Numerical Algorithms
https://doi.org/10.1007/s11075-024-01887-9

ORIGINAL PAPER

Logarithmic norm minimization of quaternion matrix decomposition for color image sparse representation

Xiao-Min Cai[1] · Yi-Fen Ke[1] · Chang-Feng Ma[1] · Ya-Jun Xie[3] · Ri-Wei Liao[2]

Received: 28 April 2024 / Accepted: 11 July 2024

Abstract
In this paper, incorporating the quaternion matrix framework, the logarithmic norm of quaternion matrices is employed to approximate rank. Unlike conventional sparse representation techniques for matrices, which treat RGB channels separately, quaternion-based methods maintain image structure by representing color images within a pure quaternion matrix. Leveraging the logarithmic norm, factorization and truncation techniques can be applied for proficient image recovery. Optimization of these approaches is facilitated through an alternate minimization framework, supplemented by meticulous mathematical scrutiny ensuring convergence. Finally, some numerical examples are used to demonstrate the effectiveness of the proposed algorithms.

Keywords Quaternion matrix · Matrix decomposition · Logarithmic norm · Color image reconstruction

1 Introduction

In today's era of information explosion, data volumes across various industries are experiencing exponential growth, posing significant challenges in storage, processing,

✉ Yi-Fen Ke
keyifen@fjnu.edu.cn

✉ Ri-Wei Liao
riweiliao2020@fjnu.edu.cn

1 School of Mathematics and Statistics & Key Laboratory of Analytical Mathematics and Applications (Ministry of Education) & Fujian Provincial Key Laboratory of Statistics and Artificial Intelligence & Fujian Key Laboratory of Analytical Mathematics and Applications (FJKLAMA) & Center for Applied Mathematics of Fujian Province (FJNU), Fujian Normal University, Fuzhou 350117, P.R. China

2 College of Photonic and Electronic Engineering & Key Laboratory of Optoelectronic Science and Technology for Medicine of Ministry of Education & Fujian Provincial Key Laboratory of Photonics Technology, Fujian Normal University, Fuzhou 350117, P.R. China

3 School of Big Data, Fuzhou University of International Studies and Trade, Fuzhou 350202, P.R. China

Published online: 19 July 2024

and analysis. High-dimensional data, such as images [1], videos [2], sensor readings [3] and genomic sequences [4], are becoming increasingly prevalent in scientific research [5], industrial applications [6] and daily life. However, high-dimensional data often come with substantial computational and storage costs, along with the potential for containing a considerable amount of redundant information. To address this challenge, researchers have turned to dimensionality reduction techniques aimed at representing complex high-dimensional data in lower-dimensional spaces while preserving their fundamental structure and information [7, 8].

A prominent dimensionality reduction technique is the sparse representation model [9–11], which aims to reduce data dimensionality by finding a sparse set of linear combinations to express the data [12]. By emphasizing the sparsity of the representation matrix, redundant information in the data can be eliminated, resulting in a more concise and interpretable data representation. The application of rank-constrained optimization problems is widespread. For example, in the field of computer vision, sparse representation has been used for tasks such as image denoising [13], image restoration [14], and image classification [15]. In signal processing, sparse representation models have been applied to audio and speech processing for tasks like source separation [16], speech enhancement [17], and audio compression [18]. Additionally, sparse representation models find wide applications in neuroscience [19], bioinformatics [20], genomics [21], and other fields.

In prior research, the nuclear norm of a matrix has been demonstrated as an effective alternative method to traditional rank function. Unlike the nuclear norm, which treats each singular value equally [22], larger singular values may contain richer and more useful information compared to smaller ones [22, 23]. To address these limitations, several non-convex alternatives have been proposed, such as the truncated nuclear norm, weighted nuclear norm, and weighted Schatten-p norm, as suggested in [22–24]. Additionally, the log-determinant alternating method has been proven to be a more accurate approximation of rank [25], particularly evident in neural networks compared to the nuclear norm. These strategies are all based on approximate optimization of rank, where singular value decomposition (SVD) plays an indispensable role. However, the high complexity involved in SVD computations poses limitations when dealing with high-dimensional or large datasets [26]. To overcome this challenge, low-rank matrix decomposition is applied to the sparse representation of matrices [26]. Specifically, decomposing the target matrix into two smaller factor matrices describes the low-rank property of the target matrix. This approach not only satisfies the requirement of low-rank matrices but also benefits from rapid numerical optimization.

Currently, sparse representation methods for color images primarily focus on processing two-dimensional data. Consequently, when dealing with color images, it is common to process the RGB channels separately. However, this approach may lead to overlooking the intrinsic relationships among the three channels, resulting in the loss of their potential connections.

Based on the analysis above, there are two main issues with low-rank structured sparse representation of color images.

The first challenge in sparse representation of color images is that the correlation between RGB channels cannot be adequately maintained Consequently, researchers

have turned their attention to color image processing methods based on quaternion frameworks. Due to the unique structure of quaternion, each pixel in a color image can be represented using pure quaternions, forming a quaternion matrix. This approach has been widely applied in areas such as face recognition [27, 28], image edge detection [29], and image denoising [30, 31]. Other applications of color image processing can be found in references [32–34].

The second challenge is how to accurately describe the underlying low-rank structure In the sparse representation of color images in quaternion space, many studies have been based on non-convex functions used for approximating the rank of quaternion matrices, including weighted Schatten-p norm, and Laplacian approximation. These functions highlight the advantages of quaternion matrices, which have been validated experimentally and theoretically. However, these methods require full processing of QSVD for quaternion matrices, which incurs high computational costs. To address this challenge, researchers have extended low-rank matrix decomposition to the quaternion algebra. The authors decompose the target quaternion matrix into dual-factor quaternion matrices for low-rank quaternion matrix completion [35]. These factorization-based methods only require optimization of two smaller quaternion matrices, thus significantly reducing any associated computational costs.

To address the aforementioned challenges, two quaternion sparse representation models have been proposed: Quaternion Logarithmic Norm Factorization Sparse Representation (QLNFSR) and Truncated Quaternion Logarithmic Norm Sparse Representation (TQLNSR). Both models are designed to approximate the rank in quaternion algebra more accurately and efficiently, thereby better utilizing the structure of color images. This paper utilizes the quaternion log-norm as a non-convex substitute for rank, which provides a more reliable description of low-rank matrices through compared to traditional approximation methods such as quaternion nuclear norm. With the increasing of singular values, the penalty of the logarithmic function becomes more lenient, hence the smaller singular values may receive more punishment. In Fig. 1, an intuitive explanation of rank approximation using the logarithmic function for scalar cases can be observed. When $\mathbf{X} = x \in \mathbb{R}$, $\text{rank}(x) = 0$ if $x = 0$ and $\text{rank}(x) = 1$ otherwise. Additionally, for x bounded by a positive constant M, denoted as $|x| \leq M$, $\frac{\|x\|_*}{M} = \frac{|x|}{M}$ represents the convex envelope of $\text{rank}(x)$ on the interval $\{x \mid |x| \leq M\}$ [36]. Consequently, the slope of $\text{rank}(x)$ at the origin is infinite, while the convex envelope ($|x| \leq M$) exhibits a consistent slope. In contrast, the slope of the logarithmic function at the origin is $1/\delta$, where δ is a small positive constant that ensures the logarithmic norm closely approximates $\text{rank}(x)$ compared to the convex envelope [37].

Subsequently, the quaternion logarithmic norm is applied to two smaller quaternion matrices, which are the factor quaternion matrices of the target quaternion matrix based on the quaternion log-norm factorization algorithm. Therefore, the expensive QSVD only needs to act on the smaller factor quaternion matrices, thereby improving algorithm efficiency. In the truncated quaternion logarithmic norm approximation algorithm, the quaternion logarithmic norm is first truncated, and then the shrinkage operator of the quaternion logarithmic norm is directly applied to optimize the target

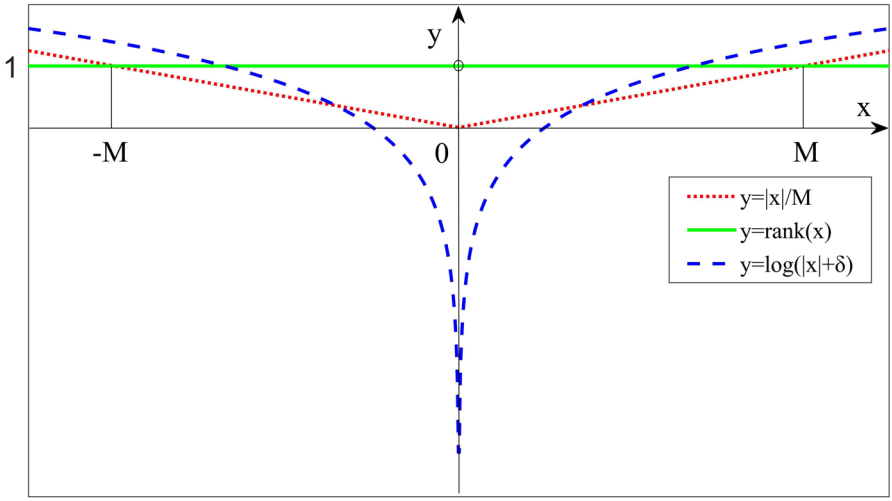

Fig. 1 The rank, nuclear norm, and the logarithmic norm for scalar x

quaternion matrix. Thus, the main contributions of this paper can be summarized as follows.

- The rank of quaternion matrices is not affected by the maximum singular value. Therefore, truncated quaternion logarithmic norm operates is to truncate the largest singular value, and then utilize the quaternion logarithmic norm in the truncation problem.
- In order to solve the QLNFSR and TQLNSR models, we adopt the idea of alternating direction method of multipliers (ADMM) and establish two main algorithms called QLNFSR algorithm (Algorithm 1) and TQLNSR algorithm (Algorithm 2). Experimental results confirm the effectiveness of two algorithms in color image sparse representation processes.

This paper is organized as follows. In Section 2, we review some notations, definitions and lemmas with regard to the quaternion matrix. In Section 3, we review sparse representation methods based on low-rank matrices and introduce two strategies for low-rank sparse representation based on quaternion. In Section 4, we provide convergence analysis of the algorithms. In Section 5, the proposed algorithms have been applied to color image reconstruction. The feasibility and effectiveness of the algorithms are verified. Finally, we give some conclusions in Section 6.

Notation In this article, $\mathbb{R}$ and $\mathbb{Q}$ denote the real space and quaternion space, respectively. A scalar and a matrix are written as a and A, respectively. $\boldsymbol{a}$ and $\boldsymbol{A}$ represent a quaternion number and a quaternion matrix, respectively. The symbols $(\cdot)^*$, $(\cdot)^{-1}$, $(\cdot)^T$ and $(\cdot)^H$ denote the conjugation, inverse, transpose and conjugate transpose, respectively. The symbols $|\cdot|$, $\|\cdot\|_F$ and $\|\cdot\|_*$ are the absolute value or modulus, the Frobenius norm and the nuclear norm, respectively. The symbols $\langle\cdot,\cdot\rangle$, $\mathrm{tr}\{\cdot\}$ and $\mathrm{rank}(\cdot)$ denote

the inner product operation, the trace and rank operators, respectively. The real part of quaternion (scalar, vector, and matrix) denotes $\Re(\cdot)$. The symbol I represents the identity matrix with appropriate size.

2 Preliminaries

In this section, we recall some preliminary results that will be used in the following discussion. Firstly, we introduce the definition of quaternion.

Definition 2.1 [38] A quaternion $\boldsymbol{q} \in \mathbb{Q}$ is expressed as

$$\boldsymbol{q} = q_0 + q_1\boldsymbol{i} + q_2\boldsymbol{j} + q_3\boldsymbol{k},$$

where $q_0, q_1, q_2, q_3 \in \mathbb{R}$, and three imaginary units $\boldsymbol{i}, \boldsymbol{j}, \boldsymbol{k}$ satisfy

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

One of the most important properties of quaternion is that the multiplication of quaternion is noncommutative as these rules, that is $\boldsymbol{pq} \neq \boldsymbol{qp}$ in general for $\boldsymbol{p}, \boldsymbol{q} \in \mathbb{Q}$. For example, it is obvious that $\boldsymbol{pq} \neq \boldsymbol{qp}$ while $\boldsymbol{p} = \boldsymbol{i}$ and $\boldsymbol{q} = \boldsymbol{j}$.

Definition 2.2 [38] A quaternion matrix $\boldsymbol{A} \in \mathbb{Q}^{m\times n}$ is expressed as

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

where $A_0, A_1, A_2, A_3 \in \mathbb{R}^{m\times n}$. The conjugate transpose matrix of $\boldsymbol{A}$ is defined as

$$\boldsymbol{A}^H = A_0^T - A_1^T\boldsymbol{i} - A_2^T\boldsymbol{j} - A_3^T\boldsymbol{k}.$$

We get the following definitions about the norm of the quaternion and quaternion matrix.

Definition 2.3 [38] Let $\boldsymbol{a} \in \mathbb{Q}$ and $\boldsymbol{A} \in \mathbb{Q}^{m\times n}$. The norm of a quaternion $\boldsymbol{a} = a_0 + a_1\boldsymbol{i} + a_2\boldsymbol{j} + a_3\boldsymbol{k}$ and the Frobenius norm of the quaternion matrix $\boldsymbol{A} = A_0 + A_1\boldsymbol{i} + A_2\boldsymbol{j} + A_3\boldsymbol{k} = (\boldsymbol{a}_{ij})$, are defined as

$$\|\boldsymbol{a}\| = \sqrt{a_0^2 + a_1^2 + a_2^2 + a_3^2}$$

and

$$\|\boldsymbol{A}\|_F = \sqrt{\|A_0\|_F^2 + \|A_1\|_F^2 + \|A_2\|_F^2 + \|A_3\|_F^2} = \sqrt{\mathrm{tr}(\boldsymbol{A}^H\boldsymbol{A})}.$$

Below we give the definition of the generalized inverse of a quaternion matrix.

