Simple Multiple Kernel k-Means with Heat Kernel Diffusion

Zhiwei Xia[1], Xiaohong Jia[1]*, Xuejun Zhang[1], Yao Zhao[2], Wenqian Yu[3]
[1]School of Electronic and Information Engineering, Lanzhou Jiaotong University, Lanzhou 730070, China
[2]Institute of Information Science, Beijing Jiaotong University, Beijing 100044, China
[3]Lanzhou No. 5 High school, Lanzhou 730000, China
xiazhiwei2024@163.com, jiaxhm@lzjtu.edu.cn, xuejunzhang@mail.lzjtu.cn, yzhao@bjtu.edu.cn, m13919406042@163.com

***Abstract*—Multiple kernel clustering (MKC) aims to capture complex nonlinear structures by integrating multiple predefined kernels. Simple multiple kernel k-means (SMKKM) jointly learns kernel weights and clustering assignments within a min–max optimization framework, achieving efficient and competitive performance. However, its direct linear fusion of original kernel matrices fails to exploit the intrinsic geometric structure of data, making it sensitive to noise and unreliable similarities. To address this issue, we propose SMKKM with heat kernel diffusion (SMKKM-HK). For each base kernel, a kernel-induced graph Laplacian is constructed, and a heat kernel operator is applied to perform bilateral spectral smoothing on kernel matrices, enhancing geometric consistency and suppressing high-frequency noise. Furthermore, a dynamic graph update mechanism guided by current clustering assignments is introduced, enabling the diffused kernels to adapt to evolving cluster structures. Extensive experiments demonstrate that SMKKM-HK achieves improved clustering performance and robustness compared with state-of-the-art MKC methods.**

***Index Terms*—Multiple kernel clustering, multiple kernel k-means, heat kernel diffusion, graph Laplacian**

I. Introduction

Clustering [1] is a fundamental task in unsupervised learning that aims to partition data into semantically consistent clusters based on sample similarity. It has been widely applied in image classification [2], image segmentation [3], and anomaly detection [4]. Traditional k-means [5] is simple and computationally efficient, but its reliance on Euclidean distance and linear separability assumptions limits its ability to model high-dimensional and nonlinear data structures. Kernel methods [6] map data into a reproducing kernel Hilbert space, leading to kernel k-means (KKM) [7]. However, its performance is highly dependent on the choice of a single kernel function. To address this issue, multiple kernel clustering (MKC) [8] enhances representation flexibility by integrating multiple predefined base kernels, among which multiple kernel k-means (MKKM) [9], as a representative method, has attracted considerable attention.

Nevertheless, classical MKKM still suffers from limitations in modeling sample heterogeneity and complex kernel structures, motivating extensive studies on kernel weight modeling and optimal kernel construction strategies. For example, localized MKKM (LMKKM) [10] introduces sample-wise kernel weights to better capture local structural variations, while MKKM with matrix-induced regularization (MKKM-MR) [11] suppresses redundant correlations among kernels through matrix-induced regularization. Building upon these studies, optimal neighborhood kernel clustering with multiple kernels (ONKC) [12] extends the search space of the optimal kernel beyond linear kernel combinations. Furthermore, optimal neighborhood MKC with adaptive local kernels (ON-ALK) [13] alleviates the rigidity caused by fixed neighborhood assumptions through adaptive local neighborhood selection.

In recent years, Liu *et al*. [14] proposed Simple MKKM (SMKKM), which significantly simplifies the optimization process of traditional MKKM by introducing a kernel alignment criterion and a min–max optimization framework. Based on the SMKKM framework, Li *et al*. [15] proposed SMKKM with kernel weight regularization (SMKKM-KWR) to alleviate excessive sparsity of kernel weights, while regularized SMKKM with kernel average alignment (R-SMKKM-KAA) [16] enhances structural constraints by incorporating clustering partition regularization. To address the limitation of global kernel alignment in ignoring sample-wise differences, Liu *et al*. [17] proposed localized SMKKM (LSMKKM), and further extended it to sample adaptive localized SMKKM (SAL-SMKKM) [18].

Meanwhile, graph learning and spectral methods have been introduced into multi-kernel clustering [19], [20] to model local geometric structures. For example, local sample-weighted MKC with consensus discriminative graph (LSWMKC) [21] implicitly optimizes neighborhood relationships by learning a consensus discriminative graph, while smooth MKKM via underlying graph filtering (SMKKM-UGF) [22] mitigates the influence of noise in the kernel space through an underlying graph filtering mechanism. However, most of these structure-enhanced methods are built upon the traditional MKKM framework and often involve relatively complex optimization procedures, making it difficult to achieve a balance between optimization simplicity and numerical stability. In contrast, although SMKKM is computationally efficient, its static kernel structure and global kernel alignment are insufficient to suppress local noise propagation, and the clustering partition is treated

*Corresponding author: Xiaohong Jia (jiaxhm@lzjtu.edu.cn)

solely as the final output without being exploited to refine kernel representations.

Motivated by the above analysis, we propose SMKKM-HK. The proposed method constructs kernel-induced graph Laplacians for each base kernel and introduces a heat kernel diffusion mechanism to smooth and enhance the local geometric structure of the kernel space. This approach preserves the optimization simplicity and stability of SMKKM while effectively suppressing noise propagation and improving the global consistency of kernel representations. The main contributions of this work are summarized as follows:

- A diffusion kernel construction method based on heat kernels, which builds a graph Laplacian for each base kernel and generates structure-enhanced diffusion kernels through bilateral heat kernel diffusion.
- An iterative update strategy is developed to adaptively refine the diffused kernels according to the structural information induced by the current clustering assignments.
- The proposed diffused kernels are integrated into the min–max optimization framework of SMKKM, enabling joint learning of diffused kernel representations, kernel weights, and clustering assignments via an alternating optimization scheme.

