Quaternion Optimized Model with Sparse Regularization for Color Image Recovery

Liqiao Yang, Yang Liu Senior Member, IEEE, Kit Ian Kou

Abstract—This paper addresses the color image completion problem in accordance with low-rank quaternion matrix optimization that is characterized by sparse regularization in a transformed domain. This research was inspired by an appreciation of the fact that different signal types, including audio formats and images, possess structures that are inherently sparse in respect of their respective bases. Since color images can be processed as a whole in the quaternion domain, we depicted the sparsity of the color image in the quaternion discrete cosine transform (QDCT) domain. In addition, the representation of a low-rank structure that is intrinsic to the color image is a vital issue in the quaternion matrix completion problem. To achieve a more superior low-rank approximation, the quaternion-based truncated nuclear norm (QTN) is employed in the proposed model. Moreover, this model is facilitated by a competent alternating direction method of multipliers (ADMM) based on the algorithm. Extensive experimental results demonstrate that the proposed method can yield vastly superior completion performance in comparison with the state-of-the-art low-rank matrix/quaternion matrix approximation methods tested on color image recovery.

Index Terms—Quaternion matrix completion, sparse, quaternion discrete cosine transform, low-rank, truncated nuclear norm.

I. INTRODUCTION

In terms of image processing, the purpose of color image completion is to use the known pixels in an image, however limited, to recover missing pixels. To achieve this, many matrix-based completion (MC) approaches have been designed, though in general, a color image is processed by separating RGB channels to three matrices and prior knowledge about the desired model is then used to achieve inpainting. An effective and widely used source of such prior knowledge is low-rankness.

In [1], the nuclear norm (NN) was proved to be the tightest feasible convex relaxation of the matrix rank function, being NP hard to further minimize. Following on from this, other researchers have shown that when the singular value is treated equally in the nuclear norm minimization process, the ensuing inpainting results are sub-optimal. Based on this, various work has been developed to optimize NN, such as the weighted nuclear norm [2], the truncated nuclear norm (TNN) [3], and the logarithmic norm [4], etc. However, although these approaches have improved the inpainting results for color images, especially as compared with classic NN methods, such matrix-based approaches involve dimension reduction, which can destroy the structure of the color image.

To avoid this, quaternion-based approaches have gradually become more commonly used in image processing, as these allow the values of one pixel in the color image to be put in three imaginary parts of one quaternion to form a more reliable quaternion matrix. As the extension of a complex number, a quaternion number provides a representation that exactly matches the structure of the color pixel usually containing three values drawn from the RGB channels separately. Thus, quaternion-based approaches can be widely used in various types of color image processing, including color image recovery [5]–[7], color image watermarking [8], color face recognition [9], and so on.

For color image completion in the quaternion domain, the typical prior knowledge type is the low-rankness, analogous to the prior matrix-based cases. As an example, the authors in [5] proposed a low-rank quaternion approximation model based on several modified quaternion nuclear norm (QNN), including laplace, geman, and weighted Schatten-γ functions. Offering two reasons why this series of methods could improve the inpainting performance. The first is that a single color image can be put in one quaternion matrix for processing, allowing correlation among RGB channels may be preserved, while the other is that these improved functions can approximate the rank of the quaternion matrix more precisely than QNN, with the latter being identical to matrix-based methods. However, calculating the quaternion singular value decomposition (QSVD) is computationally complex, with the operation needed to calculate the singular value decomposition of corresponding complex matrices being twice the size of the original quaternion matrix in each iteration.

To overcome the time-consuming nature of QSVD, researchers in [6] developed a low-rank quaternion matrix factorization approach by factorizing the target quaternion matrix into the product of two smaller quaternion factor matrices. This factorization method is based on three kinds of quaternion bilinear matrix norms, quaternion double Frobenius norm (QDFN), quaternion double nuclear norm (Q-DNN), and quaternion Frobenius/nuclear norm (Q-FNN). In this factorization model, the low-rank estimation process retains higher levels of computing efficiency due to the fact that only two smaller quaternion factor matrices need to be optimized. However, as in the matrix-based cases, this factorization method may become trapped in the local minima [10], causing the computation of QSVD for factor quaternion matrices to remain necessary in each iteration.

Liqiao Yang is with the Department of Mathematics, Faculty of Science and Technology, University of Macau, Macau, China (e-mail: liqiao-y@163.com).

Yang Liu is with the College of Mathematics and Computer Science and the College of Mathematical Medicine, Zhejiang Normal University, Jinhua 321004, China (e-mail: liuyang@zjnu.edu.cn).

Kit Ian Kou is with the Department of Mathematics, Faculty of Science and Technology, University of Macau, Macau, China (e-mail: kikou@umac.mo).

arXiv:2204.08629v1 [cs.CV] 19 Apr 2022
As described above, the various low-rank completion methods can be divided into two branches: rank estimators supported by various regularization processes and low-rank factorization. Based on such assumptions about low-rankness, the previously developed quaternion-based methods can thus process color images overall to obtain better recovery results; however, they ignore other important properties such as sparsity. The direct motivation to reconsider this can be derived from the fact that various kinds of signals, including audio and images, have naturally sparse structures, with regard to given bases such as Fourier and wavelet [11]. This fact has inspired new approaches in signal processing in recent years, and especially for vision tasks based on matrix optimization, low-rankness has now been used extensively, especially for face reorganization [12], [13], image denoising and completion [14]–[17], and similar tasks. In terms of image completion, sparsity is also an important property that can be used in MC. Specifically, in certain transformed domains, piecewise smooth functions have the property of being sparse and, based on this, in [14], such sparsity is depicted by using the $l_1$ norm in the discrete cosine transform (DCT), alongside low-rankness was as depicted by TNN. The TNN was found to obtain a more accurate approximation (DCT), alongside low-rankness was as depicted by TNN. The work focuses on quaternion matrix completion based on combining low-rankness and sparsity. The underlying concept is that the quaternion-based method can preserve the structure of the color image, allowing the sparsity to be formulated as an $l_1$ norm regularizer in the transformed domain (QDCT).

The closed form solution of the model, combining the Frobenius norm and the $l_1$ norm, is thus proposed and supported with theoretical analysis.

Extensive experimental results on real color images demonstrate the competitive performance of the proposed method in comparison with several state-of-the-art methods.

The outline of the remainder of this article is as follows. Section II generalizes some preliminaries, while Section III offers the proposed LRQR-SR model and the corresponding two-step ADMM-based optimization. Section IV then presents the experimental results to demonstrate the efficiency of the proposed LRQR-SR, while Section V offers a conclusion to the work.

