Numerical Algorithms
https://doi.org/10.1007/s11075-024-01923-8

RESEARCH

Efficient quaternion CUR method for low-rank approximation to quaternion matrix

Pengling Wu[1] · Kit Ian Kou[1] · Hongmin Cai[2] · Zhaoyuan Yu[3]

Received: 26 February 2024 / Accepted: 13 August 2024

Abstract

The low-rank quaternion matrix approximation has been successfully applied in many applications involving signal processing and color image processing. However, the cost of quaternion models for generating low-rank quaternion matrix approximation is sometimes considerable due to the computation of the quaternion singular value decomposition (QSVD), which limits their application to real large-scale data. To address this deficiency, an efficient quaternion matrix CUR (QMCUR) method for low-rank approximation is suggested, which provides significant acceleration in color image processing. We first explore the QMCUR approximation method, which uses actual columns and rows of the given quaternion matrix, instead of the costly QSVD. Additionally, two different sampling strategies are used to sample the above-selected columns and rows. Then, the perturbation analysis is performed on the QMCUR approximation of noisy versions of low-rank quaternion matrices. And we also employ the proposed QMCUR method to color image recovery problem. Extensive experiments on both synthetic and real data further reveal the superiority of the proposed algorithm compared with other algorithms for getting low-rank approximation, in terms of both efficiency and accuracy.

Keywords Quaternion matrix · quaternion CUR decomposition · low-rank approximation · color image processing

✉ Kit Ian Kou
kikou@um.edu.mo

Pengling Wu
wupengling@163.com

Hongmin Cai
hmcai@scut.edu.cn

Zhaoyuan Yu
yuzhaoyuan@njnu.edu.cn

[1] Department of Mathematics, Faculty of Science and Technology, University of Macau, 100190 Macau, China

[2] School of Computer Science & Engineering, South China University of Technology, 510006 Guangzhou, China

[3] Department School of Geography, Nanjing Normal University, 210023 Nanjing, China

Published online: 22 August 2024

Mathematics Subject Classification (2010) 65F55

1 Introduction

Quaternion [1] as a mathematical concept was originally introduced by Hamilton in 1843. As an extension of complex numbers, a quaternion number consists of one real part and three imaginary parts. A quaternion matrix is a generalization of a complex matrix in quaternion algebra. By now, quaternions and quaternion matrices have widely used in signal processing [2, 3], machine learning [4, 5], color image processing [6, 7], and other fields [8, 9]. By encoding the red, green, and blue channel pixel values of a color image on the three imaginary parts of quaternion matrix, this method perfectly fits the color image structure and effectively preserves the inter-relationship between the color channels [10].

As an emerging mathematical tool, low-rank quaternion matrix approximation (LRQA) has attracted much attention in the field of color image processing, such as color face recognition [11], color image inpainting [12–15], and color image denoising [16, 17]. For instance, Chen *et al*. [18] proposed the quaternion nuclear norm-based LRQA for color image denoising and inpainting. This method fully utilizes the high correlation among RGB channels by extending low-rank matrix approximation into the quaternion domain. Yu *et al*. [19] further extended the weighted nuclear norm minimization (WNNM) into the quaternion domain and proposed the quaternion-based WNNM method for color image restoration. Notably, the work [20] proposed the robust quaternion matrix completion to minimize the nuclear norm and l_1-norm. This approach provides an exact recovery guarantee under certain conditions and shows superior performance for color image recovery. Moreover, Jia *et al*. [21] introduced the patch group-based nonlocal self-similarity (NSS) prior scheme to learn explicit NSS models, and subsequently offered the NSS-based quaternion matrix completion (QMC) algorithm to reconstruct color images, leading to a promising result. Furthermore, they extended this approach to tensor NSS-based QMC for color videos inpainting. Additionally, in [22], the authors employed quaternion logarithmic norm to achieve a more accurate low-rank approximation. However, a key limitation of these methods for generating low-rank quaternion matrix approximation is that they need to compute the quaternion singular value decomposition (QSVD) in each iteration and suffer from computational deficiency, especially for large-scale data.

In addition to rank minimization, recent studies have utilized quaternion matrix decomposition and randomized techniques to improve the performance of LRQA [13, 23]. For instance, Miao *et al*. [24] suggested the matrix factorization for the target quaternion matrix, followed by three quaternion-based bilinear factor matrix norm factorization methods for low-rank quaternion matrix completion, which can avoid expensive calculations. Liu *et al*. [25] presented a randomized QSVD algorithm for low-rank matrix approximation. The randomized QSVD algorithm reduces the computational cost compared to traditional QSVD for large-scale data. Is there any method with lower computational cost for low-rank approximation of a quaternion matrix?

In recent years, the matrix CUR (MCUR) method [26–28] for fast low-rank approximation of real matrices has been actively investigated because of its ability to efficiently handle large-scale problems. The MCUR method approximates a low-rank matrix by directly sampling a subset of columns and rows from the original matrix and representing it as a product of three small-scale matrices. Owing to the random column/row selection strategy, the MCUR decomposition shows great potential for reducing the computational costs and preserving the properties of the original data matrix compared with other methods [29, 30].

To further enhance the approximation performance and boost the computational efficiency of LRQA, we consider the efficient quaternion matrix CUR (QMCUR) method for low-rank approximation. Precisely, the QMCUR approximation of a quaternion matrix is obtained by utilizing the actual rows and columns of the original quaternion matrix. Additionally, we employ two different sampling strategies to sample the above-selected columns and rows. In summary, the introduction of the QMCUR approximation and the utilization of sampling strategies offer promising avenues for achieving low-rank approximation. The main contributions of this paper are twofold:

- We consider the QMCUR method for low-rank approximation of a quaternion matrix. This approach helps to reduce computational costs and improve the precision of low-rank approximation.
- The perturbation error bound of the proposed approximation method is studied and demonstrated under the spectral norm. We also employ the QMCUR method to color image recovery problem and develop an algorithm to solve it.
- Experimental results for synthetic data and color images demonstrate that the proposed QMCUR approximation method achieves a substantial speed-up while maintaining accuracy compared with other methods.

The rest of the paper is organized as follows. The main notation and preliminary for quaternions are presented in Section 2. Section 3 gives the details of the proposed quaternion CUR approximation method and perturbation error bound, and presents the QMCUR method for color image recovery. We provide numerical experiments in Section 4. Finally, some concluding remarks will be given in Section 5.

2 Preliminaries

2.1 Notation

Throughout this paper, $\mathbb{R}$, $\mathbb{C}$, and $\mathbb{Q}$ respectively denote the set of real numbers, the set of complex numbers, and the set of quaternions. The set of all positive integers is denoted by $\mathbb{N}$, and the symbol $[n]$ represents the set of integers $\{1, \ldots, n\}$ for any $n \in \mathbb{N}$. A scalar, a vector, and a matrix are written as x, $\mathbf{x}$, and $\mathbf{X}$, respectively. A dot (above the variable) is used to denote a quaternion variable $\mathbb{Q}$ ($e.g.$, $\dot{x}$, $\dot{\mathbf{x}}$, and $\dot{\mathbf{X}}$ respectively represent a quaternion scalar, a quaternion vector, and a quaternion matrix). We use $\dot{\mathbf{X}}(I, :)$ and $\dot{\mathbf{X}}(:, J)$ to denote the row and column submatrices with indices I and J, respectively. Here, $(\cdot)^*$, $(\cdot)^\top$, $(\cdot)^H$, and $(\cdot)^\dagger$ represent the conjugate, transpose, conjugate transpose, and Moore-Penrose inverse, respectively.

2.2 Quaternion and quaternion matrices

The set of quaternions $\mathbb{Q}$ is a linear space over $\mathbb{R}$, with an ordered basis of 1, $\boldsymbol{i}$, $\boldsymbol{j}$, and $\boldsymbol{k}$. Here, $\boldsymbol{i}$, $\boldsymbol{j}$, and $\boldsymbol{k}$ are three imaginary units with the multiplication laws: $\boldsymbol{i}^2 = \boldsymbol{j}^2 = \boldsymbol{k}^2 = \boldsymbol{ijk} = -1$, $\boldsymbol{ij} = -\boldsymbol{ji} = \boldsymbol{k}$, $\boldsymbol{jk} = -\boldsymbol{kj} = \boldsymbol{i}$, $\boldsymbol{ki} = -\boldsymbol{ik} = \boldsymbol{j}$. An quaternion number $\dot{x} \in \mathbb{Q}$ is of the form $\dot{x} = x_0 + x_1\boldsymbol{i} + x_2\boldsymbol{j} + x_3\boldsymbol{k} \in \mathbb{Q}$ with $x_0, x_1, x_2, x_3 \in \mathbb{R}$. In particular, $\dot{x}$ is called a pure quaternion if the real component x_0 equals 0. The conjugate and modulus of x are defined as $\dot{x}^* = x_0 - x_1\boldsymbol{i} - x_2\boldsymbol{j} - x_3\boldsymbol{k}$ and $|\dot{x}| = \sqrt{x_0^2 + x_1^2 + x_2^2 + x_3^2}$, respectively.

Analogously, a quaternion matrix $\dot{\mathbf{X}} \in \mathbb{Q}^{m\times n}$ can be written as $\dot{\mathbf{X}} = \mathbf{X}_0 + \mathbf{X}_1\boldsymbol{i} + \mathbf{X}_2\boldsymbol{j} + \mathbf{X}_3\boldsymbol{k}$ with $\mathbf{X}_0, \mathbf{X}_1, \mathbf{X}_2, \mathbf{X}_3 \in \mathbb{R}^{m\times n}$. For given $\dot{\mathbf{X}} = (\dot{x}_{st}) \in \mathbb{Q}^{m\times n}$, the conjugate of $\dot{\mathbf{X}}$ is denoted by $\dot{\mathbf{X}}^* = (\dot{x}_{st}^*) \in \mathbb{Q}^{m\times n}$, the transpose of $\dot{\mathbf{X}}$ is denoted by $\dot{\mathbf{X}}^\top = (\dot{x}_{ts}) \in \mathbb{Q}^{n\times m}$, and the conjugate transpose of $\dot{\mathbf{X}}$ is denoted by $\dot{\mathbf{X}}^H = (\dot{x}_{ts}^*) \in \mathbb{Q}^{n\times m}$. Moreover, the Frobenius norm of $\dot{\mathbf{X}}$ is defined as $\|\dot{\mathbf{X}}\|_F = \sqrt{\mathrm{Tr}(\dot{\mathbf{X}}^H\dot{\mathbf{X}})} = \sqrt{\sum_{s=1}^m \sum_{t=1}^n |\dot{x}_{st}|^2}$, where $\mathrm{Tr}(\cdot)$ is the trace operator, and the spectral norm $\|\dot{\mathbf{X}}\|_2$ is defined as $\|\dot{\mathbf{X}}\|_2 = \max_{\dot{\boldsymbol{a}}\in\mathbb{Q}^n, \|\dot{\boldsymbol{a}}\|_2=1} \|\dot{\mathbf{X}}\dot{\boldsymbol{a}}\|_2$. The following gives some definitions and properties of quaternion matrices.