The remainder of this paper is organized as follows. Section II describes the proposed SMKKM-HK method and its optimization procedure. Section III reports experimental comparisons with several representative methods, followed by analyses. Section IV concludes the paper.

II. Methodology

In this section, we systematically present the proposed SMKKM-HK framework. We first formulate and analyze the overall objective function of the algorithm, and then derive the corresponding optimization problem and solution procedure.

A. Kernel-induced Graph Construction and Diffusion Enhancement

In multiple kernel clustering, different base kernels characterize data similarity from complementary perspectives. To fully exploit the local geometric structure in each kernel space and to construct graphs with good connectivity and robustness, we first build an adjacency graph independently for each base kernel $K_p (p = 1, \cdots, m)$. Specifically, for the i-th sample, the graph weight vector $s_i^{(p)}$ is obtained by solving an adaptive optimization problem [23]:

$$\min_{s_i^{(p)} \geq 0, \quad s_i^{(p)\top}\mathbf{1}=1} \quad \sum_{j=1}^{n} \left(d_{ij}^{(p)} s_{ij}^{(p)} + \beta \left\| s_i^{(p)} \right\|_2^2 \right), \tag{1}$$

where $d_{ij}^{(p)} = K_{ii}^{(p)} + K_{jj}^{(p)} - 2K_{ij}^{(p)}$ denotes the distance between samples x_i and x_j in the p-th kernel space; $s_i^{(p)} = \left[s_{i1}^{(p)}, \cdots, s_{in}^{(p)} \right]^T$ represents the graph weight vector associated with the i-th sample under the p-th kernel; and $\beta > 0$ is a balancing parameter.

To obtain a sparse and robust graph structure, a k-nearest neighbor (KNN) constraint is further imposed, such that each sample is connected only to its KNN in terms of the kernel-induced distance. In this work, k_{nn} is empirically set to 10 or 16, which yields stable performance in our experiments. This constraint guarantees local connectivity while alleviating, to some extent, the need for explicit tuning of the parameter β. Let $\mathcal{N}_k(i)$ denote the set of KNN of sample i. The above problem admits the following closed-form solution:

$$s_{ij}^{(p)} = \begin{cases} \dfrac{d_{i,(k_{\text{nn}}+1)}^{(p)} - d_{ij}^{(p)}}{k d_{i,(k_{\text{nn}}+1)}^{(p)} - \sum_{l=1}^{k} d_{i,l}^{(p)}} & j \in \mathcal{N}_k(i) \\ 0 & \text{otherwise} \end{cases}, \tag{2}$$

where $d_{i,(k_{\text{nn}}+1)}^{(p)}$ denotes the distance sample x_i to its $(k_{\text{nn}} + 1)$-th nearest neighbor, serving as an adaptive local scaling parameter, and $d_{i,l}^{(p)}$ represents the distance from x_i to its l-th nearest neighbor (sorted in ascending order). This weighting scheme adaptively characterizes local neighborhood structures while naturally satisfying the non-negativity and normalization constraints.

After obtaining the adjacency matrix S_p, the corresponding degree matrix is defined as $(D_p)_{ii} = \sum_j (S_p)_{ij}$. The symmetrically normalized graph Laplacian is then constructed as [24]:

$$L_p = I - D_p^{-1/2} S_p D_p^{-1/2}. \tag{3}$$

Although the resulting graph effectively captures local neighborhood relationships among samples, it relies primarily on direct adjacency information and thus fails to adequately model higher-order structural relationships and global manifold characteristics. To address this limitation, we further introduce a heat kernel diffusion operator on each kernel-induced graph [25], defined as:

$$G_p = \exp(-\eta L_p), \tag{4}$$

where $\eta > 0$ is the diffusion time parameter controlling the diffusion strength. This operator can be interpreted as the analytical solution of the heat diffusion equation on graphs. Through its Taylor series expansion, it can be shown that G_p is equivalent to a weighted summation of graph paths of different orders, that is $G_p = \sum_{k=0}^{\infty} \frac{(-\eta L_p)^k}{k!}$. As a result, heat kernel diffusion effectively integrates multi-hop neighborhood information, yielding smoother and structurally enhanced kernel representations while preserving local geometric properties.

B. Bilaterally Diffused Kernel Alignment Objective

In the previous subsection, we constructed structure-enhanced diffusion operators G_p for each base kernel via heat kernel diffusion. Building on this, we formalize the generation of the diffused kernels $\widetilde{K}_p = G_p K_p G_p$ as bilateral heat kernel diffusion, which symmetrically smooths the original kernel matrices and propagates structural information bidirectionally along the graph. This operation effectively integrates both local neighborhood relations and global structural information into the kernel representation. Accordingly, the diffusion-kernel-based multi-kernel k-means clustering can be formalized as the following optimization problem:

$$\min_{\gamma\in\Delta}\ \max_{H\in\Gamma} \mathrm{Tr}\left(\sum_{p=1}^{m}\gamma_p^2\left(G_pK_pG_p\right)HH^\top\right), \tag{5}$$

where $\Delta = \left\{\gamma \in \mathbb{R}^m \,\middle|\, \sum_{p=1}^m \gamma_p = 1,\ \gamma_p \geq 0,\ \forall p\right\}$ and $\Gamma = \{H \in \mathbb{R}^{n\times c} \mid H^\top H = I_c\}$. $K_p \in \mathbb{R}^{n\times n}$ denotes the p-th base kernel matrix, and γ_p represents its corresponding kernel weight. H is the clustering partition matrix, n is the number of samples, and c is the number of clusters.

During the initialization stage of the algorithm, to prevent overly strong diffusion from prematurely smoothing local structures, a relatively weak diffusion strength is adopted by setting $G_p = \exp\left(-\frac{\eta}{2}L_p\right)$. As the optimization proceeds, the full heat kernel diffusion from $G_p = \exp(-\eta L_p)$ is subsequently employed, thereby gradually strengthening the influence of structural information in the kernel fusion and clustering process.