II. NOTATIONS AND PRELIMINARIES

A. Notations

In the real domain $\mathbb{R}$, we denote scalar, vector, and matrix as $a$, $\mathbf{a}$, and $\mathbf{A}$, respectively. In the quaternion domain $\mathbb{H}$, we denote scalar, vector, and matrix as $\mathbf{a}$, $\mathbf{A}$, and $\mathbf{A}$, respectively. Besides, we denote the complex space as $\mathbb{C}$. For a quaternion $\mathbf{q}$, we denote the real part and imaginary part as $\Re(\mathbf{q})$ and $\Im(\mathbf{q})$. We denote transpose, conjugate transpose, and inverse as $(\cdot)^T$, $(\cdot)^\dagger$, and $(\cdot)^{-1}$, respectively. We use $\| \cdot \|_F$ and $\| \cdot \|_*$ to represent the Frobenius norm and nuclear norm. We define the inner product of $\mathbf{s}_1$ and $\mathbf{s}_2$ as $\langle \mathbf{s}_1, \mathbf{s}_2 \rangle \triangleq \text{tr}(\mathbf{s}_1^\dagger \mathbf{s}_2)$, where $\text{tr}(\cdot)$ is the trace function. Both $\mathbf{1}_{r \times r}$ and $\mathbf{I}_r$ denote the $r \times r$ identity matrix.

B. Preliminaries

Quaternions are proposed by Hamilton in 1843 [22]. A quaternion number $\mathbf{q} \in \mathbb{H}$ is combined by a real part and three imaginary parts, and can be written as following form:

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

where $q_n \in \mathbb{R}$ ($n = 0, 1, 2, 3$), and $i, j, k$ are three imaginary number units which have the following relationships:

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

$\Re(\mathbf{q}) \triangleq q_0$ is the real part of $\mathbf{q}$. $\Im(\mathbf{q}) \triangleq q_1 i + q_2 j + q_3 k$ is the imaginary part of $\mathbf{q}$. Hence $\mathbf{q} = \Re(\mathbf{q}) + \Im(\mathbf{q})$. Besides, when real part $q_0 = 0$, $\mathbf{q}$ is a pure quaternion. The conjugate and the modulus of $\mathbf{q}$ are defined as: $\mathbf{q}^* \triangleq q_0 - q_1 i - q_2 j - q_3 k$ and $|\mathbf{q}| \triangleq \sqrt{q_0^2 + q_1^2 + q_2^2 + q_3^2}$. Assuming two quaternions $\mathbf{p}$ and $\mathbf{q} \in \mathbb{H}$, the addition and multiplication are separately defined as following:

$$\mathbf{p} + \mathbf{q} = (p_0 + q_0) + (p_1 + q_1)i + (p_2 + q_2)j + (p_3 + q_3)k;$$
\[ p\tilde{q} = (p_0q_0 - p_1q_1 - p_2q_2 - p_3q_3) + (p_0q_1 + p_1q_0 + p_2q_3 - p_3q_2)i + (p_0q_2 - p_1q_3 + p_2q_0 + p_3q_1)j + (p_0q_3 + p_1q_2 - p_2q_1 + p_3q_0)k. \]

It is important to note that the multiplication in the quaternion domain is not commutative. For quaternion matrix \( Q \in \mathbb{H}^{M \times N} \), the Frobenius norm is defined as:

\[ \| Q \|_F = \sqrt{\sum_{i=1}^{M} \sum_{j=1}^{N} |q_{ij}|^2} = \sqrt{tr(Q^H Q)}. \]

**Definition 1 (The Cayley-Dickson form [23]):** The Cayley-Dickson form of quaternion matrix \( Q = (q_{ij}) \in \mathbb{H}^{M \times N} \) is defined as the maximum number of right (left) linearly independent columns (rows) of \( Q \).

**Theorem 1 (QSVD [24]):** Given a quaternion matrix \( Q = (q_{ij}) \in \mathbb{H}^{M \times N} \) be of rank \( r \). There are two unitary quaternion matrices \( U \in \mathbb{H}^{M \times M} \) and \( V \in \mathbb{H}^{N \times N} \) such that:

\[ Q = U \begin{pmatrix} \sigma_1 & 0 & \cdots & 0 \\ 0 & \sigma_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_r \end{pmatrix} V^H = U\Lambda V^H, \]

where \( \Sigma_r = \text{diag}(\sigma_1, \ldots, \sigma_r) \in \mathbb{R}^{r \times r} \), and all singular values \( \sigma_i > 0, i = 1, \ldots, r \).

**Definition 2 (The rank of quaternion matrix [24]):** The rank of quaternion matrix \( Q = (q_{ij}) \in \mathbb{H}^{M \times N} \) is defined as the maximum number of right (left) linearly independent columns (rows) of \( Q \).

**Theorem 2 (QSNV [24]):** The nuclear norm of the quaternion matrix \( Q \in \mathbb{H}^{M \times N} \) is given by:

\[ \| Q \|_{\text{nn}} = \sum_{i=1}^{\min(M,N)} \sigma_i(Q). \]

**Theorem 3 (The rank of quaternion matrix [24]):** For any quaternion matrix \( Q \in \mathbb{H}^{M \times N} \), and any matrices \( A \in \mathbb{H}^{M \times M} \) and \( B \in \mathbb{H}^{N \times N} \) that satisfy with \( A^H = A_{r \times r} \) and \( B^H = B_{r \times r} \) (any positive integer \( r \leq \min(M,N) \)), we have:

\[ |tr(AQB^H)| \leq \sum_{i=1}^{r} \sigma_i(Q). \]

**III. LOW-RANK QUATERNION RECOVERY WITH SPARSE REGULARIZATION**

This section is divided into three parts. Subsection III-A gives the formulation of the proposed LRQR-SR, while subsection III-B presents the quaternion discrete cosine transform that we used in this paper, and subsection III-C gives the corresponding ADMM-based optimization algorithm.

**A. Problem Formulation**

This section introduces the formulation of a low-rank quaternion optimized model for color image recovery. Let \( \Omega \in \mathbb{H}^{M \times N} \) be the partial observed color image. The purpose of the resulting problem is thus to recover \( X \in \mathbb{H}^{M \times N} \) by improving QNN as applied to the characterization of the low-rank property more accurately. The QNN model can be formulated as:

\[ \min_{X} \| \tilde{X} \|_r \quad \text{s.t.} \quad P_{\Omega}(X) = P_{\Omega}(	ilde{O}), \quad (3) \]

where \( \Omega \) is the index set of the observed data, and the linear operation \( P_{\Omega}(\cdot) \) is the operator that indicates that the elements in \( \Omega \) are remain while other elements are zero.

As the several largest singular values will not influence the rank, the TQNN model is proposed, and the estimation of low-rankness should thus be more accurate. Based on model (3) and Definition 4, the TQNN model can thus be formulated as:

\[ \min_{X} \| \tilde{X} \|_r \quad \text{s.t.} \quad P_{\Omega}(X - \tilde{O}) = 0, \quad (4) \]

**B. Low-Rank Recovery**

Being low rank is a necessary but insufficient condition for MC [17]. Motivated by this observation and the success of sparsity as utilized in MC [14]-[16] the LRQR-SR model is developed, which considers both a low-rank constraint and sparsity in the quaternion domain. The low-rank constraint is depicted using TQNN in this case, while sparsity is depicted using the \( l_1 \) norm. As in the previously mentioned strategies, the quaternion matrix is assumed to be sparse in a certain transformed domain. Hence, the resulting model can be formulated as:

\[ \min_{X} \| \tilde{X} \|_r - \sum_{i=r+1}^{\min(m,n)} \sigma_i(\tilde{X}) + \lambda \| D \|_1 \quad \text{s.t.} \quad P_{\Omega}(X - \tilde{O}) = 0, \quad (5) \]

where \( T(\cdot) \) is the transform operator, \( D \) is the transformed quaternion matrix, and \( \lambda \) is a positive number.