Definition 1 (**The rank of quaternion matrix**[31]) The maximum number of right (left) linearly independent columns (rows) of a quaternion matrix $\dot{\mathbf{X}} \in \mathbb{Q}^{m\times n}$ is called the rank of $\dot{\mathbf{X}}$.

Definition 2 (**Quaternion singular value decomposition (QSVD)**[31]) Let $\dot{\mathbf{X}} \in \mathbb{Q}^{m\times n}$ be of rank k. There exist unitary quaternion matrices $\dot{\mathbf{W}} \in \mathbb{Q}^{m\times m}$ and $\dot{\mathbf{V}} \in \mathbb{Q}^{n\times n}$ such that $\dot{\mathbf{X}} = \dot{\mathbf{W}}\Sigma\dot{\mathbf{V}}^H$, where $\Sigma = \begin{pmatrix} \mathbf{D}_k & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix} \in \mathbb{R}^{m\times n}$, and $\mathbf{D}_k = \mathrm{diag}(\sigma_1, \ldots, \sigma_k)$ $\in \mathbb{R}^{k\times k}$ is a real diagonal matrix and has k positive entries σ_k $(i = 1, \ldots, k)$ on its diagonal (i.e., positive singular values of $\dot{\mathbf{X}}$). The truncated QSVD of $\mathbf{A}$ with rank k is denoted by $\mathbf{A}_k = \dot{\mathbf{W}}_k\Sigma_k\dot{\mathbf{V}}_k^H$, where $\dot{\mathbf{W}}_k = \dot{\mathbf{W}}(:, 1 : k) \in \mathbb{Q}^{m\times k}$, $\dot{\mathbf{V}}_k = \dot{\mathbf{V}}(:, 1 : k) \in \mathbb{Q}^{n\times k}$, and Σ_k is a $k \times k$ matrix containing the largest k singular values.

Definition 3 (**Cayley-Dickson form**[32]**and Complex adjoint form** [31]) Let $\dot{\mathbf{X}} = \mathbf{X}_0 + \mathbf{X}_1\boldsymbol{i} + \mathbf{X}_2\boldsymbol{j} + \mathbf{X}_3\boldsymbol{k}$, $\dot{\mathbf{X}} \in \mathbb{Q}^{m\times n}$, $\mathbf{X}_0, \mathbf{X}_1, \mathbf{X}_2, \mathbf{X}_3 \in \mathbb{R}^{m\times n}$. Then the Cayley-Dickson form of $\dot{\mathbf{X}}$ is expressed by $\dot{\mathbf{X}} = \mathbf{X}_a + \mathbf{X}_b\boldsymbol{j} \in \mathbb{Q}^{m\times n}$, where $\mathbf{X}_a = \mathbf{X}_0 + \mathbf{X}_1\boldsymbol{i}, \mathbf{X}_b = \mathbf{X}_2 + \mathbf{X}_3\boldsymbol{j}$, and $\mathbf{X}_a, \mathbf{X}_b \in \mathbb{C}^{m\times n}$. The complex adjoint form of $\mathbf{X}$ is formulated as

$$\mathbf{P}_{\dot{\mathbf{X}}} = \begin{bmatrix} \mathbf{X}_a & \mathbf{X}_b \\ -\mathbf{X}_b^* & \mathbf{X}_a^* \end{bmatrix} \in \mathbb{C}^{2m\times 2n}, \tag{1}$$

The relation between the QSVD of quaternion matrix $\dot{\mathbf{X}}$ and the SVD of its equivalent complex matrix $\mathbf{P}_{\dot{\mathbf{X}}} \in \mathbb{C}^{2m\times 2n}$ ($\mathbf{P}_{\dot{\mathbf{X}}} = \mathbf{W}\Sigma'\mathbf{V}^H$) is defined as follows [3, 33, 34]

$$\begin{array}{rcl} \Sigma & = & \mathrm{row}(\mathrm{col}_{odd}(\Sigma')) \in \mathbb{R}^{m\times n}, \\ \dot{\mathbf{W}} & = & \mathrm{col}_{odd}(\mathbf{W_1}) + \mathrm{col}_{odd}(-\mathbf{W_2^*})\boldsymbol{j} \in \mathbb{Q}^{m\times m}, \\ \dot{\mathbf{V}} & = & \mathrm{col}_{odd}(\mathbf{V_1}) + \mathrm{col}_{odd}(-\mathbf{V_2^*})\boldsymbol{j} \in \mathbb{Q}^{n\times n}, \end{array} \tag{2}$$

such that $\dot{\mathbf{X}} = \dot{\mathbf{W}}\Sigma\dot{\mathbf{V}}^H$, where

$$\mathbf{W} = \begin{bmatrix} \mathbf{W_1}t & {}_{m\times 2m} \\ & [\mathbf{W_2}]_{m\times 2m} \end{bmatrix} \in \mathbb{C}^{2m\times 2m}, \quad \mathbf{V} = \begin{bmatrix} \mathbf{V_1}t & {}_{n\times 2n} \\ & [\mathbf{V_2}]_{n\times 2n} \end{bmatrix} \in \mathbb{C}^{2n\times 2n}, \tag{3}$$

and $\mathrm{row}_{odd}(\mathbf{M})$, $\mathrm{col}_{odd}(\mathbf{M})$ extracts the odd rows and odd columns of matrix $\mathbf{M}$, respectively. Consequently, we obtain the QSVD of quaternion matrix $\dot{\mathbf{X}}$ as $\dot{\mathbf{X}} = \dot{\mathbf{W}}\Sigma\dot{\mathbf{V}}^H$.

Based on QSVD $\dot{\mathbf{X}} = \dot{\mathbf{W}}\Sigma\dot{\mathbf{V}}^H$ we can further obtain the Moore-Penrose inverses [35, 36] of quaternion matrix as

$$\dot{\mathbf{X}}^\dagger = \dot{\mathbf{V}}\Sigma^\dagger\dot{\mathbf{W}}^H = \dot{\mathbf{V}}\begin{pmatrix} \mathbf{D}_r^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix}\dot{\mathbf{W}}^H\dot{\mathbf{X}} \in \mathbb{Q}^{n\times m}. \tag{4}$$

More information about quaternion matrices can be found in [3, 31, 35].

3 Proposed approach

In this section, we first present the QMCUR method, which is designed to efficiently compute a low-rank approximation of a large-scale quaternion matrix with low computational costs and comparable accuracy. Then we provide the perturbation estimates for the QMCUR approximation of the noisy version of low-rank matrices. Furthermore, We employ the QMCUR method to color image recovery problem and develop an algorithm to solve it.

3.1 QMCUR-based low-rank approximation method

We now present the QMCUR approximation method. Let $\dot{\mathbf{X}} \in \mathbb{Q}^{m\times n}$ be a low-rank quaternion matrix with a predefined rank k. Consider row indices $I \subseteq [m]$ and column indices $J \subseteq [n]$ satisfying $|I|, |J| \geq k$. Denote a column submatrix $\dot{\mathbf{C}} = \dot{\mathbf{X}}(:, J)$ whose columns span the column space of $\dot{\mathbf{X}}$ and a row submatrix $\dot{\mathbf{R}} = \dot{\mathbf{X}}(I, :)$ whose rows span the row space of $\dot{\mathbf{X}}$. Then the QMCUR approximation of $\dot{\mathbf{X}}$ is a product of the form

$$\dot{\mathbf{X}} \approx \dot{\mathbf{C}}\dot{\mathbf{U}}\dot{\mathbf{R}}, \tag{5}$$

where $\dot{\mathbf{C}} \in \mathbb{Q}^{m\times|J|}$ with $|J|$ columns from the quaternion matrix $\dot{\mathbf{X}}$ and $\dot{\mathbf{R}} \in \mathbb{Q}^{|I|\times n}$ with $|I|$ rows from the quaternion matrix $\dot{\mathbf{X}}$. It's obvious that the core quaternion matrix $\dot{\mathbf{U}} \in \mathbb{Q}^{|J|\times|I|}$ should be computed to yield the smallest error. The optimal choice for the core quaternion matrix $\dot{\mathbf{U}}$ in the least-squares sense is $\dot{\mathbf{U}} = \dot{\mathbf{C}}^\dagger\dot{\mathbf{X}}\dot{\mathbf{R}}^\dagger$ [37] because

$$\dot{\mathbf{C}}^\dagger\dot{\mathbf{X}}\dot{\mathbf{R}}^\dagger = \underset{\dot{\mathbf{U}}\in\mathbb{Q}^{|J|\times|I|}}{\arg\min} \ \|\dot{\mathbf{X}} - \mathbf{CUR}\|_F^2. \tag{6}$$

Note that one may replace $\dot{\mathbf{C}}^\dagger$ in (5) by $\dot{\mathbf{C}}^\dagger\dot{\mathbf{X}}\dot{\mathbf{R}}^\dagger$ to improve the approximation, resulting in the QMCUR approximation $\dot{\mathbf{C}}\dot{\mathbf{C}}^\dagger\dot{\mathbf{X}}\dot{\mathbf{R}}^\dagger\dot{\mathbf{R}}$. Algorithm 1 summarizes the overall process of the QMCUR approximation.