C. Optimization

To solve the non-convex optimization problem defined in Eq. (5), we employ an alternating optimization algorithm.

*1) **Update** $\mathbf{S}$:* The initial graph $S^{(1)}$ is constructed from kernel-induced distances. To enable dynamic updates of the diffusion kernels, we introduce a structure-adaptive graph update mechanism based on the current partition matrix H. At the t-th iteration, the pairwise distances are computed as the Euclidean distances in the embedding space, and the local similarity graph $\widetilde{S}^{(t+1)}$ is reconstructed using the same k-nearest neighbor strategy as in the initialization. A smoothing update is applied to ensure the stability of the optimization process.

$$S^{(t+1)} = (1-\alpha)S^{(t)} + \alpha\widetilde{S}^{(t+1)}, \tag{6}$$

where α is the smoothing parameter for graph updates. Based on the updated similarity matrix $S^{(t+1)}$, the corresponding normalized graph Laplacian is constructed, from which the heat kernel diffusion operator is computed, enabling the dynamic update of the diffusion kernel representation.

*2) **Update** $\mathbf{H}$:* With γ fixed, the subproblem of optimizing H in Eq. (5) can be expressed as follows:

$$\max_{H\in\Gamma} \mathrm{Tr}\left(\sum_{p=1}^m \gamma_p^2 G_pK_pG_p\, HH^\top\right). \tag{7}$$

By performing a weighted fusion of the diffusion kernels as $K_\gamma = \sum_{p=1}^m \gamma_p^2(G_pK_pG_p)$ and applying eigenvalue decomposition, the above problem can be converted into a standard spectral optimization form, whose optimal solution is obtained from the eigen-decomposition of K_γ.

*3) **Update** $\boldsymbol{\gamma}$:* With H fixed, the subproblem of optimizing γ in Eq. (5) can be equivalently formulated as the following constrained minimization problem:

$$\min_{\gamma\in\Delta}\ \mathcal{G}(\boldsymbol{\gamma}), \tag{8}$$

Algorithm 1 SMKKM-HK

Input: $\{K_p\}_{p=1}^m$, k_{nn}, c, η, α, $t = 1$.

Initialize: $\gamma^{(1)} = 1/m$, $S^{(1)}$, flag = 1.
while flag **do**
  Update graph structure $S^{(t)}$ by Eq. (6) and construct the diffused kernels $\widetilde{K}_p = G_pK_pG_p$.
  Update partition matrix $H^{(t)}$ using kernel k-means with $K_{\gamma^{(t)}} = \sum_{p=1}^m (\gamma_p^{(t)})^2\widetilde{K}_p$.
  Compute gradient by Eq. (10) and descent direction $d^{(t)}$ by Eq. (13).
  Update kernel weights $\gamma^{(t+1)} \leftarrow \gamma^{(t)} + \alpha d^{(t)}$.
  if $\max|\gamma^{(t+1)} - \gamma^{(t)}| \leq e^{-4}$ **then**
    flag = 0.
  end if
  $t \leftarrow t+1$.
end while

with

$$\mathcal{G}(\boldsymbol{\gamma}) = \max_{H\in\Gamma} \mathrm{Tr}\left(\sum_{p=1}^m \gamma_p^2 G_pK_pG_p\, HH^\top\right). \tag{9}$$

The gradient of $\mathcal{G}(\boldsymbol{\gamma})$ with respect to γ_p is given by

$$\frac{\partial\mathcal{G}(\boldsymbol{\gamma})}{\partial\gamma_p} = 2\gamma_p\,\mathrm{Tr}\left(H^\top G_pK_pG_pH\right). \tag{10}$$

To enforce the non-negativity and normalization constraints of the kernel weights, we optimize the kernel weight vector γ using the reduced gradient method [26]. Let μ denote the index of the largest component of γ; the q-th $(1 \leq q \leq m)$ component of the reduced gradient $\nabla\mathcal{G}(\gamma)$ is defined as:

$$[\nabla\mathcal{G}(\boldsymbol{\gamma})]_q = \frac{\partial\mathcal{G}(\boldsymbol{\gamma})}{\partial\gamma_q} - \frac{\partial\mathcal{G}(\boldsymbol{\gamma})}{\partial\gamma_\mu} \quad \forall q \neq \mu, \tag{11}$$

and

$$[\nabla\mathcal{G}(\boldsymbol{\gamma})]_\mu = \sum_{q=1,q\neq\mu}^{m}\left(\frac{\partial\mathcal{G}(\boldsymbol{\gamma})}{\partial\gamma_\mu} - \frac{\partial\mathcal{G}(\boldsymbol{\gamma})}{\partial\gamma_q}\right). \tag{12}$$

To improve numerical stability, $-\nabla\mathcal{G}(\gamma)$ is used as the descent direction within the reduced gradient framework. However, directly updating along this direction may violate the non-negativity constraints, particularly when $\gamma_q = 0$ and $[\nabla\mathcal{G}(\gamma)]_q > 0$. To ensure feasibility, the descent direction for such components is set to zero. Taking these constraints into account, γ is updated along the following corrected descent direction:

$$d_q = \begin{cases} 0 & \text{if } \gamma_q = 0 \text{ and } [\nabla\mathcal{G}(\boldsymbol{\gamma})]_q > 0 \\ -[\nabla\mathcal{G}(\boldsymbol{\gamma})]_q & \text{if } \gamma_q > 0 \text{ and } q \neq \mu \\ -[\nabla\mathcal{G}(\boldsymbol{\gamma})]_\mu & \text{if } q = \mu \end{cases}, \tag{13}$$

where $d = [d_1, \ldots, d_m]^\top$ denotes the descent direction, and γ is updated via $\gamma \leftarrow \gamma + \alpha d$, where the step size α is determined by a line search, such as the Armijo's rule. The complete procedure of the proposed SMKKM-HK algorithm is summarized in Algorithm 1.

