

# Vector Sparse Representation of Color Image Using Quaternion Matrix Analysis

Yi Xu, *Member, IEEE*, Licheng Yu, Hongteng Xu, Hao Zhang, and Truong Nguyen, *Fellow, IEEE*

**Abstract**—Traditional sparse image models treat color image pixel as a scalar, which represents color channels separately or concatenate color channels as a monochrome image. In this paper, we propose a vector sparse representation model for color images using quaternion matrix analysis. As a new tool for color image representation, its potential applications in several image-processing tasks are presented, including color image reconstruction, denoising, inpainting, and super-resolution. The proposed model represents the color image as a quaternion matrix, where a quaternion-based dictionary learning algorithm is presented using the K-quaternion singular value decomposition (QSVD) (generalized K-means clustering for QSVD) method. It conducts the sparse basis selection in quaternion space, which uniformly transforms the channel images to an orthogonal color space. In this new color space, it is significant that the inherent color structures can be completely preserved during vector reconstruction. Moreover, the proposed sparse model is more efficient comparing with the current sparse models for image restoration tasks due to lower redundancy between the atoms of different color channels. The experimental results demonstrate that the proposed sparse image model avoids the hue bias issue successfully and shows its potential as a general and powerful tool in color image analysis and processing domain.

**Index Terms**—Vector sparse representation, quaternion matrix analysis, color image, dictionary learning, K-QSVD, image restoration.

Manuscript received April 19, 2014; revised September 5, 2014, December 2, 2014, and January 18, 2015; accepted January 20, 2015. Date of publication January 27, 2015; date of current version February 19, 2015. This work was supported in part by the National Natural Science Foundation of China under Grant 61201384, Grant 61221001, and Grant 61025005, in part by the 111 Project B07022, in part by the National Basic Research Program under Grant 2010CB731401, and in part by the Science and Technology Commission of Shanghai Municipality under Grant 12DZ2272600. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Jong Chul Ye.

Y. Xu is with the Cooperative Medianet Innovation Center, Department of Electrical Engineering, Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: xuyi@sjtu.edu.cn).

L. Yu was with Shanghai Jiao Tong University, Shanghai 200240, China. He is now with the Department of Computer Science, University of North Carolina at Chapel Hill, Chapel Hill, NC 27599 USA (e-mail: licheng@cs.unc.edu).

H. Xu was with Shanghai Jiao Tong University, Shanghai 200240, China. He is now with the Department of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332 USA (e-mail: hxu42@gatech.edu).

H. Zhang was with Shanghai Jiao Tong University, Shanghai 200240, China. He is now with the Department of Computer Science, Colorado State University, Fort Collins, CO 80523 USA (e-mail: zhangh@cs.colostate.edu).

T. Nguyen is with the Department of Electrical and Computer Engineering, University of California at San Diego, La Jolla, CA 92093 USA (e-mail: tqn001@eng.ucsd.edu).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TIP.2015.2397314

## I. INTRODUCTION

THE theory of sparse representation has been proven as an effective model for image representation. Using an overcomplete dictionary that contains a certain number of prototype atoms as its elements, an image signal can be represented as a sparse linear combination of these atoms. The performance of sparse coding relies on the quality of dictionary, which could be chosen as a pre-defined set of bases, such as overcomplete wavelets, curvelets, contourlets, short-time Fourier kernels, unions of bases or raw samples [1], [2]. In recent years, the learning-based strategies for designing dictionary are proposed to represent input signals more sparsely [3]–[5]. Structured low-rank representations for signal classification have attracted much attention of researchers [6]–[8]. Compact and discriminative dictionary is learned to include structure information based on the results of a linear predictive classifier [9], [10]. Meanwhile, the new concepts of block sparsity [11] and group sparsity [12], [13] are defined to get more structural coefficients for different classes. The study of sparse representation has led to many applications in image processing and computer vision areas, such as compression [14], denoising [15], inpainting [1], image classification [16], object detection and recognition [9], [17].

However, there are very few works on the sparse representation model of multichannel signals, which are typically presented as color images. The sparse models in [1], [4], and [15] have achieved the state-of-the-art performance on gray-scale images. As for color images, they just treat RGB channels as three independent “gray-scale” images and process them in a monochrome way. These works completely ignore the inter-relationship among the multiple channels, which is likely to produce hue distortions in the reconstruction results. To avoid color distortions, some works proposed to concatenate RGB channels to alleviate the hue distortion problem [18], where a dictionary is trained to jointly represent the channels. Unfortunately, the operation of concatenation still attains unsatisfying results because of lacking explicit constraints on the correlations among RGB channels. In fact, it is proved that the use of concatenated channel should be restrictive since it contains only a fraction of the unfold matrices which are needed to completely represent a vector-sensor array [19]. Another strategy is using independent color channels rather than RGB channels, e.g., YCbCr and YUV [20], [21] but its application is limited in device-dependent image processing, e.g., spectral interpolation or demosaicing [22].

Since a color image does not represent a scalar, but rather a vector-valued function, it is natural to define a structure

of the chromatic information as vector-valued. Vector-based filtering methods for color images have been proposed in the last two decades, which separated the processing of multichannel signals into directional processing and magnitude processing [23]. Based on vector order statistics, these methods extend their applicability to the primitive color image processing tasks such as color edge detection [24], median filtering and denoising [25] and texture pattern extraction [26]. However, there is a lack of general model and technique for color image analysis and processing.

Fortunately, hypercomplex algebra provides an elegant mathematical tool to deal with vector signals, among which the quaternion algebra was the first hypercomplex number system to be discovered and the closest in its mathematical properties to the familiar systems of the real and complex numbers [27]. Different from the monochromatic-based techniques, which demonstrate only transference of known techniques from gray-level images to color images, the quaternion-based methods process multichannel information in a parallel way, which mimics the human perception of a visual environment. In fact, there have been several color image filtering methods based on quaternion algebra, where a color image pixel is expressed as a quaternion unit and consequently a color image is formulated as a quaternion matrix. These methods explore new solutions of classical problems, e.g., color image registration [28], color image denoising [27], color image watermarking [29], color image super-resolution [30], image colorization [31] and color image segmentation [32], [33]. For example, global and local windowed hypercomplex Fourier transforms (including quaternion Gabor transform) are proposed to provide spectral analysis of color images [34]–[36]. To achieve a more compact spatially spectral analysis, more recently, some researchers have investigated quaternion wavelets [37]–[39]. In these works, the basic concepts of vector operation, i.e., vector correlation/convolution [27], vector projection [40], PCA and SVD analysis of vector-valued image are defined using quaternion algebra [41], [42].

The current quaternion-based color image operations provide a foundation of sparse subspace analysis of color images, which we will explore in this paper. Specifically, we develop a novel vector sparse representation model for color images based on quaternion algebra. In our model, the reconstruction of the color image blocks is conducted as vector operations between the color atoms in the learned quaternion dictionary and the sparse quaternion coefficients. We propose the corresponding dictionary learning method called K-QSVD (Generalized K-means clustering for Quaternion Singular Value Decomposition). K-QSVD conducts the sparse basis selection during quaternion dictionary learning step and computes the sparse coefficient vectors using QOMP (quaternion orthogonal matching pursuit) method. In essence, the quaternion dictionary uniformly transforms the channel images to a subspace, where the redundancy between channels is removed and consequently the inherent color structures can be completely preserved during sparse reconstruction.

Differing from traditional separated and concatenated monochrome models, which consider only a fraction of the subspaces that are needed for completely representing

a vector-sensor array, the quaternion-based model can preserve the whole information of a 3D vector array. Furthermore, comparing to the tensor-based model, the quaternion-based model not only preserves the correlation among channels but also the orthogonal property for the coefficients of different channels, which achieves a structured representation. Experiments prove that the proposed sparse model is more efficient comparing to current sparse models for image restoration tasks.

The remainder of this paper is organized as follows. Section II introduces the basic concepts of quaternion algebra. Based on these concepts, we conduct the subspace analysis of color images using Quaternion Matrix Singular Value Decomposition (QSVD) in a comparison with Singular Value Decomposition (SVD) and Tensor-based SVD (T-SVD). Section III proposes our quaternion-based sparse representation model. The comparison with two typical sparse models of color images is also provided. Section IV designs a K-QSVD based dictionary learning method. Section V presents the applications of the proposed model and the comparison with the state-of-the-art methods. Finally, Section VI summarizes our work.

## II. THE BASIC CONCEPTS OF QUATERNION ALGEBRA

In this paper, scalar variables are defined using lowercase letter, e.g.,  $a \in \mathbb{R}$ , scalar vectors using bold types, e.g.,  $\mathbf{a}$ , and scalar matrices using bold capital letter, e.g.,  $\mathbf{I}$ . For the quaternion system, a dot (above the variable) is used to denote a quaternion variable, that is  $\dot{a} \in \mathbb{H}$ . Accordingly, a quaternion vector is denoted as  $\dot{\mathbf{a}}$  and a quaternion matrix is indicated as  $\dot{\mathbf{I}}$ . In this section, we summarize the basic concepts of quaternion algebra, where a more complete introduction of quaternion algebra can be referred to [43].

### A. Definition of Quaternion Algebra

Quaternion was first introduced by W. Hamilton [44] in 1832. Let  $\dot{a} \in \mathbb{H}$  be a quaternion, then

$$\dot{a} = a_0 + a_1i + a_2j + a_3k, \quad (1)$$

where  $a_l \in \mathbb{R}, l = 0, 1, 2, 3$ , and the imaginary units  $i, j, k$  obey the quaternion rules that  $i^2 = j^2 = k^2 = ijk = -1$ . As a vector entity, the quaternion is associative but non-commutative and its algebra can simultaneously manipulate all its four components. Let  $\dot{a}, \dot{b} \in \mathbb{H}, \lambda \in \mathbb{R}$ . Here we give some fundamental algebraic operations used in our work briefly, which follow the definition in [43], [45], and [46]. Readers can find more details on quaternion algebra in the references.

