANALYSIS

Here we take the BBC data set as an example to empirically investigate the convergence behavior of the proposed method. In this experiment, we fix the three parameters $\lambda = 1, \gamma = 1$, and $\xi = 1$, and set the maximal iterations to be 100 for simplicity. We first provide the change of the objective function value of our proposed method with increasing number of iterations in Figure 2(a). Like [61], we then show how the value of $\mathbf{U}, \mathbf{V}$ get close to the optimal value $\mathbf{U}^*, \mathbf{V}^*$ with respect to the iteration number t on this data set by computing $||\mathbf{U}^t - \mathbf{U}^*||_{\mathrm{F}}, ||\mathbf{V}^t - \mathbf{V}^*||_{\mathrm{F}}$, in Figure 2(b) and Figure 2(c), separately. Although it is still not easy to provide theoretical results on the convergence rate of the proposed optimization schema, we further provide more empirical results on the sequences of $\{\frac{||\mathbf{U}^{t+1} - \mathbf{U}^*||_{\mathrm{F}}}{||\mathbf{U}^t - \mathbf{U}^*||_{\mathrm{F}}}\}, \{\frac{||\mathbf{V}^{t+1} - \mathbf{V}^*||_{\mathrm{F}}}{||\mathbf{V}^t - \mathbf{V}^*||_{\mathrm{F}}}\}, \{\frac{||\mathbf{U}^{t+1}(\mathbf{V}^{t+1})^T - \mathbf{U}^*(\mathbf{V}^*)^T||_{\mathrm{F}}}{||\mathbf{U}^t(\mathbf{V}^t)^T - \mathbf{U}^*(\mathbf{V}^*)^T||_{\mathrm{F}}}\}$ in Figure 2(d), Figure 2(e) and Figure 2(f) to demonstrate the convergence rate as suggested [62]. It can be seen that the objective function indeed decreases its value with the updating rules on this data set. It can also be seen that both $\mathbf{U}^t, \mathbf{V}^t$ sequences can converge within a small number of iterations, which also verifies the effectiveness and correctness of the optimization scheme.

VI. CONCLUSION

In this paper, we propose a novel discriminative multiple kernel concept factorization method for data clustering and representation. Our method inherits the merit of concept factorization and extends to handle the problem of kernel design or selection. Our method also extracts the kernel level local discriminant model with global integration and builds the local multiple discriminant regularization to further capture the local discriminant structure of data. An iterative algorithm with convergence guarantee is also developed to find the optimal solution. Extensive experiments on 10 benchmark datasets further show that the proposed method outperforms many multiple clustering algorithms.

APPENDIX
PROOF OF CONVERGENCE

Lemma VI-1: [63] For any nonnegative matrices $\mathbf{A} \in \mathcal{R}^{n\times n}$, $\mathbf{B} \in \mathcal{R}^{k\times k}$, $\mathbf{S} \in \mathcal{R}^{n\times k}$,$\mathbf{S}' \in \mathcal{R}^{n\times k}$, *and* $\mathbf{A}$, $\mathbf{B}$ *are symmetric, then the following inequality holds*

$$\sum_{i=1}^{n}\sum_{j=1}^{k}\frac{(\mathbf{A}\mathbf{S}'\mathbf{B})_{ij}\mathbf{S}_{ij}^2}{\mathbf{S}'_{ij}} \geq \mathrm{tr}(\mathbf{S}^T\mathbf{A}\mathbf{S}\mathbf{B}) \tag{32}$$

The objective function with respect to $\mathbf{V}$ in Eq. (24) can be rewritten as

$$\begin{aligned}\mathcal{J}(\mathbf{V}) &= \mathrm{tr}(\mathbf{V}\mathbf{Q}^+\mathbf{V}^T) - \mathrm{tr}(\mathbf{V}\mathbf{Q}^-\mathbf{V}^T) - 2\mathrm{tr}(\mathbf{V}^T\mathbf{K}_{\boldsymbol{\mu}}^+\mathbf{U})\\ &\quad + 2\mathrm{tr}(\mathbf{V}^T\mathbf{K}_{\boldsymbol{\mu}}^-\mathbf{U}) + \lambda\mathrm{tr}(\mathbf{V}^T\mathbf{L}_{\boldsymbol{\mu}}^+\mathbf{V}) - \lambda\mathrm{tr}(\mathbf{V}^T\mathbf{L}_{\boldsymbol{\mu}}^-\mathbf{V})\\ &\quad + \xi\mathrm{tr}(\mathbf{V}^T\mathbf{V}\mathbf{V}^T\mathbf{V}) - 2\xi\mathrm{tr}(\mathbf{V}^T\mathbf{V})\end{aligned} \tag{33}$$

