Localized Incomplete Multiple Kernel *k*-Means With Matrix-Induced Regularization

Miaomiao Li, Jingyuan Xia, Huiying Xu, Qing Liao, *Member, IEEE*, Xinzhong Zhu, *Member, IEEE*, and Xinwang Liu, *Senior Member, IEEE*

***Abstract*—Localized incomplete multiple kernel *k*-means (LI-MKKM) is recently put forward to boost the clustering accuracy via optimally utilizing a quantity of prespecified incomplete base kernel matrices. Despite achieving significant achievement in a variety of applications, we find out that LI-MKKM does not sufficiently consider the diversity and the complementary of the base kernels. This could make the imputation of incomplete kernels less effective, and vice versa degrades on the subsequent clustering. To tackle these problems, an improved LI-MKKM, called LI-MKKM with matrix-induced regularization (LI-MKKM-MR), is proposed by incorporating a matrix-induced regularization term to handle the correlation among base kernels. The incorporated regularization term is beneficial to decrease the probability of simultaneously selecting two similar kernels and increase the probability of selecting two kernels with moderate differences. After that, we establish a three-step iterative algorithm to solve the corresponding optimization objective and analyze its convergence. Moreover, we theoretically show that the local kernel alignment is a special case of its global one with normalizing each base kernel matrices. Based on the above observation, the generalization error bound of the proposed algorithm is derived to theoretically justify its effectiveness. Finally, extensive experiments on several public datasets have been conducted to evaluate the clustering performance of the LI-MKKM-MR. As indicated, the experimental results have demonstrated that our algorithm consistently outperforms the state-of-the-art ones, verifying the superior performance of the proposed algorithm.**

***Index Terms*—Incomplete kernel learning, multiple kernel clustering (MKC), multiple view learning.**

Manuscript received January 13, 2021; revised September 14, 2021; accepted November 6, 2021. This work was supported in part by the National Key Research and Development Program of China under Project 2020AAA0107100; in part by the Natural Science Foundation of China under Project 61922088, Project 61773392, Project 61872377, and Project 61976196. This article was recommended by Associate Editor A. F. Skarmeta Gómez. *(Miaomiao Li, Jingyuan Xia, and Huiying Xu contributed equally to this work.) (Corresponding authors: Miaomiao Li; Xinzhong Zhu.)*

Miaomiao Li is with the College of Electronic Information and Electrical Engineering, Changsha University, Changsha 410073, China (e-mail: miaomiaolinudt@gmail.com).

Jingyuan Xia is with the Department of Electric and Electronic Engineering, Imperial College London, London SW72AZ, U.K.

Huiying Xu is with the College of Mathematics and Computer Science, Zhejiang Normal University, Jinhua 321004, China (e-mail: xyh@zjnu.edu.cn).

Qing Liao is with the Department of Computer Science and Technology, Harbin Institute of Technology (Shenzhen), Shenzhen 518055, China.

Xinzhong Zhu is with the College of Mathematics and Computer Science, Zhejiang Normal University, Jinhua 321004, China, and also with the Department of Artificial Intelligence, Research Institute of Ningbo Cixing Company Ltd., Ningbo 315336, China (e-mail: zxz@zjnu.edu.cn).

Xinwang Liu is with the School of Computer, National University of Defense Technology, Changsha 410073, China.

This article has supplementary material provided by the authors and color versions of one or more figures available at https://doi.org/10.1109/TCYB.2021.3126727.

Digital Object Identifier 10.1109/TCYB.2021.3126727

I. Introduction

MULTIPLE kernel clustering (MKC) [1]–[8] sufficiently integrates a number of precalculated base kernel matrices to group samples into clusters, where similar samples are in the same cluster while dissimilar ones are partitioned into different ones. MKC has attracted much attention of the data mining researchers and has been widely studied in recent years [9]–[17]. The seminal work in [9] extends the multiple kernel learning from supervised learning to unsupervised learning and proposes a margin-based MKC algorithm. It jointly optimizes the optimal kernel, the maximum margin hyperplane, and the optimal clustering labels. The widely used kernel *k*-means method has been extended in [18] for clustering analysis, where an optimal kernel is learned from multiple data sources. Similarly, the work in [12] extends the existing multiple kernel *k*-means (MKKM) algorithm by designing a localized MKKM one in order to well utilize the characteristics of each individual sample. To enhance the robustness of the existing MKKM algorithms to noisy data, Du *et al.* [13] proposed a robust MKKM algorithm by substituting the widely adopted squared error in the existing *k*-means with an $\ell_{2,1}$-norm one, and simultaneously optimized the best combination of kernels. To increase the diversity and decrease the redundancy of the selected base kernels, the recent work in [14] extends the existing MKKM algorithms by designing a matrix-induced regularization term to sufficiently explore the correlation among the prespecified base kernels. More recently, an optimal neighborhood kernel clustering (ONKC) algorithm is proposed in [19], where the representability of the optimal kernel to learn is largely boosted and the negotiation between kernel learning and clustering is also reinforced. The aforementioned MKC algorithms have been applied into many cases and reached a superior performance [15], [20]–[23].

As observed, these MKC algorithms share a common assumption: all the prespecified base kernels are complete. Nevertheless, in some real-world applications, such as image fusion [24], image retrieval [25], and document/video analysis [26], some views of a sample are usually not collected due to various reasons [27], [28]. To address this issue, the work in the literature proposes to first impute the missing elements in base kernel matrices with imputation methods and then performs the existing MKC on these imputed kernels. Several commonly used filling methods include zero-filling, mean value filling, *k*-nearest-neighbor filling (KNN), expectation-maximization (EM) filling [29], as well as several recently proposed to matrix imputation [30]–[33].

One disadvantage existing in the aforementioned "two-stage" algorithms is that the imputation is separated from the subsequent clustering. As a result, this may not be conducive to mutual negotiation between the imputation and clustering to reach the best performance. To overcome the above issue, the more recent literature [34]–[36] advocates to unify the learning procedure of imputation and clustering into a common framework, with the aim to learn an optimal imputation that best severe for the clustering tasks.

Although demonstrating superior clustering results in several practical applications, we find that these works do not sufficiently consider the redundancy and diversity among prespecified kernel matrices when performing incomplete MKC. This could lead to high redundancy and low diversity among the selected kernels [14], making the utilization ratio of these base kernel matrices insufficient and conversely decreasing the accuracy of clustering tasks. In our work, a localized incomplete MKKM with matrix-induced regularization (LI-MKKM-MR) is proposed to address the above-mentioned issue. By incorporating matrix-induced regularization, LI-MKKM-MR is able to avoid selecting two similar kernel matrices simultaneously and increase the probability of selecting two kernel matrices with large diversity, making the base kernels better utilized for clustering. In addition, it inherits the advantage of localized incomplete multiple kernel *k*-means (LI-MKKM) which only requires that the similarity of each sample to its top *k*-nearest neighbors be optimally aligned with the corresponding patch of the entire ideal similarity. This is helpful for LI-MKKM-MR to pay more attention on closer pairwise sample similarities that shall be put together, and prevents involving unreliable similarity evaluation for farther sample pairs. Furthermore, a three-step iterative optimization algorithm is designed to solve the corresponding optimization objective and its convergence has also been analyzed. After that, the generalization error bound of the clustering algorithm is derived, which theoretically guarantees its effectiveness. Comprehensive experiments on several public datasets have been conducted to evaluate the clustering performance of the proposed LI-MKKM-MR. As demonstrated, LI-MKKM-MR significantly and consistently outperforms the existing two-step-based algorithms and the newly proposed algorithm [36]. Extensive experimental results have demonstrated the superiority of involving the matrix-induced regularization.

To summarize, this work makes the following major contributions.

1) This is the first attempt to identify the kernel redundancy problem in *incomplete* MKC. We then introduce a new algorithm to improve LI-MKKM by integrating matrix-induced regularization to select low-redundant and high-diverse kernel matrices and carefully establish three-step iterative algorithm to solve the corresponding optimization objective.
2) We build the theoretical connection between global and local kernel alignment criteria, then we further derive the generalization error bound of the proposed LI-MKKM-MR, which theoretically justifies its effectiveness.
3) Comprehensive experiments on ten public datasets have demonstrated that our LI-MKKM-MR achieves the state-of-the-art performance compared with the existing advanced algorithms. This considerably verifies our identification of the aforementioned issue and the effectiveness of our solution.

Finally, we clarify the differences between LI-MKKM-MR and several recently proposed related work [14], [35]. The differences between LI-MKKM [35] and LI-MKKM-MR can be summarized from the following three aspects.

1) LI-MKKM [35] does not sufficiently consider the diversity and the complementarity of these incomplete base kernels. This could make the imputation of incomplete kernels less effective, and incur the adverse effect on the subsequent clustering. Differently, LI-MKKM-MR is proposed by incorporating matrix-induced regularization, which is helpful to reduce the probability of simultaneously selecting two similar kernels and increase the probability of selecting two kernels with moderate differences, making the base kernels better utilized for clustering.
2) Compared to LI-MKKM [35], LI-MKKM-MR provides the generalization error analysis, which measures the clustering performance of the learned clusters in the training procedure on unseen samples. This theoretically justifies the effectiveness of the proposed LI-MKKM-MR.
3) As observed from the experimental results in Section IV, LI-MKKM-MR significantly improves the clustering performance of LI-MKKM [35] in various benchmark datasets, which well validates our identification of the aforementioned issue in LI-MKKM and the effectiveness of our solution.

We then summarize the differences between [14] and our work from the following aspects. In [14], matrix-induced regularization is proposed to solve the kernel redundancy in MKC. However, it cannot effectively solve MKC with incomplete kernels. Differently, the proposed LI-MKKM-MR makes the first attempt to identify the kernel redundancy problem in *incomplete MKC*, proposes an effective solution, and conducts comprehensive experiments to validate our identification of this issue and the superiority of our algorithm.

II. Related Work

In this part, we mainly introduce the methods of MKKM clustering, MKKM with incomplete kernels (MKKM-IK), and its localized variant. Before introducing these algorithms, we present all notations which will be used in the following in Table I.

TABLE I
NOTATIONS SUMMARY

$\{\mathbf{x}_i\}_{i=1}^n$	n training samples
k	number of clusters
τ	ratio of the nearest neighbors
$\boldsymbol{\gamma} = [\gamma_1, \cdots, \gamma_m]^\top$	kernel weights
$\kappa_p(\cdot, \cdot)$	the p-th kernel function
$\phi_p(\cdot)$	feature mapping corresponding to $\kappa_p(\cdot, \cdot)$
$\phi_{\boldsymbol{\gamma}}(\cdot)$	feature mapping corresponding to $\kappa_{\boldsymbol{\gamma}}(\cdot, \cdot)$
$\{\mathbf{K}_p\}_{p=1}^m$	m base kernel matrices
$\mathbf{e}_p$	observed sample indices of $\mathbf{K}_p$
$\mathbf{H}$	partition matrix
$\mathbf{K}_p^{(dd)}$	sub-matrix of $\mathbf{K}_p$ for observed samples
$\mathbf{U}^{(i)} \in \{0,1\}^{n \times \mathrm{round}(n*\tau)}$	neighborhood indication matrix of $\mathbf{x}_i$
$\mathbf{M}$	correlation matrix among m base kernels
$\hat{\mathbf{C}} = [\hat{\mathbf{C}}_1, \cdots, \hat{\mathbf{C}}_k]$	the learned k centroids

A. Multiple Kernel k-Means

Let $\{\mathbf{x}_i\}_{i=1}^n \subseteq \mathcal{X}$ be n training samples, and $\phi_p(\cdot) : \mathbf{x} \in \mathcal{X} \mapsto \mathcal{H}_p$, $\mathbf{x}$ are mapped onto a reproducing kernel Hilbert space $\mathcal{H}_p$ $(1 \leq p \leq m)$ by the pth feature. Each sample in MKC is represented by $\phi_{\boldsymbol{\gamma}}(\mathbf{x}) = [\gamma_1 \phi_1^\top(\mathbf{x}), \ldots, \gamma_m \phi_m^\top(\mathbf{x})]^\top$, where $\boldsymbol{\gamma} = [\gamma_1, \ldots, \gamma_m]^\top$ represents the weights of m prespecified base kernel functions $\{\kappa_p(\cdot,\cdot)\}_{p=1}^m$. These kernel weights will be adaptively adjusted during MKC. Under the aforementioned definition of $\phi_{\boldsymbol{\gamma}}(\mathbf{x})$, the corresponding kernel function can be expressed as follows:

$$\kappa_{\boldsymbol{\gamma}}\left(\mathbf{x}_i, \mathbf{x}_j\right) = \phi_{\boldsymbol{\gamma}}^\top(\mathbf{x}_i)\phi_{\boldsymbol{\gamma}}(\mathbf{x}_j) = \sum_{p=1}^m \gamma_p^2 \kappa_p\left(\mathbf{x}_i, \mathbf{x}_j\right). \tag{1}$$

One can calculate a kernel matrix $\mathbf{K}_{\boldsymbol{\gamma}}$ on training samples $\{\mathbf{x}_i\}_{i=1}^n$ with the kernel function defined in (1). As a result, the objective of MKKM with $\mathbf{K}_{\boldsymbol{\gamma}}$ is formulated as

$$\begin{aligned} &\min_{\mathbf{H}, \boldsymbol{\gamma}} \ \mathrm{Tr}\Big(\mathbf{K}_{\boldsymbol{\gamma}}\Big(\mathbf{I}_n - \mathbf{H}\mathbf{H}^\top\Big)\Big) \\ &\text{s.t. } \ \mathbf{H}^\top\mathbf{H} = \mathbf{I}_k, \ \ \boldsymbol{\gamma}^\top \mathbf{1}_m = 1, \ \gamma_p \geq 0 \quad \forall p \end{aligned} \tag{2}$$

where $\mathbf{H} \in \mathbb{R}^{n \times k}$ is a soft version of the cluster assignment matrix, and $\mathbf{I}_k$ is a $k \times k$ identity matrix. Alternately updating $\mathbf{H}$ and $\boldsymbol{\gamma}$ can optimize (2).

Optimizing $\mathbf{H}$ *With Fixed* $\boldsymbol{\gamma}$*:* With $\boldsymbol{\gamma}$ fixed, the optimization in (2) toward $\mathbf{H}$ is exactly the traditional kernel k-means presented in

$$\max_{\mathbf{H}} \ \mathrm{Tr}\Big(\mathbf{H}^\top \mathbf{K}_{\boldsymbol{\gamma}} \mathbf{H}\Big) \text{ s.t. } \mathbf{H} \in \mathbb{R}^{n \times k}, \mathbf{H}^\top\mathbf{H} = \mathbf{I}_k. \tag{3}$$

The optimal $\mathbf{H}$ in (3) consists of k eigenvectors corresponding to the top-k eigenvalues of $\mathbf{K}_{\boldsymbol{\gamma}}$ [37].

Optimizing $\boldsymbol{\gamma}$ *With Fixed* $\mathbf{H}$*:* With $\mathbf{H}$ fixed, the equivalent form of optimization in (2) with regard to $\boldsymbol{\gamma}$ is as follows:

$$\min_{\boldsymbol{\gamma}} \ \sum_{p=1}^m \gamma_p^2 \mathrm{Tr}\Big(\mathbf{K}_p\Big(\mathbf{I}_n - \mathbf{H}\mathbf{H}^\top\Big)\Big) \text{ s.t. } \boldsymbol{\gamma}^\top \mathbf{1}_m = 1, \ \gamma_p \geq 0 \tag{4}$$

which has a closed-form solution.

B. MKKM With Incomplete Kernels

MKKM has recently been extended to handle incomplete MKC in [34] and [36]. Previous algorithms first manage to impute the incomplete kernel matrices and then apply the existing MKKM on the imputed kernel matrices. In contrast, they propose to unify the learning process of imputation and clustering into a common learning framework and establish an effective optimization algorithm to optimize each of them alternately. In MKKM-IK, the clustering procedure provides a guidance for the imputation of the incomplete base kernel matrices, and the clustering is further enhanced by the imputed kernels. Both procedures are alternated performed until achieving optimal results. The above idea can be achieved as follows:

$$\begin{aligned} &\min_{\mathbf{H}, \boldsymbol{\gamma}, \{\mathbf{K}_p\}_{p=1}^m} \ \mathrm{Tr}\Big(\mathbf{K}_{\boldsymbol{\gamma}}(\mathbf{I}_n - \mathbf{H}\mathbf{H}^\top)\Big) \\ &\text{s.t. } \ \mathbf{H} \in \mathbb{R}^{n \times k}, \mathbf{H}^\top\mathbf{H} = \mathbf{I}_k \\ &\qquad \boldsymbol{\gamma}^\top \mathbf{1}_m = 1, \gamma_p \geq 0 \\ &\qquad \mathbf{K}_p(\mathbf{e}_p, \mathbf{e}_p) = \mathbf{K}_p^{(dd)}, \ \mathbf{K}_p \succeq 0 \quad \forall p \end{aligned} \tag{5}$$

where $\mathbf{e}_p$ $(1 \leq p \leq m)$ denotes the sample indices, the p-th view is observed, and $\mathbf{K}_p^{(dd)}$ denotes the kernel submatrix. Note that we impose the constraint $\mathbf{K}_p(\mathbf{e}_p, \mathbf{e}_p) = \mathbf{K}_p^{(dd)}$ to make the known entries of $\mathbf{K}_p$ kept unchanged during the learning course. The imputation of incomplete kernels can be regarded as a by-product of learning, because the ultimate goal of (5) is clustering.

A trilevel optimization strategy developed in [34] develops to solve (5) alternately.

Optimizing $\mathbf{H}$ *With* $\boldsymbol{\gamma}$ *and* $\{\mathbf{K}_p\}_{p=1}^m$ *Fixed:* Given $\boldsymbol{\gamma}$ and $\{\mathbf{K}_p\}_{p=1}^m$, the optimization in (5) with respect to $\mathbf{H}$ is equivalent to a kernel k-means problem solved by (3).

Optimizing $\{\mathbf{K}_p\}_{p=1}^m$ *With* $\boldsymbol{\gamma}$ *and* $\mathbf{H}$ *Fixed:* Given $\boldsymbol{\gamma}$ and $\mathbf{H}$, (5) toward each $\mathbf{K}_p$ is equivalently reformulated as follows:

$$\begin{aligned} &\min_{\mathbf{K}_p} \ \mathrm{Tr}\Big(\mathbf{K}_p(\mathbf{I}_n - \mathbf{H}\mathbf{H}^\top)\Big) \\ &\text{s.t. } \ \mathbf{K}_p\big(\mathbf{e}_p, \mathbf{e}_p\big) = \mathbf{K}_p^{(dd)}, \ \mathbf{K}_p \succeq 0. \end{aligned} \tag{6}$$

It is proven in [34] that the optimal $\mathbf{K}_p$ in (6) has the closed-form solution as in (7), shown at the bottom of the page, where $\mathbf{Z} = \mathbf{I}_n - \mathbf{H}\mathbf{H}^\top$ and taking the elements of $\mathbf{Z}$ corresponding to the observed and unobserved sample indices can construct $\mathbf{Z}^{(dm)}$. For more details, refer to [34].

Optimizing $\boldsymbol{\gamma}$ *With* $\mathbf{H}$ *and* $\{\mathbf{K}_p\}_{p=1}^m$ *Fixed:* Given $\mathbf{H}$ and $\{\mathbf{K}_p\}_{p=1}^m$, (5) with respect to $\boldsymbol{\gamma}$ reduces to a quadratic programming (QP) with linear constraints.

C. Localized Incomplete MKKM

Although it is ingenious to unify clustering and imputation into one learning process, which is achieved by *globally* maximizing the alignment between the optimal kernel matrix $\mathbf{K}_{\boldsymbol{\gamma}}$

$$\mathbf{K}_p = \begin{bmatrix} \mathbf{K}_p^{(dd)} & -\mathbf{K}_p^{(dd)}\mathbf{Z}^{(dm)}(\mathbf{Z}^{(mm)})^{-1} \\ -(\mathbf{Z}^{(mm)})^{-1}\mathbf{Z}^{(dm)\top}\mathbf{K}_p^{(dd)} & (\mathbf{Z}^{(mm)})^{-1}\mathbf{Z}^{(dm)\top}\mathbf{K}_p^{(dd)}\mathbf{Z}^{(dm)}(\mathbf{Z}^{(mm)})^{-1} \end{bmatrix} \tag{7}$$

and the ideal matrix $\mathbf{H}\mathbf{H}^\top$, as presented in (2). This criterion does not take full advantage of the local distribution of data, and requires that all paired samples, whether closer or farther, should be consistent with the ideal similarity without distinction.

Instead of calculating the alignment between the optimal kernel and the idea matrix in a global manner as in (5), localized incomplete MKKM (LI-MKKM) [35] is proposed to utilize the local structure among data by only requiring the similarity of each sample to align with its nearest neighbors. Specifically, the objective function of LI-MKKM is as follows:

$$\begin{aligned}
\min_{\boldsymbol{\gamma},\, \{\mathbf{K}_p\}_{p=1}^m, \mathbf{H}} \quad & \sum_{i=1}^{n} \mathrm{Tr}\Big(\mathbf{K}_{\boldsymbol{\gamma}}\Big(\mathbf{A}^{(i)} - \mathbf{A}^{(i)}\mathbf{H}\mathbf{H}^\top\mathbf{A}^{(i)}\Big)\Big) \\
\text{s.t.} \quad & \mathbf{H} \in \mathbb{R}^{n\times k},\ \mathbf{H}^\top\mathbf{H} = \mathbf{I}_k,\ \boldsymbol{\gamma}^\top\mathbf{1}_m = 1,\ \gamma_p \geq 0 \\
& \mathbf{K}_p\big(\mathbf{e}_p, \mathbf{e}_p\big) = \mathbf{K}_p^{(dd)},\ \mathbf{K}_p \succeq 0 \quad \forall p
\end{aligned} \tag{8}$$

where $\mathbf{A}^{(i)} = \mathbf{U}^{(i)}\mathbf{U}^{(i)\top}$ with $\mathbf{U}^{(i)} \in \{0, 1\}^{n\times \mathrm{round}(n*\tau)}$ ($1 \leq i \leq n$) denoting the neighborhood index matrix of the ith sample. $\mathbf{U}_{jv}^{(i)} = 1$ represents that $\mathbf{x}_j$ is the vth nearest neighbor of $\mathbf{x}_i$, where $1 \leq v \leq \mathrm{round}(n*\tau)$ and τ is the ratio of the nearest neighbors.

Similar to [34], the work in [35] develops a tristep optimization algorithm to solve (8) and theoretically proves its convergence. Refer to [35] for more details.

III. Localized Incomplete Multiple Kernel k-Means With Matrix-Induced Regularization

A. Formulation

Although aligning the optimal kernel with the ideal similarity locally can improve the clustering performance, LI-MKKM dose not explicitly take the correlation among base kernels into account. This would prevent these incomplete base kernels from being well utilized. To overcome this problem, we propose an improved algorithm based on LI-MKKM via introducing matrix-induced regularization $\boldsymbol{\gamma}^\top\mathbf{M}\boldsymbol{\gamma}$ to decrease the redundancy and enhance the diversity of the selected base kernels, where M_{pq} measures the correlation between $\mathbf{K}_p$ and $\mathbf{K}_q$. By integrating this regularization into (8), the following objective is obtained:

$$\begin{aligned}
\min_{\boldsymbol{\gamma},\, \{\mathbf{K}_p\}_{p=1}^m, \mathbf{H}} \quad & \sum_{i=1}^{n} \mathrm{Tr}\Big(\mathbf{K}_{\boldsymbol{\gamma}}(\mathbf{A}^{(i)} - \mathbf{A}^{(i)}\mathbf{H}\mathbf{H}^\top\mathbf{A}^{(i)})\Big) + \frac{\lambda}{2}\boldsymbol{\gamma}^\top\mathbf{M}\boldsymbol{\gamma} \\
\text{s.t.} \quad & \mathbf{H} \in \mathbb{R}^{n\times k},\ \mathbf{H}^\top\mathbf{H} = \mathbf{I}_k \\
& \boldsymbol{\gamma}^\top\mathbf{1}_m = 1,\ \gamma_p \geq 0 \\
& \mathbf{K}_p(\mathbf{e}_p, \mathbf{e}_p) = \mathbf{K}_p^{(dd)},\ \mathbf{K}_p \succeq 0 \quad \forall p
\end{aligned} \tag{9}$$

where λ is a hyper-parameter to balance the regularization on kernel weights and the loss of local kernel k-means.

