Yokohama Publishers
ISSN 1349-8169 ONLINE JOURNAL

Pacific Journal of Optimization
Vol. 20, No. 2, pp. 265–285, 2024

COLOR LINEAR DISCRIMINANT ANALYSIS FOR FACE RECOGNITION BASED ON QUATERNION MODEL*

Si-Tao Ling, Zhe-Han Hu, Bing Yang, Yi-Ding Li†

Abstract: In this paper, a color linear discriminant analysis (color LDA) method is presented for color face reconstruction and color face recognition. Based on quaternion vectors rather than real vectors, the color LDA method takes the color information of color images into consideration. In order to circumvent the singularity problem of within-class scatter matrix, the quaternion generalized singular value decomposition (QGSVD) is used to obtain the whole Fisher projection axes, and an alternative criterion is adopted to select the optimal set of Fisher projection axes such that the extracted features are statistically uncorrelated. Taking the computational cost and storage of quaternion operations into account, the structure-preserving strategy is adopted to accelerate the color LDA method, and the fast color LDA based on randomization is proposed. Numerical results based on famous color face databases are provided to validate the feasibility and effectiveness of the proposed color LDA-like methods.

Key words: *color linear discriminant analysis; quaternion generalized singular value decomposition; structure-preserving strategy; randomization*

Mathematics Subject Classification: *65F15; 65F10; 15A18*

1 Introduction

Face recognition has been one of the most flourishing research topics due to its broad social, scientific, commercial applications. A complete face recognition system includes five parts: acquisition and preprocessing, face detection, feature extraction, feature selection and feature matching. The feature extraction procedure acts as a core task for any face recognition system. It has a significant effect on overall system performance. As effective feature extraction tools, linear discriminant analysis (LDA) and principle component analysis (PCA) have achieved great success for face recognition.

The LDA method was first developed by R. A. Fisher [4] for taxonomic classification. The original idea of LDA lies in finding a projection vector w, after the projection of image matrix onto w, the maximum between-class scatter and the minimum within-class scatter are achieved simultaneously. Generally, the single projection axis, even if it is optimal in theory, is not sufficient because much discriminative information has been lost after images

*This work is partially supported by the National Natural Science Foundation of China under grants 12171210, 11771188, and the Open Project of Key Laboratory of Data Science and Intelligence Education of the Ministry of Education under grant DSIE202203.

†Corresponding author.

 DOI: https://doi.org/10.61208/pjo-2023-047

theory, is not sufficient because much discriminative information has been lost after images being projected onto it alone. Accordingly, Foley and Sammon [5, 19] proposed the Foley-Sammon transform method to find an optimal set of discriminant vectors for LDA that attracts many researchers [2, 28]. From numerical algebra point of view, the execution of LDA can be depicted as finding generalized eigenvectors of the following generalized eigen-equation

$$S_b x = \lambda S_w x,$$

where x denotes the Fisher projection axis, and the matrices S_b, S_w represent between-class scatter matrix and within-class scatter matrix, respectively.

The major drawback of applying LDA is that it may encounter the so-called small size sample problem (i.e., S_w is singular), whenever the number of samples is smaller than the dimensionality of the samples. Many efficient algorithms have emerged to circumvent this problem, such as pseudo-inverse LDA [21], PCA+LDA [1], and regularized LDA. The two stage method PCA+LDA first discards the null space of S_w by PCA, so that S_w no longer degenerates and then LDA can be performed without trouble [14, 15]. In addition to these methods, the uncorrelated LDA (ULDA) method [11, 25] provides a novel insight of LDA that the extracted feature vectors are mutually uncorrelated in the low-dimensional subspaces. This property is highly desirable for feature extraction in many applications to contain minimum redundancy [24].

It is verified that color image provides more information than gray image [18]. The structure of color face image can be retained by a pure quaternion matrix perfectly [3, 12, 20]. But for the presence of the limitation of computation and memory while dealing with high-dimensional quaternion matrix, the face recognition method based on quaternion may achieve high recognition rate at the expense of running time. Recently, the applications of quaternion in color face recognition have been developed rapidly, including two-dimensional quaternion PCA [7, 26, 27] and quaternion singular value decomposition (QSVD) based on Lanczos method [8]. These developments mainly benefit from the structure-preserving strategy of quaternion matrix operations, which makes full use of the structure of real representations of quaternion matrices, and can decrease the amount of computation time compared with direct quaternion matrix operations. For more details about quaternion structure-preserving strategy we refer to [9].

In this paper, we present a new color LDA approach based on quaternion model for feature extraction of color images, and further design a fast and efficient structure-preserving method that improves the performance of the new method. We also consider randomization-based color LDA method for quite large sample set. The contributions of this paper are summarized in three aspects.

- We establish a trace-ratio under quaternion framework as the objective function for dimension reduction, and make an analysis for the small samples size problem. We put forward a weighted norm as the distance measure for the feature matrices that help us avoid generating projection matrices explicitly in color image recognition.
- We propose quaternion generalized singular value decomposition (QGSVD) method for solutions of color LDA model that leads to the color LDA method. We elaborate the structure-preserving strategy for performing the color LDA method. A quaternion random Gaussian matrix is borrowed to form randomization-based structure-preserving color LDA method for quite large sample set, which makes the color image recognition rate promoted with less CPU time.
- Based on two famous color face databases, we compare the proposed color LDA method

with other traditional LDA methods and representation-based methods [29, 30] for the implementation of color face recognition and color image reconstruction. Numerical results indicate that the structure-preserving color LDA method and its randomization have better recognition performances.

The rest of the paper is organized as follows. In section 2, we introduce color LDA method for color face recognition and color image reconstruction using quaternion model. In section 3, we elaborate computational issues that use real structure-preserving operations for color LDA, and propose a randomization strategy. In section 4, we report numerical results based on the famous color face database to test the feasibility and effectiveness of the proposed methods. Finally, the conclusion is presented in section 5.

Notations. In order to distinguish from the symbols of real matrices which are denoted by capital letters, we use boldfaced capital letters to represent quaternion matrices. $\mathbb{H}^{m\times n}$ and $\mathbb{R}^{m\times n}$ denote the set of $m\times n$ matrices with quaternion and real entries, respectively. $\mathbb{H}^m$ and $\mathbb{R}^m$ are short for $\mathbb{H}^{m\times 1}$ and $\mathbb{R}^{m\times 1}$, respectively. The superscript ^{T} and ^{H} denote the transpose and conjugate transpose of a matrix or a vector, respectively. $\texttt{rank}(\mathbf{A})$ and $\texttt{tr}(\mathbf{A})$ denote the rank and trace of a quaternion matrix $\mathbf{A}$, respectively. I_n denotes the $n\times n$ identity matrix. $\|\cdot\|_2$ denotes the matrix spectral norm or vector 2-norm.

2 Color LDA Based on Quaternion Model

In this section, we make a brief review of quaternions and quaternion matrices, then elaborate the color LDA method based on quaternion model.

2.1 Quaternions and quaternion matrices

Quaternions and quaternion matrices have extensive applications in many research fields. A quaternion $\mathbf{q}$ has the following form

$$\mathbf{q} = q_0 + q_1\mathbf{i} + q_2\mathbf{j} + q_3\mathbf{k} \in \mathbb{H},$$

where q_0, q_1, q_2 and q_3 are four real numbers, and the three imaginary units $\mathbf{i}, \mathbf{j}, \mathbf{k}$ satisfy

$$\begin{aligned}&\mathbf{i}^2 = \mathbf{j}^2 = \mathbf{k}^2 = \mathbf{ijk} = -1,\\ &\mathbf{jk} = -\mathbf{kj} = \mathbf{i}, \mathbf{ki} = -\mathbf{ik} = \mathbf{j}, \mathbf{ij} = -\mathbf{ji} = \mathbf{k}.\end{aligned}$$

This indicates that the quaternion skew-field $\mathbb{H}$ is an associative but non-commutative algebra. The conjugate of $\mathbf{q}\in\mathbb{H}$ is given by $\bar{\mathbf{q}} = q_0 - q_1\mathbf{i} - q_2\mathbf{j} - q_3\mathbf{k}$, and the modulus of $\mathbf{q}$ is defined by $|\mathbf{q}|^2 = q_0^2 + q_1^2 + q_2^2 + q_3^2 = \mathbf{q}\bar{\mathbf{q}} = \bar{\mathbf{q}}\mathbf{q}$. A pure quaternion is a special quaternion whose real part is zero.

An $n\times n$ square quaternion matrix $\mathbf{P} = P_0 + P_1\mathbf{i} + P_2\mathbf{j} + P_3\mathbf{k}$ is said to be nonsingular if $\mathbf{PQ} = \mathbf{QP} = I_n$ for some $\mathbf{Q}\in\mathbb{H}^{n\times n}$.

Definition 2.1 ([22, 23])**.** For a given quaternion matrix $\mathbf{A}\in\mathbb{H}^{m\times n}$, there exists a quaternion matrix $\mathbf{G}\in\mathbb{H}^{n\times m}$ satisfying all of the following Penrose functions

$$\begin{aligned}&(1)\mathbf{AGA} = \mathbf{A}; &&(2)\mathbf{GAG} = \mathbf{G};\\ &(3)(\mathbf{AG})^H = \mathbf{AG}; &&(4)(\mathbf{GA})^H = \mathbf{GA}.\end{aligned}$$

Then, $\mathbf{G}$ is the Moore–Penrose inverse of $\mathbf{A}$, denoted by $\mathbf{A}^\dagger$.