Algorithm 1 Main steps of QMCUR approximation

```
Input: $\dot{\mathbf{X}} \in \mathbb{Q}^{m\times n}$: the quaternion matrix; $k$: predefined rank; $|I|, |J|$: number of rows and columns to
    sample; $\{p_j\}, \{q_i\}$: sampling probability distribution, $j = 1, \dots, n$, and $i = 1, \dots, m$.
1: Draw sampling column indices $J \subseteq [n]$ based on the sampling probability $p_j$ and construct $\dot{\mathbf{C}} \in \mathbb{Q}^{m\times|J|}$;
2: Draw sampling row indices $I \subseteq [m]$ based on the sampling probability $q_i$ and construct $\dot{\mathbf{R}} \in \mathbb{Q}^{|I|\times n}$;
3: Compute $\dot{\mathbf{C}}^\dagger \in \mathbb{Q}^{|J|\times m}$ and $\dot{\mathbf{R}}^\dagger \in \mathbb{Q}^{n\times|I|}$ by (7);
4: Compute $\dot{\mathbf{U}}$: $\dot{\mathbf{U}} = \dot{\mathbf{C}}^\dagger\dot{\mathbf{X}}\dot{\mathbf{R}}^\dagger \in \mathbb{Q}^{|J|\times|I|}$;
Output: $\dot{\mathbf{C}}, \dot{\mathbf{U}}, \dot{\mathbf{R}}$ such that $\dot{\mathbf{X}} \approx \dot{\mathbf{C}}\dot{\mathbf{U}}\dot{\mathbf{R}}$.
```

In Algorithm 1, the matrix $\dot{\mathbf{U}} \in \mathbb{Q}^{|J|\times|I|}$ is computed by $\dot{\mathbf{U}} = \dot{\mathbf{C}}^\dagger\dot{\mathbf{X}}\dot{\mathbf{R}}^\dagger$. Here, the computation of $\dot{\mathbf{C}}^\dagger$ relies on its QSVD (see (4) for more details) [35, 36]. If the QSVD of $\dot{\mathbf{C}} \in \mathbb{Q}^{m\times|J|}$ is $\dot{\mathbf{C}} = \dot{\mathbf{W}}\Sigma\dot{\mathbf{V}}^H$, then the Moore-Penrose inverse of $\dot{\mathbf{C}}$ is given by

$$\dot{\mathbf{C}}^\dagger = \dot{\mathbf{V}}\Sigma^\dagger\dot{\mathbf{W}}^H \in \mathbb{Q}^{|J|\times m}. \tag{7}$$

And the calculation of $\dot{\mathbf{R}}^\dagger \in \mathbb{Q}^{n\times|I|}$ is similar to that of $\dot{\mathbf{C}}^\dagger$. Additionally, we set the size of $\dot{\mathbf{C}}$ to be $m \times k\log k$ and $\dot{\mathbf{R}}$ to be $k\log k \times n$ for the QMCUR approximation method. The indices used to determine $\dot{\mathbf{C}}$ and $\dot{\mathbf{R}}$ are sampled using two different strategies [38, 39]. In the first strategy, the sampling probabilities p_j^{col} and q_i^{row} for each column j and row i of the quaternion matrix $\dot{\mathbf{X}}$ are based on the Euclidean norm of the columns and rows, which are respectively defined as

$$p_j^{\text{col}} := \frac{\|\dot{\mathbf{X}}(:,j)\|_2^2}{\|\dot{\mathbf{X}}\|_F^2}, \quad j = 1, 2, \dots, n \ \text{ and } \ q_i^{\text{row}} := \frac{\|\dot{\mathbf{X}}(i,:)\|_2^2}{\|\dot{\mathbf{X}}\|_F^2}, \quad i = 1, 2, \dots, m. \tag{8}$$

This strategy is referred to as QMCUR_length. In the second strategy, the sampling probabilities p_j^{unif} and q_i^{unif} are respectively defined as

$$p_j^{\text{unif}} := \frac{1}{n}, \quad j = 1, 2, \dots, n \ \text{ and } \ q_i^{\text{unif}} := \frac{1}{m}, \quad i = 1, 2, \dots, m. \tag{9}$$

This approach is referred to as QMCUR_uniform.

3.2 Computational complexity

Now, we discuss the computational complexity of the proposed algorithm. Let $\dot{\mathbf{X}} \in \mathbb{H}^{m\times n}$ be a given quaternion matrix. According to (2), the computation of the QSVD for a $m \times n$ quaternion matrix is equivalent to the computation of the SVD of a $2m \times 2n$ complex matrix (its complex adjoint). Thus, the computational complexity of QSVD for a $m \times n$ quaternion matrix is $\mathcal{O}(\min(8mn^2, 8nm^2))$. Then, computing $\dot{\mathbf{C}}^\dagger$ requires performing QSVD on matrices with size of $m \times |J|$ and multiplication, which cost $\mathcal{O}((9|J|^2m+|J|m^2)$. Similarly, computing $\dot{\mathbf{R}}^\dagger$ requires performing QSVD on matrices with size of $|I|\times n$ and multiplication, which cost $\mathcal{O}(9|I|^2n+|I|n^2)$. Computing $\dot{\mathbf{U}}$ and $\dot{\mathbf{X}}$ only involves multiplication costing $\mathcal{O}((|I|+|J|)mn+|I||J|(m+n))$. In summary, the computational cost of the developed algorithm is $\mathcal{O}((8|J|^2m + 8|I|^2n + |J|m^2 + |I|n^2 + (|J| + |I|)mn)$.

3.3 Perturbation estimates for CUR approximation

In practical applications, quaternion matrix data is often perturbed, such as noise in pictures, which may cause huge errors. In this section, we present the perturbation analysis suggested by the QMCUR approximation described above. Our main task will be to consider matrices in the form of

$$\tilde{\dot{\mathbf{X}}} = \dot{\mathbf{X}} + \dot{\mathbf{E}}, \tag{10}$$

where $\dot{\mathbf{X}} \in \mathbb{Q}^{m\times n}$ with rank k, and $\dot{\mathbf{E}} \in \mathbb{Q}^{m\times n}$ is an arbitrary noise quaternion matrix drawn from a certain distribution.

Before analyzing the perturbation, we will first introduce some notations. If $\tilde{\dot{\mathbf{X}}} = \dot{\mathbf{X}} + \dot{\mathbf{E}}$, and we consider $\tilde{\dot{\mathbf{C}}} = \tilde{\dot{\mathbf{X}}}(:, J) \in \mathbb{Q}^{m\times|J|}$, $\tilde{\dot{\mathbf{R}}} = \tilde{\dot{\mathbf{X}}}(I, :) \in \mathbb{Q}^{|I|\times n}$, and $\tilde{\dot{\mathbf{U}}} = \tilde{\dot{\mathbf{C}}}^\dagger \tilde{\dot{\mathbf{X}}} \tilde{\dot{\mathbf{R}}}^\dagger \in \mathbb{Q}^{|J|\times|I|}$ for selected index sets I and J. Then we write

$$\begin{cases} \tilde{\dot{\mathbf{C}}} = \dot{\mathbf{X}}(:, J) + \dot{\mathbf{E}}(:, J) = \dot{\mathbf{C}} + \dot{\mathbf{E}}(:, J), \\ \tilde{\dot{\mathbf{R}}} = \dot{\mathbf{X}}(I, :) + \dot{\mathbf{E}}(I, :) = \dot{\mathbf{R}} + \dot{\mathbf{E}}(I, :), \end{cases} \tag{11}$$

where $\dot{\mathbf{C}} = \dot{\mathbf{X}}(:, J) \in \mathbb{Q}^{m\times|J|}$, $\dot{\mathbf{R}} = \dot{\mathbf{X}}(I, :) \in \mathbb{Q}^{|I|\times n}$, and $\dot{\mathbf{U}} = \dot{\mathbf{C}}^\dagger \dot{\mathbf{X}} \dot{\mathbf{R}}^\dagger \in \mathbb{Q}^{|J|\times|I|}$. For ease of notation, we will use the conventions that $\dot{\mathbf{E}}_J = \dot{\mathbf{E}}(:, J) \in \mathbb{Q}^{m\times|J|}$ and $\dot{\mathbf{E}}_I = \dot{\mathbf{E}}(I, :) \in \mathbb{Q}^{|I|\times n}$. The following theorem provides a perturbation estimate for the QMCUR approximation.

Theorem 1 Let $\tilde{\dot{\mathbf{X}}} = \dot{\mathbf{X}} + \dot{\mathbf{E}}$ for a fixed but arbitrary $\dot{\mathbf{E}} \in \mathbb{Q}^{m\times n}$. Using the notation in (11), then the following holds:

$$\|\dot{\mathbf{X}} - \tilde{\dot{\mathbf{C}}}\tilde{\dot{\mathbf{C}}}^\dagger \tilde{\dot{\mathbf{X}}}\tilde{\dot{\mathbf{R}}}^\dagger \tilde{\dot{\mathbf{R}}}\| \leq \|\dot{\mathbf{E}}_I\|\|\dot{\mathbf{X}}\dot{\mathbf{R}}^\dagger\| + \|\dot{\mathbf{E}}_J\|\|\dot{\mathbf{C}}^\dagger\dot{\mathbf{X}}\| + 3\|\dot{\mathbf{E}}\|. \tag{12}$$

Furthermore, suppose that $\dot{\mathbf{X}} \in \mathbb{Q}^{m\times n}$ has rank k and its truncated QSVD is $\dot{\mathbf{X}} = \dot{\mathbf{W}}_k \Sigma_k \dot{\mathbf{V}}_k^H$, where $\dot{\mathbf{W}}_k \in \mathbb{Q}^{m\times k}$, $\Sigma_k \in \mathbb{R}^{k\times k}$, and $\dot{\mathbf{V}}_k \in \mathbb{Q}^{n\times k}$. Then we have

$$\|\dot{\mathbf{X}} - \tilde{\dot{\mathbf{C}}}\tilde{\dot{\mathbf{C}}}^{\dagger}\tilde{\dot{\mathbf{X}}}\tilde{\dot{\mathbf{R}}}^{\dagger}\tilde{\dot{\mathbf{R}}}\| \le \|\dot{\mathbf{E}}\|(\|\dot{\mathbf{W}}_{k,I}^{\dagger}\| + \|\dot{\mathbf{V}}_{k,J}^{\dagger}\| + 3), \tag{13}$$

where $\dot{\mathbf{W}}_{k,I} = \dot{\mathbf{W}}_k(I,:) \in \mathbb{Q}^{|I|\times k}$ and $\dot{\mathbf{V}}_{k,J} = \dot{\mathbf{V}}_k(J,:) \in \mathbb{Q}^{|J|\times k}$ for index sets I, J.

