

# Enhancing Kernel Power $K$ -means: Scalable and Robust Clustering with Random Fourier Features and Possibilistic Method

Yixi Chen<sup>1</sup>, Weixuan Liang<sup>1\*</sup>, Tianrui Liu<sup>1</sup>, Jun-Jie Huang<sup>1</sup>, Ao Li<sup>2</sup>, Xueling Zhu<sup>3\*</sup>, Xinwang Liu<sup>1</sup>

<sup>1</sup>College of Computer Science and Technology, National University of Defense Technology, Changsha, China, 410073

<sup>2</sup>School of Computer Science and Technology, Harbin University of Science and Technology, Harbin, China, 150080

<sup>3</sup>Department of Radiology, Xiangya Hospital, Central South University, Changsha, China, 410008

chenyixi20@nudt.edu.cn, weixuanliang@nudt.edu.cn, xuercan@163.com

## Abstract

Kernel power  $k$ -means (KPKM) leverages a family of means to mitigate local minima issues in kernel  $k$ -means. However, KPKM faces two key limitations: (1) the computational burden of the full kernel matrix restricts its use on extensive data, and (2) the lack of authentic centroid-sample assignment learning reduces its noise robustness. To overcome these challenges, we propose RFF-KPKM, introducing the first approximation theory for applying random Fourier features (RFF) to KPKM. RFF-KPKM employs RFF to generate efficient, low-dimensional feature maps, bypassing the need for the whole kernel matrix. Crucially, we are the first to establish strong theoretical guarantees for this combination: (1) an excess risk bound of  $\mathcal{O}(\sqrt{k^3/n})$ , (2) strong consistency with membership values, and (3) a  $(1 + \varepsilon)$  relative error bound achievable using the RFF of dimension  $\text{poly}(\varepsilon^{-1} \log k)$ . Furthermore, to improve robustness and the ability to learn multiple kernels, we propose IP-RFF-MKPKM, an improved possibilistic RFF-based multiple kernel power  $k$ -means. IP-RFF-MKPKM ensures the scalability of MKPKM via RFF and refines cluster assignments by combining the merits of the possibilistic and fuzzy membership. Experiments on large-scale datasets demonstrate the superior efficiency and clustering accuracy of the proposed methods compared to the state-of-the-art alternatives.

## Introduction

Kernel  $k$ -means (KKM) is a cornerstone methodology in unsupervised learning, demonstrating broad applicability across various practical domains through its effective pattern recognition capabilities (Filippone et al. 2008; Wang et al. 2023a,b; Feng et al. 2025a,b). However, KKM remains prone to sub-optimal local minima due to the non-convexity of the underlying objective function (Jain 2010; Tang et al. 2012; Sun, Toh, and Yuan 2021). To address the problem, a series of works has focused on reducing the non-convexity of the  $k$ -means objective function. Instead of minimizing the sum of squared distances to the nearest centroid, convex clustering (Pelckmans et al. 2005) minimizes the distance between data points and their convex combinations, yielding a convex optimization problem with a unique optimal solution. In contrast,

$k$ -harmonic means (KHM) (Zhang, Hsu, and Dayal 1999) minimizes the harmonic mean of distances to all centroids. Recently, Power  $k$ -means (PKM) (Xu and Lange 2019) generalizes KHM by replacing harmonic averaging with power means and incorporating an annealing strategy to decrease the power exponent during iterations gradually. Building on this, Kernel Power  $k$ -means (KPKM) (Paul et al. 2022) extends the framework to handle non-linearly separable data through kernelization.

Although KPKM mitigates the tendency of KKM to fall into local optima, it cannot be directly applied to large-scale scenarios due to its quadratic time complexity with respect to the number of instances  $n$ . Current clustering application areas increasingly involve large-scale datasets containing thousands or even millions of instances, such as social network analysis (Malliaros and Vazirgiannis 2013), bioinformatics (Zou et al. 2020), and financial market analysis (Kou, Peng, and Wang 2014; Li et al. 2021). However, KPKM requires computing an  $n \times n$  kernel matrix, leading to  $\mathcal{O}(n^2k)$  complexity per iteration ( $k$  denotes the number of clusters). Such computational demands become prohibitive when handling large-scale datasets in the scenarios above.

To address the time complexity problem, we propose RFF-based kernel power  $k$ -means (RFF-KPKM). Specifically, we employ a novel technique known as random Fourier features (RFF) (Chitta, Jin, and Jain 2012) to avoid computing the whole kernel matrix. RFF’s advantage lies in its ability to directly construct an explicit mapping that approximates the kernel similarity via the inner product of the mapped data. RFF-KPKM significantly reduces the complexity of KPKM from  $\mathcal{O}(n^2k)$  to  $\mathcal{O}(n(k+d)D)$ , where  $d$  represents the data dimensionality and  $D$  denotes the RFF dimensionality ( $D \ll n$  typically). Beyond computational efficiency, our method is rigorously grounded in theoretical guarantees. We pioneer the first theoretical framework for approximating KPKM via RFF, establishing rigorous guarantees in three key dimensions. First, we establish an excess risk bound of  $\mathcal{O}(\sqrt{k^3/n})$  for the approximation method (Theorem 1), matching the optimal rate of the original method. This preservation of statistical guarantees demonstrates that our complexity reduction does not compromise theoretical reliability. Second, we establish a strong consistency guarantee for the established method (Theorem 2), demonstrating that as  $n \rightarrow \infty$ , RFF dimensionality  $D \rightarrow \infty$  and power parameter

\*Corresponding author

Copyright © 2026, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

$s \rightarrow -\infty$ , the membership value converges to the globally optimal solution of the expected version of KKM with any constant probability. Lastly, we discuss how many RFF are needed to preserve accuracy. We obtain that when  $n \rightarrow \infty$ ,  $s \rightarrow -\infty$  and  $D = \text{poly}(\varepsilon^{-1} \log k)$ , the membership value obtained by RFF-KPKM will produce a  $(1 + \varepsilon)$  relative error on the objective of expected KKM (Theorem 3).

Besides time complexity, another issue KPKM faces is noise sensitivity. Similar to fuzzy  $c$ -means (FCM) (Bezdek, Ehrlich, and Full 1984), the cluster centroids of KPKM are affected by the fuzzy membership value of the data. However, the fuzzy membership value  $w_{ij}$  of data point  $\mathbf{x}_i$  in cluster  $C_j$  is determined by the distance between  $\mathbf{x}_i$  to all  $k$  cluster centroids, while Krishnapuram and Keller (1993) propose that such memberships may cause a noise problem since they do not reflect the absolute degree of typicality (or “belonging”) of a point in a cluster. Possibilistic  $c$ -means (PCM) (Krishnapuram and Keller 1993) was proposed to learn the possibilistic membership of data, which is determined solely by the distance of a point from a cluster, thereby reflecting the absolute degree of typicality and resolving the noise problem.

To address the noise problem in KPKM, we integrate our RFF-KPKM framework with the robust clustering paradigm of PCM. Specifically, we replace the distance function in RFF-KPKM with the objective function of PCM to learn the possibilistic membership of data. Also, by calculating the partial derivative of the objective function, we can obtain the fuzzy membership of the data. The product of possibilistic membership and fuzzy membership forms the final membership value. By integrating fuzzy and possibility methods, our method enhances KPKM’s robustness and prevents the identical cluster issues commonly encountered in PCM (Zhang and Leung 2004). Furthermore, to extend our method to multi-view scenarios, we combine it with the kernel combination method in multiple kernel (view) learning (Gönen and Alpaydm 2011; Guan et al. 2025a,b,c), constructing reweighted RFF to learn the importance of mapping features in different views. Our method is termed improved possibilistic RFF-based multiple kernel power  $k$ -means (IP-RFF-MKPK). To obtain the optimal solution, we conduct a novel majority minimization (MM) algorithm (Mairal 2013).