For a given quaternion matrix $\mathbf{Q} = Q_0 + Q_1\mathbf{i} + Q_2\mathbf{j} + Q_3\mathbf{k} \in \mathbb{H}^{m\times n}$ with $Q_0, Q_1, Q_2, Q_3 \in \mathbb{R}^{m\times n}$, the real representation matrix of $\mathbf{Q}$ is defined [9] to be

$$\Upsilon_{\mathbf{Q}} = \begin{bmatrix} Q_0 & Q_2 & Q_1 & Q_3 \\ -Q_2 & Q_0 & Q_3 & -Q_1 \\ -Q_1 & -Q_3 & Q_0 & Q_2 \\ -Q_3 & Q_1 & -Q_2 & Q_0 \end{bmatrix} \in \mathbb{R}^{4m\times 4n}, \tag{2.1}$$

where Υ is a linear homeomorphic mapping from quaternion matrices (or vectors) to their real representation. With the help of block permutation matrices

$$J_t = \begin{bmatrix} 0 & -I_t & 0 & 0 \\ I_t & 0 & 0 & 0 \\ 0 & 0 & 0 & I_t \\ 0 & 0 & -I_t & 0 \end{bmatrix},\ R_t = \begin{bmatrix} 0 & 0 & -I_t & 0 \\ 0 & 0 & 0 & -I_t \\ I_t & 0 & 0 & 0 \\ 0 & I_t & 0 & 0 \end{bmatrix},\ S_t = \begin{bmatrix} 0 & 0 & 0 & -I_t \\ 0 & 0 & I_t & 0 \\ 0 & -I_t & 0 & 0 \\ I_t & 0 & 0 & 0 \end{bmatrix},$$

the algebraic structure of (2.1) is called JRS-symmetric [9] in the sense that

$$J_m \Upsilon_{\mathbf{Q}} J_n^T = \Upsilon_{\mathbf{Q}},\ R_m \Upsilon_{\mathbf{Q}} R_n^T = \Upsilon_{\mathbf{Q}},\ S_m \Upsilon_{\mathbf{Q}} S_n^T = \Upsilon_{\mathbf{Q}}.$$

For two given quaternion matrices $\mathbf{P}, \mathbf{Q} \in \mathbb{H}^{n\times n}$, their real representations have the following properties

$$\Upsilon_{k_1\mathbf{P}+k_2\mathbf{Q}} = k_1\Upsilon_{\mathbf{P}} + k_2\Upsilon_{\mathbf{Q}},\ k_1, k_2 \in \mathbb{R}, \tag{2.2}$$

$$\Upsilon_{\mathbf{Q}^H} = \Upsilon_{\mathbf{Q}}^T, \quad \Upsilon_{\mathbf{PQ}} = \Upsilon_{\mathbf{P}}\Upsilon_{\mathbf{Q}}, \tag{2.3}$$

$$\mathtt{rank}(\mathbf{Q}) = \frac{1}{4}\mathtt{rank}(\Upsilon_{\mathbf{Q}}). \tag{2.4}$$

A quaternion matrix $\mathbf{Q}$ is called pure quaternion matrix if it has zero real part. In RGB color space, a pixel can be encoded by a pure quaternion $R\mathbf{i} + G\mathbf{j} + B\mathbf{k}$, where R, G and B stand for the pixel values of red, green and blue components, respectively. Then an $m \times n$ color image can be represented by an $m \times n$ pure quaternion matrix $\mathbf{Q}$. Each elements of $\mathbf{Q}$ represents a pixel of the color image.

2.2 Color LDA method

Now, we introduce the color LDA method using the quaternion model that is the improvement of [6].

Suppose that there are N training face image samples and c known classes in training set T_i, $i = 1, \ldots, c$. The j-th sample in training set can be denoted as $\mathbf{a}_j \in \mathbb{H}^{mn}$, which is the vectorized representation of the color image matrix. The mean image of all training samples is denoted by $\bar{\mathbf{a}}$. $\bar{\mathbf{a}}_i$ denotes the mean image of class i for $i = 1, \ldots, c$. n_i is the number of color image samples in class i satisfying $n_1 + \cdots + n_c = N$. With these notations in hand, we introduce three scatter matrices of training set, which are within-class scatter matrix, between-class scatter matrix and total scatter matrix, as the following,

$$\mathbf{S_w} = \sum_{i=1}^{c} \sum_{\mathbf{a}_j \in T_i} (\mathbf{a}_j - \bar{\mathbf{a}}_i)(\mathbf{a}_j - \bar{\mathbf{a}}_i)^H,$$

$$\mathbf{S_b} = \sum_{i=1}^{c} n_i(\bar{\mathbf{a}}_i - \bar{\mathbf{a}})(\bar{\mathbf{a}}_i - \bar{\mathbf{a}})^H,$$

$$\mathbf{S_t} = \sum_{i=1}^{N} (\mathbf{a}_i - \bar{\mathbf{a}})(\mathbf{a}_i - \bar{\mathbf{a}})^H.$$

It is easy to verify that $\mathbf{S_t} = \mathbf{S_w} + \mathbf{S_b}$.

Suppose there is a projection axis $\mathbf{w} \in \mathbb{H}^{mn}$. The image vector $\mathbf{a}_j$ $(j = 1, \dots, N)$ can be projected onto $\mathbf{w}$ by the linear transformation $\mathbf{y}_j = \mathbf{w}^H \mathbf{a}_j$ $(j = 1, \dots, N)$. Then, $\mathbf{y}_j$ is called the projection feature or projection point, the corresponding projected class is P_i $(i = 1, \dots, c)$. $\bar{\mathbf{y}}, \bar{\mathbf{y}}_i$ denote the average of all projection points and those in the projected class P_i for $i = 1, \dots, c$, respectively. The aim of color LDA is to seek an optimal projection axis, such that the projection points of the same class shall be as close as possible, and the projection points of different samples shall be as far away as possible. Generally, the single projection axis, even if it is optimal in theory, is not sufficient because much discriminative information has been lost after images being projected onto it alone. Therefore, we should seek a set of optimal projection axes rather than only one, in this case, $\mathbf{y}_j = \mathbf{W}^H \mathbf{a}_j$ $(j = 1, \dots, N)$, where the columns of $\mathbf{W}$ are optimal projection axes. The intra-class distance and inter-class distance can be calculated by the trace of within-class scatter matrix and between-class scatter matrix. In this regard, the criterion of color LDA is given by

$$J = \max \frac{\mathtt{tr}(\tilde{\mathbf{S}}_\mathbf{b})}{\mathtt{tr}(\tilde{\mathbf{S}}_\mathbf{w})},$$

where $\tilde{\mathbf{S}}_\mathbf{b}$ and $\tilde{\mathbf{S}}_\mathbf{w}$ are between-class scatter matrix and within-class scatter matrix of projection space, respectively, derived from

$$\begin{aligned}
\tilde{\mathbf{S}}_\mathbf{b} &= \sum_{i=1}^{c} n_i (\bar{\mathbf{y}}_i - \bar{\mathbf{y}})(\bar{\mathbf{y}}_i - \bar{\mathbf{y}})^H, \\
&= \mathbf{W}^H \left(\sum_{i=1}^{c} n_i (\bar{\mathbf{a}}_i - \bar{\mathbf{a}})(\bar{\mathbf{a}}_i - \bar{\mathbf{a}})^H \right) \mathbf{W}, \\
&= \mathbf{W}^H \mathbf{S_b} \mathbf{W}, \\
\tilde{\mathbf{S}}_\mathbf{w} &= \sum_{i=1}^{c} \sum_{\mathbf{y}_j \in P_i} (\mathbf{y}_j - \bar{\mathbf{y}}_i)(\mathbf{y}_j - \bar{\mathbf{y}}_i)^H, \\
&= \mathbf{W}^H \left(\sum_{i=1}^{c} \sum_{\mathbf{a}_j \in T_i} (\mathbf{a}_j - \bar{\mathbf{a}}_i)(\mathbf{a}_j - \bar{\mathbf{a}}_i)^H \right) \mathbf{W}, \\
&= \mathbf{W}^H \mathbf{S_w} \mathbf{W}.
\end{aligned}$$

Alternatively, the criterion could be expressed by

$$J(\mathbf{W}) = \max_{\mathbf{W}} \frac{\mathtt{tr}(\mathbf{W}^H \mathbf{S_b} \mathbf{W})}{\mathtt{tr}(\mathbf{W}^H \mathbf{S_w} \mathbf{W})}. \tag{2.5}$$

This criterion can be seen as the promotion of Fisher linear projection criterion in quaternion skew-field. The columns of quaternion matrix $\mathbf{W}$ are called the Fisher optimal projection axes.

It is obvious that (2.5) requires the within-class scatter matrix $\mathbf{S_w}$ to be nonsingular. The optimization problem (2.5) can be solved by dealing with the quaternion generalized eigenvalue problem

$$\mathbf{S_b} \mathbf{w}_i = \lambda_i \mathbf{S_w} \mathbf{w}_i, \ \lambda_i \in \mathbb{R}, \ i = 1, \dots, d, \tag{2.6}$$

where $\mathbf{w}_i$ is the Fisher projection axis that consists of $\mathbf{W}$ and d is the number of projection axes that we want.