#### 1) Addition:

$$\dot{a} + \dot{b} = (a_0 + b_0) + (a_1 + b_1)i + (a_2 + b_2)j + (a_3 + b_3)k. \quad (2)$$

#### 2) Multiplication:

$$\lambda \dot{a} = (\lambda a_0) + (\lambda a_1)i + (\lambda a_2)j + (\lambda a_3)k. \quad (3)$$

$$\begin{aligned} \dot{a}\dot{b} = & (a_0b_0 - a_1b_1 - a_2b_2 - a_3b_3) \\ & + (a_0b_1 + a_1b_0 + a_2b_3 - a_3b_2)i \\ & + (a_0b_2 - a_1b_3 + a_2b_0 + a_3b_1)j \\ & + (a_0b_3 + a_1b_2 - a_2b_1 + a_3b_0)k. \end{aligned} \quad (4)$$

We formulate  $\dot{a}$  and  $\dot{b}$  as the composite of a scalar part and a vector part<sup>1</sup> by writing  $\dot{a} = (a_0, a_1, a_2, a_3) = [S(\dot{a}), V(\dot{a})]$ , where  $S(\dot{a}) = a_0$  and  $V(\dot{a}) = \{a_1, a_2, a_3\}$ . Similarly,  $\dot{b} = (b_0, b_1, b_2, b_3) = [S(\dot{b}), V(\dot{b})]$ . Then we have,

$$S(\dot{a}\dot{b}) = S(\dot{a})S(\dot{b}) - V(\dot{a}) \circ V(\dot{b}) \quad (5)$$

$$V(\dot{a}\dot{b}) = S(\dot{a})V(\dot{b}) + S(\dot{b})V(\dot{a}) + V(\dot{a}) \otimes V(\dot{b}) \quad (6)$$

Here ‘ $\circ$ ’ denotes dot product operator and ‘ $\otimes$ ’ denotes cross product operator of two vectors. The multiplication between two pure quaternions, i.e.,  $a_0 = b_0 = 0$ , is reduced to  $S(\dot{a}\dot{b}) = -V(\dot{a}) \circ V(\dot{b})$  and  $V(\dot{a}\dot{b}) = V(\dot{a}) \otimes V(\dot{b})$ .

3) *Norm, Conjugation, Unity and Reciprocal:*

$$\|\dot{a}\| = \sqrt{\dot{a}\bar{\dot{a}}} = \sqrt{a_0^2 + a_1^2 + a_2^2 + a_3^2} \quad (7)$$

where  $\bar{\dot{a}}$  is the conjugate of  $\dot{a}$  and has the form of

$$\bar{\dot{a}} = a_0 - a_1i - a_2j - a_3k. \quad (8)$$

$\dot{a}$  is called a unit quaternion if its norm is 1. The reciprocal of a quaternion is

$$\dot{a}^{-1} = \frac{\bar{\dot{a}}}{\|\dot{a}\|^2}. \quad (9)$$

4) *Vector Representation of Quaternion:* In this paper, similar to the vector of real number, the vector of quaternion is denoted as  $\dot{\mathbf{a}} = [\dot{a}_1, \dots, \dot{a}_N]^T \in \mathbb{H}^N$ , where each element is a quaternion. Furthermore, we can also define the inner product of two quaternion vectors  $\dot{\mathbf{a}}, \dot{\mathbf{b}}$  as

$$\langle \dot{\mathbf{a}}, \dot{\mathbf{b}} \rangle = \dot{\mathbf{a}}^H \dot{\mathbf{b}} = \sum_{n=1}^N \bar{\dot{a}}_n \dot{b}_n, \quad (10)$$

which is still a quaternion. Here  $\dot{\mathbf{a}}^H = [\bar{\dot{a}}_1, \dots, \bar{\dot{a}}_N]$  is the conjugate transpose of  $\dot{\mathbf{a}}$ . The norm of quaternion vector is defined as

$$\|\dot{\mathbf{a}}\| = \sqrt{\langle \dot{\mathbf{a}}, \dot{\mathbf{a}} \rangle} \quad (11)$$

$\dot{\mathbf{a}}, \dot{\mathbf{b}}$  are orthogonal if and only if  $\langle \dot{\mathbf{a}}, \dot{\mathbf{b}} \rangle = \dot{0}$ . Similarly, the matrix of quaternion is denoted as  $\dot{\mathbf{A}} = [\dot{\mathbf{a}}_1, \dots, \dot{\mathbf{a}}_M] \in \mathbb{H}^{N \times M}$ ,  $\dot{\mathbf{a}}_m \in \mathbb{H}^N$ . Given  $\dot{\mathbf{A}} \in \mathbb{H}^{N \times M}$  and  $\dot{\mathbf{B}} = [\dot{\mathbf{b}}_1, \dots, \dot{\mathbf{b}}_K] \in \mathbb{H}^{N \times K}$ , their product  $\dot{\mathbf{C}} = \dot{\mathbf{A}}^H \dot{\mathbf{B}}$ , where each element of  $\dot{\mathbf{C}}$ ,  $\dot{c}_{mk} = \langle \dot{\mathbf{a}}_m, \dot{\mathbf{b}}_k \rangle$ . The norm of matrix  $\|\dot{\mathbf{C}}\| = \sqrt{\text{tr}(\dot{\mathbf{C}}^H \dot{\mathbf{C}})}$ , where  $\text{tr}(\cdot)$  is the trace of matrix. Following the notations of real vector and matrix, we write  $\|\dot{\mathbf{a}}\|$ ,  $\|\dot{\mathbf{C}}\|$  as  $\|\dot{\mathbf{a}}\|_2$ ,  $\|\dot{\mathbf{C}}\|_F$  in the following sections.

<sup>1</sup>It should be note that the vector part of a quaternion is a bivector or pseudovector [45]. In this paper, we simplified the use of bivector as vector for two reasons: (1) We would like to emphasize that color images are reconstructed as vector signals. (2) The reconstruction can be formulated as the multiplication of the learned quaternion dictionary and the sparse coefficient matrix, which is operated as element-wise quaternion multiplication. In calculations, quaternion multiplication can be conducted as a series of vector operations in vector space.

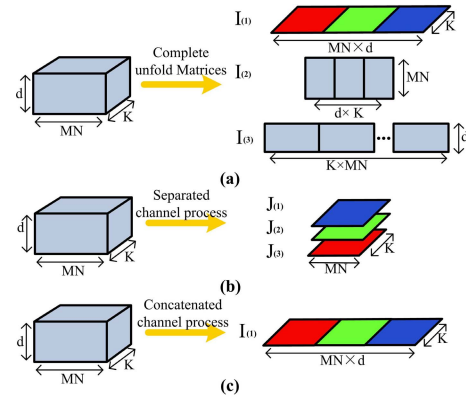


Fig. 1. Description of a 3D array using (a) complete unfold matrices, (b) separated channel process and (c) concatenated channel process.

5) *Cross Correlation:* As for two images represented as quaternion matrices  $\dot{\mathbf{I}}_1 \in \mathbb{H}^{M \times N}$  and  $\dot{\mathbf{I}}_2 \in \mathbb{H}^{M \times N}$ , their cross correlation  $C(m, n)$  is defined as,

$$C(m, n) = \sum_{p=0}^{M-1} \sum_{q=0}^{N-1} \dot{\mathbf{I}}_1(p, q) \overline{\dot{\mathbf{I}}_2(p-m, q-n)}, \quad (12)$$

where  $(p, q)$  is the row and column index of  $\dot{\mathbf{I}}_1$  and  $\dot{\mathbf{I}}_2$ . The shift operation on  $\dot{\mathbf{I}}_2$  is implemented cyclically using modulo arithmetic [47]. If  $\dot{\mathbf{I}}_1 = \dot{\mathbf{I}}_2$ , the autocorrelation of these two images is computed. If the mean value of each image is subtracted first, the cross-covariance is obtained.

In order to recover classical matrix calculus rules, in this paper, we choose the convention that matrices operate on the left side, and variables operate on the right side.

## B. Linear Subspace Analysis of Color Images Using 2D-Matrix Singular Value Decomposition (SVD)

Corresponding to the sparse representation of a group of color image patches, we consider a collection of  $K$  samples in the subspace analysis with each sample stacked into a long vector with dimension of  $MN$ .

We can represent these  $K$  sample image patches as a real 3D-array  $\mathbf{I}$ , i.e.,  $\mathbf{I} \in \mathbb{R}^{MN \times d \times K}$ , where  $d = 3$  means that three color channels are involved as the common case. As shown in Fig. 1(a), the definition of the unfolding matrices that needed to completely visualize the rank of array  $\mathbf{I}$  is [19],

$$\mathbf{I}_{(1)} = [a_{s,t}^{(1)}]_{3MN \times K}, \quad \text{where } a_{s,t}^{(1)} \in \mathbb{R}, \quad (13)$$

$$\mathbf{I}_{(2)} = [a_{s,t}^{(2)}]_{3K \times MN}, \quad \text{where } a_{s,t}^{(2)} \in \mathbb{R}, \quad (14)$$

$$\mathbf{I}_{(3)} = [a_{s,t}^{(3)}]_{KMN \times 3}, \quad \text{where } a_{s,t}^{(3)} \in \mathbb{R}. \quad (15)$$

Derived from Tucker3 model of a  $N$ -dimensional array [48], a 3D array rank is defined by the ranks of the three unfolding matrices. Accordingly, the singular value decomposition of array  $\mathbf{I}$  is given by,

$$\mathbf{I} = \mathbf{c} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \times_3 \mathbf{U}^{(3)}, \quad (16)$$

where  $\mathbf{c}$  is a core array,  $\mathbf{U}^{(i)}$  are the left eigen-matrices of the unfolding matrices  $\mathbf{I}_{(i)}$ , and operator  $\times_n$  is the  $n$ -mode product operator.

As shown in Fig. 1(b), however traditional monochromatic process treats this real 3D-array as three uncorrelated 2D real matrices,

$$\mathbf{J}(1) = [b_{(s,t)}^{(1)}]_{MN \times K}, \quad \text{where } b_{s,t}^{(1)} \in \mathbb{R}, \quad (17)$$

$$\mathbf{J}(2) = [b_{(s,t)}^{(2)}]_{MN \times K}, \quad \text{where } b_{s,t}^{(2)} \in \mathbb{R}, \quad (18)$$

$$\mathbf{J}(3) = [b_{(s,t)}^{(3)}]_{MN \times K}, \quad \text{where } b_{s,t}^{(3)} \in \mathbb{R}, \quad (19)$$

The subspace analysis is obtained using independent SVD decomposition for these three matrices. Consequently, the order of array  $\mathbf{I}$  is degenerated from 3 to 2, losing the interrelationship between channels. As for the concatenation process shown in Fig. 1(c), it only considers  $I_{(1)}$  for subspace analysis, which contains fractional rank of the unfolding matrices listed in (13-15).