This work makes the following primary contributions:

- 1) We propose RFF-KPKM to accelerate KPKM. Theoretically, we obtain that RFF with  $\Omega(nd/k)$  dimensions achieves the excess risk bound  $\mathcal{O}(\sqrt{k^3/n})$ . To the best of our knowledge, this is the first bound on excess risk for the approximate kernel clustering based on RFF.
- 2) We further establish the strong consistency of the membership matrix produced by our algorithm when  $n \rightarrow \infty$ , RFF dimensionality  $D \rightarrow \infty$ , and power parameter  $s \rightarrow -\infty$ . When  $D$  is finite, we show that  $\text{poly}(\varepsilon^{-1} \log k)$  RFF suffices to achieve the  $(1 + \varepsilon)$  relative error bound.

- 3) We propose IP-RFF-MKPKM to solve the noise sensitivity problem in KPKM and extend RFF-KPKM to multi-view settings. The proposed method is more robust than MKPKM and is scalable to large-scale schemes. Experiments on large-scale datasets validate the computational efficiency and clustering accuracy of our methods.

## Notations

To prevent ambiguity, boldface uppercase and lowercase characters are used to represent matrices and vectors. Specifically,  $\mathbf{A}$  represents a matrix and  $\mathbf{a}$  represents a vector. The component of them is denoted by  $A_{ij}$  or  $a_i$ . We denote with  $\mathcal{X} \subset \mathbb{R}^d$  the sample space and with  $\rho$  the corresponding data distribution. The training set  $S_n = \{\mathbf{x}_i\}_{i=1}^n \subset \mathcal{X}$  is drawn i.i.d from  $\rho$  with compact support  $C \subset \mathbb{R}^d$ . The empirical distribution  $\rho_n$  is defined as  $\rho_n(\mathbf{x}) = \frac{1}{n}$  if  $\mathbf{x} \in S_n$ , otherwise 0. In this paper, we adopt the Gaussian kernel as the default choice throughout our analysis. We denote the Gaussian kernel function as  $k(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\|\mathbf{x}_i - \mathbf{x}_j\|^2/2\sigma^2)$  where  $\sigma$  is bandwidth parameter. As proposed in (Aronszajn 1950), there exist a feature mapping  $\phi : \mathcal{X} \rightarrow \mathcal{H}$  such that  $\forall \mathbf{x}_i, \mathbf{x}_j \in \mathcal{X}$ ,  $k(\mathbf{x}_i, \mathbf{x}_j) = \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle$ , where  $\mathcal{H}$  is a Hilbert space. We denote with  $\text{conv}(A)$  the closed convex hull of a set  $A$ . Finally, we assume  $S_n$  is bounded.

## Related Work

### Kernel Power $K$ -means

KPKM (Paul et al. 2022) embeds the kernel  $k$ -means problem within a continuum of well-posed surrogate problems. These intermediate formulations employ progressively smoothed objectives, which tend to guide clustering toward the global minimum of the original kernel  $k$ -means optimization landscape. The formula of this problem is

$$f_s(\Theta) = \sum_{i=1}^n M_s(\|\phi(\mathbf{x}_i) - \theta_1\|^2, \dots, \|\phi(\mathbf{x}_i) - \theta_k\|^2), \quad (1)$$

where  $M_s(\mathbf{y}) = (\frac{1}{k} \sum_{i=1}^k y_i^s)^{1/s}$  for a vector  $\mathbf{y}$  and is called power means. KPKM seeks to minimize  $f_s$  iteratively while sending  $s \rightarrow -\infty$ . As stated in (Paul et al. 2022), when  $s$  is small, the objective of the power  $k$ -means is smoother than that of the kernel  $k$ -means, so the solution is less likely to fall into local minima and is more robust. When  $s \rightarrow -\infty$ ,  $f_s(\Theta)$  converges to the objective of kernel  $k$ -means since  $M_s(\mathbf{y}) \rightarrow \min\{y_1, y_2, \dots, y_k\}$ , which implies that the minimizer  $f_s(\Theta)$  gradually converges to the minimizer of the  $k$ -means.

### Random Fourier Features

RFF was first introduced in (Rahimi and Recht 2007), which maps the data into a low-dimensional space such that the inner product of the mapped data points approximates the kernel similarity between them. Specifically, for shift-invariant kernel  $k(\mathbf{x}, \mathbf{y})$  such that it can be expressed as  $K(\mathbf{x} - \mathbf{y}) : \mathbb{R}^d \rightarrow \mathbb{R}$ , function  $p : \mathbb{R}^d \rightarrow \mathbb{R}$  such that  $p(\omega) = \frac{1}{2\pi} \int_{\mathbb{R}^d} K(\mathbf{x}) e^{-i\omega^\top \mathbf{x}} d\mathbf{x}$ , the RFF mapping is defined as:

$$\tilde{\phi}(x) := \sqrt{\frac{1}{D}} \begin{pmatrix} \sin(\omega_1^\top \mathbf{x}) \\ \cos(\omega_1^\top \mathbf{x}) \\ \vdots \\ \sin(\omega_D^\top \mathbf{x}) \\ \cos(\omega_D^\top \mathbf{x}) \end{pmatrix}, \quad (2)$$

where frequency vectors  $\omega_1, \omega_2, \dots, \omega_D \in \mathbb{R}^d$  are i.i.d. sampled from distribution with density  $p$ .

Theoretically, Rahimi and Recht (2007) have proven that  $\mathbb{E}[\langle \tilde{\phi}(\mathbf{x}), \tilde{\phi}(\mathbf{y}) \rangle] = \frac{1}{D} \sum_{i=1}^D \mathbb{E}[\cos(\omega_i, \mathbf{x} - \mathbf{y})] = K(\mathbf{x} - \mathbf{y})$ . Practically, compared to the Nyström method (Williams and Seeger 2000), the RFF approach directly generates low-dimensional random feature vectors, eliminating the need to store kernel matrices while bypassing eigendecomposition and subsample selection. The application of RFF significantly reduces both space and time complexity.

## RFF-based Kernel Power $K$ -means

### Problem Statement

In RFF-KPKM, we aim to optimize the membership matrix as our primary objective rather than the cluster centroids. This is because we can directly use the membership matrix as input for algorithms like KKM or KPK to compute the approximation loss. However, the cluster centroids output by RFF-KPKM cannot since they cannot reflect the position of cluster centroids before mapping. To achieve the goal, let us review the update rule for cluster centroid  $\theta_j$  in KPKM:

$$\theta_j^{(m+1)} = \frac{\sum_{i=1}^n w_{ij}^{(m)} \phi(\mathbf{x}_i)}{\sum_{i=1}^n w_{ij}^{(m)}}, \quad (3)$$

where

$$w_{ij}^{(m)} = \frac{\frac{1}{k} \left\| \phi(\mathbf{x}_i) - \theta_j^{(m)} \right\|^{2(s-1)}}{\left( \frac{1}{k} \sum_{l=1}^k \left\| \phi(\mathbf{x}_i) - \theta_l^{(m)} \right\|^{2s} \right)^{(1-1/s)}}. \quad (4)$$

By the observation that  $\theta_j$  is formed as the convex combination of mapped data, we can reformulate KPKM as follows:

$$\begin{aligned} \min_{\mathbf{W} \in \mathbb{R}^{n \times k}} f_s(\mathbf{W}) &= \sum_{i=1}^n M_s(d_{i1}^2, \dots, d_{ik}^2), \\ \text{s.t. } \forall j \in \{1, \dots, k\}, \quad &\sum_{i=1}^n W_{ij} = 1, \\ d_{ij} &= \|\phi(\mathbf{x}_i) - \theta_j\|, \quad \theta_j = \Phi \mathbf{W}^{(j)}, \end{aligned} \quad (5)$$

where  $\Phi$  is the mapped data matrix and the  $i$ -th column of  $\Phi$  is  $\phi(\mathbf{x}_i)$ . Such reformulation is quite reasonable according to Theorem 1 in (Paul et al. 2022): the minimizer of  $f_s(\Theta)$  lies in a convex hull in the compact Cartesian product  $\text{conv}(\phi(C))^k$ . We refer to  $\mathbf{W}$  as the membership matrix, as it determines the weight of data points in influencing the position of cluster centroids. To simplify notation, we denote  $M_s(\|\phi(\mathbf{x}_i) - \Phi \mathbf{W}^{(1)}\|^2, \dots, \|\phi(\mathbf{x}_i) - \Phi \mathbf{W}^{(k)}\|^2)$  as  $M_s(\phi(\mathbf{x}_i), \mathbf{W})$ .