Define

$$\mathbf{H_b} = [\sqrt{n_1}(\bar{\mathbf{a}}_1 - \bar{\mathbf{a}}), \ldots, \sqrt{n_c}(\bar{\mathbf{a}}_c - \bar{\mathbf{a}})] \in \mathbb{H}^{mn\times c}, \tag{2.7}$$

$$\mathbf{H_w} = [\mathbf{A}_1 - \bar{\mathbf{a}}_1 e_1, \ldots, \mathbf{A}_c - \bar{\mathbf{a}}_c e_c] \in \mathbb{H}^{mn\times N}, \tag{2.8}$$

$$\mathbf{H_t} = [\mathbf{a}_1 - \bar{\mathbf{a}}, \ldots, \mathbf{a}_N - \bar{\mathbf{a}}] \in \mathbb{H}^{mn\times N}, \tag{2.9}$$

and $e_i = [1, \ldots, 1] \in \mathbb{R}^{1\times n_i}, \mathbf{A}_i = [\mathbf{a}_1, \ldots, \mathbf{a}_{n_i}], \mathbf{a}_j \in T_i$ $(j = 1, \ldots, n_i, i = 1, \ldots, c)$. It is obvious that $\mathtt{rank}(\mathbf{H_b}) = c-1, \mathtt{rank}(\mathbf{H_t}) = N-1$, and the scatter matrices can be expressed by $\mathbf{S_b} = \mathbf{H_b}\mathbf{H_b}^H, \mathbf{S_w} = \mathbf{H_w}\mathbf{H_w}^H, \mathbf{S_t} = \mathbf{H_t}\mathbf{H_t}^H$.

From (2.2) and (2.3), we have

$$\Upsilon_{\mathbf{S_w}} = \Upsilon_{\mathbf{H_w}}\Upsilon_{\mathbf{H_w}}^T,\ \Upsilon_{\mathbf{S_b}} = \Upsilon_{\mathbf{H_b}}\Upsilon_{\mathbf{H_b}}^T,\ \Upsilon_{\mathbf{S_t}} = \Upsilon_{\mathbf{H_t}}\Upsilon_{\mathbf{H_t}}^T,$$

and $\mathtt{rank}(\Upsilon_{\mathbf{S_w}}) = \mathtt{rank}(\Upsilon_{\mathbf{H_w}}), \mathtt{rank}(\Upsilon_{\mathbf{S_t}}) = \mathtt{rank}(\Upsilon_{\mathbf{H_t}})$ and $\mathtt{rank}(\Upsilon_{\mathbf{S_b}}) = \mathtt{rank}(\Upsilon_{\mathbf{H_b}})$. According to (2.4) we obtain $\mathtt{rank}(\mathbf{S_b}) = \mathtt{rank}(\mathbf{H_b}) = c - 1, \mathtt{rank}(\mathbf{S_t}) = \mathtt{rank}(\mathbf{H_t}) = N - 1$ and $\mathtt{rank}(\mathbf{S_w}) = \mathtt{rank}(\mathbf{H_w}) = \min\{mn, N - c\}$. So $\mathbf{S_t}$ and $\mathbf{S_b}$ must be singular.

As a matter of fact, the number of training samples is usually smaller than the dimensionality of the samples, this means $\mathtt{rank}(\mathbf{S_w}) = N - c < nm$. So $\mathbf{S_w}$ is always singular, which is the so-called "small samples size" problem. In this case, we can not even define the Fisher criterion (2.5), and the discriminant fails.

We aim to seek a solution that does not impose the restriction of singularity problem, and that can be found without explicitly forming $\mathbf{S_b}$ and $\mathbf{S_w}$, respectively. Toward that end, let $\lambda_i = \alpha_i^2/\beta_i^2$. Then (2.6) becomes

$$\mathbf{H_b}\mathbf{H_b}^H\mathbf{w}_i\beta_i^2 = \mathbf{H_w}\mathbf{H_w}^H\mathbf{w}_i\alpha_i^2. \tag{2.10}$$

This problem can be solved using the QGSVD, as described in the next subsection.

2.3 QGSVD applied to color LDA

The following theoretical results are about QGSVD of two given quaternion matrices with the same number of columns.

Theorem 2.2 ([10], QGSVD)**.** *For any quaternion matrices* $\mathbf{A} \in \mathbb{H}^{m\times n}, \mathbf{B} \in \mathbb{H}^{p\times n}$*, there exist unitary quaternion matrices* $\mathbf{U} \in \mathbb{H}^{m\times m}$ *and* $\mathbf{V} \in \mathbb{H}^{p\times p}$*, and a nonsingular matrix* $\mathbf{X} \in \mathbb{H}^{n\times n}$ *such that*

$$\mathbf{U}^H\mathbf{A}\mathbf{X} = \mathtt{diag}(\alpha_1, \ldots, \alpha_t), \alpha_i \geq 0,$$
$$\mathbf{V}^H\mathbf{B}\mathbf{X} = \mathtt{diag}(\beta_1, \ldots, \beta_q), \beta_i \geq 0,$$

where $\beta_1 \geq \ldots \geq \beta_s \geq \beta_{s+1} = \ldots = \beta_q = 0, t = \min\{m, n\}, q = \min\{p, n\}$*.*

According to the proof of Theorem 2.2, we outline the QGSVD algorithm as below.

Algorithm 1 Quaternion Generalized Singular Value Decomposition

Input: Two quaternion matrices with the same number of columns

$$\mathbf{A} \in \mathbb{H}^{m\times n},\ \mathbf{B} \in \mathbb{H}^{p\times n}.$$

Output: The nonsingular quaternion matrix $\mathbf{X} \in \mathbb{H}^{n\times n}$, and two unitary quaternion matrices $\mathbf{U} \in \mathbb{H}^{m\times m}, \mathbf{V} \in \mathbb{H}^{p\times p}$.

1: Compute the QSVD of $\mathbf{C} = [\mathbf{A}; \mathbf{B}] \in \mathbb{H}^{(m+p)\times n}$:

$$\mathbf{P}^H \mathbf{C}\mathbf{Q} = \begin{bmatrix} R & 0 \\ 0 & 0 \end{bmatrix}.$$

2: Let $t = \texttt{rank}(\mathbf{C})$. Compute $\mathbf{U}$ from the QSVD of $\mathbf{P}(1:m, 1:t)$:

$$\mathbf{U}^H \mathbf{P}(1:m, 1:t)\mathbf{W} = \Gamma_a.$$

3: Compute the QR decomposition of $\mathbf{P}(m+1:m+p, 1:t)\mathbf{W}$:

$$\mathbf{V}^H(\mathbf{P}(m+1:m+p, 1:t)\mathbf{W}) = \Gamma_b.$$

4: Construct

$$\mathbf{X} = \mathbf{Q}\begin{bmatrix} R^{-1}\mathbf{W} & 0 \\ 0 & I \end{bmatrix},$$

where I is the identity matrix of appropriate size.

According to Theorem 2.2, for quaternion matrix pencil $\{\mathbf{H}_{\mathbf{b}}^H, \mathbf{H}_{\mathbf{w}}^H\}$, we can find two unitary quaternion matrices $\mathbf{U}, \mathbf{V}$ and a nonsingular quaternion matrix $\mathbf{X}$, such that

$$\mathbf{U}^H \mathbf{H}_{\mathbf{b}}^H \mathbf{X} = [\Gamma_b \quad 0] \quad \text{and} \quad \mathbf{V}^H \mathbf{H}_{\mathbf{w}}^H \mathbf{X} = [\Gamma_w \quad 0], \tag{2.11}$$

$$\Gamma_b^T \Gamma_b + \Gamma_w^T \Gamma_w = I_k, \quad \text{for} \quad k = \texttt{rank}\left(\begin{bmatrix} \mathbf{H}_{\mathbf{b}}^H \\ \mathbf{H}_{\mathbf{w}}^H \end{bmatrix}\right).$$

The diagonal matrices Γ_b and Γ_w has the following forms

$$\Gamma_b = \begin{array}{c} \begin{array}{ccc} r & s & k-r-s \end{array} \\ \begin{bmatrix} I_b & & \\ & D_b & \\ & & O_b \end{bmatrix} \end{array} \begin{array}{c} r \\ s \\ c-r-s \end{array},$$

$$\Gamma_w = \begin{array}{c} \begin{array}{ccc} r & s & k-r-s \end{array} \\ \begin{bmatrix} O_w & & \\ & D_w & \\ & & I_w \end{bmatrix} \end{array} \begin{array}{c} N-k+r \\ s \\ k-r-s \end{array},$$

where

$$D_b = \texttt{diag}(\alpha_{r+1}, \ldots, \alpha_{r+s}),$$
$$D_w = \texttt{diag}(\beta_{r+1}, \ldots, \beta_{r+s}),$$

satisfies

$$1 > \alpha_{r+1} \geq \cdots \geq \alpha_{r+s} > 0,\ 0 < \beta_{r+1} \leq \cdots \leq \beta_{r+s} < 1,$$

and

$$\alpha_i^2 + \beta_i^2 = 1, \quad \text{for} \quad i = r+1, \ldots, r+s.$$

Combining with (2.10), the columns of the nonsingular quaternion matrix $\mathbf{X}$ are the Fisher projection axes that we need.

From the previous analysis, in color LDA method we do not need to compute a complete QGSVD, just the first two steps are enough. The algorithm for the color LDA is given as follows.

Algorithm 2 Color Linear Discriminant Analysis

Input: Color image vector $\mathbf{a}_j$ $(j = 1, \ldots, n_i)$ that represents the j-th sample of training set, the number of Fisher features d.

Output: The set of Fisher projection axes $\mathbf{X}$.