### C. Linear Subspace Analysis of Color Images Using 2D-Matrix Quaternion Singular Value Decomposition (QSVD)

In this subsection, a color image pixel is encoded as a pure quaternion unit, e.g.,  $\hat{a} = r \cdot i + g \cdot j + b \cdot k$ , where the coefficients  $r, g, b$  of the imaginary parts are three color channel values and the scalar part equals to 0. Then we can rewrite the real 3D-array  $\mathbf{I}$  as a 2D quaternion matrix  $\hat{\mathbf{I}}$ , i.e.,  $\hat{\mathbf{I}} \in \mathbb{H}^{MN \times K}$  with the color dimension  $d$  implied in each quaternion element. As the extension of SVD from real/complex domain to quaternion space, QSVD allows to identify the embedded linear subspaces of quaternion matrix  $\hat{\mathbf{I}}$ .

Using the Cayley-Dickson notation [42], we obtain  $\hat{\mathbf{I}} = \mathbf{A} + \mathbf{B} \cdot j$ , where  $\mathbf{A}, \mathbf{B} \in \mathbb{C}^{MN \times K}$  are two complex matrices. Then  $\hat{\mathbf{I}}$  can be converted into an equivalent complex matrix as  $\mathbf{I}_c = \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ -\mathbf{B} & \mathbf{A} \end{bmatrix}_{2MN \times 2K}$ . Using the isomorphism between  $\mathbb{H}^{MN \times K}$  and  $\mathbb{C}^{2MN \times 2K}$ , the QSVD of  $\hat{\mathbf{I}}$  can be obtained by firstly applying the classical complex SVD algorithm to  $\mathbf{I}_c$ .

We denote the singular values as  $\Lambda'$ , then get  $\mathbf{I}_c = \mathbf{U}\Lambda'\mathbf{V}^H$ , where subscript ' $H$ ' denotes Hermitian transpose operator,  $\mathbf{U}, \mathbf{V}$  are two complex matrices and the columns of  $\mathbf{U}, \mathbf{V}$  are composed of orthonormal eigenvectors of  $\mathbf{I}_c \cdot \mathbf{I}_c^H$  and  $\mathbf{I}_c^H \cdot \mathbf{I}_c$ , respectively. The relation between the QSVD of quaternion matrix  $\hat{\mathbf{I}}$  and the SVD of its equivalent complex matrix  $\mathbf{I}_c$  is defined as follows,

$$\Lambda = \text{row}_{\text{odd}}(\text{col}_{\text{odd}}(\Lambda')), \quad (20)$$

$$\hat{\mathbf{U}} = \text{col}_{\text{odd}}(\mathbf{U}^1) + \text{col}_{\text{odd}}(-\overline{\mathbf{U}^2}) \cdot j, \quad (21)$$

$$\hat{\mathbf{V}} = \text{col}_{\text{odd}}(\mathbf{V}^1) + \text{col}_{\text{odd}}(-\overline{\mathbf{V}^2}) \cdot j, \quad (22)$$

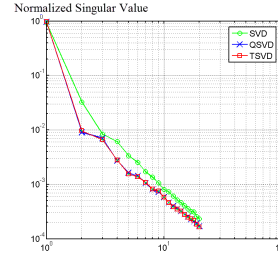
such that  $\hat{\mathbf{I}} = \hat{\mathbf{U}}\Lambda\hat{\mathbf{V}}^H$ , where

$$\mathbf{U} = \begin{bmatrix} [\mathbf{U}^1]_{MN \times 2MN} \\ [\mathbf{U}^2]_{MN \times 2MN} \end{bmatrix}_{2MN \times 2MN}, \quad \mathbf{V} = \begin{bmatrix} [\mathbf{V}^1]_{K \times 2K} \\ [\mathbf{V}^2]_{K \times 2K} \end{bmatrix}_{2K \times 2K},$$

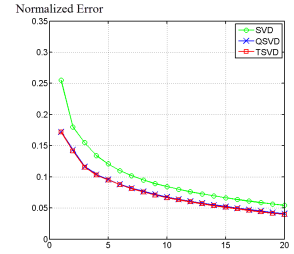
and  $\text{row}_{\text{odd}}(\mathbf{P}), \text{col}_{\text{odd}}(\mathbf{P})$  extracts the odd rows and odd columns of matrix  $\mathbf{P}$  respectively. Since eigenvalues of equivalent complex matrix  $\mathbf{I}_c$  appear by conjugate pairs along diagonal, and as  $\mathbf{I}_c \cdot \mathbf{I}_c^H$  is Hermitian, its eigenvalues are real and appear in pairs along the diagonal. Consequently, the singular values consisting of  $\Lambda$  are all real numbers. Based on QSVD  $\hat{\mathbf{I}} = \hat{\mathbf{U}}\Lambda\hat{\mathbf{V}}^H$ , we can further define the inverse



(a)



(b)



(c)

Fig. 2. Comparison of SVD, T-SVD and QSVD in data approximation. (a) Image patch dataset. (b) Plots of information distributions of SVD, T-SVD and QSVD. (c) Plots of rank- $\alpha$  approximation errors of SVD, T-SVD and QSVD.

of quaternion matrix as  $\hat{\mathbf{I}}^{-1} = \hat{\mathbf{V}}\Lambda^{-1}\hat{\mathbf{U}}^H$ , where  $\Lambda^{-1}$  is the inverse of  $\Lambda$  (If  $\Lambda$  is not a full rank matrix,  $\Lambda^{-1}$  is obtained by computing the reciprocal of non-zero elements of  $\Lambda$ ).

### D. The Rank- $\alpha$ Approximation of Color Images-QSVD v.s. SVD and Tensor-SVD

Before we provide the sparse representation model, it is informative to compare the performance of QSVD in data approximation with SVD and Tensor-based SVD (T-SVD) [49]. Since the distribution of the singular values implies information distribution of basis, we design an experiment to compare the distribution of singular values for three methods.<sup>2</sup> The input data consists of 100 color image patches with the size of  $8 \times 8$  pixels. These patches are randomly selected from 10 images shown in Fig. 2(a). For the standard SVD of monochromatic image, we first reshape each image patch from one channel into a  $64 \times 1$  vector of real numbers. These color channels are concatenated to form a  $192 \times 1$  vector of real numbers. With 100 image patches, the input matrix is then  $192 \times 100$  dimensional. For T-SVD, the input matrix is  $64 \times 100 \times 3$  dimensional. For QSVD, each image patch is reshaped as a  $64 \times 1$  quaternion vector. With 100 image patches, the input matrix for QSVD is  $64 \times 100$  dimensional. The information distributions (normalized singular values) of these image patches obtained by the three methods are illustrated in Fig. 2(b) by logarithmic scale, where x-axis is the rank related to each singular value and y-axis is the normalized singular value. On one hand, we can observe that the singular value obtained by QSVD decreases much faster than that of SVD, indicating more information is

<sup>2</sup>The singular values obtained by T-SVD are represented as a tensor, whose three frontal slices are diagonal matrices. The diagonal elements of the slices (matrices) in the same location formulate a singular value vector, and the singular value of T-SVD in this paper is the  $L_2$ -norm of the singular value vector.

contained in the tight basis of  $\dot{\mathbf{U}}$  and  $\dot{\mathbf{V}}$ . On the other hand, QSVD presents nearly the same performance as T-SVD, showing its strong potential in representing 3D arrays without information loss. For a more direct comparison of the best rank- $\alpha$  approximation, Fig. 2(c) provides three reconstruction error distributions with the eigenvectors related to the top  $\alpha$  ranks, where x-axis is the order of the rank and y-axis is the normalized reconstruction error (The Frobenius norm of reconstruction residual image divides the Frobenius norm of original image). In terms of minimizing the distance in arbitrary unitarily invariant norms, T-SVD is proven to be the optimal in the usual SVD-based reconstruction. Similar to the former experiments, we observe that QSVD consistently provides a more accurate approximation than SVD using the reconstructed matrix of the same rank, and achieves the same performance as T-SVD.

The comparison results prove that the channel-by-channel process and the concatenation process lose the interrelationship of color channels while the vector-based process i.e., quaternion-based and tensor-based processes, can completely preserve the interrelationship of color channels, achieving better approximation of color images. It should be noted that although QSVD seems to be equivalent to T-SVD in the analytic experiments of SVD-based reconstruction, its superiority arises when combining with sparse representation model. In the next two sections, we will show that with the help of QSVD, we can obtain a structured sparse representation model and an effective dictionary learning algorithm for color images.

### III. QUATERNION-BASED SPARSE REPRESENTATION MODEL FOR COLOR IMAGES

Current image sparse representation mainly focuses on patch processing. Given a color image  $\mathbf{I}$  and its overlapped image patches, we stack all the pixels of one patch in each channel and denote the generated scalar vector as  $\mathbf{y}_c \in \mathbb{R}^n$ , where  $n$  is the dimension of the patch vector, the subscript  $c = r, g, b$  represents the RGB channels respectively. As aforementioned, current sparse models of color image deal with each single channel independently with possibly different dictionaries, which can be denoted as

$$\mathbf{y}_c = \mathbf{D}_c \mathbf{a}_c, \quad c = r, g, b, \quad (23)$$

where  $\mathbf{D}_c$  is the dictionary with  $K$  atoms, i.e.,  $\mathbf{D}_c \in \mathbb{R}^{n \times K}$  and  $\mathbf{a}_c$  is the sparse coefficient vector,  $\mathbf{a}_c \in \mathbb{R}^K$ . This **monochromatic model**, however, fails to capture the interrelationship among the three color channels.

A moderate improvement is to process the concatenation of the three channels [18]. The corresponding representation model can be expressed as

$$[\mathbf{y}_r^T, \mathbf{y}_g^T, \mathbf{y}_b^T]^T = [\mathbf{D}_r^T, \mathbf{D}_g^T, \mathbf{D}_b^T]^T \mathbf{a}. \quad (24)$$

The **concatenation model** in (24) adds a constraint on the coefficient so that the coefficient vector  $\mathbf{a}$  ( $\mathbf{a} \in \mathbb{R}^K$ ) should not only be sparse but also be shared by each channel. It pays much emphasis on the spatial structure preservation at the cost of color fidelity.