It is crucial to note that we cannot directly access the explicit form of accurate cluster centroids during iterations. KPKM solves the problem by using the kernel trick because the algorithm's update process only requires computing point-to-cluster distances. However, the computation of the kernel matrix leads to a huge computational cost. Unlike KPKM, we employ RFF to calculate low-dimensional embeddings of the data as a surrogate for their high-dimensional representations  $\phi(\mathbf{x})$  in the Hilbert space. This allows us to explicitly compute cluster centroids during iterations while significantly

reducing the computational overhead associated with kernel matrix operations. By substituting the origin implicit mapping  $\phi$  in Eq. (5) with RFF mapping  $\tilde{\phi} : \mathbb{R}^d \rightarrow \mathbb{R}^D$  defined in Eq. (2) where  $D \ll d$ , we obtain the objective function  $\tilde{f}_s(\mathbf{W})$  of proposed RFF-KPKM. The optimization procedure for RFF-KPKM mirrors that of standard KPKM, with the distinction that feature vectors  $\mathbf{x}$  are replaced by our constructed RFF mapping  $\tilde{\phi}(\mathbf{x})$ . The complete pseudocode for the optimization algorithm is provided in the appendix.

To evaluate the effect upon precision and consistency when substituting  $\phi(\mathbf{x})$  with  $\tilde{\phi}(\mathbf{x})$ , we establish an excess risk bound, strong consistency, and a relative error bound theoretically. The full result is stated in the following subsection.

### Theoretical Analysis

**Excess Risk Bound** In this section, we obtain the excess risk bound of the proposed RFF-KPKM. Here, we assume that our analysis is confined to the Gaussian kernel scenario; however, our theorems also hold for all positive definite translation-invariant kernels that take strictly positive values (see the proof in Appendix <sup>1</sup>). We now define the expected version of the KPKM objective function as follows:

$$\mathcal{L}_s(\mathbf{W}, \rho) = \int M_s(\phi(\mathbf{x}), \mathbf{W}) d\rho(\mathbf{x}). \quad (6)$$

Given a training set  $S_n = \{\mathbf{x}_i\}_{i=1}^n$ , the empirical version of the KPKM and RFF-KPKM can be written as:

$$\mathcal{L}_s(\mathbf{W}, \rho_n) = \frac{1}{n} f_s(\mathbf{W}), \quad (7)$$

$$\tilde{\mathcal{L}}_s(\mathbf{W}, \rho_n) = \frac{1}{n} \tilde{f}_s(\mathbf{W}). \quad (8)$$

The empirical risk minimizer (ERM) is defined as

$$\mathbf{W}_{n,s} := \arg \min_{\mathbf{W} \in \mathcal{A}} \mathcal{L}_s(\mathbf{W}, \rho_n), \quad (9)$$

$$\tilde{\mathbf{W}}_{n,s} := \arg \min_{\mathbf{W} \in \mathcal{A}} \tilde{\mathcal{L}}_s(\mathbf{W}, \rho_n), \quad (10)$$

where  $\mathcal{A} = \{\mathbf{X} \in \mathbb{R}^{n \times k} : \sum_{i=1}^n \mathbf{X}_{ij} = 1, \forall j \in \{1, \dots, k\}\}$ . We can now define the excess risk of RFF-KPKM as:

$$\mathcal{E}(\tilde{\mathbf{W}}_{n,s}) := \mathbb{E}[\mathcal{L}_s(\tilde{\mathbf{W}}_{n,s}, \rho)] - \mathcal{L}_s^*(\rho). \quad (11)$$

**Theorem 1.** *When the dimension of RFF  $D = \Omega\left(\frac{nd \log(n/k\delta)}{k}\right)$ , with probability at least  $1 - \delta$  we have*

$$\mathcal{E}(\tilde{\mathbf{W}}_{n,s}) = \mathbb{E}[\mathcal{L}_s(\tilde{\mathbf{W}}_{n,s}, \rho)] - \mathcal{L}_s^*(\rho) \leq \mathcal{O}\left(\sqrt{\frac{k^{3-2/s}}{n}}\right), \quad (12)$$

where  $\mathcal{L}_s^*(\rho) = \inf_{\mathbf{W}} \mathcal{L}_s(\mathbf{W}, \rho)$ .

**Remark:** To our knowledge, Theorem 1 presents the first generalization error bounds that quantify the approximation efficacy of RFF in kernel clustering tasks. Although (Cheng et al. 2023) established a  $(1 + \varepsilon)$  relative error bound for RFF-based  $k$ -means, their results lack the explicit generalization performance of RFF. Separately, (Yin et al. 2022)

<sup>1</sup><https://arxiv.org/abs/2511.10392>

derived excess risk bounds for  $k$ -means based on randomized sketches. However, such sketch methods as random projection, ROS, and Nyström require computing the full or partial kernel matrix, thus failing to circumvent its computational burden.

**Strong Consistency** In this section, we analyze the strong consistency of the ERM of RFF-KPKM:  $\tilde{\mathbf{W}}_{n,s}$ . First, we denote the expected version of KKM as follows:

$$\Psi(\mathbf{W}, \rho) = \int \min_{1 \leq j \leq k} \|\phi(\mathbf{x}_i) - \Phi \mathbf{W}^{(j)}\|^2 d\rho(\mathbf{x}), \quad (13)$$

$$\tilde{\Psi}(\mathbf{W}, \rho) = \int \min_{1 \leq j \leq k} \|\tilde{\phi}(\mathbf{x}_i) - \tilde{\Phi} \mathbf{W}^{(j)}\|^2 d\rho(\mathbf{x}). \quad (14)$$

The minimizer of  $\Psi(\mathbf{W}, \rho)$  is defined as

$$\mathbf{W}^* := \arg \min_{\mathbf{W} \in \mathcal{A}} \Psi(\mathbf{W}, \rho), \quad (15)$$

$$\tilde{\mathbf{W}}^* := \arg \min_{\mathbf{W} \in \mathcal{A}} \tilde{\Psi}(\mathbf{W}, \rho). \quad (16)$$

Establishing consistency equals showing that  $\tilde{\mathbf{W}}_{n,s} \xrightarrow{\text{a.s.}} \mathbf{W}^*$  when  $n \rightarrow \infty$ , the dimensionality of RFF  $D \rightarrow \infty$  and  $s \rightarrow -\infty$ . To address this issue, we introduce the following standard assumption, which allows us to study the convergence of variables by analyzing the limit of the objective function sequence.

**Assumption 1.** Let  $B(\mathbf{W}, r)$  denote the ball with radius  $r$  around  $\mathbf{W}$ . For any  $r > 0$ , there exists  $\varepsilon > 0$  such that for all  $\mathbf{W} \in \mathcal{A} \setminus B(\mathbf{W}^*, r)$ , we have  $\Psi(\mathbf{W}, \rho) > \Psi(\mathbf{W}^*, \rho) + \varepsilon$ .

**Theorem 2.** Under Assumption 1,  $\tilde{\mathbf{W}}_{n,s} \xrightarrow{\text{a.s.}} \mathbf{W}^*$  with any constant probability when  $n \rightarrow \infty$ , RFF dimensionality  $D \rightarrow \infty$  and  $s \rightarrow -\infty$ .