However, it is hard to solve problem (5) directly due to the fact that TQNN is nonconvex. To address this problem,
Theorem 2 is applied such that problem (5) can be rewritten as
\[
\min_{\hat{X}} \| \hat{X} \|_* - \max_{A^H=1,B^H=1} |tr(\hat{A}XB^H)| + \lambda \| \hat{D} \|_1 \\
\text{s.t. } P_{11}(\hat{X} - \hat{O}) = 0
\]
(6)
where \( \hat{A} = (\hat{u}_1, \ldots, \hat{u}_r)^H \) and \( \hat{B} = (\hat{v}_1, \ldots, \hat{v}_r)^H \), \{\hat{u}_1, \ldots, \hat{u}_r\} \) and \{\hat{v}_1, \ldots, \hat{v}_r\} \) are the first \( r \) columns of \( \hat{U} \) and \( \hat{V} \). \( \hat{U} \) and \( \hat{V} \) are left and right unitary quaternion matrices that are calculated by QSVD of \( \hat{X} \).

In this way, the whole procedure of the method can be divided into two main steps: in the first step, the quaternion matrices are computed by QSVD, and then the main goal becomes to optimize problem (6). The overall procedure is summarized in Algorithm 1.

Algorithm 1 Low-rank Quaternion Recovery with Sparse Regularization

**Input:** the observed quaternion matrix \( \hat{O} \in \mathbb{H}^{M \times N} \), the position set of observed elements \( \Omega \), and the tolerance \( \varepsilon_0 \).

1. **Initial** the initial number of iteration \( \hat{k} = 1 \), \( \hat{X}_1 = \hat{O} \).
2. **Repeat**
   3. **Step 1.** Calculating the QSVD of the given \( \hat{X}_k \)
      \( [\hat{U}_k, \hat{\Sigma}_k, \hat{V}_k] = \text{QSVD}(\hat{X}_k) \)
   4. where \( \hat{U}_k = (\hat{u}_1, \ldots, \hat{u}_m) \in \mathbb{H}^{M \times M} \), \( \hat{V}_k = (\hat{v}_1, \ldots, \hat{v}_n) \in \mathbb{H}^{N \times N} \).
   5. Calculating \( \hat{A}_k = (\hat{u}_1, \ldots, \hat{u}_r)^T \in \mathbb{H}^{r \times M} \) and \( \hat{B}_k = (\hat{v}_1, \ldots, \hat{v}_r)^T \in \mathbb{H}^{r \times N} \).
   6. **Step 2.** Solving the optimization problem as followed
      \( \hat{X}_{k+1} = \arg \min_{\hat{X}} \| \hat{X} \|_* - |tr(\hat{A}_k\hat{X}\hat{B}_k^H)| + \lambda \| \hat{D} \|_1, \)
      \( \text{s.t. } P_{11}(\hat{X} - \hat{O}) = 0 \)
   7. **Until convergence** \( \| \hat{X}_{k+1} - \hat{X}_k \|_F \leq \varepsilon_0 \).

**Output:** the recovered quaternion matrix \( \hat{X}_{opt} \).

B. Quaternion Discrete Cosine Transform

Several key points of utilizing QDCT are outlined below.

1) The reasons for utilizing QDCT: Most importantly, the proposed method operates in the quaternion domain, where each color image can be handled by quaternion algebra as a whole. In order to avoid destroying the RGB structure and to improve the accuracy of image recovery, the entire process must thus be operated in the quaternion domain. In addition, the spectral coefficients obtained using QDCT have strong energy and good redundancy elimination characteristics [26], while QDCT itself is easy to quantitatively analyze. Finally, in the real and complex fields, the energy concentration of the input information throughout two-dimensional DCT is higher than that of the input information after DFT. Research into QDCT is driven by the existence of successful applications in both the real and complex domains, and for these reasons, QDCT is adopted in the proposed method.

Fig. 1 gives an illustration of the proposed sparse representation on the color image “Parrot”. The first image is the original image; the second, third, and fourth image respectively display the coefficients after QDCT, QDFT, and DCT (grayscale image) using a logarithmic scale. After transformation, the coefficients of QDCT and DCT are mainly concentrated in the upper left corner, and most of the remaining coefficients are close to zero. However, after QDFT transformation, the coefficients are mainly concentrated at four corners, which means that utilizing cosine transform is superior to Fourier transform to process images in the quaternion domain. Besides, when comparing QDCT with DCT, especially in the upper left corner, it can be observed that QDCT has higher energy compaction than DCT.

2) Definition of QDCT: As the multiplication of quaternions is non-commutative, there are two forms of QDCT: a left-handed form \( \text{QDCT}_L \) and a right-handed form \( \text{QDCT}_R \).

These can be formulated as the following equations, respectively [27]:

\[
\text{QDCT}_L(p, s) = \alpha(p)\alpha(s) \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \hat{u} \cdot \tilde{F}(m, n) \cdot C(p, s, m, n)
\]

(7)

\[
\text{QDCT}_R(p, s) = \alpha(p)\alpha(s) \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \tilde{F}(m, n) \cdot C(p, s, m, n) \cdot \hat{u},
\]

(8)

where \( \tilde{F}(m, n) \in \mathbb{H}^{M \times N} \), \( m \) and \( n \) is the row and column of quaternion matrix \( \tilde{F} \). \( \hat{u} \) is a pure quaternion and satisfies \( \hat{u}^2 = -1 \). The values of \( \alpha(p) \), \( \alpha(s) \) and \( C(p, s, m, n) \) are analogous to DCT in the real domain:

\[
\alpha(p) = \begin{cases} 
\sqrt{1/M} & p = 0 \\
\sqrt{2/M} & p \neq 0
\end{cases}, \quad \alpha(s) = \begin{cases} 
\sqrt{1/N} & s = 0 \\
\sqrt{2/N} & s \neq 0
\end{cases}
\]

\[
C(p, s, m, n) = \cos[\pi(2m+1)p] \cos[\pi(2n+1)s] / 2M.
\]

Besides, the corresponding inverse transformation of QDCT is the Inverse Quaternion Discrete Cosine Transform (IQDCT). These are thus the transformation pairs of each other, and satisfy the following relationship:

\[
\tilde{F}(m, n) = \text{IQDCT}_L[\text{QDCT}_L(\tilde{F}(m, n))]
\]

\[
\tilde{F}(m, n) = \text{IQDCT}_R[\text{QDCT}_R(\tilde{F}(m, n))].
\]

In the proposed algorithm, \( \text{QDCT}_L \) is utilized to calculate QDCT.