By applying Lemma VI-1, we have

$$\mathrm{tr}(\mathbf{V}\mathbf{Q}^+\mathbf{V}^T) \leq \sum_{i=1}^{n}\sum_{p=1}^{c}\frac{(\mathbf{V}'\mathbf{Q}^+)_{ip}\mathbf{V}_{ip}^2}{\mathbf{V}'_{ip}} \tag{34}$$

$$\mathrm{tr}(\mathbf{V}^T\mathbf{L}_{\boldsymbol{\mu}}^+\mathbf{V}) \leq \sum_{i=1}^{n}\sum_{p=1}^{c}\frac{(\mathbf{L}_{\boldsymbol{\mu}}^+\mathbf{V}')_{ip}\mathbf{V}_{ip}^2}{\mathbf{V}'_{ip}} \tag{35}$$

Moreover, by the inequality $a \leq \frac{a^2+b^2}{2b}, \forall a, b > 0$, we have the following inequality

$$\mathrm{tr}(\mathbf{V}^T\mathbf{K}_{\boldsymbol{\mu}}^-\mathbf{U}) \leq \sum_{i=1}^{n}\sum_{p=1}^{c}(\mathbf{K}_{\boldsymbol{\mu}}^-\mathbf{U})_{ip}\frac{\mathbf{V}_{ip}^2 + \mathbf{V'}_{ip}^2}{2\mathbf{V}'_{ip}} \tag{36}$$

$$\mathrm{tr}(\mathbf{V}^T\mathbf{V}\mathbf{V}^T\mathbf{V}) \leq \sum_{i=1}^{n}\sum_{p=1}^{c}\frac{(\mathbf{V}'\mathbf{V'}^T\mathbf{V}')_{ip}\mathbf{V}_{ip}^4}{\mathbf{V'}_{ip}^3} \tag{37}$$

To obtain the lower bound for the remaining terms, we use the inequality that $z \leq 1 + \log z, \forall z > 0$, then

$$\begin{aligned}&\mathrm{tr}(\mathbf{V}\mathbf{Q}^-\mathbf{V})\\ &\geq \sum_{i=1}^{n}\sum_{p=1}^{c}\sum_{q=1}^{c}\mathbf{Q}_{pq}^-\mathbf{V}'_{ip}\mathbf{V}'_{iq}(1 + \log\frac{\mathbf{V}_{ip}\mathbf{V}_{iq}}{\mathbf{V}'_{ip}\mathbf{V}'_{iq}})\end{aligned} \tag{38}$$

$$\begin{aligned}&\mathrm{tr}(\mathbf{V}^T\mathbf{K}_{\boldsymbol{\mu}}^+\mathbf{U})\\ &\geq \sum_{i=1}^{n}\sum_{p=1}^{c}(\mathbf{K}_{\boldsymbol{\mu}}^-\mathbf{U})_{ip}\mathbf{V}'_{ip}(1 + \log\frac{\mathbf{V}_{ip}}{\mathbf{V}'_{ip}})\end{aligned} \tag{39}$$

$$\begin{aligned}&\mathrm{tr}(\mathbf{V}^T\mathbf{L}_{\boldsymbol{\mu}}^-\mathbf{V})\\ &\geq \sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{p=1}^{c}(\mathbf{L}_{\boldsymbol{\mu}}^-)_{ij}\mathbf{V}'_{ip}\mathbf{V}'_{jp}(1 + \log\frac{\mathbf{V}_{ip}\mathbf{V}_{jp}}{\mathbf{V}'_{ip}\mathbf{V}'_{jp}})\end{aligned} \tag{40}$$

$$\begin{aligned}&\mathrm{tr}(\mathbf{V}^T\mathbf{V})\\ &\geq \sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{p=1}^{c}\mathbf{I}_{ij}\mathbf{V}'_{ip}\mathbf{V}'_{jp}(1 + \log\frac{\mathbf{V}_{ip}\mathbf{V}_{jp}}{\mathbf{V}'_{ip}\mathbf{V}'_{jp}})\end{aligned} \tag{41}$$

By summing over all the bounds in Eq. (34), (35), (36), (37), (38), (39), (40) and Eq. (41), we can get $\mathcal{J}(\mathbf{V}, \mathbf{V}')$, which satisfies (1) $\mathcal{J}(\mathbf{V}, \mathbf{V}') \geq \mathcal{J}(\mathbf{V})$; (2)$\mathcal{J}(\mathbf{V}, \mathbf{V}) \geq \mathcal{J}(\mathbf{V})$.