**Remark:** Notably, the assumption is standard for the analysis of the consistency of clustering (Paul et al. 2022; Pollard 1981; Chakraborty and Das 2020). According to Assumption 1, to build consistency is actually to prove that  $\Psi(\tilde{\mathbf{W}}_{n,s}, \rho) \xrightarrow{\text{a.s.}} \Psi(\mathbf{W}^*, \rho)$  when  $n, D \rightarrow \infty$  and  $s \rightarrow -\infty$ . However, Theorem 2 does not specify the appropriate dimension  $D$  for RFF in practical implementations. The following theorem addresses this gap by providing a theoretical justification for the choice of  $D$ .

**Relative Error** In this section, we study the relative error bound deriving from using ERM of RFF-KPKM with finite RFF as the input of expected KKM. We obtain that  $\text{poly}(\varepsilon^{-1} \log k)$  RFF are enough to obtain the  $(1 + \varepsilon)$  error bound. The following theorem shows the main result.

**Theorem 3.** Under Assumption 1, if RFF dimensionality  $\Omega((\log^3(\frac{k}{\delta}) + \log^3(\frac{1}{\varepsilon}) + 2^8)/\varepsilon^2)$ , then with probability at least  $1 - \delta$  we have

$$\lim_{n, s \rightarrow \infty} \Psi(\tilde{\mathbf{W}}_{n,s}, \rho) \leq \frac{1 + \varepsilon}{1 - \varepsilon} \cdot \Psi(\mathbf{W}^*, \rho). \quad (17)$$

**Remark:** Notably, by rescaling  $\varepsilon' = 2\varepsilon/(1 - \varepsilon)$ , we have  $\lim_{n, s \rightarrow \infty} \Psi(\tilde{\mathbf{W}}_{n,s}, \rho) \leq (1 + \varepsilon') \cdot \Psi(\mathbf{W}^*, \rho)$ , which shows the method achieves the  $(1 + \varepsilon')$  relative error. By Theorem 3, we recommend setting the RFF dimension  $D$  to  $\lceil 4 \log^3(2k) \rceil$  to ensure the accuracy, which is obtained by setting  $\varepsilon$  and  $\delta$  to 0.5 and preserving the main term in the optimal  $D$ . It should be noted that the optimal  $D = \text{poly}(\varepsilon^{-1} \log k)$  is independent of  $n$  and  $d$  and sublinear to  $k$ .

## Complexity Analysis

The computational complexity of RFF-KPKM primarily consists of the following main steps. For generating the RFF mapping, the complexity is  $\mathcal{O}(ndD)$ . For updating the cluster centroids  $\Theta$ , the complexity is  $\mathcal{O}(nkD)$ . For computing the distance between data points and cluster centroids, the complexity is  $\mathcal{O}(nkD)$ . In sum, RFF-MKPKM consumes a time complexity of  $\mathcal{O}(n(k + d)D)$  in each iteration, greatly reducing the time complexity  $\mathcal{O}(n^2k)$  of KPKM (Paul et al. 2022) when  $n$  is large.

## Improved Possibilistic RFF-based Multiple Kernel Power $K$ -means

### Problem Statement

To solve the noise problem in RFF-KPKM and extend it to multi-view settings, we combine the PCM and the kernel combination method with RFF-KPKM. As mentioned before, fuzzy membership functions such as Eq. (4) are not appropriate to reflect the belonging degrees since they are determined by the data point to all cluster centroids. To obtain a more appropriate membership, a natural thought is to redefine a new membership as variables to be optimized. PCM inspire us to substitute  $d_{ij} = \|\tilde{\phi}(\mathbf{x}_i) - \theta_j\|^2$  in RFF-KPKM by the following function:

$$\tilde{d}_{ij} = (u_{ij})^m \|\tilde{\phi}(\mathbf{x}_i) - \theta_j\|^2 + (1 - u_{ij})^m \eta_j, \quad (18)$$

where  $u_{ij}$  represents the possibility membership,  $\eta_j$  is a regularization hyper-parameter and  $m$  controls the fuzziness of the membership degrees. Notably, we cannot remove the term  $(1 - u_{ij})^m \eta_j$ , since it is crucial to avoid the trivial solution and produce the membership that only depends on a point to a cluster. Unlike PCM, our method obtains the possibilistic membership while preserving the fuzzy membership  $w_{ij}$ , which can be obtained naturally through calculating the partial derivative of the objective function. The final cluster centroids are formed as

$$\theta_j = \frac{\sum_{i=1}^n w_{ij} (u_{ij})^m \tilde{\phi}(\mathbf{x}_i)}{\sum_{i=1}^n w_{ij} (u_{ij})^m}. \quad (19)$$

We define the new membership function  $\gamma_{ij} = w_{ij} (u_{ij})^m$ , incorporating both possibilistic and fuzzy memberships.  $\gamma_{ij}$  enhances robustness because  $u_{ij}$  measures the absolute degree of typicality of a point in any cluster. Furthermore, the constraint  $\sum_{j=1}^k w_{ij} = 1$  within  $\gamma_{ij}$  prevents identical clusters by ensuring a data point's high fuzzy membership in one cluster necessitates lower memberships in others.

Clustering in multi-view scenarios is becoming increasingly popular and important (Wang et al. 2024a,b,c; Liang et al. 2024a, 2025; Yu et al. 2025; Zhou et al. 2025). To extend to multiple kernel settings, we substitute  $\tilde{\phi}(\mathbf{x}_i)$  by  $\tilde{\phi}_\alpha(\mathbf{x}) := (\sqrt{\alpha_1} \tilde{\phi}_1(\mathbf{x})^\top, \dots, \sqrt{\alpha_L} \tilde{\phi}_L(\mathbf{x})^\top)^\top$  and reparameterize  $\theta_j = (\frac{1}{\sqrt{\alpha_1}} \theta_{j,1}^\top, \dots, \frac{1}{\sqrt{\alpha_L}} \theta_{j,L}^\top)^\top$ , where  $\tilde{\phi}_l(\mathbf{x}_i)$  denotes the RFF for  $\mathbf{x}_i$  in the  $l$ -th view and  $\alpha = (\alpha_1, \dots, \alpha_L)$  act as weights. Finally, the objective function of IP-RFF-

MKPKM is defined as follows:

$$\begin{aligned} \min_{\alpha, \Theta, \mathbf{U}} f_s(\alpha, \Theta, \mathbf{U}) &= \sum_{i=1}^n \mathcal{M}_s(\alpha, \mathbf{d}_i) + \lambda \sum_{l=1}^L \alpha_l \log \alpha_l \\ \text{s.t. } \mathcal{M}_s(\alpha, \mathbf{d}_i) &= M_s \left( \sum_{l=1}^L \alpha_l \tilde{d}_{i1,l}, \dots, \sum_{l=1}^L \alpha_l \tilde{d}_{ik,l} \right), \\ \tilde{d}_{ij,l} &= (u_{ij})^m \|\tilde{\phi}_l(\mathbf{x}_i) - \boldsymbol{\theta}_{j,l}\|^2 + (1 - u_{ij})^m \eta_{j,l}, \\ \forall j \in \{1, \dots, k\}, \sum_{l=1}^L \alpha_l &= 1. \end{aligned} \quad (20)$$

## Optimization

We adopt the MM algorithm to minimize Eq. (20). First, we utilize the concave property of power means to obtain the surrogate function  $g_s(\alpha, \Theta, \mathbf{U})$  of  $f_s(\alpha, \Theta, \mathbf{U})$ . Then we developed an alternating optimization algorithm to optimize  $g_s$ , which simultaneously ensures the reduction of  $f_s$ . Through the tangent plane inequality described in (Paul et al. 2022), we can obtain that

$$\begin{aligned} f_s(\alpha, \Theta, \mathbf{U}) &\leq f_s(\alpha^{(t)}, \Theta^{(t)}, \mathbf{U}^{(t)}) \\ &- \sum_{i=1}^n \sum_{j=1}^k w_{ij}^{(t)} \sum_{l=1}^L \alpha_l^{(t)} \tilde{d}_{ij,l}^{(t)} - \lambda \sum_{l=1}^L \alpha_l^{(t)} \log \alpha_l^{(t)} \\ &+ \sum_{i=1}^n \sum_{j=1}^k w_{ij}^{(t)} \sum_{l=1}^L \alpha_l \tilde{d}_{ij,l} + \lambda \sum_{l=1}^L \alpha_l \log \alpha_l. \end{aligned} \quad (21)$$