3) Calculation of QDCT\(_L\): To simplify the calculation of QDCT, we take full advantage of the Cayley Dickson form seen in Definition 1, as in [27] is used. The whole process of QDCT\(_L\) calculation is as follows:

a) Transforming the given quaternion matrix \( \tilde{F}(m, n) \in \mathbb{H}^{M \times N} \) to the Cayley Dickson form \( \tilde{F}(m, n) = \).

b) Performing the transformation according to the Cayley Dickson form, the result is \( \tilde{F}(m, n) = \).

c) Calculating the QDCT coefficients on the transformed quaternion matrix, the result is \( \).

d) Transforming back to the original form by the inverse Cayley Dickson form, the result is \( \).

e) Taking the inverse transformation of the obtained coefficients, the result is \( \).

Fig. 1. The comparison of energy compaction after different transformation.
\( \mathbf{F}_p(m, n) + \mathbf{F}_q(m, n) \)j, where \( \mathbf{F}_p(m, n) \) and \( \mathbf{F}_q(m, n) \) ∈ \( \mathbb{C}^{M \times N} \)

b) Calculating the DCT of complex matrices \( \mathbf{F}_p(m, n) \) and \( \mathbf{F}_q(m, n) \). The results are denoted as \( \text{DCT}_C(\mathbf{F}_p(m, n)) \) and \( \text{DCT}_C(\mathbf{F}_q(m, n)) \), respectively.

c) Using \( \text{DCT}_C(\mathbf{F}_p(m, n)) \) and \( \text{DCT}_C(\mathbf{F}_q(m, n)) \) to form a quaternion matrix: \( \hat{\mathbf{F}}(m, n) = \text{DCT}_C(\mathbf{F}_p(m, n)) + \text{DCT}_C(\mathbf{F}_q(m, n)) \).

d) Multiplying \( \hat{\mathbf{F}}(m, n) \) with the quaternion factor \( \hat{u} \) to get the final result \( \text{QDCT}_L \):
\( \text{QDCT}_L(\hat{\mathbf{F}}(m, n)) = \hat{u} \cdot \hat{\mathbf{F}}(m, n) \).

C. ADMM-Based Optimization Algorithm

Following the model we discussed in subsection III-A and the transformation introduced in III-B, ADMM was adopted to optimize problem (6). This involves introducing auxiliary variable \( \hat{\mathbf{H}} \) and reformulating (6) as

\[
\begin{align*}
\min_{\mathbf{X}} \quad & \| \mathbf{X} \|_* - \max_{\mathbf{A}, \mathbf{B} \in \mathbb{R}^{M \times 1}, \mathbf{H} \in \mathbb{R}^{N \times 1}} |\text{tr}(\mathbf{A}\mathbf{H}\mathbf{B}^H)| + \lambda \| \mathbf{D} \|_1 \\
\text{s.t.} \quad & P_{\Omega}(\mathbf{X} - \hat{\mathbf{O}}) = 0 \\
& \hat{\mathbf{H}} = \mathbf{X} \\
& \text{QDCT}_L(\mathbf{X}) = \mathbf{D}.
\end{align*}
\]

In analogy with the ADMM framework adopted in the complex domain [28], as the multiplication is not commutative in the quaternion domain, the augmented Lagrangian function of (9) can be written as

\[
L(\mathbf{X}, \hat{\mathbf{H}}, \hat{\mathbf{D}}, \hat{\mathbf{Y}}, \hat{\mathbf{Z}}, \beta) = \\
\| \mathbf{X} \|_* - |\text{tr}(\mathbf{A}\hat{\mathbf{H}}\mathbf{B}^H)| + \lambda \| \mathbf{D} \|_1 \\
+ \Re(\text{tr}(\hat{\mathbf{Y}}^H(\mathbf{X} - \hat{\mathbf{H}}))) + \frac{\beta}{2} \| \mathbf{X} - \hat{\mathbf{H}} \|_F^2 \\
+ \Re(\text{tr}(\hat{\mathbf{Z}}^H(\hat{\mathbf{D}} - \text{QDCT}_L(\mathbf{X})))) \\
+ \frac{\beta}{2} \| \hat{\mathbf{D}} - \text{QDCT}_L(\mathbf{X}) \|_F^2,
\]

where \( \hat{\mathbf{Y}} \) and \( \hat{\mathbf{Z}} \) are the Lagrange multipliers, and \( \beta \) is the positive penalty parameter. Under the framework of ADMM, the variables \( \mathbf{X}, \hat{\mathbf{H}}, \hat{\mathbf{D}}, \hat{\mathbf{Y}}, \) and \( \hat{\mathbf{Z}} \) are updated alternately in the \( p \)-th iteration using

\[
\begin{align*}
\mathbf{X}^{p+1} &= \arg \min_{\mathbf{X}} L(\mathbf{X}, \hat{\mathbf{H}}^p, \hat{\mathbf{D}}^p, \hat{\mathbf{Y}}^p, \hat{\mathbf{Z}}^p, \beta^p), \\
\hat{\mathbf{D}}^{p+1} &= \arg \min_{\hat{\mathbf{D}}} L(\hat{\mathbf{X}}^{p+1}, \hat{\mathbf{H}}^p, \hat{\mathbf{D}}, \hat{\mathbf{Y}}^p, \hat{\mathbf{Z}}^p, \beta^p), \\
\hat{\mathbf{H}}^{p+1} &= \arg \min_{\hat{\mathbf{H}}} L(\hat{\mathbf{X}}^{p+1}, \hat{\mathbf{H}}, \hat{\mathbf{D}}^p, \hat{\mathbf{Y}}^p, \hat{\mathbf{Z}}^p, \beta^p), \\
\hat{\mathbf{Y}}^{p+1} &= \hat{\mathbf{Y}}^p + \beta^p(\hat{\mathbf{X}}^{p+1} - \hat{\mathbf{H}}^{p+1}), \\
\hat{\mathbf{Z}}^{p+1} &= \hat{\mathbf{Z}}^p + \beta^p(\hat{\mathbf{D}}^{p+1} - \text{QDCT}_L(\hat{\mathbf{X}}^{p+1})).
\end{align*}
\]

The \( \mathbf{X} \) subproblem is

\[
\begin{align*}
\mathbf{X}^{p+1} &= \arg \min_{\mathbf{X}} \| \mathbf{X} \|_* + \Re(\text{tr}(\hat{\mathbf{Y}}^H(\mathbf{X} - \hat{\mathbf{H}}^p))) \\
&\quad + \frac{\beta^p}{2} \| \mathbf{X} - \hat{\mathbf{H}}^p \|_F^2 + \Re(\text{tr}(\hat{\mathbf{Z}}^H(\hat{\mathbf{D}}^p - \text{QDCT}_L(\mathbf{X})))) \\
&\quad + \frac{\beta^p}{2} \| \hat{\mathbf{D}}^p - \text{QDCT}_L(\mathbf{X}) \|_F^2 \\
= \arg \min_{\mathbf{X}} \| \mathbf{X} \|_* + \frac{\beta^p}{2} \| \mathbf{X} - \hat{\mathbf{H}}^p + \hat{\mathbf{Y}}^p / \beta^p \|_F^2 \\
&\quad + \frac{\beta^p}{2} \| \hat{\mathbf{D}}^p - \text{QDCT}_L(\mathbf{X}) + \hat{\mathbf{Z}}^p / \beta^p \|_F^2.
\end{align*}
\]