Another strategy is **tensor-based sparse model**. Here, each patch is represented by a tensor  $\mathbf{y} \in \mathbb{R}^{n \times 1 \times 3}$ , which is equal to

the product of dictionary tensor  $\mathbf{D} \in \mathbb{R}^{n \times K \times 3}$  and coefficient tensor  $\mathbf{a} \in \mathbb{R}^{K \times 1 \times 3}$  as

$$\mathbf{y} = \mathbf{D} \times \mathbf{a}. \quad (25)$$

Differing from the methods above, in this paper we propose a **quaternion-based sparse representation model**. Applying the pure quaternion form, we denote the vector form of one RGB color image patch as  $\dot{\mathbf{y}} = 0 + \mathbf{y}_r i + \mathbf{y}_g j + \mathbf{y}_b k$ ,  $\dot{\mathbf{y}} \in \mathbb{H}^n$ . Accordingly, the dictionary and the corresponding coefficient are represented as  $\dot{\mathbf{D}} = \mathbf{D}_s + \mathbf{D}_r i + \mathbf{D}_g j + \mathbf{D}_b k$  and  $\dot{\mathbf{a}} = \mathbf{a}_0 + \mathbf{a}_1 i + \mathbf{a}_2 j + \mathbf{a}_3 k$  respectively. Then, we propose the quaternion-based sparse representation model as

$$\min_{\dot{\mathbf{a}}} \|\dot{\mathbf{a}}\|_0, \quad \text{s.t. } \dot{\mathbf{y}} = \dot{\mathbf{D}} \dot{\mathbf{a}}, \quad (26)$$

where  $\dot{\mathbf{D}} \in \mathbb{H}^{n \times K}$  is a quaternion dictionary consisting of  $K$  pure quaternion atoms,  $\dot{\mathbf{a}} \in \mathbb{H}^K$  is a sparse quaternion coefficient vector corresponding to the input data  $\dot{\mathbf{y}} \in \mathbb{H}^n$ , with its components  $\mathbf{a}_0, \mathbf{a}_1, \mathbf{a}_2$  and  $\mathbf{a}_3 \in \mathbb{R}^K$ . The objective function  $\|\dot{\mathbf{a}}\|_0$  counts the number of nonzero components in the quaternion coefficient vector.

These four color image representation models can be unified as a more generalized one,

$$[\mathbf{0}, \mathbf{y}_r, \mathbf{y}_g, \mathbf{y}_b] = [\mathbf{D}_s, \mathbf{D}_r, \mathbf{D}_g, \mathbf{D}_b] \tilde{\mathbf{a}}. \quad (27)$$

The monochromatic model in (23) and the concatenation model in (24) can be considered as a special case when we obtain coefficient vector from a particular set

$$\tilde{\mathbf{a}} = [[\mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{0}]^T, [\mathbf{0}, \mathbf{a}_r^T, \mathbf{0}, \mathbf{0}]^T, [\mathbf{0}, \mathbf{0}, \mathbf{a}_g^T, \mathbf{0}]^T, [\mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{a}_b^T]^T], \quad (28)$$

where  $\tilde{\mathbf{a}} \in \mathbb{R}^{4K \times 4}$ . The only difference is the concatenation model adds a constraint to (28) with  $\mathbf{a}_r = \mathbf{a}_g = \mathbf{a}_b$ . In (28), the coefficient vector of three color channels is orthogonal to each other. Although this property leads to a structured coefficient matrix (the columns of coefficient matrix are orthogonal), it implies that the interrelationship between the channels must be encoded in the dictionary. However, in the monochromatic model, dictionaries are learned independently, providing no assurance of channel correlation in the reconstruction. In the concatenation model, Elad *et al.* [18] introduced an extra strong constraint to guarantee that the reconstructed patch will maintain the average color of the original one, which tends to contain many gray or low chrominance atoms. In other words, the interrelationship among color channels is not well preserved in each atom during the training process.

On the other hand, rewriting the tensor-based model (25), we have

$$\mathbf{y}_r = \mathbf{D}_r \mathbf{a}_1 + \mathbf{D}_g \mathbf{a}_3 + \mathbf{D}_b \mathbf{a}_2, \quad (29)$$

$$\mathbf{y}_g = \mathbf{D}_r \mathbf{a}_2 + \mathbf{D}_g \mathbf{a}_1 + \mathbf{D}_b \mathbf{a}_3, \quad (30)$$

$$\mathbf{y}_b = \mathbf{D}_r \mathbf{a}_3 + \mathbf{D}_g \mathbf{a}_2 + \mathbf{D}_b \mathbf{a}_1, \quad (31)$$

Here  $\mathbf{D}_s = \mathbf{0}$ , which is ignored. Then we formulate (29-31) into the form of (27) to obtain

$$\tilde{\mathbf{a}} = [\mathbf{a}_s \quad \mathbf{a}_r \quad \mathbf{a}_g \quad \mathbf{a}_b] = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{a}_1 & \mathbf{a}_3 & \mathbf{a}_2 \\ \mathbf{0} & \mathbf{a}_2 & \mathbf{a}_1 & \mathbf{a}_3 \\ \mathbf{0} & \mathbf{a}_3 & \mathbf{a}_2 & \mathbf{a}_1 \end{bmatrix}. \quad (32)$$

In this model, we also obtain a structured coefficient matrix, which preserves the correlation among color channels. However, in this model, the coefficient vectors of three color channels are no longer orthogonal to each other.

According to the analysis above, we can find that the monochromatic model and the concatenation model guarantee the orthogonal property of coefficient matrix while lose the correlation among color channels. On the contrary, the tensor-based model preserves the correlation while loses the orthogonal property. Differing from these competitors, the proposed quaternion-based sparse model in (26) can be expanded as follows,

$$0 = \mathbf{D}_s \mathbf{a}_0 - \mathbf{D}_r \mathbf{a}_1 - \mathbf{D}_g \mathbf{a}_2 - \mathbf{D}_b \mathbf{a}_3, \quad (33)$$

$$\mathbf{y}_r = \mathbf{D}_s \mathbf{a}_1 + \mathbf{D}_r \mathbf{a}_0 + \mathbf{D}_g \mathbf{a}_3 - \mathbf{D}_b \mathbf{a}_2, \quad (34)$$

$$\mathbf{y}_g = \mathbf{D}_s \mathbf{a}_2 - \mathbf{D}_r \mathbf{a}_3 + \mathbf{D}_g \mathbf{a}_0 + \mathbf{D}_b \mathbf{a}_1, \quad (35)$$

$$\mathbf{y}_b = \mathbf{D}_s \mathbf{a}_3 + \mathbf{D}_r \mathbf{a}_2 - \mathbf{D}_g \mathbf{a}_1 + \mathbf{D}_b \mathbf{a}_0, \quad (36)$$

Then we formulate (33-36) into the form of (27) to obtain

$$\tilde{\mathbf{a}} = [\mathbf{a}_s \ \mathbf{a}_r \ \mathbf{a}_g \ \mathbf{a}_b] = \begin{bmatrix} \mathbf{a}_0 & \mathbf{a}_1 & \mathbf{a}_2 & \mathbf{a}_3 \\ -\mathbf{a}_1 & \mathbf{a}_0 & -\mathbf{a}_3 & \mathbf{a}_2 \\ -\mathbf{a}_2 & \mathbf{a}_3 & \mathbf{a}_0 & -\mathbf{a}_1 \\ -\mathbf{a}_3 & -\mathbf{a}_2 & \mathbf{a}_1 & \mathbf{a}_0 \end{bmatrix}. \quad (37)$$

Compared with the other three models, the advantages of quaternion-based sparse model in (26) for color image can be summarized as follows:

- The coefficient matrix preserves both the correlation among channels and the orthogonal property. Rather than selecting atoms from three independent channel dictionaries, each color channel is correlated linearly with four channel dictionaries. By training the quaternion dictionary  $\hat{\mathbf{D}}$  in a proper way, the interrelationship of the three channels for color patches  $\hat{\mathbf{y}}$  can be well preserved.
- It imposes explicit linear correlation among the four channel dictionaries as shown in (33). The coefficients  $[\mathbf{a}_0, -\mathbf{a}_1, -\mathbf{a}_2, -\mathbf{a}_3]$  is the null space of  $[\mathbf{D}_s, \mathbf{D}_r, \mathbf{D}_g, \mathbf{D}_b]$ . Therefore the correlation among channel dictionaries is described by  $\mathbf{a}_0, \mathbf{a}_1, \mathbf{a}_2$  and  $\mathbf{a}_3$ . Such correlations among color channels have been proven to be useful in color constancy [50].

Consequently, the four channel dictionaries uniformly transform  $\mathbf{y}_r, \mathbf{y}_g$  and  $\mathbf{y}_b$  into an orthogonal color space. In this color space, it is significant that the inherent color structure can be completely preserved during image channel reconstruction.