In this work, we adopt $M_{pq} = \mathrm{Tr}(\mathbf{K}_p\mathbf{K}_q)$ to measure the correlation between $\mathbf{K}_p$ and $\mathbf{K}_q$. On one hand, the incorporation of $\boldsymbol{\gamma}^\top\mathbf{M}\boldsymbol{\gamma}$ is helpful for well utilizing the base kernels, which is utilized to boost the clustering performance. On the other hand, it makes the resultant optimization more challenging since the optimization on each $\mathbf{K}_p$ is a quadratic semi-defined programming, whose computational cost is intensive and this prevents it from being applied to practical applications. To reduce the computation overhead of (9), we propose to approximate M_{pq} by $\tilde{M}_{pq} = \mathrm{Tr}(\mathbf{K}_p^{(0)}\mathbf{K}_q^{(0)})$ and keep it unchanged during the learning course, where $\mathbf{K}_p^{(0)}$ is an initial imputation of $\mathbf{K}_p$. By substituting $\mathbf{M}$ with $\tilde{\mathbf{M}}$, the objective function of the proposed LI-MKKM-MR can be expressed as follows:

$$\begin{aligned}
\min_{\boldsymbol{\gamma},\, \{\mathbf{K}_p\}_{p=1}^m, \mathbf{H}} \quad & \sum_{i=1}^{n} \mathrm{Tr}\Big(\mathbf{K}_{\boldsymbol{\gamma}}\Big(\mathbf{A}^{(i)} - \mathbf{A}^{(i)}\mathbf{H}\mathbf{H}^\top\mathbf{A}^{(i)}\Big)\Big) + \frac{\lambda}{2}\boldsymbol{\gamma}^\top\tilde{\mathbf{M}}\boldsymbol{\gamma} \\
\text{s.t.} \quad & \mathbf{H} \in \mathbb{R}^{n\times k},\ \mathbf{H}^\top\mathbf{H} = \mathbf{I}_k \\
& \boldsymbol{\gamma}^\top\mathbf{1}_m = 1,\ \gamma_p \geq 0 \\
& \mathbf{K}_p(\mathbf{e}_p, \mathbf{e}_p) = \mathbf{K}_p^{(dd)},\ \mathbf{K}_p \succeq 0 \quad \forall p.
\end{aligned} \tag{10}$$

It is reasonable to measure the correlation of pairwise kernels via observed similarity. Consequently, the approximation $\tilde{\mathbf{M}}$ can be regarded as a prior of $\mathbf{M}$. Also, although this approximation is simple, its advantages are three-folds. First, it fulfills our requirement on the kernel coefficients to enhance the diversity and decrease the redundancy. Second, it simplifies the optimization on $\{\mathbf{K}_p\}_{p=1}^m$, making it admit a closed-form solution. This significantly increases the computational cost. Third, the effectiveness of the proposed approximation can be demonstrated by experiments.

Although the matrix-induced regularization may be exploited in other related aspects, such as MKC [14], this is the first work in literature to study the regularization on incomplete MKC and design a reasonable approximation for the convenience of computation. Moreover, this would trigger more research on incomplete MKC, such as designing more informative $\mathbf{M}$, updating $\mathbf{M}$ with learned kernel weights and the imputation at each iteration, to name just a few. More importantly, our experimental study shows that the incorporation of matrix-induced regularization helps to utilize the incomplete kernels, leading to significantly improvement on clustering performance. This makes the proposed algorithm a good choice in real-world applications, such as cancer biology [12], analysis of multiple heterogeneous neuroimaging data [38], and Alzheimer's disease diagnosis [39]. In the following, we develop a tristep optimization strategy to solve it alternately in the following parts.

B. Alternate Optimization of LI-MKKM-MR

Optimizing $\mathbf{H}$ With $\boldsymbol{\gamma}$ and $\{\mathbf{K}_p\}_{p=1}^m$ Fixed: Given $\boldsymbol{\gamma}$ and $\{\mathbf{K}_p\}_{p=1}^m$, the optimization objective w.r.t $\mathbf{H}$ in (10) redefines to

$$\begin{aligned}
\max_{\mathbf{H}} \quad & \mathrm{Tr}\left(\mathbf{H}^\top \sum_{i=1}^{n}\Big(\mathbf{A}^{(i)}\mathbf{K}_{\boldsymbol{\gamma}}\mathbf{A}^{(i)}\Big)\mathbf{H}\right) \\
\text{s.t.} \quad & \mathbf{H} \in \mathbb{R}^{n\times k},\ \mathbf{H}^\top\mathbf{H} = \mathbf{I}_k
\end{aligned} \tag{11}$$

which is transformed into a classical kernel k-means-based optimization objective and can be conveniently tackled by the existing public toolkit.

Optimizing $\{\mathbf{K}_p\}_{p=1}^m$ With $\boldsymbol{\gamma}$ and $\mathbf{H}$ Fixed: Given $\boldsymbol{\gamma}$ and $\mathbf{H}$, the optimization objective w.r.t $\{\mathbf{K}_p\}_{p=1}^m$ in (10) can be formulated as

$$\min_{\{\mathbf{K}_p\}_{p=1}^m} \sum_{p=1}^m \gamma_p^2 \mathrm{Tr}\left(\mathbf{K}_p \sum_{i=1}^n \mathrm{Tr}\left(\mathbf{A}^{(i)} - \mathbf{A}^{(i)}\mathbf{H}\mathbf{H}^\top \mathbf{A}^{(i)}\right)\right)$$
$$\text{s.t. } \mathbf{K}_p(\mathbf{e}_p, \mathbf{e}_p) = \mathbf{K}_p^{(dd)},\ \mathbf{K}_p \succeq 0 \quad \forall p. \tag{12}$$

It is difficult to solve the optimization problem in (12) since there are multiple kernel matrices to be optimized simultaneously. By cautiously analyzing the optimization, we observe that: 1) each kernel matrix $\mathbf{K}_p$ has its own separate constraint and 2) the objective in (12) is a sum generated by calculating $\mathbf{K}_p$. As a result, (12) can be reformulated as m uncorrelated subobjectives equivalently, as shown in the following:

$$\min_{\mathbf{K}_p} \ \mathrm{Tr}(\mathbf{K}_p\mathbf{Q})$$
$$\text{s.t. } \mathbf{K}_p(\mathbf{e}_p, \mathbf{e}_p) = \mathbf{K}_p^{(dd)},\ \mathbf{K}_p \succeq 0 \tag{13}$$

where $\mathbf{Q} = \sum_{i=1}^n (\mathbf{A}^{(i)} - \mathbf{A}^{(i)}\mathbf{H}\mathbf{H}^\top\mathbf{A}^{(i)})$.

It seems that directly solving (13) is difficult because of the equality and PSD constraints imposed on $\mathbf{K}_p$. By following [35], we parameterize each $\mathbf{K}_p$ as:

$$\mathbf{K}_p = \begin{bmatrix} \mathbf{K}_p^{(dd)} & \mathbf{K}_p^{(dd)}\mathbf{Z}_p \\ \mathbf{Z}_p^\top \mathbf{K}_p^{(dd)} & \mathbf{Z}_p^\top \mathbf{K}_p^{(dd)}\mathbf{Z}_p \end{bmatrix} \tag{14}$$

where $\mathbf{Z}_p \in \mathbb{R}^{d\times m}$. d and m refer to the number of observed samples and unobserved ones, respectively. With (14), we assume that the observed ones represent the missing kernel entries. It is shown in [35] that $\mathbf{K}_p$ in (14) automatically satisfies both constraints after this parametrization.

Based on the parametrization in (14), the constrained optimization in (13) is equivalent to

$$\min_{\mathbf{Z}_p} \ \mathrm{Tr}\left(\begin{bmatrix} \mathbf{K}_p^{(dd)} & \mathbf{K}_p^{(dd)}\mathbf{Z}_p \\ \mathbf{Z}_p^\top \mathbf{K}_p^{(dd)} & \mathbf{Z}_p^\top \mathbf{K}_p^{(dd)}\mathbf{Z}_p \end{bmatrix}\begin{bmatrix} \mathbf{Q}^{(dd)} & \mathbf{Q}^{(dm)} \\ \mathbf{Q}^{(dm)\top} & \mathbf{Q}^{(mm)} \end{bmatrix}\right) \tag{15}$$

where $\mathbf{Q}$ is decomposed into the following submatrices $\begin{bmatrix} \mathbf{Q}^{(dd)} & \mathbf{Q}^{(dm)} \\ \mathbf{Q}^{(dm)\top} & \mathbf{Q}^{(mm)} \end{bmatrix}$.

To minimize (15), we take its derivative with respect to $\mathbf{Z}_p$ and let it vanish, leading to

$$\mathbf{Z}_p = -\mathbf{Q}^{(dm)}\left(\mathbf{Q}^{(mm)}\right)^{-1}. \tag{16}$$

As a result, we obtain an analytical solution for the optimal $\mathbf{K}_p$ by substituting $\mathbf{Z}_p$ in (16) into (14). As seen, (13) provides a guidance for the imputation of each base kernel by exploring the data structure in a local manner. Specifically, it locally estimates the alignment between the similarity of each sample and its τ-nearest neighbors with the corresponding ideal matrix. This enables the proposed algorithm to better utilize the intracluster variations among samples. Therefore, the clustering performance could be improved, mainly attributing to an effective incomplete kernels imputation measure.

Optimizing $\boldsymbol{\gamma}$ With $\{\mathbf{K}_p\}_{p=1}^m$ and $\mathbf{H}$ Fixed: Given $\{\mathbf{K}_p\}_{p=1}^m$ and $\mathbf{H}$, it is easy to present that (10) w.r.t. $\boldsymbol{\gamma}$ is as follows:

$$\min_{\boldsymbol{\gamma}} \ \frac{1}{2}\boldsymbol{\gamma}^\top\left(2\mathbf{W} + \lambda\tilde{\mathbf{M}}\right)\boldsymbol{\gamma}$$
$$\text{s.t. } \boldsymbol{\gamma}^\top\mathbf{1}_m = 1,\ \gamma_p \geq 0 \tag{17}$$

Algorithm 1 Proposed LI-MKKM-MR

Input: $\{\mathbf{K}_p^{(dd)}\}_{p=1}^m$, $\{\mathbf{e}_p\}_{p=1}^m$, k, τ, λ and ϵ_0.
Output: $\mathbf{H}$, $\boldsymbol{\gamma}$ and $\{\mathbf{K}_p\}_{p=1}^m$.
Initialize $\boldsymbol{\gamma}^{(0)} = \mathbf{1}_m/m$, $\{\mathbf{K}_p^{(0)}\}_{p=1}^m$ and $t = 1$.
Generate $\mathbf{U}^{(i)}$ for i-th samples $(1 \leq i \leq n)$ by $\mathbf{K}_{\boldsymbol{\gamma}^{(0)}}$.
Calculate $\mathbf{A}^{(i)} = \mathbf{U}^{(i)}\mathbf{U}^{(i)\top}$ for i-th samples $(1 \leq i \leq n)$.
repeat
  $\mathbf{K}_{\boldsymbol{\gamma}^{(t)}} = \sum_{p=1}^m \left(\gamma_p^{(t-1)}\right)^2 \mathbf{K}_p^{(t-1)}$.
  Update $\mathbf{H}^{(t)}$ by solving Eq. (11) with $\mathbf{K}_{\boldsymbol{\gamma}^{(t)}}$.
  Update $\{\mathbf{K}_p^{(t)}\}_{p=1}^m$ with $\mathbf{H}^{(t)}$ by Eq. (13).
  Update $\boldsymbol{\gamma}^{(t)}$ by solving Eq. (17) with $\mathbf{H}^{(t)}$ and $\{\mathbf{K}_p^{(t)}\}_{p=1}^m$.
  $t = t + 1$.
until $\left(\mathrm{obj}^{(t-1)} - \mathrm{obj}^{(t)}\right)/\mathrm{obj}^{(t)} \leq \epsilon_0$

where $\mathbf{W} = \mathrm{diag}([\mathrm{Tr}(\mathbf{K}_1\mathbf{Q}), \ldots, \mathrm{Tr}(\mathbf{K}_m\mathbf{Q})])$. Theorem 1 in the following indicates that $\mathbf{W}$ is PSD.

Theorem 1: The Hessian matrix $2\mathbf{W} + \lambda\tilde{\mathbf{M}}$ in (17) is a symmetric PSD matrix.

Proof: By defining $\mathbf{H} = [\mathbf{h}_1, \ldots, \mathbf{h}_k]$, we can find out that $\mathbf{H}\mathbf{H}^\top\mathbf{h}_c = \mathbf{h}_c (1 \leq c \leq k)$ since $\mathbf{H}^\top\mathbf{H} = \mathbf{I}_k$. This indicates that $\mathbf{H}\mathbf{H}^\top$ has k eigenvalue with 1. Besides, its rank does not exceed k. This means that its has $n-k$ eigenvalue with 0. $\mathbf{I}_n - \mathbf{H}\mathbf{H}^\top$ contains $n-k$ eigenvalue with 1 and k eigenvalue with 0. Consequently, $\mathbf{A}^{(i)}(\mathbf{I}_n - \mathbf{H}\mathbf{H}^\top)\mathbf{A}^{(i)}$ is PSD, which ensures that $\mathbf{Q} = \sum_{i=1}^n(\mathbf{A}^{(i)} - \mathbf{A}^{(i)}\mathbf{H}\mathbf{H}^\top\mathbf{A}^{(i)})$ is PSD. As a result, we have $w_p = \mathrm{Tr}(\mathbf{K}_p\mathbf{Q}) \geq 0\ \forall p$, guaranteeing the positiveness of $\mathbf{W}$. Meanwhile, $\mathbf{W}$ is also a symmetric PSD matrix according to [40]. Consequently, $2\mathbf{W}+\lambda\tilde{\mathbf{M}}$ is a symmetric PSD matrix. ∎

On the basis of Theorem 1, we can guarantee that the optimization in (17) w.r.t $\boldsymbol{\gamma}$ is a traditional QP with linear constraints. Therefore, it can be conveniently handled by the existing optimization packages.

Algorithm 1 presents an outline of solving (10) by the proposed algorithm, where we adopt the zero-filling method to initially impute the missing elements of $\{\mathbf{K}_p^{(0)}\}_{p=1}^m$ and utilize $\mathrm{obj}^{(t)}$ to represent the objective value at the t-th iteration. Besides, the neighbors of each sample remain unvaried during the optimization procedure in LI-MKKM-MR. In specific, we calculate the τ-nearest neighbors of each sample by $\mathbf{K}_{\boldsymbol{\gamma}^{(0)}}$. In this way, the optimization target of LI-MKKM-MR is guaranteed to be reduced in a monotonic manner when we update one variable and keep the others unchanged iteratively. Simultaneously, the objective is lower bounded by zero. Hence, it is guaranteed that LI-MKKM-MR converges into a local optimal solution. Experimental results have demonstrated that our method usually converges quickly.

The end of this part analyzes the computational complexity of our method. In specific, the computational complexity of LI-MKKM-MR is $\mathcal{O}(n^3 + \sum_{p=1}^m n_p^3 + m^3)$ at each iteration, where n_p $(n_p \leq n)$ and m refer to the number of observed samples of $\mathbf{K}_p$ and base kernels. The complexity of LI-MKKM-MR can be compared to that of MKKM-IK [34] and LI-MKKM [35]. Moreover, each sample of $\mathbf{K}_p$ is independent so that they can be measured in a parallel manner. By this means, our LI-MKKM-MR can scale well regardless of the variation of the base kernels number.

C. Theoretical Results

The generalization error of the k-means clustering algorithm has been widely discussed in the existing literature [36], [41], and [42]. We first establish the theoretical connection between the existing MKKM-IK [36] with LI-MKKM-MR, and further derive the generalization error bound of LI-MKKM-MR based on the theoretical results in [36]. The following theorem (Theorem 2) points out that the local kernel alignment adopted in our LI-MKKM-MR can be achieved by normalizing each base kernel matrix.

Theorem 2: The local kernel alignment criterion in (8) is equivalent to the widely adopted global kernel alignment by normalizing each base kernel matrix.

Proof: The objective function in (8) can be written as

$$\begin{aligned}
&\sum_{i=1}^{n} \mathrm{Tr}\Big(\mathbf{K}_{\boldsymbol{\gamma}}\Big(\mathbf{A}^{(i)} - \mathbf{A}^{(i)}\mathbf{H}\mathbf{H}^\top\mathbf{A}^{(i)}\Big)\Big)\\
&= \sum_{i=1}^{n}\Big\langle \mathbf{A}^{(i)} \otimes \mathbf{K}_{\boldsymbol{\gamma}}, \mathbf{A}^{(i)} \otimes \Big(\mathbf{I} - \mathbf{H}\mathbf{H}^\top\Big)\Big\rangle_{\mathrm{F}}\\
&= \sum_{i=1}^{n}\Big\langle \mathbf{A}^{(i)} \otimes \mathbf{K}_{\boldsymbol{\gamma}}, \mathbf{I} - \mathbf{H}\mathbf{H}^\top\Big\rangle_{\mathrm{F}}\\
&= \left\langle \left(\sum_{i=1}^{n}\mathbf{A}^{(i)}\right) \otimes \mathbf{K}_{\boldsymbol{\gamma}}, \mathbf{I} - \mathbf{H}\mathbf{H}^\top\right\rangle_{\mathrm{F}}\\
&= \sum_{p=1}^{m}\gamma_p^2\left\langle \left(\sum_{i=1}^{n}\mathbf{A}^{(i)}\right) \otimes \mathbf{K}_{p}, \mathbf{I} - \mathbf{H}\mathbf{H}^\top\right\rangle_{\mathrm{F}}\\
&= \sum_{p=1}^{m}\gamma_p^2\Big\langle \tilde{\mathbf{K}}_{p}, \mathbf{I} - \mathbf{H}\mathbf{H}^\top\Big\rangle_{\mathrm{F}}\\
&= \mathrm{Tr}\Big(\tilde{\mathbf{K}}_{\boldsymbol{\gamma}}\Big(\mathbf{I} - \mathbf{H}\mathbf{H}^\top\Big)\Big) \qquad (18)
\end{aligned}$$

where $\otimes$ denotes elementwise multiplication between two matrices, $\tilde{\mathbf{K}}_p = (\sum_{i=1}^{n}\mathbf{A}^{(i)}) \otimes \mathbf{K}_p$ can be treated as a normalized $\mathbf{K}_p$, and $\tilde{\mathbf{K}}_{\boldsymbol{\gamma}} = \sum_{p=1}^{m}\gamma_p^2\tilde{\mathbf{K}}_p$. Consequently, by such normalization being applied on each base kernel, we can clearly see that the local kernel alignment criterion in (8) is exactly the global kernel alignment in [36]. This completes the proof. ■

Let $t(\mathbf{x}^{(p)}) = 1$ if the pth view of $\mathbf{x}$ is available; otherwise, $\mathbf{x}^{(p)}$ should be optimized. It is worth pointing out that $t(\mathbf{x}^{(p)})$ is a random variable that depends on $\mathbf{x}$. Let $\hat{\mathbf{C}} = [\hat{\mathbf{C}}_1, \ldots, \hat{\mathbf{C}}_k]$ be the k centroids and $\hat{\boldsymbol{\gamma}}$ be the kernel weights learned by LI-MKKM-MR. k-means clustering should make the reconstruction error small

$$\mathbb{E}\left[\min_{\mathbf{y}\in\{\mathbf{e}_1,\ldots,\mathbf{e}_k\}}\left\|\phi_{\hat{\boldsymbol{\gamma}}}(\mathbf{x}) - \hat{\mathbf{C}}\mathbf{y}\right\|_{\mathcal{H}}^2\right] \qquad (19)$$

where $\phi_{\hat{\boldsymbol{\gamma}}}(\mathbf{x}) = [\hat{\gamma}_1 t(\mathbf{x}^{(1)})\phi_1^\top(\mathbf{x}^{(1)}), \ldots, \hat{\gamma}_m t(\mathbf{x}^{(m)})\phi_m^\top(\mathbf{x}^{(m)})]^\top$, $\mathbf{e}_1, \ldots, \mathbf{e}_k$ form the orthogonal bases of $\mathbb{R}^k$.

We first define a function class

$$\mathcal{F} = \Big\{f : \mathbf{x} \mapsto \min_{\mathbf{y}\in\{\mathbf{e}_1,\ldots,\mathbf{e}_k\}}\left\|\phi_{\boldsymbol{\gamma}}(\mathbf{x}) - \mathbf{C}\mathbf{y}\right\|_{\mathcal{H}}^2 \Big| \boldsymbol{\gamma}^\top\mathbf{1}_m = 1,\ \gamma_p \geq 0,\\
\mathbf{C} \in \mathcal{H}^k,\ t(\mathbf{x}_i^{(p)})t(\mathbf{x}_j^{(p)})\tilde{\kappa}_p^\top(\mathbf{x}_i^{(p)}, \mathbf{x}_j^{(p)}) \leq b, \quad \forall p \quad \forall \mathbf{x}_i \in \mathcal{X}\Big\} \qquad (20)$$

where $\mathcal{H}^k$ represents the multiple kernel Hilbert space and $\tilde{\kappa}(\cdot,\cdot)$ is a kernel function corresponding to $\tilde{\mathbf{K}}_p$.

Based on Theorem 2, we derive the generalization error bound of the proposed LI-MKKM-MR by following [36].

Theorem 3: For any $\delta > 0$, with probability at least $1 - \delta$, the following holds for all $f \in \mathcal{F}$:

$$\mathbb{E}\big[f(\mathbf{x})\big] \leq \frac{1}{n}\sum_{i=1}^{n} f(\mathbf{x}_i) + \frac{4\sqrt{\pi}mb\mathcal{G}_{1n}(\boldsymbol{\gamma}, t)}{n} + \frac{4\sqrt{\pi}mb\mathcal{G}_{2n}(\boldsymbol{\gamma}, t)}{n} + \frac{\sqrt{8\pi}bk^2}{\sqrt{n}} + 2b\sqrt{\frac{\log 1/\delta}{2n}} \qquad (21)$$

where

$$\mathcal{G}_{1n}(\boldsymbol{\gamma}, t) \triangleq \mathbb{E}_{\gamma}\left[\sup_{\boldsymbol{\gamma},t}\sum_{i=1}^{n}\sum_{p,q=1}^{m}\gamma_{ipq}t(\mathbf{x}_i^{(p)})t(\mathbf{x}_i^{(q)})\gamma_p\gamma_q\right] \qquad (22)$$

$$\mathcal{G}_{2n}(\boldsymbol{\gamma}, t) = \mathbb{E}_{\gamma}\left[\sup_{\boldsymbol{\gamma},t}\sum_{i=1}^{n}\sum_{c=1}^{k}\sum_{p=1}^{m}\gamma_{icp}\gamma_p t\Big(\mathbf{x}_i^{(p)}\Big)\right] \qquad (23)$$

and $\gamma_{ipq}, \gamma_{icp}, i \in \{1, \ldots, n\}, p, q \in \{1, \ldots, m\}, c \in \{1, \ldots, k\}$ are i.i.d. Gaussian random variables with zero mean and unit standard deviation.

According to the analyses in [36], our local kernel alignment criterion in (8), with normalized base kernel matrices, is an upper bound of $1/n\sum_{i=1}^{n} f(\mathbf{x}_i)$. As a result, by minimizing $\mathrm{Tr}(\tilde{\mathbf{K}}_{\boldsymbol{\gamma}}(\mathbf{I}_n - \mathbf{H}\mathbf{H}^\top))$, one can obtain a small $1/n\sum_{i=1}^{n} f(\mathbf{x}_i)$ for good generalization. This justifies the good generalization ability of the LI-MKKM-MR. The detailed proof has been presented in the supplementary material.

IV. Experiments

A. Experimental Settings

In our experiments, we adopt ten widely used MKL benchmark datasets to verify the proposed algorithms, including Oxford Flower17 and Flower102,[1] Caltech102,[2] Digital,[3] Protein Fold Prediction,[4] and Reuters.[5] The information of them is shown in Table II. The kernel matrices of these datasets are precomputed and can be directly obtained from the aforementioned link. Caltech102-5 refers to the number of samples belonging to each cluster is 5, and the same for the rest datasets. The publicly access codes for kernel k-means and MKKM can be found in the website.[6]

Several well-known and widely used imputation methods, such as zero filling (ZF), mean filling (MF), KNN, and alignment-maximization filling (AF) are contained in [30]. After that, researchers take the imputed kernel matrices as the input of classical MKKM. The kind of two-stage methods are called MKKM + ZF, MKKM + MF, MKKM + KNN,

[1] http://www.robots.ox.ac.uk/~vgg/data/flowers/
[2] http://files.is.tue.mpg.de/pgehler/projects/iccv09/
[3] http://ss.sysu.edu.cn/~py/
[4] http://mkl.ucsd.edu/dataset/protein-fold-prediction/
[5] http://kdd.ics.uci.edu/databases/reuters21578/
[6] https://github.com/mehmetgonen/lmkkmeans/

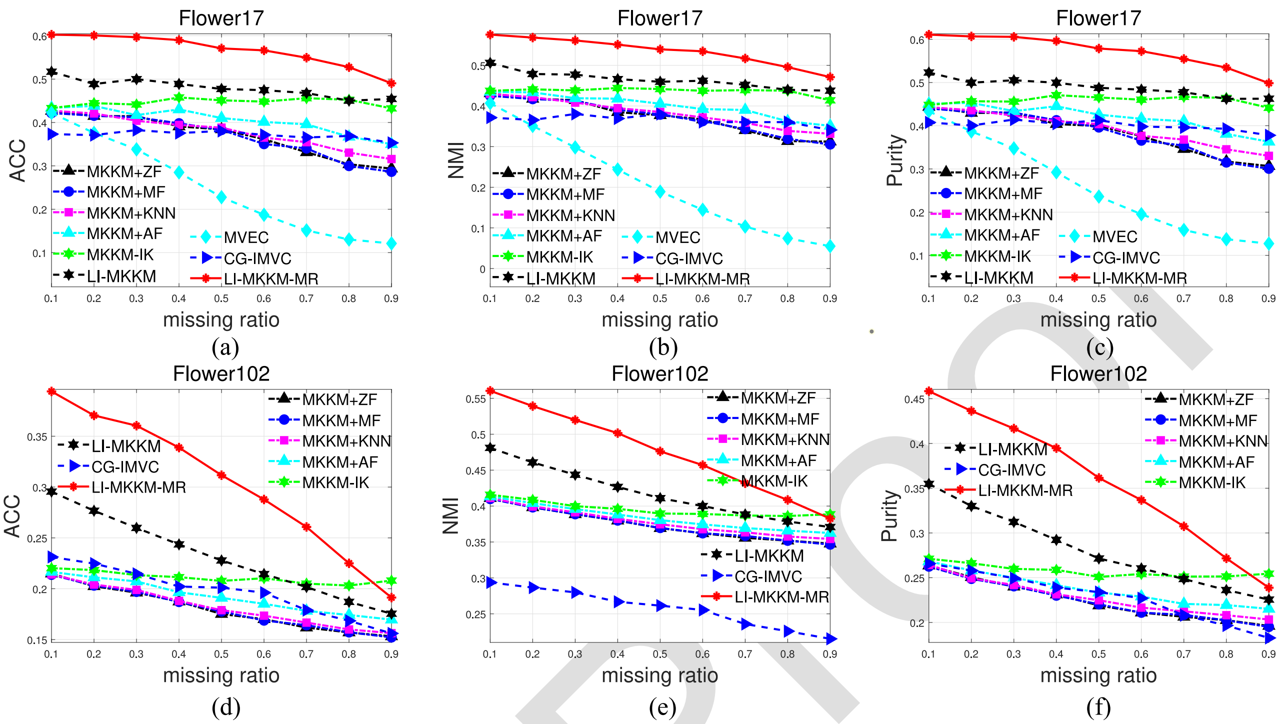


Fig. 1. Clustering ACC, NMI, and purity comparison with the variation of missing ratios on Flower17 and Flower102 datasets. (a) ACC with missing ratios on Flower17. (b) NMI with missing ratios on Flower17. (c) Purity with missing ratios on Flower17. (d) ACC with missing ratios on Flower102. (e) NMI with missing ratios on Flower102. (f) Purity with missing ratios on Flower102.