To find the minimum of $\mathcal{J}(\mathbf{V}, \mathbf{V}')$, we take

$$\begin{aligned}&\frac{\partial\mathcal{J}(\mathbf{V}, \mathbf{V}')}{\partial\mathbf{V}_{ip}}\\ &= 2\frac{(\mathbf{V}'\mathbf{Q}^+)_{ip}}{\mathbf{V}'_{ip}}\mathbf{V}_{ip} - 2\sum_{q=1}^{k}\mathbf{Q}_{qp}^-\mathbf{V}'_{iq}\frac{\mathbf{V}'_{ip}}{\mathbf{V}_{ip}}\\ &\quad - 2(\mathbf{K}_{\boldsymbol{\mu}}^+\mathbf{U})_{ip}\frac{\mathbf{V}'_{ip}}{\mathbf{V}_{ip}} + 2\frac{(\mathbf{K}_{\boldsymbol{\mu}}^-\mathbf{U})_{ip}}{\mathbf{V}'_{ip}}\mathbf{V}_{ip}\end{aligned}$$

$$+2\lambda\frac{(\mathbf{L}_{\boldsymbol{\mu}}^{+}\mathbf{V}')_{ip}}{\mathbf{V}'_{ip}}\mathbf{V}_{ip}-2\lambda\sum_{j=1}^{n}(\mathbf{L}_{\boldsymbol{\mu}}^{-})_{ij}\mathbf{V}'_{jp}\frac{\mathbf{V}'_{ip}}{\mathbf{V}_{ip}}$$
$$+4\xi\frac{(\mathbf{V}'\mathbf{V}'^{T}\mathbf{V}')_{ip}}{\mathbf{V}'^{3}_{ip}}\mathbf{V}_{ip}^{3}-4\xi\frac{\mathbf{V}'^{2}_{ip}}{\mathbf{V}_{ip}}. \tag{42}$$

and the following Hessian matrix of $\mathcal{J}(\mathbf{V},\mathbf{V}')$

$$\begin{aligned}\frac{\partial^{2}\mathcal{J}(\mathbf{V},\mathbf{V}')}{\partial\mathbf{V}_{ip}\partial\mathbf{V}_{jq}}
&= \delta_{ip}\delta_{jq}\Bigg(2\frac{(\mathbf{B}^{+}\mathbf{V}'\mathbf{P}^{+})_{ip}}{\mathbf{V}'_{ip}}+2\frac{(\mathbf{B}^{-}\mathbf{V}'\mathbf{P}^{-})_{ip}}{\mathbf{V}'_{ip}}\\
&\quad+2\sum_{j=1}^{n}\sum_{q=1}^{c}\mathbf{B}_{ij}^{-}\mathbf{P}_{qp}^{+}\frac{\mathbf{V}'_{ip}}{\mathbf{V}_{ip}^{2}}+2\sum_{j=1}^{n}\sum_{q=1}^{k}\mathbf{B}_{ij}^{+}\mathbf{P}_{qp}^{-}\frac{\mathbf{V}'_{ip}}{\mathbf{V}_{ip}^{2}}\\
&\quad+2\frac{\mathbf{M}_{ip}^{-}}{\mathbf{V}'_{ip}}+2\mathbf{M}_{ip}^{+}\frac{\mathbf{V}'_{ip}}{\mathbf{V}_{ip}^{2}}\Bigg)\\
&\quad+2\lambda_{1}\frac{(\mathbf{Z}^{+}\mathbf{V}')_{ip}}{\mathbf{V}'_{ip}}+2\lambda_{1}\sum_{j=1}^{d}\mathbf{Z}_{ij}^{-}\mathbf{V}'_{jp}\frac{\mathbf{V}'_{ip}}{\mathbf{V}_{ip}^{2}}\\
&\quad+2\lambda_{2}\frac{(\mathbf{Q}\mathbf{V}')_{ip}}{\mathbf{V}'_{ip}}\end{aligned} \tag{43}$$

is a diagonal matrix with positive diagonal elements.

