# Local Sample-Weighted Multiple Kernel Clustering With Consensus Discriminative Graph

Liang Li, Siwei Wang, Xinwang Liu, *Senior Member, IEEE*, En Zhu, Li Shen, Kenli Li, *Senior Member, IEEE*, and Keqin Li, *Fellow, IEEE*

***Abstract*— Multiple kernel clustering (MKC) is committed to achieving optimal information fusion from a set of base kernels. Constructing precise and local kernel matrices is proven to be of vital significance in applications since the unreliable distant–distance similarity estimation would degrade clustering performance. Although existing localized MKC algorithms exhibit improved performance compared with globally designed competitors, most of them widely adopt the KNN mechanism to localize kernel matrix by accounting for $\tau$-nearest neighbors. However, such a coarse manner follows an unreasonable strategy that the ranking importance of different neighbors is equal, which is impractical in applications. To alleviate such problems, this article proposes a novel local sample-weighted MKC (LSWMKC) model. We first construct a consensus discriminative affinity graph in kernel space, revealing the latent local structures. Furthermore, an optimal neighborhood kernel for the learned affinity graph is output with naturally sparse property and clear block diagonal structure. Moreover, LSWMKC implicitly optimizes adaptive weights on different neighbors with corresponding samples. Experimental results demonstrate that our LSWMKC possesses better local manifold representation and outperforms existing kernel or graph-based clustering algorithms. The source code of LSWMKC can be publicly accessed from https://github.com/liliangnudt/LSWMKC.**

***Index Terms*— Graph learning, localized kernel, multiview clustering, multiple kernel learning.**

## I. Introduction

CLUSTERING is one of the representative unsupervised learning techniques widely employed in data mining and machine learning [1]–[6]. As a popular algorithm, $k$-means has been well investigated [7]–[9]. Although achieving extensive applications, $k$-means assumes that data can be linearly separated into different clusters [10]. By employing kernel tricks, the nonlinearly separable data are embedded into a higher dimensional feature space and become linearly separable. As a consequence, kernel $k$-means (KKM) is naturally developed for handling nonlinearity issues [10], [11]. Moreover, to encode the emerging data generated from heterogeneous sources or views, multiple kernel clustering (MKC) provides a flexible and expansive framework for combining a set of kernel matrices since different kernels naturally correspond to different views [12]–[18]. Multiple KKM (MKKM) [19] and various variants are further developed and widely employed in many applications [15], [16], [20]–[23].

Most of the kernel-based algorithms follow a common assumption that all the samples are reliable to exploit the intrinsic structures of data, and thus, such a globally designed manner equally calculates the pairwise similarities of all samples [15]–[17], [20], [21], [24], [25]. Nevertheless, in a high-dimensional space, this assumption is incompatible with the well-acknowledged theory that the similarity estimation for distant samples is less reliable on account of the intrinsic manifold structures are highly complex with curved, folded, or twisted characteristics [26]–[29]. Furthermore, researchers have found that preserving reliable local manifold structures of data could achieve better effectiveness than globally preserving all the pairwise similarities in unsupervised tasks and can achieve better clustering performance, such as dimension reduction [30]–[33] and clustering [34], [35].

Therefore, many approaches are proposed to localize kernels to enhance discrimination [36]–[40]. The work in [36] develops a localized kernel maximizing alignment method that merely aligns the original kernel with $\tau$-nearest neighbors of each sample to the learned optimal kernel. Along this way, the KNN mechanism is introduced to kernel-based subspace segmentation [38]. Moreover, a recently proposed simple MKKM method [24] with min–max optimization is also localized in the same way to consider local structures [40]. Besides, such a localized manner also has been extended to handle incomplete data [37]. Although showing improved performance, most traditional localized kernel methods adopt the simple KNN mechanism to select neighbors.

As can be seen in Fig. 1(a) and (b), previous localized MKC methods with the KNN mechanism encounter two issues: 1) these methods follow the common assumption that all the neighbors are reliable without considering their variation and

Manuscript received 15 December 2021; revised 7 April 2022; accepted 12 June 2022. Date of publication 15 July 2022; date of current version 6 February 2024. This work was supported in part by the National Key Research and Development Program of China under Grant 2020AAA0107100 and in part by the National Natural Science Foundation of China under Project 61922088, Project 61773392, and Project 61976196. *(Liang Li and Siwei Wang contributed equally to this work.) (Corresponding author: Xinwang Liu.)*

Liang Li, Siwei Wang, Xinwang Liu, En Zhu, and Li Shen are with the School of Computer, National University of Defense Technology, Changsha 410073, China (e-mail: liangli@nudt.edu.cn; wangsiwei13@nudt.edu.cn; xinwangliu@nudt.edu.cn; enzhu@nudt.edu.cn; lishen@nudt.edu.cn).

Kenli Li is with the College of Computer Science and Electronic Engineering, Hunan University, Changsha 410073, China, and also with the Supercomputing and Cloud Computing Institute, Hunan University, Changsha 410073, China (e-mail: lkl@hnu.edu.cn).

Keqin Li is with the Department of Computer Science, State University of New York, New Paltz, NY 12561 USA (e-mail: lik@newpaltz.edu).

This article has supplementary downloadable material available at https://doi.org/10.1109/TNNLS.2022.3184970.

Digital Object Identifier 10.1109/TNNLS.2022.3184970

2162-237X © 2022 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: National Univ of Defense Tech. Downloaded on February 07,2024 at 15:34:14 UTC from IEEE Xplore. Restrictions apply.

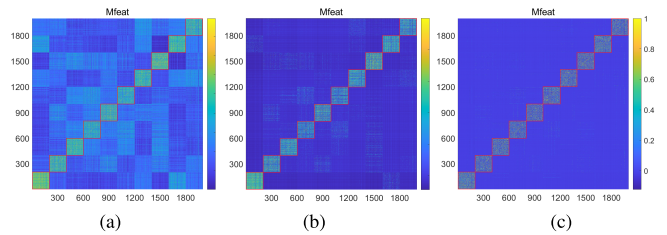



Fig. 1. Illustration of (a) original average kernel, (b) localized average kernel in KNN mechanism by carefully tuning $\tau$ within $[0.1, 0.2, \ldots, 0.9]$ and present the optimal results ($\tau = 0.1$), and (c) localized kernel learned by proposed model on Mfeat dataset.

ranking relationship. However, it is incompatible with common knowledge that the neighbors of a sample are adaptively varied, and some may have been corrupted by noise or outliers. For instance, in social networking, a closer relationship means more essential and vice versa. 2) The KNN mechanism introduces a hyperparameter neighbor ratio, which is fixed for each sample and commonly predetermined empirically. Apart from this unreasonable fixed neighbor ratio, it incurs dataset-related parameter-tuning in a wide range to obtain satisfying clustering results. From experimental results, we can observe that the KNN mechanism still preserves apparent noise compared with the original average kernel.

To alleviate these problems, we start our work with a natural thought that adaptively assigns a reasonable weight to each neighbor according to its ranking importance. However, there is no sufficient prior knowledge in kernel space to identify the ranking relationship among neighbors. Owing to the remarkable performance in exploring the complex nonlinear structures of various data, developing graph-based methods is greatly popular with scholars [27], [41]–[56]. Considering kernel matrix can be regarded as affinity graph with additional positive semidefinite (PSD) constraint, it is practicable and more flexible to learn a discriminative affinity graph with naturally sparsity and clear block diagonal structures [41], [43], [47], [57].

Based on the above-mentioned motivation and our inspiration from graph learning [41], [47], [48], [51], [57], [58], we develop a novel local sample-weighted MKC with consensus discriminative graph method (LSWMKC). Instead of using the KNN mechanism to localize the kernel matrix without considering the ranking importance of neighbors, we first learn a consensus discriminative affinity graph across multiple views in kernel space to reveal the latent manifold structures, and further heuristically learn an optimal neighborhood kernel. As Fig. 1(c) shows, the learned neighborhood kernel is naturally sparse with clear block diagonal structures. We develop an efficient iterative algorithm to simultaneously learn weights of base kernels, discriminative affinity graph, and localized consensus neighborhood kernel. Instead of empirically tuning or selecting a predefined neighbor ratio, our model can implicitly optimize adaptive weights on different neighbors with corresponding samples. Extensive experiments demonstrate that the learned neighborhood kernel can achieve clear local manifold structures, and it outperforms localized MKC methods in the KNN mechanism and other existing models. We briefly summarize the main contributions as follows:

1) A novel local sample-weighted MKC algorithm is proposed based on kernelized graph learning, which can implicitly optimize adaptive weights on different neighbors with corresponding samples according to their ranking importance.
2) We learn an optimal neighborhood kernel with more discriminative capacity by further denoising the graph, revealing the latent local manifold representation in kernel space.
3) We conduct extensive experimental evaluations on 12 MKC benchmark datasets compared with the existing 13 methods. Our proposed LSWMKC shows apparent effectiveness over localized MKC methods in the KNN mechanism and other existing methods.

## II. Background

This section introduces MKC and traditional KNN-based localized MKC methods.

### A. Multiple Kernel k-Means

For a data matrix $\mathbf{X} \in \mathbb{R}^{d\times n}$, including $n$ samples with $d$-dimensional features from $k$ clusters, nonlinear feature mapping $\psi(\cdot) : \mathbb{R}^d \mapsto \mathcal{H}$ achieves the transformation from sample space $\mathbb{R}^d$ to a reproducing kernel Hilbert space (RKHS) $\mathcal{H}$ [59]. Kernel matrix $\mathbf{K}$ is computed by

$$\mathbf{K}_{ij} = \kappa\left(\mathbf{x}_i, \mathbf{x}_j\right) = \psi(\mathbf{x}_i)^\top \psi\left(\mathbf{x}_j\right) \tag{1}$$

where $\kappa(\cdot,\cdot) : \mathbb{R}^d \times \mathbb{R}^d \mapsto \mathbb{R}$ denotes a PSD kernel function.

$k$-means is to minimize the clustering loss, that is,

$$\min_{\mathbf{S}} \sum_{i=1}^{n}\sum_{q=1}^{k} \mathbf{S}_{iq}\|\mathbf{x}_i - \mathbf{c}_q\|_2^2, \quad \text{s.t.} \sum_{q=1}^{k} \mathbf{S}_{iq} = 1 \tag{2}$$

where $\mathbf{S} \in \{0,1\}^{n\times k}$ denotes the indicator matrix, $\mathbf{c}_q$ denotes the centroid of $q$-th cluster and $n_q = \sum_{i=1}^{n} \mathbf{S}_{iq}$ denotes the corresponding amount of samples. To deal with nonlinear features, the samples are mapped into RKHS $\mathcal{H}$. KKM is formulated as

$$\min_{\mathbf{H}} \operatorname{Tr}\left(\mathbf{K}\left(\mathbf{I}_n - \mathbf{H}\mathbf{H}^\top\right)\right), \quad \text{s.t. } \mathbf{H}^\top\mathbf{H} = \mathbf{I}_k \tag{3}$$

where partition matrix $\mathbf{H} \in \mathbb{R}^{n\times k}$ is computed by taking rank-$k$ eigenvectors of $\mathbf{K}$ and then exported to $k$-means to compute the final results [10], [11].

For multiple kernel learning scenarios, $\mathbf{x}$ can be represented as $\psi_{\boldsymbol{\omega}}(\mathbf{x}) = [\omega_1\psi_1(\mathbf{x})^\top, \omega_2\psi_2(\mathbf{x})^\top, \ldots, \omega_m\psi_m(\mathbf{x})^\top]^\top$, where $\boldsymbol{\omega} = [\omega_1, \ldots, \omega_m]^\top$ denotes the coefficients of $m$ base kernel functions $\{\kappa_p(\cdot,\cdot)\}_{p=1}^m$. $\kappa_{\boldsymbol{\omega}}(\cdot,\cdot)$ is expressed as

$$\kappa_{\boldsymbol{\omega}}\left(\mathbf{x}_i, \mathbf{x}_j\right) = \psi_{\boldsymbol{\omega}}(\mathbf{x}_i)^\top \psi_{\boldsymbol{\omega}}\left(\mathbf{x}_j\right) = \sum_{p=1}^{m} \omega_p^2 \kappa_p\left(\mathbf{x}_i, \mathbf{x}_j\right). \tag{4}$$

The objective of MKKM is formulated as

$$\begin{aligned} &\min_{\mathbf{H},\boldsymbol{\omega}} \operatorname{Tr}\left(\mathbf{K}_{\boldsymbol{\omega}}\left(\mathbf{I}_n - \mathbf{H}\mathbf{H}^\top\right)\right) \\ &\text{s.t. } \mathbf{H} \in \mathbb{R}^{n\times k}, \quad \mathbf{H}^\top\mathbf{H} = \mathbf{I}_k, \ \omega_p \geq 0 \quad \forall p \end{aligned} \tag{5}$$

where the consensus kernel $\mathbf{K}_{\boldsymbol{\omega}} = \sum_{p=1}^{m} \omega_p^2 \mathbf{K}_p$ is commonly assumed as a combination of base kernels $\mathbf{K}_p$. To control the

Authorized licensed use limited to: National Univ of Defense Tech. Downloaded on February 07,2024 at 15:34:14 UTC from IEEE Xplore. Restrictions apply.

contribution of different kernels, there are some strategies on $\boldsymbol{\omega}$, such as "kernel affine weight strategy" [51], "autoweighted strategy" [43], [48], and "sum-to-one strategy" [40]. According to [19], (5) can be solved by alternatively optimizing $\boldsymbol{\omega}$ and $\mathbf{H}$.

### *B. Construction of Localized Kernel in KNN Mechanism*

Most kernel-based methods assume that all the samples are reliable and calculate fully connected pairwise similarity. However, as pointed out in [26]–[29] and [60], the similarity estimation of distant–distance samples in high-dimensional space is unreliable. Many localized kernel-based works have been developed to alleviate this problem [36], [40], [61]. Commonly, the localized kernel is constructed in the KNN mechanism.