Definition 2.4 [39] For given $\boldsymbol{A} \in \mathbb{Q}^{m\times n}$, the generalized inverse of the quaternion matrix $\boldsymbol{A}$ is defined as $\boldsymbol{X}$, which satisfies the following conditions

$$(1)\ \boldsymbol{AXA} = \boldsymbol{A},\ (2)\ \boldsymbol{XAX} = \boldsymbol{X},\ (3)\ (\boldsymbol{AX})^H = \boldsymbol{AX},\ (4)\ (\boldsymbol{XA})^H = \boldsymbol{XA}. \quad (2.1)$$

We denote $\boldsymbol{X}$ by $\boldsymbol{A}^{\dagger}$.

Specially, if $\boldsymbol{A}$ is invertible, it is clear that $\boldsymbol{X} = \boldsymbol{A}^{-1}$ trivially satisfies (2.1).

Definition 2.5 (QNN, [40]) Given $\boldsymbol{X} \in \mathbb{Q}^{m\times n}$, the nuclear norm of the quaternion matrix is

$$\|\boldsymbol{X}\|_* = \sum_{i=1}^{\min(m,n)} \sigma_i(\boldsymbol{X}), \tag{2.2}$$

where $\sigma_i(\boldsymbol{X})$ denotes the i-th singular value of $\boldsymbol{X}$.

Lemma 2.1 (Binary factorization framework, [35]) *Let the quaternion matrix* $\boldsymbol{A} \in \mathbb{Q}^{m\times n}$ *with* $\operatorname{rank}(\boldsymbol{A}) = r \leq d$. *Then the binary factorization framework is devised as*:

$$\boldsymbol{A} = \boldsymbol{U}\boldsymbol{V}^H, \tag{2.3}$$

where $\boldsymbol{U} \in \mathbb{Q}^{m\times d}$ *and* $\boldsymbol{V} \in \mathbb{Q}^{n\times d}$ *such that* $\operatorname{rank}(\boldsymbol{U}) = \operatorname{rank}(\boldsymbol{V}) = r$.

Lemma 2.2 (Quaternion Singular Value Decomposition (QSVD), [19]) *Let* $\boldsymbol{A} \in \mathbb{Q}^{m\times n}$, *then there exist two unitary quaternion matrices* $\boldsymbol{U} \in \mathbb{Q}^{m\times m}$ *and* $\boldsymbol{V} \in \mathbb{Q}^{n\times n}$ *such that*

$$\boldsymbol{A} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^H,$$

where $\boldsymbol{\Sigma} = \operatorname{diag}(\sigma_1, \sigma_2, \cdots, \sigma_l)$, $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_l \geq 0$, *and the diagonal elements of* $\boldsymbol{\Sigma}$ *are all the nonnegative singular values of the matrix* $\boldsymbol{A}$ *and* $l = \min(m, n)$.

Definition 2.6 (QLN, [41]) Let $\boldsymbol{X} \in \mathbb{Q}^{m\times n}$. The logarithmic norm of the quaternion matrix with $0 \leq p \leq 1$ and $\epsilon > 0$ is defined as

$$\|\boldsymbol{X}\|_L^p = \sum_{i=1}^{\min(m,n)} \log(\sigma_i^p(\boldsymbol{X}) + \epsilon), \tag{2.4}$$

where $\sigma_i(\boldsymbol{X})$ denotes the i-th singular value of $\boldsymbol{X}$.

Lemma 2.3 [41] *Let the quaternion matrix* $\boldsymbol{X} \in \mathbb{Q}^{m\times n}$ *with* $\operatorname{rank}(\boldsymbol{X}) = r \leq d \leq \min\{m, n\}$. *There exist* $\boldsymbol{U} \in \mathbb{Q}^{m\times d}$ *and* $\boldsymbol{V} \in \mathbb{Q}^{N\times d}$ *such that* $\boldsymbol{X} = \boldsymbol{U}\boldsymbol{V}^H$. *Then we have*:

$$\|\boldsymbol{X}\|_L^{1/2} = \min_{\substack{\boldsymbol{U},\boldsymbol{V} \\ \boldsymbol{X}=\boldsymbol{U}\boldsymbol{V}^H}} \frac{1}{2}\|\boldsymbol{U}\|_L^1 + \frac{1}{2}\|\boldsymbol{V}\|_L^1. \tag{2.5}$$

Lemma 2.4 (Quaternion Logarithmic Singular Value Thresholding (QLSVT), [41]) *Let the quaternion matrix* $\boldsymbol{A} \in \mathbb{Q}^{m\times n}$ *and* $\lambda > 0$. *If QSVD of* $\boldsymbol{A}$ *is* $\boldsymbol{A} = \boldsymbol{U}_A\boldsymbol{\Sigma}_A\boldsymbol{V}_A^H$, *then the closed solution of the problem*

$$\arg\min_{\boldsymbol{X}} \frac{1}{2}\|\boldsymbol{A} - \boldsymbol{X}\|_F^2 + \lambda\|\boldsymbol{X}\|_L^1 \tag{2.6}$$

is provided by $\boldsymbol{X} = \boldsymbol{U}_{\boldsymbol{A}} \boldsymbol{\Delta}_{\lambda,\epsilon,\boldsymbol{A}} \boldsymbol{V}_{\boldsymbol{A}}^{H}$. *Here, the soft thresholding operator* $\boldsymbol{\Delta}_{\lambda,\epsilon,\boldsymbol{A}}$ *is defined as*:

$$\boldsymbol{\Delta}_{\lambda,\epsilon,\boldsymbol{A}} = \operatorname{diag}(l_{\lambda,\epsilon}(\sigma_1),\ l_{\lambda,\epsilon}(\sigma_2),\ \cdots, l_{\lambda,\epsilon}(\sigma_r),\ 0,\ \cdots,\ 0) \in \mathbb{R}^{m\times n}$$

with

$$l_{\lambda,\epsilon}(x) = \begin{cases} 0, & \delta \le 0, \\ \underset{a\in\{0,\frac{1}{2}(x-\epsilon+\sqrt{\delta})\}}{\arg\min} \quad h(a), & \delta > 0, \end{cases} \tag{2.7}$$

where $\delta = (x-\epsilon)^2 - 4(\lambda - x\epsilon)$ *and the function* $h : \mathbb{R}^+ \to \mathbb{R}^+$ *is defined as* $h(a) := \frac{1}{2}(a-x)^2 + \lambda \log(a+\epsilon)$.

3 Quaternion matrix sparse representation model

Due to the pronounced non-local self-similarity evident in the structure of visual data, often observed as low-rank features, the goal of matrix sparse representation models is to tackle image rendering problems using the following approach:

$$\min_{X} \operatorname{rank}(X),\ \text{s.t. } B \approx AX, \tag{3.1}$$

where $\operatorname{rank}(X)$ is the rank function, B is the known data matrix, A is the constraint matrix, and X is the matrix to be found. Problem (3.1) constitutes a combinatorial optimization challenge, typically addressable by optimizing convex surrogates for the rank function [42].

Problem (3.1) briefly introduces the classic matrix sparse representation model, which fundamentally optimizes the sparse representation of grayscale images and other two-dimensional data. When processing color images, the model in (3.1) needs to decompose the RGB channels, while the quaternion matrix sparse representation model can assemble the RGB channels. Therefore, it can be expressed as:

$$\min_{\boldsymbol{X}} \operatorname{rank}(\boldsymbol{X}),\ \text{s.t. } \boldsymbol{B} \approx \boldsymbol{A}\boldsymbol{X}, \tag{3.2}$$

where $\operatorname{rank}(\boldsymbol{X})$ is the quaternion matrix rank function, $\boldsymbol{B}$ is the known quaternion matrix, $\boldsymbol{A}$ is the constraint quaternion matrix, and $\boldsymbol{X}$ is the quaternion matrix to be found.

The main sparse representation model in quaternion algebra primarily focuses on low-rank minimization. Similar to the classical matrix sparse representation model, the rank function in model (3.2) is challenging to optimize. Therefore, according to Definition 2.5, the low-rank minimization model can be expressed as:

$$\min_{\boldsymbol{X}} \|\boldsymbol{X}\|_*,\ \text{s.t. } \boldsymbol{B} \approx \boldsymbol{A}\boldsymbol{X}. \tag{3.3}$$

A rank, akin to a value in the real domain, is represented by a real number. As depicted in Fig. 1, QLN provides a finer approximation compared to QNN. Furthermore, drawing on both the bi-factor surrogate theorem for matrix logarithmic norm

mentioned in [37] and Lemma 2.1, it becomes feasible to formulate the bi-factor surrogate theorem for QLN. Therefore, combined with the Definition 2.6 and Lemma 2.3, this paper proposes two quaternion sparse representation models: Quaternion Logarithmic Norm Factorization Sparse Representation (QLNFSR) and Truncated Quaternion Logarithmic Norm Sparse Representation (TQLNSR) as follows.

3.1 Quaternion logarithmic norm factorization sparse representation

According to Definition 2.6 and Lemma 2.3, on the basis of model (3.3), the following sparse representation model based on quaternion framework is proposed:

$$\min_{\boldsymbol{X}} \|\boldsymbol{X}\|_L^{1/2}, \text{ s.t. } \boldsymbol{B} \approx \boldsymbol{A}\boldsymbol{X}, \tag{3.4}$$

where $\boldsymbol{B}$ is the known quaternion matrix, $\boldsymbol{A}$ is the constraint quaternion matrix, and $\boldsymbol{X}$ is the quaternion matrix to be found. Our aim is to minimize the disparity between $\boldsymbol{B}$ and $\boldsymbol{A}\boldsymbol{X}$, while simultaneously reducing the rank of $\boldsymbol{X}$ to its lowest possible value. Combining the aforementioned objectives into a single equation allows us to represent (3.4) as:

$$\min_{\boldsymbol{X},\boldsymbol{Y},\boldsymbol{Z}} \frac{\lambda}{2}(\|\boldsymbol{Y}\|_L^1 + \|\boldsymbol{Z}\|_L^1) + \frac{\rho}{2}\|\boldsymbol{B} - \boldsymbol{A}\boldsymbol{X}\|_F^2, \text{ s.t. } \boldsymbol{Y} = \boldsymbol{M}, \boldsymbol{Z} = \boldsymbol{N}, \boldsymbol{X} = \boldsymbol{M}\boldsymbol{N}^H. \tag{3.5}$$

It's worth noting that (3.5) decomposes the interconnected terms, enabling them to be tackled separately. Subsequently, the challenges posed by (3.5) can be addressed using the ADMM framework.

Initially, we address the task described in (3.5) by minimizing the augmented Lagrangian function given by:

$$\begin{aligned}
&\mathcal{L}(\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}, \boldsymbol{M}, \boldsymbol{N}, \boldsymbol{F}_1, \boldsymbol{F}_2, \boldsymbol{F}_3, \alpha) \\
&= \frac{\lambda}{2}(\|\boldsymbol{Y}\|_L^1 + \|\boldsymbol{Z}\|_L^1) + \frac{\rho}{2}\|\boldsymbol{B} - \boldsymbol{A}\boldsymbol{X}\|_F^2 + \Re(\langle \boldsymbol{F}_1, \boldsymbol{M} - \boldsymbol{Y}\rangle) + \Re(\langle \boldsymbol{F}_2, \boldsymbol{N} - \boldsymbol{Z}\rangle) \\
&\quad + \Re(\langle \boldsymbol{F}_3, \boldsymbol{X} - \boldsymbol{M}\boldsymbol{N}^H\rangle) + \frac{\alpha}{2}(\|\boldsymbol{Y} - \boldsymbol{M}\|_F^2 + \|\boldsymbol{Z} - \boldsymbol{N}\|_F^2 + \|\boldsymbol{X} - \boldsymbol{M}\boldsymbol{N}^H\|_F^2),
\end{aligned} \tag{3.6}$$

where $\boldsymbol{F}_1$, $\boldsymbol{F}_2$ and $\boldsymbol{F}_3$ are Lagrange multipliers, $\alpha > 0$ is the penalty parameter.

