Article

Dual Graph Laplacian RPCA Method for Face Recognition Based on Anchor Points

Shu-Ting Zhuang , Qing-Wen Wang * , Jiang-Feng Chen

Posted Date: 31 March 2025

doi: 10.20944/preprints202503.2237.v1

Keywords: robust PCA; graph Laplacian; anchor points; face recognition; dimensionality reduction

Article

Dual graph Laplacian RPCA method for face recognition based on anchor points

Shu-Ting Zhuang [1], Qing-Wen Wang [1,2,*] and Jiang-Feng Chen[1]

[1] Department of Mathematics and Newtouch Center for Mathematics, Shanghai University, Shanghai 200444, China
[2] Collaborative Innovation Center for the Marine Artificial Intelligence, Shanghai 200444, China
[*] Correspondence: wqw@shu.edu.cn

Abstract: High-dimensional data often contain noise and redundancy, which can significantly undermine the performance of machine learning. To address this challenge, we propose an advanced robust principal component analysis (RPCA) model that integrates bidirectional graph Laplacian constraints alongwith the anchor point technique. This approach constructs two graphs from both the sample and feature perspectives for a more comprehensive capture of the underlying data structure. Moreover, the anchor point technique serves to substantially reduce computational complexity, making the model more efficient and scalable. Experiments conducted on the GTdatabase demonstrate that our model maintains high accuracy and improves efficiency, particularly under challenging conditions like varying illumination and pose. The method enhances dimensionality reduction and robustness in face recognition, making it suitable for large-scale applications.

Keywords: robust PCA; graph Laplacian; anchor points; face recognition; dimensionality reduction

0. Introduction

With the rapid advancement of artificial intelligence technology, machine learning has found extensive applications in various domains, including information retrieval, person re-identification, and face recognition. In execution of these tasks, data are frequently represented as high-dimensional, inevitably exhibiting correlations and containing substantial redundancy and noisy features. The valuable characteristics may undermine the performance of subsequent tasks, such as clustering, classification, retrieval, or reconstruction. Hence, high-dimensional data pose the "curse of dimensionality" challenge to machine learning algorithms. To enhance the efficacy of machine learning methods, it is imperative to employ feature selection techniques to eliminate irrelevant, redundant, and noisy features from high-dimensional data.

Face recognition has been a prominent area of focus in machine learning. Principal component analysis (PCA) and its various variants have been successfully used for face recognition [1–7]. Sirovich and Kirby firstly applied PCA to efficiently represent the face images in a lower-dimensional space. Based on the Karhunen-Loeve procedure for the characterization of human faces [1], Turk and Pentland presented the eigenface method for face recognition [2]. Subsequently, Yang et. al., [3] proposed a novel technique known as 2DPCA to enhance the recognition rate of conventional PCA. Early PCA methods primarily addressed grayscale images by representing each image as a vector. Specifically, consider a data matrix $X = (x_1, ..., x_n) \in \mathbb{R}^{m\times n} (m \gg n)$ with rows representing features and columns representing samples. PCA is typically utilized to identify the optimal principal directions $U = (u_1, ..., u_k) \in \mathbb{R}^{m\times k} (U^T U = I)$ that define the low-dimensional (k-dim) subspace. The projected data points with the low subspace U can be denoted as the matrix $V^T = (v_1, ..., v_n) \in \mathbb{R}^{k\times n}$. Then the traditional PCA finds U and V with the following constrained problem

$$\arg\min_{U,V} \| \mathrm{X} - \mathrm{UV}^T \|_F^2, \; s.t. \; V^T V = I. \tag{1}$$

However, the performance of traditional PCA is typically poor. Outliers, which are prevalent in many scenarios due to factors such as sudden intense interference during transmission, sensor failures, and calibration errors, can significantly impact the results. To address this problem, robust PCA (RPCA) [8] has been proposed to decompose the observed matrix data into a sum of low-rank and sparse matrices, ensuring that no abnormal data disrupts the system. Intuitively, the RPCA can be formulated as the following minimization problem

$$\underset{L,S}{\arg\min}\, \text{rank}(L) + \lambda\|S\|_0,\ s.t.\ X = L + S. \tag{2}$$

To address (2) more effectively, researchers relaxed the rank function by the nuclear norm [9,10], thereby transforming the problem into a convex optimization problem, as follows

$$\underset{L,S}{\arg\min}\, \|L\|_* + \lambda\|S\|_1,\ s.t.\ X = L + S. \tag{3}$$

Although the convex model can accurately yield low-rank and sparse matrices under relatively mild conditions, it overlooks the presence of the noise with small magnitudes. This oversight is significant because, in real-world scenarios, data often suffer from contamination by the noise. To address this limitation, Zhou et. al. [11] introduced a method that decomposed the data matrix into three components: a low-rank matrix, a sparse matrix, and a noise matrix

$$\underset{U,V,S}{\arg\min}\, \|X - UV - S\|_F^2 + \lambda\|S\|_1,\ s.t.\ X = L + S + G,\ L = UV. \tag{4}$$

However, color information is not fully exploited, despite being a crucial characteristic for enhancing image discriminability. To address this, Torres et al. [4] extended traditional PCA to color face recognition by applying it separately to the color channels, with the final result obtained by fusing the outcomes from all three channels. Building upon this foundation, Yang et al. [5] proposed a general discriminant model for color face recognition, which employs a set of color component combination coefficients to merge the three color channels into a single channel, representing color face images. Furthermore, Xiang et al. [6] introduced a matrix-representation model for color images based on the PCA framework and applied 2DPCA to compute optimal projection vectors for feature extraction. A more generalized approach, as opposed to the aforementioned channel-wise processing techniques, involves representing color images through quaternion matrices, which inherently encode color channels in a holistic manner. This avoids artificial separation of RGB components and preserves inter-channel correlations. Recent works have utilized quaternion PCA (QPCA) for dimensionality reduction [12–18]. Notably, Wang et al. [19] demonstrated that the matrix equation AXB = C is fundamental to optimizing quaternion-based models, where solutions over quaternions and their extensions enable efficient color image processing while preserving color integrity, particularly in feature extraction and multi-channel image encryption. In complementary technical advancement, Jia et al. [20] constructed general solutions for split-quaternion tensor equations, extending the processing scope to dynamic videos while significantly enhancing computational efficiency and security through the pseudo-Euclidean properties of split quaternions.

