

Article

Fast and Randomized Multiple Kernel Discriminant Analysis for Bird Recognition

Ke Li ^{1,*}, Binghong Li ², Chao Wang ² and Xiaohui Wang ²¹ School of Medical Information and Engineering, Xuzhou Medical University, Xuzhou 221004, China² Air Force Logistics University, Xuzhou 221116, China; libinghong2026@163.com (B.L.); 13814426352@163.com (C.W.); 15950685505@163.com (X.W.)

* Correspondence: 100002023041@xzhmu.edu.cn

Abstract

In order to improve the bird strike avoidance management at airports, and to realize the linkage of bird detection radar and a variety of bird repellent equipment, intelligent bird repellent decision methods have attracted widespread attention. In this paper, we are committed to exploring bird target recognition approaches, so as to prepare for the subsequent bird repellent decision methods. Kernel methods are well known to be effective in dealing with nonlinear machine learning problems. However, the performance of kernel methods heavily relies on the choice of kernel parameters. To mitigate this problem, we add the ideas of multiple kernels and randomized techniques to kernel discriminant analysis (KDA), and propose a randomized and fast multiple kernel discriminant analysis (RM-KDA) method with applications to bird recognition. The main contributions of our work are two-fold. First, we propose an innovative multiple kernel discriminant analysis (M-KDA) framework to effectively solve recognition tasks with nonlinear data structures. Second, we develop a fast and efficient randomized algorithm to solve the M-KDA model without explicitly forming and storing the base kernel matrices and the ensemble kernel matrix in advance, which greatly reduces the computation and storage requirements. The extensive experiments on benchmark recognition databases demonstrate the effectiveness of the proposed algorithm.

Keywords: multiple kernel; randomized algorithm; linear discriminant analysis; feature extraction; bird recognition

MSC: 68T10; 68W20; 62H30; 46E22



Academic Editor: Jonathan Blackledge

Received: 17 April 2026

Revised: 11 July 2026

Accepted: 21 July 2026

Published: 3 August 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

Birdstrike is an accident that affects flight safety when an aircraft is struck by a bird during takeoff, climb, cruise or landing. As aircraft continue to grow in number, size and speed, these increase the likelihood of bird strikes in the air [1–3]. According to the International Civil Aviation Organization, there are more than 20,000 bird strikes each year. It is estimated that the global annual cost of bird collisions is more than 10 billion dollars, and bird strikes have become one of the most dangerous factors threatening aviation safety [4,5].

Bird detection and tracking are fundamental components of airport bird-strike prevention systems [3,5]. Bird detection aims to determine the presence of bird targets, while tracking focuses on estimating their locations, trajectories, and movement patterns. However, these approaches generally cannot identify the species or categories of the observed

birds. In practice, different bird species may pose substantially different levels of risk to aircraft due to differences in body size, flocking behavior, flight altitude, and activity characteristics. Therefore, bird recognition serves as a complementary and essential step beyond detection and tracking, providing critical information for bird-strike risk assessment, ecological management, and intelligent bird-repellent decision making [6,7]. Furthermore, the research of bird recognition technology also plays an important role in ecological protection, scientific research and nature education [6,8].

Recently, deep learning methods have achieved remarkable success in bird recognition tasks [7]. However, these methods often require many labeled training data, substantial computational resources, and extensive parameter tuning [9]. In contrast, kernel-based discriminative learning methods provide a mathematically interpretable framework and can be trained with relatively lower computational cost [10–12]. Furthermore, randomized matrix approximation techniques make large-scale kernel learning computationally feasible [13–15]. Motivated by these observations, we develop a fast and randomized multiple kernel discriminant analysis (RM-KDA) method that focuses on efficient nonlinear discriminative learning through multiple kernels and randomized matrix approximation. It is worth noting that RM-KDA should be regarded as complementary to deep learning rather than a replacement for it [12], since it can be directly applied to deep feature representations extracted by modern neural networks. Therefore, the proposed method provides a computationally efficient and mathematically interpretable framework for large-scale discriminative learning and classification.

The main contributions of our work are two-fold.

1. First, many classification tasks involve complex, nonlinear patterns that cannot be fully captured by standard linear discriminant analysis (LDA). By integrating multiple kernel learning (MKL) into the LDA, the framework effectively combines multiple feature representations to enhance discriminative power while preserving the theoretical foundation of LDA. This approach is particularly valuable for applications such as fine-grained recognition, where subtle inter-class variations and nonlinear feature interactions are prevalent.

Building on this idea, we propose a multiple kernel discriminant analysis (M-KDA) framework to effectively address recognition tasks with nonlinear data structures. The M-KDA model leverages multiple kernels to capture complementary information across features, enabling more robust and accurate classification compared to standard LDA.

2. Second, despite the success of multiple kernel methods in various applications [10–12], to the best of our knowledge, most of the existing approaches are only suitable for small-sample problems. This is because these methods typically require explicitly constructing and storing all base kernel matrices and the ensemble kernel matrix in advance [11,16–22], resulting in substantial computational and memory demands. For large-scale datasets, this requirement can become a critical limitation, making such approaches computationally prohibitive or even infeasible. Therefore, it is essential to develop new insights into multiple kernel learning and to substantially improve its numerical efficiency.

To address the challenges, we propose a fast and efficient randomized algorithm for solving the M-KDA model using a generalized Nystrom approximation. The algorithm avoids explicitly constructing and storing individual base kernel matrices as well as the ensemble kernel matrix in advance, thereby significantly reducing both computational complexity and memory consumption. Extensive experiments on benchmark datasets demonstrate that the proposed randomized M-KDA (RM-KDA) achieves competitive recognition performance while offering substantial improvements in efficiency.

The structure of this paper is as follows. In Section 2, we briefly review the original linear discriminant analysis (LDA) algorithm and some extensions. In Section 3, to reduce the computational overhead, a fast and randomized algorithm for multiple kernel discriminant analysis (RM-KDA) is proposed. Finally, in Section 4, we perform numerical experiments on benchmark recognition databases to demonstrate the superior performance of our RM-KDA algorithm. Some conclusions are drawn in Section 5. We list some notations used in this paper in Table 1.

Table 1. Some notations used in this paper.

Notations	Description
d, n	The data dimension, and the number of training samples
k	The number of classes that the samples belong to
s	The discriminant space dimensionality, i.e., the reduced dimension
N	The number of total samples, including training and test samples
X	$X = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n] \in \mathbb{R}^{d \times n}$ is the training data matrix
$m, \{K_i\}_{i=1}^m$	The number of base kernel matrices, and base kernel matrices
β	$\beta = [\beta_1, \beta_2, \dots, \beta_m]^T \in \mathbb{R}^m$ stands for combination coefficients of base kernel matrices
$\mathbb{K}$	The ensemble kernel $\mathbb{K} = \sum_{i=1}^m \beta_i K_i$
I_i	The identity matrix of dimension i
$G^T, G^\dagger$	Transpose of the matrix G and Moore–Penrose pseudoinverse of G
$\text{rank}(G), \text{tr}(G)$	Rank of the matrix G and trace of G
$\ \cdot\ _2, \ \cdot\ _F$	2-norm and Frobenius norm

2. Preliminaries

Recently, with the development of technology, bird recognition technology is playing an increasingly important role in airport bird repellent work [6,7]. In this paper, we are committed to exploring bird target recognition approaches, so as to prepare for the subsequent bird repellent decision methods. It is well known that dimensionality reduction has been a key problem in many fields of information processing, such as information retrieval, data mining and pattern recognition. In essence, dimensionality reduction is a method which projects high-dimensional data into a low-dimensional data space, and the optimization problem involved is a trace-ratio problem. However, many traditional methods can only handle small dimensionality. The effectiveness and efficiency of these traditional methods drop exponentially as the dimensionality increases, which is commonly referred to as the “curse of dimensionality” [23].

The most typical method of dimensionality reduction is linear discriminant analysis (LDA) [24–26], and its main idea is to search an orthogonal matrix maximizing between-class scatter while minimizing within-class scatter. Next, let us introduce LDA in detail. Denote by $X = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n] \in \mathbb{R}^{d \times n}$ a set of d -dimensional training data, assuming that the data set is partitioned into k classes as $X = [X_1, X_2, \dots, X_k]$, where X_j is the j -th class with n_j samples. Let $\mathbf{c}_j$ and $\mathbf{c}$ be the centroid vector of the j -th class data X_j and the global centroid vector of all training data, respectively. Set $\mathbf{1}_{n_j} = [1, 1, \dots, 1]^T \in \mathbb{R}^{n_j}$ and $H_W = [X_1 - \mathbf{c}_1 \mathbf{1}_{n_1}^T, X_2 - \mathbf{c}_2 \mathbf{1}_{n_2}^T, \dots, X_k - \mathbf{c}_k \mathbf{1}_{n_k}^T] \in \mathbb{R}^{d \times n}$, and then the within-class scatter matrix is

$$S_W = \sum_{j=1}^k \sum_{\mathbf{x}_i \in X_j} (\mathbf{x}_i - \mathbf{c}_j)(\mathbf{x}_i - \mathbf{c}_j)^T = H_W H_W^T.$$

Then, let $H_B = [\sqrt{n_1}(\mathbf{c}_1 - \mathbf{c}), \sqrt{n_2}(\mathbf{c}_2 - \mathbf{c}), \dots, \sqrt{n_k}(\mathbf{c}_k - \mathbf{c})] \in \mathbb{R}^{d \times k}$, so the between-class scatter matrix is defined as

$$S_B = \sum_{j=1}^k n_j (\mathbf{c}_j - \mathbf{c})(\mathbf{c}_j - \mathbf{c})^T = H_B H_B^T.$$

Finally, the total scatter matrix is as follows:

$$S_T = \sum_{j=1}^n (\mathbf{x}_j - \mathbf{c})(\mathbf{x}_j - \mathbf{c})^T = H_T H_T^T,$$

where $H_T = [\mathbf{x}_1 - \mathbf{c}, \mathbf{x}_2 - \mathbf{c}, \dots, \mathbf{x}_n - \mathbf{c}] \in \mathbb{R}^{d \times n}$.