III. Experiment

This section systematically evaluates the effectiveness and robustness of SMKKM-HK in terms of clustering performance, structural consistency, and stability through comprehensive comparative experiments on multiple datasets and evaluation metrics.

A. Experimental Setup

To comprehensively evaluate the performance of the proposed algorithm under different data scales and structural complexities, experiments were conducted on 10 public benchmark datasets, including Yale, BBCSport, Pima, Flower17, COIL_20, SensITVehicle, Caltech101, UCI_DIGIT, CCV, and Flower102. The datasets span multiple domains and differ markedly in scale, kernel number, and class number. Detailed information about these datasets is summarized in table I.

TABLE I
Dataset Information In Our Experiment

Dataset	Samples	Kernels	Clusters
Yale	165	5	15
BBCSport	544	2	5
Pima	768	8	2
Flower17	1360	7	17
COIL_20	1440	3	20
SensITVehicle	1500	2	3
Caltech101	1530	25	102
UCI_DIGIT	2000	3	10
CCV	6773	3	20
Flower102	8189	4	102

To ensure fairness and reproducibility, 11 representative multiple kernel clustering methods were selected as baselines, including KKM, MKKM, MKKM-MR [11], ONKC [12], LSMKKM [17], ON-ALK [13], SMKKM [14], R-SMKKM-KAA [16], LSWMKC [21], SAL-SMKKM [18], and SMKKM-UGF [22]. All baseline methods were implemented strictly following the parameter settings recommended in their original publications. For the proposed SMKKM-HK algorithm, the heat kernel diffusion parameter η was searched to $\left[2^{-2}, 2^{-1}, \cdots, 2^{4}\right]$, and the graph smoothing parameter α was varied within [0,1] with a step size of 0.1. All hyperparameters were selected via grid search based on the best performance on the validation set. Considering the sensitivity of k-means to the initialization of cluster centers, all experiments were independently repeated 20 times, and the average results were reported. All experiments were conducted in MATLAB R2024a.

Clustering performance was evaluated using four widely adopted metrics: Accuracy (ACC), Normalized Mutual Information (NMI), Purity, Adjusted Rand Index (ARI), and F-score.

B. Experimental Results and Analysis

As shown in table II, the performance comparison between the proposed method and several representative multiple kernel clustering algorithms on 10 benchmark datasets is reported, where the best result for each dataset is highlighted in bold. Based on these experimental results, the clustering performance of different algorithms is analyzed from multiple perspectives:

- The experimental results demonstrate that the proposed SMKKM-HK achieves the best clustering performance across all 10 benchmark datasets. It consistently outperforms competing methods in terms of ACC, NMI, Purity, and ARI, indicating strong stability and robustness under diverse data distributions.
- To validate the effectiveness of the proposed structural enhancement, we further compare SMKKM-HK with its baseline SMKKM. The consistent performance improvements across different datasets confirm that the incorporation of kernel-induced structural information and diffusion-based smoothing significantly enhances clustering stability and discriminative capability.
- In addition, we evaluate the scalability and robustness of SMKKM-HK on datasets with larger sample sizes and more complex structures. The significant ACC improvements observed on Caltech101, UCI_DIGIT, CCV, and Flower102 indicate that SMKKM-HK is particularly well suited for complex data scenarios and large-scale multiple kernel fusion tasks.

In summary, SMKKM-HK exhibits more stable and reliable clustering behavior in complex data scenarios, highlighting its potential advantages in multiple kernel fusion and noisy environments.

C. Kernel Weight Coefficients Analysis

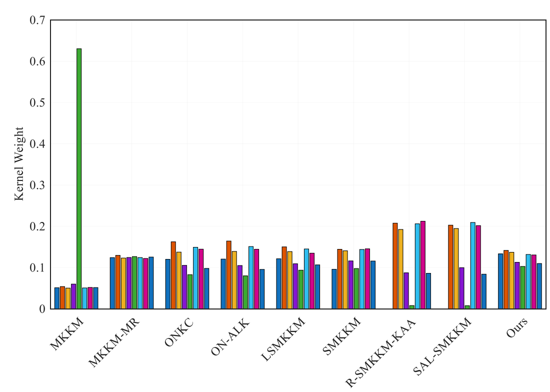


Fig. 1. Kernel weights of various algorithms on the benchmark Pima dataset.

As illustrated in Fig. 1, traditional MKKM often concentrates the majority of kernel weights on a small subset of kernels, which may amplify the influence of noisy or unreliable kernels. In contrast, SMKKM-HK achieves a more balanced weight allocation across all kernels, allowing it to better capture complementary information and reflect the relative importance of different kernels in the clustering process.

D. Parameter Analysis

To analyze the sensitivity of the algorithm to hyperparameters, we evaluated the effects of α and η on multiple benchmark datasets. As shown in Fig. 2, SMKKM-HK maintains high ACC over a wide range of parameter values, indicating that the method is relatively insensitive to parameter settings and exhibits good stability and practical usability.

TABLE II
CLUSTERING PERFORMANCE COMPARISON OF DIFFERENT METHODS ON VARIOUS BENCHMARK DATASETS.