The purpose of these non-linear dimensionality reduction techniques is to find a representation of points in a low-dimensional space, in which all points still maintain the similarity in the lower-dimensional space. In recent years, optimization models that combine linear and non-linear dimensionality reduction methods, especially graph Laplacian embedding, have demonstrated their effectiveness. Cai et al. [21] proposed a graph regularized non-negative matrix factorization (GNMF) method, which combined graph structure and non-negative matrix factorization for an improved compact representation of the original data. Building upon this, Jiang et al. [22] developed graph Laplacian PCA (GLPCA), which sought a low-dimensional representation of image data with significant improvement in clustering and image reconstruction by incorporating graph structures and PCA.

Further advancing this line of research, Feng et al. [23] employed PGLPCA based on graph Laplacian regularization and Lp-norm for feature selection and tumor clustering. Parallel developments include Liu et al.'s [24] graph Laplacian matrix formulation for semi-supervised feature extraction and Wang et al.'s [25] Laplacian regularized low-rank representation (LLRR), which successfully captures the intrinsic geometric structure of gene expression data for improved tumor sample clustering. Building on these methods, Yang et al. [26] developed an innovative online algorithm using Monte Carlo sampling for sparse graph-constrained matrix optimization, effectively solving feature selection problems while maintaining manifold structures.

To fully exploit the spatial and spectral features of images, we propose constructing two complementary graphs. One graph captures temporal or sample-based relationships among superpixels e.g., the columns of the data matrix X, while the other encodes spatial relationships among pixel locations e.g., the rows of the data matrix X. By integrating these two graphs, we effectively harness both spatial and spectral information, resulting in a more comprehensive representation of the image data. This dual-graph strategy enhances the model's capacity to capture the inherent structure of the data, thereby improving performance in tasks like image segmentation, classification, and reconstruction.

Furthermore, to expedite the construction of the graph Laplacian matrix and streamline the computational process, we introduce the notion of anchor points. These anchor points serve as representative samples that encapsulate the data's structure, significantly reducing the number of pairwise comparisons needed during graph construction. Instead of calculating relationships between all data points, we select a subset of anchor points (using methods such as random sampling, K-means, or other clustering techniques) and build the graph based on the relationships between data points and these anchors. This approach not only reduces computational complexity but also preserves the geometric and relational properties of the data, enabling the method to scale to large datasets without compromising the quality of the embeddings or the performance of downstream tasks.

The main contributions of our work are outlined as follows

1. **Integration of Graph Laplacian Embedding with RPCA** We incorporate graph Laplacian embedding into RPCA to account for the spatial information inherent in the data. By representing the data as a graph, where nodes correspond to data points and edges reflect pairwise relationships (such as similarity or distance), the graph Laplacian matrix effectively captures the dataset's underlying geometric structure.
2. **Exploitation of Two-Sided Data Structure** We leverage a dual perspective by obtaining the graph Laplacian from both the sample and feature dimensions. This approach enables us to capture intrinsic relationships not only among data points (samples) but also among features, thereby providing a more holistic representation of the data.
3. **Introduction of Anchors for Computational Efficiency** We introduce the concept of anchors to enhance the model's running speed and reduce computational complexity. Anchors serve as representative points that summarize the data's structure, thereby minimizing the number of pairwise comparisons needed during graph construction. Instead of computing relationships between all data points, we select a subset of anchors (using methods like random sampling, K-means, or other clustering techniques) and construct the graph based on the relationships between data points and these anchors. This method significantly reduces the size of the adjacency matrix and, consequently, the computational burden associated with the graph Laplacian.

The rest of the paper is organized as follows. Section 1 provides an overview of the fundamental theoretical foundations including quaternion and quaternion matrix, graphs, graph Laplacian embedding and anchor point technique. In Section 2, we present the proposed algorithm and methodology. Subsequently, Section 3 demonstrates the experimental results using real-world facial image datasets. Finally, Section 4 concludes the paper with a summary of our findings and contributions.

Table 1. Notations and Descriptions.

Notation	Description
X	data matrix of size $m \times n$
n	number of samples
m	number of features
x_i	the i-th column of X
x^i	the i-th row of X
$Tr(A)$	the trace norm of the matrix A
$\|\|A\|\|_F$	the Frobenius norm of the matrix A

1. Preliminaries

1.1. Quaternion and Quaternion Matrix

Introduced by mathematician Hamilton in 1843 [27], quaternion generalizes complex numbers by incorporating one real part and three imaginary parts. A quaternion can be expressed as

$$a = a_0 + a_1 i + a_2 j + a_3 k, \tag{5}$$

where $a_0, a_1, a_2, a_3 \in \mathbb{R}$, and i, j, k are three imaginary units that follow the multiplication rules

$$i^2 = j^2 = k^2 = -1,\ ij = -ji = k,\ jk = -kj = i,\ ki = -ik = j. \tag{6}$$

The conjugate and modulus of quaternion a are defined as follows

$$\bar{\mathrm{a}} \;=\; \mathrm{a}_0 - \mathrm{a}_1\mathrm{i} - \mathrm{a}_2\mathrm{j} - \mathrm{a}_3\mathrm{k}, \tag{7}$$