In the last term of (12), \( \mathbf{X} \) can not be separated directly as the transformation. Despite this, the Parseval theorem in the quaternion domain indicates that the total energy of signal computed in the quaternion domain and total energy of signal computed in the spatial domain must be the same [29], [30]. This means that a unitary transformation preserves energy conservation under the Frobenius norm, and thus the last term of (12) can be rewritten as

\[
\frac{\beta^p}{2} \| \hat{\mathbf{D}}^p - \text{QDCT}_L(\mathbf{X}) + \hat{\mathbf{Z}}^p / \beta^p \|_F^2.
\]

The closed solution of the above problem is

\[
\mathbf{X}^{p+1} = \mathcal{D}_{\Sigma}(\frac{1}{2}(\mathbf{X}^p + \mathbf{Y}^p / \beta^p + \mathcal{T}(\hat{\mathbf{D}}^p + \hat{\mathbf{Z}}^p / \beta^p))),
\]

where \( \mathcal{D}_{\Sigma}(\cdot) \) is the quaternion singular value shrinkage operator [5] is defined as

\[
\mathcal{D}_{\Sigma}(\mathbf{A}) = \hat{U}\hat{\Sigma}\hat{V}^H, \quad \hat{\Sigma} = \text{diag}(\max\{\sigma_i - \tau, 0\})
\]

where \( \hat{U} \), \( \hat{V} \), and \( \sigma_i \) are obtained by computing QSVD of quaternion matrix \( \mathbf{A} \).

The \( \mathbf{D} \) subproblem is

\[
\begin{align*}
\hat{\mathbf{D}}^{p+1} &= \arg \min_{\hat{\mathbf{D}}} \| \hat{\mathbf{D}} \|_1 + \Re(\text{tr}(\hat{\mathbf{Y}}^H(\hat{\mathbf{D}} - \mathcal{T}(\hat{\mathbf{X}}^{p+1})))) \\
&\quad + \frac{\beta^p}{2} \| \hat{\mathbf{D}} - \mathcal{T}(\hat{\mathbf{X}}^{p+1}) \|_F^2 \\
= \arg \min_{\hat{\mathbf{D}}} \| \hat{\mathbf{D}} \|_1 + \frac{\beta^p}{2} \| \hat{\mathbf{D}} - \mathcal{T}(\hat{\mathbf{X}}^{p+1}) + \hat{\mathbf{Z}}^H / \beta^p \|_F^2.
\end{align*}
\]

To obtain the optimal solution of (14), we have the following theorem.
Theorem 3: For any $\lambda > 0$, the closed solution of problem
\[
\min_{X} \lambda \|X\|_1 + \|Y - X\|_F^2
\]
can be given by
\[
\hat{X}_{opt} = S_{2\lambda}(Y),
\]
where $S_r(\cdot)$ represents the element-wise soft thresholding
operator defined by
\[
S_r(x) = \frac{x}{|x|} \max\{1, |x| - \tau, 0\}.
\]

The proof of Theorem 3 is given in the Appendix. Based on Theorem 3, problem (14) has a closed-form solution given by
\[
\hat{D}^{p+1} = S_{\frac{\lambda}{\beta}}(\frac{1}{\beta} \hat{H}^p + \hat{Y}^p/\beta^p + T(\hat{D}^p + \hat{Z}^p/\beta^p)).
\]

7: Update $\hat{X}^{p+1}$
\[
\hat{X}^{p+1} = \hat{D}^{p+1}S_{\frac{\lambda}{\beta}}(\frac{1}{\beta} \hat{H}^p + \hat{Y}^p/\beta^p + T(\hat{D}^p + \hat{Z}^p/\beta^p)).
\]

8: Update $\hat{H}^{p+1}$
\[
\hat{H}^{p+1} = \min_{H} \{\beta \|\hat{X}^p - \hat{H}\|_F^2 + \frac{1}{2}(\|\hat{Y}^p/\beta^p + \hat{A}^H\hat{B}/\beta^p\|_F^2).
\]

9: Until convergence $\|\hat{X}^{p+1} - \hat{X}^p\|_F \leq \varepsilon$ or $p$ reaches the set maximum iteration number.

Output: $\hat{X}^{p+1}$

IV. Experimental Results

In this section, the effectiveness of the proposed LRQR-SR method is demonstrated in comparison with various relevant state-of-the-art methods is shown. Subsection IV-A provides the experimental settings. Subsection IV-B presents the color image recovery results. Finally, the experimental results are discussed in Subsection IV-C.

A. Experimental Settings

1) Comparison Methods: Several relevant existing algorithms were used as comparison algorithms, including D-N and F-N [31], TNNR [3], TNN-SR [14], Q-DNN and Q-FNN [6], LRQA [5], QTNN [21]. The first four of these algorithms are matrix-based, while the last four algorithms are quaternion-based. D-N, F-N, Q-DNN, and Q-FNN use factorization to depict low-rankness, while LRQA is based on the developed QNN to depict low-rankness, and TNNR, TNN-SR, and QTNN utilize a truncated nuclear norm.

2) Test Data and Experimental Environment: Eight benchmark color images as shown in Figure 2, were selected from SIPI Image Database1 and McMaster Dataset to demonstrate the effectiveness of the method. In order to fully demonstrate this effectiveness, 50 color images were also randomly selected from Berkeley Segmentation Dataset (BSD)2 as further test samples. All the experiments were implemented in MATLAB R2019a, on a PC with a 3.00GHz CPU and 8GB RAM.

Fig. 2. The 8 color images (from left to right, and from top to bottom: Tree, Beans, Flower, Vegetable, House, Airplane, Barbara, Splash. They are all size 256 × 256 × 3).

3) Evaluation index setting: The peak signal to noise rate (PSNR) and the structural similarity index (SSIM) were utilized as the relevant indices, and the best numerical results are highlighted in bold font. When processing image recovery with random samples, a larger Sample Rate (SR) value means more observed pixels in a given image.

B. Color Image Recovery

1) Simulations with different parameters: As simulations with different settings of the parameters ($\beta^2$, $\lambda$ truncated number $r$) offer different performance levels, a range of parameters was used to test the performance of the proposed LRQR-SR algorithm, based on recovering random sampled images from Fig. 2.

1http://sipi.usc.edu/database/database.php
2Available: https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/
The influence of different parameter values ($\beta^1 = \{1e - 4, 5e - 4, 1e - 3, 5e - 3, 1e - 2, 5e - 2, 1e - 1, 5e - 1\}$) on the experimental results was first tested with the other parameters fixed ($\lambda = 0.1, r = 30, \rho = 1.01$) and SR = $\{0.5, 0.3, 0.1\}$. The relevant PSNR and SSIM results are plotted in Fig. 3-5, showing that when $\beta^1 \geq 1e - 2$ the recovery effect is minimal. The best recovery results are instead obtained with different degrees of sampling when $\beta^1 = 1e - 4$.