In a word, Theorem 1 shows that the error estimation in the QMCUR method is controlled by the pseudoinverses of the submatrices of the orthogonal singular vectors and is linear in the norm of the noise $\dot{\mathbf{E}}$. To establish Theorem 1, we will introduce the following lemmas.

Lemma 1 Suppose that $\dot{\mathbf{X}} \in \mathbb{Q}^{m\times n}$ has rank k with the QMCUR approximation: $\dot{\mathbf{X}} \approx \mathbf{CUR}$, where $\dot{\mathbf{C}} = \dot{\mathbf{X}}(:,J)$ with selected column indices J, $\dot{\mathbf{R}} = \dot{\mathbf{X}}(I,:)$ with selected row indices I, and $\dot{\mathbf{U}} = \dot{\mathbf{C}}^{\dagger}\dot{\mathbf{X}}\dot{\mathbf{R}}^{\dagger}$. Then we have

$$\operatorname{rank}(\dot{\mathbf{C}}) = \operatorname{rank}(\dot{\mathbf{R}}) = \operatorname{rank}(\dot{\mathbf{X}}). \tag{14}$$

Proof Recall that $\dot{\mathbf{C}} = \dot{\mathbf{X}}(:,J)$. Thus, $\operatorname{span}(\dot{\mathbf{C}}) \subset \operatorname{span}(\dot{\mathbf{X}})$. On account of the fact that $\dot{\mathbf{X}} = \dot{\mathbf{C}}\dot{\mathbf{C}}^{\dagger}\dot{\mathbf{X}}\dot{\mathbf{R}}^{\dagger}\dot{\mathbf{R}}$, we have $\operatorname{span}(\dot{\mathbf{X}}) \subset \operatorname{span}(\dot{\mathbf{C}})$. A similar argument shows that $\operatorname{span}(\dot{\mathbf{R}}^H) \subset \operatorname{span}(\dot{\mathbf{X}}^H)$. The conclusion follows. □

Lemma 2 The following hold:

$$\|\dot{\mathbf{X}} - \tilde{\dot{\mathbf{C}}}\tilde{\dot{\mathbf{C}}}^{\dagger}\tilde{\dot{\mathbf{X}}}\| \le \|\dot{\mathbf{E}}_J\|\|\dot{\mathbf{C}}^{\dagger}\dot{\mathbf{X}}\| + \|\dot{\mathbf{E}}\|,\ \|\dot{\mathbf{X}} - \tilde{\dot{\mathbf{X}}}\tilde{\dot{\mathbf{R}}}^{\dagger}\tilde{\dot{\mathbf{R}}}\| \le \|\dot{\mathbf{E}}_I\|\|\dot{\mathbf{X}}\dot{\mathbf{R}}^{\dagger}\| + \|\dot{\mathbf{E}}\|. \tag{15}$$

Proof Note that

$$\begin{aligned}\|(\mathbf{I} - \tilde{\dot{\mathbf{C}}}\tilde{\dot{\mathbf{C}}}^{\dagger})\dot{\mathbf{C}}\| &= \|(\mathbf{I} - \tilde{\dot{\mathbf{C}}}\tilde{\dot{\mathbf{C}}}^{\dagger})\tilde{\dot{\mathbf{C}}} - (\mathbf{I} - \tilde{\dot{\mathbf{C}}}\tilde{\dot{\mathbf{C}}}^{\dagger})\dot{\mathbf{E}}_J\| \\ &\le \|(\mathbf{I} - \tilde{\dot{\mathbf{C}}}\tilde{\dot{\mathbf{C}}}^{\dagger})\tilde{\dot{\mathbf{C}}}\| + \|(\mathbf{I} - \tilde{\dot{\mathbf{C}}}\tilde{\dot{\mathbf{C}}}^{\dagger})\dot{\mathbf{E}}_J\| \\ &\le \|\dot{\mathbf{E}}_J\|.\end{aligned} \tag{16}$$

The final inequality holds because the first norm term is 0 by identity of the Moore-Penrose pseudoinverse and $\|\mathbf{I} - \tilde{\dot{\mathbf{C}}}\tilde{\dot{\mathbf{C}}}^{\dagger}\|_2 \le 1$ as this is an orthogonal projection operator. Since $\operatorname{rank}(\dot{\mathbf{C}}) = \operatorname{rank}(\dot{\mathbf{X}})$, we have $\dot{\mathbf{X}} = \dot{\mathbf{C}}\dot{\mathbf{C}}^{\dagger}\dot{\mathbf{X}}$. Then

$$\begin{aligned}\|\dot{\mathbf{X}} - \tilde{\dot{\mathbf{C}}}\tilde{\dot{\mathbf{C}}}^{\dagger}\tilde{\dot{\mathbf{X}}}\| &\le \|(\mathbf{I} - \tilde{\dot{\mathbf{C}}}\tilde{\dot{\mathbf{C}}}^{\dagger})\dot{\mathbf{X}}\| + \|\dot{\mathbf{E}}\| \\ &= \|(\mathbf{I} - \tilde{\dot{\mathbf{C}}}\tilde{\dot{\mathbf{C}}}^{\dagger})\dot{\mathbf{C}}\dot{\mathbf{C}}^{\dagger}\dot{\mathbf{X}}\| + \|\dot{\mathbf{E}}\| \\ &\le \|\dot{\mathbf{E}}_J\|\|\dot{\mathbf{C}}^{\dagger}\dot{\mathbf{X}}\| + \|\dot{\mathbf{E}}\|.\end{aligned} \tag{17}$$

The second inequality is derived by mimicking the above argument. The conclusion follows. □

Lemma 3 Suppose $\dot{\mathbf{X}}$, $\dot{\mathbf{C}}$, $\dot{\mathbf{U}}$, and $\dot{\mathbf{R}}$ are as in (5) such that $\dot{\mathbf{X}} = \dot{\mathbf{C}}\dot{\mathbf{U}}\dot{\mathbf{R}}$, and suppose that $\text{rank}(\dot{\mathbf{X}}) = k$. Let $\dot{\mathbf{X}} = \dot{\mathbf{W}}_k \Sigma_k \dot{\mathbf{V}}_k^H$ be the truncated QSVD of $\dot{\mathbf{X}}$. Then the following hold:

$$\|\dot{\mathbf{X}}\dot{\mathbf{R}}^\dagger\| = \|\dot{\mathbf{W}}_{k,I}^\dagger\|, \ \ \|\dot{\mathbf{C}}^\dagger\dot{\mathbf{X}}\| = \|(\dot{\mathbf{V}}_{k,J}^H)^\dagger\|, \tag{18}$$

where $\dot{\mathbf{W}}_{k,I} = \dot{\mathbf{W}}_k(I, :) \in \mathbb{Q}^{|I|\times k}$ and $\dot{\mathbf{V}}_{k,J} = \dot{\mathbf{V}}_k(J, :) \in \mathbb{Q}^{|J|\times k}$ for selected index sets I, J.

Proof Begin with the fact that $\dot{\mathbf{R}} = \dot{\mathbf{W}}_k(I, :)\Sigma_k\dot{\mathbf{V}}_k^H = \dot{\mathbf{W}}_{k,I}\Sigma_k\dot{\mathbf{V}}_k^H$. Then we have

$$\dot{\mathbf{X}}\dot{\mathbf{R}}^\dagger = \dot{\mathbf{W}}_k\Sigma_k\dot{\mathbf{V}}_k^H(\dot{\mathbf{W}}_{k,I}\Sigma_k\dot{\mathbf{V}}_k^H)^\dagger. \tag{19}$$

Notice that $\dot{\mathbf{W}}_{k,I}$ has full column rank, Σ_k is a $k \times k$ matrix with full rank, and $\dot{\mathbf{V}}_k^H$ has orthogonal rows (i.e., $(\dot{\mathbf{V}}_k^H)^\dagger = \dot{\mathbf{V}}_k$). In this case,

$$(\dot{\mathbf{W}}_{k,I}\Sigma_k\dot{\mathbf{V}}_k^H)^\dagger = (\dot{\mathbf{V}}_k^H)^\dagger\Sigma_k^{-1}\dot{\mathbf{W}}_{k,I}^\dagger = \dot{\mathbf{V}}_k\Sigma_k^{-1}\dot{\mathbf{W}}_{k,I}^\dagger. \tag{20}$$

Consequently, the following holds:

$$\begin{aligned}\|\dot{\mathbf{X}}\dot{\mathbf{R}}^\dagger\| &= \|\dot{\mathbf{W}}_k\Sigma_k\dot{\mathbf{V}}_k^H\dot{\mathbf{V}}_k\Sigma_k^{-1}\dot{\mathbf{W}}_{k,I}^\dagger\| \\ &= \|\Sigma_k\dot{\mathbf{V}}_k^H\dot{\mathbf{V}}_k\Sigma_k^{-1}\dot{\mathbf{W}}_{k,I}^\dagger\| \\ &= \|\Sigma_k\Sigma_k^{-1}\dot{\mathbf{W}}_{k,I}^\dagger\| \\ &= \|\dot{\mathbf{W}}_{k,I}^\dagger\|.\end{aligned} \tag{21}$$