Datasets	Metric	KKM	MKKM	MKKM-MR	ONKC	LSMKKM	ON-ALK	SMKKM	R-SMKKM-KAA	LSWMKC	SAL-SMKKM	SMKKM-UGF	Ours
Yale	ACC	54.73 ± 2.34	52.00 ± 2.06	56.22 ± 2.62	56.98 ± 2.86	58.94 ± 2.28	57.12 ± 3.03	55.67 ± 2.62	56.39 ± 2.40	61.79 ± 2.37	60.33 ± 3.16	63.00 ± 1.98	**63.70 ± 1.79**
	NMI	57.32 ± 1.66	54.35 ± 1.70	58.58 ± 1.80	56.99 ± 1.83	60.46 ± 1.69	59.11 ± 1.70	58.60 ± 1.62	59.08 ± 1.36	63.15 ± 1.44	60.59 ± 2.37	62.90 ± 0.52	**63.46 ± 0.60**
	Purity	55.42 ± 2.20	52.94 ± 1.85	56.69 ± 2.47	57.53 ± 2.71	59.45 ± 2.25	57.73 ± 2.85	56.00 ± 2.60	56.76 ± 2.21	62.55 ± 2.02	60.88 ± 3.09	64.33 ± 1.41	**64.39 ± 1.56**
	ARI	33.93 ± 2.55	30.42 ± 2.19	35.40 ± 2.60	37.10 ± 2.55	38.10 ± 2.31	35.79 ± 2.72	35.57 ± 2.41	36.30 ± 2.01	41.41 ± 2.00	38.51 ± 3.38	41.91 ± 0.89	**42.20 ± 0.90**
BBCSport	ACC	63.17 ± 1.36	63.03 ± 1.48	63.17 ± 1.36	63.29 ± 1.47	73.13 ± 0.12	70.24 ± 0.13	62.79 ± 1.30	64.20 ± 1.57	86.06 ± 2.39	72.61 ± 0.00	94.02 ± 3.13	**98.16 ± 0.00**
	NMI	43.50 ± 1.10	43.58 ± 1.24	43.50 ± 1.10	43.56 ± 1.18	64.03 ± 0.15	55.79 ± 0.15	43.20 ± 1.16	44.45 ± 0.98	76.79 ± 0.54	64.92 ± 0.00	86.07 ± 3.08	**93.67 ± 0.00**
	Purity	68.06 ± 0.71	68.15 ± 0.84	68.06 ± 0.71	68.14 ± 0.72	78.69 ± 0.04	75.31 ± 0.29	67.97 ± 0.71	68.68 ± 0.87	86.51 ± 0.37	79.96 ± 0.00	94.02 ± 3.13	**98.16 ± 0.00**
	ARI	39.28 ± 1.86	39.24 ± 2.03	39.28 ± 1.68	39.42 ± 1.98	60.76 ± 0.06	53.55 ± 0.37	38.79 ± 1.88	40.80 ± 1.88	70.47 ± 0.30	60.63 ± 0.00	87.94 ± 3.94	**94.97 ± 0.00**
Pima	ACC	50.65 ± 0.00	51.30 ± 0.00	65.90 ± 0.00	65.92 ± 0.00	67.11 ± 0.16	68.36 ± 0.00	67.19 ± 0.00	66.71 ± 0.52	65.62 ± 0.00	69.01 ± 0.00	65.10 ± 0.00	**70.23 ± 0.00**
	NMI	0.01 ± 0.00	0.07 ± 0.00	7.83 ± 0.00	7.85 ± 0.00	9.24 ± 0.13	7.56 ± 0.00	9.03 ± 0.00	8.55 ± 0.58	10.10 ± 0.00	10.34 ± 0.00	0.21 ± 0.00	**11.06 ± 0.00**
	Purity	65.10 ± 0.00	65.10 ± 0.00	65.90 ± 0.00	65.92 ± 0.00	67.11 ± 0.16	68.36 ± 0.00	67.19 ± 0.00	66.71 ± 0.52	65.62 ± 0.00	69.01 ± 0.00	65.10 ± 0.00	**70.23 ± 0.00**
	ARI	−0.01 ± 0.00	−0.05 ± 0.00	10.01 ± 0.00	10.02 ± 0.00	11.60 ± 0.22	12.92 ± 0.00	11.70 ± 0.00	11.06 ± 0.73	9.57 ± 0.00	14.30 ± 0.00	0 ± 0.00	**16.14 ± 0.00**
Flower17	ACC	51.01 ± 1.27	43.57 ± 1.66	57.69 ± 1.21	57.74 ± 2.19	61.20 ± 1.78	55.95 ± 1.90	59.38 ± 1.47	59.63 ± 1.37	52.66 ± 1.12	60.99 ± 1.10	54.86 ± 1.15	**62.64 ± 1.63**
	NMI	49.66 ± 0.79	44.33 ± 1.26	56.14 ± 0.74	56.01 ± 1.36	58.95 ± 0.81	54.33 ± 1.08	57.56 ± 0.78	57.85 ± 1.04	52.31 ± 0.45	60.26 ± 0.34	54.62 ± 0.49	**61.74 ± 1.08**
	Purity	52.00 ± 0.99	45.05 ± 1.44	59.27 ± 1.17	58.79 ± 2.13	62.68 ± 1.48	56.91 ± 1.82	60.55 ± 1.39	60.79 ± 1.37	53.54 ± 0.92	62.01 ± 0.90	56.04 ± 0.99	**63.56 ± 1.56**
	ARI	32.29 ± 0.99	26.36 ± 1.31	39.48 ± 0.87	39.22 ± 1.90	43.67 ± 1.37	37.20 ± 1.37	41.51 ± 1.07	41.77 ± 1.17	34.88 ± 0.73	44.29 ± 0.46	37.32 ± 0.63	**46.23 ± 1.19**
COIL_20	ACC	75.95 ± 3.67	74.87 ± 2.41	76.02 ± 2.78	74.76 ± 2.86	75.76 ± 3.20	74.28 ± 3.54	76.33 ± 2.58	76.10 ± 3.16	75.13 ± 3.36	75.82 ± 2.91	78.18 ± 2.87	**80.16 ± 2.32**
	NMI	83.65 ± 1.86	83.01 ± 1.35	83.70 ± 1.51	82.78 ± 1.57	83.50 ± 1.53	81.81 ± 2.08	83.62 ± 1.49	83.54 ± 1.55	83.99 ± 1.36	83.20 ± 1.38	86.93 ± 1.37	**89.09 ± 1.10**
	Purity	77.97 ± 3.18	76.94 ± 2.41	77.81 ± 2.59	76.75 ± 2.59	77.49 ± 3.01	76.35 ± 3.12	77.95 ± 2.58	77.66 ± 2.73	77.25 ± 2.60	77.47 ± 2.49	80.58 ± 2.27	**82.14 ± 2.13**
	ARI	71.08 ± 3.65	69.86 ± 2.61	71.29 ± 2.70	69.76 ± 3.03	71.07 ± 3.01	68.54 ± 3.86	71.31 ± 2.86	71.15 ± 3.20	71.49 ± 3.35	70.81 ± 2.85	75.41 ± 2.83	**78.06 ± 2.09**