Fig. 3. The PSNR and SSIM values obtained by the proposed LRQR-SR algorithm using different $\beta^1$ with other parameters fixed and SR = 0.5.

Fig. 4. The PSNR and SSIM values obtained by the proposed LRQR-SR algorithm using different $\beta^1$ with other parameters fixed and SR = 0.3.

Fig. 5. The PSNR and SSIM values obtained by the proposed LRQR-SR algorithm using different $\beta^1$ with other parameters fixed and SR = 0.1.

The effect of different parameter values ($\lambda = \{0.01, 0.03, 0.05, 0.07, 0.1, 0.3, 0.5, 0.7, 1\}$) on the recovery results was then tested with other parameters fixed ($\beta^1 = 1e - 4, r = 30, \rho = 1.01$) and SR = $\{0.5, 0.3, 0.1\}$. In Fig. 6-8, the relevant PSNR and SSIM values are given, showing that if the value of $\lambda$ is too large or too small, a better recovery results cannot be obtained. The best recovery results are obtained with different SRs when $\lambda = 0.07$.

Finally, the effect of the number of truncations on the recovery effect was verified. The truncated number $r$ was set as $r = \{10, 20, 30, 40, 50, 60, 70, 80, 90\}$ with other parameters fixed ($\beta^1 = 1e - 4, \lambda = 0.07, \rho = 1.01$) and SR = $\{0.5, 0.3, 0.1\}$. In Fig. 9-11 show the relevant PSNR and SSIM values are given, indicating that the best recovery results were obtained when the degree of sampling was high (SR = 0.5) and the truncated number $r = \{40, 50\}$. However, when the degree of sampling is lower (SR = $\{0.3, 0.1\}$) and $r = \{30, 40\}$, good recovery results are also obtained. This is also consistent with the fact that the more missing pixels in the observed image, the more low-rank constraints are required to improve the recovery effect. Intuitively, when the observed image is missing a lot of pixels, the truncation would contain less useful information.

2) Images recovery with random sample: The LRQR-SR algorithm was compared with several other methods mentioned previously by setting SR = $\{0.3, 0.2, 0.1\}$. The parameters of LRQR-SR were set as $\beta^1 = 1e - 4, \lambda = 0.07, \rho = 1.01$, while the truncation number $r = \{40, 30, 20\}$ was decided by the SR: the lower the SR, the less truncation is required.

Fig. 12 displays the visual comparisons between the designed novel LRQR-SR method and the other methods of comparison for the eight tested color images when SR = 0.2.
LRQR-SR uniformly outperforms its comparators in terms of skill cannot recover the potential images accurately, while I, when the value of SR is very low, utilizing only the truncated with crisp details. As shown in the data presented in TABLE introducing sparsity to the LRQR-SR method. Utilizing low-rankness as prior is insufficient to recover an also inferior to those of TNNR-SR, highlighting that only is thus illustrated by these results. The results for TNNR are effective when operated in the quaternion domain, as in Q-

The PSNR and SSIM results of for recovery as seen in Fig. 2 with SR = \{0.3, 0.2, 0.1\} are presented in Table I. As shown, as compared with other methods, across all SR values, the results obtained by D-N and F-N do not show particularly clear recovery. However, such factorization skills are more effective when operated in the quaternion domain, as in Q-FFN and Q-DNN. The validity of quaternion-based methods is thus illustrated by these results. The results for TNNR are also inferior to those of TNNR-SR, highlighting that only utilizing low-rankness as prior is insufficient to recover an image more accurately. This supports the reasoning behind introducing sparsity to the LRQR-SR method.

In comparison with the other options, the developed LRQR-SR method also provides the most visually optimal results, with crisp details. As shown in the data presented in TABLE I, when the value of SR is very low, utilizing only the truncated skill cannot recover the potential images accurately, while LRQR-SR uniformly outperforms its comparators in terms of both PSNR and SSIM values.

Fig. 14-17 compares the visual results for all the competing completion approaches related to recovering Vegetable, House, Airplane, Barbara under SR = 0.1. The corresponding observed images are shown in Fig. 13. The image in the green box is the zoomed-in image of that in the red box. These images show that restricting recovery only to low rankness may lose some local details. Moreover, the TNNR and QTNN approaches restore the image only roughly when the sample rate is low, with other methods suffering from similar problems. Comparing just the two optimal algorithms (TNN-SR and LRQR-SR), although the effect gap is not visually apparent across the restored images, the corresponding PSNR and SSIM prove the superiority of the proposed method.

3) Images recovery under text mask: The LRQR-SR algorithm was then compared with other methods under a text mask. The parameters of LRQR-SR were set to \(\beta^1 = 1e - 4, \lambda = 0.07, \rho = 1.01\), and the truncated number \(r = 30\).

Fig. 18-21 compares the visual results for all the competing completion approaches related to recovering Tree, Beans, Flower, Splash under text mask. The image in the green box is a zoomed-in image of that in the red box. As shown, the recovered results from D-N and F-N are very blurry, especially where the content of the image is complex. It can also be observed from the zoomed-in portion of Fig. 18 that the results for TNNR, LRQA, Q-FFN, and Q-DNN still leave some obvious artifacts in the blue area. Similar problems can be observed in Fig. 19-21: In Fig. 19, there are some obvious artifacts on the red beans in the zoomed-in portion, while in Fig. 20, some vertical lines remain in the enlarged area after the restoration of the image. In Fig. 21, some further visible blemishes appear on the white part of the zoomed-in portion. In comparison, the TNNR-SR and the proposed LRQR-SR approaches would obtain better performance, and though in general there is not much difference between the two approaches based on image observation, the corresponding PSNR and SSIM results show that the proposed method can restores the image technically more effectively.

4) 50 Images recovery to further demonstrate the effectiveness of our method: A set of 50 images was subject to recovery to further demonstrate the effectiveness of the method. In this simulation, 50 images were randomly selected from BSD as a test sample, though in order to keep the parameters of the comparison algorithms as in the original set, these images were resized to 256 \times 256 \times 3 throughout under random sample (SR=0.25). The corresponding PSNR and SSIM results are reported in Fig. 22 and Fig. 23.

C. Discussion

The simulation experiment results offer various items for discussion that can be summarised as follows:

- The newly-developed LRQR-SR algorithm outperforms comparable existing algorithms both visually and numerically. The main reasons for this can be summarized in three points: this algorithm has been developed in the quaternion domain where the spatial structure information of color image is not destroyed; the model uses QTNN to depict low-rankness, which helps preserve the information contained in the first few large singular values; and,
Fig. 12. The first column is the original image and the second column is the observed image (SR=0.2). The 3-th column to 11-th column are the completion results of TNNR, TNN-SR, D-N, F-N, LRQA, Q-FFN, Q-DNN, QTNN, and LRQR-SR, respectively. The corresponding parameters are summarized in Table I.

finally, the $l_1$ norm is added to act as the regularization in the algorithm, helping to model the sparseness of the underlying quaternion matrix.