TABLE II
DATASETS SUMMARY

Dataset	#Samples	#Views	#Classes
Flower17	1360	7	17
Flower102	8189	4	102
Caltech102-5	510	48	102
Caltech102-10	1020	48	102
Caltech102-15	1530	48	102
Caltech102-20	2040	48	102
Caltech102-25	2550	48	102
Caltech102-30	3060	48	102
Digital	2000	3	10
ProteinFold	694	12	27
Reuters	18758	5	6

and MKKM + AF, respectively. Also, the newly proposed MKKM-IK [34], LI-MKKM [35], MVEC [43], and CG-IMVC [44] are also incorporated as strong baselines. The algorithms in [31], [32], and [45] are not incorporated into our experimental comparison since that these algorithms only consider the missing of input features, rather than the rows or columns of base kernel matrices in our case.

In the experiment, ε is used to denote the percentage of incomplete samples. Intuitively, the clustering performance will become less accurate when the value of ε is increasing. In our simulation, we set ε as $[0.1 : 0.1 : 0.9]$ on all the ten datasets. The performance metrics in this simulation include the clustering accuracy (ACC), normalized mutual information (NMI), and purity. For each method, we present the best result among 50 trials, where each trial started from a random initialization state. As a result, the effect of randomness caused by k-means could be alleviated. We generate "incomplete" patterns randomly for ten times and report the statistical results. For all datasets, the quantity of clusters is given and set as the ground truth of classes. The generation of the missing vectors $\{\mathbf{s}_p\}_{p=1}^{m}$ follows the approach in [34]: 1) randomly select round$(\varepsilon * n)$ samples with the rounding function round$(\cdot)$; 2) generate a random vector $\mathbf{v} = (v_1, \ldots, v_k, \ldots, v_m)$, $v_k \in [0, 1]$ and a scalar v_0, $v_0 \in [0, 1]$ for each selected sample; 3) if $v_p \geq v_0$, it presents the pth view for this sample; and 4) if there is no $v_p \geq v_0$, generate a new $\mathbf{v}$. Note that there is no requirement on complete view for each sample. In this instance, the index vector $\mathbf{s}_p$ is obtained to list the samples with the presentation on the pth view.

B. Experimental Results

Experiments on Flower17 and Flower102: Three performance metrics, including: 1) the ACC; 2) NMI; and 3) purity, of the testing algorithms with the variation of missing ratios in $[0.1, \ldots, 0.9]$ on the Flower17 and Flower102 datasets have been demonstrated in Fig. 1. We have the following observations.

1) The newly proposed MKKM-IK [36] (in green) has shown promising performance improvements on the ACC, NMI, and purity compared to the previous two-stage imputation methods. For example, the MKKM + AF outperforms MKKM-IK by $0.1\%, 0.6\%, 2.5\%, 2.8\%, 4.1\%, 4.7\%, 6.0\%, 8.5\%,$ and 8.2% in terms of clustering accuracy on Flower17, which clearly demonstrates the benefit of the joint optimization on imputation and clustering.

TABLE III
AGGREGATED ACC, NMI, AND PURITY COMPARISON (MEAN ± STD) OF DIFFERENT KINDS OF CLUSTERING ALGORITHMS ON FLOWER17 AND FLOWER102 DATASETS

Datasets	MKKM				MKKM-IK	LI-MKKM	MVEC	CG-IMVC	LI-MKKM-MR
	+ZF	+MF	+KNN	+AF [31]	[37]	[36]	[45]	[46]	Proposed
ACC									
Flower17	36.9 ± 0.8	36.8 ± 0.6	37.8 ± 0.6	40.5 ± 0.7	44.6 ± 0.6	48.0 ± 0.4	24.9 ± 0.4	37.1 ± 0.7	$\mathbf{56.6 \pm 0.3}$
Flower102	18.0 ± 0.2	18.0 ± 0.2	18.2 ± 0.1	19.2 ± 0.1	21.1 ± 0.2	23.1 ± 0.1	–	19.7 ± 0.3	$\mathbf{30.5 \pm 0.3}$
NMI									
Flower17	37.3 ± 0.4	37.3 ± 0.5	38.2 ± 0.5	40.1 ± 0.4	43.7 ± 0.3	46.4 ± 0.2	20.7 ± 0.4	36.5 ± 0.7	$\mathbf{53.5 \pm 0.2}$
Flower102	37.4 ± 0.1	37.4 ± 0.1	37.8 ± 0.1	38.4 ± 0.1	39.6 ± 0.1	41.8 ± 0.1	–	25.8 ± 0.3	$\mathbf{47.5 \pm 0.1}$
Purity									
Flower17	38.4 ± 0.6	38.3 ± 0.6	39.3 ± 0.6	42.0 ± 0.6	45.9 ± 0.5	48.9 ± 0.4	25.7 ± 0.4	40.1 ± 0.7	$\mathbf{57.3 \pm 0.2}$
Flower102	22.5 ± 0.1	22.4 ± 0.1	22.8 ± 0.1	23.7 ± 0.2	25.8 ± 0.2	28.1 ± 0.1	–	22.9 ± 0.3	$\mathbf{35.8 \pm 0.3}$

2) Also, LI-MKKM outperforms MKKM-IK by 8.4%, 4.4%, 5.8%, 3.1%, 2.6%, 2.6%, 1.2%, 0.2%, and 2.2% on Flower17. This result clearly verifies that the utilizing data's local structure further boosts the clustering performance.
3) Furthermore, our proposed LI-MKKM-MR (in red) significantly outperforms the LI-MKKM in all cases from Fig. 1(a)–(f) in the aspect of clustering performance. For example, LI-MKKM-MR further outperforms LI-MKKM by 8.5%, 11.2%, 9.7%, 10.1%, 9.4%, 9.2%, 8.2%, 7.7%, and 3.6%. This result indicates the effectiveness of incorporating the matrix-induced regularization.
4) In addition, our newly proposed method demonstrates stronger advantage when compared to previous ones, especially under low missing ratios. It is notable that in Fig. 1(a), when the missing ratio is extremely low ($\varepsilon = 0.1$), LI-MKKM-MR improves the second-best algorithm (LI-MKKM) by 8.5% in terms of clustering accuracy on Flower17.

In Table III, the aggregated ACC, NMI, purity, and the standard deviation are reported, where we show the highest performance one in bold. Similarly, the results also illustrate that MKKM + ZF, MKKM + MF, MKKM + KNN, MKKM + AF, and MKKM-IK are outperformed by the proposed algorithm. Specifically, the second-best one (LI-MKKM) is exceeded by the proposed LI-MKKM-MR by 7%.

Experiments on the Caltech102 Dataset: Fig. 2 presents ACC, NMI, and purity of all the testing algorithms over variational missing ratios on the Caltech102 datasets. We find out that the recently proposed MKKM-IK [36] (in green) achieves a comparable clustering performance with a representative two-stage imputation method MKKM + AF, while the proposed LI-MKKM outperforms MKKM-IK with significant improvements on all the performance criterions, details can be found in Fig. 2(a)–(i). More precisely, LI-MKKM obtains 6.4%, 5.0%, 5.1%, 4.7%, 4.6%, 4.5%, 3.8%, 3.2%, and 2.6% higher clustering accuracy than MKKM-IK when the missing ratios vary from 0.1 to 0.9 on Caltech102-30. This also illustrates that the well utilization of the local structure of data assures performance improvement. Furthermore, by taking into account the correlation among base kernels, LI-MKKM-MR further improves the clustering performance over the baseline LI-MKKM.

The aggregated ACC, NMI, and purity, and the standard deviation on Caltech 102 datasets are reported in Table IV. Similarly, in comparison to the MKKM + ZF, MKKM + MF, MKKM + KNN, MKKM + AF, and MKKM-IK, our method still achieves much better clustering performance. For instance, the proposed LI-MKKM-MR obtains 2.1%, 2.1%, 2.8%, 2.4%, 2.7%, and 2.4% higher clustering accuracy than LI-MKKM. In addition, LI-MKKM-MR achieves comparable clustering performance with the newly proposed CG-IMVC [44] in terms of ACC and purity on Caltech102. However, LI-MKKM-MR significantly outperforms CG-IMVC in terms of NMI. The results on Caltech102-5, Caltech102-10, and Caltech102-15 are provided in the supplementary material due to space limitation, whose results demonstrate the same conclusion as well.

Experiments on the UCI-Digital Dataset: In this simulation, we apply all the testing methods on the UCI-Digital dataset, which is widely utilized in MKC as a benchmark. For each kind of missing ratio, we generate "incomplete patterns" ten times and report their averaged results.

The ACC, NMI, and purity of all the testing methods over variational missing ratios are presented in Fig. 3. It is clear that the latest proposed MKKM-IK provides unsatisfactory results on UCI-Digital, which is even worse than MKKM+KNN. However, LI-MKKM significantly outperforms the second-best one (MKKM + KNN) by 22.2%, 21.9%, 20.6%, 19.5%, 17.9%, 17.9%, 20.4%, 23.8%, and 23.2% on accuracy. In addition, the proposed LI-MKKM-MR further consistently improves the clustering performance of LI-MKKM. The aggregated clustering results in Table V also denote the same performance.

Experiments on the Protein Fold Prediction Dataset: In this experiment, the protein fold dataset is applied to evaluate the testing methods, and we report all results in Fig. 4 and Table VI. Also, we can find that our LI-MKKM-MR also achieves much better results than the rest algorithms on ACC, NMI, and purity on the dataset.

Experiments on the Reuters Dataset: The clustering performance in terms of ACC, NMI, and purity with the variation of missing ratios on Reuters is plotted in Fig. 5. As seen, our proposed algorithm once again demonstrates significant superiority over the compared ones. We also report the aggregated ACC, NMI, and purity in Table VII, which also verify the effectiveness of the proposed LI-MKKM-MR. The

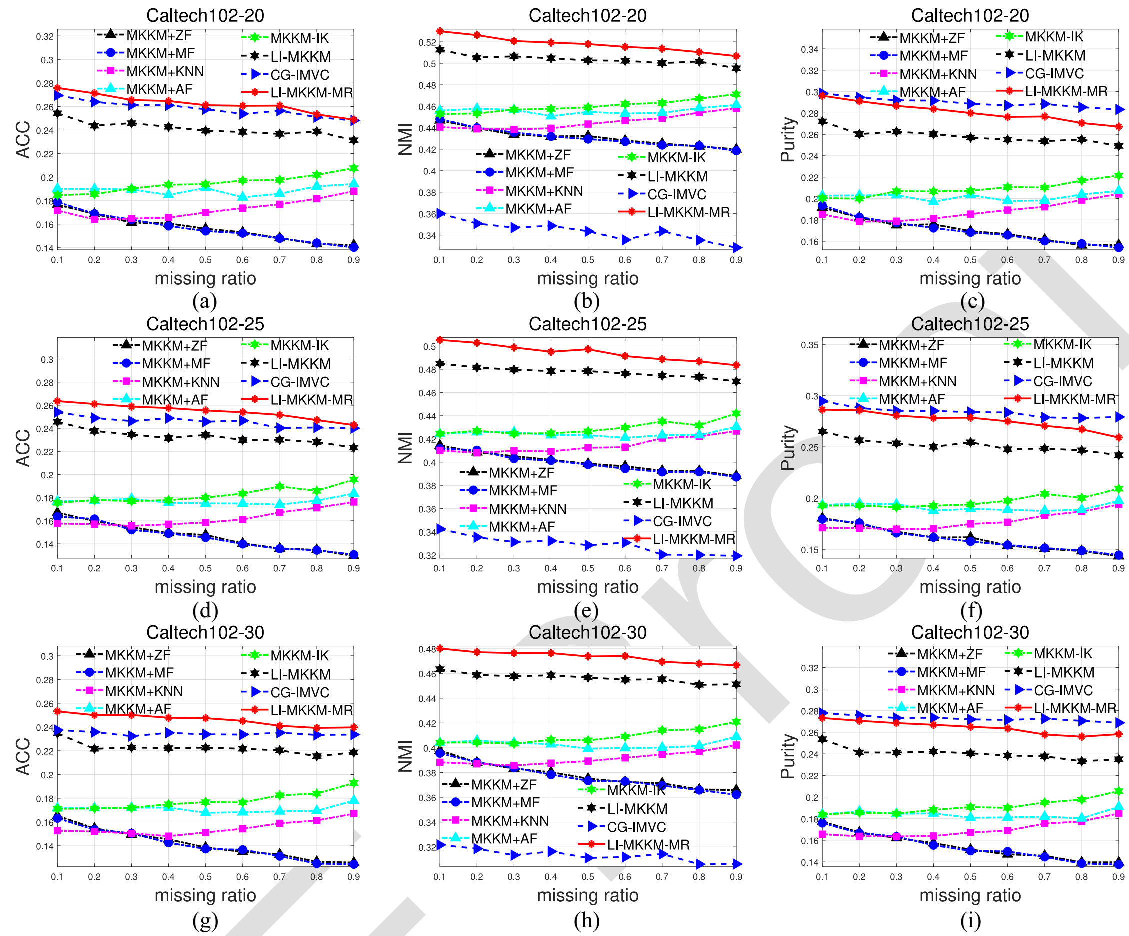


Fig. 2. Clustering ACC, NMI, and purity comparison with the variation of missing ratios on Caltech102-20, Caltech102-25, and Caltech102-30. (a) ACC with missing ratios on Caltech102-20. (b) NMI with missing ratios on Caltech102-20. (c) Purity with missing ratios on Caltech102-20. (d) ACC with missing ratios on Caltech102-25. (e) NMI with missing ratios on Caltech102-25. (f) Purity with missing ratios on Caltech102-25. g) ACC with missing ratios on Caltech102-30. (h) NMI with missing ratios on Caltech102-30. (i) Purity with missing ratios on Caltech102-30.

TABLE IV

TOTAL ACC, NMI, AND PURITY COMPARISON (MEAN ± STD) OF VARIOUS CLUSTERING ALGORITHMS ON CALTECH102. ON ACCOUNT OF OUT OF MEMORY, THE CLUSTERING RESULTS OF MVEC [43] ON CALTECH102-15, CALTECH102-20, CALTECH102-25, AND CALTECH102-30 ARE NOT REPORTED

	MKKM				MKKM-IK	LI-MKKM	MVEC	CG-IMVC	LI-MKKM-MR
	+ZF	+MF	+KNN	+AF [31]	[37]	[36]	[45]	[46]	Proposed
ACC									
Cal102-5	26.1 ± 0.3	25.7 ± 0.3	27.3 ± 0.3	29.0 ± 0.3	28.9 ± 0.3	31.4 ± 0.3	26.8 ± 0.2	33.8 ± 0.2	$\mathbf{34.0 \pm 0.3}$
Cal102-10	19.7 ± 0.2	19.7 ± 0.2	21.5 ± 0.2	22.6 ± 0.2	22.7 ± 0.2	27.3 ± 0.2	22.4 ± 0.1	$\mathbf{28.9 \pm 0.2}$	$\mathbf{28.9 \pm 0.3}$
Cal102-15	17.1 ± 0.2	17.1 ± 0.2	18.9 ± 0.1	20.3 ± 0.2	20.8 ± 0.2	25.1 ± 0.2	–	$\mathbf{27.3 \pm 0.1}$	27.0 ± 0.4
Cal102-20	15.7 ± 0.1	15.7 ± 0.2	17.3 ± 0.2	18.9 ± 0.2	19.5 ± 0.1	24.1 ± 0.2	–	25.8 ± 0.2	$\mathbf{26.3 \pm 0.2}$
Cal102-25	14.7 ± 0.2	14.6 ± 0.1	16.2 ± 0.1	17.7 ± 0.2	18.3 ± 0.2	23.3 ± 0.2	–	24.6 ± 0.2	$\mathbf{25.5 \pm 0.2}$
Cal102-30	14.2 ± 0.1	14.1 ± 0.1	15.5 ± 0.2	17.1 ± 0.2	17.8 ± 0.2	22.2 ± 0.1	–	23.5 ± 0.1	$\mathbf{24.6 \pm 0.1}$
NMI									
Cal102-5	64.3 ± 0.2	63.9 ± 0.1	65.9 ± 0.2	66.6 ± 0.1	66.5 ± 0.2	67.1 ± 0.2	65.6 ± 0.1	52.9 ± 0.4	$\mathbf{68.6 \pm 0.2}$
Cal102-10	53.6 ± 0.1	53.7 ± 0.1	55.2 ± 0.1	55.7 ± 0.2	55.8 ± 0.1	58.7 ± 0.1	55.1 ± 0.1	40.4 ± 0.5	$\mathbf{59.2 \pm 0.3}$
Cal102-15	47.4 ± 0.1	47.4 ± 0.1	48.8 ± 0.1	49.7 ± 0.1	50.1 ± 0.1	53.6 ± 0.1	–	37.0 ± 0.3	$\mathbf{54.6 \pm 0.2}$
Cal102-20	43.1 ± 0.1	43.1 ± 0.2	44.5 ± 0.1	45.6 ± 0.2	46.0 ± 0.1	50.4 ± 0.1	–	34.4 ± 0.3	$\mathbf{51.8 \pm 0.1}$
Cal102-25	40.0 ± 0.1	39.9 ± 0.1	41.5 ± 0.1	42.5 ± 0.2	43.0 ± 0.2	47.7 ± 0.2	–	32.9 ± 0.3	$\mathbf{49.4 \pm 0.1}$
Cal102-30	37.8 ± 0.1	37.7 ± 0.1	39.2 ± 0.1	40.3 ± 0.1	40.9 ± 0.1	45.6 ± 0.1	–	31.3 ± 0.2	$\mathbf{47.4 \pm 0.1}$
Purity									
Cal102-5	26.7 ± 0.4	26.4 ± 0.3	27.9 ± 0.3	29.8 ± 0.3	29.6 ± 0.3	32.6 ± 0.3	27.3 ± 0.2	$\mathbf{35.9 \pm 0.2}$	35.5 ± 0.3
Cal102-10	21.0 ± 0.2	21.0 ± 0.2	22.9 ± 0.2	24.0 ± 0.3	24.2 ± 0.2	29.0 ± 0.2	23.3 ± 0.1	$\mathbf{31.7 \pm 0.2}$	30.8 ± 0.3
Cal102-15	18.5 ± 0.2	18.5 ± 0.2	20.4 ± 0.2	21.6 ± 0.2	22.2 ± 0.2	26.7 ± 0.2	–	$\mathbf{30.2 \pm 0.1}$	28.8 ± 0.3
Cal102-20	17.1 ± 0.1	17.0 ± 0.2	18.8 ± 0.2	20.2 ± 0.2	20.9 ± 0.1	25.8 ± 0.2	–	$\mathbf{29.0 \pm 0.2}$	28.1 ± 0.2
Cal102-25	16.0 ± 0.2	16.0 ± 0.2	17.7 ± 0.2	19.1 ± 0.2	19.7 ± 0.1	25.2 ± 0.2	–	$\mathbf{28.4 \pm 0.1}$	27.6 ± 0.2
Cal102-30	15.4 ± 0.1	15.4 ± 0.1	17.0 ± 0.1	18.4 ± 0.2	19.1 ± 0.2	24.0 ± 0.1	–	$\mathbf{27.3 \pm 0.1}$	26.5 ± 0.1

results of MVEC [43] and CG-IMVC [44] on Reuters are not reported due to out of memory.

In short, we summarize that our algorithm has three advantages.

1) *Joint Optimization Based on Imputation and Clustering*: First, the process of imputation is guided by the clustering results, which makes the imputation more direct to the final goal. Second, refining the clustering results

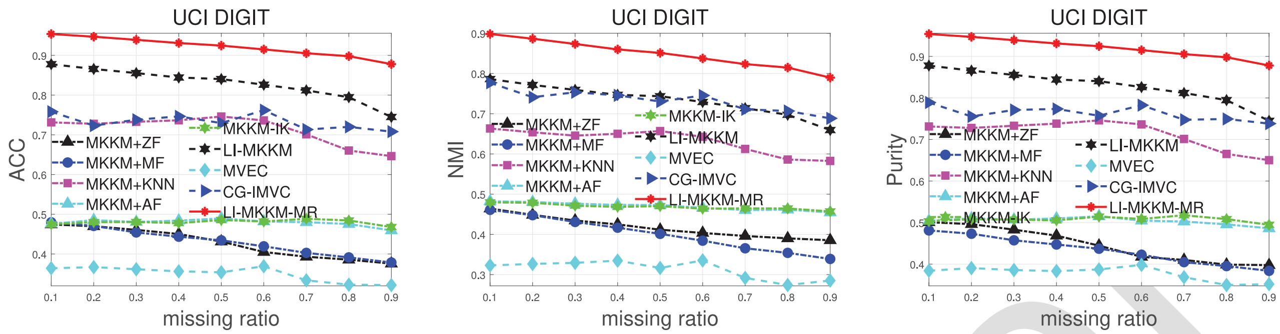


Fig. 3. Clustering ACC, NMI, and purity comparison with the variation of missing ratios on the UCI-digital dataset.

TABLE V
TOTAL ACC, NMI, AND PURITY COMPARISON (MEAN $\pm$ STD) OF VARIOUS CLUSTERING ALGORITHMS ON UCI-DIGITAL

MKKM				MKKM-IK	LI-MKKM	MVEC	CG-IMVC	LI-MKKM-MR
+ZF	+MF	+KNN	+AF [31]	[37]	[36]	[45]	[46]	Proposed
ACC								
42.7 ± 0.4	43.1 ± 0.3	71.3 ± 1.0	47.9 ± 0.5	48.0 ± 0.4	82.9 ± 0.3	35.0 ± 0.8	73.3 ± 1.1	$\mathbf{92.1 \pm 0.3}$
NMI								
41.8 ± 0.2	40.0 ± 0.2	63.3 ± 0.5	47.0 ± 0.2	46.9 ± 0.2	73.4 ± 0.3	31.3 ± 1.1	73.3 ± 0.9	$\mathbf{84.8 \pm 0.4}$
Purity								
44.6 ± 0.5	43.4 ± 0.3	71.4 ± 0.7	50.4 ± 0.3	50.8 ± 0.4	82.9 ± 0.3	37.8 ± 0.8	76.3 ± 1.0	$\mathbf{92.1 \pm 0.3}$

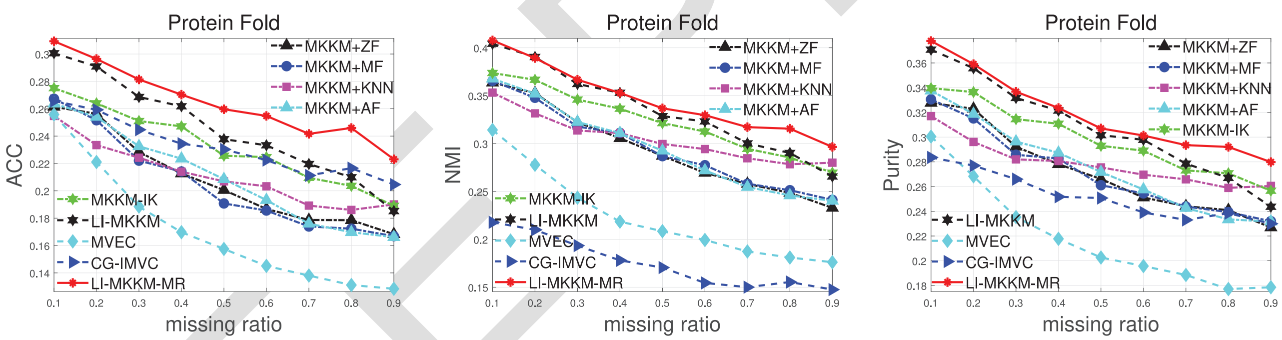


Fig. 4. Clustering ACC, NMI, and purity comparison with the variation of missing ratios on the protein Fold Prediction dataset.