1: Compute $\mathbf{H_b} \in \mathbb{H}^{mn\times c}$ and $\mathbf{H_w} \in \mathbb{H}^{mn\times N}$ from training set according to (2.7) and (2.8), respectively.

2: Compute the QSVD of $\mathbf{C} = [\mathbf{H_b}, \mathbf{H_w}]^H \in \mathbb{H}^{(c+N)\times mn}$:

$$\mathbf{P}^H\mathbf{C}\mathbf{Q} = \begin{bmatrix} R & 0 \\ 0 & 0 \end{bmatrix}.$$

3: Let $k = \texttt{rank}(\mathbf{C})$. Compute $\mathbf{V}$ from the QSVD of $\mathbf{P}(1:c, 1:k)$:

$$\mathbf{U}^H\mathbf{P}(1:c, 1:k)\mathbf{V} = \Gamma_b.$$

4: Compute the nonsingular matrix

$$\mathbf{X} = \mathbf{Q}\begin{bmatrix} R^{-1}\mathbf{V} & 0 \\ 0 & I \end{bmatrix},$$

where I is the identity matrix of appropriate size.

2.4 Feature extraction and dimension reduction

Upon we obtain the optimal Fisher projection axes $\widetilde{\mathbf{X}} \in \mathbb{H}^{mn\times d}$, which consists of d columns of $\mathbf{X}$, we can perform the classification task and reconstruction task in practical applications motivated by the idea in [16].

In color image classification, given two color images vectors $\mathbf{a}_1, \mathbf{a}_2$ from training set and test set respectively. Their projection images are calculated by

$$\tilde{\mathbf{a}}_i = \widetilde{\mathbf{X}}(\widetilde{\mathbf{X}}^H\widetilde{\mathbf{X}})^{-1}\widetilde{\mathbf{X}}^H\mathbf{a}_i, \quad i = 1, 2.$$

Whether $\mathbf{a}_1$ and $\mathbf{a}_2$ belong to the same class or not is determined by the distance between $\tilde{\mathbf{a}}_1$ and $\tilde{\mathbf{a}}_2$, that is,

$$\begin{aligned} d(\tilde{\mathbf{a}}_1, \tilde{\mathbf{a}}_2) &= \|\tilde{\mathbf{a}}_1 - \tilde{\mathbf{a}}_2\|_2 = [(\tilde{\mathbf{a}}_1 - \tilde{\mathbf{a}}_2)^H(\tilde{\mathbf{a}}_1 - \tilde{\mathbf{a}}_2)]^{\frac{1}{2}} \\ &= [(\widetilde{\mathbf{X}}(\widetilde{\mathbf{X}}^H\widetilde{\mathbf{X}})^{-1}\widetilde{\mathbf{X}}^H\mathbf{a}_1 - \widetilde{\mathbf{X}}(\widetilde{\mathbf{X}}^H\widetilde{\mathbf{X}})^{-1}\widetilde{\mathbf{X}}^H\mathbf{a}_2)^H(\widetilde{\mathbf{X}}(\widetilde{\mathbf{X}}^H\widetilde{\mathbf{X}})^{-1}\widetilde{\mathbf{X}}^H\mathbf{a}_1 - \widetilde{\mathbf{X}}(\widetilde{\mathbf{X}}^H\widetilde{\mathbf{X}})^{-1}\widetilde{\mathbf{X}}^H\mathbf{a}_2)]^{\frac{1}{2}} \\ &= [(\widetilde{\mathbf{X}}(\widetilde{\mathbf{X}}^H\widetilde{\mathbf{X}})^{-1}\widetilde{\mathbf{X}}^H(\mathbf{a}_1 - \mathbf{a}_2))^H(\widetilde{\mathbf{X}}(\widetilde{\mathbf{X}}^H\widetilde{\mathbf{X}})^{-1}\widetilde{\mathbf{X}}^H(\mathbf{a}_1 - \mathbf{a}_2))]^{\frac{1}{2}} \\ &= [(\mathbf{a}_1 - \mathbf{a}_2)^H\widetilde{\mathbf{X}}(\widetilde{\mathbf{X}}^H\widetilde{\mathbf{X}})^{-1}\widetilde{\mathbf{X}}^H(\mathbf{a}_1 - \mathbf{a}_2)]^{\frac{1}{2}} \\ &= \|\widetilde{\mathbf{X}}^H(\mathbf{a}_1 - \mathbf{a}_2)\|_{(\widetilde{\mathbf{X}}^H\widetilde{\mathbf{X}})^{-1}}. \end{aligned} \tag{2.12}$$

Obviously, the distance between $\tilde{\mathbf{a}}_1$ and $\tilde{\mathbf{a}}_2$ can be characterized by a weighted norm. In practical calculations, we just need to compute the feature vector $\mathbf{b}_i = \widetilde{\mathbf{X}}^H\mathbf{a}_i$, called "feature extraction", so as to reduce computational cost.

It is obvious that the dimension of $\mathbf{a}_i$ decreases after the feature extraction step. Actually, if we want to decrease the dimension of $\mathbf{a}_i$ to a fixed value t, we can construct $\hat{\mathbf{X}} \in \mathbb{H}^{mn\times t}$ that consists of t columns of $\mathbf{X}$, and the dimension of $\mathbf{a}_i$ can be reduced by

$$\hat{\mathbf{b}}_i = \hat{\mathbf{X}}^H\mathbf{a}_i.$$

Conversely, the reconstructed image is given by

$$\hat{\mathbf{a}}_i = \hat{\mathbf{X}}(\hat{\mathbf{X}}^H\hat{\mathbf{X}})^{-1}\hat{\mathbf{b}}_i.$$

$\boxed{2.5}$ The selection of optimal Fisher projection axes

From the previous analysis, the optimal Fisher projection axes can be chosen from the columns of nonsingular quaternion matrix $\mathbf{X}$. Notice that the dimension of $\mathbf{X}$ is mn, it is unwise to choose $\mathbf{X}$ as the optimal Fisher projection axes. The central work in this subsection is how to select the most discriminative Fisher projection axes within the columns of $\mathbf{X}$.

In pattern recognition theory, the general principle of feature extraction is to make the statistical correlation between the extracted features as small as possible. It is the best to extract irrelevant features. With this regard, we first review some statistical results in quaternion skew-field.

Definition 2.3 ([17])**.** Suppose that $\mathbf{q} = q_0 + q_1\mathbf{i} + q_2\mathbf{j} + q_3\mathbf{k}$ is a quaternion probability variable, the expectation of $\mathbf{q}$ is given by

$$\mathbb{E}(\mathbf{q}) = \mathbb{E}(q_0) + \mathbb{E}(q_1)\mathbf{i} + \mathbb{E}(q_2)\mathbf{j} + \mathbb{E}(q_3)\mathbf{k}.$$

Definition 2.4 ([17])**.** Let $\mathbf{q} = q_0 + q_1\mathbf{i} + q_2\mathbf{j} + q_3\mathbf{k}$ and $\mathbf{p} = p_0 + p_1\mathbf{i} + p_2\mathbf{j} + p_3\mathbf{k}$ be two quaternion probability variables. The covariance between $\mathbf{p}$ and $\mathbf{q}$ is defined as

$$\begin{aligned}
&cov(\mathbf{p}, \mathbf{q}) = \mathbb{E}(\mathbf{q} - \mathbb{E}(\mathbf{q}))(\overline{\mathbf{p} - \mathbb{E}(\mathbf{p})})\\
&= cov(q_0, p_0) + cov(q_1, p_1) + cov(q_2, p_2) + cov(q_3, p_3)\\
&+ \{cov(q_1, p_0) - cov(q_0, p_1) - cov(q_2, p_3) + cov(q_3, p_2)\}\mathbf{i}\\
&+ \{cov(q_2, p_0) - cov(q_0, p_2) + cov(q_1, p_3) - cov(q_3, p_1)\}\mathbf{j}\\
&+ \{cov(q_3, p_0) - cov(q_0, p_3) + cov(q_1, p_2) - cov(q_2, p_1)\}\mathbf{k}.
\end{aligned}$$

From the definition of real representation (2.1), the Fisher criterion (2.5) can also be expressed as

$$J(\Upsilon_{\mathbf{W}}) = \max_{\Upsilon_{\mathbf{W}}} \frac{\mathtt{tr}(\Upsilon_{\mathbf{W}}^T \Upsilon_{\mathbf{S_b}} \Upsilon_{\mathbf{W}})}{\mathtt{tr}(\Upsilon_{\mathbf{W}}^T \Upsilon_{\mathbf{S_w}} \Upsilon_{\mathbf{W}})}, \tag{2.13}$$

and the following relation holds

$$\Upsilon_{\mathbf{S_t}} = \Upsilon_{\mathbf{S_w}} + \Upsilon_{\mathbf{S_b}}.$$

From the analysis in [15], the optimization problem (2.13) is equivalent to

$$J(\Upsilon_{\mathbf{W}}) = \max_{\Upsilon_{\mathbf{W}}} \frac{\mathtt{tr}(\Upsilon_{\mathbf{W}}^T \Upsilon_{\mathbf{S_b}} \Upsilon_{\mathbf{W}})}{\mathtt{tr}(\Upsilon_{\mathbf{W}}^T \Upsilon_{\mathbf{S_t}} \Upsilon_{\mathbf{W}})}, \tag{2.14}$$

where the real representation matrix $\Upsilon_{\mathbf{S_w}}$ of the within-class scatter matrix in the denominator is replaced by $\Upsilon_{\mathbf{S_t}}$.