Thus $\mathcal{J}(\mathbf{V},\mathbf{V}')$ is a convex function of $\mathbf{V}$. By setting $\frac{\partial\mathcal{J}(\mathbf{V},\mathbf{V}')}{\partial\mathbf{V}_{ip}}\mathbf{V}_{ip}=0$, we have

$$\begin{aligned}0 &= 2\xi(\mathbf{V}'\mathbf{V}'^{T}\mathbf{V}')_{ip}\mathbf{V}_{ip}^{4}\\
&\quad+((\mathbf{V}'\mathbf{Q}^{+})_{ip}+(\mathbf{K}_{\boldsymbol{\mu}}^{-}\mathbf{U})_{ip}+\lambda(\mathbf{L}_{\boldsymbol{\mu}}^{+}\mathbf{V}')_{ip})\mathbf{V}'^{2}_{ip}\mathbf{V}_{ip}^{2}\\
&\quad-(\sum_{q=1}^{c}\mathbf{V}'_{iq}\mathbf{Q}_{qp}^{-}+(\mathbf{K}_{\boldsymbol{\mu}}^{+}\mathbf{U})_{ip}\\
&\quad+\lambda\sum_{j=1}^{n}(\mathbf{L}_{\boldsymbol{\mu}}^{-})_{ij}\mathbf{V}'_{jp}+2\xi\mathbf{V}'_{ip})\mathbf{V}'^{4}_{ip}\end{aligned} \tag{44}$$

Hence, we can obtain the global minimum of $\mathcal{J}(\mathbf{V},\mathbf{V}')$ according to Eq. (26).

REFERENCES

[1] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, no. 6755, pp. 788–791, Oct. 1999.

[2] D. Seung and L. Lee, "Algorithms for non-negative matrix factorization," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 13, 2001, pp. 556–562.

[3] L. Zhang, Z. Liu, J. Pu, and B. Song, "Adaptive graph regularized nonnegative matrix factorization for data representation," *Appl. Intell.*, vol. 50, no. 2, pp. 438–447, 2020.

[4] W. Xu, X. Liu, and Y. Gong, "Document clustering based on non-negative matrix factorization," in *Proc. 26th Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retr. (SIGIR)*, 2003, pp. 267–273.

[5] K.-R. Muller, S. Mika, G. Ratsch, K. Tsuda, and B. Scholkopf, "An introduction to kernel-based learning algorithms," *IEEE Trans. Neural Netw.*, vol. 12, no. 2, pp. 181–201, Mar. 2001.

[6] 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, Jun. 2011.

[7] P. Li, C. Chen, and J. Bu, "Clustering analysis using manifold kernel concept factorization," *Neurocomputing*, vol. 87, pp. 120–131, Jun. 2012.

[8] X. Li, L. Du, and Y.-D. Shen, "Graph-based marginal ranking for update summarization," in *Proc. SIAM Int. Conf. Data Mining*, Apr. 2011, pp. 486–497.

[9] L. Du, C. Ren, X. Lv, Y. Chen, P. Zhou, and Z. Hu, "Local graph reconstruction for parameter free unsupervised feature selection," *IEEE Access*, vol. 7, pp. 102921–102930, 2019.

[10] S. Ma, L. Zhang, W. Hu, Y. Zhang, J. Wu, and X. Li, "Self-representative manifold concept factorization with adaptive neighbors for clustering," in *Proc. 27th Int. Joint Conf. Artif. Intell.*, Jul. 2018, pp. 2539–2545.

[11] Z. Shu, X. Wu, P. Huang, H. Fan, Z. Liu, and F. Ye, "Multiple graph regularized concept factorization with adaptive weights," *IEEE Access*, vol. 6, pp. 64938–64945, 2018.

[12] H. Liu, Z. Yang, and Z. Wu, "Locality-constrained concept factorization," in *Proc. 22nd Int. Joint Conf. Artif. Intell.*, 2011, pp. 1378–1383.

[13] H. Liu, Z. Yang, J. Yang, Z. Wu, and X. Li, "Local coordinate concept factorization for image representation," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 6, pp. 1071–1082, Jun. 2014.

[14] W. Hua and X. He, "Discriminative concept factorization for data representation," *Neurocomputing*, vol. 74, no. 18, pp. 3800–3807, Nov. 2011.