$$|a| = \sqrt{(a_0)^2 + (a_1)^2 + (a_2)^2 + (a_3)^2}. \tag{8}$$

A quaternion matrix $A \in \mathbb{Q}^{m\times n}$ takes the form

$$A = A_0 + A_1 i + A_2 j + A_3 k, \tag{9}$$

with $A_0, ..., A_3 \in \mathbb{R}^{m\times n}$, and its conjugate transposed matrix is given by $A^* = A_0^T - A_1^T i - A_2^T j - A_3^T k$. A pure quaternion matrix provides a representation for color images, with its three imaginary components A_1, A_2, A_3 corresponding to the red, green, and blue channels, respectively. Furthermore, the real representation method is a widely used technique for transforming quaternion matrices into the corresponding real matrices, the real representation of the matrix A is shown as

$$\gamma_\mathrm{A} = \begin{pmatrix} \mathrm{A}_0 & -\mathrm{A}_2 & -\mathrm{A}_1 & -\mathrm{A}_3 \\ \mathrm{A}_2 & \mathrm{A}_0 & -\mathrm{A}_3 & \mathrm{A}_1 \\ \mathrm{A}_1 & \mathrm{A}_3 & \mathrm{A}_0 & -\mathrm{A}_2 \\ \mathrm{A}_3 & -\mathrm{A}_1 & \mathrm{A}_2 & \mathrm{A}_0 \end{pmatrix}, \tag{10}$$

and the first block column of γ_A is denoted by

$$\gamma_{A,c} = \begin{bmatrix} A_0 \\ A_2 \\ A_1 \\ A_3 \end{bmatrix}. \tag{11}$$

Properties 1.1. *For $A, B \in \mathbb{Q}^{m\times n}, C \in \mathbb{Q}^{n\times s}$, the following properties hold [28].*

1. $\gamma_{A+B} = \gamma_A + \gamma_B,\ \gamma_{AC} = \gamma_A \gamma_C$.
2. $\gamma_{A^*} = \gamma_A^T$, *where $(\cdot)^T$ denotes transpose operation.*
3. *A is a column unitary matrix if and only if γ_A is a column orthogonal matrix.*

1.2. Graph

Let $G = (V, E)$ be an undirected weighted graph. The weight value between v_i and v_j is denoted by w_{ij}. If there is no edge between v_i and v_j, i.e., $(v_i, v_j) \notin E$, then $w_{ij} = 0$. The matrix $W = (w_{ij})(i, j = 1, ..., n)$ is called the adjacency matrix of G. We define the degree matrix D, which is a diagonal matrix whose diagonal entry D_{ii} equals to the sum of weights of all edges incident to i, i.e., $d_i = \sum_{j=1}^{n} w_{ij}$. Subsequently, the graph Laplacian matrix is defined as follows

$$L = D - W. \tag{12}$$

In some applications, the normalized graph Laplacian is used, defined as

$$L_{norm} = I - D^{-\frac{1}{2}} W D^{-\frac{1}{2}}. \tag{13}$$

Properties 1.2. *Let L denote a graph Laplacian matrix. Then the following properties hold [29].*

1. *For each vector* $(f_1, ..., f_n)' = f \in \mathbb{R}^n$, *we have*

$$f^{'} L f = \frac{1}{2} \sum_{i=1}^{n} w_{ij} (f_i - f_j)^2. \tag{14}$$

2. *L is symmetric and positive semi-definite.*
3. *The smallest eigenvalue of L is* 0, *and corresponding eigenvector is* $\mathbf{1}$ *whose elements are all ones.*
4. *L has n non-negative real eigenvalues* $0 = \lambda_1 \le \lambda_2 \le ... \le \lambda_n$.

1.3. Graph Laplacian Embedding

Graph Laplacian embedding has emerged as a widely-used technique in nonlinear manifold learning, aiming to preserve the local geometric structure of data. The underlying assumption is that points which are close in the original data space should maintain their proximity in the embedded space. To achieve this, a nearest neighbor graph is constructed to capture and model the local relationships among data points.

Given the data matrix $X = (x_1, ..., x_n) \in \mathbb{R}^{m \times n}$, where x_i $(i = 1, ..., n)$ denotes a data sample or one vertex in the graph. For each data point x_i, we connect each x_i to its k nearest neighbors. Here, we adopt Euclidean distance $w_{ij} = \sqrt{\sum_{d=1}^{m} (x_{i,d} - x_{j,d})^2}$ to measure the similarity between the data samples.

Let $Z = (z_1, ..., z_n) \in \mathbb{R}^{k \times n}$ represent embedding coordinates of data samples $X = (x_1, ..., x_n) \in \mathbb{R}^{m \times n}$. The dissimilarity of the two data points in the lower-dimensional space can be measured by the Euclidean distance. Define the dissimilarity of the two data points in the lower-dimensional space as $d(z_i, z_j) = ||z_i - z_j||^2$, combining with the weight matrix W, the smoothness of the low-dimensional representation can be measured by minimizing the following equation [30]

$$\begin{aligned} \mathrm{S} &= \frac{1}{2} \sum_{i,j=1}^{n} \|z_i - z_j\|^2 w_{ij} \\ &= \sum_{i=1}^{n} z_i^T z_i D_{ii} - \sum_{i,j=1}^{n} z_i^T z_j w_{ij} \\ &= Tr(V^T D V) - Tr(V^T W V) = Tr(V^T L V), \end{aligned} \tag{15}$$

where $D = \mathrm{diag}(d_1, ..., d_n)$ is a diagonal matrix, and $L = D - W$ is the graph Laplacian matrix.