Here  $w_{ij}$  are obtained from the partial derivatives of  $M_s$ :

$$w_{ij}^{(t)} = \frac{\frac{1}{k} (\sum_{l=1}^L \alpha_l^{(t)} \tilde{d}_{ij,l}^{(t)})^{(s-1)}}{(\frac{1}{k} \sum_{c=1}^k (\sum_{l=1}^L \alpha_l^{(t)} \tilde{d}_{ic,l}^{(t)})^s)^{(1-1/s)}}, \quad (22)$$

where

$$\tilde{d}_{ij,l}^{(t)} = (u_{ij}^{(t)})^m \|\tilde{\phi}_l(\mathbf{x}_i) - \boldsymbol{\theta}_{j,l}^{(t)}\|^2 + (1 - u_{ij}^{(t)})^m \eta_{j,l}. \quad (23)$$

We now define the entire right-hand side of Inequality (20) as  $g_s(\alpha, \Theta, \mathbf{U})$ . We readily observe that the surrogate function  $g_s$  satisfies the conditions required by the MM algorithm, so we need to find  $(\alpha^{(t+1)}, \Theta^{(t+1)}, \mathbf{U}^{(t+1)})$  such that  $g_s(\alpha^{(t+1)}, \Theta^{(t+1)}, \mathbf{U}^{(t+1)}) \leq g_s(\alpha^{(t)}, \Theta^{(t)}, \mathbf{U}^{(t)})$ . Ignoring the constant terms, optimizing  $g_s(\alpha, \Theta, \mathbf{U})$  is equivalent to solving the following optimization problem when the sum of  $\alpha_l$  equals 1:

$$\min_{\alpha, \Theta, \mathbf{U}} \sum_{i=1}^n \sum_{j=1}^k w_{ij}^{(t)} \sum_{l=1}^L \alpha_l \tilde{d}_{ij,l} + \lambda \sum_{l=1}^L \alpha_l \log \alpha_l. \quad (24)$$

To address the optimization problem in Eq. (24), we propose a three-step alternating iterative algorithm. When optimizing a variable, the other variables are fixed to their previous iteration values. The complete steps are as follows.

**Step1:** Update  $\mathbf{U}$ . By substituting  $\alpha = \alpha^{(t)}$ ,  $\Theta = \Theta^{(t)}$  into Eq. (24) and removing terms unrelated to  $\mathbf{U}$ , we obtain the

following optimization problem:

$$\begin{aligned} \min_{\mathbf{U}} \sum_{i=1}^n \sum_{j=1}^k w_{ij}^{(t)} \left( (u_{ij})^m \sum_{l=1}^L \alpha_l^{(t)} \|\tilde{\phi}_l(\mathbf{x}_i) - \boldsymbol{\theta}_{j,l}^{(t)}\|^2 \right. \\ \left. + (1 - u_{ij})^m \sum_{l=1}^L \alpha_l^{(t)} \eta_{j,l} \right). \end{aligned} \quad (25)$$

Setting the derivative of Eq. (25) with respect to  $u_{ij}$  to zero, we obtain the update rule of  $u_{ij}$  as follows:

$$u_{ij}^{(t+1)} = \frac{1}{1 + \left( \frac{\sum_{l=1}^L \alpha_l^{(t)} \|\tilde{\phi}_l(\mathbf{x}_i) - \boldsymbol{\theta}_{j,l}^{(t)}\|^2}{\sum_{l=1}^L \alpha_l^{(t)} \eta_{j,l}} \right)^{\frac{1}{m-1}}}. \quad (26)$$

**Step2:** Update  $\Theta$ . By substituting  $\alpha = \alpha^{(t)}$ ,  $\mathbf{U} = \mathbf{U}^{(t+1)}$  into Eq. (24) and removing terms unrelated to  $\Theta$ , we obtain the following optimization problem:

$$\min_{\Theta} \sum_{i=1}^n \sum_{j=1}^k w_{ij}^{(t)} \left( (u_{ij}^{(t+1)})^m \sum_{l=1}^L \alpha_l^{(t)} \|\tilde{\phi}_l(\mathbf{x}_i) - \boldsymbol{\theta}_{j,l}\|^2 \right). \quad (27)$$

Setting the derivative of Eq. (27) with respect to  $\boldsymbol{\theta}_{j,l}$  to zero, we obtain the update rule of  $\boldsymbol{\theta}_{j,l}$  as follows:

$$\boldsymbol{\theta}_{j,l}^{(t+1)} = \frac{\sum_{i=1}^n w_{ij}^{(t)} (u_{ij}^{(t+1)})^m \tilde{\phi}_l(\mathbf{x}_i)}{\sum_{i=1}^n w_{ij}^{(t)} (u_{ij}^{(t+1)})^m}. \quad (28)$$

**Step3:** Update  $\alpha$ . By substituting  $\Theta = \Theta^{(t+1)}$ ,  $\mathbf{U} = \mathbf{U}^{(t+1)}$  into Eq. (24) and removing terms unrelated to  $\alpha$ , we obtain the following optimization problem:

$$\min_{\alpha} \sum_{l=1}^L \alpha_l \sum_{i=1}^n \sum_{j=1}^k w_{ij}^{(t)} \tilde{d}_{ij,l}^{(t+1)} + \lambda \sum_{l=1}^L \alpha_l \log \alpha_l. \quad (29)$$

To minimize Eq. (29) in  $\alpha$ , we consider the Lagrangian

$$\begin{aligned} \mathcal{L}(\alpha, \beta) &= \sum_{l=1}^L \alpha_l \sum_{i=1}^n \sum_{j=1}^k w_{ij}^{(t)} \tilde{d}_{ij,l}^{(t+1)} \\ &+ \lambda \sum_{l=1}^L \alpha_l \log \alpha_l - \beta \left( \sum_{l=1}^L \alpha_l - 1 \right). \end{aligned} \quad (30)$$

By setting  $\frac{\partial \mathcal{L}}{\partial \alpha_l} = 0$  and combining with  $\sum_{l=1}^L \alpha_l = 1$ , we obtain the update rule of  $\alpha_l$  as follows:

$$\alpha_l^{(t+1)} = \frac{\exp \left( -\frac{1}{\lambda} \sum_{i=1}^n \sum_{j=1}^k w_{ij}^{(t)} \tilde{d}_{ij,l}^{(t+1)} \right)}{\sum_{l=1}^L \exp \left( -\frac{1}{\lambda} \sum_{i=1}^n \sum_{j=1}^k w_{ij}^{(t)} \tilde{d}_{ij,l}^{(t+1)} \right)}. \quad (31)$$

## Complexity Analysis

The computational complexity of IP-RFF-MKPKM primarily consists of the following main steps. For generating the RFF mapping of the  $L$  kernels, the complexity is  $\mathcal{O}(ndDL)$ . For updating the membership matrix  $\mathbf{U}$ , the complexity is  $\mathcal{O}(nkDL)$ . For updating the cluster centroids  $\Theta$ , the complexity is  $\mathcal{O}(nkDL)$ . For computing the distance between data points and cluster centroids, the complexity is  $\mathcal{O}(nkDL)$ . In sum, IP-RFF-MKPKM consumes a time complexity of  $\mathcal{O}(n(k+d)DL)$  in each iteration, greatly reducing the time complexity  $\mathcal{O}(n^2kL)$  of MKPKM.

## Experiment

In this section, we mainly discuss the experimental results of the multi-view clustering method IP-RFF-MKPKM due to space constraints. Single-view experiments and parameter settings are provided in the Appendix.