[15] J. Deng, L. Du, and Y.-D. Shen, "Heterogeneous metric learning for cross-modal multimedia retrieval," in *Proc. Int. Conf. Web Inf. Syst. Eng.*, 2013, pp. 43–56.

[16] L. Wu, L. Du, B. Liu, G. Xu, Y. Ge, Y. Fu, J. Li, Y. Zhou, and H. Xiong, "Heterogeneous metric learning with content-based regularization for software artifact retrieval," in *Proc. IEEE Int. Conf. Data Mining*, Dec. 2014, pp. 610–619.

[17] W. Yan, B. Zhang, S. Ma, and Z. Yang, "A novel regularized concept factorization for document clustering," *Knowl.-Based Syst.*, vol. 135, pp. 147–158, Nov. 2017.

[18] H. Li, J. Zhang, J. Hu, C. Zhang, and J. Liu, "Graph-based discriminative concept factorization for data representation," *Knowl.-Based Syst.*, vol. 118, pp. 70–79, Feb. 2017.

[19] K. Zhan, J. Shi, J. Wang, and F. Tian, "Graph-regularized concept factorization for multi-view document clustering," *J. Vis. Commun. Image Represent.*, vol. 48, pp. 411–418, Oct. 2017.

[20] K. Zhan, J. Shi, J. Wang, H. Wang, and Y. Xie, "Adaptive structure concept factorization for multiview clustering," *Neural Comput.*, vol. 30, no. 4, pp. 1080–1103, Apr. 2018.

[21] S. Yu, L.-C. Tranchevent, X. Liu, W. Glanzel, J. A. K. Suykens, B. De Moor, and Y. Moreau, "Optimized data fusion for kernel k-means clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 5, pp. 1031–1039, May 2012.

[22] H.-C. Huang, Y.-Y. Chuang, and C.-S. Chen, "Multiple kernel fuzzy clustering," *IEEE Trans. Fuzzy Syst.*, vol. 20, no. 1, pp. 120–134, Feb. 2012.

[23] M. Gönen and A. A. Margolin, "Localized data fusion for kernel K-means clustering with application to cancer biology," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 1305–1313.

[24] L. Du, P. Zhou, L. Shi, H. Wang, M. Fan, W. Wang, and Y. Shen, "Robust multiple kernel k-means using l21-norm," in *Proc. 24th Int. Joint Conf. Artif. Intell.*, 2015, pp. 3476–3482.

[25] X. Liu, Y. Dou, J. Yin, L. Wang, and E. Zhu, "Multiple kernel K-means clustering with matrix-induced regularization," in *Proc. 30th AAAI Conf. Artif. Intell.*, 2016, pp. 1888–1894.

[26] X. Liu, S. Zhou, Y. Wang, M. Li, Y. Dou, E. Zhu, and J. Yin, "Optimal neighborhood kernel clustering with multiple kernels," in *Proc. 31st AAAI Conf. Artif. Intell.*, 2017, pp. 2266–2272.

[27] X. Zhu, X. Liu, M. Li, E. Zhu, L. Liu, Z. Cai, J. Yin, and W. Gao, "Localized incomplete multiple kernel k-means," in *Proc. 27th Int. Joint Conf. Artif. Intell.*, Jul. 2018, pp. 3271–3277.

[28] L. Platon, F. Zehraoui, and F. Tahi, "Localized multiple sources self-organizing map," in *Proc. Int. Conf. Neural Inf. Process.*, 2018, pp. 648–659.

[29] L. Xu, J. Neufeld, B. Larson, and D. Schuurmans, "Maximum margin clustering," in *Proc. Adv. Neural Inf. Process. Syst.*, 2004, pp. 1537–1544.

[30] H. Valizadegan and R. Jin, "Generalized maximum margin clustering and unsupervised kernel learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2006, pp. 1417–1424.

[31] B. Zhao, J. T. Kwok, and C. Zhang, "Multiple kernel clustering," in *Proc. SIAM Int. Conf. Data Mining*, Apr. 2009, pp. 638–649.

[32] H. Zeng and Y.-M. Cheung, "Feature selection and kernel learning for local learning-based clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 8, pp. 1532–1547, Aug. 2011.

[33] A. Kumar and H. Daumé, "A co-training approach for multi-view spectral clustering," in *Proc. 28th Int. Conf. Mach. Learn.*, 2011, pp. 393–400.

[34] A. Kumar, P. Rai, and H. Daumé, "Co-regularized multi-view spectral clustering," in *Proc. Adv. Neural Inf. Process. Syst.*, 2011, pp. 1413–1421.