TABLE VI
TOTAL ACC, NMI, AND PURITY COMPARISON (MEAN $\pm$ STD) OF VARIOUS CLUSTERING ALGORITHMS ON THE PROTEIN FOLD DATASET

MKKM				MKKM-IK	LI-MKKM	MVEC	CG-IMVC	LI-MKKM-MR
+ZF	+MF	+KNN	+AF [31]	[37]	[36]	[45]	[46]	Proposed
ACC								
20.8 ± 0.2	20.5 ± 0.3	21.1 ± 0.5	21.0 ± 0.2	23.2 ± 0.6	24.5 ± 0.5	17.1 ± 0.2	23.2 ± 0.3	$\mathbf{26.5 \pm 0.2}$
NMI								
29.3 ± 0.4	29.5 ± 0.5	30.5 ± 0.4	29.5 ± 0.3	32.3 ± 0.6	33.5 ± 0.3	22.3 ± 0.2	17.5 ± 0.6	$\mathbf{34.6 \pm 0.2}$
Purity								
27.2 ± 0.4	27.2 ± 0.4	27.9 ± 0.5	27.5 ± 0.4	29.8 ± 0.7	30.8 ± 0.4	21.8 ± 0.2	25.2 ± 0.5	$\mathbf{31.9 \pm 0.3}$

can benefits from this meaningful imputation. These two learning processes work well together, thus leading to the clustering performance improvement. In contrast, MKKM + MF, MKKM + KNNMKKM + ZF, and MKKM + AF algorithms do not fully make use of the connection between the imputation and clustering procedures. This may produce imputation, which does not well serve the subsequent clustering as originally expected, affecting the clustering performance.

2) *Considerably Utilizing Data's Local Structure*: Our local kernel alignment criterion is flexible and it makes the prespecified kernels aligned for better clustering performance.

3) *Well Considering the Correlation of Incomplete Base Kernels*: The incorporated matrix-induced regularization reduces the high redundancy and enforces low diversity among the selected kernels, making the prespecified kernels be well utilized.

These factors have led to significant improvements in cluster performance.

C. Reconstruction Error Comparison of LI-MKKM-MR

In this section, we evaluate the reconstruction errors of the LI-MKKM-MR with the aforementioned algorithms on all benchmark datasets. To do this, we calculate the reconstruction error between the ground-truth kernels and the imputed ones

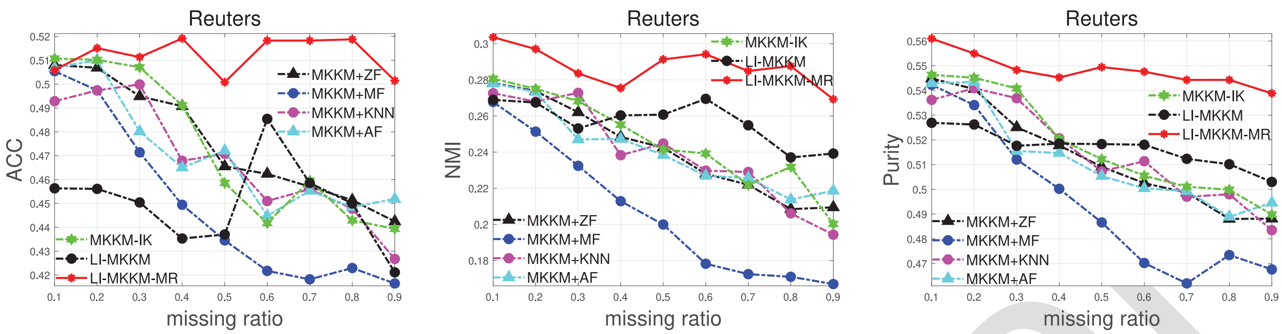


Fig. 5. Clustering ACC, NMI, and purity comparison with the variation of missing ratios on Reuters.

TABLE VII
AGGREGATED ACC, NMI, AND PURITY COMPARISON (MEAN ± STD) OF VARIOUS CLUSTERING ALGORITHMS ON REUTERS

MKKM				MKKM-IK	LI-MKKM	LI-MKKM-MR
+ZF	+MF	+KNN	+AF [31]	[37]	[36]	Proposed
ACC						
47.6 ± 0.0	44.9 ± 0.2	46.8 ± 0.2	47.1 ± 0.1	47.4 ± 0.4	45.0 ± 0.7	$\mathbf{51.2 \pm 1.0}$
NMI						
24.1 ± 0.1	20.6 ± 0.1	24.0 ± 0.3	24.1 ± 0.1	24.6 ± 0.3	25.7 ± 0.3	$\mathbf{28.7 \pm 0.1}$
Purity						
51.3 ± 0.0	49.4 ± 0.0	51.5 ± 0.2	51.2 ± 0.2	51.8 ± 0.1	51.7 ± 0.1	$\mathbf{54.8 \pm 0.0}$

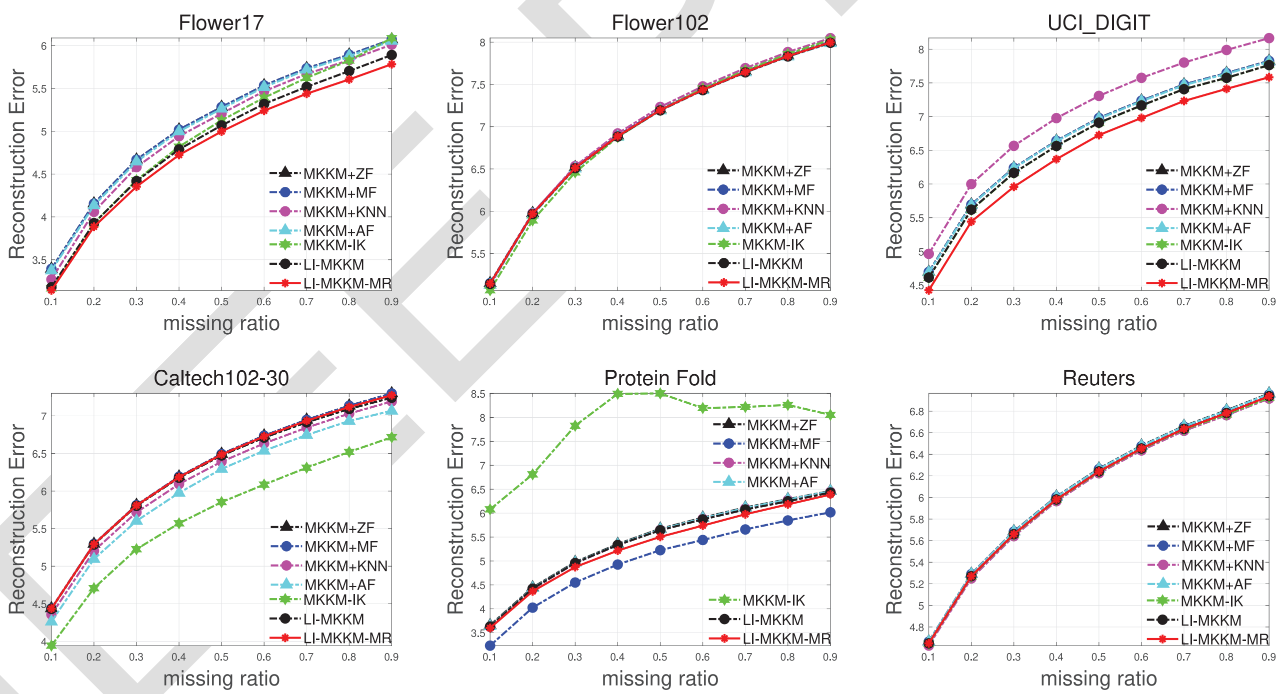


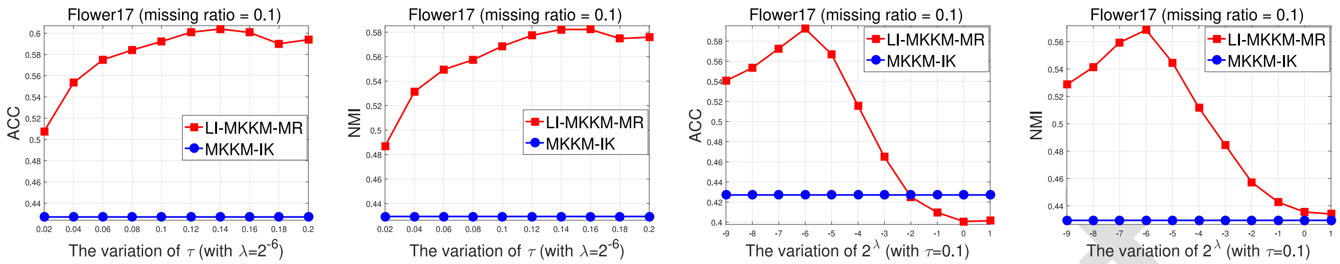
Fig. 6. Reconstruction error comparison of the compared algorithms with the variation of missing ratios on benchmark datasets.

via $\sum_{p=1}^{m} \|\mathbf{K}_p(\mathbf{s}_p, \mathbf{s}_p) - \hat{\mathbf{K}}_p(\mathbf{s}_p, \mathbf{s}_p)\|^2$, where $\mathbf{K}_p$ and $\hat{\mathbf{K}}_p$ denote the ground-truth and the imputed one, and $\mathbf{s}_p$ denotes the missing indices of the pth view. The results under various missing ratios are shown in Fig. 6. As observed, the kernels imputed by our algorithm align with the ground-truth kernels are comparable or slightly better when compared to those obtained by the existing imputation algorithms. Note that our ultimate goal in this work is clustering, while imputation is only a by-product. How to impute the missing views which not only achieves better clustering performance but also produces better imputation result is worth further exploring.

D. Parameter Sensitivity of LI-MKKM-MR

In this part, we analyze that relationship between the clustering performance and matrix-induced regularization. Referring to (10), LI-MKKM-MR induces the ratio of the nearest neighbors τ and regularization parameter λ. In the following, we conduct another experiment to show the variation of performance among different τ and λ on the Flower17 dataset.

Fig. 7(a) and (b) shows the ACC and NMI of our algorithm by varying τ in a huge range $[0.02 : 0.02 : 0.2]$ with $\lambda = 2^{-6}$. From these figures, we can find that: 1) ACC fluctuates with the monotonically increasing of τ and 2) the start points of the



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

Fig. 7. Sensitivity of the proposed LI-MKKM-MR with the variation of λ and τ. (a) ACC with variation of τ on Flower17. (b) NMI with variation of τ on Flower17. (c) ACC with variation of λ on Flower17. (d) NMI with variation of λ on Flower17.

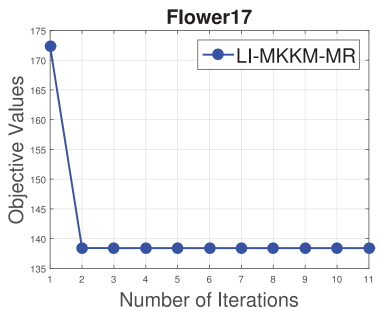


Fig. 8. Proposed algorithm convergence illustration.

ACC curves are typically higher than the end points, which induces that when the matrix-induced regularization term is dominated at ending points while the local kernel alignment maximization is dominated at starting points. These observations verify the successful joint preservation of the local structure of data and the matrix-induced regularization term in our algorithm. Similarly, Fig. 7(c) and 7(d) presents the ACC and NMI of our algorithms by tuning λ from 2^{-9} to 2 with $\tau = 0.1$. In this scenario, our algorithm also shows stable performance over variational λ.

As aforementioned, we conclude that compared to only preserving global kernel alignment, such as MKKM-IK in [36], our proposed algorithms are more essential to the clustering performance by preserving the local structure of data. Meanwhile, the clustering performance could be further improved by incorporating the correlation among base kernels. By appropriately integrating these two factors, it is possible to obtain the best clustering performance. Practically, there exists a tradeoff between the preservation of the local geometric structure and the correlation of base kernels to ensure the best clustering.

E. Convergence of LI-MKKM-MR

According to [46], the convergence of our proposed algorithm is guaranteed. We present one simulation trail of the proposed LI-MKKM-MR on the Flower 17 dataset as an example in 8. It is clearly shown that the objective value of the proposed algorithm is monotonically decreased and converges in a few iteration.

V. Conclusion

Though the newly proposed LI-MKKM is able to tackle the task of MKC with incomplete kernels, it takes the correlation among base kernels into account insufficiently. We proposed to calculate the kernel alignment to address this issue together with matrix-induced regularization in a local manner. The proposed algorithm efficiently solves the resultant optimization problem, and extensive experiments on benchmarks have demonstrated that LI-MKKM-MR consistently outperforms state-of-the-art baseline algorithms. In the future, we will design efficient and effective algorithms to solve the optimization problem directly without approximating $\mathbf{M}$ in (9).

References

[1] K. Zhan, X. Chang, J. Guan, L. Chen, Z. Ma, and Y. Yang, "Adaptive structure discovery for multimedia analysis using multiple features," *IEEE Trans. Cybern.*, vol. 49, no. 5, pp. 1826–1834, May 2019.

[2] K. Zhan, F. Nie, J. Wang, and Y. Yang, "Multiview consensus graph clustering," *IEEE Trans. Image Process.*, vol. 28, pp. 1261–1270, 2019.

[3] D. Huang, J.-H. Lai, and C.-D. Wang, "Robust ensemble clustering using probability trajectories," *IEEE Trans. Knowl. Data Eng.*, vol. 28, no. 5, pp. 1312–1326, May 2016.

[4] C.-D. Wang, J.-H. Lai, and P. S. Yu, "Multi-view clustering based on belief propagation," *IEEE Trans. Knowl. Data Eng.*, vol. 28, no. 4, pp. 1007–1021, Apr. 2016.

[5] M. Yin, J. Gao, S. Xie, and Y. Guo, "Multiview subspace clustering via tensorial t-product representation," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 3, pp. 851–864, Mar. 2019.

[6] Z. Ren, S. X. Yang, Q. Sun, and T. Wang, "Consensus affinity graph learning for multiple kernel clustering," *IEEE Trans. Cybern.*, vol. 51, no. 6, pp. 3273–3284, Jun. 2021.

[7] K. Zhan, C. Niu, C. Chen, F. Nie, C. Zhang, and Y. Yang, "Graph structure fusion for multiview clustering," *IEEE Trans. Knowl. Data Eng.*, vol. 31, no. 10, pp. 1984–1993, Oct. 2019.

[8] W. Liang *et al.*, "Multi-view spectral clustering with high-order optimal neighborhood Laplacian matrix," *IEEE Trans. Knowl. Data Eng.*, early access, Sep. 18, 2020, doi: 10.1109/TKDE.2020.3025100.

[9] B. Zhao, J. T. Kwok, and C. Zhang, "Multiple kernel clustering," in *Proc. SDM*, 2009, pp. 638–649.

[10] Z. Ren and Q. Sun, "Simultaneous global and local graph structure preserving for multiple kernel clustering," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 5, pp. 1839–1851, May 2021.

[11] S. Li, Y. Jiang, and Z. Zhou, "Partial multi-view clustering," in *Proc. AAAI*, 2014, pp. 1968–1974.

[12] M. Gönen and A. A. Margolin, "Localized data fusion for kernel k-means clustering with application to cancer biology," in *Proc. NIPS*, 2014, pp. 1305–1313.

[13] L. Du *et al.*, "Robust multiple kernel *k*-means clustering using ℓ_{21}-norm," in *Proc. IJCAI*, 2015, pp. 3476–3482.

[14] X. Liu, Y. Dou, J. Yin, L. Wang, and E. Zhu, "Multiple kernel *k*-means clustering with matrix-induced regularization," in *Proc. AAAI*, 2016, pp. 1888–1894.

[15] D. Huang, C.-D. Wang, and J.-H. Lai, "Locally weighted ensemble clustering," *IEEE Trans. Cybern.*, vol. 48, no. 5, pp. 1460–1473, May 2018.

[16] M. Li, X. Liu, L. Wang, Y. Dou, J. Yin, and E. Zhu, "Multiple kernel clustering with local kernel alignment maximization," in *Proc. IJCAI*, 2016, pp. 1704–1710.

[17] S. Wang *et al.*, “Multi-view clustering via late fusion alignment maximization,” in *Proc. IJCAI*, 2019, pp. 3778–3784.

[18] S. Yu *et al.*, “Optimized data fusion for kernel k-means clustering,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 5, pp. 1031–1039, May 2012.

[19] X. Liu *et al.*, “Optimal neighborhood kernel clustering with multiple kernels,” in *Proc. AAAI*, 2017, pp. 2266–2272.

[20] Q. Wang, Z. Qin, F. Nie, and X. Li, “Spectral embedded adaptive neighbors clustering,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 4, pp. 1265–1271, Apr. 2019.

[21] M.-S. Chen, L. Huang, C.-D. Wang, D. Huang, and P. S. Yu, “Multiview subspace clustering with grouping effect,” *IEEE Trans. Cybern.*, early access, Dec. 7, 2020, doi: 10.1109/TCYB.2020.3035043.

[22] Z. Li, F. Nie, X. Chang, L. Nie, H. Zhang, and Y. Yang, “Rank-constrained spectral clustering with flexible embedding,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 12, pp. 6073–6082, Dec. 2018.

[23] Z. Li, F. Nie, X. Chang, Y. Yang, C. Zhang, and N. Sebe, “Dynamic affinity graph construction for spectral clustering using multiple features,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 12, pp. 6323–6332, Dec. 2018.

[24] M. C. Trinidad, R. Martin-Brualla, F. Kainz, and J. Kontkanen, “Multi-view image fusion,” in *Proc. ICCV*, 2019, pp. 4100–4109.

[25] M. Hu and S. Chen, “One-pass incomplete multi-view clustering,” in *Proc. AAAI*, 2019, pp. 3838–3845.

[26] C. Xu, Z. Guan, W. Zhao, H. Wu, Y. Niu, and B. Ling, “Adversarial incomplete multi-view clustering,” in *Proc. IJCAI*, 2019, pp. 3933–3939.

[27] S. Xiang, L. Yuan, W. Fan, Y. Wang, P. M. Thompson, and J. Ye, “Multi-source learning with block-wise missing data for Alzheimer’s disease prediction,” in *Proc. ACM SIGKDD*, 2013, pp. 185–193.

[28] R. Kumar, T. Chen, M. Hardt, D. Beymer, K. Brannon, and T. F. Syeda-Mahmood, “Multiple kernel completion and its application to cardiac disease discrimination,” in *Proc. ISBI*, 2013, pp. 764–767.

[29] Z. Ghahramani and M. I. Jordan, “Supervised learning from incomplete data via an EM approach,” in *Proc. NIPS*, 1993, pp. 120–127.

[30] A. Trivedi, P. Rai, H. Daumé III, and S. L. DuVall, “Multiview clustering with incomplete views,” in *Proc. NIPS Mach. Learn. Soc. Comput. Workshop* 2010, pp. 1–7.

[31] C. Xu, D. Tao, and C. Xu, “Multi-view learning with incomplete views,” *IEEE Trans. Image Process.*, vol. 24, pp. 5812–5825, 2015.

[32] W. Shao, L. He, and P. S. Yu, “Multiple incomplete views clustering via weighted nonnegative matrix factorization with $\ell_{2,1}$ regularization,” in *Machine Learning and Knowledge Discovery in Databases (ECML PKDD)*. Cham, Switzerland: Springer, 2015, pp. 318–334.

[33] S. Bhadra, S. Kaski, and J. Rousu, “Multi-view kernel completion,” 2016, *arXiv:1602.02518*.

[34] X. Liu, M. Li, L. Wang, Y. Dou, J. Yin, and E. Zhu, “Multiple kernel k-means with incomplete kernels,” in *Proc. AAAI*, 2017, pp. 2259–2265.

[35] X. Zhu *et al.*, “Localized incomplete multiple kernel k-means,” in *Proc. IJCAI*, 2018, pp. 3271–3277.

[36] X. Liu *et al.*, “Multiple kernel k-means with incomplete kernels,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 5, pp. 1191–1204, May 2020.

[37] S. Jegelka, A. Gretton, B. Schölkopf, B. K. Sriperumbudur, and U. von Luxburg, “Generalized clustering via kernel embeddings,” in *Proc. 32nd Annu. German Conf. AI Adv. Artif. Intell.*, 2009, pp. 144–152.

[38] L. Yuan, Y. Wang, P. M. Thompson, V. A. Narayan, and J. Ye, “Multi-source feature learning for joint analysis of incomplete multiple heterogeneous neuroimaging data,” *NeuroImage*, vol. 61, no. 3, pp. 622–632, 2012.

[39] Y. Liu *et al.*, “Incomplete multi-modal representation learning for Alzheimer’s disease diagnosis,” *Med. Image Anal.*, vol. 69, Apr. 2021, Art. no. 101953.

[40] C. Cortes, M. Mohri, and A. Rostamizadeh, “Algorithms for learning kernels based on centered alignment,” *J. Mach. Learn. Res.*, vol. 13, pp. 795–828, Jan. 2012.

[41] A. Maurer and M. Pontil, “*k*-dimensional coding schemes in Hilbert spaces,” *IEEE Trans. Inf. Theory*, vol. 56, no. 11, pp. 5839–5846, Nov. 2010.

[42] T. Liu, D. Tao, and D. Xu, “Dimensionality-dependent generalization bounds for *k*-dimensional coding schemes,” *Neural Comput.*, vol. 28, no. 10, pp. 2213–2249, 2016.

[43] Z. Tao, H. Liu, S. Li, Z. Ding, and Y. Fu, “From ensemble clustering to multi-view clustering,” in *Proc. IJCAI*, 2017, pp. 2843–2849.

[44] W. Zhou, H. Wang, and Y. Yang, “Consensus graph learning for incomplete multi-view clustering,” in *Advances in Knowledge Discovery and Data Mining*. Cham, Switzerland: Springer, 2019, pp. 529–540.

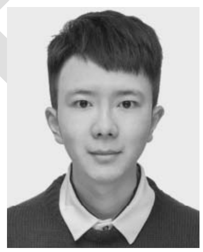
[45] H. Zhao, H. Liu, and Y. Fu, “Incomplete multimodal visual data grouping,” in *Proc. IJCAI*, 2016, pp. 2392–2398.

[46] J. C. Bezdek and R. J. Hathaway, “Convergence of alternating optimization,” *Neural Parallel Sci. Comput.*, vol. 11, no. 4, pp. 351–368, 2003.

Miaomiao Li is currently pursuing the Ph.D. degree with the National University of Defense Technology, Changsha, China.

She is currently a Lecturer with Changsha College, Changsha. She has published several peer-reviewed papers, such as IEEE Transactions on Pattern Analysis and Machine Intelligence, IEEE Transactions on Knowledge and Data Engineering, IEEE Transactions on Neural Networks and Learning Systems, IEEE Transactions on Multimedia, AAAI, IJCAI, and *Neurocomputing*. Her current research interests include kernel learning and multiview clustering.

Ms. Li serves on the Technical Program Committees of IJCAI/AAAI 2017–2020.



Jingyuan Xia received the B.Sc. and M.Sc. degrees from the National University of Defense Technology, Hunan, China, in 2020. He is currently pursuing the Ph.D. degree with the Department of Electrical and Electronic Engineering, Imperial College London, London, U.K.

His current research interests include bilinear convex optimization, low-rank matrix completion, sparse signal processing, and intelligent transportation estimation.

Huiying Xu received the M.S. degree from the National University of Defense Technology, Changsha, China, in 2006.

She is an Associate Professor with the College of Mathematics and Computer Science, Zhejiang Normal University, Jinhua, China, and also the Researcher with the Research Institute of Ningbo Cixing Company Ltd., Ningbo, China. Her research interests include kernel learning and feature selection, machine learning, deep learning, computer vision, image processing, pattern recognition, computer simulation, digital watermarking, and their applications.

Ms. Xu is a member of the China Computer Federation.

Qing Liao (Member, IEEE) received the Ph.D. degree in computer science and engineering from the Department of Computer Science and Engineering, Hong Kong University of Science and Technology, Hong Kong, in 2016, supervised by Prof. Q. Zhang.

She is currently an Assistant Professor with the School of Computer Science and Technology, Harbin Institute of Technology (Shenzhen), Shenzhen, China. Her research interests include artificial intelligence and data mining.

Xinzhong Zhu (Member, IEEE) received the M.S. degree from the National University of Defense Technology (NUDT), Changsha, China, and the Ph.D. degree from Xidian University, Xi'an, China, in 2018.

He is a Professor with the College of Mathematics and Computer Science, Zhejiang Normal University, Jinhua, China, and also the President of Research Institute of Ningbo Cixing Company Ltd., Ningbo, China. He has published more than 30 peer-reviewed papers, including those in highly regarded journals and conferences, such as the IEEE Transactions on Pattern Analysis and Machine Intelligence, the IEEE Transactions on Multimedia, the IEEE Transactions on Knowledge and Data Engineering, AAAI, and IJCAI. His research interests include machine learning, deep learning, computer vision, manufacturing informatization, robotics and system integration, and intelligent manufacturing.

Prof. Zhu served on the Technical Program Committees of IJCAI 2020 and AAAI 2020. He is a member of ACM and certified as a CCF Senior Member.

Xinwang Liu (Senior Member, IEEE) received the Ph.D. degree from the National University of Defense Technology (NUDT), Changsha, China, in 2013.