SenslTVehicle	ACC	53.73 ± 0.00	53.36 ± 0.64	54.22 ± 0.00	54.33 ± 0.93	63.33 ± 0.00	55.58 ± 4.32	54.13 ± 0.00	54.27 ± 0.00	64.33 ± 0.00	68.80 ± 0.00	60.30 ± 3.54	**69.32 ± 0.13**
	NMI	10.83 ± 0.00	10.25 ± 0.97	11.22 ± 0.00	11.43 ± 0.00	20.04 ± 0.00	12.77 ± 0.00	11.24 ± 0.00	11.24 ± 0.00	19.61 ± 0.00	25.42 ± 0.00	16.89 ± 3.96	**26.69 ± 0.19**
	Purity	53.73 ± 0.00	53.36 ± 0.64	54.22 ± 0.00	54.33 ± 0.00	63.33 ± 0.00	55.58 ± 4.32	54.13 ± 0.00	54.27 ± 0.00	64.33 ± 0.00	68.80 ± 0.00	60.30 ± 3.54	**69.32 ± 0.13**
	ARI	10.84 ± 0.00	10.29 ± 1.41	11.30 ± 0.00	11.49 ± 0.00	23.00 ± 0.00	12.99 ± 4.95	11.33 ± 0.00	11.33 ± 0.00	22.06 ± 0.00	28.29 ± 0.00	17.43 ± 4.20	**29.42 ± 0.23**
Caltech101	ACC	34.16 ± 0.98	32.81 ± 0.92	35.19 ± 1.10	35.40 ± 0.93	36.96 ± 1.13	36.22 ± 1.24	36.20 ± 1.18	36.28 ± 0.93	37.87 ± 0.85	36.95 ± 1.00	43.50 ± 0.65	**45.01 ± 1.18**
	NMI	59.30 ± 0.56	58.57 ± 0.49	59.92 ± 0.67	60.04 ± 0.52	61.37 ± 0.58	60.88 ± 0.50	60.68 ± 0.54	60.82 ± 0.38	62.24 ± 0.35	61.53 ± 0.65	64.87 ± 0.36	**66.03 ± 0.53**
	Purity	36.22 ± 0.95	34.88 ± 0.89	37.44 ± 0.99	37.61 ± 0.87	39.25 ± 1.06	38.20 ± 1.07	38.20 ± 1.07	38.43 ± 0.75	40.04 ± 0.76	39.05 ± 1.01	45.60 ± 0.56	**47.31 ± 0.97**
	ARI	18.42 ± 0.89	17.34 ± 0.73	19.37 ± 1.13	19.49 ± 0.75	21.04 ± 0.87	20.43 ± 1.00	20.40 ± 0.95	20.51 ± 0.71	22.63 ± 0.68	21.16 ± 1.02	26.33 ± 0.64	**27.29 ± 0.87**
UCI_DIGIT	ACC	88.84 ± 0.12	47.27 ± 0.67	90.92 ± 0.00	91.10 ± 0.00	94.13 ± 0.00	93.65 ± 4.77	90.47 ± 0.09	90.44 ± 0.06	90.96 ± 3.05	94.01 ± 0.12	94.56 ± 1.97	**98.03 ± 0.00**
	NMI	80.78 ± 0.21	48.78 ± 0.70	83.83 ± 0.00	83.79 ± 0.00	87.82 ± 0.00	90.47 ± 2.48	83.57 ± 0.16	83.58 ± 0.06	85.13 ± 1.44	87.80 ± 0.17	90.24 ± 1.20	**95.44 ± 0.00**
	Purity	88.84 ± 0.12	50.12 ± 0.96	90.92 ± 0.00	91.10 ± 0.00	94.13 ± 0.00	93.94 ± 4.19	90.47 ± 0.09	90.44 ± 0.06	91.13 ± 2.52	94.01 ± 0.12	94.56 ± 1.97	**98.03 ± 0.00**
	ARI	77.54 ± 0.22	31.35 ± 0.62	81.29 ± 0.00	81.62 ± 0.11	87.37 ± 0.00	89.37 ± 4.54	80.58 ± 0.18	80.59 ± 0.12	81.86 ± 2.30	87.25 ± 0.23	89.06 ± 2.15	**95.66 ± 0.00**
CCV	ACC	19.63 ± 0.59	18.00 ± 0.49	22.58 ± 0.70	22.81 ± 0.66	21.79 ± 0.74	21.83 ± 0.71	22.01 ± 0.58	22.26 ± 0.65	18.28 ± 0.46	20.50 ± 0.93	21.67 ± 0.54	**24.42 ± 0.43**
	NMI	16.84 ± 0.35	15.06 ± 0.46	18.63 ± 0.37	18.90 ± 0.41	18.03 ± 0.38	18.28 ± 0.36	18.09 ± 0.35	18.16 ± 0.30	13.62 ± 0.40	17.23 ± 0.31	18.66 ± 0.30	**20.31 ± 0.20**
	Purity	23.75 ± 0.48	22.21 ± 0.45	25.63 ± 0.47	25.91 ± 0.51	24.90 ± 0.59	24.84 ± 0.65	25.14 ± 0.43	25.33 ± 0.47	22.07 ± 0.42	23.92 ± 0.00	24.84 ± 0.50	**26.90 ± 0.30**
	ARI	6.60 ± 0.22	5.75 ± 0.24	7.76 ± 0.23	7.98 ± 0.23	7.47 ± 0.23	7.44 ± 0.23	7.54 ± 0.20	7.53 ± 0.18	5.22 ± 0.20	6.98 ± 0.17	7.78 ± 0.23	**8.88 ± 0.18**
Flower102	ACC	27.07 ± 0.84	22.41 ± 0.54	40.25 ± 1.01	40.01 ± 0.96	43.61 ± 0.95	43.08 ± 1.17	42.18 ± 1.25	42.18 ± 1.11	31.23 ± 0.80	43.40 ± 1.04	32.38 ± 0.69	**47.35 ± 1.01**
	NMI	46.02 ± 0.45	42.67 ± 0.21	56.74 ± 0.52	56.47 ± 0.50	59.76 ± 0.33	59.45 ± 0.32	58.40 ± 0.55	58.66 ± 0.59	50.76 ± 0.28	59.33 ± 0.33	52.69 ± 0.28	**62.46 ± 0.30**
	Purity	32.27 ± 0.58	27.79 ± 0.36	46.43 ± 0.97	46.09 ± 0.65	50.23 ± 0.68	49.98 ± 0.67	48.30 ± 0.83	48.82 ± 0.93	38.30 ± 0.51	49.52 ± 0.68	40.35 ± 0.58	**54.22 ± 0.66**
	ARI	15.46 ± 0.53	12.06 ± 0.38	25.54 ± 0.64	25.51 ± 0.76	29.62 ± 0.68	28.92 ± 0.89	28.02 ± 0.86	28.41 ± 0.96	19.23 ± 0.56	29.23 ± 0.78	20.34 ± 0.54	**32.63 ± 0.80**