- When compared with matrix-based methods, TNNR, D-N, and F-N only depict low-rankness by means of modified NN or low-rank factorization. Hence, the recovery results are generally not satisfactory. However, as TNN-SR is based on both low-rankness and sparse priors, the recovered results are improved. In general, for matrix-based methods, the RGB channels of color images must be processed separately causing the recovered color images to potentially have details omitted.

- When compared with quaternion matrix-based methods, LRQA, Q-FFN, Q-DNN, and QTNN are limited by only utilizing low-rankness. For LRQR-SR, the

Fig. 13. The enlargement of observed images under SR = 0.1, corresponding to Fig. 14-17.

Fig. 14. The recovered results of image Vegetable by different methods under SR = 0.1.
### TABLE I

The PSNR/SSIM values obtained by different recovery algorithms for 8 color images.

| Image   | TNNR [3] | TNN-SR [14] | D-N [31] | F-N [31] | LRQA [5] | Q-FFN [6] | Q-DNN [6] | QTNN | LRRQR-SR |
|---------|----------|-------------|---------|---------|----------|-----------|-----------|------|----------|
| Tree    | 18.517/0.545 | 21.921/0.730 | 17.455/0.433 | 17.551/0.447 | 18.910/0.551 | 18.354/0.515 | 18.554/0.523 | 18.563/0.548 | 23.093/0.769 |
| Beans   | 24.163/0.918 | 21.165/0.974 | 24.846/0.918 | 24.980/0.918 | 25.103/0.924 | 25.018/0.926 | 25.197/0.927 | 25.033/0.930 | 31.804/0.979 |
| Flower  | 20.112/0.736 | 23.590/0.866 | 19.214/0.692 | 19.310/0.698 | 20.442/0.745 | 20.068/0.725 | 20.626/0.719 | 20.465/0.753 | 24.138/0.884 |
| Vegetale| 21.674/0.781 | 25.979/0.891 | 20.299/0.732 | 20.299/0.735 | 21.947/0.791 | 21.902/0.790 | 21.737/0.777 | 22.026/0.796 | 26.802/0.909 |
| House   | 19.232/0.723 | 23.615/0.839 | 18.423/0.677 | 18.397/0.677 | 19.800/0.736 | 18.950/0.710 | 19.464/0.716 | 19.120/0.727 | 24.349/0.855 |
| Airplane| 20.601/0.419 | 24.496/0.600 | 19.641/0.332 | 19.714/0.340 | 20.899/0.421 | 20.401/0.400 | 20.680/0.404 | 20.790/0.427 | 25.002/0.636 |
| Barbara | 19.335/0.602 | 21.487/0.697 | 18.935/0.547 | 19.174/0.535 | 19.503/0.601 | 19.301/0.593 | 18.884/0.567 | 19.522/0.609 | 21.695/0.710 |
| Splash  | 24.285/0.929 | 29.473/0.970 | 23.620/0.919 | 23.689/0.920 | 24.643/0.931 | 24.211/0.929 | 24.643/0.929 | 23.940/0.924 | 29.915/0.973 |

Table of PSNR values for different images and recovery methods.

**Fig. 15.** The recovered results of image *House* by different methods under SR = 0.1.

**Fig. 16.** The recovered results of image *Airplane* by different methods under SR = 0.1.

**Fig. 17.** The recovered results of image *Barbara* by different methods under SR = 0.1.

**Fig. 18.** The recovered results of image *Tree* by different methods under text mask.
also adopted to optimize the model. The experimental results for actual color images illustrated the effectiveness of the resulting LRQR-SR, suggesting that, as the quaternion-based method can exploit correlations among color channels, it may be possible to combine this framework with deep learning methods in future work [32].

ACKNOWLEDGMENTS

This work is supported by the Science and Technology Planning Project of Guangzhou City, China (Grant No. 201907010043), University of Macau (MYRG2019-00039-FST), and the National Natural Science Foundation of China (Grant No. 62173308), and the Natural Science Foundation of Zhejiang Province of China (Grant No. LR20F030001 and D19A010003). The authors would like to express their heartfelt thanks to the health care workers on the front line of the fight against the COVID-19. It is their dedication and sacrifice that provide a safe and stable research environment for people in this special era.

APPENDIX

PROOF OF THEOREM 3

According to [5], we need to prove

$$\min _{X} \|\dot{Y} - X\| _{2} + \lambda \| X \| _{1} \tag{22}$$

has one unique optimal solution \( \dot{X} \), and \( \dot{X} \) equals \( X_{opt} \) defined in (15).

Proof: It can be observed that two terms in (22) are convex, hence, (22) has one unique optimal solution. Based on the rules of quaternion matrix derivatives in [33], \( \dot{X} \) must satisfy the following formula:

$$\hat{\dot{X}} \in \dot{X} - \dot{Y} + 2\lambda \partial \| \dot{X} \| _{1}, \tag{23}$$

where \( \partial \| \dot{X} \| _{1} \) represents the subgradient of \( \| X \| _{1} \). Following [20], the subgradient of the \( l_1 \) norm at \( \dot{X} \) is given by

$$\partial \| \dot{X} \| _{1} = \{ \dot{G} \in \mathbb{R}^{M \times N} : \dot{G} = \text{direct}(\dot{X}_{opt}) + \hat{F}, \| \dot{F} \| _{\infty} \leq 1 \}. \tag{24}$$

where \( \text{direct}(\dot{X}_{opt}) \) is a \( M \times N \) matrix with the entries computed by \( \frac{x_{ij}}{|x_{ij}|} \). Then, \( \dot{X}_{opt} \) need to be proved to satisfy (23). When \( y > 2\lambda, \dot{x} > 0 \), then \( y - \dot{x} = \dot{y} - \frac{x}{y} \), max\{\| \dot{y} - 2\lambda, 0 \} = \dot{y} - (\dot{y} - 2\lambda) = 2\lambda \).

When \( -2\lambda \leq \dot{y} \leq 2\lambda, \dot{x} = 0 \), then \( \dot{x} = \dot{y} \). Let \( \dot{F} = \frac{1}{\lambda} \dot{Y} \), then we have \( \| \dot{F} \| _{\infty} \leq 1 \).

When \( \dot{y} < -2\lambda, \dot{x} < 0 \), then \( \dot{x} = \dot{y} + (\dot{y} - 2\lambda) = -2\lambda = 2\lambda \).

Based on the above discussions, we can obtain that \( \dot{0} \in \dot{X}_{opt} - \dot{Y} + 2\lambda \partial \| \dot{X}_{opt} \| _{1} \), which means that \( \dot{X}_{opt} = \dot{X} \).