[35] H.-C. Huang, Y.-Y. Chuang, and C.-S. Chen, "Affinity aggregation for spectral clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 773–780.

[36] H.-C. Huang, Y.-Y. Chuang, and C.-S. Chen, "Multi-affinity spectral clustering," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Mar. 2012, pp. 2089–2092.

[37] Y. Lu, L. Wang, J. Lu, J. Yang, and C. Shen, "Multiple kernel clustering based on centered kernel alignment," *Pattern Recognit.*, vol. 47, no. 11, pp. 3656–3664, Nov. 2014.

[38] B. Anderson, C. Storlie, and T. Lane, "Multiple kernel learning clustering with an application to malware," in *Proc. IEEE 12th Int. Conf. Data Mining*, Dec. 2012, pp. 804–809.

[39] R. Xia, Y. Pan, L. Du, and J. Yin, "Robust multi-view spectral clustering via low-rank and sparse decomposition," in *Proc. 28th AAAI Conf. Artif. Intell.*, 2014, pp. 2149–2155.

[40] P. Zhou, Y.-D. Shen, L. Du, F. Ye, and X. Li, "Incremental multi-view spectral clustering," *Knowl.-Based Syst.*, vol. 174, pp. 73–86, Jun. 2019.

[41] Z. Kang, C. Peng, and Q. Cheng, "Twin learning for similarity and clustering: A unified kernel approach," in *Proc. 31st AAAI Conf. Artif. Intell.*, 2017, pp. 2080–2086.

[42] L. Du, X. Li, and Y.-D. Shen, "Robust nonnegative matrix factorization via half-quadratic minimization," in *Proc. IEEE 12th Int. Conf. Data Mining*, Dec. 2012, pp. 201–210.

[43] Z. Kang, C. Peng, Q. Cheng, and Z. Xu, "Unified spectral clustering with optimal graph," in *Proc. AAAI*, 2018, pp. 3366–3373.

[44] S. Zhou, E. Zhu, X. Liu, T. Zheng, Q. Liu, J. Xia, and J. Yin, "Subspace segmentation-based robust multiple kernel clustering," *Inf. Fusion*, vol. 53, pp. 145–154, Jan. 2020.

[45] S. Zhou, X. Liu, M. Li, E. Zhu, L. Liu, C. Zhang, and J. Yin, "Multiple kernel clustering with neighbor-kernel subspace segmentation," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 4, pp. 1351–1362, Apr. 2020.

[46] H. Wang, L. Du, P. Zhou, L. Shi, Y. Qian, and Y.-D. Shen, "Experimental design with multiple kernels," in *Proc. IEEE Int. Conf. Data Mining*, Nov. 2015, pp. 419–428.

[47] P. Zhou, L. Du, L. Shi, H. Wang, and Y. Shen, "Recovery of corrupted multiple kernels for clustering," in *Proc. 24th Int. Joint Conf. Artif. Intell.*, 2015, pp. 4105–4111.

[48] T. Li, Y. Dou, X. Liu, Y. Zhao, and Q. Lv, "Multiple kernel clustering with corrupted kernels," *Neurocomputing*, vol. 267, pp. 447–454, Dec. 2017.

[49] P. Zhou, F. Ye, and L. Du, "Unsupervised robust multiple kernel learning via extracting local and global noises," *IEEE Access*, vol. 7, pp. 34451–34461, 2019.

[50] S. Wang, M. Li, N. Hu, E. Zhu, J. Hu, X. Liu, and J. Yin, "K-means clustering with incomplete data," *IEEE Access*, vol. 7, pp. 69162–69171, 2019.

[51] X. Liu, W. Gao, X. Zhu, M. Li, L. Wang, E. Zhu, T. Liu, M. Kloft, D. Shen, and J. Yin, "Multiple kernel *k*-means with incomplete kernels," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 5, pp. 1191–1204, May 2020.

[52] M. Li, X. Liu, L. Wang, Y. Dou, J. Yin, and E. Zhu, "Multiple kernel clustering with local kernel alignment maximization," in *Proc. 25th Int. Joint Conf. Artif. Intell.*, 2016, pp. 1704–1710.

[53] C. Wang, E. Zhu, X. Liu, L. Gao, J. Yin, and N. Hu, "Multiple kernel clustering with global and local structure alignment," *IEEE Access*, vol. 6, pp. 77911–77920, 2018.