Updating $\boldsymbol{M}$ and $\boldsymbol{N}$ In the $(k+1)$-th iteration, while keeping the other variables at their most recent values, $\boldsymbol{M}$ and $\boldsymbol{N}$ are determined as the optimal solutions of the subsequent problems:

$$\begin{aligned}
\boldsymbol{M}^{k+1} &= \arg\min_{\boldsymbol{M}} \frac{1}{2}\|\boldsymbol{X}^k + \frac{1}{\alpha^k}\boldsymbol{F}_3^k - \boldsymbol{M}(\boldsymbol{N}^k)^H\|_F^2 + \frac{1}{2}\|\boldsymbol{M} - \boldsymbol{Y}^k + \frac{1}{\alpha^k}\boldsymbol{F}_1^k\|_F^2. \\
\boldsymbol{N}^{k+1} &= \arg\min_{\boldsymbol{N}} \frac{1}{2}\|\boldsymbol{X}^k + \frac{1}{\alpha^k}\boldsymbol{F}_3^k - \boldsymbol{M}^{k+1}\boldsymbol{N}^H\|_F^2 + \frac{1}{2}\|\boldsymbol{N} - \boldsymbol{Z}^k + \frac{1}{\alpha^k}\boldsymbol{F}_2^k\|_F^2.
\end{aligned} \tag{3.7}$$

Let

$$\mathcal{Q}(\boldsymbol{M}) := \frac{1}{2}\|\boldsymbol{X}^k + \frac{1}{\alpha^k}\boldsymbol{F}_3^k - \boldsymbol{M}(\boldsymbol{N}^k)^H\|_F^2 + \frac{1}{2}\|\boldsymbol{M} - \boldsymbol{Y}^k + \frac{1}{\alpha^k}\boldsymbol{F}_1^k\|_F^2$$

and

$$\mathcal{P}(\boldsymbol{N}) := \frac{1}{2}\|\boldsymbol{X}^k + \frac{1}{\alpha^k}\boldsymbol{F}_3^k - \boldsymbol{M}^{k+1}\boldsymbol{N}^H\|_F^2 + \frac{1}{2}\|\boldsymbol{N} - \boldsymbol{Z}^k + \frac{1}{\alpha^k}\boldsymbol{F}_2^k\|_F^2.$$

Applying the relevant principles regarding quaternion matrix derivatives as outlined in [43], the gradient of $\mathcal{Q}(\boldsymbol{M})$ can be calculated as

$$\frac{\partial \mathcal{Q}(\boldsymbol{M})}{\partial \boldsymbol{M}} = \boldsymbol{M}(\boldsymbol{N}^k)^H\boldsymbol{N}^k - (\boldsymbol{X}^k + \frac{1}{\alpha^k}\boldsymbol{F}_3^k)\boldsymbol{N}^k + \boldsymbol{M} - \boldsymbol{Y}^k + \frac{1}{\alpha^k}\boldsymbol{F}_1^k. \tag{3.8}$$

By setting (3.8) equal to zero, the solution can be derived as

$$\boldsymbol{M}^{k+1} = \big[(\boldsymbol{X}^k + \frac{1}{\alpha^k}\boldsymbol{F}_3^k)\boldsymbol{N}^k + \boldsymbol{Y}^k - \frac{1}{\alpha^k}\boldsymbol{F}_1^k\big]\big[I + (\boldsymbol{N}^k)^H\boldsymbol{N}^k\big]^{-1}. \tag{3.9}$$

Utilizing a comparable approach, we can derive the optimal solution for $\boldsymbol{N}^{k+1}$ as follows:

$$\boldsymbol{N}^{k+1} = \big[(\boldsymbol{X}^k + \frac{1}{\alpha^k}\boldsymbol{F}_3^k)^H\boldsymbol{M}^{k+1} + \boldsymbol{Z}^k - \frac{1}{\alpha^k}\boldsymbol{F}_2^k\big]\big[I + (\boldsymbol{M}^{k+1})^H\boldsymbol{M}^{k+1}\big]^{-1}. \tag{3.10}$$

Updating $\boldsymbol{Y}$ and $\boldsymbol{Z}$ In the $(k+1)$-th iteration, while maintaining the remaining variables at their most recent values, $\boldsymbol{Y}^{k+1}$ and $\boldsymbol{Z}^{k+1}$ represent the optimal solutions of the subsequent problems:

$$\begin{cases} \boldsymbol{Y}^{k+1} = \arg\min\limits_{\boldsymbol{Y}} \frac{1}{2}\|\boldsymbol{M}^{k+1} + \frac{1}{\alpha^k}\boldsymbol{F}_1^k - \boldsymbol{Y}\|_F^2 + \frac{\lambda}{2\alpha^k}\|\boldsymbol{Y}\|_L^1, \\ \boldsymbol{Z}^{k+1} = \arg\min\limits_{\boldsymbol{Z}} \frac{1}{2}\|\boldsymbol{N}^{k+1} + \frac{1}{\alpha^k}\boldsymbol{F}_2^k - \boldsymbol{Z}\|_F^2 + \frac{\lambda}{2\alpha^k}\|\boldsymbol{Z}\|_L^1. \end{cases} \tag{3.11}$$

According to Lemma 2.4, we can utilize the QLSVT technique to update $\boldsymbol{Y}^{k+1}$ and $\boldsymbol{Z}^{k+1}$ in reference to (3.11), that are

$$\begin{cases} \boldsymbol{Y}^{k+1} = \boldsymbol{U}_{S_1}\boldsymbol{\Delta}_{\frac{\lambda}{2\alpha^k},\epsilon,\boldsymbol{\Sigma}_{S_1}}\boldsymbol{V}_{S_1}^H, \\ \boldsymbol{Z}^{k+1} = \boldsymbol{U}_{S_2}\boldsymbol{\Delta}_{\frac{\lambda}{2\alpha^k},\epsilon,\boldsymbol{\Sigma}_{S_2}}\boldsymbol{V}_{S_2}^H, \end{cases} \tag{3.12}$$

where $\boldsymbol{S}_1 = \boldsymbol{M}^{k+1} + \frac{1}{\alpha^k}\boldsymbol{F}_1^k$ and $\boldsymbol{S}_2 = \boldsymbol{N}^{k+1} + \frac{1}{\alpha^k}\boldsymbol{F}_2^k$. Let $\boldsymbol{S}_1 = \boldsymbol{U}_{S_1}\boldsymbol{\Sigma}_{S_1}\boldsymbol{V}_{S_1}^H$ and $\boldsymbol{S}_2 = \boldsymbol{U}_{S_2}\boldsymbol{\Sigma}_{S_2}\boldsymbol{V}_{S_2}^H$ be the QSVD of quaternion matrices $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$, respectively.

Updating $\boldsymbol{X}$ In the $(k+1)$-th iteration, fixing the other variables at their latest values, $\boldsymbol{X}$ is the optimal solutions of the following problem:

$$\boldsymbol{X}^{k+1} = \arg\min_{\boldsymbol{X}} \frac{\rho}{2}\|\boldsymbol{B} - \boldsymbol{A}\boldsymbol{X}\|_F^2 + \frac{1}{2}\|\boldsymbol{X} - \boldsymbol{M}^{k+1}(\boldsymbol{N}^{k+1})^H + \frac{1}{\alpha^k}\boldsymbol{F}_3^k\|_F^2. \tag{3.13}$$

Let

$$\mathcal{H}(\boldsymbol{X}) := \frac{\rho}{2}\|\boldsymbol{B} - \boldsymbol{A}\boldsymbol{X}\|_F^2 + \frac{1}{2}\|\boldsymbol{X} - \boldsymbol{M}^{k+1}(\boldsymbol{N}^{k+1})^H + \frac{1}{\alpha^k}\boldsymbol{F}_3^k\|_F^2.$$

The gradient of $\mathcal{H}(\boldsymbol{X})$ can be calculated as

$$\frac{\partial \mathcal{H}(\boldsymbol{X})}{\partial \boldsymbol{X}} = \rho \boldsymbol{A}^H(\boldsymbol{A}\boldsymbol{X} - \boldsymbol{B}) + (\boldsymbol{X} + \frac{\boldsymbol{F}_3^k}{\alpha^k} - \boldsymbol{M}^{k+1}(\boldsymbol{N}^{k+1})^H). \tag{3.14}$$

Setting (3.14) to zero, we can obtain a unique solution

$$\boldsymbol{X}^{k+1} = \left[I + \rho \boldsymbol{A}^H \boldsymbol{A}\right]^{-1}\left[\rho \boldsymbol{A}^H B + \boldsymbol{M}^{k+1}(\boldsymbol{N}^{k+1})^H - \frac{1}{\alpha^k}\boldsymbol{F}_3^k\right]. \tag{3.15}$$

Updating $\boldsymbol{F_1}$, $\boldsymbol{F_2}$, $\boldsymbol{F_3}$ and $\boldsymbol{\alpha}$ The update formats are as follows:

$$\begin{cases} \boldsymbol{F}_1^{k+1} = \boldsymbol{F}_1^k + \alpha^k(\boldsymbol{Y}^{k+1} - \boldsymbol{M}^{k+1}), \\ \boldsymbol{F}_2^{k+1} = \boldsymbol{F}_2^k + \alpha^k(\boldsymbol{Z}^{k+1} - \boldsymbol{N}^{k+1}), \\ \boldsymbol{F}_3^{k+1} = \boldsymbol{F}_3^k + \alpha^k(\boldsymbol{X}^{k+1} - \boldsymbol{M}^{k+1}(\boldsymbol{N}^{k+1})^H), \\ \alpha^{k+1} = \min(\beta\alpha^k, \alpha_{\max}). \end{cases} \tag{3.16}$$

Algorithm 1 outlines the complete process, detailing each step sequentially.

Algorithm 1 QLNFSR algorithm.

Input: $\boldsymbol{A}$, $\boldsymbol{B}$, ρ, β, $\alpha_{\max}$, $d(d \le \min\{m, n\})$
Output: $\boldsymbol{X}^{k+1}$

1: **Initialize** $\boldsymbol{M}^0$, $\boldsymbol{N}^0$, $\boldsymbol{X}^0$, $\boldsymbol{Y}^0$, $\boldsymbol{Z}^0$, $\boldsymbol{F}_1^0$, $\boldsymbol{F}_2^0$, $\boldsymbol{F}_3^0$ and α^0
2: **Repeat**
3: Update $\boldsymbol{M}^{k+1}$ and $\boldsymbol{N}^{k+1}$ via (3.9) and (3.10).
4: Update $\boldsymbol{Y}^{k+1}$ and $\boldsymbol{Z}^{k+1}$ via (3.12).
5: Update $\boldsymbol{X}^{k+1}$ via (3.15).
6: Updating $\boldsymbol{F}_1^{k+1}$, $\boldsymbol{F}_2^{k+1}$, $\boldsymbol{F}_3^{k+1}$ and α^{k+1} via (3.16).
7: Set $k \leftarrow k+1$.
8: Until convergence

3.2 Truncated quaternion logarithmic norm sparse representation

The truncated nuclear norm can achieve a better approximation of the rank function than the nuclear norm. According to this property, the method adopted in this paper

combines the truncation skill and QLN, and the definition of truncated logarithmic norm based on quaternion is introduced below.

Definition 3.1 (TQLN, [10]) Given $\boldsymbol{X} \in \mathbb{Q}^{m\times n}$, the truncated logarithmic norm of the quaternion matrix with $0 \leq p \leq 1$ and $\epsilon > 0$ is defined as the sum of logarithmic function of $\min(M, N) - r$ minimum singular values:

$$\|\boldsymbol{X}\|_{L,r}^{p} = \sum_{i=r+1}^{\min(m,n)} \log(\sigma_i^p(\boldsymbol{X}) + \epsilon), \tag{3.17}$$

where $\sigma_i(\boldsymbol{X})$ denotes the i-th singular value of $\boldsymbol{X}$.

Given that the initial large singular values do not affect the rank, they are disregarded in TQLN. Subsequently, the focus shifts towards optimizing the smallest $\min(M, N) - r$ singular values to achieve a more precise low-rank estimation. According to TQLN principles, the completion process based on the low-rank minimization model (3.2) can be expressed as follows:

$$\min_{\boldsymbol{X}} \|\boldsymbol{X}\|_{L,r}^{p}, \text{ s.t. } \boldsymbol{B} \approx \boldsymbol{A}\boldsymbol{X}. \tag{3.18}$$

Lemma 3.1 [41] *Let $\boldsymbol{X} \in \mathbb{Q}^{m\times n}$, and the matrices $\boldsymbol{U} \in \mathbb{Q}^{r\times m}$ and $\boldsymbol{V} \in \mathbb{Q}^{r\times n}$ with $\boldsymbol{U}\boldsymbol{U}^H = I_r$, $\boldsymbol{V}\boldsymbol{V}^H = I_r$. Here, r is any integer ($r \leq \min(m, n)$). Then it has*

$$\left\|\mathrm{tr}(\boldsymbol{U}\boldsymbol{X}\boldsymbol{V}^H)\right\| \leq \sum_{i=1}^{r}\sigma_i(\boldsymbol{X}), \quad \max\left\|\mathrm{tr}(\boldsymbol{U}\boldsymbol{X}\boldsymbol{V}^H)\right\| = \sum_{i=1}^{r}\sigma_i(\boldsymbol{X}).$$

Based on Definition 3.1 and Lemma 3.1, we introduce a sparse representation model utilizing the quaternion-based framework:

$$\min_{\boldsymbol{X}} \lambda\|\boldsymbol{X}\|_{L}^{p} - \max_{\boldsymbol{C}\boldsymbol{C}^H=I_r, \boldsymbol{D}\boldsymbol{D}^H=I_r} \left\|\mathrm{tr}(\boldsymbol{C}\boldsymbol{X}\boldsymbol{D}^H)\right\|, \text{ s.t. } \boldsymbol{B} \approx \boldsymbol{A}\boldsymbol{X}. \tag{3.19}$$

The procedure is outlined in Algorithm 2.

In Algorithm 2, $\boldsymbol{C}^k$ and $\boldsymbol{D}^k$ are first obtained by QSVD of $\boldsymbol{X}^k$. Next, we will focus on **Step** 5 of Algorithm 2, which can be expressed as the following formula:

$$\min_{\boldsymbol{X},\boldsymbol{K}} \lambda\|\boldsymbol{X}\|_{L}^{p} - \left\|\mathrm{tr}(\boldsymbol{C}^k\boldsymbol{K}(\boldsymbol{D}^k)^H)\right\| + \frac{\rho}{2}\|\boldsymbol{B} - \boldsymbol{A}\boldsymbol{K}\|_F^2, \text{ s.t. } \boldsymbol{X} = \boldsymbol{K}. \tag{3.20}$$

It's worth noting that the problem (3.20) can be addressed using the ADMM framework. Initially, we tackle the problem (3.5) by minimizing the augmented Lagrangian function provided below:

$$\mathcal{L}(\boldsymbol{K}, \boldsymbol{X}, \boldsymbol{F}, \alpha) = \lambda\|\boldsymbol{X}\|_{L}^{p} - \left\|\mathrm{tr}(\boldsymbol{C}^k\boldsymbol{K}(\boldsymbol{D}^k)^H)\right\| + \frac{\rho}{2}\|\boldsymbol{B} - \boldsymbol{A}\boldsymbol{K}\|_F^2 + \Re(\langle \boldsymbol{F}, \boldsymbol{K} - \boldsymbol{X}\rangle) + \frac{\alpha}{2}\|\boldsymbol{K} - \boldsymbol{X}\|_F^2, \tag{3.21}$$

where α and $\boldsymbol{F}$ are a positive penalty parameter and a Lagrange multiplier, respectively.