The second equality above follows from the unitary invariance of the norm [40]. Similarly, we have

$$\|\dot{\mathbf{C}}^\dagger\dot{\mathbf{X}}\| = \|(\dot{\mathbf{V}}_{k,J}^H)^\dagger\|, \tag{22}$$

whereupon the conclusion follows from the fact that $\|(\dot{\mathbf{V}}_{k,J}^H)^\dagger = (\dot{\mathbf{V}}_{k,J}^\dagger)^H$ which has the same norm as $\dot{\mathbf{V}}_{k,J}^\dagger$. The conclusion follows. □

Proof of Theorem 1 First note that

$$\begin{aligned}\|\dot{\mathbf{X}} - \tilde{\dot{\mathbf{C}}}\tilde{\dot{\mathbf{C}}}^\dagger\tilde{\dot{\mathbf{X}}}\tilde{\dot{\mathbf{R}}}^\dagger\tilde{\dot{\mathbf{R}}}\| &= \|\dot{\mathbf{X}} - \tilde{\dot{\mathbf{C}}}\tilde{\dot{\mathbf{C}}}^\dagger\tilde{\dot{\mathbf{X}}} + \tilde{\dot{\mathbf{C}}}\tilde{\dot{\mathbf{C}}}^\dagger\tilde{\dot{\mathbf{X}}}(\mathbf{I} - \dot{\mathbf{R}}^\dagger\dot{\mathbf{R}})\| \\ &\le \|\dot{\mathbf{X}} - \tilde{\dot{\mathbf{C}}}\tilde{\dot{\mathbf{C}}}^\dagger\tilde{\dot{\mathbf{X}}}\| + \|\tilde{\dot{\mathbf{C}}}\tilde{\dot{\mathbf{C}}}^\dagger\|\|\tilde{\dot{\mathbf{X}}}(\mathbf{I} - \dot{\mathbf{R}}^\dagger\dot{\mathbf{R}})\| \\ &\le \|\dot{\mathbf{X}} - \tilde{\dot{\mathbf{C}}}\tilde{\dot{\mathbf{C}}}^\dagger\tilde{\dot{\mathbf{X}}}\| + \|\tilde{\dot{\mathbf{X}}}(\mathbf{I} - \dot{\mathbf{R}}^\dagger\dot{\mathbf{R}})\|.\end{aligned} \tag{23}$$

The second inequality follows from the fact that $\|\dot{\mathbf{C}}\dot{\mathbf{C}}^\dagger\|_F = 1$. Next, using the formula $\tilde{\dot{\mathbf{X}}} = \dot{\mathbf{X}} + \dot{\mathbf{E}}$, we have $\|\tilde{\dot{\mathbf{X}}}(\mathbf{I} - \tilde{\dot{\mathbf{R}}}^\dagger\tilde{\dot{\mathbf{R}}})\| \le \|\dot{\mathbf{X}}(\mathbf{I} - \tilde{\dot{\mathbf{R}}}^\dagger\tilde{\dot{\mathbf{R}}})\| + \|\dot{\mathbf{E}}\|$ since $\|\mathbf{I} - \tilde{\dot{\mathbf{R}}}^\dagger\tilde{\dot{\mathbf{R}}}\|_2 \le 1$. Then applying Lemma 2, we obtain

$$\begin{aligned}\|\dot{\mathbf{X}} - \tilde{\dot{\mathbf{C}}}\tilde{\dot{\mathbf{C}}}^\dagger\tilde{\dot{\mathbf{X}}}\tilde{\dot{\mathbf{R}}}^\dagger\tilde{\dot{\mathbf{R}}}\| &\le \|\dot{\mathbf{X}} - \tilde{\dot{\mathbf{C}}}\tilde{\dot{\mathbf{C}}}^\dagger\tilde{\dot{\mathbf{X}}}\| + \|\dot{\mathbf{X}}(\mathbf{I} - \tilde{\dot{\mathbf{R}}}^\dagger\tilde{\dot{\mathbf{R}}})\| + \|\dot{\mathbf{E}}\| \\ &\le \|\dot{\mathbf{E}}_I\|\|\dot{\mathbf{X}}\dot{\mathbf{R}}^\dagger\| + \|\dot{\mathbf{E}}_J\|\|\dot{\mathbf{C}}^\dagger\dot{\mathbf{X}}\| + 3\|\dot{\mathbf{E}}\|.\end{aligned} \tag{24}$$

Moreover, from Lemma 3, it follows that

$$\|\dot{\mathbf{X}} - \tilde{\dot{\mathbf{C}}}\tilde{\dot{\mathbf{C}}}^{\dagger}\tilde{\dot{\mathbf{X}}}\tilde{\dot{\mathbf{R}}}^{\dagger}\tilde{\dot{\mathbf{R}}}\| \leq \|\dot{\mathbf{E}}\|(\|\dot{\mathbf{W}}_{k,I}^{\dagger}\| + \|\dot{\mathbf{V}}_{k,J}^{\dagger}\| + 3). \tag{25}$$

3.4 QMCUR approximation for color image recovery

In this section, we apply the QMCUR approximation method to recover color images. The problem of recovering color images can be formulated as a quaternion matrix completion problem, aiming to find its missing entries through the following optimization. The mathematical model is

$$\min_{\dot{\mathbf{X}}} \ \mathrm{rank}(\dot{\mathbf{X}}) \ \text{ s.t. } \ P_{\Omega}(\dot{\mathbf{X}}) = P_{\Omega}(\dot{\mathbf{Y}}), \tag{26}$$

where rank$(\cdot)$ denotes the rank function, $\dot{\mathbf{X}} \in \mathbb{Q}^{m\times n}$ and $\dot{\mathbf{Y}} \in \mathbb{Q}^{m\times n}$ represent the recovered and observed quaternion matrices, respectively. Ω represents the set of observed elements, and $P_{\Omega}(\dot{\mathbf{X}})$ is the projection operator that keeps entries in Ω and zeros out others. However, directly solving the problem (26) is difficult as the rank minimization problem is known as NP-hard [41].

The quaternion nuclear norm minimization (QNNM) [18] is a popular convex surrogate of non-convex rank function, and its minimization method is as following

$$\min_{\dot{\mathbf{X}}} \ \|\dot{\mathbf{X}}\|_{*} \ \text{ s.t. } \ P_{\Omega}(\dot{\mathbf{X}}) = P_{\Omega}(\dot{\mathbf{Y}}), \tag{27}$$

However, the computational of QNNM can be quite expensive for large-scale data, due to calculating the QSVD of quaternion matrices. Therefore, if the rank of the matrix $\dot{\mathbf{X}}$ is known [42–44], the QNNM method can be more effectively replaced by the concept of quaternion decomposition. In this case, the problem (26) is presented as follows

$$\min_{\dot{\mathbf{X}}} \ \|P_{\Omega}(\dot{\mathbf{X}}) - P_{\Omega}(\dot{\mathbf{Y}})\|_F^2 \ \ \text{ s.t. } \ \mathrm{rank}(\dot{\mathbf{X}}) = k, \tag{28}$$

Intuitively, this problem seeks the matrix $\dot{\mathbf{X}}$ of rank k that best fits the given data $\dot{\mathbf{Y}}$.

Similar to the Algorithm framework in [43], the optimization problem (28) can be expressed as follows

$$\min_{\dot{\mathbf{X}}} \ \|\dot{\mathbf{X}} - \dot{\mathbf{M}}\|_F^2 \ \ \text{ s.t. } \ \mathrm{rank}(\dot{\mathbf{X}}) = k, \ \ P_{\Omega}(\dot{\mathbf{X}}) = P_{\Omega}(\dot{\mathbf{Y}}). \tag{29}$$

where $\dot{\mathbf{M}} \in \mathbb{Q}^{m\times n}$ is the auxiliary tensor variable. Therefore, we can solve the optimization problem (28) alternatively over the variables $\dot{\mathbf{X}}$ and $\dot{\mathbf{M}}$. The solution to the minimization problem (28) can be approximated using the following iterative procedures:

$$\dot{\mathbf{M}}^{t+1} \leftarrow \mathcal{L}(\dot{\mathbf{X}}^{t}), \tag{30}$$

$$\dot{\mathbf{X}}^{t+1} \leftarrow \Omega \circledast \dot{\mathbf{Y}} + (\mathbf{1} - \Omega) \circledast \dot{\mathbf{M}}^{t+1}, \tag{31}$$

where t represents the iteration number, $\mathcal{L}$ is the an operator used to calculate a low-rank quaternion matrix approximation of the quaternion matrix $\dot{\mathbf{X}}$, which can be achieved by Algorithm 1, and $\mathbf{1}$ is a matrix in which all components are equal to one. The algorithm mainly consists of two steps: low-rank quaternion matrix approximation (30) and masking computation (31). We summarize the proposed method in Algorithm 2. It starts with the initial incomplete data $\dot{\mathbf{X}}^0$ and iteratively enhances the approximate solution until a stopping criterion is satisfied or the maximum number of iterations is reached.

Algorithm 2 QMCUR approximation for color image recovery

```
Input: An incomplete quaternion matrix $\dot{\mathbf{Y}} \in \mathbb{Q}^{m\times n}$, observation locations $\Omega$, rows and columns (i.e., $I$
  and $J$) indices that define $\dot{\mathbf{R}} \in \mathbb{Q}^{|I|\times n}$ and $\dot{\mathbf{C}} \in \mathbb{Q}^{m\times|J|}$ respectively, and maximal iteration $t_{max}$=200.
Output: The recovered quaternion matrix $\dot{\mathbf{X}}^*$.
$\dot{\mathbf{X}}^0 = \dot{\mathbf{Y}} \in \mathbb{Q}^{m\times n}$ is the observed quaternion matrix with missing pixels.
$\dot{\mathbf{M}}^0 \in \mathbb{Q}^{m\times n}$ is a zero quaternion matrix.
For $t = 0, 1, 2, \ldots$ do
    $\dot{\mathbf{M}}^{t+1} \leftarrow$ compute QMCUR approximation of the data matrix $\dot{\mathbf{X}}^t$ using Algorithm 1.
    $\dot{\mathbf{X}}^{t+1} = (\dot{\mathbf{M}}^{t+1})_{\Omega^c} + \dot{\mathbf{Y}}_\Omega$, where $\Omega^c$ denotes the complementary set of $\Omega$.
    If $\|\dot{\mathbf{X}}^{t+1} - \dot{\mathbf{X}}^t\|_F / \|\dot{\mathbf{X}}^t\|_F \le 10^{-4}$, or $t \ge t_{max}$ then
        $\dot{\mathbf{X}}^* = \dot{\mathbf{X}}^{t+1}$ and break,
    end
end
```

4 Numerical experiments

In this section, we conduct numerical experiments to assess the accuracy and computation time of the proposed QMCUR method for low-rank quaternion matrix approximation.