#### IV. QUATERNION-BASED DICTIONARY TRAINING

##### A. Single Dictionary Training

The single quaternion-based dictionary training process is an extension of the model in (26), in which both the dictionary and coefficients are unknown variables. This process can be formulated as

$$\{\hat{\mathbf{D}}, \hat{\mathbf{A}}\} = \underset{\mathbf{D}, \mathbf{A}}{\operatorname{argmin}} \|\hat{\mathbf{Y}} - \mathbf{D}\hat{\mathbf{A}}\|_F^2 + \lambda \|\hat{\mathbf{A}}\|_0, \quad (38)$$

where  $\hat{\mathbf{Y}} = \{\hat{\mathbf{y}}_i, 1 \leq i \leq N\}$  is the set of the sample image patches and  $\hat{\mathbf{Y}} \in \mathbb{H}^{n \times N}$ ,  $\hat{\mathbf{D}} = \{\hat{\mathbf{d}}_i, 1 \leq i \leq K\}$  is the quaternion-based dictionary composed of  $K$  atoms and  $\hat{\mathbf{D}} \in \mathbb{H}^{n \times K}$ ,  $\hat{\mathbf{A}} = \{\hat{\mathbf{a}}_i, 1 \leq i \leq N\}$  is the coefficient

TABLE I  
QUATERNION ORTHOGONAL MATCHING PURSUIT

<p><b>1. Initialization:</b> Residual <math>\hat{\varepsilon}^{(0)} = \hat{\mathbf{y}}</math>, atom set <math>\hat{\mathbf{S}}^{(0)} = \emptyset</math>.</p> <p><b>2. For <math>j=1 \sim L</math>:</b></p> <p>1) For every <math>\hat{\mathbf{d}}_m \in \hat{\mathbf{D}} \setminus \hat{\mathbf{S}}^{(j-1)}</math>, compute correlation:</p> $C_m^{(j)} = \langle \hat{\mathbf{d}}_m, \hat{\varepsilon}^{(j-1)} \rangle = \hat{\mathbf{d}}_m^H \hat{\varepsilon}^{(j-1)}.$ <p>2) Atom selection: <math>m^{(j)} = \operatorname{argmax}_m \ C_m^{(j)}\ </math>, <math>\hat{\mathbf{d}}^{(j)} = [\hat{\mathbf{D}} \setminus \hat{\mathbf{S}}^{(j-1)}]_{m^{(j)}}</math>,</p> $\hat{\mathbf{S}}^{(j)} = \hat{\mathbf{S}}^{(j-1)} \cup \hat{\mathbf{d}}^{(j)}$ <p>3) Compute coefficients:</p> $\hat{\mathbf{a}}^{(j)} = ((\hat{\mathbf{S}}^{(j)})^H \hat{\mathbf{S}}^{(j)})^{-1} (\hat{\mathbf{S}}^{(j)})^H \hat{\mathbf{y}} = (\hat{\mathbf{S}}^{(j)})^\dagger \hat{\mathbf{y}}$ <p>4) Residual update: <math>\hat{\varepsilon}^{(j)} = \hat{\mathbf{y}} - \hat{\mathbf{S}}^{(j)} \hat{\mathbf{a}}^{(j)}</math></p>
--

matrix which is supposed to be sparse,  $\hat{\mathbf{A}} \in \mathbb{H}^{K \times N}$  and  $\|\hat{\mathbf{A}}\|_0 = \sum_{i=1}^N \|\hat{\mathbf{a}}_i\|_0$  counts the nonzero entries of the columns of  $\hat{\mathbf{A}}$ . To achieve the optimized dictionary, we propose a training algorithm as the counterpart of K-SVD [18], [21] in the quaternion form, which we call K-QSVD (Generalized K-means clustering for Quaternion Singular Value Decomposition). It is separated into two steps, i.e., sparse coding stage and dictionary updating stage.

During the sparse coding stage, a sparse coefficient matrix  $\hat{\mathbf{A}}$  is to be solved given a fixed  $\hat{\mathbf{D}}$  in (38). Methods like matching pursuit (MP) [51], basis pursuit (BP) [52] and so on are suitable for this sparse coding problem. In this paper, we choose the orthogonal matching pursuit (OMP) [53] algorithm for quaternion extension in consideration of its high efficiency, to design the QOMP (quaternion orthogonal matching pursuit) algorithm. The QOMP algorithm solves the problem of decomposing signal  $\hat{\mathbf{y}} \in \mathbb{H}^n$  on a quaternion dictionary  $\hat{\mathbf{D}} \in \mathbb{H}^{n \times K}$  such that,

$$\hat{\mathbf{a}} = \underset{\hat{\mathbf{a}}}{\operatorname{argmin}} \|\hat{\mathbf{y}} - \hat{\mathbf{D}}\hat{\mathbf{a}}\|_2^2, \quad \text{s.t. } \|\hat{\mathbf{a}}\|_0 \leq L, \quad (39)$$

where  $\hat{\mathbf{a}} \in \mathbb{H}^K$  is the sparse coefficient vector and  $\|\hat{\mathbf{a}}\|_0 \leq L$  is the stopping criteria. It alleviates the NP-hard  $l_0$ -norm sparse coding problem by specifying the maximum number of non-zero coefficients per signal.

The implementation details of QOMP for each patch are given in Table I.

- 1) We initialize the residual  $\hat{\varepsilon}^{(0)} = \hat{\mathbf{y}}$  as the input patch  $\hat{\mathbf{y}}$  itself, and the atom set  $\hat{\mathbf{S}}$  as an empty set.
- 2) At the  $j$ -th iteration, QOMP selects the atom that produces the largest projection onto current residual. First, we compute the correlation between current residual and each atom  $\hat{\mathbf{d}}_m$  from the atom pool  $\hat{\mathbf{D}} \setminus \hat{\mathbf{S}}^{(j-1)}$ , i.e.,  $C_m^{(j)} = \langle \hat{\mathbf{d}}_m, \hat{\varepsilon}^{(j-1)} \rangle$ . Then we add the atom which achieves the highest correlation value into atom set  $\hat{\mathbf{S}}^{(j)}$ .
- 3) We compute coefficients by  $\hat{\mathbf{a}}^{(j)} = ((\hat{\mathbf{S}}^{(j)})^H \hat{\mathbf{S}}^{(j)})^{-1} (\hat{\mathbf{S}}^{(j)})^H \hat{\mathbf{y}} = (\hat{\mathbf{S}}^{(j)})^\dagger \hat{\mathbf{y}}$ , where the superscript  $\dagger$  denotes the quaternionic pseudoinverse operation.  $((\hat{\mathbf{S}}^{(j)})^H \hat{\mathbf{S}}^{(j)})^{-1}$  is calculated as follows: we first compute the QSVD of  $(\hat{\mathbf{S}}^{(j)})^H \hat{\mathbf{S}}^{(j)}$ , then replace all nonzero singular values by their reciprocals.
- 4) We refine the residual signal as  $\hat{\varepsilon}^{(j)} = \hat{\mathbf{y}} - \hat{\mathbf{S}}^{(j)} \hat{\mathbf{a}}^{(j)}$ .

TABLE II

QUATERNION-BASED DICTIONARY LEARNING USING K-QSVD METHOD

- 1. Initialization:** Construct the training color data  $\hat{\mathbf{Y}} = \{\hat{\mathbf{y}}_i, 1 \leq i \leq N\}$ , and initialize the dictionary matrix  $\hat{\mathbf{D}} = \{\hat{\mathbf{d}}_i, 1 \leq i \leq K\}$  as random samples from  $\hat{\mathbf{Y}}$ , where each atom  $\hat{\mathbf{d}}_i \in \mathbb{H}^n$  and block patch  $\hat{\mathbf{y}}_i \in \mathbb{H}^n$ .
- 2. Repeat J times:**
- 1) Sparse Coding Stage: Use QOMP to compute the coefficient matrix  $\hat{\mathbf{A}} = \{\hat{\alpha}_i, 1 \leq i \leq N\}$ , where each coefficient column  $\hat{\alpha}_i \in \mathbb{H}^K$ .
  - 2) Codebook update Stage: Update each dictionary atom  $\hat{\mathbf{d}}_k$  in  $\hat{\mathbf{D}}^J$  through (i)-(iii) steps.
    - (i) Find the set of patches that use atom  $\hat{\mathbf{d}}_k$ , the index  $\omega_k = \{i | 1 \leq i \leq N, \hat{\mathbf{A}}(k, i) \neq 0\}$ , where  $\hat{\mathbf{A}}(k, i)$  indicates the entry at k-th row and i-th column of the coefficient matrix  $\hat{\mathbf{A}}$ .
    - (ii) Compute the error  $\hat{\mathbf{E}}_k = \hat{\mathbf{Y}} - \sum_{j \neq k} \hat{\mathbf{d}}_j \hat{\mathbf{A}}^j$  and select the columns corresponding to  $\omega_k$  to form  $\hat{\mathbf{E}}_k^R = \hat{\mathbf{E}}_k(i, j)|_{j \in \omega_k}$  for QSVD:
 
$$\hat{\mathbf{E}}_k^R = \hat{\mathbf{U}} \hat{\mathbf{\Lambda}} \hat{\mathbf{V}}^H.$$
    - (iii) Update  $\hat{\mathbf{d}}_k$  as the first column vector of  $\hat{\mathbf{U}}$ , and set its corresponding nonzero coefficient  $\hat{\alpha}_R^k = \omega_k^T \hat{\alpha}^k$  to be the multiplication of the first column of  $\hat{\mathbf{V}}^H$  and  $\Lambda(1, 1)$ .

Once we obtain sparse coefficient for every patch  $\hat{\mathbf{y}}_i$ , the sparse coefficient matrix  $\hat{\mathbf{A}}$  is generated as the columns of  $\hat{\mathbf{a}}_i, 1 \leq i \leq N$ .

Obviously, QOMP is an extension of traditional OMP, which replaces real number by quaternion. At each step, the reconstruction residual is reduced. Because it has the same framework with OMP, QOMP is still a greedy algorithm — the more nonzero coefficients we obtain, the smaller reconstruction residual we have. In other words, the reconstruction residual converges monotonously with the increase of iteration number  $j$ . Similar to OMP, we set an upper bound of iteration number, which achieves the trade-off between the reconstruction residual and the sparsity of coefficient vector.

Given the sparse coding solution, the quaternion-based dictionary  $\hat{\mathbf{D}}$  can be trained. Different from traditional ways which fix the coefficient matrix during dictionary learning, K-QSVD is highly efficient due to its ability to update coefficient simultaneously. For each atom  $\hat{\mathbf{d}}_k$  and the corresponding coefficients  $\hat{\mathbf{A}}^k$  - the  $k$ -th row of  $\hat{\mathbf{A}}$ , we update both of them by decomposing the remaining representation error  $\hat{\mathbf{E}}_k = \hat{\mathbf{Y}} - \sum_{j \neq k} \hat{\mathbf{d}}_j \hat{\mathbf{A}}^j$  using QSVD. In section II, we observe that the first basis of QSVD contains more information than that of SVD for color images. This indicates that the update of atoms and their corresponding coefficients can be more efficient using K-QSVD. The details of K-QSVD algorithm is shown in Table II.