Algorithm 2 TQLNSR algorithm.

Input: $\boldsymbol{A}, \boldsymbol{B} \in \mathbb{Q}^{m\times n}$, $\rho, \beta, \alpha_{\max}$
Output: $\boldsymbol{X}^{k+1}$
1: **Initialize** $\boldsymbol{X}^0$, $\boldsymbol{F}^0$, α^0
2: **Repeat**
3: Computing $[\hat{\boldsymbol{U}}^k, \sim, \hat{\boldsymbol{V}}^k] = \mathrm{QSVD}(\boldsymbol{X}^k)$, where

$$\hat{\boldsymbol{U}}^k = (\boldsymbol{u}_1, \cdots, \boldsymbol{u}_m) \in \mathbb{Q}^{m\times m}, \hat{\boldsymbol{V}}^k = (\boldsymbol{v}_1, \cdots, \boldsymbol{v}_n) \in \mathbb{Q}^{n\times n}.$$

4: Let $\boldsymbol{C}^k = (\boldsymbol{u}_1, \cdots, \boldsymbol{u}_r)^T \in \mathbb{Q}^{r\times m}$, $\boldsymbol{D}^k = (\boldsymbol{v}_1, \cdots, \boldsymbol{v}_r)^T \in \mathbb{Q}^{r\times n}$.
5: Solving $\boldsymbol{X}^{k+1} = \arg\min_{\substack{\boldsymbol{X}\\ \boldsymbol{B}=\boldsymbol{A}\boldsymbol{X}}} \lambda\|\boldsymbol{X}\|_L^p - \left\|\mathrm{tr}(\boldsymbol{C}^k\boldsymbol{X}(\boldsymbol{D}^k)^H)\right\|$.
6: Set $k \leftarrow k+1$.
7: Until convergence

Updating $\boldsymbol{K}$ In the $(k+1)$-th iteration, fixing the other variables at their latest values, $\boldsymbol{K}$ is the optimal solution of the following problem:

$$\begin{aligned}\boldsymbol{K}^{k+1} &= \arg\min_{\boldsymbol{K}} \frac{\rho}{2}\|\boldsymbol{B}-\boldsymbol{A}\boldsymbol{K}\|_F^2 + \frac{1}{2}\|\boldsymbol{K}-\boldsymbol{X}^k+\frac{1}{\alpha^k}\boldsymbol{F}^k\|_F^2 - \left\|\mathrm{tr}(\boldsymbol{C}^k\boldsymbol{K}(\boldsymbol{D}^k)^H)\right\|.\\ &= \arg\min_{\boldsymbol{K}} \frac{\rho}{2}\|\boldsymbol{B}-\boldsymbol{A}\boldsymbol{K}\|_F^2 + \frac{1}{2}\|\boldsymbol{K}-\boldsymbol{X}^k+\frac{1}{\alpha^k}(\boldsymbol{F}^k-(\boldsymbol{C}^k)^H\boldsymbol{D}^k)\|_F^2.\end{aligned} \tag{3.22}$$

Let

$$\mathcal{W}(\boldsymbol{K}) := \frac{\rho}{2}\|\boldsymbol{B}-\boldsymbol{A}\boldsymbol{K}\|_F^2 + \frac{1}{2}\|\boldsymbol{K}-\boldsymbol{X}^k+\frac{1}{\alpha^k}(\boldsymbol{F}^k-(\boldsymbol{C}^k)^H\boldsymbol{D}^k)\|_F^2.$$

Applying the relevant principles regarding quaternion matrix derivatives as outlined in [43], the gradient of $\mathcal{W}(\boldsymbol{K})$ can be calculated as

$$\frac{\partial\mathcal{W}(\boldsymbol{K})}{\partial\boldsymbol{K}} = -\rho\boldsymbol{A}^H(\boldsymbol{B}-\boldsymbol{A}\boldsymbol{K}) + \boldsymbol{K}-\boldsymbol{X}^k+\frac{1}{\alpha^k}(\boldsymbol{F}^k-(\boldsymbol{C}^k)^H\boldsymbol{D}^k). \tag{3.23}$$

Setting (3.23) to zero, we can obtain a unique solution

$$\boldsymbol{K}^{k+1} = [\rho\boldsymbol{I}+\boldsymbol{A}^H\boldsymbol{A}]^{-1}\big[\rho\boldsymbol{A}^H\boldsymbol{B}+\boldsymbol{X}^k+\frac{1}{\alpha^k}(\boldsymbol{F}^k-(\boldsymbol{C}^k)^H\boldsymbol{D}^k)\big]. \tag{3.24}$$

Updating $\boldsymbol{X}$ In the $(k+1)$-th iteration, while keeping the other variables constant at their most recent values, $\boldsymbol{X}^{k+1}$ represents the optimal solution of the subsequent problem:

$$\boldsymbol{X}^{k+1} = \arg\min_{\boldsymbol{X}} \frac{1}{2}\|\boldsymbol{K}^{k+1}+\frac{1}{\alpha^k}\boldsymbol{F}^k-\boldsymbol{X}\|_F^2 + \frac{\lambda}{\alpha^k}\|\boldsymbol{X}\|_L^p. \tag{3.25}$$

By Lemma 2.4, the QLSVT can be applied to (3.25) for updating $\boldsymbol{X}^{k+1}$ when $p = 1$, that is

$$\boldsymbol{X}^{k+1} = \boldsymbol{U}_{S_3} \boldsymbol{\Delta}_{\lambda,\alpha^k,\boldsymbol{\Sigma}_{S_3}} \boldsymbol{V}_{S_3}^H, \tag{3.26}$$

where $\boldsymbol{S}_3 = \boldsymbol{K}^{k+1} + \frac{1}{\alpha^k}\boldsymbol{F}^k$. Let $\boldsymbol{S}_3 = \boldsymbol{U}_{S_3}\boldsymbol{\Sigma}_{S_3}\boldsymbol{V}_{S_3}^H$ be the QSVD of quaternion matrices $\boldsymbol{S}_3$.

Updating $\boldsymbol{F}$ and $\boldsymbol{\alpha}$ The update formats are as follows:

$$\begin{cases} \boldsymbol{F}^{k+1} = \boldsymbol{F}^k + \alpha^k(\boldsymbol{X}^{k+1} - \boldsymbol{K}^{k+1}), \\ \alpha^{k+1} = \min(\beta\alpha^k, \alpha_{\max}). \end{cases} \tag{3.27}$$

Algorithm 3 outlines the complete process, detailing each step sequentially.

Algorithm 3 The ADMM optimization process is used in Step 5 of the TQLNSR algorithm.

Input: $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}^k$, $\boldsymbol{D}^k$, ρ, $\alpha_{\max}$, β
Output: $\boldsymbol{X}^{k+1}$
1: **Initialize** $\boldsymbol{X}^0$, $\boldsymbol{F}^0$, α^0
2: **Repeat**
3: Update $\boldsymbol{K}^{k+1}$ via (3.24).
4: Update $\boldsymbol{X}^{k+1}$ via (3.26).
5: Updating $\boldsymbol{F}^{k+1}$ and α^{k+1} via (3.27).
6: Set $k \leftarrow k + 1$.
7: Until convergence

The termination condition for Algorithms 1, 2 and 3 is defined as the following relative error:

$$\text{RE} := \frac{\|\boldsymbol{A}\boldsymbol{X}^k - \boldsymbol{B}\|_F^2}{\|\boldsymbol{X}^k\|_F^2} \leq \text{tol},$$

where tol > 0 is the stopping tolerance. In the experiments, we set tol=1e-4.

4 Convergence analysis

No definite convergence property of the ADMM has been established for non-convex problems (or convex problems involving more than two blocks of variables), even within the real number field. Therefore, concerning the formidable problems (3.5) and (3.20), we empirically demonstrate their convergence behavior. Additionally, alongside the empirical observations, we provide a weak convergence property for problem (3.5) (similarly for problem (3.20)), subject to mild conditions, as outlined in the following theorems.

Theorem 4.1 *Let* $\Theta_1 := (\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}, \boldsymbol{M}, \boldsymbol{N}, \boldsymbol{F}_1, \boldsymbol{F}_2, \boldsymbol{F}_3)$ *and* $\{\Theta_1^k\}_{k=1}^{\infty}$ *be generated by Algorithm* 1. *Assume that* $\{\Theta_1^k\}_{k=1}^{\infty}$ *is bounded, and* $\{\alpha^k\}_{k=1}^{\infty}$ *is non-decreasing and bounded. Then,*

(1) $\{\boldsymbol{X}^k\}, \{\boldsymbol{Y}^k\}, \{\boldsymbol{Z}^k\}, \{\boldsymbol{M}^k\}$ *and* $\{\boldsymbol{N}^k\}$ *are Cauchy sequences;*

(2) *any accumulation point of* $\{\Theta_1^k\}_{k=1}^{\infty}$ *satisfies the Karush-Kuhn-Tucker* (*KKT*) *conditions for the problem* (3.5).

Proof (1) According to (3.16), we have

$$\begin{aligned}
\boldsymbol{M}^{k+1} - \boldsymbol{Y}^{k+1} &= \frac{1}{\alpha^k}(\boldsymbol{F}_1^{k+1} - \boldsymbol{F}_1^k),\\
\boldsymbol{N}^{k+1} - \boldsymbol{Z}^{k+1} &= \frac{1}{\alpha^k}(\boldsymbol{F}_2^{k+1} - \boldsymbol{F}_2^k),\\
\boldsymbol{X}^{k+1} - \boldsymbol{M}^{k+1}(\boldsymbol{N}^{k+1})^H &= \frac{1}{\alpha^k}(\boldsymbol{F}_3^{k+1} - \boldsymbol{F}_3^k).
\end{aligned}$$

Due to the assumptions that the quaternion matrix sequences $\{\boldsymbol{F}_1^k\}$, $\{\boldsymbol{F}_2^k\}$ and $\{\boldsymbol{F}_3^k\}$ are bounded, we have

$$\begin{aligned}
\sum_{k=0}^{\infty} \|\boldsymbol{M}^{k+1} - \boldsymbol{Y}^{k+1}\|_F &\le \sum_{k=0}^{\infty} \frac{\alpha^{k+1}}{(\alpha^k)^2} \|\boldsymbol{F}_1^{k+1} - \boldsymbol{F}_1^k\|_F < \infty,\\
\sum_{k=0}^{\infty} \|\boldsymbol{N}^{k+1} - \boldsymbol{Z}^{k+1}\|_F &\le \sum_{k=0}^{\infty} \frac{\alpha^{k+1}}{(\alpha^k)^2} \|\boldsymbol{F}_2^{k+1} - \boldsymbol{F}_2^k\|_F < \infty,\\
\sum_{k=0}^{\infty} \|\boldsymbol{X}^{k+1} - \boldsymbol{M}^{k+1}(\boldsymbol{N}^{k+1})^H\|_F &\le \sum_{k=0}^{\infty} \frac{\alpha^{k+1}}{(\alpha^k)^2} \|\boldsymbol{F}_3^{k+1} - \boldsymbol{F}_3^k\|_F < \infty,
\end{aligned}$$

which imply that

$$\begin{aligned}
\lim_{k\to\infty} \|\boldsymbol{M}^{k+1} - \boldsymbol{Y}^{k+1}\|_F &= 0,\\
\lim_{k\to\infty} \|\boldsymbol{N}^{k+1} - \boldsymbol{Z}^{k+1}\|_F &= 0,\\
\lim_{k\to\infty} \|\boldsymbol{X}^{k+1} - \boldsymbol{M}^{k+1}(\boldsymbol{N}^{k+1})^H\|_F &= 0.
\end{aligned}$$

Hence, $\{(\boldsymbol{X}^k, \boldsymbol{Y}^k, \boldsymbol{Z}^k, \boldsymbol{M}^k, \boldsymbol{N}^k)\}$ indeed approaches to a feasible solution.
Next, we show that $\{\boldsymbol{M}^k\}$ and $\{\boldsymbol{N}^k\}$ are Cauchy sequences. Note that

$$\begin{cases}
\boldsymbol{F}_1^k = \boldsymbol{F}_1^{k-1} + \alpha^{k-1}(\boldsymbol{Y}^k - \boldsymbol{M}^k),\\
\boldsymbol{F}_2^k = \boldsymbol{F}_2^{k-1} + \alpha^{k-1}(\boldsymbol{Z}^k - \boldsymbol{N}^k),\\
\boldsymbol{F}_3^k = \boldsymbol{F}_3^{k-1} + \alpha^{k-1}(\boldsymbol{X}^k - \boldsymbol{M}^k(\boldsymbol{N}^k)^H).
\end{cases}$$