He is currently a Professor with the School of Computer, NUDT. His current research interests include kernel learning and unsupervised feature learning. He has published over 70 peer-reviewed papers, including those in highly regarded journals and conferences, such as IEEE Transactions on Pattern Analysis and Machine Intelligence, IEEE Transactions on Knowledge and Data Engineering, IEEE Transactions on Image Processing, IEEE Transactions on Neural Networks and Learning Systems, IEEE Transactions on Multimedia, IEEE Transactions on Information Forensics and Security, ICML, NeurIPS, CVPR, ICCV, AAAI, and IJCAI. More information can be found at https://xinwangliu.github.io/.

Localized Incomplete Multiple Kernel k-Means With Matrix-Induced Regularization

Miaomiao Li, Jingyuan Xia, Huiying Xu, Qing Liao, *Member, IEEE*, Xinzhong Zhu, *Member, IEEE*, and Xinwang Liu, *Senior Member, IEEE*

***Abstract*—Localized incomplete multiple kernel k-means (LI-MKKM) is recently put forward to boost the clustering accuracy via optimally utilizing a quantity of prespecified incomplete base kernel matrices. Despite achieving significant achievement in a variety of applications, we find out that LI-MKKM does not sufficiently consider the diversity and the complementary of the base kernels. This could make the imputation of incomplete kernels less effective, and vice versa degrades on the subsequent clustering. To tackle these problems, an improved LI-MKKM, called LI-MKKM with matrix-induced regularization (LI-MKKM-MR), is proposed by incorporating a matrix-induced regularization term to handle the correlation among base kernels. The incorporated regularization term is beneficial to decrease the probability of simultaneously selecting two similar kernels and increase the probability of selecting two kernels with moderate differences. After that, we establish a three-step iterative algorithm to solve the corresponding optimization objective and analyze its convergence. Moreover, we theoretically show that the local kernel alignment is a special case of its global one with normalizing each base kernel matrices. Based on the above observation, the generalization error bound of the proposed algorithm is derived to theoretically justify its effectiveness. Finally, extensive experiments on several public datasets have been conducted to evaluate the clustering performance of the LI-MKKM-MR. As indicated, the experimental results have demonstrated that our algorithm consistently outperforms the state-of-the-art ones, verifying the superior performance of the proposed algorithm.**

***Index Terms*—Incomplete kernel learning, multiple kernel clustering (MKC), multiple view learning.**

Manuscript received January 13, 2021; revised September 14, 2021; accepted November 6, 2021. This work was supported in part by the National Key Research and Development Program of China under Project 2020AAA0107100; in part by the Natural Science Foundation of China under Project 61922088, Project 61773392, Project 61872377, and Project 61976196. This article was recommended by Associate Editor A. F. Skarmeta Gómez. *(Miaomiao Li, Jingyuan Xia, and Huiying Xu contributed equally to this work.) (Corresponding authors: Miaomiao Li; Xinzhong Zhu.)*

Miaomiao Li is with the College of Electronic Information and Electrical Engineering, Changsha University, Changsha 410073, China (e-mail: miaomiaolinudt@gmail.com).

Jingyuan Xia is with the Department of Electric and Electronic Engineering, Imperial College London, London SW72AZ, U.K.

Huiying Xu is with the College of Mathematics and Computer Science, Zhejiang Normal University, Jinhua 321004, China (e-mail: xyh@zjnu.edu.cn).

Qing Liao is with the Department of Computer Science and Technology, Harbin Institute of Technology (Shenzhen), Shenzhen 518055, China.

Xinzhong Zhu is with the College of Mathematics and Computer Science, Zhejiang Normal University, Jinhua 321004, China, and also with the Department of Artificial Intelligence, Research Institute of Ningbo Cixing Company Ltd., Ningbo 315336, China (e-mail: zxz@zjnu.edu.cn).

Xinwang Liu is with the School of Computer, National University of Defense Technology, Changsha 410073, China.

This article has supplementary material provided by the authors and color versions of one or more figures available at https://doi.org/10.1109/TCYB.2021.3126727.

Digital Object Identifier 10.1109/TCYB.2021.3126727

I. Introduction

Multiple kernel clustering (MKC) [1]–[8] sufficiently integrates a number of precalculated base kernel matrices to group samples into clusters, where similar samples are in the same cluster while dissimilar ones are partitioned into different ones. MKC has attracted much attention of the data mining researchers and has been widely studied in recent years [9]–[17]. The seminal work in [9] extends the multiple kernel learning from supervised learning to unsupervised learning and proposes a margin-based MKC algorithm. It jointly optimizes the optimal kernel, the maximum margin hyperplane, and the optimal clustering labels. The widely used kernel k-means method has been extended in [18] for clustering analysis, where an optimal kernel is learned from multiple data sources. Similarly, the work in [12] extends the existing multiple kernel k-means (MKKM) algorithm by designing a localized MKKM one in order to well utilize the characteristics of each individual sample. To enhance the robustness of the existing MKKM algorithms to noisy data, Du *et al.* [13] proposed a robust MKKM algorithm by substituting the widely adopted squared error in the existing k-means with an $\ell_{2,1}$-norm one, and simultaneously optimized the best combination of kernels. To increase the diversity and decrease the redundancy of the selected base kernels, the recent work in [14] extends the existing MKKM algorithms by designing a matrix-induced regularization term to sufficiently explore the correlation among the prespecified base kernels. More recently, an optimal neighborhood kernel clustering (ONKC) algorithm is proposed in [19], where the representability of the optimal kernel to learn is largely boosted and the negotiation between kernel learning and clustering is also reinforced. The aforementioned MKC algorithms have been applied into many cases and reached a superior performance [15], [20]–[23].

As observed, these MKC algorithms share a common assumption: all the prespecified base kernels are complete. Nevertheless, in some real-world applications, such as image fusion [24], image retrieval [25], and document/video analysis [26], some views of a sample are usually not collected due to various reasons [27], [28]. To address this issue, the work in the literature proposes to first impute the missing elements in base kernel matrices with imputation methods and then performs the existing MKC on these imputed kernels. Several commonly used filling methods include zero-filling, mean value filling, k-nearest-neighbor filling (KNN), expectation-maximization (EM) filling [29], as well as several recently proposed to matrix imputation [30]–[33].

One disadvantage existing in the aforementioned "two-stage" algorithms is that the imputation is separated from the subsequent clustering. As a result, this may not be conducive to mutual negotiation between the imputation and clustering to reach the best performance. To overcome the above issue, the more recent literature [34]–[36] advocates to unify the learning procedure of imputation and clustering into a common framework, with the aim to learn an optimal imputation that best severe for the clustering tasks.

Although demonstrating superior clustering results in several practical applications, we find that these works do not sufficiently consider the redundancy and diversity among prespecified kernel matrices when performing incomplete MKC. This could lead to high redundancy and low diversity among the selected kernels [14], making the utilization ratio of these base kernel matrices insufficient and conversely decreasing the accuracy of clustering tasks. In our work, a localized incomplete MKKM with matrix-induced regularization (LI-MKKM-MR) is proposed to address the above-mentioned issue. By incorporating matrix-induced regularization, LI-MKKM-MR is able to avoid selecting two similar kernel matrices simultaneously and increase the probability of selecting two kernel matrices with large diversity, making the base kernels better utilized for clustering. In addition, it inherits the advantage of localized incomplete multiple kernel k-means (LI-MKKM) which only requires that the similarity of each sample to its top k-nearest neighbors be optimally aligned with the corresponding patch of the entire ideal similarity. This is helpful for LI-MKKM-MR to pay more attention on closer pairwise sample similarities that shall be put together, and prevents involving unreliable similarity evaluation for farther sample pairs. Furthermore, a three-step iterative optimization algorithm is designed to solve the corresponding optimization objective and its convergence has also been analyzed. After that, the generalization error bound of the clustering algorithm is derived, which theoretically guarantees its effectiveness. Comprehensive experiments on several public datasets have been conducted to evaluate the clustering performance of the proposed LI-MKKM-MR. As demonstrated, LI-MKKM-MR significantly and consistently outperforms the existing two-step-based algorithms and the newly proposed algorithm [36]. Extensive experimental results have demonstrated the superiority of involving the matrix-induced regularization.

To summarize, this work makes the following major contributions.

1) This is the first attempt to identify the kernel redundancy problem in *incomplete* MKC. We then introduce a new algorithm to improve LI-MKKM by integrating matrix-induced regularization to select low-redundant and high-diverse kernel matrices and carefully establish three-step iterative algorithm to solve the corresponding optimization objective.
2) We build the theoretical connection between global and local kernel alignment criteria, then we further derive the generalization error bound of the proposed LI-MKKM-MR, which theoretically justifies its effectiveness.
3) Comprehensive experiments on ten public datasets have demonstrated that our LI-MKKM-MR achieves the state-of-the-art performance compared with the existing advanced algorithms. This considerably verifies our identification of the aforementioned issue and the effectiveness of our solution.

Finally, we clarify the differences between LI-MKKM-MR and several recently proposed related work [14], [35]. The differences between LI-MKKM [35] and LI-MKKM-MR can be summarized from the following three aspects.

1) LI-MKKM [35] does not sufficiently consider the diversity and the complementarity of these incomplete base kernels. This could make the imputation of incomplete kernels less effective, and incur the adverse effect on the subsequent clustering. Differently, LI-MKKM-MR is proposed by incorporating matrix-induced regularization, which is helpful to reduce the probability of simultaneously selecting two similar kernels and increase the probability of selecting two kernels with moderate differences, making the base kernels better utilized for clustering.
2) Compared to LI-MKKM [35], LI-MKKM-MR provides the generalization error analysis, which measures the clustering performance of the learned clusters in the training procedure on unseen samples. This theoretically justifies the effectiveness of the proposed LI-MKKM-MR.
3) As observed from the experimental results in Section IV, LI-MKKM-MR significantly improves the clustering performance of LI-MKKM [35] in various benchmark datasets, which well validates our identification of the aforementioned issue in LI-MKKM and the effectiveness of our solution.

We then summarize the differences between [14] and our work from the following aspects. In [14], matrix-induced regularization is proposed to solve the kernel redundancy in MKC. However, it cannot effectively solve MKC with incomplete kernels. Differently, the proposed LI-MKKM-MR makes the first attempt to identify the kernel redundancy problem in *incomplete MKC*, proposes an effective solution, and conducts comprehensive experiments to validate our identification of this issue and the superiority of our algorithm.

II. Related Work

In this part, we mainly introduce the methods of MKKM clustering, MKKM with incomplete kernels (MKKM-IK), and its localized variant. Before introducing these algorithms, we present all notations which will be used in the following in Table I.

TABLE I
NOTATIONS SUMMARY

$\{\mathbf{x}_i\}_{i=1}^n$	n training samples
k	number of clusters
τ	ratio of the nearest neighbors
$\boldsymbol{\gamma} = [\gamma_1, \cdots, \gamma_m]^\top$	kernel weights
$\kappa_p(\cdot,\cdot)$	the p-th kernel function
$\phi_p(\cdot)$	feature mapping corresponding to $\kappa_p(\cdot,\cdot)$
$\phi_{\boldsymbol{\gamma}}(\cdot)$	feature mapping corresponding to $\kappa_{\boldsymbol{\gamma}}(\cdot,\cdot)$
$\{\mathbf{K}_p\}_{p=1}^m$	m base kernel matrices
$\mathbf{e}_p$	observed sample indices of $\mathbf{K}_p$
$\mathbf{H}$	partition matrix
$\mathbf{K}_p^{(dd)}$	sub-matrix of $\mathbf{K}_p$ for observed samples
$\mathbf{U}^{(i)} \in \{0,1\}^{n \times \mathrm{round}(n*\tau)}$	neighborhood indication matrix of $\mathbf{x}_i$
$\mathbf{M}$	correlation matrix among m base kernels
$\hat{\mathbf{C}} = [\hat{\mathbf{C}}_1, \cdots, \hat{\mathbf{C}}_k]$	the learned k centroids

A. Multiple Kernel k-Means

Let $\{\mathbf{x}_i\}_{i=1}^n \subseteq \mathcal{X}$ be n training samples, and $\phi_p(\cdot) : \mathbf{x} \in \mathcal{X} \mapsto \mathcal{H}_p$, $\mathbf{x}$ are mapped onto a reproducing kernel Hilbert space $\mathcal{H}_p$ $(1 \le p \le m)$ by the pth feature. Each sample in MKC is represented by $\phi_{\boldsymbol{\gamma}}(\mathbf{x}) = [\gamma_1\phi_1^\top(\mathbf{x}), \ldots, \gamma_m\phi_m^\top(\mathbf{x})]^\top$, where $\boldsymbol{\gamma} = [\gamma_1, \ldots, \gamma_m]^\top$ represents the weights of m prespecified base kernel functions $\{\kappa_p(\cdot,\cdot)\}_{p=1}^m$. These kernel weights will be adaptively adjusted during MKC. Under the aforementioned definition of $\phi_{\boldsymbol{\gamma}}(\mathbf{x})$, the corresponding kernel function can be expressed as follows:

$$\kappa_{\boldsymbol{\gamma}}\left(\mathbf{x}_i, \mathbf{x}_j\right) = \phi_{\boldsymbol{\gamma}}^\top(\mathbf{x}_i)\phi_{\boldsymbol{\gamma}}(\mathbf{x}_j) = \sum_{p=1}^m \gamma_p^2 \kappa_p\left(\mathbf{x}_i, \mathbf{x}_j\right). \tag{1}$$

One can calculate a kernel matrix $\mathbf{K}_{\boldsymbol{\gamma}}$ on training samples $\{\mathbf{x}_i\}_{i=1}^n$ with the kernel function defined in (1). As a result, the objective of MKKM with $\mathbf{K}_{\boldsymbol{\gamma}}$ is formulated as

$$\begin{aligned} &\min_{\mathbf{H},\boldsymbol{\gamma}} \ \mathrm{Tr}\left(\mathbf{K}_{\boldsymbol{\gamma}}\left(\mathbf{I}_n - \mathbf{H}\mathbf{H}^\top\right)\right) \\ &\text{s.t.} \ \mathbf{H}^\top\mathbf{H} = \mathbf{I}_k, \ \boldsymbol{\gamma}^\top\mathbf{1}_m = 1, \ \gamma_p \ge 0 \quad \forall p \end{aligned} \tag{2}$$

where $\mathbf{H} \in \mathbb{R}^{n\times k}$ is a soft version of the cluster assignment matrix, and $\mathbf{I}_k$ is a $k \times k$ identity matrix. Alternately updating $\mathbf{H}$ and $\boldsymbol{\gamma}$ can optimize (2).

Optimizing $\mathbf{H}$ *With Fixed* $\boldsymbol{\gamma}$*:* With $\boldsymbol{\gamma}$ fixed, the optimization in (2) toward $\mathbf{H}$ is exactly the traditional kernel k-means presented in

$$\max_{\mathbf{H}} \ \mathrm{Tr}\left(\mathbf{H}^\top\mathbf{K}_{\boldsymbol{\gamma}}\mathbf{H}\right) \ \text{s.t.} \ \mathbf{H} \in \mathbb{R}^{n\times k}, \mathbf{H}^\top\mathbf{H} = \mathbf{I}_k. \tag{3}$$

The optimal $\mathbf{H}$ in (3) consists of k eigenvectors corresponding to the top-k eigenvalues of $\mathbf{K}_{\boldsymbol{\gamma}}$ [37].

Optimizing $\boldsymbol{\gamma}$ *With Fixed* $\mathbf{H}$*:* With $\mathbf{H}$ fixed, the equivalent form of optimization in (2) with regard to $\boldsymbol{\gamma}$ is as follows:

$$\min_{\boldsymbol{\gamma}} \ \sum_{p=1}^m \gamma_p^2 \mathrm{Tr}\left(\mathbf{K}_p\left(\mathbf{I}_n - \mathbf{H}\mathbf{H}^\top\right)\right) \ \text{s.t.} \ \boldsymbol{\gamma}^\top\mathbf{1}_m = 1, \ \gamma_p \ge 0 \tag{4}$$

which has a closed-form solution.

B. MKKM With Incomplete Kernels

MKKM has recently been extended to handle incomplete MKC in [34] and [36]. Previous algorithms first manage to impute the incomplete kernel matrices and then apply the existing MKKM on the imputed kernel matrices. In contrast, they propose to unify the learning process of imputation and clustering into a common learning framework and establish an effective optimization algorithm to optimize each of them alternately. In MKKM-IK, the clustering procedure provides a guidance for the imputation of the incomplete base kernel matrices, and the clustering is further enhanced by the imputed kernels. Both procedures are alternated performed until achieving optimal results. The above idea can be achieved as follows:

$$\begin{aligned} &\min_{\mathbf{H},\,\boldsymbol{\gamma},\,\{\mathbf{K}_p\}_{p=1}^m} \ \mathrm{Tr}\left(\mathbf{K}_{\boldsymbol{\gamma}}(\mathbf{I}_n - \mathbf{H}\mathbf{H}^\top)\right) \\ &\text{s.t.} \ \mathbf{H} \in \mathbb{R}^{n\times k}, \mathbf{H}^\top\mathbf{H} = \mathbf{I}_k \\ &\quad\ \ \boldsymbol{\gamma}^\top\mathbf{1}_m = 1, \gamma_p \ge 0 \\ &\quad\ \ \mathbf{K}_p(\mathbf{e}_p, \mathbf{e}_p) = \mathbf{K}_p^{(dd)}, \ \mathbf{K}_p \succeq 0 \quad \forall p \end{aligned} \tag{5}$$

where $\mathbf{e}_p$ $(1 \le p \le m)$ denotes the sample indices, the p-th view is observed, and $\mathbf{K}_p^{(dd)}$ denotes the kernel submatrix. Note that we impose the constraint $\mathbf{K}_p(\mathbf{e}_p, \mathbf{e}_p) = \mathbf{K}_p^{(dd)}$ to make the known entries of $\mathbf{K}_p$ kept unchanged during the learning course. The imputation of incomplete kernels can be regarded as a by-product of learning, because the ultimate goal of (5) is clustering.

A trilevel optimization strategy developed in [34] develops to solve (5) alternately.

Optimizing $\mathbf{H}$ *With* $\boldsymbol{\gamma}$ *and* $\{\mathbf{K}_p\}_{p=1}^m$ *Fixed:* Given $\boldsymbol{\gamma}$ and $\{\mathbf{K}_p\}_{p=1}^m$, the optimization in (5) with respect to $\mathbf{H}$ is equivalent to a kernel k-means problem solved by (3).

Optimizing $\{\mathbf{K}_p\}_{p=1}^m$ *With* $\boldsymbol{\gamma}$ *and* $\mathbf{H}$ *Fixed:* Given $\boldsymbol{\gamma}$ and $\mathbf{H}$, (5) toward each $\mathbf{K}_p$ is equivalently reformulated as follows:

$$\begin{aligned} &\min_{\mathbf{K}_p} \ \mathrm{Tr}\left(\mathbf{K}_p(\mathbf{I}_n - \mathbf{H}\mathbf{H}^\top)\right) \\ &\text{s.t.} \ \mathbf{K}_p\left(\mathbf{e}_p, \mathbf{e}_p\right) = \mathbf{K}_p^{(dd)}, \ \mathbf{K}_p \succeq 0. \end{aligned} \tag{6}$$

It is proven in [34] that the optimal $\mathbf{K}_p$ in (6) has the closed-form solution as in (7), shown at the bottom of the page, where $\mathbf{Z} = \mathbf{I}_n - \mathbf{H}\mathbf{H}^\top$ and taking the elements of $\mathbf{Z}$ corresponding to the observed and unobserved sample indices can construct $\mathbf{Z}^{(dm)}$. For more details, refer to [34].

Optimizing $\boldsymbol{\gamma}$ *With* $\mathbf{H}$ *and* $\{\mathbf{K}_p\}_{p=1}^m$ *Fixed:* Given $\mathbf{H}$ and $\{\mathbf{K}_p\}_{p=1}^m$, (5) with respect to $\boldsymbol{\gamma}$ reduces to a quadratic programming (QP) with linear constraints.

C. Localized Incomplete MKKM

Although it is ingenious to unify clustering and imputation into one learning process, which is achieved by *globally* maximizing the alignment between the optimal kernel matrix $\mathbf{K}_{\boldsymbol{\gamma}}$

$$\mathbf{K}_p = \begin{bmatrix} \mathbf{K}_p^{(dd)} & -\mathbf{K}_p^{(dd)}\mathbf{Z}^{(dm)}(\mathbf{Z}^{(mm)})^{-1} \\ -(\mathbf{Z}^{(mm)})^{-1}\mathbf{Z}^{(dm)\top}\mathbf{K}_p^{(dd)} & (\mathbf{Z}^{(mm)})^{-1}\mathbf{Z}^{(dm)\top}\mathbf{K}_p^{(dd)}\mathbf{Z}^{(dm)}(\mathbf{Z}^{(mm)})^{-1} \end{bmatrix} \tag{7}$$

and the ideal matrix $\mathbf{H}\mathbf{H}^\top$, as presented in (2). This criterion does not take full advantage of the local distribution of data, and requires that all paired samples, whether closer or farther, should be consistent with the ideal similarity without distinction.

Instead of calculating the alignment between the optimal kernel and the idea matrix in a global manner as in (5), localized incomplete MKKM (LI-MKKM) [35] is proposed to utilize the local structure among data by only requiring the similarity of each sample to align with its nearest neighbors. Specifically, the objective function of LI-MKKM is as follows:

$$\begin{aligned}
\min_{\boldsymbol{\gamma},\{\mathbf{K}_p\}_{p=1}^m,\mathbf{H}} \quad & \sum_{i=1}^{n} \mathrm{Tr}\Big(\mathbf{K}_{\boldsymbol{\gamma}}\Big(\mathbf{A}^{(i)} - \mathbf{A}^{(i)}\mathbf{H}\mathbf{H}^\top\mathbf{A}^{(i)}\Big)\Big)\\
\text{s.t.} \quad & \mathbf{H} \in \mathbb{R}^{n\times k},\ \mathbf{H}^\top\mathbf{H} = \mathbf{I}_k,\ \boldsymbol{\gamma}^\top\mathbf{1}_m = 1,\ \gamma_p \geq 0\\
& \mathbf{K}_p\big(\mathbf{e}_p,\mathbf{e}_p\big) = \mathbf{K}_p^{(dd)},\ \mathbf{K}_p \succeq 0 \quad \forall p
\end{aligned} \tag{8}$$

where $\mathbf{A}^{(i)} = \mathbf{U}^{(i)}\mathbf{U}^{(i)\top}$ with $\mathbf{U}^{(i)} \in \{0,1\}^{n\times \mathrm{round}(n*\tau)}$ $(1 \leq i \leq n)$ denoting the neighborhood index matrix of the ith sample. $\mathbf{U}_{jv}^{(i)} = 1$ represents that $\mathbf{x}_j$ is the vth nearest neighbor of $\mathbf{x}_i$, where $1 \leq v \leq \mathrm{round}(n*\tau)$ and τ is the ratio of the nearest neighbors.

Similar to [34], the work in [35] develops a tristep optimization algorithm to solve (8) and theoretically proves its convergence. Refer to [35] for more details.

III. Localized Incomplete Multiple Kernel k-Means With Matrix-Induced Regularization

A. Formulation

Although aligning the optimal kernel with the ideal similarity locally can improve the clustering performance, LI-MKKM dose not explicitly take the correlation among base kernels into account. This would prevent these incomplete base kernels from being well utilized. To overcome this problem, we propose an improved algorithm based on LI-MKKM via introducing matrix-induced regularization $\boldsymbol{\gamma}^\top\mathbf{M}\boldsymbol{\gamma}$ to decrease the redundancy and enhance the diversity of the selected base kernels, where M_{pq} measures the correlation between $\mathbf{K}_p$ and $\mathbf{K}_q$. By integrating this regularization into (8), the following objective is obtained:

$$\begin{aligned}
\min_{\boldsymbol{\gamma},\{\mathbf{K}_p\}_{p=1}^m,\mathbf{H}} \quad & \sum_{i=1}^{n} \mathrm{Tr}\Big(\mathbf{K}_{\boldsymbol{\gamma}}(\mathbf{A}^{(i)} - \mathbf{A}^{(i)}\mathbf{H}\mathbf{H}^\top\mathbf{A}^{(i)})\Big) + \frac{\lambda}{2}\boldsymbol{\gamma}^\top\mathbf{M}\boldsymbol{\gamma}\\
\text{s.t.} \quad & \mathbf{H} \in \mathbb{R}^{n\times k},\ \mathbf{H}^\top\mathbf{H} = \mathbf{I}_k\\
& \boldsymbol{\gamma}^\top\mathbf{1}_m = 1,\ \gamma_p \geq 0\\
& \mathbf{K}_p(\mathbf{e}_p,\mathbf{e}_p) = \mathbf{K}_p^{(dd)},\ \mathbf{K}_p \succeq 0 \quad \forall p
\end{aligned} \tag{9}$$

where λ is a hyper-parameter to balance the regularization on kernel weights and the loss of local kernel k-means.