1.4. Anchors

Anchor point techniques play a critical role in data representation and learning, effectively reducing computational costs while capturing the underlying structure of the data. In machine learning, anchor points are a set of representative samples or features selected from a dataset. Given a dataset $X = (x_1, x_2, ..., x_n) \in \mathbb{R}^{m\times n}$, anchor points $A = (a_1, a_2, ..., a_k) \in \mathbb{R}^{m\times k}$ with $k \ll n$, act as reference points in the data space, which can be strategically selected based on prior knowledge (e.g., class centers) or learned through methods like K-means clustering. Anchor points facilitate the construction of similarity matrices or lower-dimensional embeddings, thereby reducing computational complexity and enhancing model generalization, particularly for large-scale data. Following this introduction, we will explore two methodologies for deriving these anchor points, which are crucial for subsequent steps in constructing similarity matrices.

1. **K-means method for anchor points**
   We adopt the K-means clustering algorithm to derive a set of anchor points, which serve as representative prototypes for the underlying data distribution. Considering a dataset $X = (x_1, x_2, ..., x_n) \in \mathbb{R}^{m\times n}$, the K-means algorithm aims to partition X into k disjoint clusters $C = \{C_1, C_2, ..., C_k\}$ by minimizing the within-cluster sum of squares, formulated as [31]

$$f(x) = \sum_{i=1}^{k} \sum_{x\in C_i} \|x - u_i\|^2, \tag{16}$$

   where C_i denotes the i-th cluster and u_i is its centroid, namely the anchor points.
2. **BKHK method for anchor points**
   The Balanced and Hierarchical K-means (BKHK) algorithm is a hierarchical anchor point selection method that combines K-means and hierarchical clustering to recursively construct evenly distributed anchor points, thereby improving representation ability. In contrast to conventional K-means algorithms that execute a single-step partitioning of the dataset into a predetermined number of clusters, BKHK employs a hierarchical partitioning strategy. It recursively divides the dataset X into m sub-clusters, performing binary K-means at each step. This process continues until the desired number of clusters is achieved. The objective function of Balanced Binary K-means is defined as follows [32]

$$\underset{H\in\mathbb{B}^{n\times 2}, 1_n^T H = [\lfloor \frac{n}{2} \rfloor, n - \lfloor \frac{n}{2} \rfloor]}{\arg\min} \|X - CH^T\|_F^2, \tag{17}$$

   where $C \in \mathbb{R}^{d\times 2}$ represents the two class centers, and $H \in \mathbb{B}^{n\times 2}$ is the class indicator matrix.

The construction of the similarity matrix based on representative anchors is a well-explored problem. In our approach, we choose to measure the distance using the Euclidean distance. To be specific, our objective is to learn a similarity matrix $Z \in \mathbb{R}^{n\times m}$, in a way that a smaller distance corresponds to a larger affinity value. This allows us to formulate the optimization problem as follows

$$\underset{z^i 1_m = 1, z^i > 0}{\arg\min} \sum_{j=1}^{m} \|x_i - u_j\|_2^2 + \gamma \sum_{j=1}^{m} z_{ij}^2. \tag{18}$$

In many cases, we prefer Z to be a sparse matrix, such that z^i has k nonzero values corresponding to the k-nearest anchors. Consequently, the maximal γ is determined by solving the following problem

$$\max_{\gamma, \|z^i\|_0 = k} \gamma. \tag{19}$$

Suppose $e_{ij} = \|x_i - u_i\|_2^2$, and e_{ij} are arranged from small to large. The solution to (19) is $\gamma = \frac{k}{2} e_{i,k+1} - \frac{1}{2}\sum_{j=1}^{k} e_{ij}$. Therefore, the solution to (18) can be obtained as follows

$$z_{ij} = \begin{cases} \frac{e_{i,k+1} - e_{ij}}{k e_{i,k+1} - \sum_{h=1}^{k} e_{ih}}, & j \le k, \\ 0, & j > k. \end{cases} \tag{20}$$

After obtaining the matrix Z, we construct the similarity matrix A using the formula

$$A = Z\Delta^{-1}Z^T. \tag{21}$$

Here, $\Delta \in \mathbb{R}^{m\times m}$ is a diagonal matrix with the j-th diagonal element calculated as $\sum_{i=1}^{n} z_{ij}$. In practical applications, we can obtain A without explicitly computing it by utilizing $A = PP^T$, where $P = Z\Delta^{-\frac{1}{2}}$.

2. Methodology

The sample dataset comprises l color facial images, where n labeled images form the training set X, denoted as $X = \{X_1, ..., X_n\}$, and $l - n$ unlabeled images constitute the test set Y, denoted as $Y = \{X_{n+1}, ..., X_l\}$. Each color facial image is initially represented as a quaternion matrix $Q = Q_0 + Q_1 i + Q_2 j + Q_3 k \in \mathbb{Q}^{p\times q}$, where $\mathbb{Q}$ denotes the quaternion space. To facilitate numerical computation, the quaternion matrix Q is transformed into its real representation matrix $\gamma_Q \in \mathbb{R}^{4p\times 4q}$. Due to the information redundancy of the real representation matrix, the first column block of γ_Q, $[Q_0, Q_2, Q_1, Q_3]^T$ is vectorized into $X_i \in \mathbb{R}^{m\times 1}$, where $m = 4p \times q$. The data matrix X and the test matrix Y are constructed by column-wise concatenation of these vectors, resulting in $X = \{X_1, ..., X_n\} \in \mathbb{R}^{m\times n}$ and $Y = \{X_{n+1}, ..., X_l\} \in \mathbb{R}^{m\times (l-n)}$.