Then, from (3.12) and (3.13), we have

$$\begin{aligned}
&\frac{1}{2}\big(\boldsymbol{M}^{k+1}-\boldsymbol{Y}^k+\frac{1}{\alpha^k}\boldsymbol{F}_1^k\big)+\frac{1}{2}\big(\boldsymbol{M}^{k+1}(\boldsymbol{N}^k)^H-\boldsymbol{X}^k-\frac{1}{\alpha^k}\boldsymbol{F}_3^k\big)\boldsymbol{N}^k\\
=&\frac{1}{2}\big(\boldsymbol{M}^{k+1}-\boldsymbol{M}^k+\boldsymbol{M}^k-\boldsymbol{Y}^k+\frac{\boldsymbol{F}_1^k}{\alpha^k}\big)+\frac{1}{2}\big(\boldsymbol{M}^{k+1}(\boldsymbol{N}^k)^H-\boldsymbol{M}^k(\boldsymbol{N}^k)^H-\frac{1}{\alpha^{k-1}}\boldsymbol{F}_3^k\\
&+\frac{1}{\alpha^{k-1}}\boldsymbol{F}_3^{k-1}-\frac{1}{\alpha^k}\boldsymbol{F}_3^k\big)\boldsymbol{N}^k\\
=&\frac{1}{2}[(\boldsymbol{M}^{k+1}-\boldsymbol{M}^k)(I+(\boldsymbol{N}^k)^H(\boldsymbol{N}^k))+\frac{1}{\alpha^{k-1}}(\boldsymbol{F}_1^k-\boldsymbol{F}_1^{k-1})+\frac{1}{\alpha^k}\boldsymbol{F}_1^k-(\frac{1}{\alpha^{k-1}}(\boldsymbol{F}_3^k-\boldsymbol{F}_3^{k-1})\\
&+\frac{1}{\alpha^k}\boldsymbol{F}_3^k)\boldsymbol{N}^k]=O,
\end{aligned}\tag{4.1}$$

and

$$\begin{aligned}
&\frac{1}{2}\big(\boldsymbol{N}^{k+1}-\boldsymbol{Z}^k+\frac{1}{\alpha^k}\boldsymbol{F}_2^k\big)+\frac{1}{2}\big(\boldsymbol{N}^{k+1}(\boldsymbol{M}^{k+1})^H-(\boldsymbol{X}^k)^H-\frac{1}{\alpha^k}(\boldsymbol{F}_3^k)^H\big)\boldsymbol{M}^{k+1}\\
=&\frac{1}{2}\big(\boldsymbol{N}^{k+1}-\boldsymbol{N}^k+\boldsymbol{N}^k-\boldsymbol{Z}^k+\frac{1}{\alpha^k}\boldsymbol{F}_2^k\big)+\frac{1}{2}\big(\boldsymbol{N}^{k+1}(\boldsymbol{M}^{k+1})^H-\boldsymbol{N}^k(\boldsymbol{M}^k)^H-\frac{1}{\alpha^{k-1}}\boldsymbol{F}_3^k\\
&+\frac{1}{\alpha^{k-1}}\boldsymbol{F}_3^{k-1}-\frac{1}{\alpha^k}\boldsymbol{F}_3^k\big)\boldsymbol{M}^{k+1}\\
=&\frac{1}{2}[(\boldsymbol{N}^{k+1}-\boldsymbol{N}^k)(I+(\boldsymbol{M}^{k+1})^H(\boldsymbol{M}^{k+1}))+\frac{1}{\alpha^{k-1}}(\boldsymbol{F}_2^k-\boldsymbol{F}_2^{k-1})+\frac{1}{\alpha^k}\boldsymbol{F}_2^k\\
&+\boldsymbol{N}^k(\boldsymbol{M}^{k+1}-\boldsymbol{M}^k)^H\boldsymbol{M}^{k+1}-(\frac{1}{\alpha^{k-1}}(\boldsymbol{F}_3^k-\boldsymbol{F}_3^{k-1})+\frac{1}{\alpha^k}\boldsymbol{F}_3^k)\boldsymbol{M}^{k+1}]=O.
\end{aligned}\tag{4.2}$$

Based on (4.1) and (4.2), we can respectively obtain

$$\begin{aligned}
&\boldsymbol{M}^{k+1}-\boldsymbol{M}^k\\
&=2\big[\frac{1}{\alpha^{k-1}}(\boldsymbol{F}_1^{k-1}-\boldsymbol{F}_1^k)-\frac{1}{\alpha^k}\boldsymbol{F}_1^k+(\frac{1}{\alpha^{k-1}}(\boldsymbol{F}_3^k-\boldsymbol{F}_3^{k-1})+\frac{1}{\alpha^k}\boldsymbol{F}_3^k)\boldsymbol{N}^k\big][I+(\boldsymbol{N}^k)^H(\boldsymbol{N}^k)]^{-1}
\end{aligned}\tag{4.3}$$

and

$$\begin{aligned}
&\boldsymbol{N}^{k+1}-\boldsymbol{N}^k\\
&=2\big[\frac{1}{\alpha^{k-1}}(\boldsymbol{F}_2^{k-1}-\boldsymbol{F}_2^k)-\frac{1}{\alpha^k}\boldsymbol{F}_2^k+\boldsymbol{N}^k(\boldsymbol{M}^{k+1}-\boldsymbol{M}^k)^H\boldsymbol{M}^{k+1}+\boldsymbol{N}^k(\boldsymbol{M}^{k+1}-\boldsymbol{M}^k)^H\\
&\quad+(\frac{1}{\alpha^{k-1}}(\boldsymbol{F}_3^k-\boldsymbol{F}_3^{k-1})+\frac{1}{\alpha^k}\boldsymbol{F}_3^k)\boldsymbol{M}^{k+1}\big]\big[I+(\boldsymbol{M}^{k+1})^H(\boldsymbol{M}^{k+1})\big]^{-1}.
\end{aligned}\tag{4.4}$$

Recall that $\alpha^k=\min(\beta\alpha^{k-1},\alpha_{\max})$, it follows that $\{\alpha^k\}_{k=1}^{\infty}$ is non-decreasing and bounded, we have

$$\sum_{k=0}^{\infty}\|\boldsymbol{M}^{k+1}-\boldsymbol{M}^k\|_F\le\sum_{k=0}^{\infty}\frac{\eta_1}{\alpha^k}\le\sum_{k=0}^{\infty}\frac{\alpha^{k+1}\eta_1}{(\alpha^k)^2}<\infty,\tag{4.5}$$

where the constant η_1 is defined as

$$\begin{aligned}\eta_1 =&\max\{2(\beta\|\boldsymbol{F}_1^{k-1}-\boldsymbol{F}_1^k\|_F+\|\boldsymbol{F}_1^k\|_F+(\beta\|\boldsymbol{F}_3^k-\boldsymbol{F}_3^{k-1}\|_F+\|\boldsymbol{F}_3^k\|_F)\|\boldsymbol{N}^k\|_F)\\ &\|\big(I+(\boldsymbol{N}^k)^H(\boldsymbol{N}^k)\big)^{-1}\|_F, k=1,2,\cdots\}.\end{aligned}$$

And it has

$$\begin{aligned}&\sum_{k=0}^{\infty}\|\boldsymbol{N}^{k+1}-\boldsymbol{N}^k\|_F\le\sum_{k=0}^{\infty}\frac{\eta_2}{\alpha^k}+\sum_{k=0}^{\infty}\eta_3\|\boldsymbol{M}^k-\boldsymbol{M}^{k-1}\|_F\\ &\le\sum_{k=0}^{\infty}\frac{\alpha^{k+1}\eta_2}{(\alpha^k)^2}+\sum_{k=0}^{\infty}\eta_3\|\boldsymbol{M}^k-\boldsymbol{M}^{k-1}\|_F<\infty,\end{aligned}\tag{4.6}$$

where the constants η_2 and η_3 are defined as

$$\begin{aligned}\eta_2 =&\max\{2(\beta\|\boldsymbol{F}_2^{k-1}-\boldsymbol{F}_2^k\|_F+\|\boldsymbol{F}_2^k\|_F+(\beta\|\boldsymbol{F}_3^k-\boldsymbol{F}_3^{k-1}\|_F+\|\boldsymbol{F}_3^k\|_F)\|\boldsymbol{M}^{k+1}\|_F)\\ &\|\big(I+(\boldsymbol{M}^{k+1})^H(\boldsymbol{M}^{k+1})\big)^{-1}\|_F, k=1,2,\cdots\}\end{aligned}$$

and

$$\eta_3=\max\{2\|\boldsymbol{N}^k\|_F\|\boldsymbol{M}^{k+1}\|_F\|\big(I+(\boldsymbol{M}^{k+1})^H(\boldsymbol{M}^{k+1})\big)^{-1}\|_F, k=1,2,\cdots\}.$$

From (4.5) and (4.6), we know that $\lim_{k\to\infty}\|\boldsymbol{M}^{k+1}-\boldsymbol{M}^k\|_F=0$ and $\lim_{k\to\infty}\|\boldsymbol{N}^{k+1}-\boldsymbol{N}^k\|_F=0$. Hence, $\{\boldsymbol{M}^k\}$ and $\{\boldsymbol{N}^k\}$ are Cauchy sequences. Similarly, one can also verify $\{\boldsymbol{X}^k\}$ is Cauchy sequence.

Later, we establish that the sequences $\{\boldsymbol{Y}^k\}$, $\{\boldsymbol{Z}^k\}$ and $\{\boldsymbol{X}^k\}$ are Cauchy sequences. Let $\boldsymbol{U}_y^k\boldsymbol{\Sigma}_y^k(\boldsymbol{V}_y^H)^k$ be QSVD of the matrix $\boldsymbol{M}^k+\frac{1}{\alpha^k}\boldsymbol{F}_1^k$ in the $(k+1)$-th iteration. Then utilizing the QLSVT, we can get:

$$\boldsymbol{Y}^{k+1}=\boldsymbol{U}_y^k\boldsymbol{\Delta}_{\frac{\lambda}{2\alpha^k},\epsilon,\boldsymbol{\Sigma}_y^k}(\boldsymbol{V}_y^H)^k.\tag{4.7}$$

The soft thresholding operator $\boldsymbol{\Delta}_{\lambda,\epsilon,\boldsymbol{A}}$ is defined as:

$$\boldsymbol{\Delta}_{\lambda,\epsilon,\boldsymbol{A}}=\mathrm{diag}(l_{\lambda,\epsilon}(\sigma_1),\ l_{\lambda,\epsilon}(\sigma_2),\ \cdots,l_{\lambda,\epsilon}(\sigma_r),\ 0,\ \cdots,\ 0)\in\mathbb{R}^{m\times n}$$

with

$$l_{\lambda,\epsilon}(x)=\begin{cases}0, & \delta\le 0,\\ \underset{a\in\{0,\frac{1}{2}(x-\epsilon+\sqrt{\delta})\}}{\arg\min} h(a), & \delta>0,\end{cases}\tag{4.8}$$

where $\delta=(x-\epsilon)^2-4(\lambda-x\epsilon)$ and the function $h:\mathbb{R}^+\to\mathbb{R}^+$ is defined as $h(a):=\frac{1}{2}(a-x)^2+\lambda\log(a+\epsilon)$. Let x_0 present an arbitrary singular value ($x_0\ge 0$) and $g(\lambda,x_0)=x_0-l_{\lambda,\epsilon}(x_0)$, then we can obtain $\max\big(g(\lambda,x_0)\big)=2\sqrt{\lambda}$ as follows.

Case 1: $\delta \leq 0$. From the definition of δ, we have $0 \leq x_0 \leq 2\sqrt{\lambda} - \epsilon$. Moreover, $l_{\lambda,\epsilon}(x_0)$ in this case, so we have $\max\big(g(\lambda, x_0)\big) = 2\sqrt{\lambda}$.

Case 2: $\delta > 0$. In this case, $x_0 > 2\sqrt{\lambda} - \epsilon$ and we have

$$\max\big(g(\lambda, x_0)\big) = x_0 - \mathop{\arg\min}_{a \in \{0, \frac{1}{2}(x-\epsilon+\sqrt{\delta})\}} h(a) = x_0 - \frac{1}{2}(x_0 - \epsilon + \sqrt{\delta}).$$

Then it holds

$$\frac{\partial \max\big(g(\lambda, x_0)\big)}{\partial x_0} = \frac{1}{2}\Big(1 - \frac{x_0 + \epsilon}{\sqrt{(x_0+\epsilon)^2 - 4\lambda}}\Big) < 0.$$

We could know that $g(\lambda, x_0) < g(\lambda, 2\sqrt{\lambda} - \epsilon) = \sqrt{\lambda}$. For $\alpha^k > 0$, from (3.16), we have

$$\begin{aligned}
&\lim_{k\to\infty} \|\boldsymbol{Y}^{k+1} - \boldsymbol{Y}^k\|_F \\
&= \lim_{k\to\infty} \|\boldsymbol{M}^{k+1} + \frac{1}{\alpha^k}(\boldsymbol{F}_1^{k+1} - \boldsymbol{F}_1^k) - \boldsymbol{Y}^k\|_F \\
&= \lim_{k\to\infty} \|\boldsymbol{M}^{k+1} - \boldsymbol{M}^k + \frac{1}{\alpha^k}\boldsymbol{F}_1^{k+1} + \boldsymbol{M}^k - \boldsymbol{Y}^k + \frac{1}{\alpha^k}\boldsymbol{F}_1^k\|_F \\
&\leq \lim_{k\to\infty} (\|\boldsymbol{M}^{k+1} - \boldsymbol{M}^k\|_F + \frac{1}{\alpha^k}\|\boldsymbol{F}_1^{k+1}\|_F + \|\boldsymbol{M}^k - \boldsymbol{Y}^k + \frac{1}{\alpha^k}\boldsymbol{F}_1^k\|_F) \\
&= \lim_{k\to\infty} (\|\boldsymbol{M}^{k+1} - \boldsymbol{M}^k\|_F + \frac{1}{\alpha^k}\|\boldsymbol{F}_1^{k+1}\|_F + \|\boldsymbol{M}^k + \frac{1}{\alpha^k}\boldsymbol{F}_1^k - \boldsymbol{Y}^k\|_F) \\
&= \lim_{k\to\infty} (\|\boldsymbol{M}^{k+1} - \boldsymbol{M}^k\|_F + \frac{1}{\alpha^k}\|\boldsymbol{F}_1^{k+1}\|_F + \|\boldsymbol{U}_y^k\boldsymbol{\Sigma}_y^k(\boldsymbol{V}_y^H)^k - \boldsymbol{U}_y^k\boldsymbol{\Delta}_{\frac{\lambda}{2\alpha^k},\epsilon,\boldsymbol{\Sigma}_y^k}(\boldsymbol{V}_y^H)^k\|_F) \\
&= \lim_{k\to\infty} (\|\boldsymbol{M}^{k+1} - \boldsymbol{M}^k\|_F + \frac{1}{\alpha^k}\|\boldsymbol{F}_1^{k+1}\|_F + \|\boldsymbol{U}_y^k\big(\boldsymbol{\Sigma}_y^k - \boldsymbol{\Delta}_{\frac{\lambda}{2\alpha^k},\epsilon,\boldsymbol{\Sigma}_y^k}\big)(\boldsymbol{V}_y^H)^k\|_F) \\
&\leq \lim_{k\to\infty} \big(\frac{1}{\alpha^k}\|\boldsymbol{F}_1^{k+1}\|_F + \max(m,n)\sqrt{\frac{\lambda}{2\alpha^k}}\big) \\
&= 0.
\end{aligned}$$

Similarly, one can also verify that $\{\boldsymbol{Z}^k\}$ and $\{\boldsymbol{X}^k\}$ are Cauchy sequences.