The construction of a localized kernel mainly includes two steps, i.e., neighbor searching and localized kernel construction. First, in average kernel space, the neighbors of each sample are identified by labeling its $\tau$-nearest samples. Denoting the neighbor mask matrix as $\mathbf{N} \in \{0, 1\}^{n\times n}$. The neighbor searching is defined as follows:

$$\mathbf{N}_{ij} = \begin{cases} 1, & \mathbf{x}_j \in \mathrm{KNN}(\mathbf{x}_i), \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

where $j$ denotes the neighbor index of $i$-th sample. For each row, there are $round(\tau n)$ elements are labeled as neighbors, where neighbor ratio $\tau$ is commonly predetermined empirically and carefully tuned by grid search, such as $\tau$ varies within $[0.1, 0.2, \ldots, 0.9]$, and finally, obtain the optimal clustering results. If we set neighbor ratio $\tau = 1$, the KNN structure will be full-connected. For the precomputed base kernels $\mathbf{K}_p$, the corresponding localized kernel $\mathbf{K}_{p(l)}$ is formulated as

$$\mathbf{K}_{p(l)} = \mathbf{N} \odot \mathbf{K}_p \tag{7}$$

where $\odot$ is the Hadamard product.

Although the traditional KNN mechanism to localize kernel is simple and has improved performance than globally designed methods, this manner neglects a critical issue the variation of neighbors. Therefore, it is important and practical to assign reasonable weights to different neighbors according to their ranking relationship. Another issue is that the initial neighbor ratio $\tau$ of each sample is usually fixed and predetermined empirically and needs to be tuned to report the best clustering result. As Fig. 1(a) and (b) shows, the obtained localized kernels preserve much noise, which will incur degeneration of clustering performance.

## III. Methodology

This section presents our proposed LSWMKC in detail and provides an efficient three-step optimization solution. Moreover, we analyze convergence, computational complexity, limitation, and extensions.

### *A. Motivation*

From our aforementioned analysis of the traditional localized kernel method in the KNN mechanism, we find that: 1) This seemingly simple method neglects the ranking importance of the neighbors, which may degrade the clustering performance due to the impact of the unreliable distant–distance relationship. 2) The neighbor ratio is commonly predetermined empirically and needs to be tuned to report the best results.

The above-mentioned issues inspire us to rethink the manner of constructing localized MKC, and a natural motivation is to exploit their ranking relationship and assign a reasonable weight to each neighbor. However, there is no sufficient prior knowledge in kernel space to identify the ranking importance of neighbors. In recent years, graph-based algorithms have been greatly popular with scholars to explore the nonlinear structures of data. An ideal affinity graph exhibits two good properties: 1) clear block diagonal structures with $k$ connected blocks, each corresponding to one cluster. 2) The affinity represents the similarity of pairwise samples, and the intracluster affinities are nonzero, while the extra-cluster affinities are zeros. Considering the kernel matrix can be regarded as the affinity graph with additional PSD constraint, a discriminative graph can reveal the latent local manifold representation in kernel space. These issues inspire us to exploit the capacity of graph learning in capturing nonlinear structures of kernel space.

### *B. Proposed Formula*

Here, we briefly introduce the affinity graph learning method, which will be the base of our proposed model.

For sample set $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$, it is desirable to learn an affinity graph $\mathbf{Z} \in \mathbb{R}^{n\times n}$ with distinct distance $\|\mathbf{x}_i - \mathbf{x}_j\|_2^2$ corresponding to small similarity $z_{ij}$, which is formulated as

$$\begin{aligned} &\min_{\mathbf{Z}} \sum_{i,j=1}^{n} \|\mathbf{x}_i - \mathbf{x}_j\|_2^2 z_{ij} + \gamma z_{ij}^2 \\ &\text{s.t. } \mathbf{Z}_{i,:}\mathbf{1}_n = 1,\ z_{ij} \geq 0,\ z_{ii} = 0 \end{aligned} \tag{8}$$

where $\gamma$ is a hyperparameter, $\mathbf{Z}_{i,:}\mathbf{1}_n = 1$ is for normalization, $z_{ij} \geq 0$ is to ensure the nonnegative property, and $z_{ii} = 0$ can avoid trivial solutions. Commonly, the second term $\ell_2$ norm regularization is to avoid undesired trivial solutions [42], [62].

However, the existing graph-based methods are developed in sample space $\mathbb{R}^d$, rather than RKHS $\mathcal{H}$ kernel space, significantly limiting their applications. To fill this gap and exploit their potent capacity to capture nonlinear structures in kernel space, by using kernel tricks, the first term of (8) can be extended as

$$\begin{aligned} &\min_{\mathbf{Z}} \sum_{i,j=1}^{n} \|\psi(\mathbf{x}_i) - \psi(\mathbf{x}_j)\|_2^2 z_{ij} \\ &= \min_{\mathbf{Z}} \sum_{i,j=1}^{n} (\psi(\mathbf{x}_i)^\top \psi(\mathbf{x}_i) - 2\psi(\mathbf{x}_i)^\top \psi(\mathbf{x}_j) + \psi(\mathbf{x}_j)^\top \psi(\mathbf{x}_j)) z_{ij} \\ &= \min_{\mathbf{Z}} \sum_{i,j=1}^{n} (\kappa(\mathbf{x}_i, \mathbf{x}_i) - 2\kappa(\mathbf{x}_i, \mathbf{x}_j) + \kappa(\mathbf{x}_j, \mathbf{x}_j)) z_{ij} \\ &= \min_{\mathbf{Z}} 2n - \sum_{i,j=1}^{n} 2\kappa(\mathbf{x}_i, \mathbf{x}_j) z_{ij} \Leftrightarrow \min_{\mathbf{Z}} \sum_{i,j=1}^{n} -\kappa(\mathbf{x}_i, \mathbf{x}_j) z_{ij} \\ &\text{s.t. } \mathbf{Z}_{i,:}\mathbf{1}_n = 1, \quad z_{ij} \geq 0,\ z_{ii} = 0. \end{aligned} \tag{9}$$

Authorized licensed use limited to: National Univ of Defense Tech. Downloaded on February 07,2024 at 15:34:14 UTC from IEEE Xplore. Restrictions apply.

Note that the condition for (9) is that we assume $\kappa(\mathbf{x}_i, \mathbf{x}_i) = 1$. However, it is not always valid for all the kernel functions. A common choice is the Gaussian kernel which satisfies $\kappa(\mathbf{x}_i, \mathbf{x}_i) = 1$. The present work utilizes this manner or directly downloads the public kernel datasets. Moreover, all the base kernels are first centered and then normalized following [63] and [64], which further guarantees $\kappa(\mathbf{x}_i, \mathbf{x}_i) = 1$.

We have the following insights from the kernelized affinity graph learning model: 1) compared with using $\|\mathbf{x}_i - \mathbf{x}_j\|_2^2$ to estimate the pairwise distance in sample space, we should adopt $-\kappa(\mathbf{x}_i, \mathbf{x}_j)$ in kernel space. 2) Such compact form achieves affinity graph learning in kernel space to explore the complex nonlinear structures.

In multiple kernel learning scenarios, it is commonly assumed that the ideal kernel is optimally combined by given base kernels, and (9) can be extended as

$$\begin{aligned} &\min_{\mathbf{Z},\boldsymbol{\omega}} \sum_{p=1}^{m} \sum_{i,j=1}^{n} -\omega_p \kappa_p(\mathbf{x}_i, \quad \mathbf{x}_j) z_{ij} + \gamma z_{ij}^2 \\ &\text{s.t.} \begin{cases} \mathbf{Z}_{i,:}\mathbf{1}_n = 1, & z_{ij} \geq 0, \quad z_{ii} = 0 \\ \sum_{p=1}^{m} \omega_p^2 = 1, & \omega_p \geq 0 \end{cases} \end{aligned} \tag{10}$$

where $\omega_p$ is the weight of $p$-th base kernel. Since using $\sum_{p=1}^{m} \omega = 1$ will only activate the best kernel, and it incurs the multi-kernel scenario degraded into the undesirable single-kernel scenario. We employ the squared $\ell_2$ norm constraint of $\omega_p$ to smooth the weights and avoid the sparse trivial solution. Other weight strategies can refer to [43], [48], and [51]. The above-mentioned formula achieves multiple kernel-based graph learning by jointly optimizing kernel weights and consensus affinity graph. Specifically, the learned consensus discriminative graph reveals kernel space's intrinsic local manifold structures by graph learning mechanism and fuses latent clustering information across multiple kernels by weight learning mechanism.

Recall we aim to estimate the ranking relationship of neighbors with corresponding samples in kernel space. The above-mentioned discriminative consensus graph inspires us to further learn an optimal neighborhood kernel, which obtains a consensus kernel with naturally sparse properties and precise block diagonal structures. This idea can be naturally modeled by minimizing squared F-norm loss $\|\mathbf{K}^* - \mathbf{Z}\|_{\mathrm{F}}^2$ with constraints $\mathbf{K}^* \succeq 0$ and $\mathbf{K}^* = \mathbf{K}^{*\top}$. We define the optimization goal as follows:

$$\begin{aligned} &\min_{\mathbf{Z},\mathbf{K}^*,\boldsymbol{\omega}} -\mathrm{Tr}\left(\sum_{p=1}^{m} \omega_p \mathbf{K}_p \mathbf{Z}^\top\right) + \|\mathbf{G} \odot \mathbf{Z}\|_{\mathrm{F}}^2 + \alpha \|\mathbf{K}^* - \mathbf{Z}\|_2^2 \\ &\text{s.t.} \begin{cases} \mathbf{Z}\mathbf{1}_n = \mathbf{1}_n, \quad \mathbf{Z} \geq 0, \ \mathbf{Z}_{ii} = 0 \\ \mathbf{K}^* \succeq 0, \quad \mathbf{K}^* = \mathbf{K}^{*\top}, \ \sum_{p=1}^{m} \omega_p^2 = 1, \ \omega_p \geq 0 \end{cases} \end{aligned} \tag{11}$$

where $\mathbf{G} = \mathbf{1}_n^\top \otimes \boldsymbol{\gamma}$, $\boldsymbol{\gamma} = (\sqrt{\gamma_1}, \sqrt{\gamma_2}, \ldots, \sqrt{\gamma_n})^\top$ denotes hyperparameter $\gamma_i$ with corresponding $i$-row of $\mathbf{Z}$, $\otimes$ is outer product, $\odot$ is the Hadamard product, and $\alpha$ is the balanced hyperparameter for neighborhood kernel construction.

Note that $n$ hyperparameters $\gamma$ corresponding to $n$ rows of $\mathbf{Z}$ respectively, which is due to the following considerations: 1) as our analysis in (10), reasonable hyperparameters $\gamma$ can avoid trivial solutions, i.e., $\gamma \to 0$ or $\gamma \to \infty$ will incur undesired extremely sparse or dense affinity matrix, respectively. 2) Section III-C2 also illustrates the subproblem of optimizing $\mathbf{Z}$ involves $n$-row formed independent optimization. It is reasonable to assign different $\gamma_i$ to each problem, considering their variations. Such issues inspire us to learn reasonable $\gamma$ instead of empirical and time-consuming parameter tuning. We derive a theoretical solution in Section III-D and experimentally validate the ablation study on tuning $\gamma$ by grid search in Section IV-J.

From the above-mentioned formula, our proposed LSWMKC model jointly optimizes the kernel weights, the consensus affinity graph, and the consensus neighborhood kernel into a unified framework. Although the formula is straightforward, LSWMKC has the following merits: 1) it addresses localized kernel problems via a heuristic manner, rather than the traditional KNN mechanism, which achieves implicitly optimizing adaptive weights on different neighbors with corresponding samples according to their ranking relationship. 2) Instead of tuning hyperparameter $\boldsymbol{\gamma}$ by grid search, we propose an elegant solution to predetermine it. 3) More advanced graph learning methods in kernel space can be easily introduced to this framework.

### C. Optimization

Simultaneously optimizing all the variables in (11) is difficult since the optimization objective is not convex. This section provides an effective alternate optimization strategy by optimizing each variable with others been fixed. The original problem is separated into three subproblems such that each one is convex.

*1) Optimization $\omega_p$ With Fixed $\mathbf{Z}$ and $\mathbf{K}^*$:* With fixed $\mathbf{Z}$ and $\mathbf{K}^*$, the objective in (11) is formulated as

$$\max_{\boldsymbol{\omega}} \sum_{p=1}^{m} \omega_p \delta_p, \quad \text{s.t.} \sum_{p=1}^{m} \omega_p^2 = 1, \omega_p \geq 0 \tag{12}$$

where $\delta_p = \mathrm{Tr}(\mathbf{K}_p \mathbf{Z}^\top)$. This problem could be easily solved with closed-form solution as follows:

$$\omega_p = \frac{\delta_p}{\sqrt{\sum_{p=1}^{m} \delta_p^2}}. \tag{13}$$

The computational complexity is $\mathcal{O}(mn^2)$.

*2) Optimization $\mathbf{Z}$ With Fixed $\mathbf{K}^*$ and $\omega_p$:* With fixed $\mathbf{K}^*$ and $\omega_p$, (11) is transformed to $n$ subproblems, and each one can be independently solved by

$$\begin{aligned} &\min_{\mathbf{Z}_{i,:}} (\gamma_i + \alpha)\mathbf{Z}_{i,:}\mathbf{Z}_{i,:}^\top - \left(2\alpha \mathbf{K}_{i,:}^* + \sum_{p=1}^{m} \omega_p \mathbf{K}_{p[i,:]}\right)\mathbf{Z}_{i,:}^\top \\ &\text{s.t. } \mathbf{Z}_{i,:}\mathbf{1}_n = 1, \quad \mathbf{Z}_{i,:} \geq 0, \ \mathbf{Z}_{ii} = 0 \end{aligned} \tag{14}$$

where $\mathbf{K}_{p[i,:]}$ denotes the $i$-th row of the $p$-th base kernel.