In this work, we adopt $M_{pq} = \mathrm{Tr}(\mathbf{K}_p\mathbf{K}_q)$ to measure the correlation between $\mathbf{K}_p$ and $\mathbf{K}_q$. On one hand, the incorporation of $\boldsymbol{\gamma}^\top\mathbf{M}\boldsymbol{\gamma}$ is helpful for well utilizing the base kernels, which is utilized to boost the clustering performance. On the other hand, it makes the resultant optimization more challenging since the optimization on each $\mathbf{K}_p$ is a quadratic semi-defined programming, whose computational cost is intensive and this prevents it from being applied to practical applications. To reduce the computation overhead of (9), we propose to approximate M_{pq} by $\tilde{M}_{pq} = \mathrm{Tr}(\mathbf{K}_p^{(0)}\mathbf{K}_q^{(0)})$ and keep it unchanged during the learning course, where $\mathbf{K}_p^{(0)}$ is an initial imputation of $\mathbf{K}_p$. By substituting $\mathbf{M}$ with $\tilde{\mathbf{M}}$, the objective function of the proposed LI-MKKM-MR can be expressed as follows:

$$\begin{aligned}
\min_{\boldsymbol{\gamma},\{\mathbf{K}_p\}_{p=1}^m,\mathbf{H}} \quad & \sum_{i=1}^{n} \mathrm{Tr}\Big(\mathbf{K}_{\boldsymbol{\gamma}}\Big(\mathbf{A}^{(i)} - \mathbf{A}^{(i)}\mathbf{H}\mathbf{H}^\top\mathbf{A}^{(i)}\Big)\Big) + \frac{\lambda}{2}\boldsymbol{\gamma}^\top\tilde{\mathbf{M}}\boldsymbol{\gamma}\\
\text{s.t.} \quad & \mathbf{H} \in \mathbb{R}^{n\times k},\ \mathbf{H}^\top\mathbf{H} = \mathbf{I}_k\\
& \boldsymbol{\gamma}^\top\mathbf{1}_m = 1,\ \gamma_p \geq 0\\
& \mathbf{K}_p(\mathbf{e}_p,\mathbf{e}_p) = \mathbf{K}_p^{(dd)},\ \mathbf{K}_p \succeq 0 \quad \forall p.
\end{aligned} \tag{10}$$

It is reasonable to measure the correlation of pairwise kernels via observed similarity. Consequently, the approximation $\tilde{\mathbf{M}}$ can be regarded as a prior of $\mathbf{M}$. Also, although this approximation is simple, its advantages are three-folds. First, it fulfills our requirement on the kernel coefficients to enhance the diversity and decrease the redundancy. Second, it simplifies the optimization on $\{\mathbf{K}_p\}_{p=1}^m$, making it admit a closed-form solution. This significantly increases the computational cost. Third, the effectiveness of the proposed approximation can be demonstrated by experiments.

Although the matrix-induced regularization may be exploited in other related aspects, such as MKC [14], this is the first work in literature to study the regularization on incomplete MKC and design a reasonable approximation for the convenience of computation. Moreover, this would trigger more research on incomplete MKC, such as designing more informative $\mathbf{M}$, updating $\mathbf{M}$ with learned kernel weights and the imputation at each iteration, to name just a few. More importantly, our experimental study shows that the incorporation of matrix-induced regularization helps to utilize the incomplete kernels, leading to significantly improvement on clustering performance. This makes the proposed algorithm a good choice in real-world applications, such as cancer biology [12], analysis of multiple heterogeneous neuroimaging data [38], and Alzheimer's disease diagnosis [39]. In the following, we develop a tristep optimization strategy to solve it alternately in the following parts.

B. Alternate Optimization of LI-MKKM-MR

Optimizing $\mathbf{H}$ *With* $\boldsymbol{\gamma}$ *and* $\{\mathbf{K}_p\}_{p=1}^m$ *Fixed:* Given $\boldsymbol{\gamma}$ and $\{\mathbf{K}_p\}_{p=1}^m$, the optimization objective w.r.t $\mathbf{H}$ in (10) redefines to

$$\begin{aligned}
\max_{\mathbf{H}} \quad & \mathrm{Tr}\left(\mathbf{H}^\top \sum_{i=1}^{n}\Big(\mathbf{A}^{(i)}\mathbf{K}_{\boldsymbol{\gamma}}\mathbf{A}^{(i)}\Big)\mathbf{H}\right)\\
\text{s.t.} \quad & \mathbf{H} \in \mathbb{R}^{n\times k},\ \mathbf{H}^\top\mathbf{H} = \mathbf{I}_k
\end{aligned} \tag{11}$$

which is transformed into a classical kernel k-means-based optimization objective and can be conveniently tackled by the existing public toolkit.

Optimizing $\{\mathbf{K}_p\}_{p=1}^m$ *With* $\boldsymbol{\gamma}$ *and* $\mathbf{H}$ *Fixed:* Given $\boldsymbol{\gamma}$ and $\mathbf{H}$, the optimization objective w.r.t $\{\mathbf{K}_p\}_{p=1}^m$ in (10) can be formulated as

$$\min_{\{\mathbf{K}_p\}_{p=1}^m} \sum_{p=1}^m \gamma_p^2 \mathrm{Tr}\left(\mathbf{K}_p \sum_{i=1}^n \mathrm{Tr}\left(\mathbf{A}^{(i)} - \mathbf{A}^{(i)}\mathbf{H}\mathbf{H}^\top \mathbf{A}^{(i)}\right)\right)$$
$$\text{s.t. } \mathbf{K}_p(\mathbf{e}_p, \mathbf{e}_p) = \mathbf{K}_p^{(dd)},\ \mathbf{K}_p \succeq 0 \quad \forall p. \tag{12}$$

It is difficult to solve the optimization problem in (12) since there are multiple kernel matrices to be optimized simultaneously. By cautiously analyzing the optimization, we observe that: 1) each kernel matrix $\mathbf{K}_p$ has its own separate constraint and 2) the objective in (12) is a sum generated by calculating $\mathbf{K}_p$. As a result, (12) can be reformulated as m uncorrelated subobjectives equivalently, as shown in the following:

$$\min_{\mathbf{K}_p} \ \mathrm{Tr}(\mathbf{K}_p\mathbf{Q})$$
$$\text{s.t. } \mathbf{K}_p(\mathbf{e}_p, \mathbf{e}_p) = \mathbf{K}_p^{(dd)},\ \mathbf{K}_p \succeq 0 \tag{13}$$

where $\mathbf{Q} = \sum_{i=1}^n (\mathbf{A}^{(i)} - \mathbf{A}^{(i)}\mathbf{H}\mathbf{H}^\top \mathbf{A}^{(i)})$.

It seems that directly solving (13) is difficult because of the equality and PSD constraints imposed on $\mathbf{K}_p$. By following [35], we parameterize each $\mathbf{K}_p$ as:

$$\mathbf{K}_p = \begin{bmatrix} \mathbf{K}_p^{(dd)} & \mathbf{K}_p^{(dd)}\mathbf{Z}_p \\ \mathbf{Z}_p^\top \mathbf{K}_p^{(dd)} & \mathbf{Z}_p^\top \mathbf{K}_p^{(dd)}\mathbf{Z}_p \end{bmatrix} \tag{14}$$

where $\mathbf{Z}_p \in \mathbb{R}^{d\times m}$. d and m refer to the number of observed samples and unobserved ones, respectively. With (14), we assume that the observed ones represent the missing kernel entries. It is shown in [35] that $\mathbf{K}_p$ in (14) automatically satisfies both constraints after this parametrization.

Based on the parametrization in (14), the constrained optimization in (13) is equivalent to

$$\min_{\mathbf{Z}_p} \ \mathrm{Tr}\left(\begin{bmatrix} \mathbf{K}_p^{(dd)} & \mathbf{K}_p^{(dd)}\mathbf{Z}_p \\ \mathbf{Z}_p^\top \mathbf{K}_p^{(dd)} & \mathbf{Z}_p^\top \mathbf{K}_p^{(dd)}\mathbf{Z}_p \end{bmatrix}\begin{bmatrix} \mathbf{Q}^{(dd)} & \mathbf{Q}^{(dm)} \\ \mathbf{Q}^{(dm)\top} & \mathbf{Q}^{(mm)} \end{bmatrix}\right) \tag{15}$$

where $\mathbf{Q}$ is decomposed into the following submatrices $\begin{bmatrix} \mathbf{Q}^{(dd)} & \mathbf{Q}^{(dm)} \\ \mathbf{Q}^{(dm)\top} & \mathbf{Q}^{(mm)} \end{bmatrix}$.

To minimize (15), we take its derivative with respect to $\mathbf{Z}_p$ and let it vanish, leading to

$$\mathbf{Z}_p = -\mathbf{Q}^{(dm)}\left(\mathbf{Q}^{(mm)}\right)^{-1}. \tag{16}$$

As a result, we obtain an analytical solution for the optimal $\mathbf{K}_p$ by substituting $\mathbf{Z}_p$ in (16) into (14). As seen, (13) provides a guidance for the imputation of each base kernel by exploring the data structure in a local manner. Specifically, it locally estimates the alignment between the similarity of each sample and its τ-nearest neighbors with the corresponding ideal matrix. This enables the proposed algorithm to better utilize the intracluster variations among samples. Therefore, the clustering performance could be improved, mainly attributing to an effective incomplete kernels imputation measure.

Optimizing $\boldsymbol{\gamma}$ With $\{\mathbf{K}_p\}_{p=1}^m$ and $\mathbf{H}$ Fixed: Given $\{\mathbf{K}_p\}_{p=1}^m$ and $\mathbf{H}$, it is easy to present that (10) w.r.t. $\boldsymbol{\gamma}$ is as follows:

$$\min_{\boldsymbol{\gamma}} \ \frac{1}{2}\boldsymbol{\gamma}^\top\left(2\mathbf{W} + \lambda\tilde{\mathbf{M}}\right)\boldsymbol{\gamma}$$
$$\text{s.t. } \boldsymbol{\gamma}^\top \mathbf{1}_m = 1,\ \gamma_p \geq 0 \tag{17}$$

Algorithm 1 Proposed LI-MKKM-MR

```
Input: {K_p^(dd)}_{p=1}^m, {e_p}_{p=1}^m, k, τ, λ and ε_0.
Output: H, γ and {K_p}_{p=1}^m.
Initialize γ^(0) = 1_m/m, {K_p^(0)}_{p=1}^m and t = 1.
Generate U^(i) for i-th samples (1 ≤ i ≤ n) by K_{γ^(0)}.
Calculate A^(i) = U^(i) U^(i)⊤ for i-th samples (1 ≤ i ≤ n).
repeat
    K_{γ^(t)} = Σ_{p=1}^m (γ_p^(t−1))^2 K_p^(t−1).
    Update H^(t) by solving Eq. (11) with K_{γ^(t)}.
    Update {K_p^(t)}_{p=1}^m with H^(t) by Eq. (13).
    Update γ^(t) by solving Eq. (17) with H^(t) and {K_p^(t)}_{p=1}^m.
    t = t + 1.
until (obj^(t−1) − obj^(t))/obj^(t) ≤ ε_0
```

where $\mathbf{W} = \mathrm{diag}([\mathrm{Tr}(\mathbf{K}_1\mathbf{Q}), \ldots, \mathrm{Tr}(\mathbf{K}_m\mathbf{Q})])$. Theorem 1 in the following indicates that $\mathbf{W}$ is PSD.

Theorem 1: The Hessian matrix $2\mathbf{W} + \lambda\tilde{\mathbf{M}}$ in (17) is a symmetric PSD matrix.

Proof: By defining $\mathbf{H} = [\mathbf{h}_1, \ldots, \mathbf{h}_k]$, we can find out that $\mathbf{H}\mathbf{H}^\top \mathbf{h}_c = \mathbf{h}_c (1 \leq c \leq k)$ since $\mathbf{H}^\top\mathbf{H} = \mathbf{I}_k$. This indicates that $\mathbf{H}\mathbf{H}^\top$ has k eigenvalue with 1. Besides, its rank does not exceed k. This means that its has $n-k$ eigenvalue with 0. $\mathbf{I}_n - \mathbf{H}\mathbf{H}^\top$ contains $n-k$ eigenvalue with 1 and k eigenvalue with 0. Consequently, $\mathbf{A}^{(i)}(\mathbf{I}_n - \mathbf{H}\mathbf{H}^\top)\mathbf{A}^{(i)}$ is PSD, which ensures that $\mathbf{Q} = \sum_{i=1}^n (\mathbf{A}^{(i)} - \mathbf{A}^{(i)}\mathbf{H}\mathbf{H}^\top\mathbf{A}^{(i)})$ is PSD. As a result, we have $w_p = \mathrm{Tr}(\mathbf{K}_p\mathbf{Q}) \geq 0\ \forall p$, guaranteeing the positiveness of $\mathbf{W}$. Meanwhile, $\mathbf{W}$ is also a symmetric PSD matrix according to [40]. Consequently, $2\mathbf{W}+\lambda\tilde{\mathbf{M}}$ is a symmetric PSD matrix. ■

On the basis of Theorem 1, we can guarantee that the optimization in (17) w.r.t $\boldsymbol{\gamma}$ is a traditional QP with linear constraints. Therefore, it can be conveniently handled by the existing optimization packages.

Algorithm 1 presents an outline of solving (10) by the proposed algorithm, where we adopt the zero-filling method to initially impute the missing elements of $\{\mathbf{K}_p^{(0)}\}_{p=1}^m$ and utilize $\mathrm{obj}^{(t)}$ to represent the objective value at the t-th iteration. Besides, the neighbors of each sample remain unvaried during the optimization procedure in LI-MKKM-MR. In specific, we calculate the τ-nearest neighbors of each sample by $\mathbf{K}_{\boldsymbol{\gamma}^{(0)}}$. In this way, the optimization target of LI-MKKM-MR is guaranteed to be reduced in a monotonic manner when we update one variable and keep the others unchanged iteratively. Simultaneously, the objective is lower bounded by zero. Hence, it is guaranteed that LI-MKKM-MR converges into a local optimal solution. Experimental results have demonstrated that our method usually converges quickly.

The end of this part analyzes the computational complexity of our method. In specific, the computational complexity of LI-MKKM-MR is $\mathcal{O}(n^3 + \sum_{p=1}^m n_p^3 + m^3)$ at each iteration, where n_p ($n_p \leq n$) and m refer to the number of observed samples of $\mathbf{K}_p$ and base kernels. The complexity of LI-MKKM-MR can be compared to that of MKKM-IK [34] and LI-MKKM [35]. Moreover, each sample of $\mathbf{K}_p$ is independent so that they can be measured in a parallel manner. By this means, our LI-MKKM-MR can scale well regardless of the variation of the base kernels number.

C. Theoretical Results

The generalization error of the k-means clustering algorithm has been widely discussed in the existing literature [36], [41], and [42]. We first establish the theoretical connection between the existing MKKM-IK [36] with LI-MKKM-MR, and further derive the generalization error bound of LI-MKKM-MR based on the theoretical results in [36]. The following theorem (Theorem 2) points out that the local kernel alignment adopted in our LI-MKKM-MR can be achieved by normalizing each base kernel matrix.

Theorem 2: The local kernel alignment criterion in (8) is equivalent to the widely adopted global kernel alignment by normalizing each base kernel matrix.

Proof: The objective function in (8) can be written as

$$\begin{aligned}
&\sum_{i=1}^{n} \mathrm{Tr}\Big(\mathbf{K}_{\boldsymbol{\gamma}}\Big(\mathbf{A}^{(i)} - \mathbf{A}^{(i)}\mathbf{H}\mathbf{H}^{\top}\mathbf{A}^{(i)}\Big)\Big) \\
&= \sum_{i=1}^{n} \Big\langle \mathbf{A}^{(i)} \otimes \mathbf{K}_{\boldsymbol{\gamma}}, \mathbf{A}^{(i)} \otimes \Big(\mathbf{I} - \mathbf{H}\mathbf{H}^{\top}\Big)\Big\rangle_{\mathrm{F}} \\
&= \sum_{i=1}^{n} \Big\langle \mathbf{A}^{(i)} \otimes \mathbf{K}_{\boldsymbol{\gamma}}, \mathbf{I} - \mathbf{H}\mathbf{H}^{\top}\Big\rangle_{\mathrm{F}} \\
&= \left\langle \left(\sum_{i=1}^{n} \mathbf{A}^{(i)}\right) \otimes \mathbf{K}_{\boldsymbol{\gamma}}, \mathbf{I} - \mathbf{H}\mathbf{H}^{\top}\right\rangle_{\mathrm{F}} \\
&= \sum_{p=1}^{m} \gamma_p^2 \left\langle \left(\sum_{i=1}^{n} \mathbf{A}^{(i)}\right) \otimes \mathbf{K}_p, \mathbf{I} - \mathbf{H}\mathbf{H}^{\top}\right\rangle_{\mathrm{F}} \\
&= \sum_{p=1}^{m} \gamma_p^2 \Big\langle \tilde{\mathbf{K}}_p, \mathbf{I} - \mathbf{H}\mathbf{H}^{\top}\Big\rangle_{\mathrm{F}} \\
&= \mathrm{Tr}\Big(\tilde{\mathbf{K}}_{\boldsymbol{\gamma}}\Big(\mathbf{I} - \mathbf{H}\mathbf{H}^{\top}\Big)\Big)
\end{aligned} \tag{18}$$

where $\otimes$ denotes elementwise multiplication between two matrices, $\tilde{\mathbf{K}}_p = (\sum_{i=1}^{n} \mathbf{A}^{(i)}) \otimes \mathbf{K}_p$ can be treated as a normalized $\mathbf{K}_p$, and $\tilde{\mathbf{K}}_{\boldsymbol{\gamma}} = \sum_{p=1}^{m} \gamma_p^2 \tilde{\mathbf{K}}_p$. Consequently, by such normalization being applied on each base kernel, we can clearly see that the local kernel alignment criterion in (8) is exactly the global kernel alignment in [36]. This completes the proof. ■

Let $t(\mathbf{x}^{(p)}) = 1$ if the pth view of $\mathbf{x}$ is available; otherwise, $\mathbf{x}^{(p)}$ should be optimized. It is worth pointing out that $t(\mathbf{x}^{(p)})$ is a random variable that depends on $\mathbf{x}$. Let $\hat{\mathbf{C}} = [\hat{\mathbf{C}}_1, \ldots, \hat{\mathbf{C}}_k]$ be the k centroids and $\hat{\boldsymbol{\gamma}}$ be the kernel weights learned by LI-MKKM-MR. k-means clustering should make the reconstruction error small

$$\mathbb{E}\left[\min_{\mathbf{y}\in\{\mathbf{e}_1,\ldots,\mathbf{e}_k\}} \left\|\phi_{\hat{\boldsymbol{\gamma}}}(\mathbf{x}) - \hat{\mathbf{C}}\mathbf{y}\right\|_{\mathcal{H}}^2\right] \tag{19}$$

where $\phi_{\hat{\boldsymbol{\gamma}}}(\mathbf{x}) = [\hat{\gamma}_1 t(\mathbf{x}^{(1)})\phi_1^{\top}(\mathbf{x}^{(1)}), \ldots, \hat{\gamma}_m t(\mathbf{x}^{(m)})\phi_m^{\top}(\mathbf{x}^{(m)})]^{\top}$, $\mathbf{e}_1, \ldots, \mathbf{e}_k$ form the orthogonal bases of $\mathbb{R}^k$.

We first define a function class

$$\begin{aligned}
\mathcal{F} = \Big\{ &f : \mathbf{x} \mapsto \min_{\mathbf{y}\in\{\mathbf{e}_1,\ldots,\mathbf{e}_k\}} \|\phi_{\boldsymbol{\gamma}}(\mathbf{x}) - \mathbf{C}\mathbf{y}\|_{\mathcal{H}}^2 \Big| \boldsymbol{\gamma}^{\top}\mathbf{1}_m = 1,\ \gamma_p \geq 0, \\
&\mathbf{C} \in \mathcal{H}^k,\ t(\mathbf{x}_i^{(p)})t(\mathbf{x}_j^{(p)})\tilde{\kappa}_p^{\top}(\mathbf{x}_i^{(p)}, \mathbf{x}_j^{(p)}) \leq b, \quad \forall p \quad \forall \mathbf{x}_i \in \mathcal{X} \Big\}
\end{aligned} \tag{20}$$

where $\mathcal{H}^k$ represents the multiple kernel Hilbert space and $\tilde{\kappa}(\cdot,\cdot)$ is a kernel function corresponding to $\tilde{\mathbf{K}}_p$.

Based on Theorem 2, we derive the generalization error bound of the proposed LI-MKKM-MR by following [36].

Theorem 3: For any $\delta > 0$, with probability at least $1 - \delta$, the following holds for all $f \in \mathcal{F}$:

$$\begin{aligned}
\mathbb{E}\big[f(\mathbf{x})\big] \leq &\frac{1}{n}\sum_{i=1}^{n} f(\mathbf{x}_i) + \frac{4\sqrt{\pi}mb\mathcal{G}_{1n}(\boldsymbol{\gamma}, t)}{n} + \frac{4\sqrt{\pi}mb\mathcal{G}_{2n}(\boldsymbol{\gamma}, t)}{n} \\
&+ \frac{\sqrt{8\pi}bk^2}{\sqrt{n}} + 2b\sqrt{\frac{\log 1/\delta}{2n}}
\end{aligned} \tag{21}$$

where

$$\mathcal{G}_{1n}(\boldsymbol{\gamma}, t) \triangleq \mathbb{E}_{\gamma}\left[\sup_{\boldsymbol{\gamma},t} \sum_{i=1}^{n} \sum_{p,q=1}^{m} \gamma_{ipq} t(\mathbf{x}_i^{(p)}) t(\mathbf{x}_i^{(q)}) \gamma_p \gamma_q\right] \tag{22}$$

$$\mathcal{G}_{2n}(\boldsymbol{\gamma}, t) = \mathbb{E}_{\gamma}\left[\sup_{\boldsymbol{\gamma},t} \sum_{i=1}^{n} \sum_{c=1}^{k} \sum_{p=1}^{m} \gamma_{icp} \gamma_p t\Big(\mathbf{x}_i^{(p)}\Big)\right] \tag{23}$$

and $\gamma_{ipq}, \gamma_{icp}, i \in \{1, \ldots, n\}, p, q \in \{1, \ldots, m\}, c \in \{1, \ldots, k\}$ are i.i.d. Gaussian random variables with zero mean and unit standard deviation.

According to the analyses in [36], our local kernel alignment criterion in (8), with normalized base kernel matrices, is an upper bound of $1/n \sum_{i=1}^{n} f(\mathbf{x}_i)$. As a result, by minimizing $\mathrm{Tr}(\tilde{\mathbf{K}}_{\boldsymbol{\gamma}}(\mathbf{I}_n - \mathbf{H}\mathbf{H}^{\top}))$, one can obtain a small $1/n \sum_{i=1}^{n} f(\mathbf{x}_i)$ for good generalization. This justifies the good generalization ability of the LI-MKKM-MR. The detailed proof has been presented in the supplementary material.

IV. Experiments

A. Experimental Settings

In our experiments, we adopt ten widely used MKL benchmark datasets to verify the proposed algorithms, including Oxford Flower17 and Flower102,[1] Caltech102,[2] Digital,[3] Protein Fold Prediction,[4] and Reuters.[5] The information of them is shown in Table II. The kernel matrices of these datasets are precomputed and can be directly obtained from the aforementioned link. Caltech102-5 refers to the number of samples belonging to each cluster is 5, and the same for the rest datasets. The publicly access codes for kernel k-means and MKKM can be found in the website.[6]

Several well-known and widely used imputation methods, such as zero filling (ZF), mean filling (MF), KNN, and alignment-maximization filling (AF) are contained in [30]. After that, researchers take the imputed kernel matrices as the input of classical MKKM. The kind of two-stage methods are called MKKM + ZF, MKKM + MF, MKKM + KNN,

[1] http://www.robots.ox.ac.uk/˜+vgg/data/flowers/
[2] http://files.is.tue.mpg.de/pgehler/projects/iccv09/
[3] http://ss.sysu.edu.cn/˜+py/
[4] http://mkl.ucsd.edu/dataset/protein-fold-prediction/
[5] http://kdd.ics.uci.edu/databases/reuters21578/
[6] https://github.com/mehmetgonen/lmkkmeans/

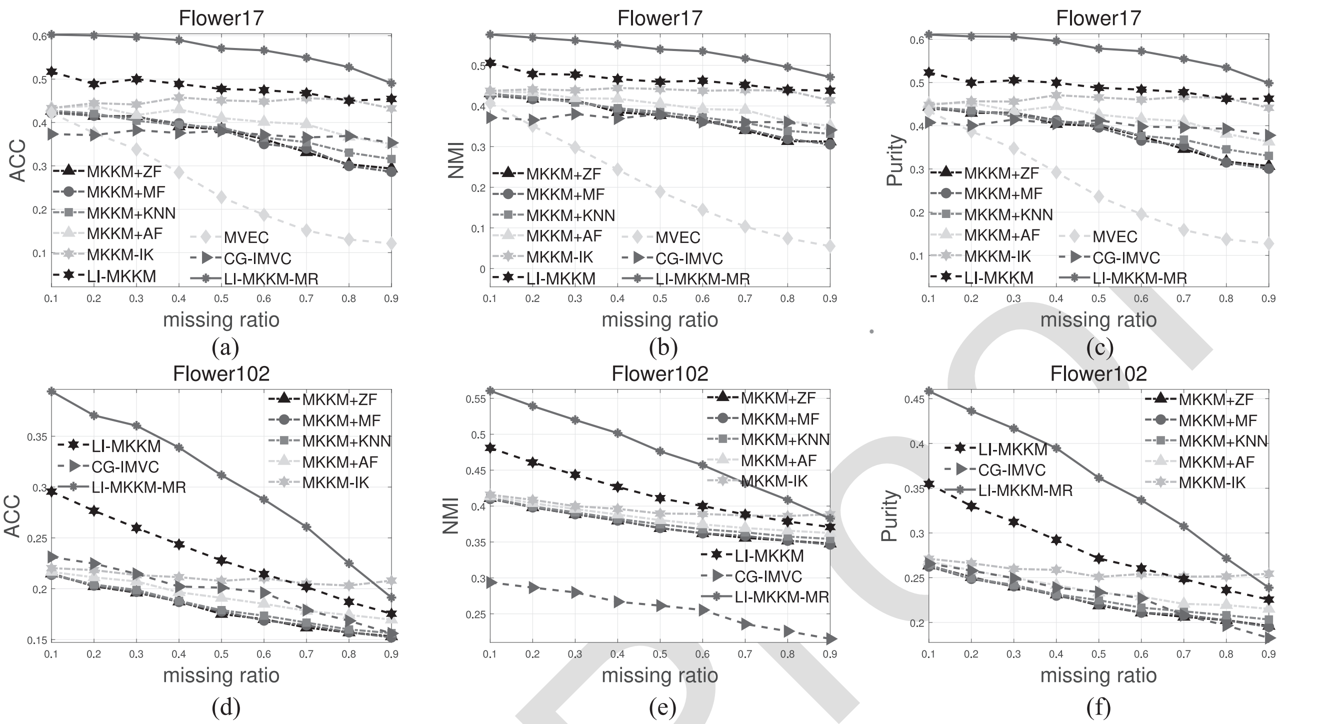


Fig. 1. Clustering ACC, NMI, and purity comparison with the variation of missing ratios on Flower17 and Flower102 datasets. (a) ACC with missing ratios on Flower17. (b) NMI with missing ratios on Flower17. (c) Purity with missing ratios on Flower17. (d) ACC with missing ratios on Flower102. (e) NMI with missing ratios on Flower102. (f) Purity with missing ratios on Flower102.