(2) Let $(\boldsymbol{Y}_*, \boldsymbol{Z}_*, \boldsymbol{M}_*, \boldsymbol{N}_*, \boldsymbol{X}_*)$ be a stationary point of (3.5). Then, it satisfies the following KKT conditions

$$\begin{cases} O \in \frac{\lambda}{2}\partial\|\boldsymbol{Y}_*\|_L^1 + \boldsymbol{F}_{1*}, \ O \in \frac{\lambda}{2}\partial\|\boldsymbol{Z}_*\|_L^1 + \boldsymbol{F}_{2*}, \ O = \rho\boldsymbol{A}^H(\boldsymbol{B} - \boldsymbol{A}\boldsymbol{X}_*) + \boldsymbol{F}_{3*}, \\ \boldsymbol{X}_* = \boldsymbol{M}_*(\boldsymbol{N}_*)^H, \qquad \boldsymbol{Y}_* = \boldsymbol{M}_*, \qquad\qquad \boldsymbol{Z}_* = \boldsymbol{N}_*. \end{cases} \tag{4.9}$$

Subsequently, we will confirm that every limit point of $(\boldsymbol{Y}^k, \boldsymbol{Z}^k, \boldsymbol{M}^k, \boldsymbol{N}^k, \boldsymbol{X}^k)$ aforementioned KKT conditions.

From (3.11) and (3.13), we can respectively obtain

$$\begin{aligned} O &\in \frac{\lambda}{2\alpha^k}\partial\|\boldsymbol{Y}^{k+1}\|_L^1 + (\boldsymbol{M}^{k+1} - \boldsymbol{Y}^{k+1} + \frac{\boldsymbol{F}_1^k}{\alpha^k}), \\ O &\in \frac{\lambda}{2\alpha^k}\partial\|\boldsymbol{Z}^{k+1}\|_L^1 + (\boldsymbol{N}^{k+1} - \boldsymbol{Z}^{k+1} + \frac{\boldsymbol{F}_2^k}{\alpha^k}), \\ O &= \rho\boldsymbol{A}^H(\boldsymbol{B} - \boldsymbol{A}\boldsymbol{X}^{k+1}) + (\boldsymbol{X}^{k+1} - \boldsymbol{M}^{k+1}(\boldsymbol{N}^{k+1})^H + \frac{\boldsymbol{F}_3^k}{\alpha^k}). \end{aligned} \tag{4.10}$$

Since $\{\boldsymbol{Y}^k\}, \{\boldsymbol{Z}^k\}, \{\boldsymbol{M}^k\}$, $\{\boldsymbol{N}^k\}$ and $\{\boldsymbol{X}^k\}$ are Cauchy sequences, we can let $\boldsymbol{Y}^\infty, \boldsymbol{Z}^\infty, \boldsymbol{M}^\infty, \boldsymbol{N}^\infty$ and $\boldsymbol{X}^\infty$ be their accumulation points, respectively. Then, together with the results in Algorithm 1, we have that $\boldsymbol{Y}^\infty = \boldsymbol{M}^\infty$, $\boldsymbol{Z}^\infty = \boldsymbol{N}^\infty$, $\boldsymbol{X}^\infty = \boldsymbol{Y}^\infty(\boldsymbol{Z}^\infty)^H$. Thus, when $k \to \infty$, (4.10) becomes

$$\begin{aligned} O &\in \frac{\lambda}{2}\partial\|\boldsymbol{Y}^\infty\|_L^1 + \boldsymbol{F}_1^\infty, \\ O &\in \frac{\lambda}{2}\partial\|\boldsymbol{Z}^\infty\|_L^1 + \boldsymbol{F}_2^\infty, \\ O &= \rho\boldsymbol{A}^H(\boldsymbol{B} - \boldsymbol{A}\boldsymbol{X}^\infty) + \boldsymbol{F}_3^\infty. \end{aligned} \tag{4.11}$$

Consequently, any accumulation point $\{\boldsymbol{Y}^\infty, \boldsymbol{Z}^\infty, \boldsymbol{M}^\infty, \boldsymbol{N}^\infty, \boldsymbol{X}^\infty\}$ of the sequence $\{(\boldsymbol{Y}^k, \boldsymbol{Z}^k, \boldsymbol{M}^k, \boldsymbol{N}^k, \boldsymbol{X}^k)\}$ generated by Algorithm 1 indeed satisfies the KKT conditions for the problem (3.5). This proof is complete. □

Below we give the weak convergence property of the problem (3.20), but under mild conditions, as described in the following theorem.

Theorem 4.2 *Let* $\Theta_2 := (\boldsymbol{K}, \boldsymbol{X}, \boldsymbol{F})$ *and* $\{\Theta_2^k\}_{k=1}^\infty$ *be generated by Algorithm* 3. *Assume that* $\{\Theta_2^k\}_{k=1}^\infty$ *is bounded, and* $\{\alpha^k\}_{k=1}^\infty$ *is non-decreasing and bounded. Then,*

(1) $\{\boldsymbol{K}^k\}$ *and* $\{\boldsymbol{X}^k\}$ *are Cauchy sequences;*

(2) *any accumulation point of* $\{\Theta_2^k\}_{k=1}^\infty$ *satisfies the Karush-Kuhn-Tucker* (*KKT*) *conditions for the problem* (3.20).

Proof (1) According to (3.27), we have

$$\boldsymbol{K}^{k+1} - \boldsymbol{X}^{k+1} = \frac{1}{\alpha^k}(\boldsymbol{F}^{k+1} - \boldsymbol{F}^k).$$

Due to the assumptions that $\{\boldsymbol{F}^k\}$ is bounded and $\{\alpha^k\}_{k=1}^\infty$ is non-decreasing and bounded, we have

$$\sum_{k=0}^{\infty}\|\boldsymbol{K}^{k+1} - \boldsymbol{X}^{k+1}\|_F \le \sum_{k=0}^{\infty}\frac{\alpha^{k+1}}{(\alpha^k)^2}\|\boldsymbol{F}^{k+1} - \boldsymbol{F}^k\|_F < \infty,$$

which implies that

$$\lim_{k\to\infty}\|\boldsymbol{K}^{k+1} - \boldsymbol{X}^{k+1}\|_F = 0.$$

Hence, $\{(\boldsymbol{K}^k, \boldsymbol{X}^k)\}$ indeed approaches to a feasible solution.

Next, we show that $\{\boldsymbol{K}^k\}$ and $\{\boldsymbol{X}^k\}$ are Cauchy sequences. Let $\boldsymbol{U}_x^k \boldsymbol{\Sigma}_x^k (\boldsymbol{V}_x^H)^k$ be the QSVD of $\boldsymbol{K}^k + \frac{1}{\alpha^k}\boldsymbol{F}^k$ in the $(k+1)$-th iteration.

For $\alpha^k > 0$, from (3.27) and the proof of Theorem 4.1, we know that

$$
\begin{aligned}
&\lim_{k\to\infty} \|\boldsymbol{X}^{k+1} - \boldsymbol{X}^k\|_F \\
&= \lim_{k\to\infty} \|\boldsymbol{K}^{k+1} + \frac{1}{\alpha^k}(\boldsymbol{F}^{k+1} - \boldsymbol{F}^k) - \boldsymbol{X}^k\|_F \\
&= \lim_{k\to\infty} \|\boldsymbol{K}^{k+1} - \boldsymbol{K}^k + \frac{1}{\alpha^k}\boldsymbol{F}^{k+1} + \boldsymbol{K}^k - \boldsymbol{X}^k + \frac{1}{\alpha^k}\boldsymbol{F}^k\|_F \\
&\le \lim_{k\to\infty} (\|\boldsymbol{K}^{k+1} - \boldsymbol{K}^k\|_F + \frac{1}{\alpha^k}\|\boldsymbol{F}^{k+1}\|_F + \|\boldsymbol{K}^k - \boldsymbol{X}^k + \frac{1}{\alpha^k}\boldsymbol{F}^k\|_F) \\
&= \lim_{k\to\infty} (\|\boldsymbol{K}^{k+1} - \boldsymbol{K}^k\|_F + \frac{1}{\alpha^k}\|\boldsymbol{F}^{k+1}\|_F + \|\boldsymbol{K}^k + \frac{1}{\alpha^k}\boldsymbol{F}^k - \boldsymbol{X}^k\|_F) \\
&= \lim_{k\to\infty} (\|\boldsymbol{K}^{k+1} - \boldsymbol{K}^k\|_F + \frac{1}{\alpha^k}\|\boldsymbol{F}^{k+1}\|_F + \|\boldsymbol{U}_x^k \boldsymbol{\Sigma}_x^k (\boldsymbol{V}_x^H)^k - \boldsymbol{U}_x^k \boldsymbol{\Delta}_{\frac{\lambda}{\alpha^k},\epsilon,\boldsymbol{\Sigma}_x^k} (\boldsymbol{V}_x^H)^k\|_F) \\
&= \lim_{k\to\infty} (\|\boldsymbol{K}^{k+1} - \boldsymbol{K}^k\|_F + \frac{1}{\alpha^k}\|\boldsymbol{F}^{k+1}\|_F + \|\boldsymbol{U}_x^k \big(\boldsymbol{\Sigma}_x^k - \boldsymbol{\Delta}_{\frac{\lambda}{\alpha^k},\epsilon,\boldsymbol{\Sigma}_x^k}\big)(\boldsymbol{V}_x^H)^k\|_F) \\
&\le \lim_{k\to\infty} \big(\frac{1}{\alpha^k}\|\boldsymbol{F}^{k+1}\|_F + \max(m,n)\sqrt{\frac{\lambda}{\alpha^k}}\big) \\
&= 0.
\end{aligned}
$$

Similarly, one can also verify that $\{\boldsymbol{H}^k\}$ is Cauchy sequence.

(2) Let $(\boldsymbol{K}_*, \boldsymbol{X}_*)$ be a stationary point of (3.20). Then, it satisfies the following KKT conditions

$$
\begin{cases}
O \in \lambda \partial \|\boldsymbol{X}_*\|_L^1 + \boldsymbol{F}_*, \\
O = \rho \boldsymbol{A}^H (\boldsymbol{B} - \boldsymbol{A}\boldsymbol{K}_*) + \boldsymbol{F}_*, \\
\boldsymbol{X}_* = \boldsymbol{K}_*.
\end{cases} \tag{4.12}
$$

Subsequently, we will confirm that every limit point of $(\boldsymbol{K}^k, \boldsymbol{X}^k)$ aforementioned KKT conditions.

From (3.22) and (3.25), we can respectively obtain

$$
\begin{aligned}
&O \in \frac{\lambda}{\alpha^k}\partial \|\boldsymbol{X}^{k+1}\|_L^1 + (\boldsymbol{K}^{k+1} - \boldsymbol{X}^{k+1} + \frac{1}{\alpha^k}\boldsymbol{F}^k), \\
&O = \rho \boldsymbol{A}^H (\boldsymbol{B} - \boldsymbol{A}\boldsymbol{K}^{k+1}) + \big(\boldsymbol{K}^{k+1} - \boldsymbol{X}^{k+1} + \frac{1}{\alpha^k}(\boldsymbol{F}^k - \boldsymbol{C}^H \boldsymbol{D})\big).
\end{aligned} \tag{4.13}
$$

Since $\{\boldsymbol{K}^k\}$ and $\{\boldsymbol{X}^k\}$ are Cauchy sequences, let $\boldsymbol{K}^\infty$ and $\boldsymbol{X}^\infty$ be their accumulation points, respectively. Then, together with the results in Algorithm 3, we have that $\boldsymbol{K}^\infty = \boldsymbol{X}^\infty$. Thus, when $k \to \infty$, (4.13) becomes

$$\begin{aligned} O &\in \lambda \partial \|\boldsymbol{X}^\infty\|_L^1 + \boldsymbol{F}^\infty, \\ O &= \rho \boldsymbol{A}^H(\boldsymbol{B} - \boldsymbol{A}\boldsymbol{K}^\infty) + \boldsymbol{F}^\infty. \end{aligned} \tag{4.14}$$

Consequently, any accumulation point $\{\boldsymbol{K}^\infty, \boldsymbol{X}^\infty\}$ of the sequence $\{(\boldsymbol{K}^k, \boldsymbol{X}^k)\}$ generated by Algorithm 3 indeed satisfies the KKT conditions for the problem (3.20). □

5 Numerical experiments

In this section, based on the discussions in Sections 3 and 4, we give some numerical examples of color image sparse representation to prove the feasibility and effectiveness of Algorithms 1-3. We implemented all algorithms in MATLAB R2020a on a personal computer with Inter(R) Core(TM) i7-10700 CPU @ 2.90GHz and 8 GB memory.