Recall that the total scatter matrix $\mathbf{S_t}$ is singular. Motivated by [11] the optimization problem (2.14) can be approximated by

$$J(\Upsilon_{\mathbf{W}}) = \max_{\Upsilon_{\mathbf{W}}} \mathtt{tr}((\Upsilon_{\mathbf{W}}^T \Upsilon_{\mathbf{S_t}} \Upsilon_{\mathbf{W}})^{\dagger} (\Upsilon_{\mathbf{W}}^T \Upsilon_{\mathbf{S_b}} \Upsilon_{\mathbf{W}})), \tag{2.15}$$

which can overcome the singularity problem of $\mathbf{S_t}$. Thus, for the color LDA method, we can take an alternative criterion to approximate (2.5), i.e.,

$$\tilde{J}(\mathbf{W}) = \max_{\mathbf{W}} \mathtt{tr}((\mathbf{W}^H \mathbf{S_t} \mathbf{W})^{\dagger} (\mathbf{W}^H \mathbf{S_b} \mathbf{W})). \tag{2.16}$$

It is easy to confirm that the nonsingular matrix $\mathbf{X}$ constructed in Algorithm 2 can diagonalize $\mathbf{S_t}$ to be

$$\mathbf{X}^H\mathbf{S_t}\mathbf{X} = \begin{bmatrix} I_k & 0 \\ 0 & 0 \end{bmatrix}.$$

Combining with (2.11) we have

$$(\mathbf{X}^H\mathbf{S_t}\mathbf{X})^\dagger(\mathbf{X}^H\mathbf{S_b}\mathbf{X}) = \begin{bmatrix} I_k & 0 \\ 0 & 0 \end{bmatrix}\begin{bmatrix} \Gamma_b^T\Gamma_b & 0 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} I_b & & \\ & D_b^2 & \\ & & 0 \end{bmatrix},$$

which means $\tilde{J}(\mathbf{W}) = r + \sum_{i=r+1}^{r+s}\beta_i^2$. So the set of optimal Fisher projection axes consists of the leftmost $r+s$ columns of $\mathbf{X}$, where $r+s$ is the rank of $\mathbf{S_b}$.

Theorem 2.5. *Let the transformation matrix for color LDA be $\tilde{\mathbf{X}} = [\mathbf{x}_1, \ldots, \mathbf{x}_d]$, for some $d \geq 0$. The original color image vector $\mathbf{a}$ is transformed into $\mathbf{b} = \tilde{\mathbf{X}}^H\mathbf{a}$, where the i-th feature component of $\mathbf{b}$ is $\mathbf{b}_i = \mathbf{x}_i^H\mathbf{a}$. Assume that $\mathbf{x}_i$ and $\mathbf{x}_j$ are $\mathbf{S_t}$-orthogonal to each other, i.e., $\mathbf{x}_i^H\mathbf{S_t}\mathbf{x}_j = 0$ for $i \neq j$. Then the correlation between $\mathbf{b}_i$ and $\mathbf{b}_j$ is 0 for $i \neq j$. That is, $\mathbf{b}_i$ and $\mathbf{b}_j$ are uncorrelated to each other.*

Proof. The covariance between $\mathbf{b}_i$ and $\mathbf{b}_j$ can be computed as

$$\begin{aligned} Cov(\mathbf{b}_i, \mathbf{b}_j) &= \mathbb{E}(\mathbf{b}_i - \mathbb{E}\mathbf{b}_i)(\overline{\mathbf{b}_j - \mathbb{E}\mathbf{b}_j}) \\ &= \mathbb{E}(\mathbf{x}_i^H\mathbf{a} - \mathbb{E}(\mathbf{x}_i^H\mathbf{a}))(\overline{\mathbf{x}_j^H\mathbf{a} - \mathbb{E}(\mathbf{x}_j^H\mathbf{a})}) \\ &= \mathbf{x}_i^H\{\mathbb{E}(\mathbf{a} - \mathbb{E}(\mathbf{a}))(\mathbf{a} - \mathbb{E}(\mathbf{a}))^H\}\mathbf{x}_j \\ &= \mathbf{x}_i^H\mathbf{S_t}\mathbf{x}_j = 0. \end{aligned}$$

As a result, the correlation coefficient of $\mathbf{b}_i$ and $\mathbf{b}_j$ is zero from the formula

$$Corr(\mathbf{b}_i, \mathbf{b}_j) = \frac{cov(\mathbf{b}_i, \mathbf{b}_j)}{\sqrt{cov(\mathbf{b}_i, \mathbf{b}_i)}\sqrt{cov(\mathbf{b}_j, \mathbf{b}_j)}}.$$

$\square$

Theorem 2.5 means that the set of optimal Fisher projection axes chosen from criterion (2.16) can extract features that are statistically uncorrelated. This property is highly desirable for feature extraction in many applications in order to contain minimum redundancy.

3 Computational Issues for Color LDA

It is well known that the basic quaternion operations including addition and multiplication between quaternions. They usually need more computational cost and as a result waste more CPU time compared with those of real and complex operations. Some of the existing algorithms are based on the real or complex representation, which transform the quaternion matrix problem to that of real or complex matrix problem at the cost of dimensional expansion in quadruple or in double. Recently emerged structure-preserving algorithms take advantage of the block structure of real representation matrix (2.1), the algorithms only deal with real operations but never lead to dimensional expansion [23]. Explicitly, we just need to store and perform on the first block row of the real representation matrix, denoted by

$$\Upsilon_{\mathbf{Q}}^c = [Q_0, Q_2, Q_1, Q_3].$$

Suppose that $\mathbf{A} = A_0 + A_1\mathbf{i} + A_2\mathbf{j} + A_3\mathbf{k} \in \mathbb{H}^{m\times m}, \mathbf{B} = B_0 + B_1\mathbf{i} + B_2\mathbf{j} + B_3\mathbf{k} \in \mathbb{H}^{m\times m}$ and $\mathbf{C} = C_0 + C_1\mathbf{i} + C_2\mathbf{j} + C_3\mathbf{k} \in \mathbb{H}^{m\times m}$ are three quaternion matrices. It is easy to verify the following equivalent relations.

Addition: $\mathbf{C} = \mathbf{A} + \mathbf{B} \Leftrightarrow \Upsilon^c_{\mathbf{C}} = \Upsilon^c_{\mathbf{A}} + \Upsilon^c_{\mathbf{B}}$.

Multiplication: $\mathbf{C} = \mathbf{AB} \Leftrightarrow \Upsilon^c_{\mathbf{C}} = \Upsilon^c_{\mathbf{A}}\Upsilon_{\mathbf{B}} \Leftrightarrow$

$$\begin{aligned} C_0 &= A_0B_0 - A_2B_2 - A_1B_1 - A_3B_3, \\ C_2 &= A_0B_2 + A_2B_0 - A_1B_3 + A_3B_1, \\ C_1 &= A_0B_1 + A_2B_3 + A_1B_0 - A_3B_2, \\ C_3 &= A_0B_3 - A_2B_1 + A_1B_2 + A_3B_0. \end{aligned} \tag{3.1}$$

With these real operations, the JRS-symmetric structure of real representation matrix is always inherited. The quaternion algorithm carried out by virtue of this strategy is called "structure-preserving method". For example, by applying operational rules in (3.1), the multiplication between two quaternion matrices can be performed on real matrices by the following structure-preserving method.

Algorithm 3 `timesQ` (Structure-preserving Multiplication Between Two Quaternion Matrices).

Input: The first block row of real representation matrices of $\mathbf{A}$ and $\mathbf{B}$:

$$\Upsilon^c_{\mathbf{A}} = [A_0\ A_2\ A_1\ A_3], \quad \Upsilon^c_{\mathbf{B}} = [B_0\ B_2\ B_1\ B_3].$$

Output: The first block row of real representation matrix of $\mathbf{C} = \mathbf{AB}$.

1. Compute multiplications of real matrices by (3.1).

2. Construct $\Upsilon^c_{\mathbf{C}} = [C_0\ C_2\ C_1\ C_3]$.

It is seen that the color LDA algorithm (Algorithm 2) consists of two QSVD's. The structure-preserving QSVD algorithm is presented in [13] and it is divided into two stages. The first stage is about the structure-preserving bidiagonalization process that is summarized in the following theorem.

Theorem 3.1 ([13], Bidiagonalization)**.** *Suppose that $\mathbf{Q} \in \mathbb{H}^{m\times n}$ and $\Upsilon_{\mathbf{Q}}$ is the real representation of $\mathbf{Q}$. Then, there exists orthogonal JRS-symplectic matrices $\mathbf{U} \in \mathbb{R}^{4m\times 4m}$ and $\mathbf{V} \in \mathbb{R}^{4n\times 4n}$ such that*

$$\mathbf{U}^T\Upsilon_{\mathbf{Q}}\mathbf{V} = \begin{bmatrix} D & 0 & 0 & 0 \\ 0 & D & 0 & 0 \\ 0 & 0 & D & 0 \\ 0 & 0 & 0 & D \end{bmatrix},$$

where $D \in \mathbb{R}^{m\times n}$ is a bidiagonal matrix.

The detailed structure-preserving bidiagonalization process for a quaternion matrix is referred to [13, Algorithm 3.3–`Bidiagq`]. The second stage is about the SVD of real bidiagonal matrix D, which can be directly implemented by the MATLAB order `svd`.

3.1 Structure-preserving color LDA

Combining Algorithm 2 with the structure-preserving strategy, we propose the structure-preserving color LDA algorithm in this subsection.