Let $A \in \mathbb{R}^{d \times s}$ denote the projection matrix consisting of s discriminant projection vectors; then, the fundamental problem with the LDA method reduces to the trace-ratio problem

$$\max_{\substack{A \in \mathbb{R}^{d \times s} \\ A^T A = I_s}} \frac{\text{tr}(A^T S_B A)}{\text{tr}(A^T S_W A)}, \quad (1)$$

where $s \ll d$ represents the dimension of projection subspace. Using the projection matrix A , an input sample $\mathbf{x}_i \in \mathbb{R}^d$ can be projected from the original high-dimensional space into a s -dimensional subspace according to

$$\mathbf{y}_i = A^T \mathbf{x}_i,$$

where $\mathbf{y}_i \in \mathbb{R}^s$ denotes the low-dimensional representation of $\mathbf{x}_i$.

For the sake of simplification, the original trace-ratio problem (1) is often replaced by a ratio-trace problem [27–29]

$$\max_{\substack{A \in \mathbb{R}^{d \times s} \\ A^T A = I_s}} \text{tr}((A^T S_W A)^{-1} (A^T S_B A)), \quad (2)$$

whose solution can be obtained by the generalized eigenvalue problem

$$S_B \mathbf{a} = \lambda S_W \mathbf{a}. \quad (3)$$

Obviously, S_W, S_B, S_T are symmetric positive semidefinite matrices; hence, the “singularity problem” may occur. Fortunately, exponential discriminant analysis [30,31], regularized linear discriminant analysis (RLDA) [32], LDA/QR [33], generalizing discriminant analysis using generalized singular value decomposition (LDA/GSVD) [25,34], and linear algorithms based on principal component analysis (PCA) [35–38] have been proposed to overcome this defect. Further, relatively mature research exists on the ratio-trace problem with high-dimensional large samples. But there is no good scheme for the trace-ratio problem with high-dimensional large samples. In fact, the ratio-trace problem is not equivalent to the trace-ratio problem. Moreover, some results have been shown that the recognition rate of the algorithms based on trace-ratio models is better than that of ratio-trace models [39,40], which has caused a large number of scholars to turn to the trace-ratio problem.

The well-known LDA method is a classical linear method and fails for nonlinear problems. To deal with this limitation, a nonlinear extension through the “kernel trick” of LDA, known as kernel discriminant analysis (KDA), has been proposed [23,41,42]. The main idea of the kernel method is to map the input data to a feature space via a nonlinear mapping, where the inner products in the feature space can be computed by a kernel function without knowing the nonlinear mapping explicitly. When the data are highly nonlinearly distributed, KDA can achieve better numerical performance than LDA [23].

The mathematical framework of KDA is as follows. For a chosen nonlinear mapping ϕ ,

$$\phi: \mathbb{R}^d \rightarrow \mathcal{F},$$

an inner product in the feature space $\mathcal{F}$ can be obtained in the following way:

$$\langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle = \kappa(\mathbf{x}_i, \mathbf{x}_j)$$

where $\kappa(\cdot, \cdot)$ is a positive semi-definite kernel function. Let S_W^ϕ , S_B^ϕ and S_T^ϕ denote the between-class, within-class and total scatter matrices in the feature space $\mathcal{F}$, respectively. We have

$$S_W^\phi = \sum_{j=1}^k \sum_{\mathbf{x}_i \in X_j} (\phi(\mathbf{x}_i) - \mathbf{c}_\phi^{(j)}) (\phi(\mathbf{x}_i) - \mathbf{c}_\phi^{(j)})^T \in \mathbb{R}^{|\mathcal{F}| \times |\mathcal{F}|},$$

$$S_B^\phi = \sum_{j=1}^k n_j (\mathbf{c}_\phi^{(j)} - \mathbf{c}_\phi) (\mathbf{c}_\phi^{(j)} - \mathbf{c}_\phi)^T \in \mathbb{R}^{|\mathcal{F}| \times |\mathcal{F}|},$$

$$S_T^\phi = \sum_{j=1}^n (\phi(\mathbf{x}_j) - \mathbf{c}_\phi) (\phi(\mathbf{x}_j) - \mathbf{c}_\phi)^T \in \mathbb{R}^{|\mathcal{F}| \times |\mathcal{F}|},$$

where $\mathbf{c}_\phi^{(j)}$ and $\mathbf{c}_\phi$ are the centroids of the j -th class and the global centroid in the feature space $\mathcal{F}$, respectively. Denote by $V = [\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_s] \in \mathbb{R}^{|\mathcal{F}| \times s}$ the projection matrix in the feature space; then, the corresponding objective function (1) is

$$V_{opt} = \operatorname{argmax}_{\substack{V \in \mathbb{R}^{|\mathcal{F}| \times s} \\ V^T V = I_s}} \frac{\operatorname{tr}(V^T S_B^\phi V)}{\operatorname{tr}(V^T S_W^\phi V)}, \quad (4)$$

which can be solved by the eigenproblem

$$S_B^\phi \mathbf{v} = \lambda S_W^\phi \mathbf{v}. \quad (5)$$

However, solving (5) directly is impractical since $|\mathcal{F}|$ is very large or even infinite in practice.

Fortunately, according to Representer Theorem [43–45], there exist coefficients $\{\alpha_i\}_{i=1}^n$ such that $\mathbf{v}_j = \sum_{i=1}^n \alpha_{ij} \phi(\mathbf{x}_i)$, i.e.,

$$V = \Phi A,$$

where $A = [\alpha_1, \alpha_2, \dots, \alpha_s] \in \mathbb{R}^{n \times s}$ with $\alpha_j = [\alpha_{1j}, \alpha_{2j}, \dots, \alpha_{nj}]^T$, $j = 1, 2, \dots, s$, and $\Phi = [\phi(\mathbf{x}_1), \phi(\mathbf{x}_2), \dots, \phi(\mathbf{x}_n)] \in \mathbb{R}^{|\mathcal{F}| \times n}$. Moreover, denote by $W = [w_{ij}] \in \mathbb{R}^{n \times n}$ the affinity matrix whose entry w_{ij} measures the similarity between the i -th and j -th training samples [11], and define each w_{ij} as a function of the label vector $\mathbf{g}$ in the following manner:

$$w_{ij} = \begin{cases} \frac{1}{n_{g_i}}, & \text{if } g_i = g_j, \\ 0, & \text{otherwise,} \end{cases}$$

where g_i is the label of the i -th sample, and n_{g_i} is the number of samples in the g_i -th class [11,18,20]. Then, it follows from [43] that (4) is equivalent to

$$A = \operatorname{argmax}_{\substack{A \in \mathbb{R}^{n \times s} \\ A^T A = I_s}} \frac{\operatorname{tr}(A^T K W K A)}{\operatorname{tr}(A^T K K A)}, \quad (6)$$

and the corresponding eigenproblem is

$$K W K \alpha_j = \lambda_j K K \alpha_j, \quad (7)$$

where $K \in \mathbb{R}^{n \times n}$ is the kernel matrix with the ij -th element $K_{ij} = \kappa(\mathbf{x}_i, \mathbf{x}_j)$.

Most kernel approaches, such as KDA, are computationally very expensive. It is because kernel versions need to compute the kernel matrix $K \in \mathbb{R}^{n \times n}$ explicitly, where n is the training set cardinality. However, since the computational complexities and the storage requirements are $\mathcal{O}(n^3)$ and $\mathcal{O}(n^2)$ [15,46,47], respectively, KDA as mentioned above will become computationally intractable as n is large. On the other hand, it is well known that numerical performance of the kernel-based methods closely relies on the choice of kernel parameters [48–50]. However, for a particular machine learning task, the most suitable kernel is generally not so easy to determine in advance [22,49]. Hence, choosing the proper kernel is a challenging task that requires careful consideration. Fortunately, multiple kernel learning (MKL) can address this shortcoming to some extent. Its purpose is to learn a new kernel function $\mathbb{K} = \sum_{i=1}^m \beta_i K_i$ from a family of given kernels $K_1, K_2, \dots, K_m$ with kernel weight coefficients $\beta = [\beta_1, \beta_2, \dots, \beta_m]^T \in \mathbb{R}^m$ [10,51,52].

However, to the best of our knowledge, there are few algorithms that provide the version of multiple kernels for KDA. To fill this gap, on the one hand, we propose the idea of multiple kernels for LDA, i.e., the multiple kernel discriminant analysis (M-KDA) model. On the other hand, most multiple kernel approaches need to explicitly form and store all the base kernel matrices and an ensemble kernel matrix in advance [11,16–22], which requires a huge amount of computation and storage. For large-scale datasets, this problem will become a fatal flaw for multiple kernel approaches. Thus, we propose a fast and randomized multiple kernel discriminant analysis algorithm (RM-KDA) that avoids explicitly forming and storing the base kernel matrices.

3. A Fast and Randomized Multiple Kernel Discriminant Analysis Algorithm

Multiple-kernelization in LDA is accomplished in a similar way to that in KDA, but with the key difference of using multiple kernels $\{K_p\}_{p=1}^m$. Next, we propose the randomized multiple kernel discriminant analysis (RM-KDA) algorithm in more detail.