### Experimental Settings

**Datasets** To evaluate the effectiveness of the proposed RFF-MKPKM, we employ six single-view datasets derived from the first view of six multi-view datasets: Yale (Cai, He, and Han 2005), Caltech101-20 (Li et al. 2015), 100leaves(Wang, Yang, and Liu 2019), CIFAR10 (Krizhevsky, Hinton et al. 2009), YTF10 and YTF20 (Wolf, Hassner, and Maoz 2011). To evaluate the effectiveness of the proposed IP-RFF-MKPKM, we use seven widely used multi-view datasets: LGG (Network 2015), Caltech101-7 (Li et al. 2015), HW2 (Kevin Bache 2013), NUS-WIDE-SCENE (Chua et al. 2009), NUS-WIDE-OBJECT (Chua et al. 2009) and CIFAR10 (Krizhevsky, Hinton et al. 2009). Comprehensive details regarding these datasets are accessible in the appendix.

**Compared Methods** To contextualize the innovation of RFF-KPKM, we compare the proposed RFF-KPKM against four pivotal baselines: kernel  $k$ -means (Girolami 2002), spectral clustering (Von Luxburg 2007), power  $k$ -means (Xu and Lange 2019), and kernel power  $k$ -means (Paul et al. 2022). The proposed IP-RFF-MKPKM is evaluated against eight contemporary multi-view clustering (MVC) methods, including: MVASM (Han et al. 2020), MKPKM (Paul et al. 2022), OMVFC-LICAG (Zhang et al. 2024), LKRGDF (Chen et al. 2024), SVD-SMKKM (Liang et al. 2024b), INMKC (Feng et al. 2025b), SMKKM-UGF (Yang et al. 2025), MVHGB (Zhao et al. 2025).

**Evaluation** In all experiments, we use RFF of dimension  $\lceil 4 \log^3(2k) \rceil$  according to Theorem 3. For statistical robustness, every trial was repeated 20 times with average performance metrics. Clustering quality was evaluated through three standard criteria: Accuracy (ACC), Normalized Mutual Information (NMI), and Purity. The computational environment comprised an Intel Core i9-10900X CPU, 64GB RAM, and MATLAB 2020b (64-bit).

### Experimental Results

Table 1 shows the clustering outcomes of IP-RFF-MKPKM on the seven benchmark datasets, and Table 2 displays the results with and without the possibilistic method. Furthermore, Fig. 3 compares the running time of IP-RFF-MKPKM with other state-of-the-art methods. Analysis of the experimental findings yields the following conclusions:

1. Our proposed IP-RFF-MKPKM algorithm generally outperforms existing MKC methods on most datasets. In terms of ACC, IP-RFF-MKPKM surpasses the second-best algorithm on the LGG, Caltech101-7, HW2, NUS-WIDE-OBJECT, and CIFAR10 datasets by margins of 9.74%, 8.32%, 3.61%, 7.02%, 2.42% respectively.
2. When compared to the original MKPKM method, our approach generally achieves superior performance. On the datasets LGG, Caltech101-7, HW2, Caltech101-20,

NUS-WIDE-SCENE, IP-RFF-MKPKM consistently outperforms MKPKM by 22.85%, 21.92%, 3.61%, 9.68%, and 10.47% in terms of ACC. To demonstrate the effectiveness of the proposed possibilistic method, we conducted experiments on RFF-MKPKM by replacing  $\tilde{d}_{ij}$  in IP-RFF-MKPKM with  $d_{ij}$ . As shown in Table 2, the possibilistic method significantly improves the clustering performance of RFF-MKPKM.

3. As shown in Fig. 3, IP-RFF-MKPKM can run on the CIFAR dataset containing 60000 samples, while six methods cannot since memory limitation. When compared to the powerful and efficient MVC method, INMKC, IP-RFF-MKPKM surpasses it by 9.74%, 33.11%, 17.8%, 20.54%, 8.86%, and 7.27% in terms of ACC across all datasets. The results demonstrate the superior scalability and performance of our method.

### Convergence and Sensitivity Analysis

We conduct an experimental series elucidating IP-RFF-MKPKM's convergence characteristics. As depicted in Fig. 1a, the objective function value monotonically decreases per iteration, achieving convergence at approximately 30 iterations on Caltech101-7. We also examine the sensitivity of the parameters  $s_0$  and  $\lambda$ . As illustrated in Fig. 1b, the proposed method is not significantly affected by  $\lambda$  within  $[1, 1000]$ , and performs better when  $s_0$  is close to 15 on the Caltech101-7 dataset. The results show that the addition of an entropy regularization term facilitates learning smoother combination coefficients, and the adjustment of  $s_0$  helps to find the optimal global solution of IP-RFF-MKPKM.

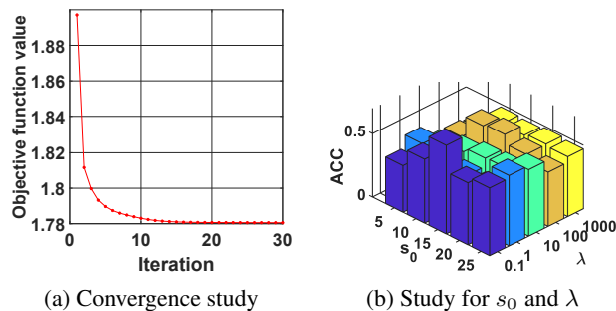


Figure 1: Convergence analysis and sensitivity analysis of  $s_0$  and  $\lambda$  of IP-RFF-MKPKM on Caltech101-7 dataset. Convergence and sensitivity studies on other benchmark datasets are given in the Appendix.

### Approximation Analysis

To evaluate the impact of RFF dimensions on clustering accuracy, we tested the performance of RFF-KPKM across dimensions ranging from 5 to 100 on six datasets. As shown in Fig. 2, below 40 RFF dimensions, dimensionality significantly impacts clustering accuracy (ACC); beyond 40 dimensions, ACC gains diminish and gradually approach stability. An RFF dimensionality of 20 to 40 represents the computational sweet spot, delivering  $\geq 85\%$  of the maximum achievable accuracy for the datasets.

Dataset	Metric	MVASM	MKPKM	OMVFC-LICAG	LKRGDF	SVD-SMKKM	SMKKM-UGF	MVHBG	INMKC	Proposed
LGG	ACC	59.25	58.05	57.30	43.45	69.29	50.94	61.42	71.16	<b>80.90</b>
	NMI	33.02	29.22	27.33	0.07	36.61	18.30	22.90	37.11	<b>45.12</b>
	Purity	51.23	62.17	59.93	50.19	69.29	55.81	61.80	71.16	<b>80.90</b>
Caltech101-7	ACC	61.29	47.69	41.99	52.17	51.42	44.52	52.24	36.50	<b>69.61</b>
	NMI	50.54	45.74	39.58	36.65	36.37	50.48	43.88	28.85	<b>51.63</b>
	Purity	82.89	83.24	81.61	84.33	83.38	83.72	84.80	80.94	<b>85.89</b>
HW2	ACC	73.23	86.54	72.75	57.00	62.5	78.57	62.6	72.35	<b>90.15</b>
	NMI	73.77	75.06	70.25	66.69	56.57	81.01	69.28	68.45	<b>81.56</b>
	Purity	65.03	85.12	74.70	64.45	65.90	81.37	67.50	74.80	<b>90.15</b>
Caltech101-20	ACC	34.93	44.51	38.60	39.10	43.42	37.21	<b>58.42</b>	33.65	54.19
	NMI	43.56	61.11	47.71	39.99	43.87	55.36	60.64	40.25	<b>61.61</b>
	Purity	52.27	76.15	68.11	65.21	67.73	73.31	74.73	64.63	<b>76.82</b>
NUS-WIDE-SCENE	ACC	11.22	11.61	9.26	14.68	17.12	11.53	20.19	9.74	<b>22.08</b>
	NMI	8.81	8.70	8.34	9.64	8.28	9.43	8.52	7.11	<b>10.00</b>
	Purity	17.15	28.89	30.60	33.11	33.80	<b>34.13</b>	30.60	32.94	31.99
NUS-WIDE-OBJECT	ACC	13.47	-	11.17	-	14.10	-	-	12.26	<b>21.12</b>
	NMI	12.40	-	8.13	-	10.50	-	-	9.27	<b>13.53</b>
	Purity	9.80	-	21.02	-	22.54	-	-	22.01	<b>24.30</b>
CIFAR10	ACC	-	-	-	-	25.46	-	-	20.61	<b>27.88</b>
	NMI	-	-	-	-	12.32	-	-	7.63	<b>14.91</b>
	Purity	-	-	-	-	25.78	-	-	21.88	<b>28.56</b>

Table 1: Clustering performance comparison of IP-RFF-MKPKM with eight baselines on seven benchmark datasets concerning clustering accuracy (ACC), normalized mutual information (NMI), and Purity. Maximum values are highlighted in boldface; the hyphen symbol (-) designates execution termination due to memory overallocation.