TABLE II
DATASETS SUMMARY

Dataset	#Samples	#Views	#Classes
Flower17	1360	7	17
Flower102	8189	4	102
Caltech102-5	510	48	102
Caltech102-10	1020	48	102
Caltech102-15	1530	48	102
Caltech102-20	2040	48	102
Caltech102-25	2550	48	102
Caltech102-30	3060	48	102
Digital	2000	3	10
ProteinFold	694	12	27
Reuters	18758	5	6

and MKKM + AF, respectively. Also, the newly proposed MKKM-IK [34], LI-MKKM [35], MVEC [43], and CG-IMVC [44] are also incorporated as strong baselines. The algorithms in [31], [32], and [45] are not incorporated into our experimental comparison since that these algorithms only consider the missing of input features, rather than the rows or columns of base kernel matrices in our case.

In the experiment, ε is used to denote the percentage of incomplete samples. Intuitively, the clustering performance will become less accurate when the value of ε is increasing. In our simulation, we set ε as $[0.1 : 0.1 : 0.9]$ on all the ten datasets. The performance metrics in this simulation include the clustering accuracy (ACC), normalized mutual information (NMI), and purity. For each method, we present the best result among 50 trials, where each trial started from a random initialization state. As a result, the effect of randomness caused by k-means could be alleviated. We generate "incomplete" patterns randomly for ten times and report the statistical results. For all datasets, the quantity of clusters is given and set as the ground truth of classes. The generation of the missing vectors $\{\mathbf{s}_p\}_{p=1}^m$ follows the approach in [34]: 1) randomly select round$(\varepsilon * n)$ samples with the rounding function round$(\cdot)$; 2) generate a random vector $\mathbf{v} = (v_1, \ldots, v_k, \ldots, v_m)$, $v_k \in [0, 1]$ and a scalar v_0, $v_0 \in [0, 1]$ for each selected sample; 3) if $v_p \geq v_0$, it presents the pth view for this sample; and 4) if there is no $v_p \geq v_0$, generate a new $\mathbf{v}$. Note that there is no requirement on complete view for each sample. In this instance, the index vector $\mathbf{s}_p$ is obtained to list the samples with the presentation on the pth view.

B. Experimental Results

Experiments on Flower17 and Flower102: Three performance metrics, including: 1) the ACC; 2) NMI; and 3) purity, of the testing algorithms with the variation of missing ratios in $[0.1, \ldots, 0.9]$ on the Flower17 and Flower102 datasets have been demonstrated in Fig. 1. We have the following observations.

1) The newly proposed MKKM-IK [36] (in green) has shown promising performance improvements on the ACC, NMI, and purity compared to the previous two-stage imputation methods. For example, the MKKM + AF outperforms MKKM-IK by 0.1%, 0.6%, 2.5%, 2.8%, 4.1%, 4.7%, 6.0%, 8.5%, and 8.2% in terms of clustering accuracy on Flower17, which clearly demonstrates the benefit of the joint optimization on imputation and clustering.

TABLE III
AGGREGATED ACC, NMI, AND PURITY COMPARISON (MEAN $\pm$ STD) OF DIFFERENT KINDS OF CLUSTERING ALGORITHMS ON FLOWER17 AND FLOWER102 DATASETS

Datasets	MKKM				MKKM-IK	LI-MKKM	MVEC	CG-IMVC	LI-MKKM-MR
	+ZF	+MF	+KNN	+AF [31]	[37]	[36]	[45]	[46]	Proposed
ACC									
Flower17	36.9 ± 0.8	36.8 ± 0.6	37.8 ± 0.6	40.5 ± 0.7	44.6 ± 0.6	48.0 ± 0.4	24.9 ± 0.4	37.1 ± 0.7	$\mathbf{56.6 \pm 0.3}$
Flower102	18.0 ± 0.2	18.0 ± 0.2	18.2 ± 0.1	19.2 ± 0.1	21.1 ± 0.2	23.1 ± 0.1	–	19.7 ± 0.3	$\mathbf{30.5 \pm 0.3}$
NMI									
Flower17	37.3 ± 0.4	37.3 ± 0.5	38.2 ± 0.5	40.1 ± 0.4	43.7 ± 0.3	46.4 ± 0.2	20.7 ± 0.4	36.5 ± 0.7	$\mathbf{53.5 \pm 0.2}$
Flower102	37.4 ± 0.1	37.4 ± 0.1	37.8 ± 0.1	38.4 ± 0.1	39.6 ± 0.1	41.8 ± 0.1	–	25.8 ± 0.3	$\mathbf{47.5 \pm 0.1}$
Purity									
Flower17	38.4 ± 0.6	38.3 ± 0.6	39.3 ± 0.6	42.0 ± 0.6	45.9 ± 0.5	48.9 ± 0.4	25.7 ± 0.4	40.1 ± 0.7	$\mathbf{57.3 \pm 0.2}$
Flower102	22.5 ± 0.1	22.4 ± 0.1	22.8 ± 0.1	23.7 ± 0.2	25.8 ± 0.2	28.1 ± 0.1	–	22.9 ± 0.3	$\mathbf{35.8 \pm 0.3}$

2) Also, LI-MKKM outperforms MKKM-IK by 8.4%, 4.4%, 5.8%, 3.1%, 2.6%, 2.6%, 1.2%, 0.2%, and 2.2% on Flower17. This result clearly verifies that the utilizing data's local structure further boosts the clustering performance.
3) Furthermore, our proposed LI-MKKM-MR (in red) significantly outperforms the LI-MKKM in all cases from Fig. 1(a)–(f) in the aspect of clustering performance. For example, LI-MKKM-MR further outperforms LI-MKKM by 8.5%, 11.2%, 9.7%, 10.1%, 9.4%, 9.2%, 8.2%, 7.7%, and 3.6%. This result indicates the effectiveness of incorporating the matrix-induced regularization.
4) In addition, our newly proposed method demonstrates stronger advantage when compared to previous ones, especially under low missing ratios. It is notable that in Fig. 1(a), when the missing ratio is extremely low ($\varepsilon = 0.1$), LI-MKKM-MR improves the second-best algorithm (LI-MKKM) by 8.5% in terms of clustering accuracy on Flower17.

In Table III, the aggregated ACC, NMI, purity, and the standard deviation are reported, where we show the highest performance one in bold. Similarly, the results also illustrate that MKKM + ZF, MKKM + MF, MKKM + KNN, MKKM + AF, and MKKM-IK are outperformed by the proposed algorithm. Specifically, the second-best one (LI-MKKM) is exceeded by the proposed LI-MKKM-MR by 7%.

Experiments on the Caltech102 Dataset: Fig. 2 presents ACC, NMI, and purity of all the testing algorithms over variational missing ratios on the Caltech102 datasets. We find out that the recently proposed MKKM-IK [36] (in green) achieves a comparable clustering performance with a representative two-stage imputation method MKKM + AF, while the proposed LI-MKKM outperforms MKKM-IK with significant improvements on all the performance criterions, details can be found in Fig. 2(a)–(i). More precisely, LI-MKKM obtains 6.4%, 5.0%, 5.1%, 4.7%, 4.6%, 4.5%, 3.8%, 3.2%, and 2.6% higher clustering accuracy than MKKM-IK when the missing ratios vary from 0.1 to 0.9 on Caltech102-30. This also illustrates that the well utilization of the local structure of data assures performance improvement. Furthermore, by taking into account the correlation among base kernels, LI-MKKM-MR further improves the clustering performance over the baseline LI-MKKM.

The aggregated ACC, NMI, and purity, and the standard deviation on Caltech 102 datasets are reported in Table IV. Similarly, in comparison to the MKKM + ZF, MKKM + MF, MKKM + KNN, MKKM + AF, and MKKM-IK, our method still achieves much better clustering performance. For instance, the proposed LI-MKKM-MR obtains 2.1%, 2.1%, 2.8%, 2.4%, 2.7%, and 2.4% higher clustering accuracy than LI-MKKM. In addition, LI-MKKM-MR achieves comparable clustering performance with the newly proposed CG-IMVC [44] in terms of ACC and purity on Caltech102. However, LI-MKKM-MR significantly outperforms CG-IMVC in terms of NMI. The results on Caltech102-5, Caltech102-10, and Caltech102-15 are provided in the supplementary material due to space limitation, whose results demonstrate the same conclusion as well.

Experiments on the UCI-Digital Dataset: In this simulation, we apply all the testing methods on the UCI-Digital dataset, which is widely utilized in MKC as a benchmark. For each kind of missing ratio, we generate "incomplete patterns" ten times and report their averaged results.

The ACC, NMI, and purity of all the testing methods over variational missing ratios are presented in Fig. 3. It is clear that the latest proposed MKKM-IK provides unsatisfactory results on UCI-Digital, which is even worse than MKKM+KNN. However, LI-MKKM significantly outperforms the second-best one (MKKM + KNN) by 22.2%, 21.9%, 20.6%, 19.5%, 17.9%, 17.9%, 20.4%, 23.8%, and 23.2% on accuracy. In addition, the proposed LI-MKKM-MR further consistently improves the clustering performance of LI-MKKM. The aggregated clustering results in Table V also denote the same performance.

Experiments on the Protein Fold Prediction Dataset: In this experiment, the protein fold dataset is applied to evaluate the testing methods, and we report all results in Fig. 4 and Table VI. Also, we can find that our LI-MKKM-MR also achieves much better results than the rest algorithms on ACC, NMI, and purity on the dataset.

Experiments on the Reuters Dataset: The clustering performance in terms of ACC, NMI, and purity with the variation of missing ratios on Reuters is plotted in Fig. 5. As seen, our proposed algorithm once again demonstrates significant superiority over the compared ones. We also report the aggregated ACC, NMI, and purity in Table VII, which also verify the effectiveness of the proposed LI-MKKM-MR. The

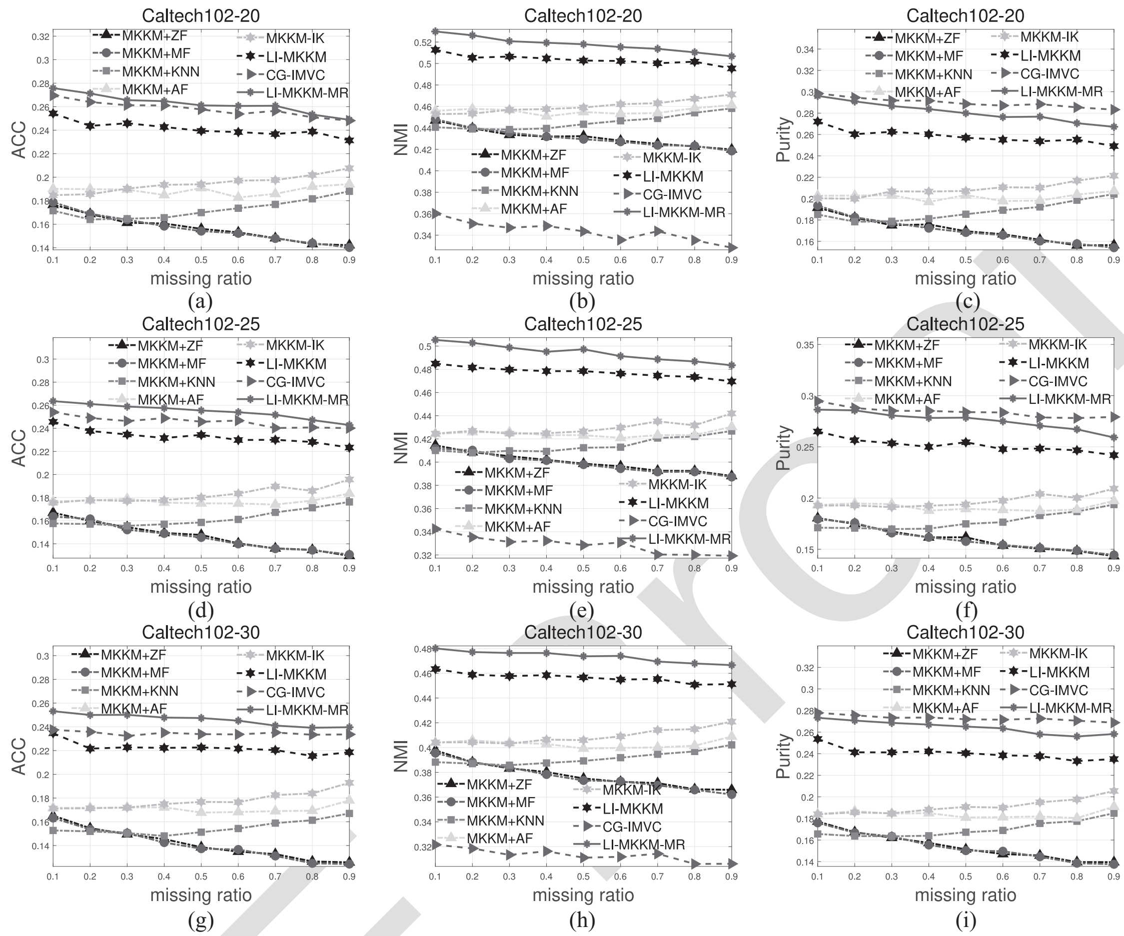


Fig. 2. Clustering ACC, NMI, and purity comparison with the variation of missing ratios on Caltech102-20, Caltech102-25, and Caltech102-30. (a) ACC with missing ratios on Caltech102-20. (b) NMI with missing ratios on Caltech102-20. (c) Purity with missing ratios on Caltech102-20. (d) ACC with missing ratios on Caltech102-25. (e) NMI with missing ratios on Caltech102-25. (f) Purity with missing ratios on Caltech102-25. g) ACC with missing ratios on Caltech102-30. (h) NMI with missing ratios on Caltech102-30. (i) Purity with missing ratios on Caltech102-30.

TABLE IV

Total ACC, NMI, and Purity Comparison (Mean $\pm$ Std) of Various Clustering Algorithms on Caltech102. On Account of Out of Memory, the Clustering Results of MVEC [43] on Caltech102-15, Caltech102-20, Caltech102-25, and Caltech102-30 Are Not Reported

	MKKM				MKKM-IK	LI-MKKM	MVEC	CG-IMVC	LI-MKKM-MR
	+ZF	+MF	+KNN	+AF [31]	[37]	[36]	[45]	[46]	Proposed
					ACC				
Cal102-5	26.1 ± 0.3	25.7 ± 0.3	27.3 ± 0.3	29.0 ± 0.3	28.9 ± 0.3	31.4 ± 0.3	26.8 ± 0.2	33.8 ± 0.2	$\mathbf{34.0 \pm 0.3}$
Cal102-10	19.7 ± 0.2	19.7 ± 0.2	21.5 ± 0.2	22.6 ± 0.2	22.7 ± 0.2	27.3 ± 0.2	22.4 ± 0.1	$\mathbf{28.9 \pm 0.2}$	$\mathbf{28.9 \pm 0.3}$
Cal102-15	17.1 ± 0.2	17.1 ± 0.2	18.9 ± 0.1	20.3 ± 0.2	20.8 ± 0.2	25.1 ± 0.2	–	$\mathbf{27.3 \pm 0.1}$	27.0 ± 0.4
Cal102-20	15.7 ± 0.1	15.7 ± 0.2	17.3 ± 0.2	18.9 ± 0.2	19.5 ± 0.1	24.1 ± 0.2	–	25.8 ± 0.2	$\mathbf{26.3 \pm 0.2}$
Cal102-25	14.7 ± 0.2	14.6 ± 0.1	16.2 ± 0.1	17.7 ± 0.2	18.3 ± 0.2	23.3 ± 0.2	–	24.6 ± 0.2	$\mathbf{25.5 \pm 0.2}$
Cal102-30	14.2 ± 0.1	14.1 ± 0.1	15.5 ± 0.2	17.1 ± 0.2	17.8 ± 0.2	22.2 ± 0.1	–	23.5 ± 0.1	$\mathbf{24.6 \pm 0.1}$
					NMI				
Cal102-5	64.3 ± 0.2	63.9 ± 0.1	65.9 ± 0.2	66.6 ± 0.1	66.5 ± 0.2	67.1 ± 0.2	65.6 ± 0.1	52.9 ± 0.4	$\mathbf{68.6 \pm 0.2}$
Cal102-10	53.6 ± 0.1	53.7 ± 0.1	55.2 ± 0.1	55.7 ± 0.2	55.8 ± 0.1	58.7 ± 0.1	55.1 ± 0.1	40.4 ± 0.5	$\mathbf{59.2 \pm 0.3}$
Cal102-15	47.4 ± 0.1	47.4 ± 0.1	48.8 ± 0.1	49.7 ± 0.1	50.1 ± 0.1	53.6 ± 0.1	–	37.0 ± 0.3	$\mathbf{54.6 \pm 0.2}$
Cal102-20	43.1 ± 0.1	43.1 ± 0.2	44.5 ± 0.1	45.6 ± 0.2	46.0 ± 0.1	50.4 ± 0.1	–	34.4 ± 0.3	$\mathbf{51.8 \pm 0.1}$
Cal102-25	40.0 ± 0.1	39.9 ± 0.1	41.5 ± 0.1	42.5 ± 0.2	43.0 ± 0.2	47.7 ± 0.2	–	32.9 ± 0.3	$\mathbf{49.4 \pm 0.1}$
Cal102-30	37.8 ± 0.1	37.7 ± 0.1	39.2 ± 0.1	40.3 ± 0.1	40.9 ± 0.1	45.6 ± 0.1	–	31.3 ± 0.2	$\mathbf{47.4 \pm 0.1}$
					Purity				
Cal102-5	26.7 ± 0.4	26.4 ± 0.3	27.9 ± 0.3	29.8 ± 0.3	29.6 ± 0.3	32.6 ± 0.3	27.3 ± 0.2	$\mathbf{35.9 \pm 0.2}$	35.5 ± 0.3
Cal102-10	21.0 ± 0.2	21.0 ± 0.2	22.9 ± 0.2	24.0 ± 0.3	24.2 ± 0.2	29.0 ± 0.2	23.3 ± 0.1	$\mathbf{31.7 \pm 0.2}$	30.8 ± 0.3
Cal102-15	18.5 ± 0.2	18.5 ± 0.2	20.4 ± 0.2	21.6 ± 0.2	22.2 ± 0.2	26.7 ± 0.2	–	$\mathbf{30.2 \pm 0.1}$	28.8 ± 0.3
Cal102-20	17.1 ± 0.1	17.0 ± 0.2	18.8 ± 0.2	20.2 ± 0.2	20.9 ± 0.1	25.8 ± 0.2	–	$\mathbf{29.0 \pm 0.2}$	28.1 ± 0.2
Cal102-25	16.0 ± 0.2	16.0 ± 0.2	17.7 ± 0.2	19.1 ± 0.2	19.7 ± 0.1	25.2 ± 0.2	–	$\mathbf{28.4 \pm 0.1}$	27.6 ± 0.2
Cal102-30	15.4 ± 0.1	15.4 ± 0.1	17.0 ± 0.1	18.4 ± 0.2	19.1 ± 0.2	24.0 ± 0.1	–	$\mathbf{27.3 \pm 0.1}$	26.5 ± 0.1

results of MVEC [43] and CG-IMVC [44] on Reuters are not reported due to out of memory.

In short, we summarize that our algorithm has three advantages.

1) *Joint Optimization Based on Imputation and Clustering*: First, the process of imputation is guided by the clustering results, which makes the imputation more direct to the final goal. Second, refining the clustering results

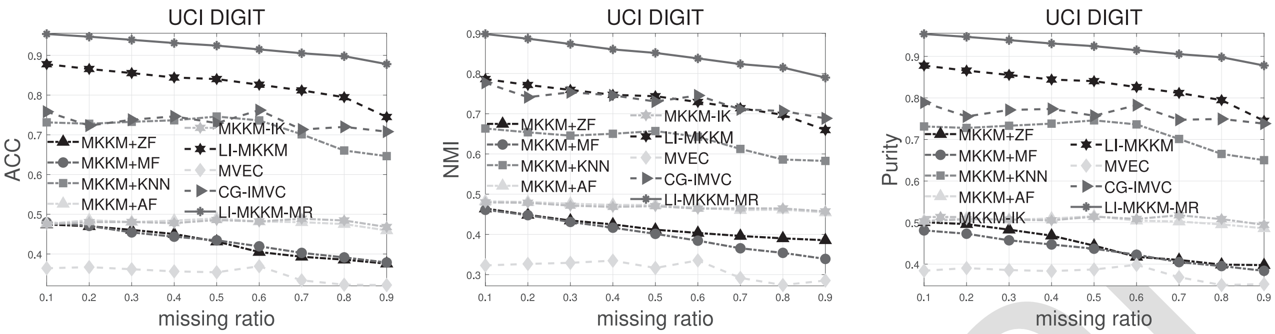


Fig. 3. Clustering ACC, NMI, and purity comparison with the variation of missing ratios on the UCI-digital dataset.

TABLE V
TOTAL ACC, NMI, AND PURITY COMPARISON (MEAN $\pm$ STD) OF VARIOUS CLUSTERING ALGORITHMS ON UCI-DIGITAL

MKKM				MKKM-IK	LI-MKKM	MVEC	CG-IMVC	LI-MKKM-MR
+ZF	+MF	+KNN	+AF [31]	[37]	[36]	[45]	[46]	Proposed
ACC								
42.7 ± 0.4	43.1 ± 0.3	71.3 ± 1.0	47.9 ± 0.5	48.0 ± 0.4	82.9 ± 0.3	35.0 ± 0.8	73.3 ± 1.1	$\mathbf{92.1 \pm 0.3}$
NMI								
41.8 ± 0.2	40.0 ± 0.2	63.3 ± 0.5	47.0 ± 0.2	46.9 ± 0.2	73.4 ± 0.3	31.3 ± 1.1	73.3 ± 0.9	$\mathbf{84.8 \pm 0.4}$
Purity								
44.6 ± 0.5	43.4 ± 0.3	71.4 ± 0.7	50.4 ± 0.3	50.8 ± 0.4	82.9 ± 0.3	37.8 ± 0.8	76.3 ± 1.0	$\mathbf{92.1 \pm 0.3}$

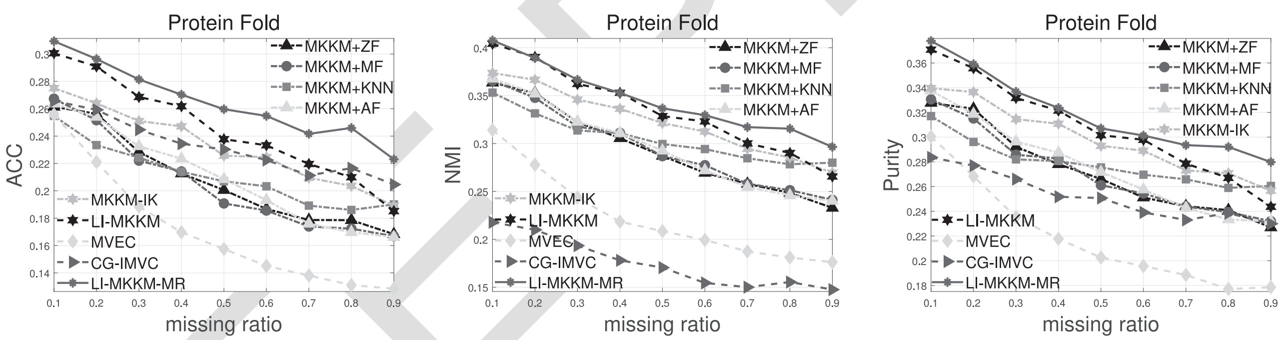


Fig. 4. Clustering ACC, NMI, and purity comparison with the variation of missing ratios on the protein Fold Prediction dataset.