3.1. The RM-KDA Algorithm

Denote by $X = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n] \in \mathbb{R}^{d \times n}$ a set of d -dimensional training data, and let $W = [w_{ij}] \in \mathbb{R}^{n \times n}$ be the corresponding affinity matrix. Suppose we have a set of base kernel functions $\{\kappa_p\}_{p=1}^m$ or base kernel matrices $\{K_p\}_{p=1}^m$, where $K_p(i, j) = \kappa_p(\mathbf{x}_i, \mathbf{x}_j)$; then, an ensemble kernel function κ or an ensemble kernel matrix $\mathbb{K}$ is defined by

$$\kappa(\mathbf{x}_i, \mathbf{x}_j) = \sum_{p=1}^m \beta_p \kappa_p(\mathbf{x}_i, \mathbf{x}_j), \quad \text{where } \sum_{p=1}^m \beta_p = 1, \quad \beta_p \geq 0, \quad (8)$$

and

$$\mathbb{K} = \sum_{p=1}^m \beta_p K_p, \quad \text{where } \sum_{p=1}^m \beta_p = 1, \quad \beta_p \geq 0, \quad (9)$$

respectively.

For simplicity and computational efficiency, the ensemble kernel $\mathbb{K}$ in this paper is constructed as a linear combination of equal-weighted base kernels. Instead of learning the kernel weights adaptively, equal weights are adopted to avoid the additional optimization procedure required for kernel weight estimation, thereby reducing the computational cost. Therefore, this paper focuses on the integration of Nyström-based randomized approximation into the multiple kernel discriminant analysis framework, while leaving

adaptive kernel weight optimization as a topic for future investigation. Finally, M-KDA is defined as follows:

$$\mathbb{A} = \underset{\substack{\mathbb{A} \in \mathbb{R}^{n \times s} \\ \mathbb{A}^T \mathbb{A} = I_s}}{\operatorname{argmax}} \frac{\operatorname{tr}(\mathbb{A}^T \mathbb{K} \mathbb{W} \mathbb{K} \mathbb{A})}{\operatorname{tr}(\mathbb{A}^T \mathbb{K} \mathbb{K} \mathbb{A})}, \quad (10)$$

Notice that the optimization problem (10) is a trace-ratio problem. However, the trace-ratio problem is difficult to solve [27,53], and one often solves the ratio-trace problem instead, which reduces to a generalized eigenproblem. Consequently, the columns of the optimal $\mathbb{A} = [\tilde{\alpha}_1, \tilde{\alpha}_2, \dots, \tilde{\alpha}_s] \in \mathbb{R}^{n \times s}$ are the eigenvectors corresponding to the first s largest eigenvalues in the generalized eigenproblem

$$\mathbb{K} \mathbb{W} \mathbb{K} \tilde{\alpha} = \lambda \mathbb{K} \mathbb{K} \tilde{\alpha}. \quad (11)$$

Remark 1. It should be noted that the generalized eigenvalue problem in (11) is obtained through the conventional ratio-trace relaxation of the original trace-ratio objective rather than an exact equivalent transformation. This relaxation has been widely adopted in KDA-based methods [23,41,42,54,55] due to its computational efficiency and favorable empirical performance. Furthermore, since the ensemble kernel matrix $\mathbb{K} \in \mathbb{R}^{n \times n}$ may be rank-deficient when $n \gg d$, the proposed method will perform discriminant analysis using Nyström low-rank approximation and employ the Moore–Penrose pseudoinverse whenever necessary, thereby improving numerical stability while preserving most of the discriminative information.

Next, we adopt two steps to efficiently solve $\{\tilde{\alpha}_i\}_{i=1}^s$:

1. Solve the generalized eigenproblem

$$\mathbb{W} \mathbf{y} = \lambda \mathbf{y} \quad (12)$$

to obtain the eigenvectors $\{\mathbf{y}_i\}_{i=1}^s$ corresponding to the largest s eigenvalues. Cai et al. in [41] showed how this eigenproblem (12) is trivial and the eigenvectors can be directly obtained with a graph matrix $\mathbb{W}$. Follow the strategy in [41], and denote $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_s] \in \mathbb{R}^{n \times s}$.

2. With the matrix $\mathbf{Y} \in \mathbb{R}^{n \times s}$ at hand, we find $\mathbb{A} = [\tilde{\alpha}_1, \tilde{\alpha}_2, \dots, \tilde{\alpha}_s]$ which satisfies $\mathbb{K} \tilde{\alpha}_i = \mathbf{y}_i$, $i = 1, 2, \dots, s$. In other words, we solve the following equation:

$$\mathbb{K} \mathbb{A} = \mathbf{Y}. \quad (13)$$

Since the kernel matrix may be rank-deficient when $n \gg d$, the linear system is solved by using the Moore–Penrose pseudoinverse, yielding the unique minimum-norm solution. This choice, i.e., $\mathbb{A} = \mathbb{K}^+ \mathbf{Y}$, is widely adopted in kernel learning methods and improves numerical stability by avoiding the ambiguity associated with infinite feasible solutions.

In order to avoid forming these base kernel matrices $\{K_p\}_{p=1}^m$ and the ensemble kernel $\mathbb{K} = \sum_{p=1}^m \beta_p K_p$ directly, we use the low-rank approximation technique, i.e., Nyström algorithm, to solve (13) quickly. In detail, given the target rank k , the standard Nyström method [13,14] constructs a rank- k approximation to an arbitrary symmetric positive semidefinite (SPSD) kernel matrix by using only a few columns (or rows) of the matrix. Due to $\mathbb{K} = \sum_{p=1}^m \beta_p K_p$ in (13), we can achieve the columns of the ensemble kernel matrix $\mathbb{K}$ by selecting the columns of the base kernel matrices, thereby effectively avoiding the explicit formation of all the base kernel matrices and the ensemble kernel matrix. More precisely, let q be the oversampling number, and we denote $C_p \in \mathbb{R}^{n \times q}$ ($n \gg q > k$), $p = 1, 2, \dots, m$; the matrices consist of q columns sampled from all the base kernel matrices K_p , $p = 1, 2, \dots, m$. In this work, these columns are selected by uniform random sampling without replacement,

so that each column has an equal probability of being chosen. Specifically, a common index set $\mathcal{I} \subset \{1, 2, \dots, n\}$ containing q sampled indices is first generated, and these matrices C_p are constructed as $C_p = K_p(:, \mathcal{I})$, $p = 1, 2, \dots, m$. Therefore, all base kernel matrices share the same sampling indices, ensuring sample correspondence across different kernels when forming the ensemble sampled matrix. This randomized sampling strategy introduces negligible preprocessing cost while providing an efficient low-rank approximation. Then, the columns of the ensemble kernel matrix $\mathbb{K}$ can be represented as $\mathbb{C} = \sum_{p=1}^m \beta_p C_p$. With the matrix $\mathbb{C}$ at hand, the intersection matrix $\mathbb{W} \in \mathbb{R}^{q \times q}$ can be formed by the intersection of these q columns and the corresponding q rows. Finally, The rank- q Nyström approximation is

$$\mathbb{K} \simeq \mathbb{C}\mathbb{W}^{\dagger}\mathbb{C}^T. \quad (14)$$

Notice that if the $\{C_p\}$ s share the same sampling indices in $\{K_p\}$ s, then (14) reduces to a standard Nyström approximation to $\mathbb{K}$, and we call (14) a generalized Nyström approximation. For given matrices $\mathbb{C} \in \mathbb{R}^{n \times q}$ and $\mathbb{W} \in \mathbb{R}^{q \times q}$, an efficient method to compute the best rank- k approximation of (14) is presented in [13]. To be specific, let the economized QR decomposition of $\mathbb{C} \in \mathbb{R}^{n \times q}$ be $\mathbb{C} = \mathbb{Q}\mathbb{R}$ with $\mathbb{Q} \in \mathbb{R}^{n \times q}$, $\mathbb{R} \in \mathbb{R}^{q \times q}$, and the eigenvalue decomposition of the $q \times q$ matrix $\mathbb{R}\mathbb{W}^{\dagger}\mathbb{R}^T$ be $\mathbb{R}\mathbb{W}^{\dagger}\mathbb{R}^T = \tilde{\Lambda}\tilde{\mathbb{V}}\tilde{\mathbb{V}}^T$, so one considers the best rank- k approximation of $\mathbb{C}\mathbb{W}^{\dagger}\mathbb{C}^T$ in any unitarily invariant norm,

$$\begin{aligned} \mathbb{K}_{(k)}^{opt} &= \underset{\tilde{\mathbb{K}}: \text{rank}(\tilde{\mathbb{K}}) \leq k}{\operatorname{argmin}} \quad \|\mathbb{C}\mathbb{W}^{\dagger}\mathbb{C}^T - \tilde{\mathbb{K}}\| \\ &= \underset{\tilde{\mathbb{K}}: \text{rank}(\tilde{\mathbb{K}}) \leq k}{\operatorname{argmin}} \quad \|\mathbb{Q}(\mathbb{R}\mathbb{W}^{\dagger}\mathbb{R}^T)\mathbb{Q}^T - \tilde{\mathbb{K}}\| \\ &= \underset{\tilde{\mathbb{K}}: \text{rank}(\tilde{\mathbb{K}}) \leq k}{\operatorname{argmin}} \quad \|(\mathbb{Q}\tilde{\mathbb{V}})\tilde{\Lambda}(\mathbb{Q}\tilde{\mathbb{V}})^T - \tilde{\mathbb{K}}\| \\ &= (\mathbb{Q}\tilde{\mathbb{V}}(:, 1:k))\tilde{\Lambda}(1:k, 1:k)(\mathbb{Q}\tilde{\mathbb{V}}(:, 1:k))^T \end{aligned} \quad (15)$$

Denote $\mathbb{U} = \mathbb{Q}\tilde{\mathbb{V}}(:, 1:k) \in \mathbb{R}^{n \times k}$; we note that the columns of matrix $\mathbb{U}$ are orthonormal because $\mathbb{Q}$ and $\tilde{\mathbb{V}}$ both have orthonormal columns. Thus, let $\Lambda = \tilde{\Lambda}(1:k, 1:k) \in \mathbb{R}^{k \times k}$; we get matrices $\mathbb{U} \in \mathbb{R}^{n \times k}$ and $\Lambda \in \mathbb{R}^{k \times k}$ with the eigenvalue decomposition $\mathbb{K} \approx \mathbb{U}\Lambda\mathbb{U}^T$. Finally, the matrix $\mathbb{A}$ to be solved takes the approximate form

$$\mathbb{A} \approx \mathbb{U}(\Lambda^{\dagger}(\mathbb{U}^T\mathbb{Y})). \quad (16)$$

It is worth mentioning that we do not need to form the matrix $\mathbb{K}_{(k)}^{opt}$ in (15) explicitly, and only $\mathbb{U}$ and Λ are sufficient for the approximation (16).