Metrics	Method	Datasets						
		LGG	Caltech101-7	HW2	Caltech101-20	NUS-WIDE-SCENE	NUS-WIDE-OBJECT	CIFAR10
ACC	RFF-MKPKM	62.55	61.67	83.00	53.39	12.41	13.95	26.08
	IP-RFF-MKPKM	<b>80.90</b>	<b>69.61</b>	<b>90.15</b>	<b>54.19</b>	<b>22.08</b>	<b>21.12</b>	<b>27.88</b>

Table 2: ACC comparison with and without the possibilistic method.

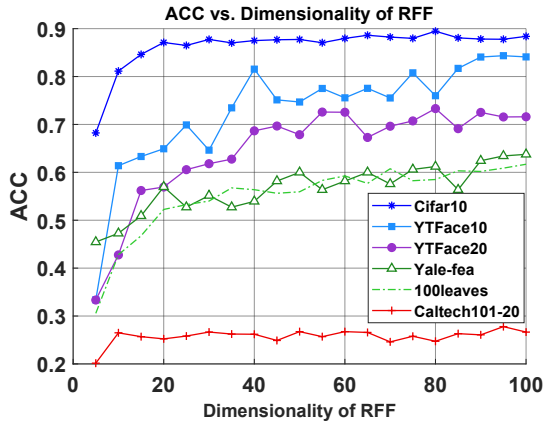


Figure 2: Impact of RFF dimension on RFF-KPKM clustering accuracy on six benchmark datasets. The RFF dimension varies from 5 to 100.

## Conclusion

This work utilizes random Fourier features (RFF) and a probabilistic method to address the time complexity and noise issue of KPKM. We demonstrate the effectiveness of RFF-KPKM by establishing its excess risk bound, strong consistency, and relative error, which establishes the first theoretical guarantees for the combination of RFF and KPKM. Building on

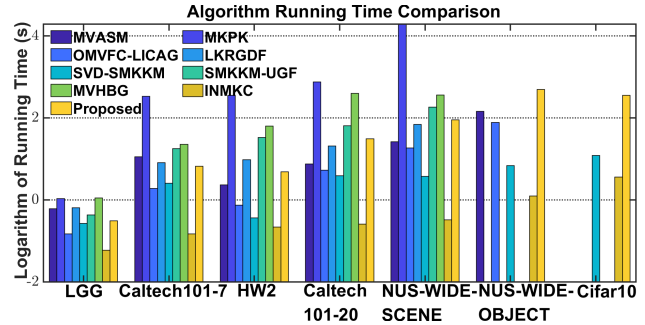


Figure 3: Logarithmic running time comparison of IP-RFF-MKPKM with eight benchmark methods on seven benchmark datasets. Bar absence denotes an out-of-memory runtime exception during method execution.

this, we design IP-RFF-MKPKM by integrating possibilistic memberships and fuzzy memberships to redefine cluster affiliations, thereby eliminating outlier-induced centroid drift and extending scalability to multi-view settings. We conduct experiments on six benchmark single-view datasets and seven benchmark multi-view datasets and compare the proposed methods with state-of-the-art baseline methods. Experimental results demonstrate the effectiveness and efficiency of the proposed RFF-KPKM and IP-RFF-MKPKM.

## Acknowledgments

This work is supported by the National Science Fund for Distinguished Young Scholars of China (No. 62325604), and the National Natural Science Foundation of China (No. 62441618, 62276271, 62201600, 62201604, 62572480 and 62506369).

## References

- Aronszajn, N. 1950. Theory of reproducing kernels. *Transactions of the American mathematical society*, 68(3): 337–404.
- Bezdek, J. C.; Ehrlich, R.; and Full, W. 1984. FCM: The fuzzy c-means clustering algorithm. *Computers & geosciences*, 10(2-3): 191–203.
- Cai, D.; He, X.; and Han, J. 2005. Using graph model for face analysis. *Technical Report*.
- Chakraborty, S.; and Das, S. 2020. Detecting meaningful clusters from high-dimensional data: A strongly consistent sparse center-based clustering approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(6): 2894–2908.
- Chen, Y.; Du, L.; Zhou, P.; Duan, L.; and Qian, Y. 2024. Multiple kernel clustering with local kernel reconstruction and global heat diffusion. *Information Fusion*, 105: 102219.
- Cheng, K.; Jiang, S. H.-C.; Wei, L.; and Wei, Z. 2023. On The Relative Error of Random Fourier Features for Preserving Kernel Distance. In *ICLR*.
- Chitta, R.; Jin, R.; and Jain, A. K. 2012. Efficient kernel clustering using random fourier features. In *2012 IEEE 12th international conference on data mining*, 161–170. IEEE.
- Chua, T.-S.; Tang, J.; Hong, R.; Li, H.; Luo, Z.; and Zheng, Y. 2009. Nus-wide: a real-world web image database from national university of singapore. In *Proceedings of the ACM international conference on image and video retrieval*, 1–9.
- Feng, Y.; Liang, W.; Wan, X.; Liu, J.; Li, M.; and Liu, X. 2025a. Incremental Multi-View Clustering: Exploring Stream-View Correlations to Learn Consistency and Diversity. *IEEE Transactions on Knowledge and Data Engineering*.
- Feng, Y.; Liang, W.; Wan, X.; Liu, J.; Liu, S.; Qu, Q.; Guan, R.; Xu, H.; and Liu, X. 2025b. Incremental Nyström-based Multiple Kernel Clustering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 16613–16621.
- Filippone, M.; Camastra, F.; Masulli, F.; and Rovetta, S. 2008. A survey of kernel and spectral methods for clustering. *Pattern recognition*, 41(1): 176–190.
- Girolami, M. 2002. Mercer kernel-based clustering in feature space. *IEEE transactions on neural networks*, 13(3): 780–784.
- Gönen, M.; and Alpaydm, E. 2011. Multiple kernel learning algorithms. *The Journal of Machine Learning Research*, 12: 2211–2268.
- Guan, R.; Li, J.; Wang, S.; Tu, W.; Li, M.; Zhu, E.; Liu, X.; and Chen, P. 2025a. Multi-view Graph Clustering with Dual Relation Optimization for Remote Sensing Data. In *Proceedings of the 33rd ACM International Conference on Multimedia*, 7346–7355.
- Guan, R.; Liu, T.; Tu, W.; Tang, C.; Luo, W.; and Liu, X. 2025b. Sampling Enhanced Contrastive Multi-View Remote Sensing Data Clustering with Long-Short Range Information Mining. *IEEE Transactions on Knowledge and Data Engineering*, 1–15.
- Guan, R.; Tu, W.; Wang, S.; Liu, J.; Hu, D.; Tang, C.; Feng, Y.; Li, J.; Xiao, B.; and Liu, X. 2025c. Structure-Adaptive Multi-View Graph Clustering for Remote Sensing Data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 16933–16941.
- Han, J.; Xu, J.; Nie, F.; and Li, X. 2020. Multi-view k-means clustering with adaptive sparse memberships and weight allocation. *IEEE Transactions on Knowledge and Data Engineering*, 34(2): 816–827.
- Jain, A. K. 2010. Data clustering: 50 years beyond K-means. *Pattern recognition letters*, 31(8): 651–666.
- Kevin Bache, M. L. 2013. UCI machine learning repository.
- Kou, G.; Peng, Y.; and Wang, G. 2014. Evaluation of clustering algorithms for financial risk analysis using MCDM methods. *Information sciences*, 275: 1–12.
- Krishnapuram, R.; and Keller, J. M. 1993. A possibilistic approach to clustering. *IEEE transactions on fuzzy systems*, 1(2): 98–110.
- Krizhevsky, A.; Hinton, G.; et al. 2009. Learning multiple layers of features from tiny images.
- Li, T.; Kou, G.; Peng, Y.; and Yu, P. S. 2021. An integrated cluster detection, optimization, and interpretation approach for financial data. *IEEE transactions on cybernetics*, 52(12): 13848–13861.
- Li, Y.; Nie, F.; Huang, H.; and Huang, J. 2015. Large-scale multi-view spectral clustering via bipartite graph. In *Proceedings of the AAAI conference on artificial intelligence*, volume 29.
- Liang, K.; Meng, L.; Li, H.; Liu, M.; Wang, S.; Zhou, S.; Liu, X.; and He, K. 2024a. MGKsite: Multi-Modal Knowledge-Driven Site Selection via Intra and Inter-Modal Graph Fusion. *IEEE Transactions on Multimedia*.
- Liang, K.; Meng, L.; Li, H.; Wang, J.; Lan, L.; Li, M.; Liu, X.; and Wang, H. 2025. From Concrete to Abstract: Multi-view Clustering on Relational Knowledge. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1–18.
- Liang, W.; Tang, C.; Liu, X.; Liu, Y.; Liu, J.; Zhu, E.; and He, K. 2024b. On the consistency and large-scale extension of multiple kernel clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(10): 6935–6947.
- Mairal, J. 2013. Stochastic majorization-minimization algorithms for large-scale optimization. *Advances in Neural Information Processing Systems*, 26.
- Malliaros, F. D.; and Vazirgiannis, M. 2013. Clustering and community detection in directed networks: A survey. *Physics reports*, 533(4): 95–142.
- Network, C. G. A. R. 2015. Comprehensive, integrative genomic analysis of diffuse lower-grade gliomas. *New England Journal of Medicine*, 372(26): 2481–2498.