In robust principal component analysis (RPCA), it is well-established that data signals are typically decomposed into a sum of low-rank terms, sparse terms, and noise terms that are statistically modeled. This decomposition can be mathematically expressed as

$$X = L + S + E. \tag{22}$$

The columns of matrix L span a low dimensional subspace, while matrix S contains only a sparse set of non-zero entries. Typically, it is assumed that the elements of matrix E are drawn from a Gaussian distribution with a mean of zero. The low-rank component L can be factorized as $L = UV^T$, allowing equation (22) to be reformulated as

$$X = UV^T + S + E, \tag{23}$$

where $U \in \mathbb{R}^{m\times k}, V \in \mathbb{R}^{n\times k}$ and $k \ll m, n$. It is important to note that, due to the inherent constraints imposed by this factorization, a low-rank structure is implicitly enforced on matrix L, even though k might still exceed the true rank of L. In equation (23), the columns of matrix U define the low-dimensional subspace in which the columns of matrix L reside, while matrix V represents the corresponding coefficients within this subspace.

To capture the geometric structures of the sample and feature manifolds in facial data, we use two graphs, namely a sample graph for columns and a feature graph for rows, each tailored to its respective dimension.

Initially, a k-nearest-neighbors sample graph is constructed, with its vertices corresponding to the data points $(x_1, ..., x_n)$. We adopt a 0-1 weighting scheme to establish the k-nearest-neighbors data graph. Consequently, the sample weight matrix can be formally defined as

$$[W_V]_{ij} = \begin{cases} 1, & if\ x_j \in N(x_i), \\ 0, & otherwise, \end{cases} \quad i, j = 1, ..., n, \tag{24}$$

where $N(x_i)$ is the k-nearest neighbor of x_i and $L_V = D_V - W_V$ is the graph Laplacian matrix of the sample, where $[D_V]_{ii} = \sum_j [W_V]_{ij}$ is a diagonal degree matrix.

Similarly to the methodology employed for constructing the sample weight matrix, we obtain the feature weight matrix as follows

$$[W_U]_{ij} = \begin{cases} 1, & if\ x^{j,T} \in N(x^{i,T}), \\ 0, & otherwise, \end{cases} \quad i, j = 1, ..., m. \tag{25}$$

The feature graph Laplacian matrix is defined as $L_U = D_U - W_U$.

Via integrating graph regularization for both sample and feature manifolds with the RPCA model, we propose a novel dual graph regularized principal component analysis model. This model is designed to simultaneously capture the inherent geometric structures within the data across both manifolds. The corresponding objective function is formulated as follows

$$\begin{gathered}\min_{U,V} \|X - UV^T - S\|_F^2 + \gamma Tr(U^T L_U U) + \gamma Tr(V^T L_V V) + \lambda \|S\|_1, \\ s.t.\ U^T U = I,\end{gathered} \tag{26}$$

where the parameter γ is used to balance the contribution of the graph Laplacian regularization term and the parameter λ is used to balance the contribution of the sparse term.

We employ the Augmented Lagrangian Multiplier (ALM) method to optimize the objective function. This approach reformulates the constrained optimization problem into a sequence of unconstrained subproblems by incorporating Lagrange multipliers and penalty terms. Through this iterative process, the method progressively converges towards the optimal solution. In the context of utilizing the ALM method to derive the optimal solution, we substitute Z for $UV^T + S$, and (26) can be equivalently reformulated as follows

$$\begin{gathered}\min_{U,V} \|X - Z\|_F^2 + \gamma Tr(U^T L_U U) + \gamma Tr(V^T L_V V) + \lambda \|S\|_1, \\ s.t.\ U^T U = I,\ Z = UV^T + S.\end{gathered} \tag{27}$$

According to the ALM method, (27) can be equivalently minimized as

$$\begin{aligned}L_{\mu,\nu,\Upsilon,\Gamma}(S,Z,U,V) =& \|X - Z\|_F^2 + \gamma Tr(U^T L_U U) + \gamma Tr(V^T L_V V) \\ &+ \lambda \|S\|_1 + Tr(\Upsilon^T (Z - UV^T - S)) + \frac{\mu}{2}\|Z - UV^T - S\|_F^2 \\ &+ Tr(\Gamma^T (U^T U - I)) + \frac{\nu}{2}\|U^T U - I\|_F^2,\end{aligned} \tag{28}$$

where Γ and Λ are Lagrange multipliers, and μ and ν are the step size in the update rule.

Given that there are four variables requiring solution, the Alternating Direction Method (ADM) is employed to address this problem. It simplifies the solution process by allowing us to solve for a single variable while keeping the others fixed. By applying this method, the optimization problem represented by (28) can be naturally decomposed into four subproblems.

Problem 1: With variables U, V, Z fixed, the variable S is solved by rewriting (28)

$$L_{\mu,\nu,\Upsilon,\Gamma}(S,Z,U,V) = \lambda \|S\|_1 + \frac{\mu}{2}\|Z - UV^T - S + \frac{\Upsilon}{\mu}\|_F^2. \tag{29}$$

(29) can be solved by the proximal shrink operator denoted as follows

$$\begin{gathered}S^{k+1} = \text{soft}(Z^{k+1} - U^{k+1}V^{k+1,T} + \frac{\Upsilon^k}{\mu}, \frac{\lambda}{\mu}), \\ \text{soft}(a, \tau) = \text{sign}(a)\max(|a| - \tau, 0).\end{gathered} \tag{30}$$

Problem 2: With variables V, S, Z fixed, the variable U is solved by rewriting (28)

$$\begin{aligned}L_{\mu,\nu,\Upsilon,\Gamma}(S,Z,U,V) =& \gamma Tr(U^T L_U U) + Tr(\Upsilon^T (Z - UV^T - S)) \\ &+ \frac{\mu}{2}\|Z - UV^T - S\|_F^2 + Tr(\Gamma^T (U^T U - I)) + \frac{\nu}{2}\|U^T U - I\|_F^2,\end{aligned} \tag{31}$$

$$
\begin{aligned}
L =&\gamma Tr(U^T L_U U) - Tr(V^T Y^T U) - \mu Tr(Z^T U V^T) + \frac{\mu}{2} Tr(V U^T U V^T) \\
&+ \mu Tr(V^T S^T U) + Tr(\Gamma^T U^T U) + \frac{\nu}{2} \|U^T U - I\|_F^2.
\end{aligned}
\tag{32}
$$