Furthermore, (14) can be rewritten as quadratic programming (QP) problem

$$\begin{aligned} &\min_{\mathbf{Z}_{i,:}} \frac{1}{2}\mathbf{Z}_{i,:}\mathbf{A}\mathbf{Z}_{i,:}^\top + \mathbf{e}_i \mathbf{Z}_{i,:}^\top \\ &\text{s.t. } \mathbf{Z}_{i,:}\mathbf{1}_n = 1, \quad \mathbf{Z}_{i,:} \geq 0, \ \mathbf{Z}_{ii} = 0 \end{aligned} \tag{15}$$

Authorized licensed use limited to: National Univ of Defense Tech. Downloaded on February 07,2024 at 15:34:14 UTC from IEEE Xplore. Restrictions apply.

where $\mathbf{A} = 2(\gamma_i+\alpha)\mathbf{I}_n$, $\mathbf{e}_i = -(2\alpha\mathbf{K}^*_{i,:}+\sum_{p=1}^m \omega_p\mathbf{K}_{p[i,:]})$. The global optimal solution of QP problem can be easily solved by the toolbox of MATLAB. Since $\mathbf{Z}_{i,:}$ is a $n$-dimensional row vector, the computational complexity of (15) is $\mathcal{O}(n^3 + mn)$ and the total complexity is $\mathcal{O}(n^4 + mn^2)$.

Furthermore, (15) can be simplified as

$$\min_{\mathbf{Z}_{i,:}} \frac{1}{2}\left\|\mathbf{Z}_{i,:} - \hat{\mathbf{Z}}_{i,:}\right\|_2^2 \quad \text{s.t. } \mathbf{Z}_{i,:}\mathbf{1}_n = 1, \quad \mathbf{Z}_{i,:} \geq 0,\ \mathbf{Z}_{ii} = 0 \tag{16}$$

where $\hat{\mathbf{Z}}_{i,:} = -(\mathbf{e}_i/(2(\alpha + \gamma_i)))$.

Mathematically, the following Theorem 1 illustrates that the solution of (16) can be analytically solved.

*Theorem 1:* The analytical solution of (16) is as follows:

$$\mathbf{Z}_{i,:} = \max\left(\hat{\mathbf{Z}}_{i,:} + \beta_i\mathbf{1}_n^\top, 0\right), \quad \mathbf{Z}_{ii} = 0 \tag{17}$$

where $\beta_i$ can be solved by Newton's method efficiently.

*Proof:* For $i$-th row of $\mathbf{Z}$, the Lagrangian function of (16) is as follows:

$$\mathcal{L}\left(\mathbf{Z}_{i,:}, \beta_i, \boldsymbol{\eta}_i\right) = \frac{1}{2}\left\|\mathbf{Z}_{i,:} - \hat{\mathbf{Z}}_{i,:}\right\|_2^2 - \beta_i\left(\mathbf{Z}_{i,:}\mathbf{1}_n - 1\right) - \boldsymbol{\eta}_i\mathbf{Z}_{i,:}^\top \tag{18}$$

where scalar $\beta_i$ and row vector $\boldsymbol{\eta}_i$ are Lagrangian multipliers. According to the KKT condition

$$\begin{cases} \mathbf{Z}_{i,:} - \hat{\mathbf{Z}}_{i,:} - \beta_i\mathbf{1}_n^\top - \boldsymbol{\eta}_i = \mathbf{0}^\top \\ \boldsymbol{\eta}_i \odot \mathbf{Z}_{i,:} = \mathbf{0}^\top. \end{cases} \tag{19}$$

We have

$$\mathbf{Z}_{i,:} = \max\left(\hat{\mathbf{Z}}_{i,:} + \beta_i\mathbf{1}_n^\top, 0\right), \quad \mathbf{Z}_{ii} = 0. \tag{20}$$

Note that $\mathbf{Z}_{i,:}\mathbf{1}_n$ increases monotonically with respect to $\beta_i$ according to (20), $\beta_i$ can be solved by Newton's method efficiently with the constraint $\mathbf{Z}_{i,:}\mathbf{1}_n = 1$. This completes the proof. □

By computing the closed-formed solution, the computational complexity of (15) is reduced to $\mathcal{O}(mn)$, which is mainly from computing $\mathbf{e}_i$. The total complexity is $\mathcal{O}(mn^2)$.

*3) Optimization $\mathbf{K}^*$ With Fixed $\mathbf{Z}$ and $\omega_p$:* With fixed $\mathbf{Z}$ and $\omega_p$, the original objective (11) can be converted to

$$\min_{\mathbf{K}^*} \left\|\mathbf{K}^* - \mathbf{Z}\right\|_{\mathrm{F}}^2 \quad \text{s.t. } \mathbf{K}^* \succeq 0,\ \mathbf{K}^* = \mathbf{K}^{*\top}. \tag{21}$$

However, this seemingly simple subproblem is hard to be directly solved. Theorem 2 provides an equivalent solution.

*Theorem 2:* The optimization in (21) has the same solution as (22)

$$\min_{\mathbf{K}^*} \left\|\mathbf{K}^* - \frac{1}{2}(\mathbf{Z} + \mathbf{Z}^\top)\right\|_{\mathrm{F}}^2 \quad \text{s.t. } \mathbf{K}^* \succeq 0,\ \mathbf{K}^* = \mathbf{K}^{*\top}. \tag{22}$$

*Proof:* According to the PSD property of $\mathbf{K}^*$, we can derive that the original optimization objective $\|\mathbf{K}^* - \mathbf{Z}\|_{\mathrm{F}}^2$ in (21) is equivalent to $\|\mathbf{K}^* - \mathbf{Z}^\top\|_{\mathrm{F}}^2$. Therefore, the solution of (21) is the same as (22). This completes the proof. □

According to Theorem 2, supposing the eigenvalue decomposition result of $(\mathbf{Z} + \mathbf{Z}^\top)/2$ is $\mathbf{U}_\mathbf{Z}\Sigma_\mathbf{Z}\mathbf{U}_\mathbf{Z}^\top$. The optimal $\mathbf{K}^*$ can be easily obtained by imposing $\mathbf{K}^* = \mathbf{U}_\mathbf{Z}\Sigma\mathbf{U}_\mathbf{Z}^\top$, where $\Sigma = \max(\Sigma_\mathbf{Z}, 0)$. Note that the learned $\mathbf{K}^*$ can further denoise the $\mathbf{Z}$ from the above-mentioned optimization. Once we obtain $\mathbf{K}^*$, it is exported to KKM to calculate the final results.

### *D. Initialize the Affinity Graph $\mathbf{Z}$ and Hyperparameter $\gamma_i$*

For graph-based clustering methods, the performance is sensitive to the initial affinity graph. A bad graph construction will degrade the overall performance. For the proposed algorithm, we aim to learn a neighborhood kernel $\mathbf{K}^*$ of the consensus affinity graph $\mathbf{Z}$. This section proposes a strategy to initialize the affinity matrix $\mathbf{Z}$ and the hyperparameter $\gamma_i$.

Recalling our objective in (11), a sparse discriminative affinity graph is preferred. Theoretically, by constraining $\gamma_i$ within reasonable bounds, $\mathbf{Z}$ will be naturally sparse. The $c$ nonzero values of $\mathbf{Z}_{i,:}$ denotes the affinity of each instance corresponding to its initialized neighbors. Therefore, with all the other parameters fixed, we learn an initialized $\mathbf{Z}$ with the maximal $\gamma_i$. Based on our objective in (11), by constraining the $\ell_0$-norm of $\mathbf{Z}_{i,:}$ to be $c$, we solve the following problem:

$$\max_{\gamma_i} \gamma_i, \quad \text{s.t. } \|\mathbf{Z}_{i,:}\|_0 = c. \tag{23}$$

Recall the subproblem of optimizing $\mathbf{Z}$ in (16), its equivalent form can be written as follows:

$$\min_{\mathbf{Z}_{i,:}\mathbf{1}_n=1,\ \mathbf{Z}_{i,:}\geq 0,\ \mathbf{Z}_{ii}=0} \frac{1}{2}\left\|\mathbf{Z}_{i,:} + \frac{\mathbf{e}_i}{2(\alpha + \gamma_i)}\right\|_2^2 \tag{24}$$

where $\mathbf{e}_i = -(2\alpha\mathbf{K}^*_{i,:} + \sum_{p=1}^m \omega_p\mathbf{K}_{p[i,:]})$. The Lagrangian function of (24) is

$$\mathcal{L}\left(\mathbf{Z}_{i,:}, \zeta, \boldsymbol{\lambda}_i\right) = \frac{1}{2}\left\|\mathbf{Z}_{i,:} + \frac{\mathbf{e}_i}{2(\alpha + \gamma_i)}\right\|_2^2 - \zeta\left(\mathbf{Z}_{i,:}\mathbf{1}_n - 1\right) - \boldsymbol{\lambda}_i\mathbf{Z}_{i,:}^\top \tag{25}$$

where scalar $\zeta$ and row vector $\boldsymbol{\lambda}_i \geq \mathbf{0}^\top$ denote the Lagrange multipliers. The optimal solution $\mathbf{Z}^*_{i,:}$ satisfy that the derivative of (25) equal to zero, that is,

$$\mathbf{Z}^*_{i,:} + \frac{\mathbf{e}_i}{2(\alpha + \gamma_i)} - \zeta\mathbf{1}_n^\top - \boldsymbol{\lambda}_i = \mathbf{0}^\top. \tag{26}$$

For the $j$-th element of $\mathbf{Z}^*_{i,:}$, we have

$$z^*_{ij} + \frac{e_{ij}}{2(\alpha + \gamma_i)} - \zeta - \lambda_{ij} = 0. \tag{27}$$

According to the KKT condition that $z_{ij}\lambda_{ij} = 0$, we have

$$z^*_{ij} = \max\left(-\frac{e_{ij}}{2(\alpha + \gamma_i)} + \zeta, 0\right). \tag{28}$$

To construct a sparse affinity graph with $c$ valid neighbors, we suppose each row $e_{i1}, e_{i2}, \ldots, e_{in}$ are ordered in ascending order. Naturally, $e_{ii}$ ranks first. Considering $\mathbf{Z}_{ii} = 0$, the invalid $e_{ii}$ should be neglected since the similarity with itself is useless. That is $\mathbf{Z}_{i,2}, \mathbf{Z}_{i,3}, \ldots, \mathbf{Z}_{i,c+1}>0$ and $\mathbf{Z}_{i,c+2}, \mathbf{Z}_{i,c+3}, \ldots, \mathbf{Z}_{i,n} = 0$, we further derive

$$-\frac{e_{i,c+1}}{2(\alpha + \gamma_i)} + \zeta > 0, \quad -\frac{e_{i,c+2}}{2(\alpha + \gamma_i)} + \zeta \leq 0. \tag{29}$$

Authorized licensed use limited to: National Univ of Defense Tech. Downloaded on February 07,2024 at 15:34:14 UTC from IEEE Xplore. Restrictions apply.

According to (28) and constraint $\mathbf{Z}_{i,:}\mathbf{1}_n = 1$, we obtain

$$\sum_{j=2}^{c+1}\left(-\frac{e_{ij}}{2(\alpha+\gamma_i)}+\zeta\right)=1. \tag{30}$$

$\zeta$ is formulated as

$$\zeta = \frac{1}{c}+\frac{1}{2c(\alpha+\gamma_i)}\sum_{j=2}^{c+1}e_{ij}. \tag{31}$$

Therefore, we have

$$\frac{c}{2}e_{i,c+1}-\frac{1}{2}\sum_{j=2}^{c+1}e_{ij}-\alpha<\gamma_i\leq\frac{c}{2}e_{i,c+2}-\frac{1}{2}\sum_{j=2}^{c+1}e_{ij}-\alpha. \tag{32}$$

According to the aforementioned derivation, to satisfy $\|\mathbf{Z}^*_{i,:}\|_0 = c$, the maximal $\gamma_i$ is as follows:

$$\gamma_i = \frac{c}{2}e_{i,c+2}-\frac{1}{2}\sum_{j=2}^{c+1}e_{ij}-\alpha. \tag{33}$$

In the meantime, the initial $z^*_{ij}$ is as follows:

$$z^*_{ij}=\begin{cases}\dfrac{e_{i,c+2}-e_{i,j+1}}{ce_{i,c+2}-\sum_{h=2}^{c+1}e_{ih}}, & j\leq c\\ 0, & j>c.\end{cases} \tag{34}$$

From the above-mentioned analysis, we initialize a sparse discriminative affinity graph with each row having $c$ nonzero values and derive the maximal $\gamma_i$. Note that (32) involves an undesired hyperparameter $\alpha$, to get rid of its impact, we directly impose $\alpha = 0$. Once the initial $\gamma_i$ are computed, these coefficients will remain unchanged during the iteration. According to the initialization, we have the following observations: 1) the construction is simple with basic operations, but can effectively initialize a sparse discriminative affinity graph with block-diagonal structures, contributing to the subsequent learning process. 2) The hyperparameter $\gamma_i$ can be predetermined to avoid the undesired tuning by grid search. 3) Initializing the affinity graph involves a parameter, i.e., the number of neighbors $c$. For most cases, $5 \leq c \leq 10$ is likely to achieve reasonable results and $c$ is fixed at 5 in this work.

### E. Analysis and Extensions

*1) Computational Complexity:* According to the aforementioned alternate optimization steps, the computational complexity of our LSWMKC model includes three parts. Updating $\omega_p$ in (12) needs $\mathcal{O}(mn^2)$ to obtain the closed-form solution. When updating $\mathbf{Z}$, the complex QP problem in (15) is transformed into an equivalent closed-form solution in (16) whose computational complexity is $\mathcal{O}(mn^2)$. Updating $\mathbf{K}^*$ in (22) needs $\mathcal{O}(n^3)$ cost by eigenvalue decomposition. Commonly, $n \gg m$, the total computational complexity of our LSWKMC is $\mathcal{O}(n^3)$ in each iteration.

For the postprocessing of $\mathbf{K}^*$, we perform KKM to obtain the clustering partition and labels whose computational complexity is $\mathcal{O}(n^3)$. Although the computational complexity of our LSWMKC algorithm is the same as the compared models [14]–[16], [19], [24], [36], [40], [48], [51], its clustering performance exhibits significant improvement, as reported in Section IV-D.

**Algorithm 1** LSWMKC

**Input**: Base kernel matrices $\{\mathbf{K}_p\}_{p=1}^m$, clusters $k$, neighbors $c$, hyperparameter $\alpha$.
**Initialize:** $\mathbf{Z}$ by (34); $\mathbf{K}^* = \sum_{p=1}^m \omega_p \mathbf{K}_p$; $\gamma_i$ by (33); $\omega_p = \sqrt{1/m}$.
**while** *not converged* **do**
  Compute $\omega_p$ according to (12);
  Compute $\mathbf{Z}$ according to (16);
  Compute $\mathbf{K}^*$ according to (22);
**end**
**Output**: Perform kernel $k$-means on $\mathbf{K}^*$.