Example 4.1 In this experiment, we demonstrate the error estimation in Theorem 1 using simulation data. We randomly generate a quaternion matrix $\dot{\mathbf{X}}$ as a product $\dot{\mathbf{X}} = \dot{\mathbf{W}}_k \Sigma_k \dot{\mathbf{V}}_k^H \in \mathbb{Q}^{m\times m}$, where $\dot{\mathbf{W}}_k \in \mathbb{Q}^{m\times k}$, $\dot{\mathbf{V}}_k \in \mathbb{Q}^{m\times k}$ are unitary matrices, and Σ_k is a diagonal $k \times k$ matrix with positive diagonal elements. In addition, the random noise quaternion matrix is given by $\dot{\mathbf{E}} = \mathbf{E}_0 + \mathbf{E}_1 \boldsymbol{i} + \mathbf{E}_2 \boldsymbol{j} + \mathbf{E}_3 \boldsymbol{k} \in \mathbb{Q}^{m\times n}$, where the entries of $\mathbf{E}_t \, (t = 0, 1, 2, 3)$ are i.i.d. Gaussian variables with mean zero and variance σ. Then we obtain the perturbed quaternion matrix

$$\tilde{\dot{\mathbf{X}}} = \dot{\mathbf{X}} + \dot{\mathbf{E}}. \tag{32}$$

In this experiment, we assume that $m = 500$ and $k = 50$. By applying Algorithm 1 to compute the rank-k approximation of $\tilde{\dot{\mathbf{X}}}$ with $\sigma = 10^{-1}, 10^{-2}, \ldots, 10^{-6}$, Fig. 1 shows the results. As observed, the error $\|\dot{\mathbf{X}} - \tilde{\dot{\mathbf{C}}}\tilde{\dot{\mathbf{U}}}\tilde{\dot{\mathbf{R}}}\|_2$ increases linearly as $\|\dot{\mathbf{E}}\|_2$ increases, which confirms the conclusion of Theorem 1.

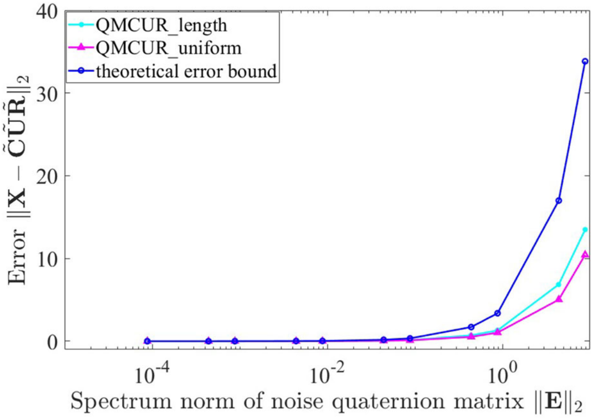

Fig. 1 $\|\dot{\mathbf{X}}-\tilde{\dot{\mathbf{C}}}\tilde{\dot{\mathbf{U}}}\tilde{\dot{\mathbf{R}}}\|_2$ vs. $\|\dot{\mathbf{E}}\|_2$

Example 4.2 In this experiment, we test the proposed QMCUR method on simulated quaternion data and compare their performance with that of QSVD [33], qSVD4 [45], lansvdQ [46], and CSP-QM [7]. We still consider the perturbed quaternion matrix $\tilde{\dot{\mathbf{X}}}$ in Example 4.1. Figure 2 displays the relative error and running time computed by different methods with the varying size m when k =10. This error is measured by

$$\frac{\|\dot{\mathbf{M}}-\dot{\mathbf{X}}\|_F}{\|\dot{\mathbf{X}}\|_F}, \tag{33}$$

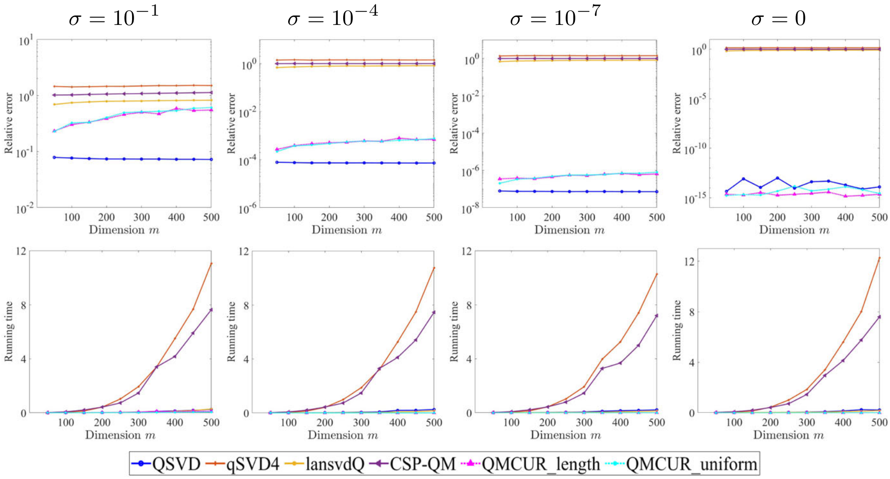

Fig. 2 Comparison of low-rank quaternion matrix approximation methods under different noise levels σ. Rank k=10 is used in all tests and m varies from 50 to 500. Top row: relative approximation errors vs. quaternion matrix dimensions. Bottom row: running time vs. quaternion matrix dimension

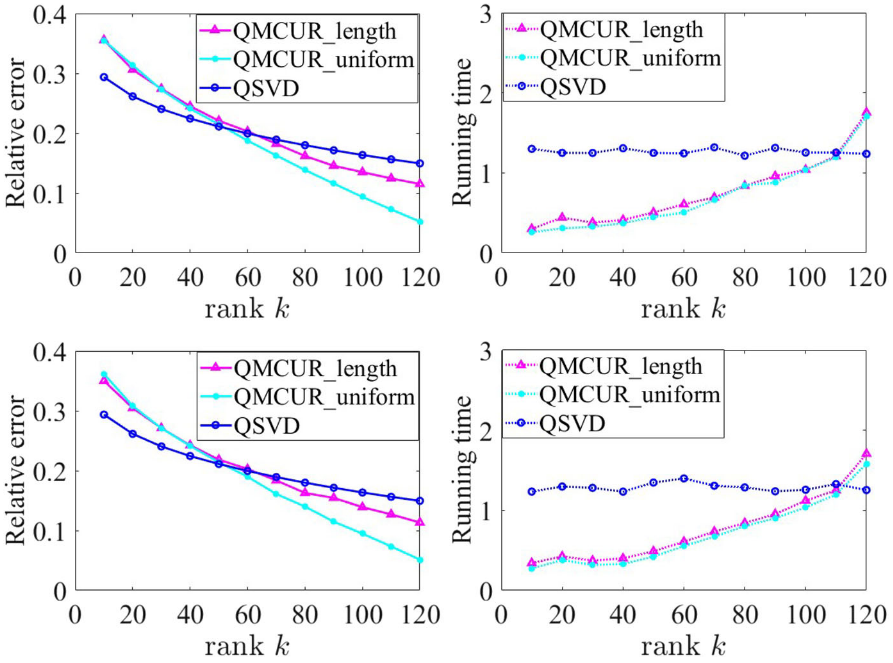

Fig. 3 Numerical simulation results of different methods with different sampling strategies to color images with different rank k. From top to bottom: Image01 and Image02, respectively

where $\dot{\mathbf{X}}$ denotes the quaternion matrix that represents the clean data and $\dot{\mathbf{M}}$ is obtained by all methods.

As observed, QMCUR_length and QMCUR_uniform methods in Algorithm 1 are significantly faster than QSVD when the data size (i.e., m) is sufficiently large. In particular, the errors for QMCUR_uniform are smaller than those obtained by the QSVD algorithm in the noiseless case (i.e., $\sigma = 0$). This is mainly because only the QSVD of the smaller sizes $\dot{\mathbf{C}}$ and $\dot{\mathbf{R}}$ need to be computed, not the QSVD of the quaternion matrix $\dot{\mathbf{X}}$. The proposed methods require less computation time and give more accurate approximations than the qSVD4, lansvdQ, and CSP-QM methods. This observation verifies that the proposed QMCUR method enhance computational performance by calculating the QSVD of selected rows and columns. In summary, the proposed methods achieve a substantial speed-up while maintaining accuracy compared to other methods, especially when m is sufficiently large.

Example 4.3 In this example, we evaluate the performance of the proposed QMCUR method for color image compression. The dataset used in this example is the Set27[1] and two images are shown in Fig. 4. Each image, with a size of 683 × 1024 × 3, is represented by a pure quaternion matrix $\dot{\mathbf{X}} = \mathbf{R}\boldsymbol{i} + \mathbf{G}\boldsymbol{j} + \mathbf{B}\boldsymbol{k} \in \mathbb{Q}^{683\times 1024}$, where $\mathbf{R}$, $\mathbf{G}$, and $\mathbf{B}$ represent the RGB (red, green, blue) channels of color image. The elements of $\dot{\mathbf{X}}$ are $\dot{x}_{st} = r_{st}\boldsymbol{i} + g_{st}\boldsymbol{j} + b_{st}\boldsymbol{k}$, where r_{st}, g_{st}, and b_{st} denote the red, green, and

[1] https://github.com/Huang-chao-yan/dataset.