See Algorithm 1 for our randomized multiple kernel discriminant analysis algorithm (RM-KDA).

Algorithm 1 A fast and randomized multiple kernel discriminant analysis algorithm (RM-KDA)

Require: The data matrix $X = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n] \in \mathbb{R}^{d \times n}$, the number of base kernel matrices m , the number of oversampling q , a target rank $k = q - 10$ and the reducing dimension s ;

Ensure: The projection matrix $\mathbb{A} = [\tilde{\mathbf{a}}_1, \tilde{\mathbf{a}}_2, \dots, \tilde{\mathbf{a}}_s] \in \mathbb{R}^{n \times s}$.

1. Let $\beta = \text{ones}(m, 1)/m$;
 2. Compute the affinity matrix $W \in \mathbb{R}^{n \times n}$;
 3. Obtain directly the largest s eigenvectors $\{\mathbf{y}_i\}_{i=1}^s$ of eigenproblem (12) according to [41], and denote by $Y = [\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_s] \in \mathbb{R}^{n \times s}$;
 4. Compute $\mathbb{U} \in \mathbb{R}^{n \times k}$ and $\Lambda \in \mathbb{R}^{k \times k}$ with the approximate eigenvalue decomposition $\mathbb{K} \approx \mathbb{U}\Lambda\mathbb{U}^T$ using Nyström approximation (15);
 5. Use $\mathbb{U}(\Lambda^{\dagger}(\mathbb{U}^T\mathbb{Y}))$ in (16) to approximate the matrix $\mathbb{A}$.
-

Remark 2. Unlike conventional MKL-based discriminant analysis methods which solve the discriminant problem directly on the entire ensemble kernel matrix $\mathbb{K}$ [11,16–22], the proposed RM-KDA incorporates a generalized Nyström randomized approximation (14) and performs discriminant learning in the resulting reduced kernel subspace. The proposed method does not modify the kernel combination mechanism; instead, it reformulates the discriminant learning process through randomized low-rank kernel approximation (16). As a result, RM-KDA preserves most of the discriminative information while substantially reducing the computational burden. This mathematical reformulation distinguishes RM-KDA from existing MKL-based discriminant analysis approaches and provides improved scalability for multiple kernel classification tasks.

3.2. Complexity Analysis

To better demonstrate the computational advantage of the proposed RM-KDA, we analyze the asymptotic time and memory complexities of its major computational stages. Throughout the analysis, let n denote the number of training samples, d the feature dimension, m the number of base kernels, q the oversampling parameter in the Nyström approximation ($q \ll n$), k the number of classes ($k \ll n$), s the discriminant space dimensionality, and n_i the number of samples in the i -th class, consistent with the notations used in Table 1.

(I) Construction of the affinity matrix $W \in \mathbb{R}^{n \times n}$:

The affinity matrix W is constructed using the `constructW` routine with the `LDA` option (`bLDA` = 1). Since nonzero entries only exist between samples belonging to the same class, W is a sparse block-diagonal matrix. Assuming balanced classes, the number of nonzero entries is $\sum_{i=1}^k n_i^2 \approx n^2/c$, where n_i denotes the number of samples in the i -th class. Consequently, both the construction time and memory complexities are $\mathcal{O}(\sum_{i=1}^k n_i^2)$.

(II) Solution $Y = [y_1, y_2, \dots, y_s] \in \mathbb{R}^{n \times s}$ of the eigenproblem (12):

The cost of the step is mainly the cost of the Gram–Schmidt method. The computational complexity is $\mathcal{O}(nk^2 + k^3)$, while the memory complexity is $\mathcal{O}(ns)$.

(III) The approximate eigenvalue decomposition in (15) using the generalized Nyström approximation:

The core of our approach lies in this stage, which comprises several crucial computational components, namely, randomized column sampling, the construction of the matrix $\mathbb{C} = \sum_{p=1}^m \beta_p C_p$, the economized QR decomposition, the construction and eigenvalue decomposition of the reduced matrix $\mathbb{R}W^{\dagger}\mathbb{R}^T$, and the associated matrix multiplications. Specific details are as follows:

- (1) Randomized column sampling and construction of matrix $\mathbb{C} = \sum_{p=1}^m \beta_p C_p$: For each base kernel matrix K_p , $p = 1, 2, \dots, m$, q columns are selected uniformly at random without replacement to form $C_p \in \mathbb{R}^{n \times q}$. Since q sampled columns are computed for each of the m base kernels, the computational cost is $\mathcal{O}(nmq)$. Moreover, the sampled matrices are then linearly combined to obtain $\mathbb{C} = \sum_{p=1}^m \beta_p C_p$. This requires adding m matrices of size $n \times q$, whose complexity is likewise $\mathcal{O}(nmq)$. Only the sampled matrices are stored during the computation, resulting in a memory complexity of $\mathcal{O}(nq)$.
- (2) The economized QR decomposition: For the ensemble sampled matrix $\mathbb{C} \in \mathbb{R}^{n \times q}$ we perform the economized QR decomposition $\mathbb{C} = \mathbb{Q}\mathbb{R}$, where $\mathbb{Q} \in \mathbb{R}^{n \times q}$, $\mathbb{R} \in \mathbb{R}^{q \times q}$. For a tall matrix ($n \gg q$), the economized QR decomposition requires $\mathcal{O}(nq^2)$ floating-point operations. The matrices $\mathbb{Q}$ and $\mathbb{R}$ occupy $nq + q^2$ entries, giving a memory complexity of $\mathcal{O}(nq)$.
- (3) Construction and eigenvalue decomposition of the reduced matrix $\mathbb{R}W^{\dagger}\mathbb{R}^T$: First, using the intersection matrix $\mathbb{W} \in \mathbb{R}^{q \times q}$, the reduced matrix $\mathbb{R}W^{\dagger}\mathbb{R}^T$ is constructed. The computation consists of Moore–Penrose pseudoinverse of $\mathbb{W}$, matrix multiplication

$\mathbb{R}W^\dagger$, and the multiplication by $\mathbb{R}^T$. Since every matrix involved is $q \times q$, each operation requires $\mathcal{O}(q^3)$ operations. Therefore, the computational complexity is $\mathcal{O}(q^3)$, while the memory complexity is $\mathcal{O}(q^2)$. Second, the eigenvalue decomposition of the $q \times q$ matrix $\mathbb{R}W^\dagger\mathbb{R}^T$ is performed, i.e., $\mathbb{R}W^\dagger\mathbb{R}^T = \mathbb{V}\tilde{\Lambda}\mathbb{V}^T$. Hence, the computational complexity is $\mathcal{O}(q^3)$, while the memory complexity is $\mathcal{O}(q^2)$. Finally, the overall computational complexity of the step is $\mathcal{O}(q^3)$, and memory complexity is $\mathcal{O}(q^2)$.

- (4) Construction of the approximate eigenvalue decomposition in (15): The approximate eigendecomposition of the ensemble kernel matrix $\mathbb{K}$ is obtained as $\mathbb{K} \approx (\mathbb{Q}\mathbb{V}(:, 1:k))\tilde{\Lambda}(1:k, 1:k)(\mathbb{Q}\mathbb{V}(:, 1:k))^T \doteq \mathbb{U}\mathbb{A}\mathbb{U}^T$, where $\mathbb{U} = \mathbb{Q}\mathbb{V}(:, 1:k) \in \mathbb{R}^{n \times k}$, $\mathbb{A} = \tilde{\Lambda}(1:k, 1:k) \in \mathbb{R}^{k \times k}$. The multiplication of an $n \times q$ matrix with a $q \times k$ matrix requires $\mathcal{O}(nqk)$ operations. The resulting eigenvector matrix $\mathbb{U} \in \mathbb{R}^{n \times k}$ occupies memory $\mathcal{O}(nk)$.

Combining all four steps, the generalized Nyström approximation has $\mathcal{O}(n(mq + q^2 + qk) + q^3)$ floating-point operations and the memory complexity is $\mathcal{O}(n(q + k))$.

(IV) Computation of the final projection matrix $\mathbb{A}$ using (16):

Finally, the projection matrix $\mathbb{A}$ is approximated by matrix multiplication $\mathbb{U}(\Lambda^\dagger(\mathbb{U}^T\mathbb{Y}))$ in (16), whose computational complexity is $\mathcal{O}(nks + k^3)$, and memory complexity is $\mathcal{O}(ns)$.