Algorithm 4 Structure-preserving Color LDA

Input: Color image vector $\mathbf{a}_j$ $(j = 1, \ldots, n_i)$ that represents the j-th sample of training set, and the number of classes c.

Output: The set of Fisher optimal projection axes $\mathbf{G}$.

1: Compute $\mathbf{H_b} \in \mathbb{H}^{mn\times c}$ and $\mathbf{H_w} \in \mathbb{H}^{mn\times N}$ from training set according to (2.7) and (2.8), respectively.

2: Compute two JRS-symplectic matrices $\Upsilon_{\mathbf{P}_1}$ and $\Upsilon_{\mathbf{Q}_1}$ via `Bidiagq` in [13, Algorithm 3.3] such that

$$\Upsilon_{\mathbf{P_1}}^T \Upsilon_{\mathbf{C}} \Upsilon_{\mathbf{Q}_1} = \Upsilon_D,$$

where $\mathbf{C} = [\mathbf{H_b}, \mathbf{H_w}]^H \in \mathbb{H}^{(c+N)\times mn}$, and $D \in \mathbb{R}^{(c+N)\times mn}$ is a bidiagonal matrix.

3: Compute the SVD of D

$$P_2^T D Q_2 = \begin{bmatrix} R & 0 \\ 0 & 0 \end{bmatrix},$$

and obtain $\mathbf{Q} = \mathbf{Q}_1 Q_2, \mathbf{P} = \mathbf{P}_1 P_2$ via `timesQ`.

4: Let $t = \texttt{rank}(\mathbf{C})$. Bidiagonalize $\mathbf{P}(1 : c, 1 : t)$ via `Bidiagq` such that

$$\Upsilon_{\mathbf{U}_1}^T \Upsilon_{\mathbf{P}(1:c,1:t)} \Upsilon_{\mathbf{V}_1} = \Upsilon_{D_1},$$

where $\Upsilon_{\mathbf{U}_1}$ and $\Upsilon_{\mathbf{V}_1}$ are two JRS-symplectic matrices, and D_1 is a real bidiagonal matrix.

5: Compute the SVD of D_1

$$U_2^T D_1 V_2 = \begin{bmatrix} R_1 & 0 \\ 0 & 0 \end{bmatrix},$$

and obtain $\mathbf{V} = \mathbf{V}_1 V_2$ via `timesQ`.

6: Compute the nonsingular matrix

$$\mathbf{X} = \mathbf{Q} \begin{bmatrix} R^{-1}\mathbf{V} & 0 \\ 0 & I \end{bmatrix}$$

via `timesQ`, where I is the identity matrix of appropriate size.

7: Compute $\mathbf{G} = \mathbf{X}(:, 1 : c - 1)$.

Remark 3.2. Notice that for a given quaternion matrix $\mathbf{Q} \in \mathbb{H}^{m\times n}$, the algorithm `Bidiagq` is suitable for the case where $m \geq n$. In the practical case of $(c + N) \leq mn$, we should perform `Bidiagq` on the conjugate transpose of $\mathbf{C} = [\mathbf{H_b}, \mathbf{H_w}]^H$.

3.2 Structure-preserving Color LDA based on randomization

In practical data processing, the dimensions of $\mathbf{H}_b$ and $\mathbf{H}_w$ could be quite large. So Algorithm 4 may cost too much CPU time. In order to make Algorithm 4 faster, we propose the following structure-preserving color LDA method based on randomization. For quaternion matrix pencil $\{\mathbf{H_b}^H, \mathbf{H_w}^H\}$, if the number of rows are larger than columns, we aim to sample the rows of these two quaternion matrices by the aid of a quaternion random Gaussian matrix $\mathbf{\Omega} = \Omega_0 + \Omega_1\mathbf{i} + \Omega_2\mathbf{j} + \Omega_3\mathbf{k} \in \mathbb{H}^{(c+N)\times s}$, where the entries in $\Omega_0, \Omega_1, \Omega_2, \Omega_3$ are random and independently drawn from the $N(0, 1)$-normal distribution, and s is a target numerical rank.

Let $\mathbf{D} = [\mathbf{H_b}, \mathbf{H_w}] \in \mathbb{H}^{mn\times(c+N)}$. We sample $\mathbf{D}$ by $\mathbf{\Omega}$ to get $\tilde{\mathbf{D}} = \mathbf{D}\mathbf{\Omega} \in \mathbb{H}^{mn\times s}$ satisfying $s < \mathtt{rank}(\mathbf{D})$. Then we can obtain the approximately orthogonal basis matrix $\mathbf{\Phi}$ by the orthogonalization of $\tilde{\mathbf{D}}$. Since the sampling $\tilde{\mathbf{D}}$ is a rectangular matrix and it may not be of full rank, in order to guarantee good numerical stability, we use the quaternion QR decomposition that is implemented by the structure-preserving quaternion modified Gram-Schmidt (QMGS) on the columns of $\tilde{\mathbf{D}}$ [23, Chapter 2.4.3]. Once we get $\mathbf{\Phi}$ in hand, we reduce the dimensions of $\mathbf{H_w}$ and $\mathbf{H_b}$ by $\tilde{\mathbf{H}}_\mathbf{w} = \mathbf{\Phi}^H\mathbf{H_w}$ and $\tilde{\mathbf{H}}_\mathbf{b} = \mathbf{\Phi}^H\mathbf{H_b}$. The approximate Fisher projection axes can be computed by the QGSVD of quaternion matrix pencil $\{\tilde{\mathbf{H}}_\mathbf{b}^H, \tilde{\mathbf{H}}_\mathbf{w}^H\}$ via Algorithm 1 cooperated with the structure-preserving strategy. We summarize this process in Algorithm 5.

Algorithm 5 Structure-preserving Color LDA Based on Randomization

---

Input: Color image vector $\mathbf{a}_j$ that represents the j-th sample of training set for $j = 1, \ldots, n_i$, the target numerical rank s, and the number of classes c.

Output: The set of Fisher optimal projection axes $\mathbf{G}$.

1: Compute $\mathbf{H_b} \in \mathbb{H}^{mn\times c}$ and $\mathbf{H_w} \in \mathbb{H}^{mn\times N}$ from training set according to (2.7) and (2.8), respectively.

2: Let $\mathbf{D} = [\mathbf{H_b}, \mathbf{H_w}]$. Sample $\mathbf{D}$ to obtain $\tilde{\mathbf{D}} = \mathbf{D}\mathbf{\Omega} \in \mathbb{H}^{mn\times s}$ by a quaternion random Gaussian matrix $\mathbf{\Omega} \in \mathbb{H}^{(c+N)\times s}$.

3: Orthogonalize the columns of $\tilde{\mathbf{D}}$ by quaternion QR decomposition based on structure-preserving QMGS [23, Chapter 2.4.3] and obtain $\mathbf{\Phi} \in \mathbb{H}^{mn\times s}$.

4: Reduce the dimensions of $\mathbf{H_w}$ and $\mathbf{H_b}$ by $\tilde{\mathbf{H}}_\mathbf{w} = \mathbf{\Phi}^H\mathbf{H_w} \in \mathbb{H}^{s\times N}$, $\tilde{\mathbf{H}}_\mathbf{b} = \mathbf{\Phi}^H\mathbf{H_b} \in \mathbb{H}^{s\times c}$.

5: Compute two JRS-symplectic matrices $\Upsilon_\mathbf{P}$ and $\Upsilon_{\mathbf{Q}_1}$ via `Bidiagq` such that

$$\Upsilon_{\mathbf{P_1}}^T \Upsilon_\mathbf{C} \Upsilon_{\mathbf{Q}_1} = \Upsilon_D,$$

where $\mathbf{C} = [\tilde{\mathbf{H}}_\mathbf{b}, \tilde{\mathbf{H}}_\mathbf{w}]^H \in \mathbb{H}^{(c+N)\times s}$, and $D \in \mathbb{R}^{(c+N)\times s}$ is a bidiagonal matrix.

6: Compute the SVD of D

$$P_2^T D Q_2 = \begin{bmatrix} R & 0 \\ 0 & 0 \end{bmatrix},$$

and obtain $\mathbf{Q} = \mathbf{Q}_1 Q_2$, $\mathbf{P} = \mathbf{P}_1 P_2$ via `timesQ`.

7: Let $t = \mathtt{rank}(\mathbf{C})$. Bidiagonalize $\mathbf{P}(1:c, 1:t)$ via `Bidiagq` such that

$$\Upsilon_{\mathbf{U}_1}^T \Upsilon_{\mathbf{P}(1:c,1:t)} \Upsilon_{\mathbf{V}_1} = \Upsilon_{D_1},$$

where $\Upsilon_{\mathbf{U}_1}$ and $\Upsilon_{\mathbf{V}_1}$ are two JRS-symplectic matrices, and D_1 is a real bidiagonal matrix.

8: Compute the SVD of D_1

$$U_2^T D_1 V_2 = \begin{bmatrix} R_1 & 0 \\ 0 & 0 \end{bmatrix},$$

and obtain $\mathbf{V} = \mathbf{V}_1 V_2$ via `timesQ`.

9: Compute the nonsingular matrix

$$\mathbf{X} = \mathbf{Q} \begin{bmatrix} R^{-1}\mathbf{V} & 0 \\ 0 & I \end{bmatrix}$$

via `timesQ`, where I is the identity matrix of appropriate size.

10: Compute $\mathbf{G} = \mathbf{\Phi}\mathbf{X}(:, 1:c-1)$ via `timesQ`.