*2) Convergence:* Jointly optimizing all the variables in (11) is problematic since our algorithm is nonconvex. Instead, as Algorithm 1 shows, we adopt an alternate optimization manner, and each of the subproblems is strictly convex. For each subproblem, the objective function decreases monotonically during iteration. Consequently, as pointed out in [65], the proposed model can theoretically obtain a local minimum solution.

*3) Limitation and Extension:* The proposed model provides a heuristic insight into the localized mechanism in kernel space. Nevertheless, we should emphasize the promising performance obtained at the expense of $\mathcal{O}(n^3)$ computational complexity, which limits wide applications in large-scale clustering. Introducing more advanced and efficient graph learning methods to this framework deserve future investigation, especially for prototype or anchor learning [49], [52], [66], which may reduce the complexity from $\mathcal{O}(n^3)$ to $\mathcal{O}(n^2)$, even $\mathcal{O}(n)$. Moreover, the present work still requires postprocessing to get the final clustering results, i.e., $k$-means. Interestingly, several concise strategies, such as rank constraint [41], [48], [52] or one-pass manner [25], provide promising solutions of directly obtaining the clustering labels, these deserve further research.

## IV. Experiment

This section conducts extensive experiments to evaluate the performance of our proposed algorithm, including clustering performance, running time, comparison with the KNN mechanism, kernel weights, visualization, convergence, parameter sensitivity analysis, and ablation study.

### A. Datasets

Table I lists 12 widely employed multi-kernel benchmark datasets, including the following:

1) **YALE**[1] includes 165 face gray-scale images from 15 individuals with different facial expressions or configurations, and each subject includes 11 images.
2) **MSRA** derived from MSRCV1 [67], contains 210 images with seven clusters, including airplane, bicycle, building, car, caw, face, and tree.

[1] http://vision.ucsd.edu/content/yale-face-database

Authorized licensed use limited to: National Univ of Defense Tech. Downloaded on February 07,2024 at 15:34:14 UTC from IEEE Xplore. Restrictions apply.

TABLE I
DATASETS SUMMARY

| Datasets | Samples | Views | Clusters |
|---|---|---|---|
| YALE | 165 | 5 | 15 |
| MSRA | 210 | 6 | 7 |
| Caltech101-7 | 441 | 6 | 7 |
| PsortPos | 541 | 69 | 4 |
| BBC | 544 | 2 | 5 |
| BBCSport | 544 | 6 | 5 |
| ProteinFold | 694 | 12 | 27 |
| PsortNeg | 1444 | 69 | 5 |
| Caltech101-mit | 1530 | 25 | 102 |
| Handwritten | 2000 | 6 | 10 |
| Mfeat | 2000 | 12 | 10 |
| Scene15 | 4485 | 3 | 15 |

3) **Caltech101-7 and Caltech101-mit**[2] originated from Caltech101, including 101 object categories (e.g., "face," "dollar bill," and "helicopter") and a background category.
4) **PsortPos** and **PsortNeg**[3] are bioinformatics MKL datasets used for protein subcellular localization research.
5) **BBC** and **BBCSport**[4] are two news corpora datasets derived from BBC News, consisting of various documents corresponding to stories or sports news in five areas.
6) **ProteinFold**[5] is a bioinformatics dataset containing 694 protein patterns and 27 protein folds.
7) **Handwritten**[6] and **Mfeat**[7] are image datasets originated from the UC Irvine Machine Learning (UCI ML) repository, including 2000 digits of handwritten numerals ("0"–"9").
8) **Scene-15**[8] contains 4485 gray-scale images, 15 environmental categories, and three features [Generalized Search Trees (GIST), Pyramid Histogram of Gradients (PHOG), and Local Binary Patterns (LBP)].

All the precomputed base kernels within the datasets are publicly available on websites and are centered and then normalized following [63] and [64].

### B. Compared Algorithms

Thirteen existing multiple kernel or graph-based algorithms are compared with our proposed model, including the following:

1) **Avg-KKM** combines base kernels with uniform weights.
2) **MKKM** [19] optimally combines multiple kernels by alternatively performing KKM and updating the kernel weights.
3) **Localized Multiple Kernel k-means (LMKKM)** [14] can optimally fuse base kernels via an adaptive sample-weighted strategy.
4) **Multiple Kernel k-Means Clustering with Matrix-Induced Regularization (MKKM-MR)** [15] improve the diversity of kernels by introducing a matrix-induced regularization term.
5) **Multiple Kernel Clustering with Local Alignment Maximization (LKAM)** [36] introduces localized kernel maximizing alignment by constraining $\tau$-nearest neighbors of each sample.
6) **Optimal Neighborhood Kernel Clustering (ONKC)** [16] regards the optimal kernel as the neighborhood kernel of the combined kernel.
7) **Self-weighted Multiview Clustering with Multiple Graphs (SwMC)** [57] eliminates the undesired hyperparameter via a self-weighted strategy.
8) **Multi-view Clustering via Late Fusion Alignment Maximization (LF-MVC)** [17] aims to achieve maximal alignment of consensus partition and base ones via a late fusion manner.
9) **Simultaneous Global and Local Graph Structure Preserving for Multiple Kernel Clustering (SPMKC)** [51] simultaneously performs consensus kernel learning and graph learning.
10) **Simple Multiple Kernel k-means (SMKKM)** [24] proposes a novel min–max optimization based on kernel alignment criterion.
11) **Consensus Affinity Graph Learning for Multiple Kernel Clustering (CAGL)** [48] proposes a multi-kernel graph-based clustering model to directly learn a consensus affinity graph with rank constraint.
12) **One Pass Late Fusion Multi-view Clustering (OPLFMVC)** [25] can directly learn the cluster labels on the base partition level.
13) **Localized Simple Multiple Kernel k-means (LSMKKM)** [40] is localized SMKKM in the KNN method.

### C. Experimental Settings

Regarding the benchmark datasets, it is commonly assumed that the true number of clusters $k$ is known. For the methods involving $k$-means, the centroid of clusters is repeatedly and randomly initialized 50 times to reduce its randomness and report the best results. Regarding all the compared algorithms, we directly download the public MATLAB code and carefully tune the hyperparameters following the original suggestion. For our proposed LSWMKC, the balanced hyperparameter $\alpha$ varies in $[2^0, 2^1, \ldots, 2^{10}]$ by grid search. The clustering performance is evaluated by four widely employed criteria, including clustering accuracy (ACC), normalized mutual information (NMI), purity, and adjusted rand index (ARI). The experimental results are obtained from a desktop with Intel Core i7 8700K CPU (3.7 GHz), 64-GB RAM, and MATLAB 2020b (64bit).

### D. Experimental Results

Table II reports ACC, NMI, Purity, and ARI comparisons of 14 algorithms on 12 datasets. Red bold denotes the optimal results. Blue bold denotes the suboptimal results while "-" denotes unavailable results due to overmuch execution time. According to the experimental results, it can be seen that the following holds.

1) Our proposed LSWMKC algorithm achieves optimal or suboptimal performance on most datasets. Particularly,

[2]http://www.vision.caltech.edu/Image_Datasets/Caltech101/
[3]https://bmi.inf.ethz.ch/supplements/protsubloc
[4]http://mlg.ucd.ie/datasets/bbc.html
[5]mkl.ucsd.edu/dataset/protein-fold-prediction
[6]http://archive.ics.uci.edu/ml/datasets/
[7]https://datahub.io/machine-learning/mfeat-pixel
[8]https://www.kaggle.com/yiklunchow/scene15

Authorized licensed use limited to: National Univ of Defense Tech. Downloaded on February 07,2024 at 15:34:14 UTC from IEEE Xplore. Restrictions apply.

TABLE II

ACC, NMI, PURITY, AND ARI COMPARISONS OF 14 CLUSTERING ALGORITHMS ON 12 BENCHMARK DATASETS

| Datasets | Avg-KKM | MKKM (2011) | LMKKM (2014) | MKKM-MR (2016) | LKAM (2016) | ONKC (2017) | SwMC (2017) | LF-MVC (2019) | SPMKC (2020) | SMKKM (2020) | CAGL (2020) | OPLFMVC (2021) | LSMKKM (2021) | Proposed |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | ACC (%) | | | | | | | | | | | | | |
| YALE | 54.73 | 52.00 | 52.27 | 56.24 | 58.88 | 56.36 | 46.67 | 55.00 | **65.45** | 56.03 | 53.33 | 55.76 | 59.24 | **66.67** |
| MSRA | 83.33 | 81.29 | 81.93 | 88.07 | 89.14 | 85.36 | 23.33 | 87.76 | 79.05 | 86.50 | **99.05** | 87.14 | **91.19** | 90.95 |
| Caltech101-7 | 59.17 | 52.15 | 53.89 | 68.44 | 70.39 | 69.42 | 54.65 | 71.39 | 62.59 | 68.15 | **78.91** | 73.47 | 76.21 | **76.64** |
| PsortPos | 56.94 | 60.70 | **61.84** | 49.21 | 53.08 | 50.41 | 37.71 | 53.21 | 36.04 | 43.70 | 48.80 | 56.38 | 49.50 | **65.06** |
| BBC | 63.17 | 63.03 | 63.90 | 63.17 | 73.85 | 63.35 | 36.03 | 76.42 | 88.79 | 64.20 | 76.10 | **90.26** | 73.58 | **96.51** |
| BBCSport | 66.25 | 66.24 | 66.58 | 66.17 | 76.58 | 66.43 | 36.03 | 76.46 | 40.81 | 66.76 | **89.15** | 81.25 | 77.11 | **97.24** |
| ProteinFold | 28.97 | 26.99 | 22.41 | 34.72 | **37.73** | 36.27 | 14.99 | 33.00 | 21.61 | 34.68 | 32.28 | 35.88 | 35.91 | **36.60** |
| PsortNeg | 41.01 | **51.88** | - | 39.71 | 40.53 | 40.15 | 26.59 | 45.52 | 25.14 | 41.54 | 27.77 | 48.13 | 45.69 | **52.77** |
| Caltech101-mit | 34.16 | 32.81 | 27.94 | 34.75 | 32.28 | 34.02 | 22.42 | 34.41 | 36.99 | 35.85 | **44.18** | 24.84 | 36.96 | **39.35** |
| Handwritten | 95.99 | 64.94 | 65.03 | 88.66 | 95.40 | 89.51 | 58.50 | 95.80 | 28.15 | 93.57 | 88.25 | 92.25 | **96.48** | **97.45** |
| Mfeat | 93.83 | 64.31 | - | 88.53 | 82.28 | 88.85 | 78.65 | 92.85 | 16.95 | 94.19 | 87.50 | 93.80 | **96.95** | **97.50** |
| Scene15 | 43.17 | 41.18 | 40.85 | 38.41 | 41.42 | 39.93 | 11.33 | **45.82** | 11.82 | 43.60 | 22.30 | 43.26 | 43.80 | **48.58** |
| | NMI (%) | | | | | | | | | | | | | |
| YALE | 57.32 | 54.35 | 54.56 | 58.63 | 60.23 | 59.54 | 48.86 | 57.54 | **64.11** | 58.91 | 59.93 | 56.90 | 60.31 | **66.15** |
| MSRA | 73.99 | 73.22 | 75.01 | 77.59 | 79.83 | 74.89 | 22.86 | 79.39 | 69.35 | 75.17 | **97.85** | 78.96 | 82.63 | **85.15** |
| Caltech101-7 | 59.07 | 51.60 | 52.13 | 64.12 | 65.35 | 63.52 | 58.20 | 70.08 | 56.06 | 63.73 | **83.85** | 69.37 | **74.15** | 71.72 |
| PsortPos | 28.73 | 35.50 | **37.16** | 21.13 | 24.54 | 25.43 | 2.28 | 24.95 | 5.48 | 23.76 | 24.19 | 28.33 | 24.01 | **39.65** |
| BBC | 43.50 | 43.58 | 44.01 | 43.46 | 65.42 | 43.53 | 2.00 | 58.86 | 74.60 | 44.45 | **80.81** | 79.69 | 65.09 | **90.05** |
| BBCSport | 54.18 | 54.09 | 54.37 | 53.85 | 54.50 | 53.51 | 3.71 | 57.59 | 6.75 | 49.34 | **79.82** | 65.25 | 54.81 | **91.03** |
| ProteinFold | 40.32 | 38.03 | 34.68 | 43.70 | **46.25** | 44.38 | 7.91 | 41.72 | 33.03 | 44.44 | 41.56 | 41.90 | 45.15 | **46.03** |
| PsortNeg | 17.39 | **32.16** | - | 21.65 | 21.76 | 21.03 | 0.66 | 18.75 | 0.31 | 19.05 | 12.21 | 23.25 | 17.01 | **30.20** |
| Caltech101-mit | 59.30 | 58.57 | 55.26 | 59.72 | 58.48 | 59.30 | 30.91 | 59.55 | 60.11 | 60.35 | **66.12** | 52.86 | 61.37 | **62.91** |
| Handwritten | 91.09 | 64.79 | 64.74 | 79.44 | 91.83 | 80.66 | 61.38 | 90.91 | 15.98 | 87.42 | 92.30 | 84.80 | **93.56** | **94.17** |
| Mfeat | 89.09 | 59.82 | - | 80.41 | 84.89 | 80 | 84.56 | 88.60 | 3.82 | 88.64 | 91.34 | 87.09 | **93.18** | **94.31** |
| Scene15 | 41.31 | 38.62 | 38.79 | 37.25 | 42.14 | 37.73 | 2.61 | **42.71** | 2.89 | 40.60 | 29.36 | 41.88 | 40.97 | **46.70** |
| | Purity (%) | | | | | | | | | | | | | |
| YALE | 55.42 | 52.94 | 53.06 | 56.58 | 59.42 | 57.18 | 50.91 | 56.03 | **66.06** | 56.42 | 55.15 | 56.97 | 59.88 | **67.27** |
| MSRA | 83.33 | 81.45 | 81.93 | 88.07 | 89.14 | 85.36 | 30.48 | 87.76 | 79.05 | 86.50 | **99.05** | 87.14 | **91.19** | 90.95 |
| Caltech101-7 | 68.05 | 63.84 | 66.39 | 72.93 | 76.55 | 73.97 | 64.63 | 79.59 | 68.93 | 72.34 | **83.22** | 80.27 | **81.42** | 81.41 |
| PsortPos | 60.74 | 66.66 | **68.03** | 56.14 | 61.03 | 60.79 | 37.71 | 57.07 | 46.03 | 57.63 | 48.80 | 60.63 | 53.72 | **68.76** |
| BBC | 68.06 | 68.15 | 68.40 | 68.03 | 79.39 | 68.10 | 36.76 | 76.75 | 88.79 | 68.68 | 76.29 | **90.26** | 79.17 | **96.51** |
| BBCSport | 77.33 | 77.27 | 77.50 | 77.10 | 76.58 | 76.99 | 37.87 | 78.30 | 40.81 | 73.52 | **89.15** | 81.25 | 77.11 | **97.24** |
| ProteinFold | 37.39 | 33.70 | 31.16 | 41.89 | **43.70** | 42.67 | 18.30 | 39.33 | 28.24 | 41.79 | 35.88 | 38.33 | 42.52 | **42.80** |
| PsortNeg | 43.33 | **56.61** | - | 44.66 | 45.29 | 44.67 | 27.22 | 48.22 | 27.08 | 42.17 | 30.96 | 51.80 | 47.17 | **57.06** |
| Caltech101-mit | 36.22 | 34.88 | 29.56 | 36.77 | 34.30 | 36.16 | 26.08 | 36.65 | 39.22 | 37.96 | **46.80** | 25.75 | 39.25 | **41.31** |
| Handwritten | 95.99 | 65.84 | 65.52 | 88.66 | 95.44 | 89.51 | 58.70 | 95.80 | 30.50 | 93.57 | 88.25 | 92.25 | **96.52** | **97.45** |
| Mfeat | 94.13 | 64.95 | - | 88.53 | 86.02 | 88.85 | 78.80 | 93.27 | 17.60 | 94.19 | 87.85 | 93.80 | **96.95** | **97.70** |
| Scene15 | 47.85 | 44.29 | 44.30 | 42.40 | 46.01 | 43.60 | 11.62 | **49.36** | 13.00 | 48.38 | 22.52 | 47.65 | 48.62 | **50.81** |
| | ARI (%) | | | | | | | | | | | | | |
| YALE | 33.93 | 30.42 | 30.50 | 35.49 | 37.31 | 36.56 | 13.17 | 34.29 | **43.70** | 35.86 | 32.56 | 34.21 | 37.89 | **45.06** |
| MSRA | 68.14 | 66.22 | 68.00 | 74.46 | 76.66 | 69.76 | 6.90 | 74.52 | 59.60 | 71.17 | **97.77** | 74.11 | 80.63 | **81.38** |
| Caltech101-7 | 46.02 | 38.30 | 41.23 | 55.62 | 59.44 | 56.75 | 40.59 | 65.19 | 45.01 | 55.64 | **74.40** | 65.14 | 68.81 | **74.34** |
| PsortPos | 24.36 | **32.19** | **33.98** | 18.93 | 26.68 | 21.44 | 0.80 | 19.60 | 4.42 | 19.50 | 11.24 | 23.94 | 18.45 | 31.80 |
| BBC | 39.28 | 39.24 | 40.33 | 39.27 | 62.27 | 39.45 | -0.03 | 56.97 | 74.28 | 40.80 | 61.50 | **82.40** | 61.79 | **89.66** |
| BBCSport | 48.10 | 47.97 | 48.11 | 47.77 | 54.46 | 47.12 | 0.34 | 54.76 | 3.47 | 42.64 | **75.59** | 63.69 | 48.10 | **92.01** |
| ProteinFold | 14.36 | 12.11 | 7.76 | 17.15 | **20.08** | 18.01 | -0.04 | 16.08 | 7.65 | 17.61 | 7.44 | 19.71 | 19.83 | **20.36** |
| PsortNeg | 13.14 | **26.75** | - | 16.85 | 16.04 | 16.93 | -0.17 | 16.09 | -0.08 | 13.13 | 1.88 | 19.76 | 13.84 | **27.44** |
| Caltech101-mit | 18.42 | 17.34 | 13.37 | 18.78 | 16.82 | 18.32 | 0.90 | 18.79 | 18.54 | 19.83 | 14.82 | 12.30 | **21.04** | **23.75** |
| Handwritten | 91.33 | 51.76 | 50.38 | 77.16 | 91.65 | 78.70 | 37.97 | 90.98 | 8.30 | 86.45 | 85.72 | 83.82 | **93.49** | **94.45** |
| Mfeat | 88.36 | 46.88 | - | 77.36 | 79.25 | 77.32 | 77.73 | 87.09 | 1.37 | 87.68 | 88.11 | 86.80 | **93.32** | **94.54** |
| Scene15 | 26.03 | 22.62 | 22.87 | 22.70 | 24.84 | 23.46 | 0.20 | 27.31 | 0.70 | 25.37 | 5.84 | **27.37** | 25.77 | **29.99** |