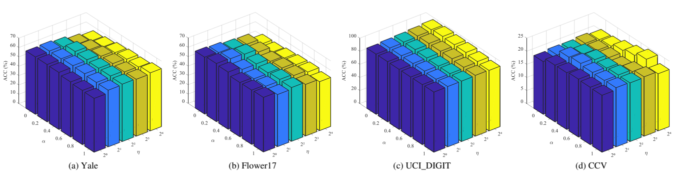


Fig. 2. Sensitivity of SMKKM-HK to hyperparameters α and η in terms of ACC.

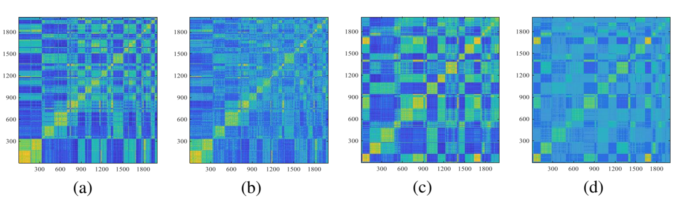


Fig. 3. Kernel matrix visualizations on the UCI_DIGIT dataset. (a) Initial fused kernel of SMKKM. (b) Final learned kernel of SMKKM. (c) Initial kernel of SMKKM-HK. (d) Final learned kernel of SMKKM-HK.

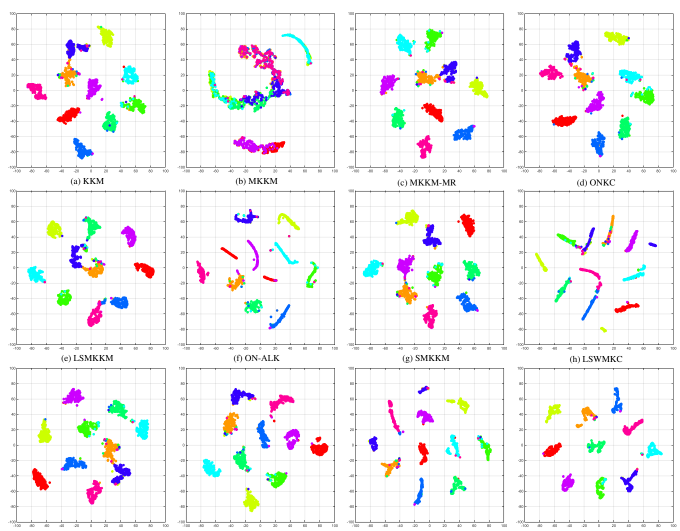


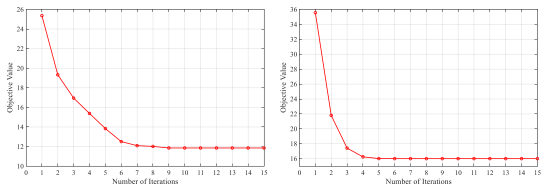
Fig. 4. Comparison of t-SNE visualizations generated by different multiple kernel clustering methods.

E. Visualization Analysis

To visually compare the ability of SMKKM and SMKKM-HK to capture sample similarity structures, Fig. 3 illustrates that the SMKKM kernel matrix is relatively dispersed and irregular, whereas the final SMKKM-HK kernel matrix exhibits a continuous and clear block-diagonal pattern, indicating a more effective representation of the intrinsic similarity structure of the data.

To visually compare the clustering results, Fig. 4 presents two-dimensional t-distributed stochastic neighbor embedding (t-SNE) visualizations of each algorithm on the UCI_DIGIT dataset. It can be observed that some comparison methods still exhibit cluster overlap in the low-dimensional space, whereas SMKKM-HK produces more compact and well-separated clusters with higher intra-cluster consistency, indicating its effectiveness in capturing of the data.

F. Convergence Analysis



(a) Pima (b) Flower17

Fig. 5. Convergence curves of SMKKM-HK on dataset.

As shown in Fig. 5, the objective function of SMKKM-HK decreases rapidly in the initial stage and stabilizes after a few iterations, indicating efficient and stable convergence of the proposed algorithm. This convergence behavior is consistent with its ability to capture the underlying structure of multi-kernel data.