TABLE VI
TOTAL ACC, NMI, AND PURITY COMPARISON (MEAN $\pm$ STD) OF VARIOUS CLUSTERING ALGORITHMS ON THE PROTEIN FOLD DATASET

MKKM				MKKM-IK	LI-MKKM	MVEC	CG-IMVC	LI-MKKM-MR
+ZF	+MF	+KNN	+AF [31]	[37]	[36]	[45]	[46]	Proposed
ACC								
20.8 ± 0.2	20.5 ± 0.3	21.1 ± 0.5	21.0 ± 0.2	23.2 ± 0.6	24.5 ± 0.5	17.1 ± 0.2	23.2 ± 0.3	$\mathbf{26.5 \pm 0.2}$
NMI								
29.3 ± 0.4	29.5 ± 0.5	30.5 ± 0.4	29.5 ± 0.3	32.3 ± 0.6	33.5 ± 0.3	22.3 ± 0.2	17.5 ± 0.6	$\mathbf{34.6 \pm 0.2}$
Purity								
27.2 ± 0.4	27.2 ± 0.4	27.9 ± 0.5	27.5 ± 0.4	29.8 ± 0.7	30.8 ± 0.4	21.8 ± 0.2	25.2 ± 0.5	$\mathbf{31.9 \pm 0.3}$

can benefits from this meaningful imputation. These two learning processes work well together, thus leading to the clustering performance improvement. In contrast, MKKM + MF, MKKM + KNNMKKM + ZF, and MKKM + AF algorithms do not fully make use of the connection between the imputation and clustering procedures. This may produce imputation, which does not well serve the subsequent clustering as originally expected, affecting the clustering performance.

2) *Considerably Utilizing Data's Local Structure*: Our local kernel alignment criterion is flexible and it makes the prespecified kernels aligned for better clustering performance.

3) *Well Considering the Correlation of Incomplete Base Kernels*: The incorporated matrix-induced regularization reduces the high redundancy and enforces low diversity among the selected kernels, making the prespecified kernels be well utilized.

These factors have led to significant improvements in cluster performance.

C. Reconstruction Error Comparison of LI-MKKM-MR

In this section, we evaluate the reconstruction errors of the LI-MKKM-MR with the aforementioned algorithms on all benchmark datasets. To do this, we calculate the reconstruction error between the ground-truth kernels and the imputed ones

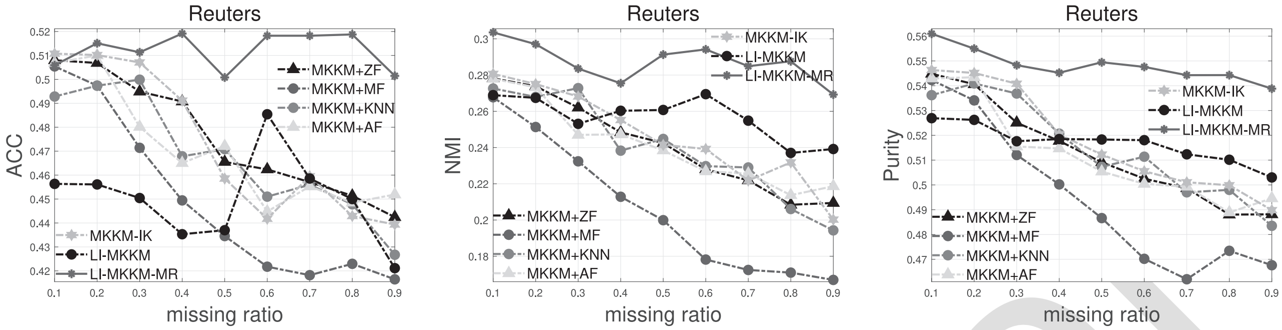


Fig. 5. Clustering ACC, NMI, and purity comparison with the variation of missing ratios on Reuters.

TABLE VII
AGGREGATED ACC, NMI, AND PURITY COMPARISON (MEAN ± STD) OF VARIOUS CLUSTERING ALGORITHMS ON REUTERS

MKKM				MKKM-IK	LI-MKKM	LI-MKKM-MR
+ZF	+MF	+KNN	+AF [31]	[37]	[36]	Proposed
ACC						
47.6 ± 0.0	44.9 ± 0.2	46.8 ± 0.2	47.1 ± 0.1	47.4 ± 0.4	45.0 ± 0.7	$\mathbf{51.2 \pm 1.0}$
NMI						
24.1 ± 0.1	20.6 ± 0.1	24.0 ± 0.3	24.1 ± 0.1	24.6 ± 0.3	25.7 ± 0.3	$\mathbf{28.7 \pm 0.1}$
Purity						
51.3 ± 0.0	49.4 ± 0.0	51.5 ± 0.2	51.2 ± 0.2	51.8 ± 0.1	51.7 ± 0.1	$\mathbf{54.8 \pm 0.0}$

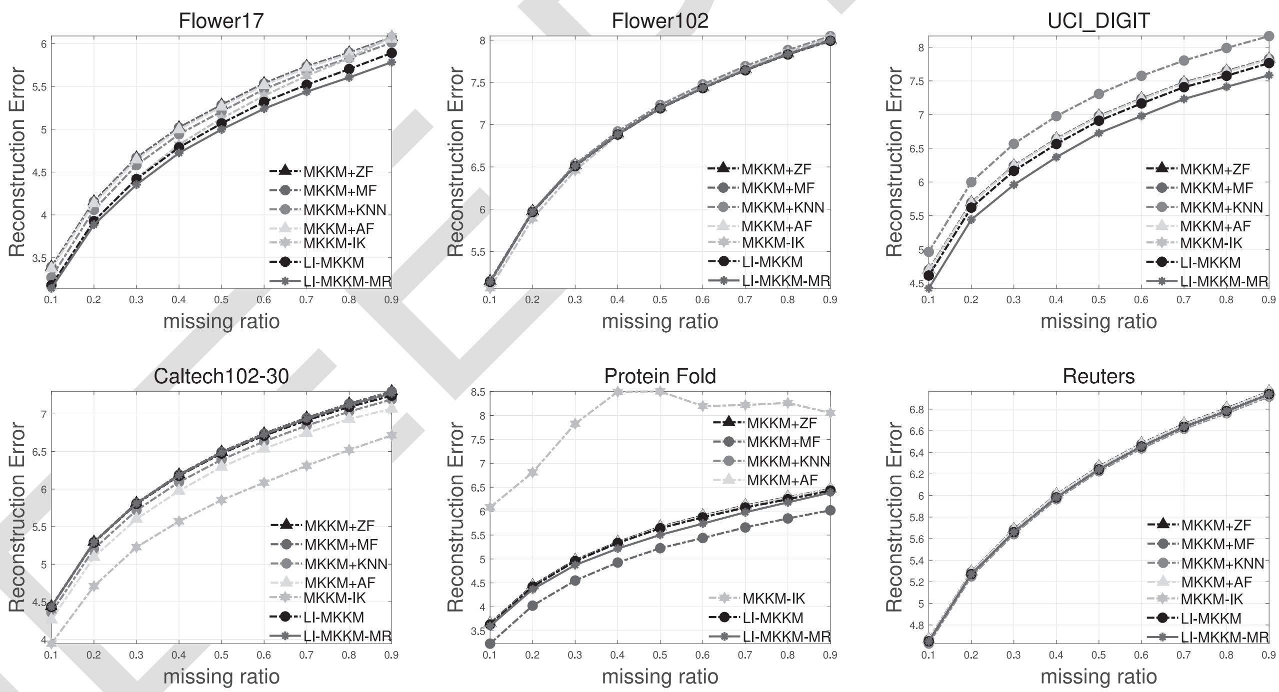


Fig. 6. Reconstruction error comparison of the compared algorithms with the variation of missing ratios on benchmark datasets.

via $\sum_{p=1}^{m} \|\mathbf{K}_p(\mathbf{s}_p, \mathbf{s}_p) - \hat{\mathbf{K}}_p(\mathbf{s}_p, \mathbf{s}_p)\|^2$, where $\mathbf{K}_p$ and $\hat{\mathbf{K}}_p$ denote the ground-truth and the imputed one, and $\mathbf{s}_p$ denotes the missing indices of the pth view. The results under various missing ratios are shown in Fig. 6. As observed, the kernels imputed by our algorithm align with the ground-truth kernels are comparable or slightly better when compared to those obtained by the existing imputation algorithms. Note that our ultimate goal in this work is clustering, while imputation is only a by-product. How to impute the missing views which not only achieves better clustering performance but also produces better imputation result is worth further exploring.

D. Parameter Sensitivity of LI-MKKM-MR

In this part, we analyze that relationship between the clustering performance and matrix-induced regularization. Referring to (10), LI-MKKM-MR induces the ratio of the nearest neighbors τ and regularization parameter λ. In the following, we conduct another experiment to show the variation of performance among different τ and λ on the Flower17 dataset.

Fig. 7(a) and (b) shows the ACC and NMI of our algorithm by varying τ in a huge range $[0.02 : 0.02 : 0.2]$ with $\lambda = 2^{-6}$. From these figures, we can find that: 1) ACC fluctuates with the monotonically increasing of τ and 2) the start points of the

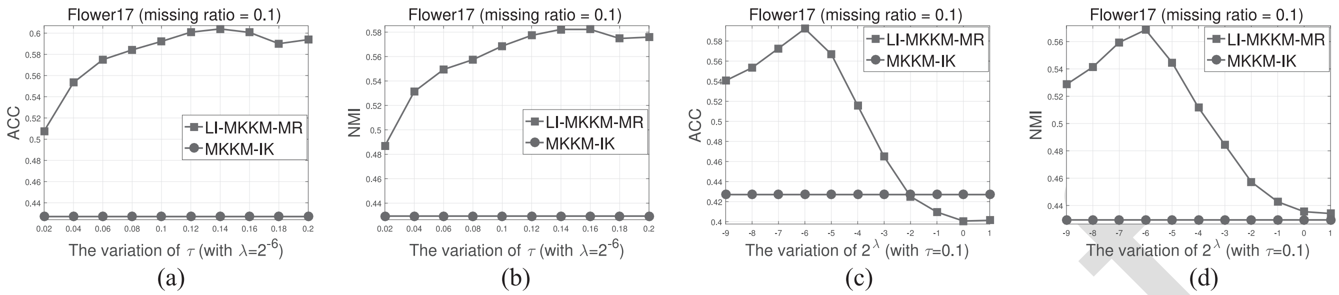


Fig. 7. Sensitivity of the proposed LI-MKKM-MR with the variation of λ and τ. (a) ACC with variation of τ on Flower17. (b) NMI with variation of τ on Flower17. (c) ACC with variation of λ on Flower17. (d) NMI with variation of λ on Flower17.

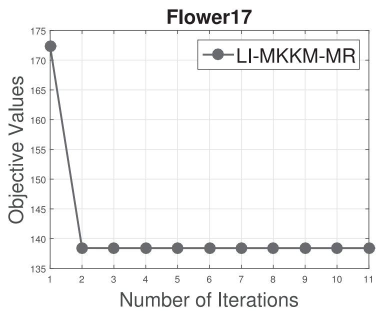


Fig. 8. Proposed algorithm convergence illustration.

ACC curves are typically higher than the end points, which induces that when the matrix-induced regularization term is dominated at ending points while the local kernel alignment maximization is dominated at starting points. These observations verify the successful joint preservation of the local structure of data and the matrix-induced regularization term in our algorithm. Similarly, Fig. 7(c) and 7(d) presents the ACC and NMI of our algorithms by tuning λ from 2^{-9} to 2 with $\tau = 0.1$. In this scenario, our algorithm also shows stable performance over variational λ.

As aforementioned, we conclude that compared to only preserving global kernel alignment, such as MKKM-IK in [36], our proposed algorithms are more essential to the clustering performance by preserving the local structure of data. Meanwhile, the clustering performance could be further improved by incorporating the correlation among base kernels. By appropriately integrating these two factors, it is possible to obtain the best clustering performance. Practically, there exists a tradeoff between the preservation of the local geometric structure and the correlation of base kernels to ensure the best clustering.

E. Convergence of LI-MKKM-MR

According to [46], the convergence of our proposed algorithm is guaranteed. We present one simulation trail of the proposed LI-MKKM-MR on the Flower 17 dataset as an example in 8. It is clearly shown that the objective value of the proposed algorithm is monotonically decreased and converges in a few iteration.

V. Conclusion

Though the newly proposed LI-MKKM is able to tackle the task of MKC with incomplete kernels, it takes the correlation among base kernels into account insufficiently. We proposed to calculate the kernel alignment to address this issue together with matrix-induced regularization in a local manner. The proposed algorithm efficiently solves the resultant optimization problem, and extensive experiments on benchmarks have demonstrated that LI-MKKM-MR consistently outperforms state-of-the-art baseline algorithms. In the future, we will design efficient and effective algorithms to solve the optimization problem directly without approximating $\mathbf{M}$ in (9).

References

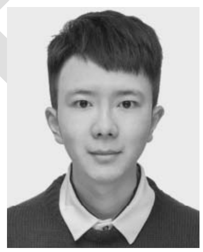
[1] K. Zhan, X. Chang, J. Guan, L. Chen, Z. Ma, and Y. Yang, "Adaptive structure discovery for multimedia analysis using multiple features," *IEEE Trans. Cybern.*, vol. 49, no. 5, pp. 1826–1834, May 2019.
[2] K. Zhan, F. Nie, J. Wang, and Y. Yang, "Multiview consensus graph clustering," *IEEE Trans. Image Process.*, vol. 28, pp. 1261–1270, 2019.
[3] D. Huang, J.-H. Lai, and C.-D. Wang, "Robust ensemble clustering using probability trajectories," *IEEE Trans. Knowl. Data Eng.*, vol. 28, no. 5, pp. 1312–1326, May 2016.
[4] C.-D. Wang, J.-H. Lai, and P. S. Yu, "Multi-view clustering based on belief propagation," *IEEE Trans. Knowl. Data Eng.*, vol. 28, no. 4, pp. 1007–1021, Apr. 2016.
[5] M. Yin, J. Gao, S. Xie, and Y. Guo, "Multiview subspace clustering via tensorial t-product representation," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 3, pp. 851–864, Mar. 2019.
[6] Z. Ren, S. X. Yang, Q. Sun, and T. Wang, "Consensus affinity graph learning for multiple kernel clustering," *IEEE Trans. Cybern.*, vol. 51, no. 6, pp. 3273–3284, Jun. 2021.
[7] K. Zhan, C. Niu, C. Chen, F. Nie, C. Zhang, and Y. Yang, "Graph structure fusion for multiview clustering," *IEEE Trans. Knowl. Data Eng.*, vol. 31, no. 10, pp. 1984–1993, Oct. 2019.
[8] W. Liang *et al.*, "Multi-view spectral clustering with high-order optimal neighborhood Laplacian matrix," *IEEE Trans. Knowl. Data Eng.*, early access, Sep. 18, 2020, doi: 10.1109/TKDE.2020.3025100.
[9] B. Zhao, J. T. Kwok, and C. Zhang, "Multiple kernel clustering," in *Proc. SDM*, 2009, pp. 638–649.
[10] Z. Ren and Q. Sun, "Simultaneous global and local graph structure preserving for multiple kernel clustering," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 5, pp. 1839–1851, May 2021.
[11] S. Li, Y. Jiang, and Z. Zhou, "Partial multi-view clustering," in *Proc. AAAI*, 2014, pp. 1968–1974.
[12] M. Gönen and A. A. Margolin, "Localized data fusion for kernel k-means clustering with application to cancer biology," in *Proc. NIPS*, 2014, pp. 1305–1313.
[13] L. Du *et al.*, "Robust multiple kernel k-means clustering using ℓ_{21}-norm," in *Proc. IJCAI*, 2015, pp. 3476–3482.
[14] X. Liu, Y. Dou, J. Yin, L. Wang, and E. Zhu, "Multiple kernel k-means clustering with matrix-induced regularization," in *Proc. AAAI*, 2016, pp. 1888–1894.
[15] D. Huang, C.-D. Wang, and J.-H. Lai, "Locally weighted ensemble clustering," *IEEE Trans. Cybern.*, vol. 48, no. 5, pp. 1460–1473, May 2018.
[16] M. Li, X. Liu, L. Wang, Y. Dou, J. Yin, and E. Zhu, "Multiple kernel clustering with local kernel alignment maximization," in *Proc. IJCAI*, 2016, pp. 1704–1710.

[17] S. Wang *et al.*, "Multi-view clustering via late fusion alignment maximization," in *Proc. IJCAI*, 2019, pp. 3778–3784.
[18] S. Yu *et al.*, "Optimized data fusion for kernel k-means clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 5, pp. 1031–1039, May 2012.
[19] X. Liu *et al.*, "Optimal neighborhood kernel clustering with multiple kernels," in *Proc. AAAI*, 2017, pp. 2266–2272.
[20] Q. Wang, Z. Qin, F. Nie, and X. Li, "Spectral embedded adaptive neighbors clustering," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 4, pp. 1265–1271, Apr. 2019.
[21] M.-S. Chen, L. Huang, C.-D. Wang, D. Huang, and P. S. Yu, "Multiview subspace clustering with grouping effect," *IEEE Trans. Cybern.*, early access, Dec. 7, 2020, doi: 10.1109/TCYB.2020.3035043.
[22] Z. Li, F. Nie, X. Chang, L. Nie, H. Zhang, and Y. Yang, "Rank-constrained spectral clustering with flexible embedding," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 12, pp. 6073–6082, Dec. 2018.
[23] Z. Li, F. Nie, X. Chang, Y. Yang, C. Zhang, and N. Sebe, "Dynamic affinity graph construction for spectral clustering using multiple features," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 12, pp. 6323–6332, Dec. 2018.
[24] M. C. Trinidad, R. Martin-Brualla, F. Kainz, and J. Kontkanen, "Multi-view image fusion," in *Proc. ICCV*, 2019, pp. 4100–4109.
[25] M. Hu and S. Chen, "One-pass incomplete multi-view clustering," in *Proc. AAAI*, 2019, pp. 3838–3845.
[26] C. Xu, Z. Guan, W. Zhao, H. Wu, Y. Niu, and B. Ling, "Adversarial incomplete multi-view clustering," in *Proc. IJCAI*, 2019, pp. 3933–3939.
[27] S. Xiang, L. Yuan, W. Fan, Y. Wang, P. M. Thompson, and J. Ye, "Multi-source learning with block-wise missing data for Alzheimer's disease prediction," in *Proc. ACM SIGKDD*, 2013, pp. 185–193.
[28] R. Kumar, T. Chen, M. Hardt, D. Beymer, K. Brannon, and T. F. Syeda-Mahmood, "Multiple kernel completion and its application to cardiac disease discrimination," in *Proc. ISBI*, 2013, pp. 764–767.
[29] Z. Ghahramani and M. I. Jordan, "Supervised learning from incomplete data via an EM approach," in *Proc. NIPS*, 1993, pp. 120–127.
[30] A. Trivedi, P. Rai, H. Daumé III, and S. L. DuVall, "Multiview clustering with incomplete views," in *Proc. NIPS Mach. Learn. Soc. Comput. Workshop* 2010, pp. 1–7.
[31] C. Xu, D. Tao, and C. Xu, "Multi-view learning with incomplete views," *IEEE Trans. Image Process.*, vol. 24, pp. 5812–5825, 2015.
[32] W. Shao, L. He, and P. S. Yu, "Multiple incomplete views clustering via weighted nonnegative matrix factorization with $\ell_{2,1}$ regularization," in *Machine Learning and Knowledge Discovery in Databases (ECML PKDD)*. Cham, Switzerland: Springer, 2015, pp. 318–334.
[33] S. Bhadra, S. Kaski, and J. Rousu, "Multi-view kernel completion," 2016, *arXiv:1602.02518*.
[34] X. Liu, M. Li, L. Wang, Y. Dou, J. Yin, and E. Zhu, "Multiple kernel k-means with incomplete kernels," in *Proc. AAAI*, 2017, pp. 2259–2265.
[35] X. Zhu *et al.*, "Localized incomplete multiple kernel k-means," in *Proc. IJCAI*, 2018, pp. 3271–3277.
[36] X. Liu *et al.*, "Multiple kernel k-means with incomplete kernels," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 5, pp. 1191–1204, May 2020.
[37] S. Jegelka, A. Gretton, B. Schölkopf, B. K. Sriperumbudur, and U. von Luxburg, "Generalized clustering via kernel embeddings," in *Proc. 32nd Annu. German Conf. AI Adv. Artif. Intell.*, 2009, pp. 144–152.
[38] L. Yuan, Y. Wang, P. M. Thompson, V. A. Narayan, and J. Ye, "Multi-source feature learning for joint analysis of incomplete multiple heterogeneous neuroimaging data," *NeuroImage*, vol. 61, no. 3, pp. 622–632, 2012.
[39] Y. Liu *et al.*, "Incomplete multi-modal representation learning for Alzheimer's disease diagnosis," *Med. Image Anal.*, vol. 69, Apr. 2021, Art. no. 101953.
[40] C. Cortes, M. Mohri, and A. Rostamizadeh, "Algorithms for learning kernels based on centered alignment," *J. Mach. Learn. Res.*, vol. 13, pp. 795–828, Jan. 2012.
[41] A. Maurer and M. Pontil, "*k*-dimensional coding schemes in Hilbert spaces," *IEEE Trans. Inf. Theory*, vol. 56, no. 11, pp. 5839–5846, Nov. 2010.
[42] T. Liu, D. Tao, and D. Xu, "Dimensionality-dependent generalization bounds for *k*-dimensional coding schemes," *Neural Comput.*, vol. 28, no. 10, pp. 2213–2249, 2016.
[43] Z. Tao, H. Liu, S. Li, Z. Ding, and Y. Fu, "From ensemble clustering to multi-view clustering," in *Proc. IJCAI*, 2017, pp. 2843–2849.
[44] W. Zhou, H. Wang, and Y. Yang, "Consensus graph learning for incomplete multi-view clustering," in *Advances in Knowledge Discovery and Data Mining*. Cham, Switzerland: Springer, 2019, pp. 529–540.
[45] H. Zhao, H. Liu, and Y. Fu, "Incomplete multimodal visual data grouping," in *Proc. IJCAI*, 2016, pp. 2392–2398.
[46] J. C. Bezdek and R. J. Hathaway, "Convergence of alternating optimization," *Neural Parallel Sci. Comput.*, vol. 11, no. 4, pp. 351–368, 2003.

Miaomiao Li is currently pursuing the Ph.D. degree with the National University of Defense Technology, Changsha, China.

She is currently a Lecturer with Changsha College, Changsha. She has published several peer-reviewed papers, such as IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, IEEE TRANSACTIONS ON MULTIMEDIA, AAAI, IJCAI, and *Neurocomputing*. Her current research interests include kernel learning and multiview clustering.

Ms. Li serves on the Technical Program Committees of IJCAI/AAAI 2017–2020.



Jingyuan Xia received the B.Sc. and M.Sc. degrees from the National University of Defense Technology, Hunan, China, in 2020. He is currently pursuing the Ph.D. degree with the Department of Electrical and Electronic Engineering, Imperial College London, London, U.K.

His current research interests include bilinear convex optimization, low-rank matrix completion, sparse signal processing, and intelligent transportation estimation.

Huiying Xu received the M.S. degree from the National University of Defense Technology, Changsha, China, in 2006.

She is an Associate Professor with the College of Mathematics and Computer Science, Zhejiang Normal University, Jinhua, China, and also the Researcher with the Research Institute of Ningbo Cixing Company Ltd., Ningbo, China. Her research interests include kernel learning and feature selection, machine learning, deep learning, computer vision, image processing, pattern recognition, computer simulation, digital watermarking, and their applications.

Ms. Xu is a member of the China Computer Federation.

Qing Liao (Member, IEEE) received the Ph.D. degree in computer science and engineering from the Department of Computer Science and Engineering, Hong Kong University of Science and Technology, Hong Kong, in 2016, supervised by Prof. Q. Zhang.

She is currently an Assistant Professor with the School of Computer Science and Technology, Harbin Institute of Technology (Shenzhen), Shenzhen, China. Her research interests include artificial intelligence and data mining.

Xinzhong Zhu (Member, IEEE) received the M.S. degree from the National University of Defense Technology (NUDT), Changsha, China, and the Ph.D. degree from Xidian University, Xi'an, China, in 2018.

He is a Professor with the College of Mathematics and Computer Science, Zhejiang Normal University, Jinhua, China, and also the President of Research Institute of Ningbo Cixing Company Ltd., Ningbo, China. He has published more than 30 peer-reviewed papers, including those in highly regarded journals and conferences, such as the IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, the IEEE TRANSACTIONS ON MULTIMEDIA, the IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, AAAI, and IJCAI. His research interests include machine learning, deep learning, computer vision, manufacturing informatization, robotics and system integration, and intelligent manufacturing.

Prof. Zhu served on the Technical Program Committees of IJCAI 2020 and AAAI 2020. He is a member of ACM and certified as a CCF Senior Member.

Xinwang Liu (Senior Member, IEEE) received the Ph.D. degree from the National University of Defense Technology (NUDT), Changsha, China, in 2013.

He is currently a Professor with the School of Computer, NUDT. His current research interests include kernel learning and unsupervised feature learning. He has published over 70 peer-reviewed papers, including those in highly regarded journals and conferences, such as IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, IEEE TRANSACTIONS ON IMAGE PROCESSING, IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, IEEE TRANSACTIONS ON MULTIMEDIA, IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, ICML, NeurIPS, CVPR, ICCV, AAAI, and IJCAI. More information can be found at https://xinwangliu.github.io/.