CAGL can be regarded as the strongest competitor in affinity graph multi-kernel clustering, our LSWMKC still exceeds CAGL with a large margins improvement of 13.34%, 16.26%, 20.41%, 8.09%, 25.00%, 9.20%, 10.00%, and 26.28% on the YALE, PsortPos, BBC, BBCSport, PsortNeg, Handwritten, Mfeat, and Scene15 datasets, respectively, in terms of ACC, which well demonstrates the superiority of our model over existing methods.

2) Compared with LKAM and LSMKKM that utilize the KNN mechanism to localize base kernel, our LSWMKC still exhibits promising performance. Especially, LSMKKM can be regarded as the most competitive method in multi-kernel clustering, the ACC of our LSWMKC exceeds that of them 7.42%, 0.43%, 11.99%, 22.66%, 20.13%, 7.08%, 2.39%, 0.97%, 0.55%, and 4.78% on ten datasets, respectively, which sufficiently illustrates the reasonableness of our model. Similarly, NMI, Purity, and ARI of our algorithm also outperform other methods on most datasets.

In summary, the quantitative comparison results can adequately substantiate the promising capability of our LSWMKC algorithm. The superiority of our algorithm can be attributed to the following two aspects: 1) our MKC model first learns a discriminative graph to explore the intrinsic local manifold structures in kernel space, which can reveal the ranking relationship of samples. The noise or outliers are sufficiently removed, which directly serves for clustering. 2) An optimal neighborhood kernel is obtained with naturally sparse property and clear block diagonal structures, which can further denoise the affinity graph. Our model achieves implicitly optimizing adaptive weights on different neighbors with corresponding samples in kernel space. Compared with the existing KNN mechanism, the unreliable distant–distance neighbors in our model can be removed or assigned small weights. The obtained localized kernel is more reasonable in comparison with the one from the KNN mechanism. Such two aspects conduce to obvious improvement in applications.

### *E. Running Time Comparison*

Fig. 2 plots the time-consuming comparison of 14 algorithms. To simplify, the elapsed time of OPLFMVC is set as the baseline and we take the logarithm of all results. As our analysis that our LSWMKC shares the same computational complexity with MKKM, LMKKM, LKAM, ONKC, SMKKM, SPMKC, CAGL, and LSMKKM, the empirical

Authorized licensed use limited to: National Univ of Defense Tech. Downloaded on February 07,2024 at 15:34:14 UTC from IEEE Xplore. Restrictions apply.

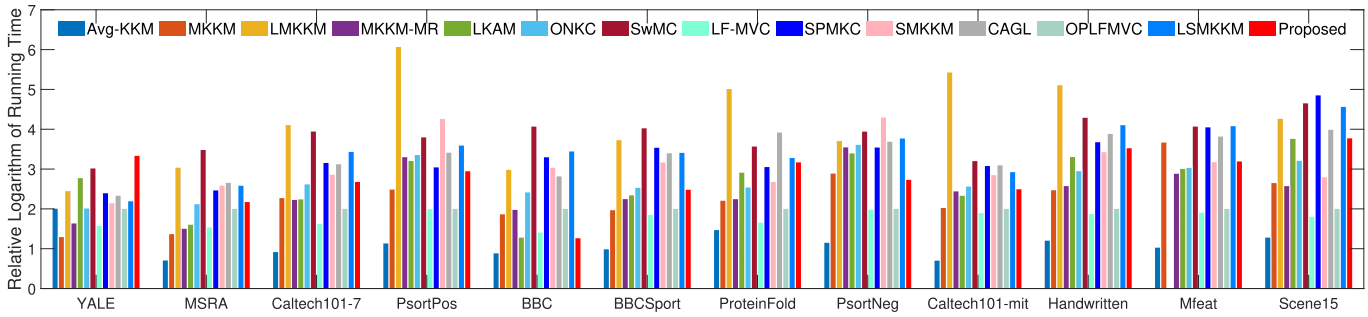



Fig. 2. Relative logarithm time-consuming comparison of 14 models on 12 datasets.

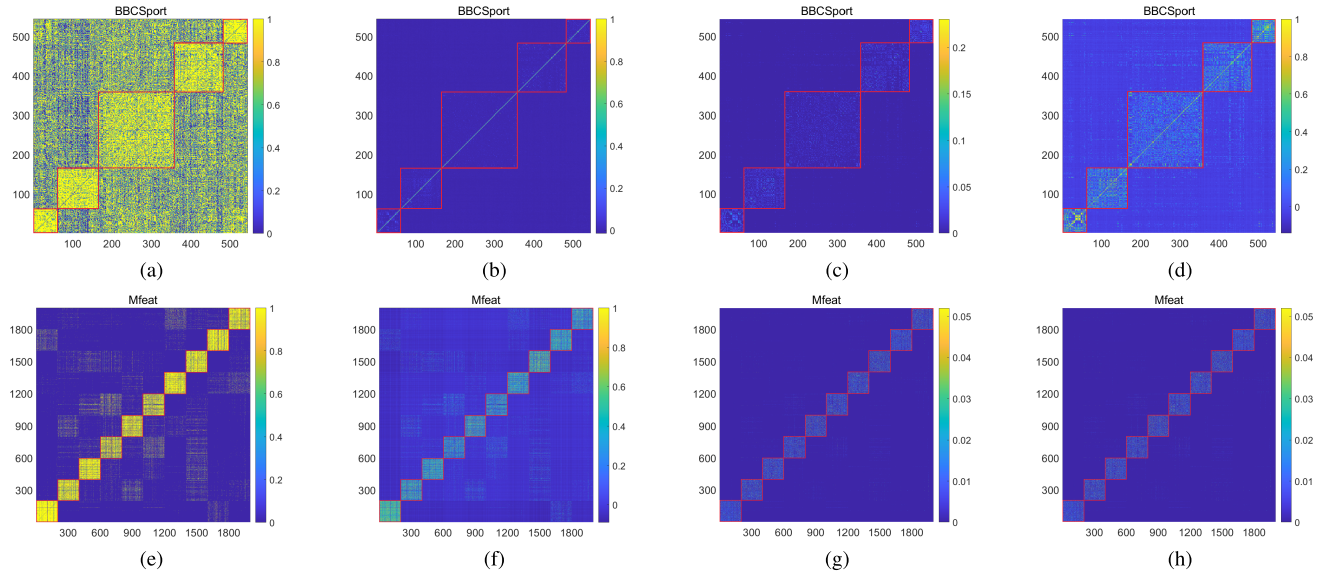



Fig. 3. Visualization of neighbor index and localized $\mathbf{K}_{(l)}$ in KNN mechanism, the affinity graph $\mathbf{Z}$, and localized $\mathbf{K}^*$ of the proposed algorithm on BBCSport and Mfeat datasets. (a) KNN (neighbor index). (b) KNN ($\mathbf{K}_{(l)}$). (c) Proposed ($\mathbf{Z}$). (d) Proposed ($\mathbf{K}^*$). (e) KNN (neighbor index). (f) KNN ($\mathbf{K}_{(l)}$). (g) Proposed ($\mathbf{Z}$). (h) Proposed ($\mathbf{K}^*$).

TABLE III

ACC, NMI, PURITY, AND ARI COMPARISONS OF OUR PROPOSED ALGORITHM AND KNN MECHANISM ON 12 BENCHMARK DATASETS

| Datasets | YALE | MSRA | Caltech101-7 | PsortPos | BBC | BBCSport | ProteinFold | PsortNeg | Caltech101-mit | Handwritten | Mfeat | Scene15 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | ACC (%) | | | | | | | | | | | |
| KNN | 63.03 | 90.48 | 74.15 | 64.14 | 71.69 | 72.06 | 36.31 | 51.73 | 37.32 | 96.75 | 96.75 | 46.82 |
| Proposed | **66.67** | **90.95** | **76.64** | **65.06** | **96.51** | **97.24** | **36.60** | **52.77** | **39.35** | **97.45** | **97.50** | **48.58** |
| | NMI (%) | | | | | | | | | | | |
| KNN | 62.00 | 83.90 | 68.78 | 35.48 | 55.66 | 48.53 | 44.22 | 28.08 | 61.74 | 92.87 | 92.88 | 42.33 |
| Proposed | **66.15** | **85.15** | **72.12** | **39.65** | **90.05** | **91.03** | **46.03** | **30.20** | **62.91** | **94.17** | **94.31** | **46.70** |
| | Purity (%) | | | | | | | | | | | |
| KNN | 63.64 | 90.48 | 78.91 | 68.39 | 73.16 | 73.16 | 42.36 | 53.88 | 39.22 | 96.75 | 96.75 | 49.63 |
| Proposed | **67.27** | **90.95** | **81.41** | **68.76** | **96.51** | **97.24** | **42.80** | **57.06** | **41.31** | **97.45** | **97.50** | **50.81** |
| | ARI (%) | | | | | | | | | | | |
| KNN | 40.19 | 79.95 | 67.50 | **34.73** | 45.11 | 42.93 | 19.44 | 24.02 | 21.35 | 92.95 | 92.94 | 28.31 |
| Proposed | **45.06** | **81.38** | **74.34** | 31.80 | **86.66** | **92.01** | **20.36** | **27.44** | **23.75** | **94.45** | **94.54** | **29.99** |