Based on this analysis, we derive the overall computational complexity of RM-KDA as $\mathcal{O}((k^2 + qk + q^2 + ks)n + \sum_{i=1}^k n_i^2)$, and the corresponding memory complexity is $\mathcal{O}((s + q + k)n + \sum_{i=1}^k n_i^2)$. In the proposed RM-KDA, only low-rank matrices of size $n \times q$ are stored during the computation. In contrast, conventional kernel discriminant analysis requires constructing and storing the full $n \times n$ kernel matrix with memory complexity $\mathcal{O}(n^2)$, together with solving an $n \times n$ eigenvalue problem whose computational complexity is typically $\mathcal{O}(n^3)$. Therefore, the proposed RM-KDA substantially reduces both computational and memory costs while preserving competitive recognition performance. Finally, we summarize our complexity analysis results in Table 2.

Table 2. Theoretical time and memory complexities of different methods.

Methods	Time Complexity	Memory Complexity
LDA [24]	$\mathcal{O}(nd^2 + d^3)$	$\mathcal{O}(nd + d^2)$
KDA [41]	$\mathcal{O}(n^3 + dn^2)$	$\mathcal{O}(n^2)$
RSKDA [41]	$\mathcal{O}(n^3 + dn^2)$	$\mathcal{O}(n^2)$
RM-KDA	$\mathcal{O}((k^2 + qk + q^2 + ks)n + \sum_{i=1}^k n_i^2)$	$\mathcal{O}((s + q + k)n + \sum_{i=1}^k n_i^2)$

4. Numerical Experiments

In this section, to illustrate the numerical behavior of our proposed randomized multiple kernel discriminant analysis algorithm (RM-KDA, see Algorithm 1), we perform numerical experiments on the benchmark databases. First, in Section 4.1, we describe the databases and some kernel function design used in the experiments, as well as some compared algorithms. Then, numerical experiments for our randomized multiple kernel discriminant analysis algorithm (RM-KDA) with applications to recognition are shown in Section 4.2. Finally, sensitivity analysis of the oversampling parameter q is performed in Section 4.3.

All the numerical experiments are carried on an Intel(R) Core(TM) i9-14900KF processor, and with CPU 3.20 GHz and RAM 64 GB. The operation system is 64-bit Windows 10, and the numerical results are obtained from running the MATLAB R2023b software. All experiments are performed using a CPU without GPU acceleration. Memory consumption is monitored through MATLAB memory statistics, and the algorithms that exceeded the available physical memory are marked as O.M. (out of memory) in the experimental tables.

4.1. Databases, Kernel Functions Design and Compared Algorithms

To comprehensively evaluate both the application performance in bird recognition and the generalization ability of the proposed algorithm, experiments are conducted on two bird recognition datasets, Caltech-UCSD Birds-200-2011 (CUB-200-2011) [8] and North America Birds (NABirds), as well as the character image dataset FounderType-20 [56].

Let us summarize some basic information of these databases.

- The Caltech-UCSD Birds-200-2011 (CUB-200-2011 (https://www.vision.caltech.edu/datasets/cub_200_2011/, accessed on 17 April 2026)) database is an extended version of CUB-200, a challenging dataset of $k = 200$ bird species [8]. The dataset is one of the most widely used benchmarks for fine-grained visual classification tasks. In this experiment, the original images are represented as 32×32 -pixel-sized grayscale images. Making use of “appearance-based pattern recognition”, we interpret each pixel as a feature, resulting in $d = 1024$ -dimensional feature vectors.
- The NABirds (<https://dl.allaboutbirds.org/nabirds>, accessed on 17 April 2026) dataset is designed for fine-grained image recognition, particularly focused on bird species identification. It contains a total of 48,562 images spanning 555 bird categories, capturing birds in various states. This dataset serves as a valuable resource for advancing research in computer vision, especially in the area of avian species recognition. For the NABirds dataset, which is a fine-grained bird classification dataset, direct classification using raw pixel values or traditional handcrafted features generally leads to unsatisfactory performance. Therefore, we employ a pre-trained residual network-50 (ResNet-50) model to extract $d = 2048$ -dimensional deep feature representations. In addition, categories with fewer than 50 images are removed, resulting in a total of $N = 45,476$ samples with $k = 487$ categories.
- FounderType-200 (<https://www.wict.pku.edu.cn/zlian/representa/ijc015/index.htm>, accessed on 17 April 2026) consists of 200 different fonts produced by a company named FounderType with each font containing 6763 Chinese character images. In this experiment, we randomly pick $k = 15$ fonts; namely, $N = 101,445$ samples are used for training and testing. Moreover, the original images are vectorized to $d = 4096$ -dimensional vectors.

One refers to Figures 1–3 for some samples from the three data sets.

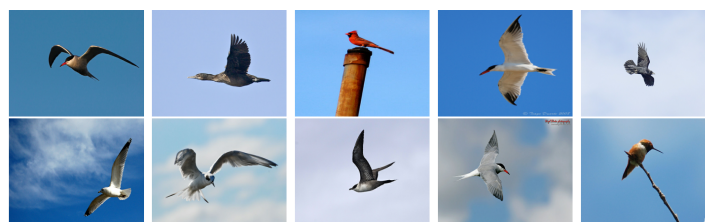


Figure 1. Some samples of CUB-200-2011 database.



Figure 2. Some samples of NABirds database.



Figure 3. Some samples of Chinese character image database FounderType-200.

Next, we introduce the base kernel matrices utilized in our experiment. Following the strategy utilized in most multiple kernel learning methods [22,57], three well-known kernel functions (with different kernel parameters resulting in 14 kernels) are utilized to construct the base kernel matrices, including seven Gaussian kernels, six polynomial kernels and one linear kernel, and the specific settings are shown below.

- (1) Gaussian kernels: The (i, j) -th element of the kernel matrix is taken as

$$\kappa(x_i, x_j) = \exp\left(-\frac{\|x_i - x_j\|_2^2}{2\tau\sigma^2}\right), \quad i, j = 1, 2, \dots, n,$$

where $\tau \in \{0.01, 0.05, 0.1, 1, 10, 50, 100\}$. So there are seven Gaussian kernels altogether.

- (2) Polynomial kernels: The (i, j) -th element of the kernel matrix is taken as

$$\kappa(x_i, x_j) = (\alpha + x_i^T x_j)^\zeta, \quad i, j = 1, 2, \dots, n,$$

where $\alpha \in \{0, 1\}$ and $\zeta \in \{2, 4, 6\}$, and there are six polynomial kernels altogether.

- (3) Linear kernel: The (i, j) -th element of the kernel matrix is taken as

$$\kappa(x_i, x_j) = \frac{x_i^T x_j}{\|x_i\|_2 \|x_j\|_2}, \quad i, j = 1, 2, \dots, n.$$

Finally, let us briefly introduce the compared algorithms.

1. Linear Discriminant Analysis (LDA) [24]: It is a well-known linear algorithm that provides benchmark performance.
2. Kernel Discriminant Analysis (KDA) [42]: The regularization parameter is chosen by five-fold cross-validation on the training set.
3. Spectral Regression Kernel Discriminant Analysis (SRKDA) [23,41,42]: The selection of regularization parameters is similar to the selection of the KDA method.
4. Randomized Multiple Kernel Discriminant Analysis (RM-KDA): It is our approach proposed in this paper. The method uses multiple kernel matrices instead of a single simple kernel.

4.2. Numerical Experiments for Randomized Multiple Kernel Discriminant Analysis Algorithm with Applications to Recognition

In this subsection, we try to show the superiority of Algorithm 1 (RM-KDA) over many state-of-the-art dimensionality reduction methods on the benchmark datasets. Tables 3 and 4 list the numerical results of mean recognition accuracy, mean CPU time (in seconds), and corresponding standard deviations (Std-Dev) from the compared methods over 10 independent runs on the bird databases CUB-200-2011 and NABirds. For a fair comparison, the reported CPU times only measure the execution time of the discriminative learning algorithms after the input features have been prepared. Specifically, the reported CPU times include kernel construction (or Nyström approximation for RM-KDA), discriminant analysis, and classification, but exclude image preprocessing (e.g., image resizing, vectorization, and ResNet-50 deep feature extraction). These preprocessing steps are identical for all competing methods and are performed only once before the classification stage.

To provide a more comprehensive statistical evaluation, besides the mean recognition accuracy and standard deviation, we additionally report the 95% confidence intervals (95% CIs) of the recognition accuracies in Table 4. Moreover, paired *t*-tests are performed to evaluate the statistical significance of the performance differences between the proposed RM-KDA and the competing methods. Since several baseline methods failed to complete the experiments because of memory overflow on CUB-200-2011 and FounderType-200 datasets, statistical significance tests are only performed on the NABirds dataset, and one refers to Table 5 for more details. Three remarks are in order.

Table 3. Numerical performance of the compared methods on bird database CUB-200-2011. Here “O.M.” indicates that the algorithm is “out of memory”.