Let $\boldsymbol{A} = \text{rand}\,(n, n) + \text{rand}\,(n, n)\,\boldsymbol{i} + \text{rand}\,(n, n)\,\boldsymbol{j} + \text{rand}\,(n, n)\,\boldsymbol{k} \in \mathbb{Q}^{n\times n}$ with $n = 20$. We employ Algorithms 1-3 to individually compute the reconstructed quaternion matrix and random matrix under the models (3.5) and (3.19). The three-dimensional correlation between the target value and the parameter values α and ρ is illustrated in Figs. 2 and 3.

From Figs. 2 and 3, it's evident that the matrices reconstructed by our proposed Algorithms 1-3 consistently yield lower target values within the model compared to the random matrices. Therefore, the feasibility of our proposed algorithms is demonstrated. We represent a color image as $\boldsymbol{A} = R\boldsymbol{i} + G\boldsymbol{j} + B\boldsymbol{k} \in \mathbb{Q}^{m\times n}$, where R, G and B respectively represent the real matrix corresponding to the red, green and blue channels in the color image. The original color images selected in the experiments are $64 \times 64 \times 3$ pixels. Obviously, every color image matrix is a pure imaginary

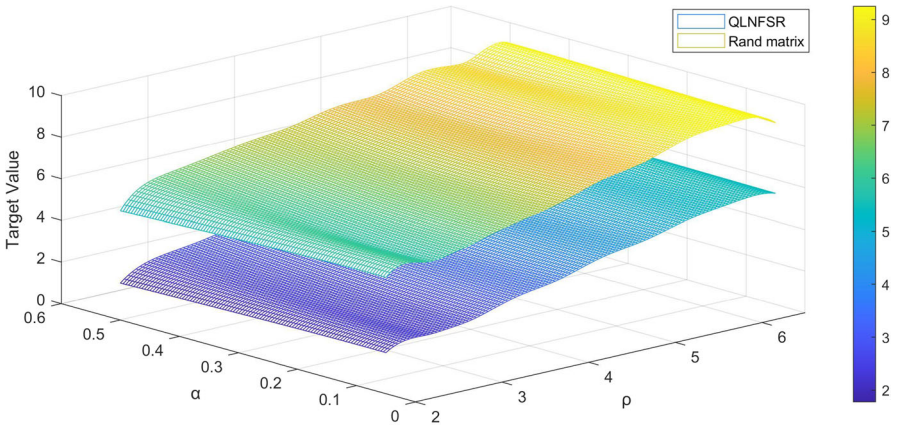

Fig. 2 Comparison of target values between quaternion matrix reconstructed by QLNFSR algorithm and random quaternion matrix under different parameters

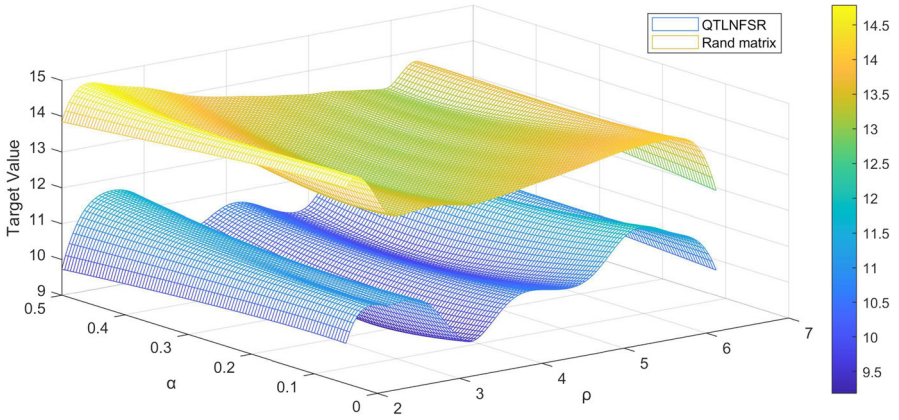

Fig. 3 Comparison of target values between quaternion matrix reconstructed by QTLNSR algorithm and random quaternion matrix under different parameters

quaternion matrix. Using this representation method, we can make better use of the relationship between the three channels of the color image and process a color image as a whole. We compare the proposed algorithm with truncated QSVD algorithm [44] and quaternion nuclear norm algorithm [45].

Parameters and initialization setting we set $\boldsymbol{B} = \boldsymbol{A}$, $\lambda = 0.05\sqrt{\max(m,n)}$, $\rho = 0.05$, $d = 10$, $\alpha_{\max} = 10^7$ and $\beta = 1.03$. Let $\alpha^0 = 0.01$, $\boldsymbol{X}_0 = \mathbf{randQ}(m,n)$, $[\boldsymbol{U}^x, \sim, \boldsymbol{V}^x] = \mathbf{svdQ}(\boldsymbol{X}_0)$, $\boldsymbol{M}^0 = \boldsymbol{Y}^0 = \boldsymbol{U}^x(1:d,:)$, $\boldsymbol{N}^0 = \boldsymbol{Z}^0 = \boldsymbol{V}^x(1:d,:)$ and $\boldsymbol{F}_1^0 = \boldsymbol{F}_2^0 = \boldsymbol{F}_3^0 = \mathbf{0}$. In TQLNSR, as the exact number of truncated singular values is unknown beforehand, experimenting with $r = 1$ can yield useful insights. Adjusting r appropriately can lead to improved outcomes. The randomly generated quaternion matrix function **randQ** and the quaternion singular value decomposition function **svdQ** are derived from the Structure-preserving Quaternion Toolbox [46].

Quantitative assessment To assess the effectiveness of the proposed methods, we utilize three commonly employed quantitative quality metrics, namely the peak signal-to-noise ratio (PSNR), the mean structural similarity index (MSSIM) and the sparsity of the proportion of zero elements, in addition to evaluating visual quality. The calculation formulas are as follows.

(1) PSNR

$$\mathrm{PSNR} = 10\log_{10}\left(\frac{L^2}{\mathrm{MSE}}\right), \mathrm{MSE} = \frac{1}{3mn}\sum_{h=1}^{3}\sum_{w=1}^{m}\sum_{u=1}^{n}[X_h(w,u) - Y_h(w,u)]^2,$$

where L is the maximum value of the data type of the color image. MSE is the mean square error, where $\boldsymbol{X} = X_1\boldsymbol{i} + X_2\boldsymbol{j} + X_3\boldsymbol{k} \in \mathbb{Q}^{m\times n}$, $\boldsymbol{Y} = Y_1\boldsymbol{i} + Y_2\boldsymbol{j} + Y_3\boldsymbol{k} \in \mathbb{Q}^{m\times n}$ and (w,u) represent the original image, the reconstructed image and the coordinates,

respectively. m and n are the number of rows and columns of the image matrix, respectively.

Generally speaking, PSNR in the range of 30(dB) to 40(dB) indicates good image quality. That is, the distortion is perceptible but acceptable. When PSNR is higher than 40(dB), the image quality is excellent. The encrypted image is very close to the original image.

(2) MSSIM

$$\mathrm{SSIM}(X_h, Y_h) = \frac{\big(2\mu(X_h)\mu(Y_h) + c_1\big)\big(2\sigma(X_h, Y_h) + c_2\big)}{\big(\mu^2(X_h) + \mu^2(Y_h) + c_1\big)(\sigma^2(X_h) + \sigma^2(Y_h) + c_2)},$$

$$\mathrm{MSSIM}(\boldsymbol{X}, \boldsymbol{Y}) = \frac{1}{3}\sum_{h=1}^{3} \mathrm{SSIM}(X_h, Y_h),$$

where $\mu(X_h)/\mu(Y_h)$ and $\sigma(X_h)/\sigma(Y_h)$ represent the mean value and standard deviation of X_h/Y_h with $h = 1, 2, 3$; $\sigma(X_h, Y_h)$ denotes the covariance matrix of X_h and Y_h. c_1 and c_2 are constants in order to avoid denominators of 0 and maintain stability. The value of MSSIM ranges from 0 to 1. Notice that it has a well similarity of two images while the value of MSSIM is closed to 1.

(3) Sparsity

$$\mathrm{Sparsity} = \frac{m \times n - T_z}{m \times n},$$

where m and n represent the dimensions of the quaternion matrix row and column, respectively. T_z represents the number of non-zero elements of the quaternion matrix. The sparsity measure ranges between 0 and 1. A sparsity value closer to 0 indicates a denser matrix, while a value closer to 1 indicates a sparser matrix.

The reconstruction results about PSNR, MSSIM, Sparsity and CPU time(s) for four test methods on eight test images are shown in Table 1. When the value of PSNR is inf, the image similarity is very high. Additionally, Fig. 4 illustrates the visual contrast between the four test methods on the eight test color images. As can be seen from Fig. 4 and Table 1, Algorithms 1-3 proposed in this paper can be well applied to the reconstruction after sparse representation of color images.

6 Conclusions

In conclusion, this paper introduces a novel approach for image restoration using quaternion matrix framework and logarithmic norm. By representing color images in a pure quaternion matrix, the proposed method preserves image structure while approximating rank efficiently. Furthermore, leveraging factorization and truncation techniques based on the logarithmic norm ensures effective image recovery. The alternate minimization framework facilitates optimization of these techniques, with

Table 1 Color image (Male, Blossom, Baboon, Peppers, Female, Tree, Splash and Plane) sparse representation results in sparse representation of various indicators in different algorithms, among which algorithms include QSVD, QSVT, QLNFSR and TQLNSR

Images	Indexes	QSVD	QSVT	QLNFSR	TQLNSR	Images	Indexes	QSVD	QSVT	QLNFSR	TQLNSR
Male	PSNR	30.6364	47.8182	45.1478	**Inf**	Female	PSNR	35.1398	**50.8157**	46.6618	47.3604
	MSSIM	0.9976	**0.9982**	**0.9982**	**0.9982**		MSSIM	0.9895	0.9896	**0.9897**	0.9895
	Sparsity	–	0.2197	5.8350	**62.3169**		Sparsity	–	0.6470	8.3923	**36.4990**
	Time(s)	**0.4700**	34.5430	49.1700	42.1500		Time(s)	**0.0620**	35.7130	51.4000	163.5470
Blossom	PSNR	24.1799	48.8771	48.4918	**Inf**	Tree	PSNR	22.6504	49.0091	47.8476	**Inf**
	MSSIM	0.9934	0.9968	0.9968	**0.9972**		MSSIM	0.9982	**0.9988**	**0.9988**	**0.9988**
	Sparsity	–	7.6355	6.0242	**40.4968**		Sparsity	–	0.2563	7.4158	**70.1904**
	Time(s)	**0.0880**	33.1880	47.2740	33.5910		Time(s)	**0.0770**	36.8030	51.7320	45.6770
Baboon	PSNR	28.5914	48.5548	46.5395	**Inf**	Splash	PSNR	31.3718	49.0811	46.6514	**62.5420**
	MSSIM	0.9968	**0.9970**	**0.9970**	**0.9970**		MSSIM	0.9970	0.9970	0.9970	**0.9971**
	Sparsity	–	0.1648	5.1575	**84.6130**		Sparsity	–	0.1343	**5.5908**	3.4546
	Time(s)	**0.0830**	33.8530	49.4230	57.5320		Time(s)	**0.0590**	34.7040	49.0140	211.1460
Peppers	PSNR	24.5930	48.6894	47.3019	**Inf**	Plane	PSNR	26.2500	48.9213	46.1655	**67.0666**
	MSSIM	0.9949	0.9969	0.9969	**0.9970**		MSSIM	0.9960	0.9973	0.9973	**0.9974**
	Sparsity	–	5.9082	5.7739	**22.5708**		Sparsity	–	0.1221	7.9468	**14.2151**
	Time(s)	**0.0550**	33.6900	48.5150	32.5380		Time(s)	**0.0770**	33.9470	49.7050	220.7000