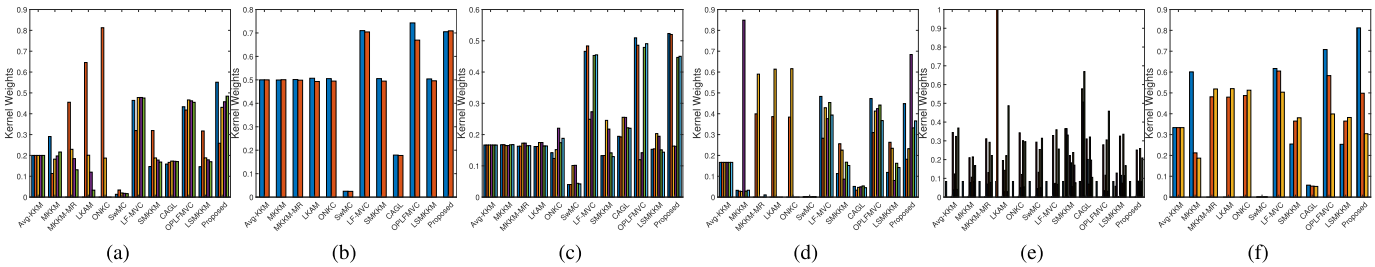
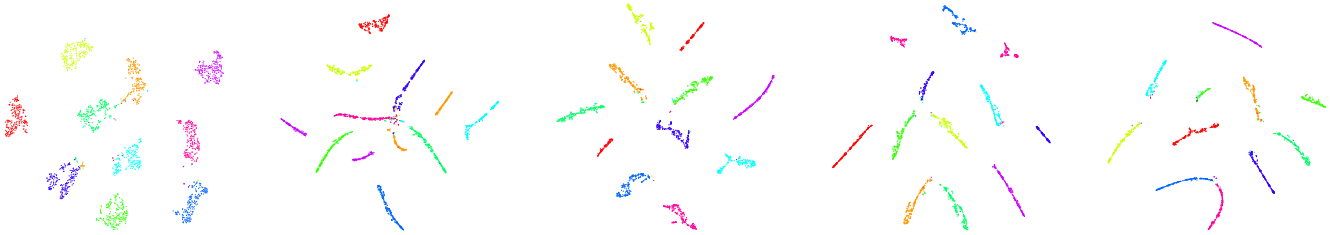


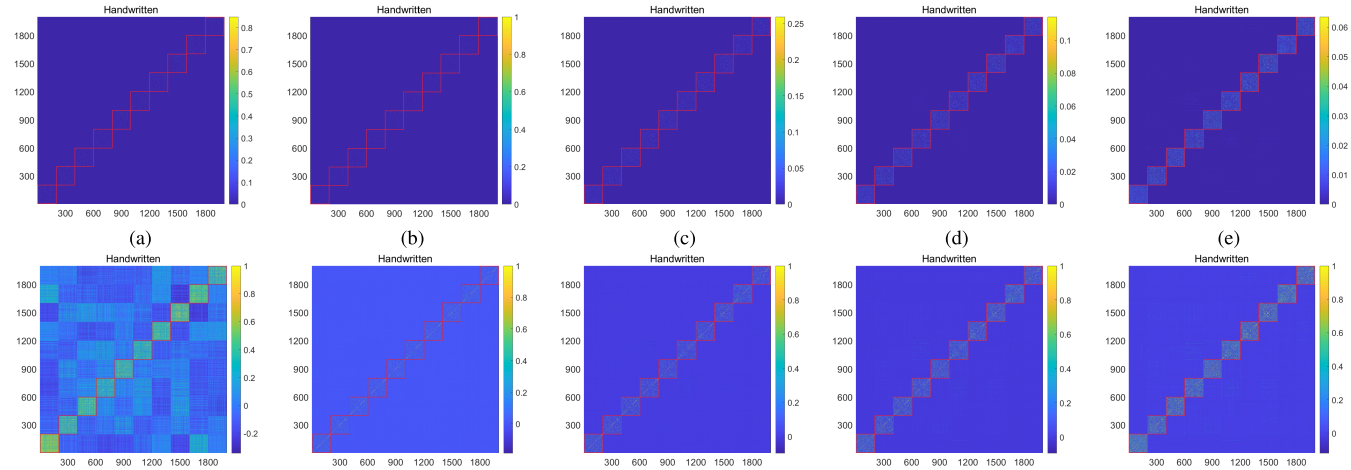

Fig. 4. Comparison of the learned kernel weights of different algorithms on six datasets. Other datasets' results are provided in the supplementary material. (a) YALE. (b) BBC. (c) BBCSport. (d) Handwritten. (e) Mfeat. (f) Scene15.

Authorized licensed use limited to: National Univ of Defense Tech. Downloaded on February 07,2024 at 15:34:14 UTC from IEEE Xplore. Restrictions apply.



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

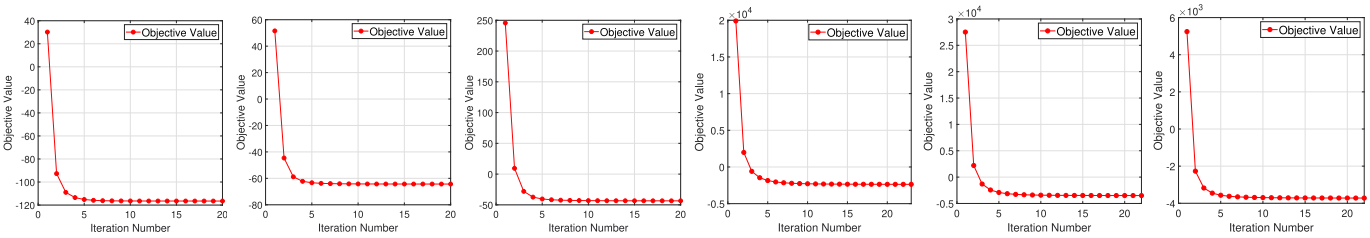
Fig. 5. Evolution of data distribution by t-SNE on Handwritten dataset. (a) Initialized. (b) First iteration. (c) Fifth iteration. (d) Tenth iteration. (e) Twentieth iteration.




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

(f) (g) (h) (i) (j)

Fig. 6. Evolution of affinity graph $\mathbf{Z}$ and neighborhood kernel $\mathbf{K}^*$ learned by our proposed algorithm on Handwritten dataset. (a) Initialized ($\mathbf{Z}$). (b) First iteration ($\mathbf{Z}$). (c) Third iteration ($\mathbf{Z}$). (d) Fifth iteration ($\mathbf{Z}$). (e) Tenth iteration ($\mathbf{Z}$). (f) Initialized ($\mathbf{K}^*$). (g) First iteration ($\mathbf{K}^*$). (h) Third iteration ($\mathbf{K}^*$). (i) Fifth iteration ($\mathbf{K}^*$). (j) Tenth iteration ($\mathbf{K}^*$).




(a) (b) (c) (d) (e) (f)

Fig. 7. Convergence of the proposed LSWMKC on six datasets. Other datasets' results are provided in the supplementary material. (a) YALE. (b) BBC. (c) BBCSport. (d) Handwritten. (e) Mfeat. (f) Scene15.

time evaluation also demonstrates that our LSWMKC costs comparative and even shorter running time. More importantly, our LSWMKC exhibits promising performance.

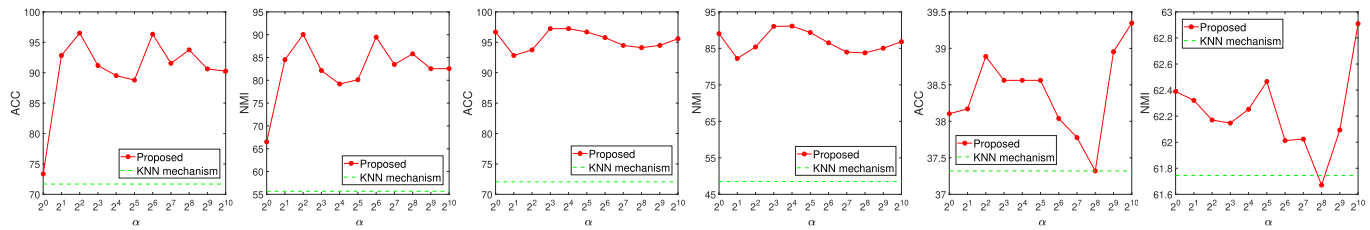
### *F. Comparing With KNN Mechanism*

Recall our motivation to learn localized kernel by considering the ranking importance of neighbors in contrast to the traditional KNN mechanism. Here, we conduct comparison experiments with the KNN mechanism (labeled as KNN). Specifically, we tune the neighbor ratio $\tau$ varying in $[0.1, 0.2, \ldots, 0.9]$ by grid search in average kernel space and report the best results. As Table III shows, our algorithm consistently outperforms the KNN mechanism. Moreover, as Fig. 3 shows, for the KNN mechanism, we plot the visualization of the neighbor index and $\mathbf{K}_{(l)}$, for our model, we visualize the learned affinity graph $\mathbf{Z}$ and neighborhood kernel $\mathbf{K}^*$ on the BBCSport and Mfeat datasets. Regarding the KNN mechanism, the neighbor index involves noticeable noise, especially on the BBCSport dataset, caused by the unreasonable neighbor-building strategy. Such coarse localized manner directly incurs the corrupted $\mathbf{K}_{(l)}$ with much noise. In contrast, the affinity graphs learned by our neighbor learning mechanism achieve more precise block structures, which directly serve for learning localized $\mathbf{K}^*$. All the above-mentioned results sufficiently illustrate the effectiveness of our neighbor-building strategy.

### *G. Kernel Weight Analysis*

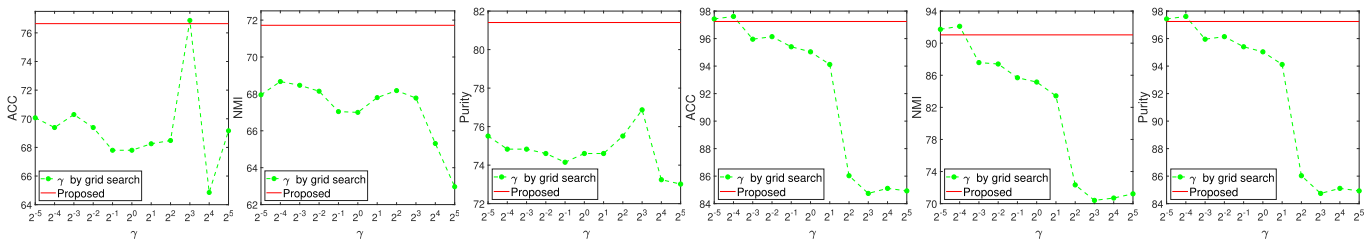
We further evaluate the distribution of the learned kernel weights on 12 datasets. As Fig. 4 shows, the kernel weight distributions of MKKM-MR, ONKC, and LKAM vary greatly and are highly sparse on most datasets. Such sparsity would incur clustering information across multiple views that cannot be fully utilized. In contrast, the weight distributions of our

Authorized licensed use limited to: National Univ of Defense Tech. Downloaded on February 07,2024 at 15:34:14 UTC from IEEE Xplore. Restrictions apply.




(a) (b) (c) (d) (e) (f)

Fig. 8. Parameter sensitivity study of hyperparameter $\alpha$ on BBC, BBCSport, and Caltech101-mit datasets. (a) BBC (ACC). (b) BBC (NMI). (c) BBCSport (ACC). (d) BBCSport (NMI). (e) Caltech101-mit (ACC). (f) Caltech101-mit (NMI).




(a) (b) (c) (d) (e) (f)

Fig. 9. Ablation study of $\gamma$ by grid search on Caltech101-7 and BBCSport datasets. Other datasets' results are provided in the supplementary material. (a) Caltech101-7 (ACC). (b) Caltech101-7 (NMI). (c) Caltech101-7 (Purity). (d) BBCSport (ACC). (e) BBCSport (NMI). (f) BBCSport (Purity).

proposed algorithm are nonsparse on all the datasets, and thus, the latent clustering information can be significantly exploited.

### *H. Visualization*

To visually demonstrate the learning process of the proposed localized building strategy, Fig. 5 plots the t-SNE visual results on the Handwritten dataset, which clearly shows the separation of different clusters during the iteration. Moreover, Fig. 6 plots the evolution of the learned affinity graph $\mathbf{Z}$ and neighborhood kernel $\mathbf{K}^*$ on the Handwritten dataset. Clearly, the noises are gradually removed and the clustering structures become clearer. Besides, $\mathbf{K}^*$ can further denoise $\mathbf{Z}$, which exhibits more evident block diagonal structures. These results can well illustrate the effectiveness of our localized strategy.

### *I. Convergence and Parameter Sensitivity*

According to our previous theoretical analysis, the convergence of our LSWMKC model has been verified with a local optimal. Here, experimental verification is further conducted to illustrate this issue. Fig. 7 reports the evolvement of optimization goals during iteration. Obviously, the objective function values monotonically decrease and quickly converge during the iteration.

We further evaluate the parameter sensitivity of $\alpha$ by grid search varying in $[2^0, 2^1, \ldots, 2^{10}]$ on the BBC, BBCSport, and Caltech101-mit datasets. From Fig. 8, we find the proposed method exhibits much better performance compared with the KNN mechanism in a wide range of $\alpha$, making it practical in real-world applications.

### *J. Ablation Study on Tuning $\gamma$ by Grid Search*

To evaluate the effectiveness of our learning $\gamma$ manner in Section III-D, we perform ablation study by tuning $\gamma$ in $[2^{-5}, 2^{-4}, \ldots, 2^5]$. The range of $\alpha$ still varies in $[2^0, 2^1, \ldots, 2^{10}]$. Fig. 9 plots the results on the Caltech101-7 and BBCSport datasets. The red line denotes our reported results. The green dashed line denotes the tuning results, for simplicity, $\alpha$ is fixed at the index of the optimal results.

As can be seen, our learning manner exceeds the tuning manner with a large margin in a wide range of $\gamma$. Although tuning manner may achieve better performance at several values of $\gamma$ , it is mainly due to tuning by grid search enlarges the search region of hyperparameter $\gamma$, it dramatically increases the running time as well. In contrast, our learning manner can significantly reduce the search region and achieve comparable or much better performance.