- Paul, D.; Chakraborty, S.; Das, S.; and Xu, J. 2022. Implicit annealing in kernel spaces: A strongly consistent clustering approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(5): 5862–5871.
- Pelckmans, K.; De Brabanter, J.; Suykens, J. A.; and De Moor, B. 2005. Convex clustering shrinkage. In *PASCAL workshop on statistics and optimization of clustering workshop*, volume 1524.
- Pollard, D. 1981. Strong consistency of k-means clustering. *The annals of statistics*, 135–140.
- Rahimi, A.; and Recht, B. 2007. Random features for large-scale kernel machines. *Advances in neural information processing systems*, 20.
- Sun, D.; Toh, K.-C.; and Yuan, Y. 2021. Convex clustering: Model, theoretical guarantee and efficient algorithm. *Journal of Machine Learning Research*, 22(9): 1–32.
- Tang, R.; Fong, S.; Yang, X.-S.; and Deb, S. 2012. Integrating nature-inspired optimization algorithms to K-means clustering. In *Seventh International Conference on Digital Information Management (ICDIM 2012)*, 116–123. IEEE.
- Von Luxburg, U. 2007. A tutorial on spectral clustering. *Statistics and computing*, 17: 395–416.
- Wang, F.; Jin, J.; Hu, J.; Liu, S.; Yang, X.; Wang, S.; Liu, X.; and Zhu, E. 2024a. Evaluate then Cooperate: Shapley-based View Cooperation Enhancement for Multi-view Clustering. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.
- Wang, F.; Wang, S.; Jin, J.; Dong, Z.; Yang, X.; Feng, Y.; Zhu, X.; Liu, T.; Liu, X.; and Zhu, E. 2024b. View Gap Matters: Cross-view Topology and Information Decoupling for Multi-view Clustering. In *ACM Multimedia 2024*.
- Wang, H.; Yang, Y.; and Liu, B. 2019. GMC: Graph-based multi-view clustering. *IEEE Transactions on Knowledge and Data Engineering*, 32(6): 1116–1129.
- Wang, J.; Li, Z.; Tang, C.; Liu, S.; Wan, X.; and Liu, X. 2024c. Multiple kernel clustering with adaptive multi-scale partition selection. *IEEE Transactions on Knowledge and Data Engineering*, 36(11): 6641–6652.
- Wang, J.; Tang, C.; Wan, Z.; Zhang, W.; Sun, K.; and Zomaya, A. Y. 2023a. Efficient and effective one-step multiview clustering. *IEEE Transactions on Neural Networks and Learning Systems*, 35(9): 12224–12235.
- Wang, J.; Tang, C.; Zheng, X.; Liu, X.; Zhang, W.; Zhu, E.; and Zhu, X. 2023b. Fast approximated multiple kernel k-means. *IEEE Transactions on Knowledge and Data Engineering*, 36(11): 6171–6180.
- Williams, C.; and Seeger, M. 2000. Using the Nyström method to speed up kernel machines. *Advances in neural information processing systems*, 13.
- Wolf, L.; Hassner, T.; and Maoz, I. 2011. Face recognition in unconstrained videos with matched background similarity. In *CVPR 2011*, 529–534. IEEE.
- Xu, J.; and Lange, K. 2019. Power k-means clustering. In *International conference on machine learning*, 6921–6931. PMLR.
- Yang, W.; Tang, C.; Liu, X.; Yue, G.; Liu, Y.; Zhang, C.; and Zhu, E. 2025. Smooth Multiple Kernel k-Means via Underlying Graph Filtering. *IEEE Transactions on Neural Networks and Learning Systems*.
- Yin, R.; Liu, Y.; Wang, W.; and Meng, D. 2022. Randomized Sketches for Clustering: Fast and Optimal Kernel k-Means. *Advances in Neural Information Processing Systems*, 35: 6424–6436.
- Yu, H.; Liang, W.; LIANG, K.; Liu, S.; Liu, M.; and Liu, X. 2025. On the Adversarial Robustness of Multi-Kernel Clustering. In *ICML*.
- Zhang, B.; Hsu, M.; and Dayal, U. 1999. K-harmonic means—a data clustering algorithm. *Hewlett-Packard Labs Technical Report HPL-1999-124*, 55.
- Zhang, C.; Chen, L.; Shi, Z.; and Ding, W. 2024. Latent information-guided one-step multi-view fuzzy clustering based on cross-view anchor graph. *Information Fusion*, 102: 102025.
- Zhang, J.-S.; and Leung, Y.-W. 2004. Improved possibilistic c-means clustering algorithms. *IEEE transactions on fuzzy systems*, 12(2): 209–217.
- Zhao, Z.; Wang, T.; Xin, H.; Wang, R.; and Nie, F. 2025. Multi-view clustering via high-order bipartite graph fusion. *Information Fusion*, 113: 102630.
- Zhou, T.; Dong, Z.; Wang, S.; Liang, K.; Li, M.; Liu, X.; Zhu, E.; and Dong, X. 2025. DPFMVC: Dynamic Progressive Fusion for Multi-view Clustering. In *Proceedings of the 33rd ACM International Conference on Multimedia*, MM '25, 1102–1111. New York, NY, USA: Association for Computing Machinery. ISBN 9798400720352.
- Zou, Q.; Lin, G.; Jiang, X.; Liu, X.; and Zeng, X. 2020. Sequence clustering in bioinformatics: an empirical study. *Briefings in bioinformatics*, 21(1): 1–10.