By computing the partial derivative with respect to the variable U in (32), we obtain

$$
\begin{aligned}
\frac{\partial L}{\partial U} &= 2\gamma L_U U - YV - \mu Z V + \mu U(V^T V) \\
&+ \mu S V + 2U\Gamma + \nu(2UU^T U - U).
\end{aligned}
\tag{33}
$$

By setting the equation (33) equal to 0 , the updated iteration formula for variable U can be addressed via the Sylvester equation, as showed below

$$
(2\gamma L_U + \nu I)U + U(\mu V^T V + 2\Gamma) = YV + \mu(Z - S)V. \tag{34}
$$

Problem 3: With variables U, S, Z fixed, the variable V is solved by rewriting (28)

$$
\begin{aligned}
L_{\mu,\nu,Y,\Gamma}(S, Z, U, V) =&\gamma Tr(V^T L_V V) + Tr(Y^T(Z - UV^T - S)) \\
&+ \frac{\mu}{2} \|Z - UV^T - S\|_F^2,
\end{aligned}
\tag{35}
$$

$$
\begin{aligned}
L =&\gamma Tr(V^T L_V V) - Tr(V^T Y^T U) - \mu Tr(Z^T U V^T) \\
&+ \frac{\mu}{2} Tr(V U^T U V^T) + \mu Tr(V^T S^T U).
\end{aligned}
\tag{36}
$$

Similar to the procedure outlined in (33), we derive the partial derivative of the variable U in equation (36)

$$
\frac{\partial L}{\partial V} = 2\gamma L_V V - Y^T U - \mu Z^T U + \mu V(U^T U) + \mu S^T U. \tag{37}
$$

The updated iteration formula for variable V also involves solving the Sylvester equation, as detailed below

$$
2\gamma L_V V + V(\mu U^T U) = Y^T U + \mu(Z^T - S^T)U. \tag{38}
$$

Problem 4: With variables U, V, S fixed, the variable Z is solved by rewriting (28)

$$
L_{\mu,\nu,Y,\Gamma}(S, Z, U, V) = \|X - Z\|_F^2 + \frac{\mu}{2} \|Z - UV^T - S + \frac{Y}{\mu}\|_F^2. \tag{39}
$$

The problem described by equation (39) is convex, which facilitates a straightforward update of the variable Z as follows

$$
Z^{k+1} = \frac{2X + \mu(UV^T + S - \frac{Y}{\mu})}{2 + \mu}. \tag{40}
$$

Algorithm 1 The solution to optimize (26)

Require: Facial image data matrix: $X_{m \times n}$, Parameters: $k, r, \lambda, \gamma, \mu, \nu$
Ensure: $U_{m \times k}, V_{n \times k}$

```
Initialize $Z, Y, S, U, V$
repeat
    Update $S$ by (30).
    Update $U$ by (34).
    Update $V$ by (38).
    Update $Z$ by (40).
    Update $Y$ by $Y^{r+1} = Y^r + \mu^r(Z^T - UV^T - S)$.
    Update $\mu$ by $\mu^{r+1} = \rho\mu^r$.
until convergence
```

3. Experiments

3.1. Datasets

The proposed model is rigorously evaluated on the Georgia Tech face dataset [33], a well-established benchmark in face recognition research comprising facial images from 50 distinct subjects, with each subject contributing 15 unique samples. These samples exhibit significant variations in lighting conditions, facial expressions (e.g., neutral, smiling, surprised), and head poses (e.g., frontal, side views), rendering the dataset highly effective for assessing the robustness and generalization capabilities of face recognition models. This diversity facilitates robust assessment of model generalization under challenging real-world conditions. All images in the GTdatabase are preprocessed by manual cropping and resizing to a uniform resolution of 44 × 33 pixels, with representative samples visualized in Fig.1.

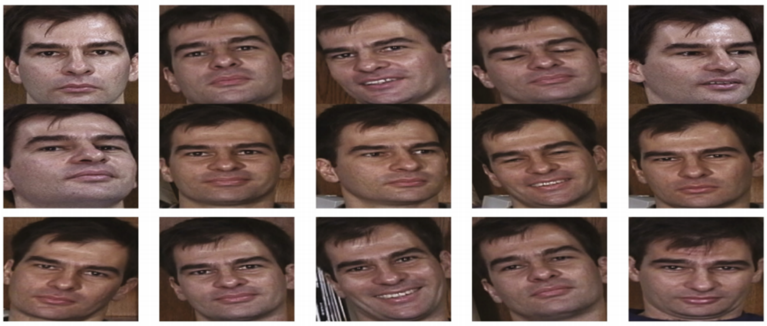

Figure 1. Sample images for one individual of the GTdatabase.

3.2. Parameter Selection

Our model involves six critical parameters: $\gamma, \lambda, \mu, \nu, k$ and r, which must be carefully tuned in the formulation (28). Based on empirical evidence from prior studies and our experimental validation, we set $\gamma = 0.005, \mu = 0.005$ and $\nu = 0.001$. For the regularization parameter γ, we adopt $\lambda = \frac{1}{\sqrt{\max(m,n)}}$ as suggested in [5]. Additionally, we select $k = 15$ as the number of nearest neighbors for graph construction, ensuring a balanced representation of local data structures. Furthermore, we respectively set $r = 200$ and $r = 100$ to compare the final results. These parameter choices are carefully calibrated to optimize the model's performance across diverse scenarios.