[54] Z. Liu, K. Shi, K. Zhang, W. Ou, and L. Wang, "Discriminative sparse embedding based on adaptive graph for dimension reduction," *Eng. Appl. Artif. Intell.*, vol. 94, Sep. 2020, Art. no. 103758.

[55] C. Peng and Q. Cheng, "Discriminative ridge machine: A classifier for high-dimensional data or imbalanced data," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Jul. 21, 2020, doi: 10.1109/TNNLS.2020.3006877.

[56] Y. Yang, D. Xu, F. Nie, S. Yan, and Y. Zhuang, "Image clustering using local discriminant models and global integration," *IEEE Trans. Image Process.*, vol. 19, no. 10, pp. 2761–2773, Oct. 2010.

[57] J. Ye, Z. Zhao, and M. Wu, "Discriminative K-means for clustering," in *Proc. Adv. Neural Inf. Process. Syst.*, 2007, pp. 1649–1656.

[58] W. Xu and Y. Gong, "Document clustering by concept factorization," in *Proc. 27th Annu. Int. Conf. Res. Develop. Inf. Retr. (SIGIR)*, 2004, pp. 202–209.

[59] Z. Kang, L. Wen, W. Chen, and Z. Xu, "Low-rank kernel learning for graph-based clustering," *Knowl.-Based Syst.*, vol. 163, pp. 510–517, Jan. 2019.

[60] C. Yang, Z. Ren, Q. Sun, M. Wu, M. Yin, and Y. Sun, "Joint correntropy metric weighting and block diagonal regularizer for robust multiple kernel subspace clustering," *Inf. Sci.*, vol. 500, pp. 48–66, Oct. 2019.

[61] C. Peng, Z. Kang, C. Chen, and Q. Cheng, "Nonnegative matrix factorization with local similarity learning," 2019, *arXiv:1907.04150*. [Online]. Available: http://arxiv.org/abs/1907.04150

[62] C. Peng, Y. Chen, Z. Kang, C. Chen, and Q. Cheng, "Robust principal component analysis: A factorization-based approach with linear complexity," *Inf. Sci.*, vol. 513, pp. 581–599, Mar. 2020.

[63] C. H. Q. Ding, T. Li, and M. I. Jordan, "Convex and semi-nonnegative matrix factorizations," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 1, pp. 45–55, Jan. 2010.

LIN MU received the Ph.D. degree in computer software and theory from the Graduate University of Chinese Academy of Sciences. He is currently an Assistant Professor with the Institute of Scientific and Technical Information of China, who has previous experience in artificial intelligence and big data.

HAIYING ZHANG is currently pursuing the master's degree with Shanxi University, Taiyuan, China. Her researches focus on data mining and machine learning algorithms.

LIANG DU (Member, IEEE) received the B.E. degree in software engineering from Wuhan University, in 2007, and the Ph.D. degree in computer science from the Institute of Software, University of Chinese Academy of Sciences, in 2013. He is currently a Lecturer with Shanxi University. Prior to that, he was an Assistant Researcher with the State Key Laboratory of Computer Science, Institute of Software, Chinese Academy of Sciences. In July 2013 and July 2014, he was a Software Engineer with Alibaba Group. He is particularly interested in the following topics: clustering with noise and heterogeneous data, ranking for feature selection, active learning, and document summarization. He has published more than 40 papers in top conferences and journals, including KDD, IJCAI, AAAI, ICDM, TNNLS, TKDE, SDM, and CIKM.

JIE GUI received the Ph.D. degree in management science and engineering from the Beijing Institute of Technology, China, in 2007. In 2007, she joined the Institute of Scientific and Technical Information of China (ISTIC), Beijing, where she is currently an Associate Researcher. Her research interests include S&T innovation management and patent data mining.

AIDAN LI received the B.S. degree in computer science and technology from Tianjin University, China, the M.S. degree in engineering management from the New Jersey Institute of Technology, USA, and the Ph.D. degree in management science and engineering from the Beijing Institute of Technology, China. She is currently an Assistant Professor with the Institute of Scientific and Technical Information of China. She has previous experience in knowledge management and data services.

XI ZHANG received the bachelor's degree in financial management from Beijing Wuzi University, China, and the master's degree in library and information studies from Heilongjiang University, China. She is currently an Assistant Professor with the Institute of Scientific and Technical Information of China. She has previous experience in management of sci-tech information and data services.