CUB-200-2011			
Algorithms (Methods)	Recognition Rate $\pm$ Std-Dev		
	$n = 40\% N$	$n = 60\% N$	$n = 80\% N$
LDA	99.58 $\pm$ 0.0727%	99.88 $\pm$ 0.0246%	99.90 $\pm$ 0.0281%
KDA	99.58 $\pm$ 0.0977%	O.M.	O.M.
SRKDA	99.59 $\pm$ 0.0937%	99.89 $\pm$ 0.0245%	O.M.
RM-KDA	99.55 $\pm$ 0.0877%	99.87 $\pm$ 0.0267%	99.87 $\pm$ 0.0197%
Algorithms (Methods)	CPU Time/s $\pm$ Std-Dev		
	$n = 40\% N$	$n = 60\% N$	$n = 80\% N$
LDA	0.3220 $\pm$ 0.0521	0.3752 $\pm$ 0.0116	0.4381 $\pm$ 0.0177
KDA	1295.6357 $\pm$ 15.9674	O.M.	O.M.
SRKDA	64.4217 $\pm$ 0.2583	592.2034 $\pm$ 28.0281	O.M.
RM-KDA	3.5656 $\pm$ 0.0761	8.9091 $\pm$ 0.1459	12.7663 $\pm$ 1.8181

Here N represents the total number of samples.

Table 4. Numerical performance of the compared methods on bird database NABirds. Here “O.M.” indicates that the algorithm is “out of memory”.

NABirds			
Algorithms (Methods)	Recognition Rate $\pm$ Std-Dev		
	$n = 40\% N$	$n = 60\% N$	$n = 80\% N$
LDA	41.91 $\pm$ 0.2447%	44.85 $\pm$ 0.4078%	46.59 $\pm$ 0.2647%
KDA	48.99 $\pm$ 5.6116%	49.83 $\pm$ 6.0683%	52.89 $\pm$ 4.7372%
SRKDA	50.46 $\pm$ 4.2077%	51.84 $\pm$ 4.8761%	55.17 $\pm$ 3.8175%
RM-KDA	52.06 $\pm$ 0.5416%	53.25 $\pm$ 0.2657%	55.13 $\pm$ 0.0527%
Algorithms (Methods)	95% Confidence Intervals (CI)		
	$n = 40\% N$	$n = 60\% N$	$n = 80\% N$
LDA	[41.73%, 42.08%]	[44.56%, 45.14%]	[46.40%, 46.78%]
KDA	[44.98%, 53.01%]	[45.49%, 54.17%]	[49.50%, 56.28%]
SRKDA	[47.45%, 53.47%]	[48.35%, 55.33%]	[52.44%, 57.90%]
RM-KDA	[51.67%, 52.45%]	[53.06%, 53.44%]	[55.09%, 55.17%]
Algorithms (Methods)	CPU Time/s $\pm$ Std-Dev		
	$n = 40\% N$	$n = 60\% N$	$n = 80\% N$
LDA	0.6577 $\pm$ 0.0343	0.8569 $\pm$ 0.0084	1.0566 $\pm$ 0.0033
KDA	115.9037 $\pm$ 0.1966	366.5554 $\pm$ 0.1052	834.7591 $\pm$ 1.6618
SRKDA	9.1170 $\pm$ 0.0634	23.5240 $\pm$ 0.0246	47.9783 $\pm$ 0.2360
RM-KDA	8.0029 $\pm$ 0.0378	18.7819 $\pm$ 0.1781	36.1642 $\pm$ 0.0609

Here N represents the total number of samples.

First, in terms of recognition accuracy, the experimental results on both CUB-200-2011 and NABirds demonstrate that the proposed RM-KDA achieves competitive or superior classification performance compared with existing methods. On the CUB-200-2011 dataset, all approaches yield very high recognition rates, and LDA slightly outperforms RM-KDA, indicating that the data are close to linearly separable. In contrast, on the more challenging NABirds dataset, RM-KDA consistently outperforms LDA, confirming the advantage of incorporating nonlinear discriminative information through multiple kernels. Compared with KDA and SRKDA, RM-KDA also achieves comparable or better accuracy on both datasets.

It is worth noting that the recognition rates on CUB-200-2011 are significantly higher than those on NABirds. In particular, LDA achieves recognition rates close to 100% on CUB-200-2011, indicating that the adopted feature representation exhibits a high degree of linear separability. This phenomenon can be attributed to the smaller number of classes (200 in CUB-200-2011 versus 487 in NABirds) and the generally larger inter-class variations in CUB-200-2011. In contrast, NABirds contains many closely related bird species with subtle visual differences, making the classification task substantially more challenging. Consequently, the performance gains achieved by nonlinear kernel methods on CUB-200-2011 are naturally limited, whereas their advantages become more evident on the more challenging NABirds database.

Second, in terms of CPU time, the results on both CUB-200-2011 and NABirds databases clearly demonstrate the computational advantage of the proposed RM-KDA. As shown from Table 3, RM-KDA runs significantly faster than classical KDA and SRKDA, achieving speedups of approximately 20 to 100 times on the CUB-200-2011 dataset. While RM-KDA is slightly slower than LDA, this is because RM-KDA is a multiple kernel method that involves additional kernel-related computations, whereas LDA is a purely linear approach. More importantly, as the size of the training data increases, KDA and SRKDA suffer from prohibitive computational and memory costs and fail to run due to out-of-memory issues. In contrast, RM-KDA consistently completes all experimental settings on both datasets and requires substantially less CPU time than other kernel-based methods, while maintaining high recognition accuracy. These results indicate that RM-KDA achieves a favorable trade-off between accuracy and efficiency, making it particularly suitable for the fine-grained recognition tasks.

Third, to further assess the reliability of the experimental results, we additionally report the 95% confidence intervals in Table 4, and conduct paired *t*-tests based on ten independent runs in Table 5. As shown in Table 4, the proposed RM-KDA consistently achieves the highest or most competitive mean recognition accuracy while exhibiting relatively narrow confidence intervals, indicating good robustness over repeated experiments. Moreover, the paired *t*-test results in Table 5 further show that RM-KDA significantly outperforms LDA under all training ratios ($p < 0.05$). Compared with KDA and SRKDA, RM-KDA achieves competitive recognition performance, although the corresponding differences are not statistically significant ($p > 0.05$). These statistical analyses indicate that the proposed method provides competitive classification performance together with good experimental stability.

Although the primary application considered in this paper is bird recognition, RM-KDA is a general multiple kernel discriminant learning algorithm. Therefore, in addition to two bird recognition datasets (CUB-200-2011 and NABirds), the Chinese character image dataset FounderType-200 (with $N = 101,445$) is included to evaluate the generalization capability and computational scalability of the proposed method on another visual recognition task. Since FounderType-200 is characterized by high-dimensional features, numerous

categories, and substantial inter-class similarity, it provides a complementary benchmark for assessing the robustness and effectiveness of RM-KDA beyond bird recognition.

Table 5. Statistical significance tests (paired *t*-test) on bird database NABirds.

Training Ratio	Comparison	<i>p</i> -Value	Significant?
40%	RM-KDA vs. LDA	0.0018	Yes
	RM-KDA vs. KDA	0.4727	No
	RM-KDA vs. SRKDA	0.6143	No
60%	RM-KDA vs. LDA	0.0013	Yes
	RM-KDA vs. KDA	0.4449	No
	RM-KDA vs. SRKDA	0.6767	No
80%	RM-KDA vs. LDA	0.0004	Yes
	RM-KDA vs. KDA	0.4951	No
	RM-KDA vs. SRKDA	0.9871	No

On one hand, as shown in Table 6, kernel-based methods consistently outperform linear LDA in terms of recognition accuracy, indicating the presence of strong nonlinear structures in Chinese character images. Among all methods, RM-KDA achieves competitive or superior recognition performance and attains the highest accuracy when 80% of the samples are trained. On the other hand, in terms of computational efficiency, classical KDA suffers from prohibitive memory and time costs and becomes infeasible for large training sizes, and SRKDA also fails under large data size. In contrast, RM-KDA successfully completes all experimental settings with substantially lower CPU time than other kernel-based approaches.

Table 6. Numerical performance of the compared methods on FounderType-200 database. Here “O.M.” indicates that the algorithm is “out of memory”.

Algorithms (Methods)	Recognition Rate $\pm$ Std-Dev		
	$n = 40\% N$	$n = 60\% N$	$n = 80\% N$
LDA	69.69 $\pm$ 0.1386%	71.00 $\pm$ 0.1623%	71.70 $\pm$ 0.2993%
KDA	88.03 $\pm$ 0.8491%	O.M.	O.M.
SRKDA	89.18 $\pm$ 0.6255%	89.90 $\pm$ 0.6470%	O.M.
RM-KDA	87.87 $\pm$ 0.0460%	89.89 $\pm$ 0.2512%	90.63 $\pm$ 0.0456%
Algorithms (Methods)	CPU Time/s $\pm$ Std-Dev		
	$n = 40\% N$	$n = 60\% N$	$n = 80\% N$
LDA	2.4770 $\pm$ 0.0553	3.3384 $\pm$ 0.0643	4.1704 $\pm$ 0.0816
KDA	1155.8240 $\pm$ 10.2655	O.M.	O.M.
SRKDA	60.2605 $\pm$ 1.1230	690.9324 $\pm$ 84.4988	O.M.
RM-KDA	64.2525 $\pm$ 0.0742	157.3210 $\pm$ 1.1527	298.7151 $\pm$ 5.2532

Here *N* represents the total number of samples.

These results indicate that the proposed randomized algorithm effectively balances recognition accuracy and computational efficiency and exhibits strong generalization beyond bird recognition tasks.