---

4 Numerical Experiments

In this section, we test the efficiency of the color LDA method proposed in this paper, and compare it with the traditional LDA methods. All the experiments are performed on a personal computer with 2.4GHz Intel Xeon E5 and 64 GB memory using MATLAB-R2020b. Machine epsilon is $2.2e-16$. The selected face databases are the famous Georgia Tech face database and Faces95 database.

Georgia Tech face database contains various pose faces with different expressions on cluttered background. All images in Georgia Tech face database are manually cropped, and then resized to 33×44 pixels. The samples of the cropped images are shown in Figure 1. There are 50 persons to be used and per person has 15 images.

Figure 1: Sample images for one individual in Georgia Tech face database

Faces95 database contains 72 individual images, every person has 20 images. Faces95 database collects the facial images of subjects when they spoke. The purpose of requiring subjects to speak is to collect facial expression changes on color face images. The images in faces95 database are also resized to 33×44 pixels, and the background color is brown. Figure 2 shows some examples of faces95 database.

4.1 Color image reconstruction

Suppose that we have obtained the Fisher projection matrix $\hat{\mathbf{X}} \in \mathbb{H}^{mn \times d}$. In subsection 2.4, we have shown that for a given color image $\mathbf{a} \in \mathbb{H}^{mn}$, the feature vector of $\mathbf{a}$ is given by

Ara V. Nefian, The Georgia Tech Face Database. http://www.anefian.com/research/face_reco.htm

L. Spacek's Facial Images Databases. https://cmp.felk.cvut.cz/∼spacelib/faces/

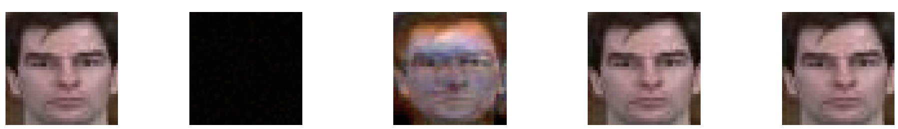
Figure 2: Examples of facial images from faces95 database

$\hat{\mathbf{b}} = \hat{\mathbf{X}}^H \mathbf{a}$, and the reconstructed image of $\mathbf{a}$ is

$$\tilde{\mathbf{a}} = \hat{\mathbf{X}}(\hat{\mathbf{X}}^H \hat{\mathbf{X}})^{-1} \hat{\mathbf{b}}.$$

We call $\tilde{\mathbf{a}}$ as a *reconstructed image* of $\mathbf{a}$, which has the same size as the original color image $\mathbf{a}$. This means that we use a set of Fisher features to reconstruct the original image. If $d = mn$, then we can entirely reconstruct the image: $\tilde{\mathbf{a}} = \mathbf{a}$. But in most cases, d is smaller than mn, so $\tilde{\mathbf{a}}$ is an approximation of $\mathbf{a}$. Define the reconstruction ratio of image reconstruction [7] by

$$\mathtt{Ratio} = 1 - \frac{\|\tilde{\mathbf{a}} - \mathbf{a}\|_2}{\|\mathbf{a}\|_2}.$$



original image	d=50	d=300	d=750	d=1452
	Ratio=0.03%	Ratio=61.81%	Ratio=99.20%	Ratio=100%

Figure 3: Some reconstructed images of one individual

In Figure 3, we take the color face image s01_01.jpg in Georgia Tech face database as an example. Some reconstructed images with different d and the original image are presented. The variant d denotes the number of Fisher projection axes computed by Algorithm 4. As shown from these images, the reconstructed image $\tilde{\mathbf{a}}$ is more and more similar to the original image as the value of d increasing. The result of this experiment indicates that the color LDA method can be used for color face reconstruction, and it can almost completely reconstruct the original image when we only use half of the Fisher projection axes.

4.2 Color image classification

Suppose that the Fisher optimal projection matrix is $\tilde{X} \in \mathbb{H}^{mn\times(c-1)}$, where c is the number of classes in training set. Given two color image vectors $\mathbf{a}_1, \mathbf{a}_2 \in \mathbb{H}^{mn}$, the corresponding projection vectors are denoted by $\mathbf{b}_1, \mathbf{b}_2$. Then the distance between $\mathbf{b}_1$ and $\mathbf{b}_2$ is defined as

$$d(\mathbf{b}_1, \mathbf{b}_2) = \|\mathbf{b}_1 - \mathbf{b}_2\|_{(\tilde{\mathbf{X}}^H\tilde{\mathbf{X}})^{-1}},$$

where $\|\cdot\|_{(\tilde{\mathbf{X}}^H\tilde{\mathbf{X}})^{-1}}$ is the weighted norm mentioned in (2.12).

There are Fisher feature vectors of training images $\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_N$, and each image belongs to a class T_i. For a given test image vector $\mathbf{a}$, $\mathbf{b}$ is the corresponding projection vector. If $d(\mathbf{b}, \mathbf{b}_i) = \min_{l\in\{1,2,\ldots,N\}}\{d(\mathbf{b}, \mathbf{b}_l)\}$, and $\mathbf{b}_i \in T_j$, then the color image $\mathbf{a}$ should be classified into T_j.

$\boxed{\textbf{4.2.1}}$ Comparison with other traditional LDA methods

In this subsection, we compare the color LDA method with other traditional LDA methods based on two mentioned databases. For the famous Georgia Tech face database, the number of chosen Fisher projection axes is 49, where $c = 50$. The first x $(= 7, 10$ or $13)$ images of each individual are chosen as the training set and the remaining as the testing set. For the faces95 database, the number of chosen Fisher projection axes is 71, where $c = 72$. The first x $(= 10, 13,$ or $16)$ images of each individual are chosen as the training set and the remaining as the testing set.

For each case, we repeat the process five times. The average face recognition rate (AR) and the average CPU times (CPU) of three LDA-based methods are shown in Tables 1-2, in which the notations have the following meanings:

CLDA: color LDA proposed by Algorithm 2.

CLDA-SP: structure-preserving color LDA proposed by Algorithm 4.

LDA-grey: traditional LDA using greyscale of color face images proposed in [6].

CLDA-rand(s): structure-preserving color LDA based on randomization proposed by Algorithm 5, where s is the numerical rank.

AR: the arithmetic mean of face recognition rate of five repeated experiments.

Table 1: Average face recognition rate and average CPU time (in seconds) based on Georgia Tech face database

x	CLDA		CLDA-SP		LDA-grey		CLDA-rand ($s = 100$)	
	AR	CPU	AR	CPU	AR	CPU	AR	CPU
7	68.20%	322.30	68.20%	120.91	69.20%	**2.14**	**82.50%**	14.40
10	66.72%	473.14	66.72%	177.20	64.60%	**3.60**	**86.24%**	23.03
13	66.40%	640.40	66.40%	234.80	66.64%	**2.70**	**90.40%**	34.22

The results in Table 1 show that the LDA methods based on quaternion model (CLDA, CLDA-SP) have better performances than traditional LDA using grey images. But the CLDA methods cost more CPU time than traditional LDA. The structure-preserving CLDA can save almost two-thirds CPU time compared with CLDA, but it is also much slower

Table 2: Average face recognition rate and average CPU time (in seconds) based on faces95 database

x	CLDA		CLDA-SP		LDA-grey		CLDA-rand ($s = 115$)	
	AR	CPU	AR	CPU	AR	CPU	AR	CPU
10	81.97%	695.65	82.00%	318.43	80.53%	**6.25**	**90.19%**	39.40
13	81.31%	964.81	81.87%	443.53	79.84%	**7.29**	**92.58%**	58.85
16	80.49%	1326.90	85.00%	583.18	78.54%	**9.31**	**95.28%**	84.81

than LDA-grey. If we combine structure-preserving CLDA with randomization, then the recognition rate of CLDA-rand(100) promotes about 30 percent than that of LDA-grey at the cost of ten times of CPU. Analogous conclusions can be deduced from the numerical results in Table 2.

Additionally, the recognition rate of CLDA-rand(s) increases robustly with the increasing number of projection axes, while the other LDA methods behave not so robust. As a matter of fact, the effect of extracted features of the original color images in the training sets is not effectively enlarged with more projection axes. Using randomization strategy in structure-preserving CLDA can contribute to extract the principle features. Weighted projection method for CLDA will be another consideration to overcome the shortcoming that the recognition rate is not robust with the increasing number of projection axes.

4.2.2 Comparison with three representation-based methods

In [29, 30] the authors proposed three representation-based classification methods using quaternion model, called quaternion collaborative representation-based classification (QCRC), quaternion sparse representation-based classification (QSRC) and quaternion block sparse representation-based classification (QBSRC). All the three methods need to firstly stack the columns of a color image matrix into a quaternion column vector, then form the dictionary by storing the training color image vectors into a matrix. They aim to code the test facial image vector with a representation vector over a dictionary and then use it to infer the correct identity of the test image. There is a positive regularization term in the three representation-based methods, which is tuned by searching from a discrete set

$$\{10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1, 10\}$$

and the best results are reported.

In this experiment, we compare the performances of the proposed structure-preserving CLDA method with three representation-based methods. We resize the images in both faces95 database and Georgia Tech face database to 8×10 at first. To make the results more convincing, we repeat the experiment with five runs and compute the average, maximum and minimum recognition rates for each method, and limit the total running time of classification methods to 48 hours. Notice that within the limited running time, the QBSRC method does not produce results for regularization parameters in the discrete set, so we omits the results of QBSRC.