The bold entries indicate better data results in the comparison experiment

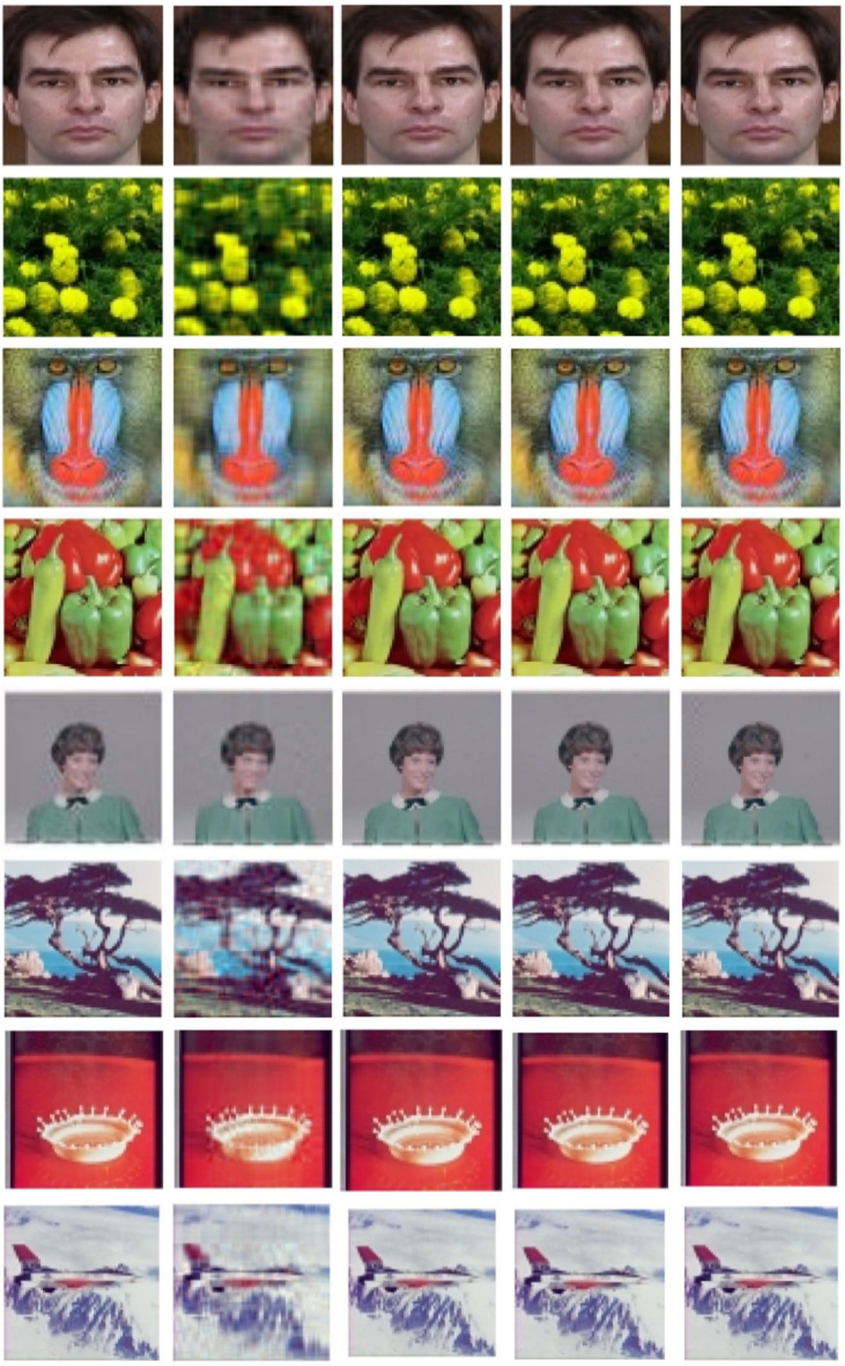

Fig. 4 Color image (Male, Blossom, Baboon, Peppers, Female, Tree, Splash and Plane) sparse representation results after reconstruction. The sequence is the results of the original image, QSVD, QSVT, QLNFSR and TQLNSR algorithms from left to right, respectively

rigorous mathematical analysis validating convergence. Numerical examples provided in the paper demonstrate the efficacy of the proposed algorithms in practice, highlighting their potential for various applications in image processing and computer vision.

Acknowledgements The authors are grateful for the Editor-in-Chief, Associate Editor, and Reviewers for their valuable comments and insightful suggestions that helped to improve this research significantly.

Author Contributions All authors contributed to the study conception and design. All authors performed material preparation, data collection, and analysis. The authors read and approved the final manuscript.

Funding Information This research is supported by the National Natural Science Foundation of China (Nos. 62105064 and 12371378) and the Natural Science Foundation of Fujian Province, China (Nos. 2023J011127 and 2023J01955).

Data Availability No datasets were generated or analysed during the current study.

Declarations

Conflict of Interest The author declares that they have no conflict of interest.

Ethical Approval This manuscript does not contain any studies with human participants or animals performed by any of the authors.

Competing Interests The authors declare no competing interests

References

1. Gao, L.L., Song, J.K., Liu, X.Y., Shao, J.M., Liu, J.J., Shao, J.: Learning in high-dimensional multimedia data: the state of the art. Multimed Syst. **23**, 303–313 (2017)
2. Kompella, V.R., Stollenga, M., Luciw, M., Schmidhuber, J.: Continual curiosity-driven skill acquisition from high-dimensional video inputs for humanoid robots. Artif. Intell. **247**, 313–335 (2017)
3. Zhang, C., Liu, Y.A., Wu, F., Fan, W.H., Tang, J.L., Liu, H.S.: Multi-dimensional joint prediction model for IoT sensor data search. IEEE Access. **7**, 90863–90873 (2019)
4. Tillquis, R.C., Lladser, M.E.: Low-dimensional representation of genomic sequences. J. Math. Biol. **79**, 1–29 (2019)
5. Ray, P., Reddy, S.S., Banerjee, T.: Various dimension reduction techniques for high dimensional data analysis: a review. Artif Intell Rev. **54**, 3473–3515 (2021)
6. Zhang, L., Lin, J., Karim, R.: An angle-based subspace anomaly detection approach to high-dimensional data: with an application to industrial fault detection. Reliab. Eng. Syst. Safe. **142**, 482–497 (2015)
7. Reddy, G.T., Reddy, M.P.K., Lakshmanna, K., Kaluri, R.: Analysis of dimensionality reduction techniques on big data. IEEE Access. **8**, 54776–54788 (2020)
8. Ayesha, S., Hanif, M.K., Talib, R.: Overview and comparative study of dimensionality reduction techniques for high dimensional data. Inf. Fusion. **59**, 44–58 (2020)
9. Yu, Z.Y., Zheng, X.P., Huang, F.W., Guo, W.Z., Lin, S., Yu, Z.W.: A framework based on sparse representation model for time series prediction in smart city. Front. Comput. Sci. **15**, 1–13 (2021)
10. Zhang, Z., Xu, Y., Yang, J., Li, X.L., Zhang, D.: A survey of sparse representation: algorithms and applications. IEEE Access. **3**, 490–530 (2015)
11. Bai, T., Li, Y.F.: Robust visual tracking with structured sparse representation appearance model. Pattern Recogniti. **45**, 2390–2404 (2012)
12. Zhang, L., Zhang, L., Du, B.: Deep learning for remote sensing data: a technical tutorial on the state of the art. IEEE Trans. Geosci. Remote Sens. **4**, 22–40 (2016)
13. Jia, X.X., Feng, X.C., Wang, W.W.: Rank constrained nuclear norm minimization with application to image denoising. Signal Process. **129**, 1–11 (2016)

14. Liu, Y.Y., Zhao, X.L., Zheng, Y.B., Ma, T.H., Zhang, H.: Hyperspectral image restoration by tensor fibered rank constrained optimization and plug-and-play regularization. IEEE Trans. Geosci. Remote. **60**, 1–17 (2021)
15. Hui, K.F., Shen, X.J., Abhadiomhen, S.E., Zhan, Y.Z.: Robust low-rank representation via residual projection for image classification. Knowl Based Syst. **241**, 108230 (2022)
16. Kondo, Y., Kubo, Y., Takamune, N., Kitamura, D., Saruwatari, H.: Deficient-basis-complementary rank-constrained spatial covariance matrix estimation based on multivariate generalized Gaussian distribution for blind speech extraction. EURASIP J. Adv. Signal Process. **1**, 88–112 (2022)
17. Min, G., Zhang, X. W., Yang, J. B., Han, W., Zou, X.: A perceptually motivated approach via sparse and low-rank model for speech enhancement. 2016 IEEE International Conference on Multimedia and Expo (ICME), pp. 1-6 (2016)
18. Abood, E.W., Hussien, Z.A., Kawi, H.A., Abduljabbar, Z.A., Nyangaresi, V.O., Kalafy, S.A.: Provably secure and efficient audio compression based on compressive sensing. Int. J. Electr. Comput. Eng. (IJECE). **13**, 335–346 (2023)
19. Asari, H., Pearlmutter, B.A., Zador, A.M.: Sparse representations for the cocktail party problem. J. Neurosci. **26**, 7477–7490 (2006)
20. Han, S.G., Wang, N., Guo, Y.X., Tang, F.R., Xu, L., Ju, Y., Shi, L.: Application of sparse representation in bioinformatics. Front Genet. **12**, 810875 (2021)
21. Pique-Regi, R., Monso-Varona, J., Ortega, A., Seeger, R.C., Triche, T.J., Asgharzadeh, S.: Sparse representation and Bayesian detection of genome copy number alterations from microarray data. Bioinformatics. **24**, 309–318 (2008)
22. Hu, Y., Zhang, D.B., Ye, J.P., Li, X.L., He, X.F.: Fast and accurate matrix completion via truncated nuclear norm regularization. IEEE Trans. Pattern Anal. Mach. Intell. **35**, 2117–2130 (2012)
23. Xie, Y., Gu, S.H., Liu, Y., Zuo, W.M., Zhang, W.S., Zhang, L.: Weighted Schatten p-norm minimization for image denoising and background subtraction. IEEE Trans. Image Process. **25**, 4842–4857 (2016)
24. Gu, S.H., Xie, Q., Meng, D.Y., Zuo, W.M., Feng, X.C., Zhang, L.: Weighted nuclear norm minimization and its applications to low level vision. Int. J. Comput Vision. **121**, 183–208 (2017)
25. Kang, Z., Peng, C., Cheng, J., Cheng, Q.: Logdet rank minimization with application to subspace clustering. Comput. Intel. Neurosc. **1**, 1–10 (2015)
26. Shang, F., Cheng, J., Liu, Y., Luo, Z.Q., Lin, Z.: Bilinear factor matrix norm minimization for robust PCA: algorithms and applications. IEEE Trans. Pattern Anal. Mach. Intell. **40**, 2066–2080 (2017)
27. Ke, Y.F., Ma, C.F., Jia, Z.G., Xie, Y.J., Liao, R.W.: Quasi non-negative quaternion matrix factorization with application to color face recognition. J. Sci. Comput. **95**, 38 (2023)
28. Jia, Z.G., Ma, R.R., Zhao, M.X.: A new structure-preserving method for recognition of color face images. Computer Science and Artificial Intelligence: Proceedings of the International Conference on Computer Science and Artificial Intelligence (CSAI2016), pp. 427-432 (2018)
29. Liu, D.J., Pu, G.L., Wu, X.Y.: Quaternion-based improved cuckoo algorithm for colour UAV image edge detection. IET Image Process. **16**, 926–935 (2022)
30. Jia, Z.G., Ng, M.K., Wang, W.: Color image restoration by saturation-value total variation. SIAM J. Imaging Sci. **12**, 972–1000 (2019)
31. Xu, X., Zhang, Z., Crabbe, M.J.C.: Quaternion quasi-Chebyshev non-local means for color image denoising. Chinese J. Electron. **32**, 1–18 (2023)
32. Miao, J.F., Kou, K.I.: Color image recovery using low-rank quaternion matrix completion algorithm. IEEE Trans. Image Process. **31**, 190–201 (2021)
33. Miao, J.F., Kou, K.I., Cheng, D., Liu, W.K.: Quaternion higher-order singular value decomposition and its applications in color image processing. Inform Fusion. **92**, 139–153 (2023)
34. Miao, J.F., Kou, K.I., Yang, L.Q., Han, J.: Quaternion matrix completion using untrained quaternion convolutional neural network for color image inpainting. Signal Process. **221**, 109504 (2024)
35. Miao, J.F., Kou, K.I.: Quaternion-based bilinear factor matrix norm minimization for color image inpainting. IEEE Trans. Signal Process. **68**, 5617–5631 (2020)
36. Fazel, M., Hindi, H., Boyd, S. P.: Log-det heuristic for matrix rank minimization with applications to Hankel and Euclidean distance matrices. Proc. of the 2003 American Control Conf. IEEE. **3**, 2156-2162 (2003)
37. Chen, L., Jiang, X., Liu, X.Z., Zhou, Z.X.: Logarithmic norm regularized low-rank factorization for matrix and tensor completion. IEEE Trans. Image Process. **30**, 3434–3449 (2021)
38. Zhang, F.Z.: Quaternions and matrices of quaternions. Linear Algebra Appl. **251**, 21–57 (1997)

39. Wei, M.S., Li, Y., Zhang, F.X., Zhao, J.L.: Quaternion Matrix Computations. Nova Science Publishers (2018)
40. Chen, Y.Y., Xiao, X.L., Zhou, Y.C.: Low-rank quaternion approximation for color image processing. IEEE Trans. Image Process. **29**, 1426–1439 (2019)
41. Yang, L.Q., Miao, J.F., Kou, K.I.: Quaternion-based color image completion via logarithmic approximation. Inf. Sci. **588**, 82–105 (2022)
42. Tang, K.W., Liu, R.S., Su, Z.X., Zhang, J.: Structure-constrained low-rank representation. IEEE Trans. Neural Networks Learn. Syst. **25**, 2167–2179 (2014)
43. Xu, D., Mandic, D.P.: The theory of quaternion matrix derivatives. IEEE Trans. Signal Process. **63**, 1543–1556 (2015)
44. Jia, Z.G., Ng, M.K., Song, G.J.: Lanczos method for large-scale quaternion singular value decomposition. Numer Algorithms. **82**, 699–717 (2019)
45. Yu, Y.B., Zhang, Y.L., Yuan, S.F.: Quaternion-based weighted nuclear norm minimization for color image denoising. Neurocomputing. **332**, 283–297 (2019)
46. Jia, Z.G.: Structure-preserving quaternion toolbox. http://maths.jsnu.edu.cn/_t1395/5134/main.htm