3.3. Experiment Results

Using the learned projection matrix U, we project the test samples into the low-dimensional space and compute their similarity to the training samples in this reduced space using Euclidean distance. This enables nearest neighbor classification, where recognition results are determined by identifying the closest matches based on similarity. To evaluate the model's performance, we employ two metrics: the standard accuracy (ACC) and the convergence time. ACC measures the face matching accuracy, defined as the ratio of correctly classified samples to the total number of test samples, providing a quantitative assessment of the model's recognition capability. Specifically, ACC is calculated as

$$ACC = \frac{\sum_{i=1}^{n} \sigma(p_i, map(q_i))}{n} \times 100\%, \tag{41}$$

where p_i is the predicted label of a test sample and $map(q_i)$ is the real label of the test sample. $\sigma(x, y)$ equals 1 if $x = y$ and 0 otherwise.

For the 50 × 15 samples in the GTdatabase, we partitioned the dataset into training and test sets with a ratio of 0.7. To ensure statistical reliability and robustness, the results reported are the average values obtained from 10 independent trials, providing a more precise and representative assessment of the model's performance.

For the $k = 15$-nearest neighbors, we set the dimensionality reduction parameter to $r = 200$. The corresponding accuracy and runtime of the model, evaluated using the original graph Laplacian matrix, the graph Laplacian matrix constructed via K-means, and the graph Laplacian matrix generated by the BKHK method, are summarized in Table 2.

Table 2. Results of RPCA with graph Laplacian constraint(r = 200).

Method	Accuracy (%)	Running Time (s)	Notes
Standard graph Laplacian	73.8%	991.94	Baseline method
K-means + graph Laplacian	73.3%	**935.39**	K-means for anchor selection
BKHK + graph Laplacian	73.3%	973.15	BKHK for anchor selection

Similarly, for the $k = 15$-nearest neighbors, we conducted an additional experiment with the dimensionality reduction parameter set to $r = 100$. The accuracy and runtime of the model with original graph Laplacian matrix, graph Laplacian matrix constructed via K-means, and graph Laplacian matrix generated by the BKHK method are summarized in Table 3.

Table 3. Results of RPCA with graph Laplacian constraint(r = 100).

Method	Accuracy (%)	Running Time (s)	Notes
Standard graph Laplacian	73.3%	2924.71	Baseline method
K-means + graph Laplacian	72.9%	2765.83	K-means for anchor selection
BKHK + graph Laplacian	72.9%	**2132.40**	BKHK for anchor selection

From the experimental results summarized in Table 2 and Table 3, we can draw the following conclusions. First, all three methods—standard graph Laplacian, K-means-based graph Laplacian, and BKHK-based graph Laplacian—demonstrate high recognition accuracy, with the standard method exhibiting slightly superior performance due to its ability to capture the underlying data structure more comprehensive. Second, the anchor-based methods, particularly the BKHK approach, exhibit significantly lower computational complexity, especially in lower-dimensional settings (e.g., r = 100). This efficiency advantage positions the BKHK-based method as a highly scalable solution for large-scale datasets, where computational efficiency is paramount. In conclusion, while all methods deliver comparable accuracy, the anchor-based approaches, particularly BKHK, provide substantial computational savings, making them more suitable for real-world applications requiring both performance and scalability.

4. Conclusions

In this study, we present an advanced RPCA framework incorporating bidirectional graph Laplacian constraints and an anchor point strategy, achieving enhanced computational efficiency while preserving recognition accuracy in facial analysis. The dual-graph architecture captures intrinsic geometric relationships from both sample and feature perspectives, maintaining structural and discriminative characteristics of high-dimensional data. Anchor points effectively reduce computational complexity without sacrificing robustness. Extensive experiments on the GTdatabase demonstrate superior processing speed and sustained accuracy under challenging conditions including illumination variations, expressions, and pose changes. Our work advances dimensionality reduction techniques while providing a practical solution for face recognition. Future research will extend this framework to larger datasets and explore broader applications in computer vision.

Author Contributions: Conceptualization, S.-T.Z. and J.-F.C.; methodology, S.-T.Z. and J.-F.C.; software, S.-T.Z. and J.-F.C.; validation, S.-T.Z. and J.-F.C.; formal analysis, S.-T.Z.; resources, Q.-W.W.; data curation, S.-T.Z.; writing—original draft preparation, Q.-W.W. and S.-T.Z.; writing—review and editing, Q.-W.W., S.-T.Z. and J.-F.C.; visualization, S.-T.Z. and J.-F.C.; supervision, Q.-W.W.; project administration, Q.-W.W.; funding acquisition, Q.-W.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research is supported by the National Natural Science Foundation of China (No. 12371023)

Data Availability Statement: Data are contained within the article.