### B. Further Analysis

Because K-QSVD uses the same framework of traditional K-SVD [15], [54], [55], its convergence is also similar to that of K-SVD. In each iteration, the K-QSVD consists of sparse coding phase and dictionary learning phase. In the sparse coding phase, we fix dictionary and obtain sparse codes by QOMP, which is the quaternion version of traditional OMP. Like OMP, our QOMP algorithm is a greedy algorithm — the more nonzero coefficients we select, the smaller reconstruction residual we have. In other words, the reconstruction residual

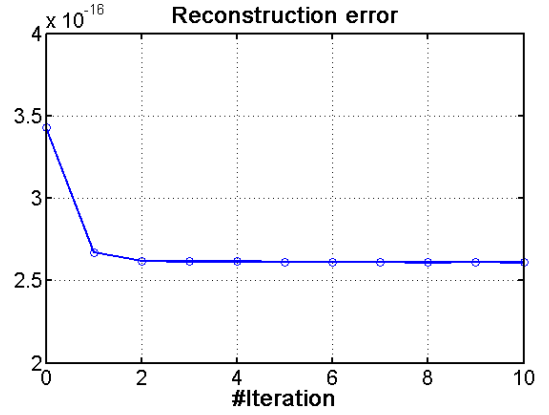


Fig. 3. The reconstruction error  $\|\hat{\mathbf{Y}} - \hat{\mathbf{D}}\hat{\mathbf{A}}\|_F$  with respect to the number of iteration of K-QSVD.

reduces with respect to the number of iteration (the nonzero coefficients we obtain). In the dictionary learning phase, we fix current sparse codes and optimize dictionary as K-SVD does: for each atom, its updating ensures that the energy of the previous residual error matrix to be reduced. Although this process is heuristic, its performance is satisfying in practice.<sup>3</sup> In fact, we further verify the convergence of our K-QSVD algorithm in the following analytic experiment, observing the rapid reduction of the reconstruction error. Specifically, given 10000 color image patches, the size of which is  $8 \times 8$ , we train a dictionary  $\hat{\mathbf{D}} \in \mathbb{H}^{64 \times 256}$  by our K-QSVD algorithm. The reconstruction error  $\|\hat{\mathbf{Y}} - \hat{\mathbf{D}}\hat{\mathbf{A}}\|_F$  with respect to the number of iteration is shown in Fig. 3. We note that the reconstruction error converges quickly after 4 iterations.

Compared with traditional OMP and K-SVD, the proposed QOMP and K-QSVD algorithms have higher computational complexity. According to the definitions in Section II.A, the quaternion addition operation requires 4 floating-point operations (FLOPs) and the quaternion multiplication operation requires 28 FLOPs (16 floating-point multiplications and 12 floating-point additions). Furthermore, focusing on color image representation, the real part of quaternion is set to be 0, so the quaternion addition and multiplication require 3 FLOPs and 14 FLOPs (9 floating-point multiplications and 5 floating-point additions). Suppose that we have  $N$  samples corresponding to color patches, whose dimension is  $3 \times D$ . The dictionary size is  $3D \times K$  for the real dictionary (or  $D \times K$  for the quaternion dictionary), where  $K$  is the number of atoms. The sparsity constraint is  $3L$  for the sparse code of each color channel (or  $L$  for the quaternion sparse code). As a result, according to [56], the complexity of OMP is  $\mathcal{O}(3LN(3D)^3)$  and the complexity of QOMP is  $\mathcal{O}(LN(\frac{14D}{3})^3)$ . Similarly, the complexity of K-SVD is  $\mathcal{O}(3LN(3D)^3)$  and the complexity of K-QSVD is  $\mathcal{O}(LN(\frac{14D}{3})^3)$  as well.<sup>4</sup> In other words,

<sup>3</sup>Actually, K-SVD applies the same alternative optimization strategy, which has been proven to be useful and widely used in many practical applications.

<sup>4</sup>Because both OMP (QOMP) and K-SVD (K-QSVD) spend most of time computing the SVD of matrix in each iteration, their computational costs are comparable, which are in the same order of magnitude. In the worst case, the complexity of SVD is about  $\mathcal{O}((3D)^3)$  for the real dictionary and  $\mathcal{O}((\frac{14D}{3})^3)$  for the quaternion dictionary.

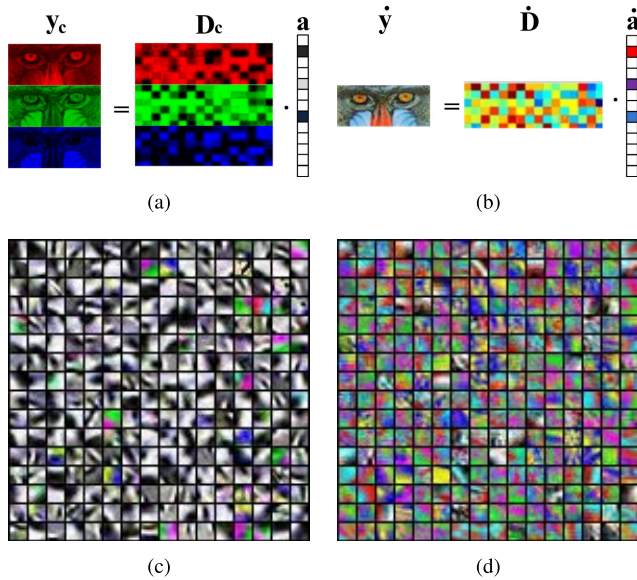


Fig. 4. Dictionaries with 256 atoms learned on a generic database of animal images. Each block is scaled and shifted to the range  $[0, 255]$  for each channel. (a) Concatenation sparse model. (b) Quaternion-based sparse model. (c) K-SVD learned dictionary. (d) K-QSVD learned dictionary.

the computational complexity of learning the quaternion sparse model is about 3.7 times that of learning the real sparse model. In terms of memory cost, the memory usage for learning the quaternion sparse model is 1.33 times that for learning the real sparse model because we need to store the real part of quaternion. Fortunately, because most of quaternions in the algorithm have zero real parts, the increase of memory cost for our algorithms can be ignored.

It should be noted that the benefits obtained by the high computational complexity is obvious. Fig. 4 shows the results of dictionary training methods using the concatenation sparse model (24) and the quaternion-based sparse model (26), where the corresponding trained dictionaries are demonstrated for comparison. Given RGB color images, the four channel dictionaries are linearly correlated, as shown in (33). Consequently, we can constrain the color atoms in the quaternion dictionary as triplets so that a color image is reconstructed using only the three dictionaries  $\mathbf{D}_r$ ,  $\mathbf{D}_g$  and  $\mathbf{D}_b$  while enforcing  $\mathbf{D}_s = 0$ . We observe that the learned dictionary from K-SVD algorithm using the concatenation sparse model tends to be monochromatic. As mentioned in [18], the atoms generated by K-SVD are not rich enough to represent the diversity of colors, since K-SVD emphasizes the basic spatial structures of concatenated channel images. In contrast, the learned quaternion dictionary has more color which captures the interrelationship between color channels as well as the spatial coherence better. In Section V, the proposed quaternion dictionary training process is directly used in image reconstruction and extended to image restoration, which achieves better color fidelity with fewer iterations.

### C. Joint Dictionary Training

In image restoration problem, it is beneficial to have two dictionaries that capture some linear relationship between



Fig. 5. Examples of training images for color image reconstruction.

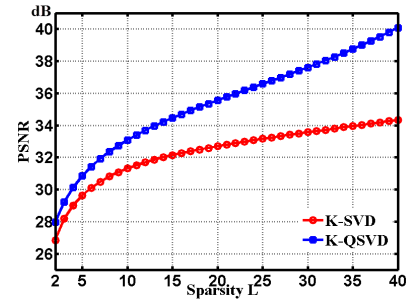


Fig. 6. Comparison of K-QSVD sparse model and K-SVD sparse model for color image reconstruction - PSNR values vs. sparse parameter  $L$ .

the original image and the damaged image. Motivated by the assumption of the similarity of sparse representation between low- and high-resolution image patches over their own dictionaries [20], [57], we propose a joint dictionary training method for the proposed quaternion-based sparse model.

Let  $\mathbf{F}_1$  and  $\mathbf{F}_2$  be two linear filters (projection matrix) for image sample patches  $\dot{\mathbf{Y}}_1$  and  $\dot{\mathbf{Y}}_2$ , they are both obtained by linear filtering of the same sample patches  $\dot{\mathbf{Y}}$ . We seek a sparse representation for each patch of  $\mathbf{F}_1 \dot{\mathbf{Y}}_1$ , and then use the coefficients of this representation to generate the sparse representation of  $\mathbf{F}_2 \dot{\mathbf{Y}}_2$ . First, we use K-QSVD method to solve sparse coding problem of (40) and obtain  $\hat{\mathbf{D}}_1, \hat{\mathbf{A}}$ .

$$\{\hat{\mathbf{D}}_1, \hat{\mathbf{A}}\} = \min_{\mathbf{D}_1, \mathbf{A}} \|\mathbf{F}_1 \dot{\mathbf{Y}}_1 - \mathbf{D}_1 \mathbf{A}\|_F^2 + \lambda_1 \|\mathbf{A}\|_0. \quad (40)$$

Then we enforce the shared sparse coding constraint on (41),

$$\hat{\mathbf{D}}_2 = \min_{\mathbf{D}_2} \|\mathbf{F}_2 \dot{\mathbf{Y}}_2 - \mathbf{D}_2 \hat{\mathbf{A}}\|_F^2. \quad (41)$$

and calculate dictionary  $\hat{\mathbf{D}}_2$  as

$$\hat{\mathbf{D}}_2 = (\mathbf{F}_2 \dot{\mathbf{Y}}_2) \hat{\mathbf{A}}^\dagger. \quad (42)$$

In some applications,  $\mathbf{F}_2$  is set as an identity matrix. In these cases, we learn joint dictionaries  $\hat{\mathbf{D}}_2$  and  $\hat{\mathbf{D}}_1$  from samples, which encode the linear relationship between the non-corrupted image  $\dot{\mathbf{Y}}_2$  and the damaged image  $\dot{\mathbf{Y}}_1$ , so that we could recover the original image from  $\hat{\mathbf{D}}_2$  and sparse coding of the damaged image.

## V. APPLICATIONS TO IMAGE PROCESSING AND EXPERIMENTAL RESULTS

For validation of the proposed quaternion-based sparse representation model, we apply it to natural color image processing, such as reconstruction, denoising, inpainting and super-resolution.