Finally, to provide an intuitive illustration of the discriminative representation learned by RM-KDA, we randomly selected four representative categories (Classes 1, 36, 108, and 145) from the CUB-200-2011 dataset. Twenty training samples are randomly chosen from each category and projected onto two representative discriminant coordinates learned by RM-KDA. The corresponding visualization is shown in Figure 4. As illustrated in Figure 4, samples from different categories exhibit a certain degree of overlap in the

two-dimensional projection, and no completely separated clusters are observed. This phenomenon is expected because RM-KDA learns a discriminative embedding in a $(k - 1)$ -dimensional kernel subspace, where k denotes the number of classes. For visualization purposes, only two representative discriminant coordinates are displayed. Such a two-dimensional projection inevitably discards a substantial portion of the discriminative information preserved in the original high-dimensional embedding. Therefore, overlap among classes in the visualization does not necessarily imply poor discriminative capability of the learned representation.

Moreover, it should be noted that the primary objective of RM-KDA is to learn a discriminative projection for classification rather than to optimize visual separability in a two-dimensional plane. Accordingly, the effectiveness of the proposed method is mainly reflected by the quantitative recognition results, whereas Figure 4 serves only as an intuitive visualization of the learned embedding.

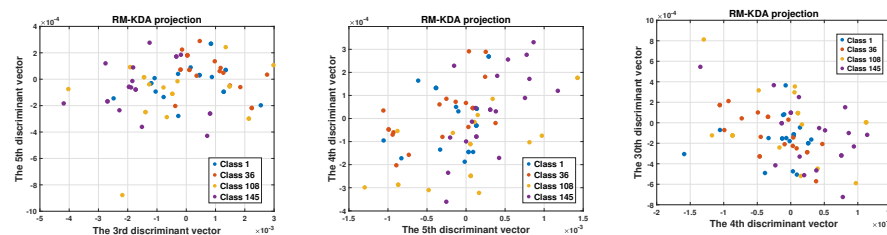


Figure 4. 2D Visualization of the Learned Discriminative Subspace by MR-KDA on CUB-200-2011.

4.3. Sensitivity Analysis of the Oversampling Parameter q

The parameter q is the oversampling size used in the generalized Nyström approximation. In general, a larger value of q provides a more accurate low-rank approximation of the ensemble kernel matrix $\mathbb{K}$, while simultaneously increasing the computational cost. Therefore, the selection of q represents a trade-off between approximation accuracy and computational efficiency rather than a theoretically unique optimal choice.

To investigate its influence, we conducted a sensitivity analysis on the NABirds dataset. The number of training samples is fixed at $n = 40\% N$, and the oversampling parameter is varied over $q = \{10, 20, 40, 50, 60, 80, 100, 150, 200, 300, 400, 600, 800, 1000, 1200, 1500, 2000\}$. For each value of q , RM-KDA is independently repeated ten times. Figure 5 reports the mean recognition accuracy and CPU time together with corresponding standard deviations, where the solid curves represent the mean values, while the shaded regions indicate one standard deviation.

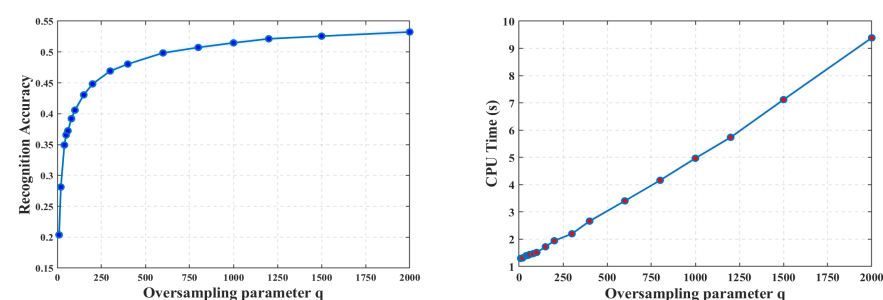


Figure 5. The recognition accuracy and CPU time versus the oversampling parameter q on NABirds.

As shown in Figure 5, as the oversampling parameter increases, the recognition accuracy first improves rapidly and then gradually stabilizes after q reaches a moderate size. In contrast, the computational time increases monotonically. Moreover, the relatively narrow shaded regions demonstrate that the proposed RM-KDA is robust with respect to random sampling. These results indicate that the oversampling parameter is not overly

sensitive within a reasonable range and that a moderate value achieves a favorable balance between approximation and computational efficiency.

5. Conclusions

The importance of bird recognition is that birds are not only an important part of the ecosystem, but also key indicators of biodiversity. By identifying birds, we can better understand the health of the ecosystem, monitor environmental changes, and protect wildlife and their habitats. In addition, bird recognition is also the basis of airport bird strike prevention work, which plays an important role in improving the flight safety of aircraft.

In this paper, we proposed a fast and randomized multiple kernel discriminant analysis (RM-KDA) method with applications to bird recognition, so as to prepare for the subsequent bird repellent decision methods. By incorporating a generalized Nyström approximation into the multiple kernel discriminant analysis framework, the proposed method avoids constructing and decomposing the full ensemble kernel matrix directly, thereby substantially reducing both computational complexity and memory consumption while preserving the discriminative capability of multiple kernels.

Extensive experiments on the CUB-200-2011, NABirds, and FounderType-200 datasets demonstrate that RM-KDA achieves competitive or superior classification accuracy while requiring significantly less computational time than conventional kernel discriminant analysis methods. In addition, the statistical analysis based on 95% confidence intervals and statistical significance tests (paired *t*-tests) further confirms the effectiveness and robustness of the proposed method.

Although RM-KDA significantly reduces the computational complexity and memory requirement through the Nyström-based randomized approximation, its performance still depends on the quality of the low-rank approximation. When the sampling rank is insufficient or the underlying kernel matrix is not well approximated by a low-rank structure (e.g., when its singular values decay slowly), the approximation error may increase and slightly degrade the classification performance. Therefore, RM-KDA involves a trade-off between computational efficiency and approximation accuracy. In addition, fixed equal kernel weights are adopted in this paper for simplicity and computational efficiency, since learning adaptive kernel weights would require solving an additional optimization problem jointly over the discriminant projection and the kernel coefficients, thereby increasing the computational cost. Future works will focus on adaptive sampling strategies, adaptive rank selection, more accurate randomized kernel approximation techniques, and adaptive kernel weight learning to further improve the efficiency, robustness, and classification performance of multiple kernel learning.

Author Contributions: Conceptualization, K.L. and B.L.; methodology, K.L., B.L. and C.W.; software, B.L. and C.W.; validation, B.L. and X.W.; data curation, X.W.; writing—original draft preparation, K.L.; writing—review and editing, B.L. and C.W.; visualization, X.W.; supervision, K.L.; project administration, K.L. All authors have read and agreed to the published version of the manuscript.