IV. Conclusion

This paper proposes SMKKM-HK, a simple multiple kernel k-means clustering method that incorporates heat kernel diffusion. By integrating a diffusion mechanism into the kernel space, the proposed method effectively combines local neighborhood relationships with global structural information, enabling the learned kernel representation to preserve the geometric characteristics of the data while exhibiting enhanced consistency and discriminative capability. Furthermore, the model jointly updates the kernel representation and clustering assignments during optimization, allowing them to mutually reinforce each other and progressively improve, thereby alleviating the structural bias caused by fixed kernel representations. Future work will focus on theoretical convergence analysis and extensions to large-scale datasets and complex multi-view scenarios.

Acknowledgment

This work was supported in part by the National Natural Science Foundation of China (Grant No. 62366029), in part by the Key Research and Development Program of Gansu Province (Grant No. 25YFFA089).

References

[1] J. Li, R. Ma, M. Deng, X. Cao, X. Wang, and X. Wang, "A comparative study of clustering algorithms for intermittent heating demand considering time series," *Appl. Energy*, vol. 353, pp. 122046, 2024.

[2] Y. Su, L. Gao, M. Jiang, A. Plaza, X. Sun, and B. Zhang, "NSCKL: Normalized spectral clustering with kernel-based learning for semisupervised hyperspectral image classification," *IEEE Trans. Cybern.*, vol. 53, no. 10, pp. 6649–6662, 2023.

[3] H. Liu, W. Li, W. Jia, H. Sun, M. Zhang, L. Song, and Y. Gui, "Clusterformer for pine tree disease identification based on UAV remote sensing image segmentation," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, pp. 1–15, 2024.

[4] S. Qiu, J. Ye, J. Zhao, L. He, L. Liu, and X. Huang, "Video anomaly detection guided by clustering learning," *Pattern Recognit.*, vol. 153, pp. 110550, 2024.

[5] Z. Zhang, X. Chen, C. Wang, R. Wang, W. Song, and F. Nie, "Structured multi-view k-means clustering," *Pattern Recognit.*, vol. 160, pp. 111113, 2025.

[6] K. Xu, L. Chen, and S. Wang, "Towards robust nonlinear subspace clustering: A kernel learning approach," *IEEE Trans. Artif. Intell.*, vol. 7, no. 2, pp. 726-739, 2026.

[7] I. S. Dhillon, Y. Guan, and B. Kulis, "Kernel k-means: Spectral clustering and normalized cuts," in *Proc. 10th ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.*, 2004, pp. 551–556.

[8] M. Gönen and E. Alpaydın, "Multiple kernel learning algorithms," *J. Mach. Learn. Res.*, vol. 12, pp. 2211–2268, 2011.

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

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

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

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

[13] J. Liu, X. Liu, J. Xiong, Q. Liao, S. Zhou, S. Wang, and Y. Yang, "Optimal neighborhood multiple kernel clustering with adaptive local kernels," *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 6, pp. 2872–2885, 2022.

[14] X. Liu, "SimpleMKKM: Simple multiple kernel k-means," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 4, pp. 5174–5186, 2023.

[15] M. Li, Y. Zhang, S. Liu, Z. Liu, and X. Zhu, "Simple multiple kernel k-means with kernel weight regularization," *Inf. Fusion*, vol. 100, pp. 101902, 2023.

[16] M. Li, Y. Zhang, C. Ma, S. Liu, Z. Liu, J. Yin, and Q. Liao, "Regularized simple multiple kernel k-means with kernel average alignment," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 35, no. 11, pp. 15910–15919, 2024.

[17] X. Liu, S. Zhou, L. Liu, C. Tang, S. Wang, J. Liu, and Y. Zhang, "Localized simple multiple kernel k-means," in *Proc. IEEE Int. Conf. Computer Vision*, 2021, pp. 9293–9301.

[18] X. Liu, Y. Zhang, L. Liu, C. Tang, L. Peng, L. Lan, and D. Hu, "Sample adaptive localized simple multiple kernel k-means and its application in parcellation of human cerebral cortex," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 47, no. 9, pp. 8134–8147, 2025.

[19] R. Han, G. Yao, M. Li, W. Zhang, Y. Li, W. Xin, H. Lei, M. Li, Z. Zhang, C. Du, and Y. Tian, "Multi-relation graph-kernel strengthen network for graph-level clustering," in *Proc. 2025 Int. Joint Conf. Neural Networks*, 2025, pp. 1–8.

[20] J. Wang, Z. Li, C. Tang, S. Liu, X. Wan, and X. Liu, "Multiple kernel clustering with adaptive multi-scale partition selection," *IEEE Trans. Knowl. Data Eng.*, vol. 36, no. 11, pp. 6641–6652, 2024.

[21] L. Li, S. Wang, X. Liu, E. Zhu, L. Shen, K. Li, and K. Li, "Local sample-weighted multiple kernel clustering with consensus discriminative graph," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 35, no. 2, pp. 1721–1734, 2024.

[22] W. Yang, C. Tang, X. Liu, G. Yue, Y. Liu, C. Zhang, and E. Zhu, "Smooth multiple kernel k-means via underlying graph filtering," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 36, no. 8, pp. 14855–14868, 2025.

[23] G. Yang, J. Zou, Y. Chen, L. Du, and P. Zhou, "Heat kernel diffusion for enhanced late fusion multi-view clustering," *IEEE Signal Process. Lett.*, vol. 31, pp. 2310–2314, 2024.

[24] J. You, Z. Ren, F. R. Yu, and X. You, "One-stage shifted Laplacian refining for multiple kernel clustering," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 35, no. 8, pp. 11501–11513, 2024.

[25] Y. Chen, L. Du, P. Zhou, L. Duan, and Y. Qian, "Multiple kernel clustering with local kernel reconstruction and global heat diffusion," *Inf. Fusion*, vol. 105, pp. 102219, 2024.

[26] A. Rakotomamonjy, F. Bach, S. Canu, and Y. Grandvalet, "SimpleMKL," *J. Mach. Learn. Res.*, vol. 9, pp. 2491–2521, 2008.