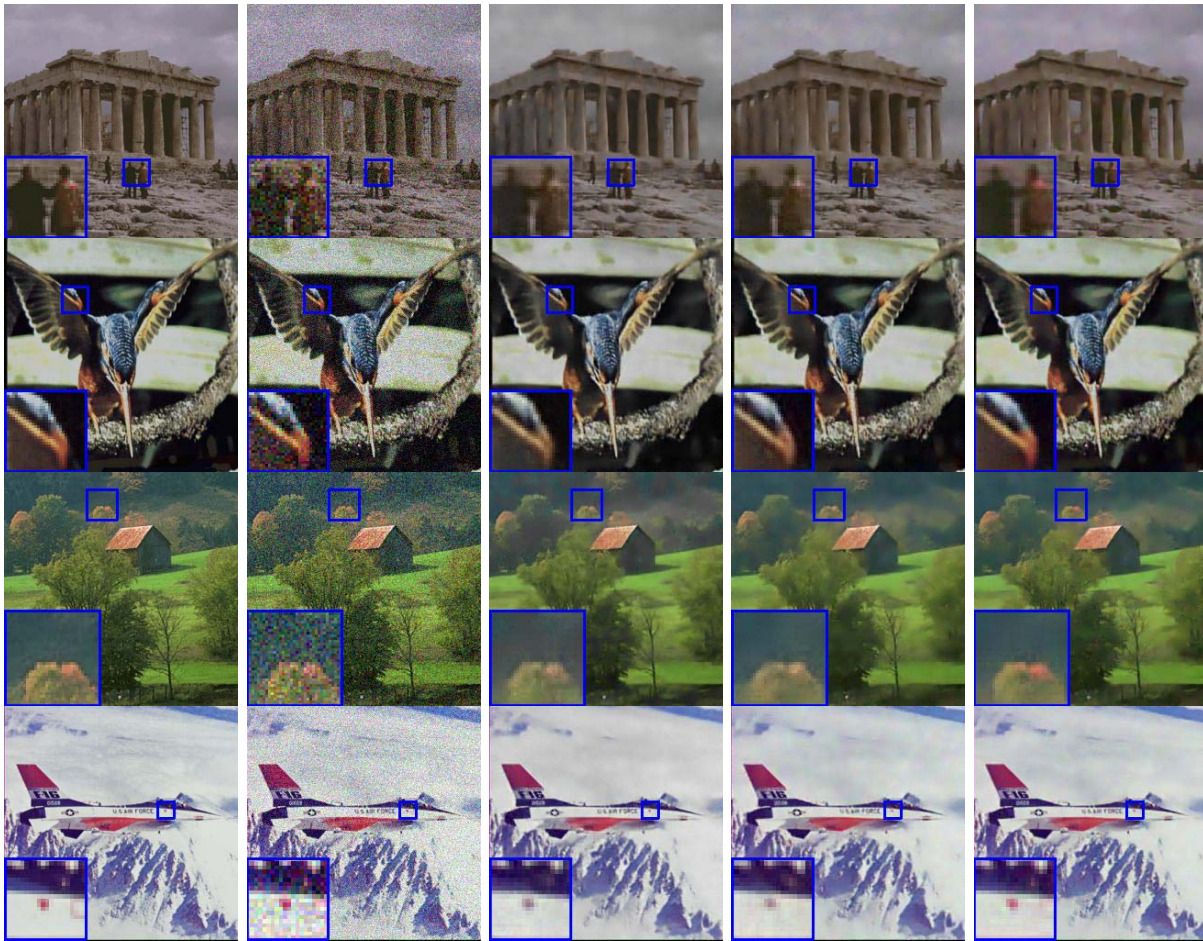


Fig. 7. Columns from left to right: Original image, noisy image with  $\sigma = 25$ , K-SVD denoising results [15], improved K-SVD denoising results [18], and the proposed K-QSVD denoising results.

TABLE III

PSNR(dB)/SSIM VALUES OF DIFFERENT DENOISING METHODS. EACH CASE IS COMPRISED OF THREE PARTS: THE TOP RESULTS ARE BASED ON MODEL (24) [15], THE MIDDLE ONES ARE OBTAINED BY ITS IMPROVED VERSION WITH THE CONSTRAINT OF UNCHANGED AVERAGE COLOR [18], AND THE BOTTOM ONES ARE OUR RESULTS USING QUATERNION-BASED COLOR IMAGE SPARSE MODEL

$\sigma$	barn	athens	blueeye	bee	bird	plane	aquatic	sunflower
15	30.83/0.83	31.80/0.91	34.07/0.91	33.40/0.90	32.06/0.94	32.68/0.92	31.26/0.92	33.18/0.92
	30.79/0.82	32.33/0.91	34.30/0.92	33.42/0.91	32.08/0.94	32.80/0.93	31.37/0.92	33.23/0.92
	<b>30.91/0.84</b>	<b>32.40/0.93</b>	<b>34.42/0.92</b>	<b>33.47/0.91</b>	<b>32.09/0.94</b>	<b>32.91/0.93</b>	<b>31.39/0.92</b>	<b>33.41/0.93</b>
20	29.19/0.76	30.11/0.88	32.43/0.89	31.82/0.88	30.28/0.92	31.02/0.90	29.37/0.88	31.56/0.91
	29.29/0.74	<b>30.52/0.87</b>	32.90/0.90	31.93/0.87	30.34/0.92	<b>31.23/0.90</b>	<b>29.63/0.88</b>	31.77/0.90
	<b>29.41/0.75</b>	30.49/0.89	<b>33.07/0.91</b>	<b>32.18/0.88</b>	<b>30.50/0.93</b>	31.22/0.91	29.50/0.89	<b>31.79/0.91</b>
25	27.25/0.67	27.73/0.82	30.24/0.83	27.84/0.80	28.97/0.89	29.95/0.88	27.70/0.85	30.24/0.89
	27.30/0.65	28.14/0.80	30.47/0.83	<b>28.12/0.83</b>	29.01/0.90	29.96/0.89	28.30/0.85	30.43/0.88
	<b>27.34/0.67</b>	<b>28.26/0.83</b>	<b>30.69/0.84</b>	<b>28.24/0.81</b>	<b>29.19/0.91</b>	<b>29.96/0.89</b>	<b>28.30/0.86</b>	<b>30.55/0.89</b>

### A. Color Image Reconstruction

We first compare the proposed sparse model with the model in (24) for color image reconstruction. The dataset for training consists of 50,000 image sample patches of size  $8 \times 8$ , which are randomly selected from a wide variety of animal images with different scenes. Some of them are shown in Fig. 5. Then we train the dictionaries using K-SVD and K-QSVD separately on the same training samples. In order to keep a reasonable computational complexity, both dictionaries are relatively small with 256 atoms. To provide comparison of our K-QSVD sparse model and Elad's K-SVD sparse model [18],

we randomly pick 20 images and concatenate them as a full image for reconstruction.

We first compute the PSNR(dB) values over different sparse parameter  $L$  for both models. As shown in Fig. 6, the quaternion-based sparse model is able to present higher PSNR values than the model in (24) with the same sparse parameter. The advantage becomes even greater with the increasing number of atoms used.

We further compare the number of atoms to be used under the same PSNR. From Fig. 6, we compute the ratio of sparsity  $L$  with the PSNR value ranging from 28dB to 34dB,

and get the average ratio to be 2.56, which means that we only need about 1/3 number of atoms for the quaternion-based model than that in (24) for achieving a similar reconstruction performance with a reasonable sparse parameter. Moreover, an interesting phenomenon is observed that the advantage of K-QSVD becomes even more obvious when more atoms are allowed to be used. This is due to the lower intra-redundancy between the channel components of each atom and the lower inter-redundancy between each pair of atoms in the quaternion-based dictionary. As mentioned in Section IV, the K-QSVD trained dictionary is able to present more colorful structures, which indicates lower intra-redundancy between the channel components of each atom. Then we compute the average correlation among atoms using  $\frac{1}{K(K-1)/2} \sum_{i \neq j} \langle \mathbf{d}_i, \mathbf{d}_j \rangle$  for K-SVD and K-QSVD trained dictionaries and get 0.70 and 0.41 respectively, which indicates lower inter-redundancy of the proposed K-QSVD sparse model.

### B. Color Image Denoising

Another common application of sparse representation is denoising. Let  $\hat{\mathbf{X}} \in \mathbb{H}^{\sqrt{N} \times \sqrt{N}}$  be a clean image, with noisy version:

$$\hat{\mathbf{Y}} = \hat{\mathbf{X}} + \hat{\mathbf{w}}, \quad (43)$$

where  $\hat{\mathbf{w}}$  is the white Gaussian noise in quaternion form with a spatially uniform deviation  $\sigma$ . We assume all patches of size  $\sqrt{n} \times \sqrt{n}$  in the clear image  $\hat{\mathbf{X}}$  admit sparse representations.

The denoising problem can be formulated as the minimization of the following objective function:

$$\{\hat{\mathbf{D}}, \hat{\mathbf{a}}_{ij}, \hat{\mathbf{X}}\} = \min_{\hat{\mathbf{D}}, \hat{\mathbf{a}}_{ij}, \hat{\mathbf{X}}} \{\lambda \|\hat{\mathbf{X}} - \hat{\mathbf{Y}}\|_2^2 \quad (44)$$

$$+ \sum_{i,j} \mu_{ij} \|\hat{\mathbf{a}}_{ij}\|_0 + \sum_{i,j} \|\hat{\mathbf{D}}\hat{\mathbf{a}}_{ij} - R_{ij}\hat{\mathbf{X}}\|_2^2\}, \quad (45)$$

where  $\hat{\mathbf{X}}$  is the estimation of  $\hat{\mathbf{X}}$ , and the dictionary  $\hat{\mathbf{D}}$  of size  $n \times K$  is the estimation of the optimal dictionary which leads to the sparsest representation of the recovered image patches. The indices  $[i, j]$  mark the location of patches, thus  $R_{i,j}$  is the operator extracting the  $\sqrt{n} \times \sqrt{n}$  square patch at coordinates  $[i, j]$  from  $\hat{\mathbf{X}}$ , and the vector  $\hat{\mathbf{a}}_{ij}$  of size  $K \times 1$  is the coefficient vectors for the patch at index  $[i, j]$ . The first term in (44) enforces the likelihood that demands proximity between  $\hat{\mathbf{X}}$  and  $\hat{\mathbf{Y}}$ . The second and the third terms impose the image prior, assuming each quaternion patch can be sparsely represented without noise over dictionary  $\hat{\mathbf{D}}$ .