Funding: The authors declare that no funds, grants, or other support were received.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Acknowledgments: We would like to express our sincere thanks to the anonymous referees and our editor for their insightful comments and suggestions that greatly improved the representation of this paper.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Cai, J.; Bao, H.; Zuo, H.; Huang, Y. Safety evaluation of airworthiness requirement of bird-strike on aeroplane. *Eng. Fail. Anal.* **2019**, *102*, 407–416. [\[CrossRef\]](#)
2. Isabel, M.; Joost, E.; Thorsten, M. The Bird Strike Challenge. *Compos. Struct.* **2016**, *149*, 385–397.
3. Wang, J.; Ning, H.; Chen, W. Airport Bird-strike Risk Assessment Model with Grey Clustering Evaluation Method. *Chin. J. Electron.* **2012**, *21*, 409–413.
4. Avrenli, K.; Dempsey, B. Statistical Analysis of Aircraft-Bird Strikes Resulting in Engine Failure. *Transp. Res. Rec.* **2014**, *2449*, 14–23. [\[CrossRef\]](#)
5. Wayne, C. Using Smeed's Law to Estimate Number of Bird Strikes in Utah. *Transp. Res. Rec.* **2011**, *2214*, 27–33. [\[CrossRef\]](#)
6. Kwan, C.; Ho, K.; Mei, G. An automated acoustic system to monitor and classify birds. *Eurasip J. Appl. Signal Process.* **2006**, *2006*, 096706. [\[CrossRef\]](#)
7. Tang, Q.; Xu, L.; Zheng, B. Transound: Hyper-head attention transformer for birds sound recognition. *Ecol. Inform.* **2023**, *75*, 102001. [\[CrossRef\]](#)
8. Wah, C.; Branson, S.; Welinder, P.; Perona, P.; Belongie, S. *The Caltech-UCSD Birds-200-2011 Dataset, Computation & Neural Systems Technical Report*; California Institute of Technology: Pasadena, CA, USA, 2011. Available online: <https://resolver.caltech.edu/CaltechAUTHORS:20111026-120541847> (accessed on 17 April 2026).
9. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [\[CrossRef\]](#) [\[PubMed\]](#)
10. Lanckriet, G.; Cristianini, N.; Bartlett, P.; Ghaoui, L.; Jordan, M. Learning the kernel matrix with semidefinite programming. *J. Mach. Learn. Res.* **2004**, *5*, 27–72.
11. Lin, Y.; Liu, T.; Fuh, C. Multiple kernel learning for dimensionality reduction. *IEEE Trans. Pattern Anal. Mach. Intell.* **2011**, *33*, 1147–1160. [\[CrossRef\]](#) [\[PubMed\]](#)
12. Wang, T.; Zhang, L.; Hu, W. Bridging deep and multiple kernel learning: A review. *Inf. Fusion* **2021**, *67*, 3–13. [\[CrossRef\]](#)
13. Anaraki, F.; Becker, S. Improved fixed-rank Nyström approximation via QR decomposition: Practical and theoretical aspects. *Neurocomputing* **2019**, *363*, 261–272. [\[CrossRef\]](#)
14. Drineas, P.; Mahoney, M. On the Nyström method for approximating a gram matrix for improved kernel-based learning. *J. Mach. Learn. Res.* **2005**, *6*, 2153–2175.
15. Li, K.; Wu, G. Randomized approximate class-specific kernel spectral regression analysis for large-scale face verification. *Mach. Learn.* **2022**, *111*, 2037–2091. [\[CrossRef\]](#)
16. Du, L.; Zhou, P.; Shi, L.; Wang, H.; Fan, M. Robust multiple kernel k-means using $l_{2,1}$ -norm. In Proceedings of the 24 International Joint Conference on Artificial Intelligence, Buenos Aires, Argentina, 25–31 July 2015; pp. 3476–3482.
17. Gönen, M.; Margolin, A.A. Localized data fusion for kernel k-means clustering with application to cancer biology. In *Advances in Neural Information Processing Systems*, Curran Associates, Inc.: Red Hook, NY, USA, 2014; pp. 1305–1313.
18. Jiang, W.; Chung, F. A trace ratio maximization approach to multiple kernel-based dimensionality reduction. *Neural Netw.* **2014**, *49*, 96–106. [\[CrossRef\]](#) [\[PubMed\]](#)
19. Liu, X.; Dou, Y.; Yin, J.; Wang, L.; Zhu, E. Multiple kernel k-means clustering with matrix-induced regularization. *Proc. Thirtieth AAAI Conf. Artif. Intell. (AAAI-16)* **2016**, *30*, 1888–1894. [\[CrossRef\]](#)
20. Liu, M.; Sun, W.; Liu, B. Multiple kernel dimensionality reduction via spectral regression and trace ratio maximization. *Knowl.-Based Syst.* **2015**, *83*, 159–169. [\[CrossRef\]](#)
21. Wang, S.; Liu, X.; Liu, L.; Zhou, S.; Zhu, E. Late Fusion Multiple Kernel Clustering with Proxy Graph Refinement. *IEEE Trans. Neural Netw. Learn. Syst.* **2023**, *34*, 4359–4370. [\[CrossRef\]](#) [\[PubMed\]](#)
22. Yao, Y.; Li, Y.; Jiang, B.; Chen, H. Multiple kernel k-means clustering by selecting representative kernels. *IEEE Trans. Neural Netw. Learn. Syst.* **2021**, *32*, 4983–4996. [\[CrossRef\]](#) [\[PubMed\]](#)
23. Cai, D.; He, X.; Han, J. Speed up kernel discriminant analysis. *VLDB J.* **2011**, *20*, 21–33.
24. Fukunaga, K. *Introduction to Statistical Pattern Recognition*, 2nd ed.; Academic Press: San Diego, CA, USA, 1991.
25. Ye, J.; Janardan, R.; Park, C.H.; Park, H. An optimization criterion for generalized discriminant analysis on undersampled problems. *IEEE Trans. Pattern Anal. Mach. Intell.* **2004**, *26*, 982–994. [\[CrossRef\]](#) [\[PubMed\]](#)
26. Yu, H.; Yang, J. A direct LDA algorithm for high dimensional data-with application to face recognition. *Pattern Recognit.* **2001**, *34*, 2067–2070. [\[CrossRef\]](#)
27. Jia, Y.; Nie, F.; Zhang, C. Trace ratio problem revisited. *IEEE Trans. Neural Netw.* **2009**, *20*, 729–735. [\[CrossRef\]](#) [\[PubMed\]](#)
28. Nie, F.; Xiang, S.; Jia, Y.; Zhang, C.; Yan, S. Trace ratio criterion for feature selection. In Proceedings of the 23rd AAAI Conference on Artificial Intelligence, Chicago, IL, USA, 13–17 July 2008; pp. 671–676.
29. Park, C.; Park, H. A comparison of generalized linear discriminant analysis algorithms. *Pattern Recognit.* **2008**, *41*, 1083–1097. [\[CrossRef\]](#)
30. Ahmed, N. Exponential discriminant regularization using nonnegative constraint and image descriptor. In Proceedings of the IEEE 9th International Conference on Emerging Technologies, Islamabad, Pakistan, 9–10 December 2013; pp. 1–6.

31. Yan, L.; Pan, J. Two-dimensional exponential discriminant analysis and its application to face recognition. In Proceedings of the International Conference on Computational Aspects of Social Networks (CASoN), Vellore, India, 19–21 December 2010; pp. 528–531.
32. Friedman, J. Regularized discriminant analysis. *J. Am. Stat. Assoc.* **1989**, *84*, 165–175. [\[CrossRef\]](#)
33. Ye, J.; Li, Q. LDA/QR: An efficient and effective dimension reduction algorithm and its theoretical foundation. *Pattern Recognit.* **2004**, *37*, 851–854. [\[CrossRef\]](#)
34. Howland, P.; Park, H. Generalizing discriminant analysis using the generalized singular value decomposition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2004**, *26*, 995–1006. [\[CrossRef\]](#) [\[PubMed\]](#)
35. Belhumeur, P.; Hespanha, J.; Kriegman, D. Eigenfaces vs. fisherface: Recognition using class-specific linear projection. *IEEE Trans. Pattern Anal. Mach. Intell.* **1997**, *19*, 711–720. [\[CrossRef\]](#)
36. Jolliffe, I.T. *Principal Component Analysis*; Springer-Verlag: Berlin, Germany, 1986.
37. Martinez, A.; Kak, A.C. PCA Versus LDA. *IEEE Trans. Pattern Anal. Mach. Intell.* **2001**, *23*, 228–233. [\[CrossRef\]](#)
38. Turk, M.; Pentland, A. Face recognition using eigenfaces. In Proceedings of the IEEE Conference Computer Vision and Pattern Recognition, Maui, HI, USA, 3–6 June 1991; pp. 586–591.
39. Xiang, S.; Nie, F.; Zhang, C. Learning a mahalanobis distance metric for data clustering and classification. *Pattern Recognit.* **2008**, *41*, 3600–3612. [\[CrossRef\]](#)
40. Yan, S.; Tang, X. Trace quotient problems revisited. In *Proceedings of the European Conference on Computer Vision, Graz, Austria, 7–13 May 2006*; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2006; pp. 232–244.
41. Cai, D.; He, X.; Han, J. *Efficient Kernel Discriminant Analysis via Spectral Regression*; Department of Computer Science Technical Report No. 2888; University of Illinois at Urbana-Champaign (UIUCDCS-R-2007-2888): Urbana, IL, USA, 2007.
42. Cai, D.; He, X.; Han, J. SRDA: An Efficient Algorithm for Large-Scale Discriminant Analysis. In *Proceedings of the IEEE TKDE*; Institute of Electrical and Electronics Engineers (IEEE): Los Alamitos, CA, USA, 2008; pp. 1–12.
43. Baudat, G.; Anouar, F. Generalized discriminant analysis using a kernel approach. *Neural Comput.* **2000**, *12*, 2385–2404. [\[CrossRef\]](#) [\[PubMed\]](#)
44. Schölkopf, B.; Smola, A. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*; MIT Press: Cambridge, MA, USA, 2002.
45. Zhou, Z. *Machine Learning*, 2nd ed.; Tsinghua University Press: Beijing, China, 2016. (In Chinese)
46. Iosifidis, A.; Tefas, A.; Pitas, I. Class-specific reference discriminant analysis with application in human behavior analysis. *IEEE Trans. Hum.-Mach. Syst.* **2015**, *45*, 315–326. [\[CrossRef\]](#)
47. Tavernier, J.; Simm, J.; Meerbergen, K.; Wegner, J.K.; Ceulemans, H.; Moreau, Y. Fast semi-supervised discriminant analysis for binary classification of large data sets. *Pattern Recognit.* **2019**, *91*, 86–99. [\[CrossRef\]](#)
48. Duda, R.; Hart, P.; Stork, D. *Pattern Classification*, 2nd ed.; Wiley: Hoboken, NJ, USA, 2000.
49. Gönen, M.; Alpaydin, E. Multiple kernel learning algorithms. *J. Mach. Learn. Res.* **2011**, *12*, 2211–2268.
50. Taylor, J.; Cristianini, N. *Kernel Methods for Pattern Analysis*; Cambridge University Press: Cambridge, UK, 2004.
51. Micchelli, C.; Pontil, M. Learning the kernel function via regularization. *J. Mach. Learn. Res.* **2005**, *6*, 1099–1125.
52. Rakotomamonjy, A.; Bach, F.; Canu, S.; Grandvalet, Y. SimpleMKL. *J. Mach. Learn. Res.* **2008**, *9*, 2491–2521.
53. Wang, H.; Yan, S.; Xu, D.; Huang, X. Trace-ratio vs. ratio-trace for dimensionality reduction. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Minneapolis, MN, USA, 17–22 June 2007; pp. 1–8.
54. Liu, B.; Xia, S.; Zhou, Y. Multiple kernel spectral regression for dimensionality reduction. *J. Appl. Math.* **2013**, *2013*, 1–8. [\[CrossRef\]](#)
55. Shi, W.; Wu, G. New algorithms for trace-ratio problem with application to high-dimension and large-sample data dimensionality reduction. In *Machine Learning, Special Issue on Feature Engineering*, Springer: Dordrecht, The Netherlands, 2024; pp. 3889–3916.
56. Liu, J.; Lian, Z.; Wang, Y.; Xiao, J. Incremental kernel null space discriminant analysis for novelty detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 792–800.
57. Ren, Z.; Lei, H.; Sun, Q.; Yang, C. Simultaneous learning coefficient matrix and affinity graph for multiple kernel clustering. *Inf. Sci.* **2021**, *547*, 289–306. [\[CrossRef\]](#)

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.