## V. Conclusion

This article proposes a novel localized MKC algorithm LSWMKC. In contrast to traditional localized methods in the KNN mechanism, which neglects the ranking relationship of neighbors, this article adopts a heuristic manner to implicitly optimize adaptive weights on different neighbors according to the ranking relationship. We first learn a consensus discriminative graph across multiple views in kernel space, revealing the latent local manifold structures. We further learn a neighborhood kernel with more discriminative capacity by denoising the consensus graph, which achieves naturally sparse property and clearer block diagonal property. Extensive experimental results on 12 datasets sufficiently demonstrate the superiority of our proposed algorithm over the existing 13 methods. Our algorithm provides a heuristic insight into localized methods in kernel space.

However, we should emphasize the promising performance obtained at the expense of $\mathcal{O}(n^3)$ computational complexity, which restricts applications in large-scale clustering. Introducing more advanced and efficient graph learning strategies deserve future investigation, especially for prototype or anchor

Authorized licensed use limited to: National Univ of Defense Tech. Downloaded on February 07,2024 at 15:34:14 UTC from IEEE Xplore. Restrictions apply.

learning, which may reduce the complexity from $\mathcal{O}(n^3)$ to $\mathcal{O}(n^2)$, even $\mathcal{O}(n)$. Moreover, the present work still requires postprocessing to get the final clustering labels, i.e., $k$-means. Interestingly, several concise strategies, such as rank constraint or one-pass mechanism, provide promising solutions to this issue, which deserves further research.

## Acknowledgment

The authors would like to thank the anonymous reviewers who provided constructive comments for improving the quality of this work.

## References

[1] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: A review," *ACM Comput. Surv.*, vol. 31, no. 3, pp. 264–323, Nov. 1999.

[2] R. Xu and D. C. Wunsch, "Survey of clustering algorithms," *IEEE Trans. Neural Netw.*, vol. 16, no. 3, pp. 645–678, Nov. 2005.

[3] L. Liao, K. Li, K. Li, Q. Tian, and C. Yang, "Automatic density clustering with multiple kernels for high-dimension bioinformatics data," in *Proc. IEEE Int. Conf. Bioinf. Biomed. (BIBM)*, Kansas City, MO, USA, Nov. 2017, pp. 2105–2112.

[4] L. Liao, K. Li, K. Li, C. Yang, and Q. Tian, "A multiple kernel density clustering algorithm for incomplete datasets in bioinformatics," *BMC Syst. Biol.*, vol. 12, no. S6, pp. 99–116, Nov. 2018.

[5] Y. Yang and H. Wang, "Multi-view clustering: A survey," *Bid Data Mining Anal.*, vol. 1, no. 2, pp. 83–107, Sep. 2018.

[6] N. Xiao, K. Li, X. Zhou, and K. Li, "A novel clustering algorithm based on directional propagation of cluster labels," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Budapest, Hungary, Jul. 2019, pp. 1–8.

[7] J. A. Hartigan and M. A. Wong, "Algorithm as 136: A k-means clustering algorithm," *J. Roy. Stat. Soc. C, Appl. Statist.*, vol. 28, no. 1, pp. 100–108, 1979.

[8] K. Wagstaff, C. Cardie, S. Rogers, and S. Schrödl, "Constrained k-means clustering with background knowledge," in *Proc. 28th Int. Conf. Mach. Learn. (ICML)*. Williamstown, MA, USA: Williams College, Nov. 2001, pp. 577–584.

[9] X. Peng, Y. Li, I. W. Tsang, H. Zhu, J. Lv, and J. T. Zhou, "XAI beyond classification: Interpretable neural clustering," *J. Mach. Learn. Res.*, vol. 23, pp. 6:1–6:28, Feb. 2022.

[10] B. Schölkopf, A. Smola, and K.-R. Müller, "Nonlinear component analysis as a kernel eigenvalue problem," *Neural Comput.*, vol. 10, no. 5, pp. 1299–1319, Jul. 1998.

[11] I. S. Dhillon, Y. Guan, and B. Kulis, "Kernel k-means: Spectral clustering and normalized cuts," in *Proc. 10th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, Seattle, WA, USA, Nov. 2004, pp. 551–556.

[12] B. Zhao, J. T. Kwok, and C. Zhang, "Multiple kernel clustering," in *Proc. SIAM Int. Conf. Data Mining (SDM)*, Sparks, NV, USA, Apr. 2009, pp. 638–649.

[13] 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.

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

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

[16] X. Liu *et al.*, "Optimal neighborhood kernel clustering with multiple kernels," in *Proc. 31st AAAI Conf. Artif. Intell.*, San Francisco, CA, USA, Feb. 2017, pp. 2266–2272.

[17] S. Wang *et al.*, "Multi-view clustering via late fusion alignment maximization," in *Proc. 28th Int. Joint Conf. Artif. Intell.*, Macao, China, Aug. 2019, pp. 3778–3784.

[18] S. Wang, X. Liu, L. Liu, S. Zhou, and E. Zhu, "Late fusion multiple kernel clustering with proxy graph refinement," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Oct. 14, 2021, doi: 10.1109/TNNLS.2021.3117403.

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

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

[21] L. Du *et al.*, "Robust multiple kernel k-means using L21-norm," in *Proc. 24th Int. Joint Conf. Artif. Intell. (IJCAI)*, Buenos Aires, Argentina, Sep. 2015, pp. 3476–3482.

[22] 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.

[23] X. Liu, "Incomplete multiple kernel alignment maximization for clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, Oct. 1, 2021, doi: 10.1109/TPAMI.2021.3116948.

[24] X. Liu, E. Zhu, J. Liu, T. M. Hospedales, Y. Wang, and M. Wang, "SimpleMKKM: Simple multiple kernel k-means," Sep. 2020, *arXiv:2005.04975*.

[25] X. Liu *et al.*, "One pass late fusion multi-view clustering," in *Proc. 38th Int. Conf. Mach. Learn. (ICML)*, vol. 139, Aug. 2021, pp. 6850–6859.

[26] J. B. Tenenbaum, V. de Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, Dec. 2000.

[27] 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.

[28] 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.

[29] C. Yao, J. Han, F. Nie, F. Xiao, and X. Li, "Local regression and global information-embedded dimension reduction," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 10, pp. 4882–4893, Oct. 2018.

[30] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, Dec. 2000.

[31] C. de Bodt, D. Mulders, M. Verleysen, and J. A. Lee, "Nonlinear dimensionality reduction with missing data using parametric multiple imputations," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 4, pp. 1166–1179, Apr. 2019.

[32] X. Liu, L. Wang, J. Zhang, J. Yin, and H. Liu, "Global and local structure preservation for feature selection," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 6, pp. 1083–1095, Jun. 2014.

[33] M. Sun *et al.*, "Projective multiple kernel subspace clustering," *IEEE Trans. Multimedia*, vol. 24, pp. 2567–2579, 2022.

[34] D. Zhou and C. J. C. Burges, "Spectral clustering and transductive learning with multiple views," in *Proc. 24th Int. Conf. Mach. Learn. (ICML)*, Corvallis, OR, USA, vol. 227, Nov. 2007, pp. 1159–1166.

[35] Y. Zhao, Y. Ming, X. Liu, E. Zhu, K. Zhao, and J. Yin, "Large-scale k-means clustering via variance reduction," *Neurocomputing*, vol. 307, pp. 184–194, Nov. 2018.

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

[37] X. Zhu *et al.*, "Localized incomplete multiple kernel k-means," in *Proc. 27th Int. Joint Conf. Artif. Intell. (IJCAI)*, Stockholm, Sweden, Jun. 2018, pp. 3271–3277.

[38] S. Zhou *et al.*, "Multiple kernel clustering with neighbor-kernel subspace segmentation," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 4, pp. 1351–1362, Apr. 2020.

[39] T. Zhang, X. Liu, L. Gong, S. Wang, X. Niu, and L. Shen, "Late fusion multiple kernel clustering with local kernel alignment maximization," *IEEE Trans. Multimedia*, early access, Dec. 16, 2021, doi: 10.1109/TMM.2021.3136094.

[40] X. Liu *et al.*, "Localized simple multiple kernel k-means," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Montreal, QC, Canada, Oct. 2021, pp. 9273–9281.

[41] F. Nie, X. Wang, M. I. Jordan, and H. Huang, "The constrained Laplacian rank algorithm for graph-based clustering," in *Proc. 30th AAAI Conf. Artif. Intell.*, Phoenix, AZ, USA, Sep. 2016, pp. 1969–1976.

[42] F. Nie, W. Zhu, and X. Li, "Unsupervised feature selection with structured graph optimization," in *Proc. 30th AAAI Conf. Artif. Intell.*, Phoenix, AZ, USA, Feb. 2016, pp. 1302–1308.

Authorized licensed use limited to: National Univ of Defense Tech. Downloaded on February 07,2024 at 15:34:14 UTC from IEEE Xplore. Restrictions apply.

[43] F. Nie, J. Li, and X. Li, "Parameter-free auto-weighted multiple graph learning: A framework for multiview clustering and semi-supervised classification," in *Proc. 25th Int. Joint Conf. Artif. Intell. (IJCAI)*, New York, NY, USA, Feb. 2016, pp. 1881–1887.

[44] X. Peng, J. Feng, S. Xiao, W.-Y. Yau, J. T. Zhou, and S. Yang, "Structured autoencoders for subspace clustering," *IEEE Trans. Image Process.*, vol. 27, no. 10, pp. 5076–5086, Oct. 2018.

[45] R. Zhou, X. Chang, L. Shi, Y.-D. Shen, Y. Yang, and F. Nie, "Person reidentification via multi-feature fusion with adaptive graph learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 5, pp. 1592–1601, May 2020.

[46] R. Wang, F. Nie, Z. Wang, H. Hu, and X. Li, "Parameter-free weighted multi-view projected clustering with structured graph learning," *IEEE Trans. Knowl. Data Eng.*, vol. 32, no. 10, pp. 2014–2025, Oct. 2020.

[47] F. Nie, D. Wu, R. Wang, and X. Li, "Self-weighted clustering with adaptive neighbors," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 9, pp. 3428–3441, Sep. 2020.

[48] 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.

[49] X. Li, H. Zhang, R. Wang, and F. Nie, "Multiview clustering: A scalable and parameter-free bipartite graph fusion method," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 1, pp. 330–344, Jan. 2022.

[50] Z. Ren, H. Li, C. Yang, and Q. Sun, "Multiple kernel subspace clustering with local structural graph and low-rank consensus kernel learning," *Knowl.-Based Syst.*, vol. 188, Jan. 2020, Art. no. 105040.

[51] 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.

[52] F. Nie, W. Chang, R. Wang, and X. Li, "Learning an optimal bipartite graph for subspace clustering via constrained Laplacian rank," *IEEE Trans. Cybern.*, early access, Oct. 12, 2021, doi: 10.1109/TCYB.2021.3113520.

[53] F. Nie, S. Shi, J. Li, and X. Li, "Implicit weight learning for multi-view clustering," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Nov. 1, 2021, doi: 10.1109/TNNLS.2021.3121246.

[54] S. Shi, F. Nie, R. Wang, and X. Li, "Multi-view clustering via nonnegative and orthogonal graph reconstruction," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Jul. 21, 2021, doi: 10.1109/TNNLS.2021.3093297.

[55] Z. Ren, H. Lei, Q. Sun, and C. Yang, "Simultaneous learning coefficient matrix and affinity graph for multiple kernel clustering," *Inf. Sci.*, vol. 547, pp. 289–306, Feb. 2021.

[56] Y. Liu *et al.*, "Deep graph clustering via dual correlation reduction," arXiv Preprint, 2021, *arXiv:2112.14772*.

[57] F. Nie, J. Li, and X. Li, "Self-weighted multiview clustering with multiple graphs," in *Proc. 26th Int. Joint Conf. Artif. Intell. (IJCAI)*, Melbourne, VIC, Australia, Feb. 2017, pp. 2564–2570.

[58] M. Luo, F. Nie, X. Chang, Y. Yang, A. G. Hauptmann, and Q. Zheng, "Adaptive unsupervised feature selection with structure regularization," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 4, pp. 944–956, Apr. 2018.

[59] G. Tzortzis and A. Likas, "The global kernel k-means algorithm for clustering in feature space," *IEEE Trans. Neural Netw.*, vol. 20, no. 7, pp. 1181–1194, Nov. 2009.

[60] J. Han, H. Liu, and F. Nie, "A local and global discriminative framework and optimization for balanced clustering," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 10, pp. 3059–3071, Oct. 2019.

[61] Q. Wang, Y. Dou, X. Liu, F. Xia, Q. Lv, and K. Yang, "Local kernel alignment based multi-view clustering using extreme learning machine," *Neurocomputing*, vol. 275, pp. 1099–1111, Jan. 2018.

[62] L. Du and Y.-D. Shen, "Unsupervised feature selection with adaptive structure learning," in *Proc. 21st ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Sydney, NSW, Australia, Aug. 2015, pp. 209–218.

[63] C. Cortes, M. Mohri, and A. Rostamizadeh, "Algorithms for learning kernels based on centered alignment," *J. Mach. Learn. Res.*, vol. 13, no. 1, pp. 795–828, Mar. 2012.

[64] J. Shawe-Taylor and N. Cristianini, *Kernel Methods for Pattern Analysis*. Cambridge, U.K.: Cambridge Univ. Press, May 2004.

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

[66] S. Wang *et al.*, "Fast parameter-free multi-view subspace clustering with consensus anchor guidance," *IEEE Trans. Image Process.*, vol. 31, pp. 556–568, 2022.

[67] J. Winn and N. Jojic, "LOCUS: Learning object classes with unsupervised segmentation," in *Proc. 10th IEEE Int. Conf. Comput. Vis. (ICCV)*, Beijing, China, Oct. 2005, pp. 756–763.

**Liang Li** received the bachelor's degree from the Huazhong University of Science and Technology, Wuhan, China, in 2018, and the master's degree from the National University of Defense Technology, Changsha, China, in 2020, where he is currently pursuing the Ph.D. degree.

His current research interests include multiple-view learning, multiple kernel learning, scalable clustering, and incomplete clustering.

**Siwei Wang** is currently pursuing the Ph.D. degree with the National University of Defense Technology, Changsha, China.