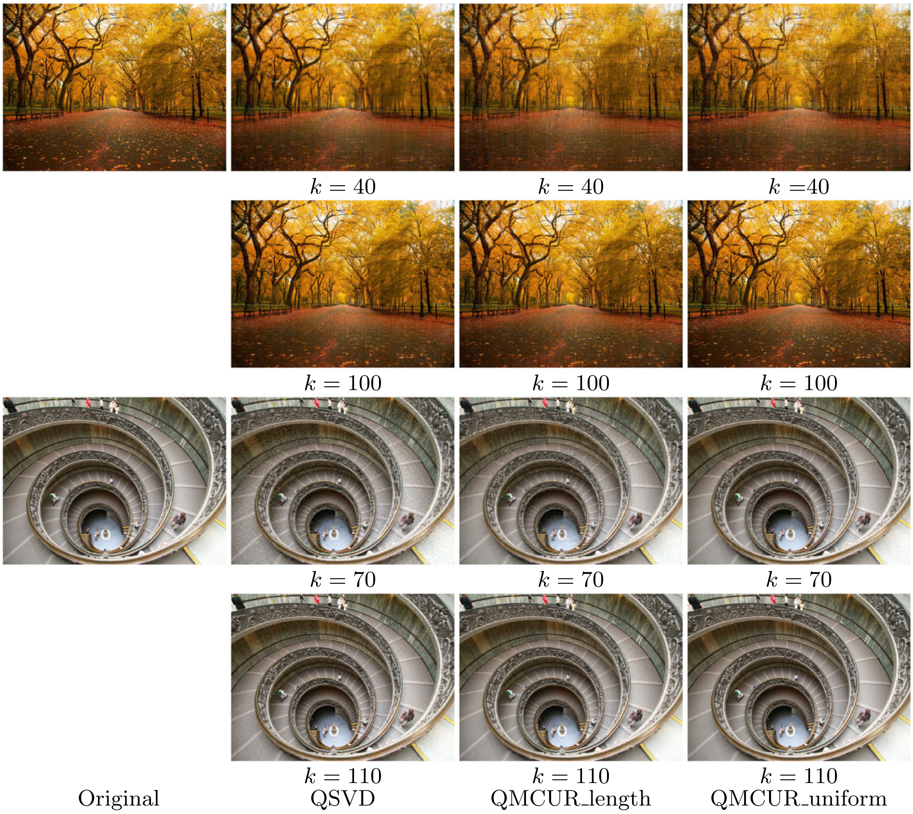

Fig. 4 Low-rank color image reconstruction. From top to bottom: Image01 and Image02, respectively

blue pixel values at the position (s, t) of the color image, respectively. Notably, the number of rows and columns is sethui to $k\log k$, where k represents the predefined rank of the quaternion matrix $\dot{\mathbf{X}} \in \mathbb{Q}^{m\times n}$.

Figure 3 illustrates the relative error and running time of all methods for various selected ranks. As observed, the performance of the proposed methods is superior to that of QSVD in terms of relative error when k is set to a suitable size. In terms of running time, the proposed methods are much faster than QSVD. Fig. 4 presents the reconstructed image quality for various target ranks, showing little difference in the visual effect between each original image and the approximate image obtained by the proposed methods and QSVD.

Example 4.4 In this part, we test the Algorithm 2 using four-color images of size 512×768×3 from the Kodak PhotoCD Dataset[2] under the random missing scenario. Several low-rank matrix completion methods are compared in this experiment, including LRQA-1 [18], QLNF [22], and Q-DFN [24]. The missing ratio (MR) is defined as the ratio of the number of missing elements to the total number of elements. To

[2] http://r0k.us/graphics/kodak/

provide a numerical evaluation of the recovered results, the peak signal-to-noise rate (PSNR), structural similarity (SSIM), and running time per iteration (in seconds) are employed. The higher PSNR and SSIM value imply better recovered results.

In Table 1, we list the PSNR, SSIM, and time values of different methods on color images, and the best values are highlighted in bold. Compared with competing methods, the proposed methods (QMCUR_length and QMCUR_uniform) achieve the best results in terms of all quantitative metrics. Furthermore, the proposed QMCUR_uniform is the fastest, demonstrating its feasibility on large-scale color images. To clearly observe the recovery performance, Fig. 5 presents the recovered images of four color images by all methods with MR = 80%. As observed, the

Table 1 PSNR, SSIM, and running time per iteration (in seconds) of results by different methods with different MRs on color images. The best and second best values are respectively highlighted in boldface and underlined

Data	Method	MR=90%			MR=80%			MR=70%		
		PSNR	SSIM	Time	PSNR	SSIM	Time	PSNR	SSIM	Time
 Image03	Observed	9.18	0.038	0.000	9.69	0.104	0.000	10.26	0.184	0.000
	LRQA-1	24.71	**0.954**	0.460	26.73	0.967	0.446	28.07	0.975	0.503
	DFN	24.89	0.953	0.348	27.04	0.967	0.280	28.51	**0.976**	0.350
	QLNF	24.63	0.949	0.738	26.53	0.964	1.178	27.98	0.972	3.999
	QMCUR_length	**25.12**	0.951	0.226	27.20	0.967	0.257	28.63	**0.976**	0.316
	QMCUR_uniform	24.99	0.950	**0.159**	**27.23**	**0.968**	**0.166**	**28.66**	**0.976**	**0.265**
 Image04	Observed	8.00	0.024	0.000	8.50	0.050	0.000	9.09	0.082	0.000
	LRQA-1	22.79	0.771	0.456	25.50	**0.868**	0.498	27.24	**0.905**	0.478
	DFN	23.10	0.757	0.270	25.96	0.852	0.350	27.79	0.889	0.326
	QLNF	23.25	0.772	1.188	25.87	0.854	2.727	27.15	0.856	3.768
	QMCUR_length	23.49	**0.776**	0.184	26.23	0.858	0.321	27.88	0.889	0.346
	QMCUR_uniform	**23.67**	**0.776**	**0.135**	**26.36**	0.848	**0.277**	**27.98**	0.883	**0.322**
 Image05	Observed	7.22	0.022	0.000	7.73	0.045	0.000	8.30	0.072	0.000
	LRQA-1	21.85	**0.725**	0.478	23.98	**0.798**	0.451	25.41	**0.841**	0.542
	DFN	21.86	0.702	0.362	24.13	0.783	0.335	25.60	0.825	0.349
	QLNF	21.85	0.690	1.483	23.83	0.764	1.583	18.75	0.648	2.867
	QMCUR_length	22.04	0.690	0.220	24.20	0.779	0.349	25.50	0.824	0.351
	QMCUR_uniform	**22.15**	0.688	**0.165**	**24.22**	0.769	**0.283**	**25.62**	0.822	**0.247**
 Image06	Observed	7.20	0.026	0.000	7.71	0.058	0.000	8.29	0.095	0.000
	LRQA-1	21.75	0.815	0.434	24.85	0.894	0.512	26.84	0.927	0.501
	DFN	22.58	**0.830**	0.286	25.46	**0.900**	0.318	**27.57**	**0.930**	0.362
	QLNF	22.37	0.810	1.310	25.02	0.883	2.789	26.44	0.907	4.110
	QMCUR_length	22.76	0.819	0.230	25.50	0.892	0.283	27.51	0.923	0.341
	QMCUR_uniform	**22.98**	0.829	**0.168**	**25.60**	0.894	**0.227**	27.55	0.924	**0.327**

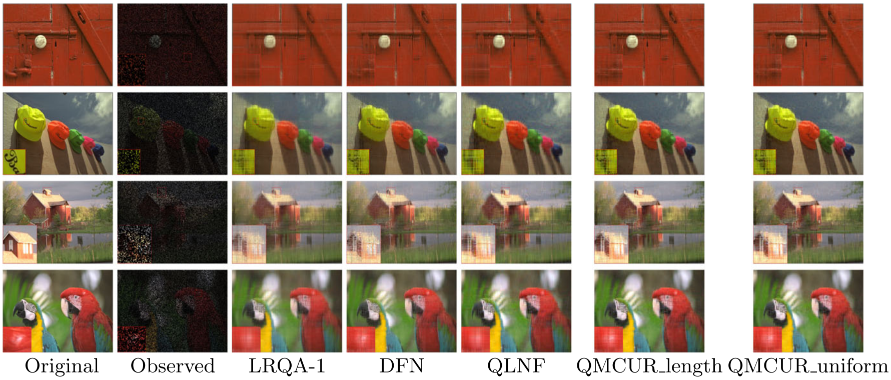

Fig. 5 Visual comparison of different methods for color image recovery. From top to bottom: Image03, Image04, Image05, and Image06, respectively

proposed methods generate better results than the compared methods. These observations demonstrate that our methods can be applicable for large-scale data processing.

5 Conclusion

This work presented an efficient quaternion matrix CUR method for computing low-rank approximation of a quaternion matrix. The QMCUR approximation offers a balance between accuracy and computational costs by selecting specific column and row submatrices of a given quaternion matrix. Besides, we conducted a perturbation analysis of the proposed approximation method and concluded that the error in the quaternion spectral norm is correlated with the noise quaternion matrix in the first order. Experimental results illustrated that the QMCUR approximation methods are significantly faster than comparative low-rank quaternion matrix approximation methods, without sacrificing the quality of reconstruction on both synthetic and color image datasets. In the future, we will extend the proposed methods to other high-dimensional data processing, e.g., quaternion tensor completion [47], tensor completion [48, 49] and multi-dimensional image recovery [50].

Acknowledgements This research is supported by University of Macau (MYRG2022-00108-FST, MYRG-CRG2022-00010-ICMS), the Science and Technology Development Fund, Macau S.A.R (0036/2021/AGJ), the National Key Research Project (2022YFB3904104), and the Key Project from National Natural Science Foundation of China (42230406).

Author Contributions Pengling Wu: Methodology, Software, Writing - Original Draft. Kit Ian Kou: Conceptualization, Validation, Resources. Hongmin Cai: Methodology, Validation, Writing - Review and Editing. Zhaoyuan Yu: Formal analysis, Writing - Review and Editing.

Funding This research is supported by University of Macau (MYRG2022-00108-FST, MYRG-CRG2022-00010-ICMS), the Science and Technology Development Fund, Macau S.A.R (0036/2021/AGJ), the National Key Research Project (2022YFB3904104), and the Key Project from National Natural Science Foundation of China (42230406).