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

References

1. Kirby, M.; Sirovich, L. Application of the Karhunen-Loeve procedure for the characterization of human faces. *IEEE Trans. Pattern Anal. Mach. Intell.*, **1990**, 12, 103–108.
2. Turk, M.; Pentland, A. Eigenfaces for recognition. *J. Cognit. Neurosci.*, **1991**, 3, 71-86.
3. Yang, J.; Zhang, D.; Frangi, A.F.; Yang, J.Y. Two-dimensional PCA: A new approach to appearance-based face representation and recognition. *IEEE Trans. Pattern Anal. Mach. Intell.*, **2004**, 26, 131–137.
4. Torres, L.; Reutter, J.Y.; Lorente, L. The importance of the color information in face recognition. *IEEE International Conference on Image Processing*, **1999**, 3, 627–631.
5. Yang, J.; Liu, C. A general discriminant model for color face recognition. *IEEE 11th International Conference on Computer Vision*, **2007**, 1–6.
6. Xiang, X.; Yang, J.; Chen, Q. Color face recognition by PCA-like approach. *Neurocomputing*, **2015**, 228, 231–235.
7. Zhu, Y.; Zhu, C.; Li, X. Improved principal component analysis and linear regression classification for face recognition. *Signal Process.*, **2018**, 145, 175–182.
8. Wright, J.; Ganesh, A.; Rao, S.; Peng, Y.; Ma, Y. Robust Principal Component Analysis: Exact Recovery of Corrupted Low-rank Matrices via Convex Optimization. *Proc, Adv. Neural Inf. Process. Syst.*, **2009**, 2080-2088.
9. Candès, E.J.; Li, X.; Ma, Y.; Wright, J. Robust principal component analysis. *J. ACM*, **2009**, 58, 1–37.
10. Wright, J.; Ma, Y. Dense error correction via l_1-minimization. *IEEE Trans. Inf. Theory*, **2010**, 56(7), 3540-3560.
11. Zhou, T.; Tao, D. Greedy Bilateral Sketch, Completion & Smoothing. *Proc. IMLR Workshop Conf*, **2013**, 31, 650-658.
12. Bihan, N.Le; Sangwine, S.J. Quaternion principal component analysis of color images. *Proc. Int. Conf. Image Process.*, **2003**, 809–812.
13. Jia, Z. The Eigenvalue Problem of Quaternion Matrix: Structure-Preserving Algorithms and Applications. Beijing, China: *Science Press*, **2019**.
14. Denis, P.; Carre, P.; Fernandez-Maloigne, C. Spatial and spectral quaternionic approaches for colour images. *Comput. Vis. Image Understand.*, **2007**, 107, 74–87.
15. Shi,L.; Funt, B. Quaternion color texture segmentation. *Comput. Vis. Image Understand.*, **2007**,107, 88–96.
16. Zou,C.; Kou, K.I.; Wang, Y. Quaternion collaborative and sparse representation with application to color face recognition. *IEEE Trans. Image Process.*, **2016**, 25, 3287–3302.
17. Xiao, X.; Chen,Y.; Gong, Y.J.; Zhou, Y. 2D quaternion sparse discriminant analysis. *IEEE Trans. Image Process.*, **2020**, 29, 2271–2286.
18. Chen,Y.; Xiao,X.; Zhou,Y. Low-rank quaternion approximation for color image processing. *IEEE Trans. Image Process.*, **2020**, 29, 1426–1439.
19. Wang, Q.W.; Xie, L.M.; Gao, Z.H. A Survey on Solving the Matrix Equation AXB = C with Applications, *Mathematics*, **2025**, 13(3), 450.
20. Jia, Z.R.; Wang, Q.W. The General Solution to a System of Tensor Equations over the Split Quaternion Algebra with Applications. *Mathematics*, **2025**, 13(4), 644.
21. Cai, D.; He, X.F.; Han, J.W.; Huang, T.S. Graph regularized nonnegative matrix factorization for data representation. *IEEE Trans. Pattern Anal. Mach. Intell.*, **2011**, 33, 1548–1560.
22. Jiang, B.; Ding, C.; Luo, B.; Tang, J. Graph-laplacian PCA: closed-form solution and robustness. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, **2013**.
23. Feng, C.; Gao, Y.L.; Liu, J.X.; Zheng, C.H.; Yu, J. PCA based on graph laplacian regularization and P-norm for gene selection and clustering. *IEEE Trans. Nanobiosci.*, **2017**, 16, 257–265.
24. Liu, J.X.; Wang, D.; Gao, Y.L.; Zheng, C.H.; Shang, J.L.; Liu, F.; et al. A joint-$L_{2,1}$-norm-constraint-based semi-supervised feature extraction for RNA Seq data analysis. *Neurocomputing*, **2017**, 228, 263–269.
25. Wang, J.; Liu, J.X.; Kong, X.Z.; Yuan, S.S.; Dai, L.Y. Laplacian regularized low-rank representation for cancer samples clustering. *Comput. Biol. Chem.*, **2019**, 78, 504–509.
26. Yang, N.Y.; Duan, X.F.;Li, C.M.; Wang, Q.W. A new algorithm for solving a class of matrix optimization problem arising in unsupervised feature selection. *Numer. Algor.*, **2024**.
27. Hamilton, W.R. Lectures on Quaternions. In *Landmark Writings in Western Mathematics 1640–1940*; Hodges and Smith: Dublin, Ireland, **1853**.

28. Jiang, T. Algebraic methods for diagonalization of a quaternion matrix in quaternionic quantum theory. *J. Math. Phys.*, **2005**, 46, 052106.
29. Ding, L.; Li, C.; Jin, D.; Ding, S.F. Survey of spectral clustering based on graph theory, *Pattern Recognition*, **2024**, 151, 110366.
30. Kong, X.Z.; Song, Y.; Liu, J.X.; Zheng, C.H.; Yuan, S.S.; Wang, J.; Dai, L.Y. Joint Lp-Norm and $L_{2,1}$-Norm Constrained Graph Laplacian PCA for Robust Tumor Sample Clustering and Gene Network Module Discovery. *Front. Genet.*, **2021**, 12, 621317.
31. Zhao, H.l. Design and Implementation of an Improved K-Means Clustering Algorithm. *Mob. Inf. Syst.*, **2022**, 6041484.
32. Gao, C.H.; Chen, W.Z.; Nie, F.P.; Yu, W.Z.; Wang, Z.H. Spectral clustering with linear embedding: A discrete clustering method for large-scale data. *Pattern Recognit.*, **2024**, 151, 110396.
33. The Georgia Tech face database. http://www.anefian.com/research/face_reco.htm

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.