He has authored or coauthored and served as a reviewer for some highly regarded journals and conferences, such as IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING (TKDE), IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, IEEE TRANSACTIONS ON IMAGE PROCESSING (TIP), IEEE TRANSACTIONS ON CYBERNETICS (TYCB), IEEE TRANSACTIONS ON MULTIMEDIA (TMM), International Conference on Machine Learning (ICML), Computer Vision and Pattern Recognition (CVPR), European Conference on Computer Vision (ECCV), International Conference on Computer Vision (ICCV), AAAI Conference on Artificial Intelligence (AAAI), and International Joint Conference on Artificial Intelligence (IJCAI). His current research interests include kernel learning, unsupervised multiple-view learning, scalable clustering, and deep unsupervised learning.

**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 Full Professor with the School of Computer, NUDT. He has authored or coauthored over 80 peer-reviewed papers, including those in highly regarded journals and conferences, such as the IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE (T-PAMI), IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING (T-KDE), IEEE TRANSACTIONS ON IMAGE PROCESSING (TIP), the IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, IEEE TRANSACTIONS ON MULTIMEDIA (TMM), IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY (TIFS), International Conference on Machine Learning (ICML), NeurIPS, International Conference on Computer Vision (ICCV), Computer Vision and Pattern Recognition (CVPR), AAAI Conference on Artificial Intelligence (AAAI), and International Joint Conference on Artificial Intelligence (IJCAI). His current research interests include kernel learning and unsupervised feature learning.

Dr. Liu serves as an Associated Editor of the *Information Fusion Journal* and the IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS journal. More information can be found at https://xinwangliu.github.io.

Authorized licensed use limited to: National Univ of Defense Tech. Downloaded on February 07,2024 at 15:34:14 UTC from IEEE Xplore. Restrictions apply.

**En Zhu** received the Ph.D. degree from the National University of Defense Technology (NUDT), Changsha, China, in 2005.

He is currently a Professor with the School of Computer Science, NUDT. He has authored or coauthored more than 60 peer-reviewed papers, including the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY (TCSVT), IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS (TNNLS), *Pattern Recognition (PR)*, AAAI Conference on Artificial Intelligence (AAAI), and International Joint Conference on Artificial Intelligence (IJCAI). His current research interests include pattern recognition, image processing, machine vision, and machine learning.

Dr. Zhu was a recipient of the China National Excellence Doctoral Dissertation.

**Li Shen** received the Ph.D. degree from the National University of Defense Technology (NUDT), Changsha, China, in 2003.

He is currently a Professor with the School of Computer Science, NUDT. His current research interests include image super-resolution, machine learning, and performance optimization of machine learning systems. He has authored or coauthored 40 research papers, including the IEEE TRANSACTIONS ON COMPUTERS, IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS (TPDS), *Micro*, IEEE International Symposium on High-Performance Computer Architecture (HPCA), and Design Automation Conference (DAC).

**Kenli Li** (Senior Member, IEEE) received the Ph.D. degree in computer science from the Huazhong University of Science and Technology, Wuhan, China, in 2003.

He has authored or coauthored more than 200 research papers in international conferences and journals, such as the IEEE TRANSACTIONS ON COMPUTERS, the IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, and International Conference on Parallel Processing (ICPP). His current research interests include parallel computing, high-performance computing, and grid and cloud computing.

Dr. Li serves on the Editorial Board of the IEEE TRANSACTIONS ON COMPUTERS.

**Keqin Li** (Fellow, IEEE) is currently a SUNY Distinguished Professor of computer science with the State University of New York, New Paltz, NY, USA. He is also a National Distinguished Professor with Hunan University, Changsha, China. He has authored or coauthored over 830 journal articles, book chapters, and refereed conference papers. He holds over 60 patents announced or authorized by the Chinese National Intellectual Property Administration. His current research interests include cloud computing, fog computing, mobile edge computing, energy-efficient computing and communications, embedded systems, cyber–physical systems, heterogeneous computing systems, big data computing, high-performance computing, CPU–GPU hybrid and cooperative computing, computer architectures and systems, computer networking, machine learning, and intelligent and soft computing.

Dr. Li received several best paper awards. He was the chair of many international conferences. He is currently an Associate Editor of the *ACM Computing Surveys* and the *CCF Transactions on High-Performance Computing*. He has served on the Editorial Board of the IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, the IEEE TRANSACTIONS ON COMPUTERS, the IEEE TRANSACTIONS ON CLOUD COMPUTING, the IEEE TRANSACTIONS ON SERVICES COMPUTING, and the IEEE TRANSACTIONS ON SUSTAINABLE COMPUTING. He is among the world's top 5 most influential scientists in parallel and distributed computing based on a composite indicator of the Scopus citation database.

Authorized licensed use limited to: National Univ of Defense Tech. Downloaded on February 07,2024 at 15:34:14 UTC from IEEE Xplore. Restrictions apply.

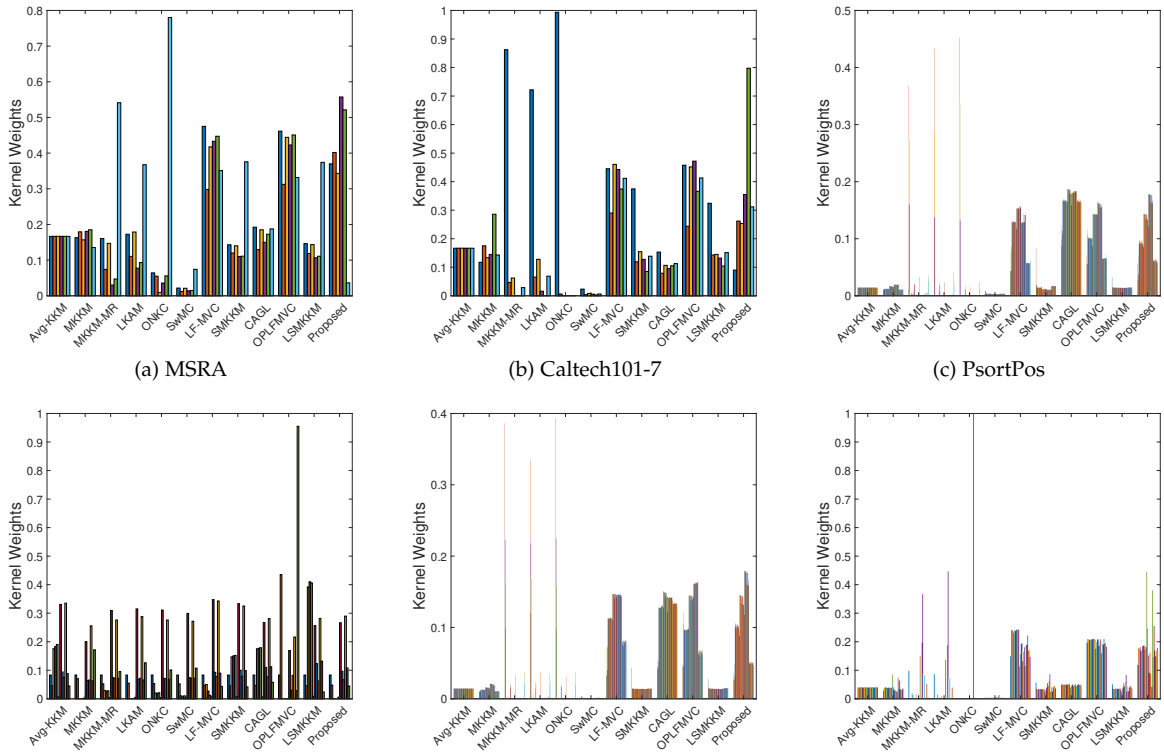
# Appendix of "Local Sample-weighted Multiple Kernel Clustering with Consensus Discriminative Graph"

Liang Li, Siwei Wang, Xinwang Liu, En Zhu, Li Shen, Kenli Li, and Keqin Li

## 1 Experiment

### 1.1 Kernel Weight Analysis

Figure. 1 shows the kernel weight distribution of different algorithms on MSRA, Caltech101-7, PsortPos, ProteinFold, PsortNeg, and Caltech101-mit datasets.




(a) MSRA (b) Caltech101-7 (c) PsortPos

(d) ProteinFold (e) PsortNeg (f) Caltech101-mit

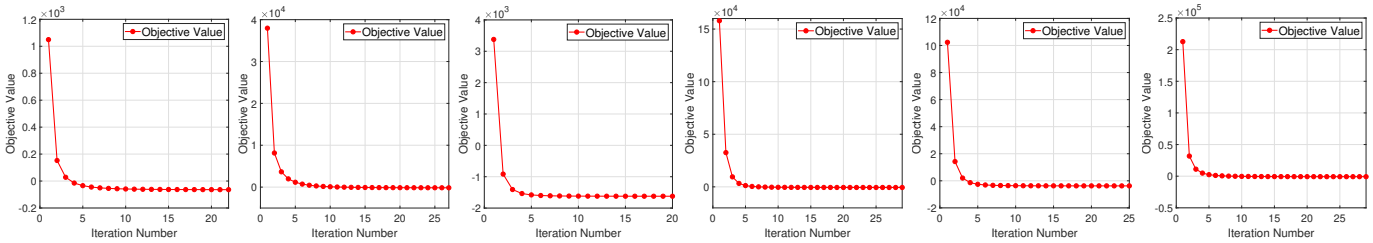
Fig. 1: Comparison of the learned kernel weights of different algorithms on six datasets.


- *L. Li, S. Wang, X. Liu, E. Zhu, and L. Shen are with School of Computer, National University of Defense Technology, Changsha 410073, China (E-mail: {liangli, wangsiwei13, xinwangliu, enzhu, lishen}@nudt.edu.cn).*
- *K. Li is with College of Computer Science and Electronic Engineering, Hunan University, and also with Supercomputing and Cloud Computing Institute, Changsha 410073, China (E-mail: lkl@hnu.edu.cn).*
- *K. Li is with the Department of Computer Science, State University of New York, New Paltz, New York 12561, USA (E-mail: lik@newpaltz.edu).*


*Manuscript received December 19, 2021.*

### 1.2 Convergence and Parameter Sensitivity

Figure. 2 shows the convergence of our LSWMKC algorithm on MSRA, Caltech101-7, PsortPos, ProteinFold, PsortNeg, and Caltech10-mit datasets.

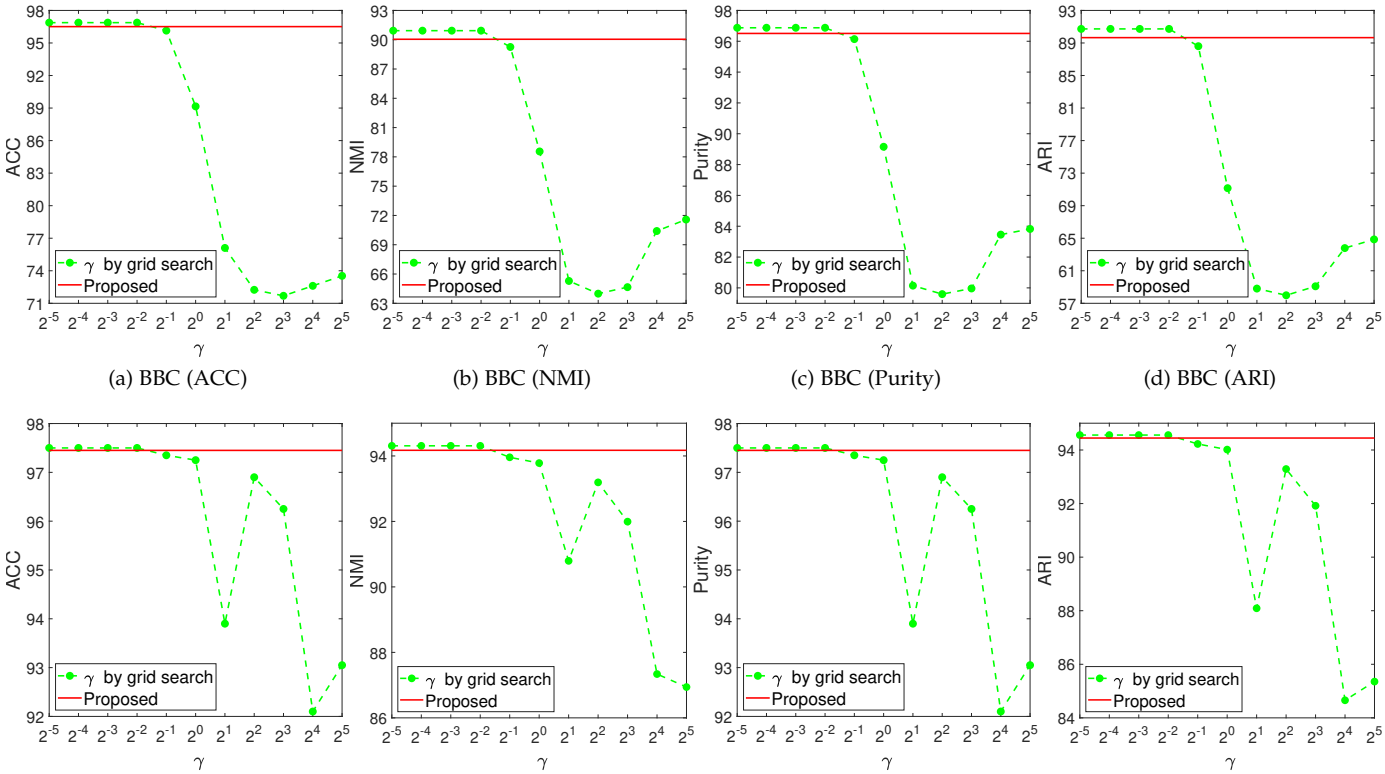



(a) MSRA (b) Caltech101-7 (c) PsortPos (d) ProteinFold (e) PsortNeg (f) Caltech10-mit

Fig. 2: The convergence of the proposed LSWMKC on six datasets.

### 1.3 Ablation Study on $\gamma$ by Grid Search

Figure. 3 shows the ablation study of $\gamma$ by grid search on BBC and Handwritten datasets.




(a) BBC (ACC) (b) BBC (NMI) (c) BBC (Purity) (d) BBC (ARI)

(e) Handwritten (ACC) (f) Handwritten (NMI) (g) Handwritten (Purity) (h) Handwritten (ARI)

Fig. 3: Ablation study on $\gamma$ by grid search on BBC and Handwritten datasets.