Availability of supporting data The data sets generated during and/or analyzed during the current study are available from the corresponding author on reasonable request.

Declarations

Competing interests The authors declare no competing interests.

Ethical Approval Not applicable.

References

1. Hamilton, W.R.: Elements of quaternions. Green, & Company, Longmans (1866)
2. Ell, T.A., Le Bihan, N., Sangwine, S.J.: Quaternion fourier transforms for signal and image processing. John Wiley & Sons (2014)
3. Le Bihan, N., Mars, J.: Singular value decomposition of quaternion matrices: a new tool for vector-sensor signal processing. Signal Processing. **84**(7), 1177–1199 (2004)
4. Hirose, A., Yoshida, S.: Generalization characteristics of complex-valued feedforward neural networks in relation to signal coherence. IEEE Transactions on Neural Networks and Learning Systems. **23**(4), 541–551 (2012)
5. Zhou, H., Zhang, X., Zhang, C., Ma, Q.: Quaternion convolutional neural networks for hyperspectral image classification. Engineering Applications of Artificial Intelligence. **123**, 106234 (2023)
6. Wang, G., Zhang, D., Vasiliev, V.I., Jiang, T.: A complex structure-preserving algorithm for the full rank decomposition of quaternion matrices and its applications. Numerical Algorithms. **91**(4), 1461–1481 (2022)
7. Zhang, D., Jiang, T., Jiang, C., Wang, G.: A complex structure-preserving algorithm for computing the singular value decomposition of a quaternion matrix and its applications. Numerical Algorithms. **95**(4), 267–283 (2024)
8. Guo, Z., Jiang, T., Vasil'ev, V., Wang, G.: Complex structure-preserving method for schrödinger equations in quaternionic quantum mechanics. Numerical Algorithms. 1–17 (2023)
9. Jiang, T., Guo, Z., Zhang, D., Vasil'ev, V.: A fast algorithm for the schrödinger equation in quaternionic quantum mechanics. Applied Mathematics Letters. **150**, 108975 (2024)
10. Qi, L., Luo, Z., Wang, Q., Zhang, X.: Quaternion matrix optimization: Motivation and analysis. Journal of Optimization Theory and Applications. **193**(1–3), 621–648 (2022)
11. Zou, C., Kou, K.I., Wang, Y.: Quaternion collaborative and sparse representation with application to color face recognition. IEEE Transactions on Image Processing. **25**(7), 3287–3302 (2016)
12. Chen, J., Ng, M.K.: Color image inpainting via robust pure quaternion matrix completion: Error bound and weighted loss. SIAM Journal on Imaging Sciences. 15(3), 1469– 1498 (2022)
13. Li, C., Liu, Y., Wu, F., Che, M.: Randomized block krylov subspace algorithms for low-rank quaternion matrix approximations. Numerical Algorithms. 1–31 (2023)
14. Xu, T., Kong, X., Shen, Q., Chen, Y., Zhou, Y.: Deep and low-rank quaternion priors for color image processing. IEEE Transactions on Circuits and Systems for Video Technology. **33**(7), 3119–3132 (2023)
15. Jia, Z., Ng, M.K., Wang, W.: Color image restoration by saturation-value total variation. SIAM Journal on Imaging Sciences. **12**(2), 972–1000 (2019)
16. Gai, S., Yang, G., Wan, M., Wang, L.: Denoising color images by reduced quaternion matrix singular value decomposition. Multidimensional Systems and Signal Processing. **26**, 307–320 (2015)
17. Huang, C., Li, Z., Liu, Y., Wu, T., Zeng, T.: Quaternion-based weighted nuclear norm minimization for color image restoration. Pattern Recognition. **128**, 108665 (2022)
18. Chen, Y., Xiao, X., Zhou, Y.: Low-rank quaternion approximation for color image processing. IEEE Transactions on Image Processing. **29**, 1426–1439 (2020)
19. Yu, Y., Zhang, Y., Yuan, S.: Quaternion-based weighted nuclear norm minimization for color image denoising. Neurocomputing. **332**, 283–297 (2019)
20. Jia, Z., Ng, M.K., Song, G.: Robust quaternion matrix completion with applications to image inpainting. Numerical Linear Algebra with Applications. **26**(4), 2245 (2019)
21. Jia, Z., Jin, Q., Ng, M.K., Zhao, X.: Non-local robust quaternion matrix completion for large-scale color image and video inpainting. IEEE Transactions on Image Processing. **31**, 3868–3883 (2022)
22. Yang, L., Miao, J., Kou, K.I.: Quaternion-based color image completion via logarithmic approximation. Information Sciences. **588**, 82–105 (2022)

23. Ren, H., Ma, R., Liu, Q., Bai, Z.: Randomized quaternion qlp decomposition for low-rank approximation. Journal of Scientific Computing. **92**(3), 80 (2022)
24. Miao, J., Kou, K.I.: Quaternion-based bilinear factor matrix norm minimization for color image inpainting. IEEE Transactions on Signal Processing. **68**, 5617–5631 (2020)
25. Liu, Q., Ling, S., Jia, Z.: Randomized quaternion singular value decomposition for low-rank matrix approximation. SIAM Journal on Scientific Computing. 44(2), 870–900 (2022)
26. Goreinov, S.A., Tyrtyshnikov, E.E., Zamarashkin, N.L.: A theory of pseudoskeleton approximations. Linear Algebra and its Applications. **261**(1), 1–21 (1997)
27. Aldroubi, A., Hamm, K., Koku, A.B., Sekmen, A.: CUR decompositions, similarity matrices, and subspace clustering. Frontiers in Applied Mathematics and Statistics. **4** (2019)
28. Lin, P., Peng, S., Xiang, Y., Li, C., Cui, X., Zhang, W.: Structure-oriented CUR low-rank approximation for random noise attenuation of seismic data. IEEE Transactions on Geoscience and Remote Sensing. **61**, 1–13 (2023)
29. Drineas, P., Mahoney, M.W., Muthukrishnan, S.: Relative-error CUR matrix decompositions. SIAM Journal on Matrix Analysis and Applications. **30**(2), 844–881 (2008)
30. Wang, S., Zhang, Z.: Improving CUR matrix decomposition and the nyström approximation via adaptive sampling. The Journal of Machine Learning Research. **14**(1), 2729–2769 (2013)
31. Zhang, F.: Quaternions and matrices of quaternions. Linear Algebra and its Applications. **251**, 21–57 (1997)
32. Schafer, R.D.: On the algebras formed by the Cayley-Dickson process. American Journal of Mathematics. **76**(2), 435–446 (1954)
33. Xu, Y., Yu, L., Xu, H., Zhang, H., Nguyen, T.: Vector sparse representation of color image using quaternion matrix analysis. IEEE Transactions on Image Processing. **24**(4), 1315–1329 (2015)
34. Xiao, X., Chen, Y., Gong, Y., Zhou, Y.: 2D quaternion sparse discriminant analysis. IEEE Transactions on Image Processing. **29**, 2271–2286 (2020)
35. Kyrchei, I.: Weighted singular value decomposition and determinantal representations of the quaternion weighted Moore-Penrose inverse. Applied Mathematics and Computation. **309**, 1–16 (2017)
36. Kyrchei, I.: Determinantal representations of the Moore-Penrose inverse over the quaternion skew field and corresponding cramer's rules. Linear and Multilinear Algebra. **59**(4), 413–431 (2011)
37. Wang, Q.: The general solution to a system of real quaternion matrix equations. Computers & Mathematics with Applications. **49**(5), 665–675 (2005)
38. Drineas, P., Kannan, R., Mahoney, M.W.: Fast monte carlo algorithms for matrices iii: Computing a compressed approximate matrix decomposition. SIAM Journal on Computing. 36(1), 184–206 (2006)
39. Chiu, J., Demanet, L.: Sublinear randomized algorithms for skeleton decompositions. SIAM Journal on Matrix Analysis and Applications. 34(3), 1361–1383 (2013)
40. Ling, C., He, H., Qi, L.: Singular values of dual quaternion matrices and their low-rank approximations. Numerical Functional Analysis and Optimization. **43**(12), 1423–1458 (2022)
41. Gillis, N., Glineur, F.: Low-rank matrix approximation with weights or missing data is np-hard. SIAM Journal on Matrix Analysis and Applications. **32**(4), 1149–1165 (2011)
42. Liu, J., Musialski, P., Wonka, P., Ye, J.: Tensor completion for estimating missing values in visual data. IEEE Transactions on Pattern Analysis and Machine Intelligence. **35**(1), 208–220 (2013)
43. Ahmadi-Asl, S., Asante-Mensah, M.G., Cichocki, A., Phan, A.H., Oseledets, I., Wang, J.: Fast cross tensor approximation for image and video completion. Signal Processing. **213**, 109121 (2023)
44. Vandereycken, B.: Low-rank matrix completion by riemannian optimization. SIAM Journal on Optimization. **23**(2), 1214–1236 (2013)
45. Li, Y., Wei, M., Zhang, F., Zhao, J.: Real structure-preserving algorithms of householder based transformations for quaternion matrices. Journal of Computational and Applied Mathematics. **305**, 82–91 (2016)
46. Jia, Z., Ng, M.K., Song, G.: Lanczos method for large-scale quaternion singular value decomposition. Numerical Algorithms. **82**, 699–717 (2019)
47. Miao, J., Kou, K.I., Liu, W.: Low-rank quaternion tensor completion for recovering color videos and images. Pattern Recognition. **107**, 107505 (2020)
48. Li, B., Zhao, X., Zhang, X., Ji, T., Chen, X., Ng, M.K.: A learnable group-tube transform induced tensor nuclear norm and its application for tensor completion. SIAM Journal on Imaging Sciences. 16(3), 1370–1397 (2023)
49. Li, B., Zhao, X., Ji, T., Zhang, X., Huang, T.: Nonlinear transform induced tensor nuclear norm for tensor completion. Journal of Scientific Computing. **92**(3), 83 (2022)

50. Lin, J., Huang, T., Zhao, X., Ji, T., Zhao, Q.: Tensor robust kernel PCA for multidimensional data. IEEE Transactions on Neural Networks and Learning Systems. 1–13 (2024)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.