REFERENCES

[1] E. J. Candès and B. Recht, “Exact matrix completion via convex optimization,” Found. Comput. Math., vol. 9, no. 6, pp. 717–772, 2009.
[2] S. Gu, Q. Xie, D. Meng, W. Zuo, X. Feng, and L. Zhang, “Weighted nuclear norm minimization and its applications to low level vision,” Int. J. Comput. Vis., vol. 121, no. 2, pp. 183–208, 2017.
[3] Y. Hu, D. Zhang, J. Ye, X. Li, and X. He, “Fast and accurate matrix completion via truncated nuclear norm regularization,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 35, no. 9, pp. 2117–2130, 2013.
Fig. 22. Comparison of the PSNR results of different algorithms for recovering 50 color images selected from BSD (SR = 0.25).

Fig. 23. Comparison of the SSIM results of different algorithms for recovering 50 color images selected from BSD (SR = 0.25).

[4] L. Chen, X. Jiang, X. Liu, and Z. Zhou, “Logarithmic norm regularized low-rank factorization for matrix and tensor completion,” IEEE Trans. Image Process., vol. 30, pp. 3434–3449, 2021.

[5] Y. Chen, X. Xiao, and Y. Zhou, “Low-rank quaternion approximation for color image processing,” IEEE Trans. Image Process., vol. 29, pp. 1426–1439, 2020.

[6] J. Miao and K. I. Kou, “Quaternion-based bilinear factor matrix norm minimization for color image inpainting,” IEEE Trans. Signal Process., vol. 68, pp. 5617–5631, 2020.

[7] Y. Yu, Y. Zhang, and S. Yuan, “Quaternion-based weighted nuclear norm minimization for color image denoising,” Neurocomputing, vol. 332, pp. 283–297, 2019.

[8] X. Wang, C. Wang, H. Yang, and P. Niu, “A robust blind color image watermarking in quaternion fourier transform domain,” J. Syst. Softw., vol. 86, no. 2, pp. 255–277, 2013.

[9] C. Zou, K. I. Kou, and Y. Wang, “Quaternion collaborative and sparse representation with application to color face recognition,” IEEE Trans. Image Process., vol. 25, no. 7, pp. 3287–3302, 2016.

[10] R. S. Cabral, F. D. la Torre, J. P. Costeira, and A. Bernardino, “Unifying nuclear norm and bilinear factorization approaches for low-rank matrix decomposition,” in 2013 IEEE International Conference on Computer Vision, 2013, pp. 2488–2495.

[11] J. Wright, Y. Ma, J. Mairal, G. Sapiro, T. S. Huang, and S. Yan, “Sparse representation for computer vision and pattern recognition,” Proc. IEEE, vol. 98, no. 6, pp. 1031–1044, 2010.

[12] C. Lu, H. Min, J. Gui, L. Zhu, and Y. Lei, “Face recognition via weighted sparse representation,” J. Vis. Commun. Image Represent., vol. 24, no. 2, pp. 111–116, 2013.

[13] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, “Robust face recognition via sparse representation,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 31, no. 2, pp. 210–227, 2009.

[14] J. Dong, Z. Xue, J. Guan, Z. Han, and W. Wang, “Low rank matrix completion using truncated nuclear norm and sparse regularizer,” Signal Process. Image Commun., vol. 68, pp. 76–87, 2018.

[15] J. Xu, L. Zhang, and D. Zhang, “A trilateral weighted sparse coding scheme for real-world image denoising,” in Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part VIII, ser. Lecture Notes in Computer Science, vol. 11212. Springer, 2018, pp. 21–38.

[16] J. Fan and T. W. S. Chow, “Matrix completion by least-square, low-rank, and sparse self-representations,” Pattern Recognit., vol. 71, pp. 290–305, 2017.
[18] Y. Xu, L. Yu, H. Xu, H. Zhang, and T. Nguyen, “Vector sparse representation of color image using quaternion matrix analysis,” *IEEE Trans. Image Process.*, vol. 24, no. 4, pp. 1315–1329, 2015.

[19] Y. Wang, K. I. Kou, C. Zou, and Y. Y. Tang, “Robust sparse representation in quaternion space,” *IEEE Trans. Image Process.*, vol. 30, pp. 3637–3649, 2021.

[20] Z. Jia, M. K. Ng, and G. Song, “Robust quaternion matrix completion with applications to image inpainting,” *Numer. Linear Algebra Appl.*, vol. 26, no. 4, 2019.

[21] L. Yang, K. I. Kou, and J. Miao, “Weighted truncated nuclear norm regularization for low-rank quaternion matrix completion,” *J. Vis. Commun. Image Represent.*, vol. 81, p. 103335, 2021.

[22] S. W. R. H. L. P. F. H. M. R. S. Ed. and D. H. or Corr. M., “Li. on quaternions; or on a new system of imaginaries in algebra,” *Philosophical Magazine Series 3*, vol. 25, no. 163, pp. 10–13, 1844.

[23] N. L. Bihan and J. I. Mars, “Singular value decomposition of quaternion matrices: a new tool for vector-sensor signal processing,” *Signal Process.*, vol. 84, no. 7, pp. 1177–1199, 2004.

[24] F. Zhang, “Quaternions and matrices of quaternions,” *Linear Algebra Appl.*, vol. 251, pp. 21–57, 1997.

[25] N. L. Bihan and S. J. Sangwine, “Quaternion principal component analysis of color images,” in *Proceedings of the 2003 International Conference on Image Processing, ICIP 2003, Barcelona, Catalonia, Spain, September 14-18, 2003*. IEEE, 2003, pp. 809–812.

[26] C. Li, Q. Ma, L. Xiao, M. Li, and A. Zhang, “Image splicing detection based on markov features in QDCT domain,” *Neurocomputing*, vol. 228, pp. 29–36, 2017.

[27] W. Feng and B. Hu, “Quaternion discrete cosine transform and its application in color template matching,” in *2008 Congress on Image and Signal Processing*, vol. 2. IEEE, 2008, pp. 252–256.

[28] L. Li, X. Wang, and G. Wang, “Alternating direction method of multipliers for separable convex optimization of real functions in complex variables,” *Mathematical Problems in Engineering*, vol. 2015, 2015.

[29] M. Bahri, E. S. M. Hitzer, A. Hayashi, and R. Ashino, “An uncertainty principle for quaternion fourier transform,” *Comput. Math. Appl.*, vol. 56, no. 9, pp. 2398–2410, 2008.

[30] E. Hitzer, “Quaternion fourier transform on quaternion fields and generalizations,” *CoRR*, vol. abs/1306.1023, 2013.

[31] F. Shang, J. Cheng, Y. Liu, Z. Luo, and Z. Lin, “Bilinear factor matrix norm minimization for robust PCA: algorithms and applications,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 9, pp. 2066–2080, 2018.

[32] Y. Cao, Y. Fu, Z. Zhu, and Z. Rao, “Color random valued impulse noise removal based on quaternion convolutional attention denoising network,” *IEEE Signal Processing Letters*, 2021.

[33] D. Xu and D. P. Mandic, “The theory of quaternion matrix derivatives,” *IEEE Trans. Signal Process.*, vol. 63, no. 6, pp. 1543–1556, 2015.