For the faces95 database, we randomly choose fifteen images in each subject for training and the rest for testing. The results are shown in Table 3. From which we can see that, the recognition rate of structure-preserving CLDA is the highest among three quaternion-based methods. QSRC method costs the most CPU time in training process. Recall that

Table 3: Average, maximal and minimum face recognition rate and average CPU time (in seconds) based on faces95 database

	CLDA-SP	QCRC($\lambda = 0.01$)	QSRC($\lambda = 1$)
average recognition rate	**96.61%**	79.94%	90.11%
maximum recognition rate	**98.61%**	82.50%	91.94%
minimum recognition rate	**95.56%**	75.28%	87.50%
training time	24.7811	**10.4338**	148.9934
testing time	0.5793	**0.0595**	0.0987

for a given test color face image, the CLDA-SP method compares the distances between its projection color image vector and those of all the training image samples. The QCRC and QSRC methods obtain the representation image vector for each class by performing the linear combination of training samples in that class. They only need to compare the distances between the projection image vector of the test image and those of the representation image vectors. That is why both QSRC and QCRC need fewer classification time than CLDA-SP. However, QSRC solves a quaternion Lasso model by the alternating direction method of multipliers framework that converges slowly. Relatively speaking, QCRC costs fewer training time because it gets a projection matrix by solving a quaternion regularized least squares model directly. For the Georgia Tech face database, we randomly choose ten color images in each subject for training and the rest for testing. The numerical results are reported in Table 4, from which analogous conclusions can be deduced.

Table 4: Average, maximal and minimum face recognition rate and average CPU time (in seconds) based on Georgia Tech face database

	CLDA-SP	QCRC($\lambda = 0.01$)	QSRC($\lambda = 1$)
average recognition rate	**87.68%**	69.36%	40.48%
maximum recognition rate	**88.80%**	70.80%	43.20%
minimum recognition rate	**86.00%**	68.00%	38.80%
training time	3.8565	**3.0237**	218.6980
testing time	0.0938	**0.0171**	0.0249

5 Conclusion

In this paper, we have proposed novel structure-preserving color LDA methods based on quaternion model. The proposed methods overcome the small sample size problem successfully by virtual of QGSVD. We have also introduced a criterion to select the Fisher optimal projection axes, which may have the highest recognition rate. The features extracted by these projection axes are statistically uncorrelated. In order to reduce the CPU time, we have proposed a fast structure-preserving color LDA method based on randomization (CLDA-rand(s)), which reduces the dimensions of $\mathbf{H_b}$ and $\mathbf{H_w}$ by the aid of a quaternion random Gaussian matrix. Finally, a large amounts of numerical experiments illustrate that the color LDA methods can be used in color face reconstruction and classification tasks, which have higher recognition rate than traditional LDA, QCRC and QSRC methods. The structure-preserving color LDA method can save more than two-thirds CPU time. The CLDA-rand(s) method can further save much CPU time and its recognition rate becomes higher than traditional LDA methods.

References

[1] P.N. Belhumeur, J.P. Hespanha and D.J. Kriegman, Eigenfaces vs. fisherfaces: Recognition using class specific linear projection, *IEEE Transactions on pattern analysis and machine intelligence* 19 (1997) 711–720.

[2] Z.Z. Chen, Feature extraction by Foley-Sammon transform with kernels, in: Eighth ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing (SNPD 2007), vol. 1, IEEE, 2007, pp. 453–457.

[3] P. Denis, P. Carré and F.-M. Christine, Spatial and spectral quaternionic approaches for colour images, *Computer Vision and Image Understanding* 107 (2007) 74–87.

[4] R.A. Fisher, The use of multiple measurements in taxonomic problems, *Annals of Eugenics* 7 (1936) 179–188.

[5] D.H. Foley and J.W. Sammon, An optimal set of discriminant vectors, *IEEE Transactions on Computers* 100 (1975) 281–289.

[6] P. Howland and H. Park, Generalizing discriminant analysis using the generalized singular value decomposition, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 26 (2004) 995–1006.

[7] Z.G. Jia, S.T. Ling and M.X. Zhao, Color two-dimensional principal component analysis for face recognition based on quaternion model, in: International Conference on Intelligent Computing, Springer, 2017, pp. -177-189.

[8] Z.G. Jia, M.K. Ng and G.J. Song, Lanczos method for large-scale quaternion singular value decomposition, *Numerical Algorithms* 82 (2019) 699–717.

[9] Z.G. Jia, M.S. Wei and S.T. Ling, A new structure-preserving method for quaternion Hermitian eigenvalue problems, *Journal of Computational and Applied Mathematics* 239 (2013) 12–24.

[10] T.S. Jiang, Y.H. Liu and M.S. Wei, Quaternion generalized singular value decomposition and its applications, *Applied Mathematics-A Journal of Chinese Universities* 21 (2006) 113–118.

[11] Z. Jin, J.Y. Yang, Z.S. Hu and Z. Lou, Face recognition based on the uncorrelated discriminant transformation, *Pattern recognition* 34 (2001) 1405–1416.

[12] N. Le Bihan and S.J. Sangwine, Quaternion principal component analysis of color images, in: Proceedings 2003 International Conference on Image Processing (Cat. No.03CH37429), Barcelona, Spain, 2003, pp. I-809.

[13] Y. Li, M.S. Wei, F.X. Zhang and J.L. Zhao, A fast structure-preserving method for computing the singular value decomposition of quaternion matrices, *Applied Mathematics Computation* 235 (2014) 157–167.

[14] K. Liu, Y.Q. Cheng and J.Y. Yang, A generalized optimal set of discriminant vectors, *Pattern Recognition* 25 (1992) 731–739.

[15] K. Liu, Y.Q. Cheng, J.Y. Yang and X. Liu, An efficient algorithm for Foley-Sammon optimal set of discriminant vectors by algebraic method, *International Journal of Pattern Recognition and Artificial Intelligence* 6 (1992) 817–829.

[16] S.T. Ling, Y.D. Li, B. Yang and Z.G. Jia, Joint diagonalization for a pair of Hermitian quaternion matrices and applications to color face recognition, *Signal Processing* 198 (2022): 108560.

[17] M. T. Loots, The development of the quaternion normal distribution, PhD thesis, University of Pretoria, 2010.

[18] Y.L. Luo and D.S. Chen, Face recognition based on color gabor features, *Journal of Image and Graphics* 13 (2007) 242–246.

[19] J.W. Sammon, An optimal discriminant plane, *IEEE Transactions on Computers* 100 (1970) 826–829.

[20] L.L. Shi and B. Funt, Quaternion color texture segmentation, *Computer Vision and Image Understanding* 107 (2007) 88–96.

[21] Q. Tian, Y. Fainman, Z. Gu and S.H. Lee, Comparison of statistical pattern-recognition algorithms for hybrid processing. I. linear-mapping algorithms, *JOSAA* 5 (1988) 1655–1669.

[22] Q.W. Wang and J. Jing, Extreme ranks of (skew-) Hermitian solutions to a quaternion matrix equation, *The Electronic Journal of Linear Algebra* 20 (2010) 552–573.

[23] M.S. Wei, Y. Li, Z.F. Xiang and J.L. Zhao, *Quaternion Matrix Computations*, Nova Science Publishers, New York, 2018.

[24] J.P. Ye, R. Janardan, Q. Li and H. Park, Feature reduction via generalized uncorrelated linear discriminant analysis, *IEEE Transactions on Knowledge and Data Engineering* 18 (2006) 1312–1322.

[25] J.P. Ye and B. Yu, Characterization of a family of algorithms for generalized discriminant analysis on undersampled problems, *Journal of Machine Learning Research* 6 (2005) 483–502.

[26] M.X. Zhao, Z.G. Jia, Y.F. Cai, X. Chen and D.W. Gong, Advanced variations of two-dimensional principal component analysis for face recognition, *Neurocomputing* 452 (2021) 653–664.

[27] M.X. Zhao, Z.G. Jia and D.W. Gong, Improved two-dimensional quaternion principal component analysis, IEEE Access **7** (2019) 79409–79417.

[28] W.M. Zheng, L. Zhao and C.R. Zou, Foley-Sammon optimal discriminant vectors using kernel approach, *IEEE Transactions on Neural Networks* 16 (2005) 1–9.

[29] C.M. Zou, K.I. Kou and Y.L. Wang, Quaternion collaborative and sparse representation with application to color face recognition, *IEEE Transactions on image processing* 25 (2016) 3287–3302.

[30] C.M. Zou, K.I. Kou, Y.L. Wang and Y.Y. Tang, Quaternion block sparse representation for signal recovery and classification, *Signal Processing* 179 (2021): 107849.

Manuscript received 28 May 2023
revised 13 August 2023
accepted for publication 4 September 2023

Si-Tao Ling
School of Mathematics
China University of Mining and Technology
Xuzhou 221116, P.R. China
Jiangsu Center for Applied Mathematics (CUMT)
Xuzhou 221116, P.R. China
E-mail address: lingsitao2004@163.com

Zhe-Han Hu
School of Mathematics
China University of Mining and Technology
Xuzhou 221116, P.R. China
E-mail address: zhhu@cumt.edu.cn

Bing Yang
School of Mathematics
China University of Mining and Technology
Xuzhou 221116, P.R. China
E-mail address: bjxdxfsh@163.com

Yi-Ding Li
School of Mathematics
China University of Mining and Technology
Xuzhou 221116, P.R. China
E-mail address: liyd@cumt.edu.cn