The solution to (44) is an extension of [15], with all algebra operations in quaternion system, where the key part for suppressing noise falls on the QOMP implementation,

$$\min_{\hat{\mathbf{a}}_{ij}} \|\hat{\mathbf{a}}_{ij}\|_0, \quad \text{s.t.} \quad \|\hat{\mathbf{D}}\hat{\mathbf{a}}_{ij} - R_{ij}\hat{\mathbf{X}}\|_2^2 \leq n(C\sigma)^2, \quad (46)$$

which stops searching the best candidate atom once the approximation reaches the sphere of radius  $\sqrt{n(C\sigma)^2}$  in each patch's quaternion space.

Fig. 7 shows the comparison of several denoising methods based on sparse representation. The K-SVD denoising algorithm based on the model in (24) [15] tends to introduce color bias and blurring effects (third column). The fourth column

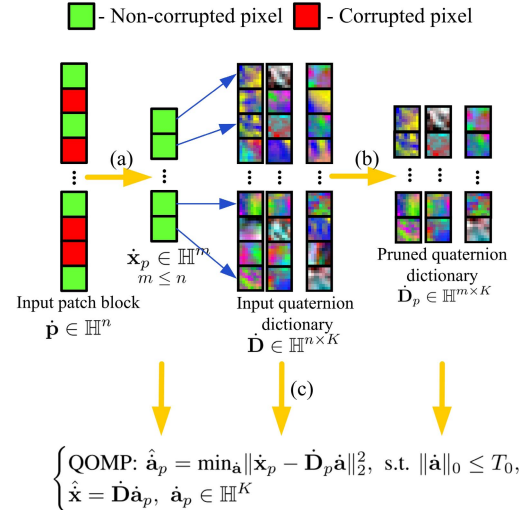


Fig. 8. The workflow of color image inpainting using quaternion-based sparse model. (a) extract non-corrupted pixels. (b) prune the quaternion dictionary corresponding to non-corrupted pixels. (c) reconstruct the original blocks according to the projections of non-corrupted pixels.

shows the results of the improved OMP method where an additional regularization term is added to ensure the average value of each channel remains unchanged during sparse representation [18]. It reduces some hue bias, but still loses channel interrelationship. As shown in Fig. 7, the color distortions appear in the “gray” sky. In contrast, the proposed quaternion-based method (last column) can present colors with better fidelity after denoising. Table III summaries the PSNR and SSIM results where the proposed method mostly has the highest values, which further verify the advantages of the quaternion-based model.

### C. Color Image Inpainting

Image inpainting refers to filling the missing information in an image. Limited by the patch size, the learning-based method can only handle small holes. In this paper, we focus on filling missing areas within the order of 30 pixels. We randomly choose one full image which is damaged by randomly deleting a fraction  $r$  of the pixels, usually  $r \in [0.2, 0.9]$ . Our goal is to re-fill them. Fig. 8 shows the workflow of the proposed color image inpainting:

- We only consider the projections of non-corrupted pixels onto dictionary in the QOMP algorithm.
- The coefficient vector for each patch  $\hat{\mathbf{p}}$  can be estimated only on the non-corrupted pixels  $\hat{\mathbf{x}}_p$  using the pruned dictionary  $\hat{\mathbf{D}}_p$  by selecting corresponding rows of  $\hat{\mathbf{D}}$ .
- The computed coefficient vector  $\hat{\mathbf{a}}_p$  can be shared with those missing pixels, considering its validity for the whole complete patch block  $\hat{\mathbf{p}}$ . Therefore, the reconstructed block  $\hat{\mathbf{x}}$  is obtained as  $\hat{\mathbf{x}} = \hat{\mathbf{D}} \hat{\mathbf{a}}_p$ .

Fig. 9 shows the comparison of inpainting performance using the model in (24) and the proposed quaternion-based model in (26), with the computed PSNR values. It can be observed that higher quality image restoration with fewer artifacts is obtained using the proposed model.

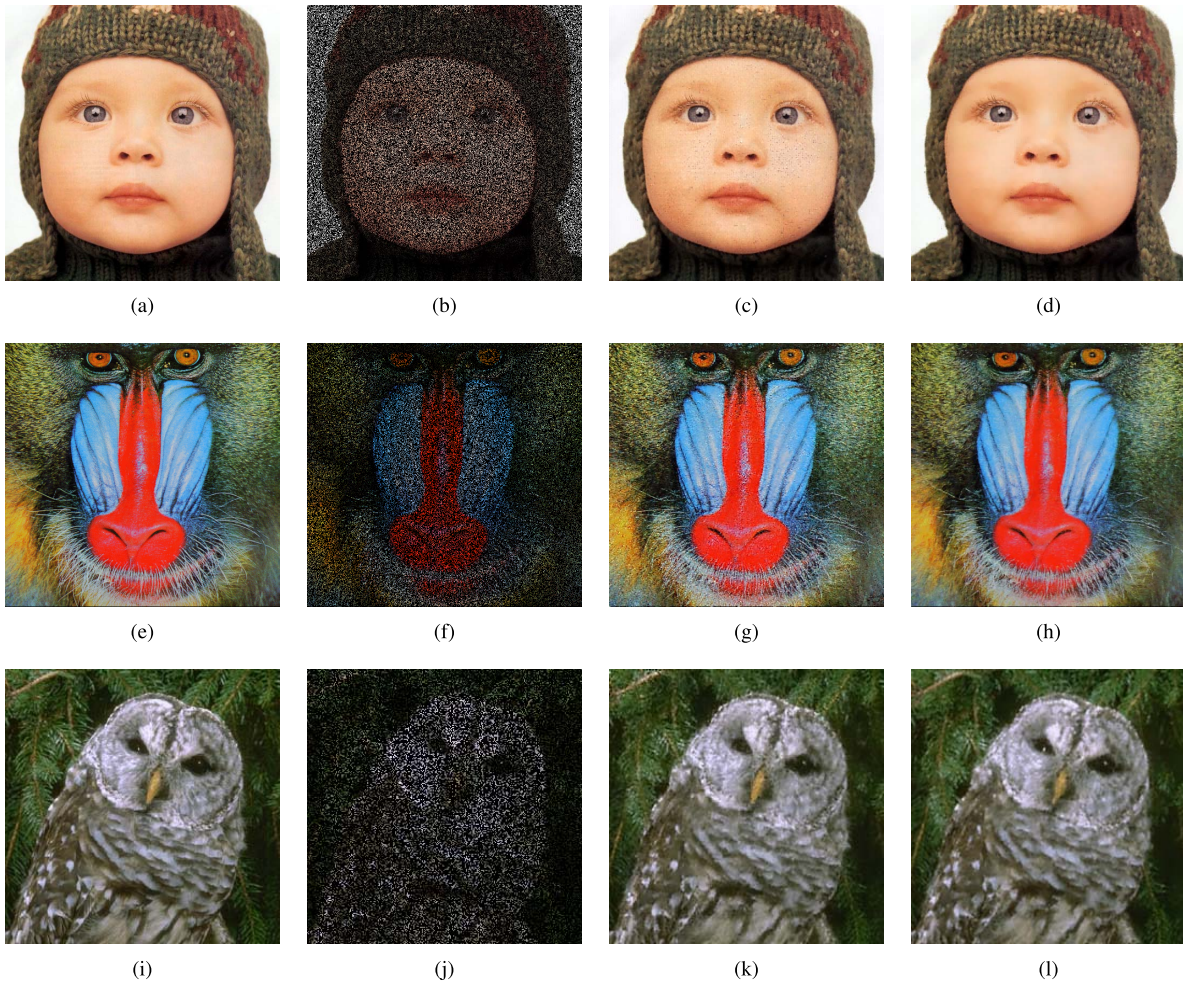


Fig. 9. Visual comparisons and PSNR(dB) results of K-SVD method [18] and the quaternion-based sparse model on image inpainting. (a) Ground truth. (b) Damaged (70% missing). (c) K-SVD [18] (31.488). (d) K-QSVD (32.379). (e) Ground truth. (f) Damaged (70% missing). (g) K-SVD [18] (19.749). (h) K-QSVD (22.100). (i) Ground truth. (j) Damaged (70% missing). (k) K-SVD [18] (25.621). (l) K-QSVD (26.227).

TABLE IV  
MORE PSNR(dB)/SSIM VALUES OF 3X SUPER-RESOLUTION RESULTS USING DIFFERENT ALGORITHMS

Methods	baboon	leaf	eye	flower	plane	birds
Bi-cubic	21.383/0.517	31.96/0.891	36.777/0.969	24.454/0.757	27.256/0.867	27.180/0.845
Shan [58]	20.087/0.404	32.09/0.888	36.738/0.961	24.615/0.756	27.079/0.853	27.863/0.851
Yang [20]	21.358/0.520	31.96/0.888	35.994/0.956	24.651/0.766	26.939/0.850	28.006/0.860
Zeyede [57]	21.383/0.517	31.96/0.890	36.891/0.969	24.630/0.761	<b>27.256/0.867</b>	28.080/0.871
OnlineQ	21.268/0.501	32.405/0.895	37.793/0.969	<b>26.023/0.796</b>	26.697/0.851	27.859/0.863
Proposed	<b>21.414/0.522</b>	<b>32.94/0.910</b>	<b>38.704/0.974</b>	24.713/0.768	27.149/0.864	<b>28.584/0.879</b>
Methods	house	jellybeans	lenna	monarch	pepper	sailboat
Bi-cubic	29.131/0.802	31.140/0.940	28.111/0.808	27.437/0.899	30.126/0.790	26.367/0.734
Shan [58]	30.111/0.807	32.305/0.940	28.750/0.812	28.104/0.907	30.721/0.784	26.948/0.742
Yang [20]	29.971/0.804	31.985/0.935	29.017/0.822	28.241/0.908	30.576/0.777	26.760/0.740
Zeyede [57]	<b>30.109/0.820</b>	32.552/0.948	<b>29.018/0.833</b>	28.111/0.908	<b>30.952/0.801</b>	<b>27.068/0.757</b>
OnlineQ	29.701/0.810	31.846/0.945	28.910/0.829	<b>29.070/0.917</b>	30.596/0.794	26.576/0.741
Proposed	30.083/0.820	<b>32.780/0.953</b>	28.932/0.832	28.089/0.909	<b>30.922/0.803</b>	27.016/0.756

It should be noted that in [31], another vector sparse representation model is proposed for color image inpainting as well. However, that model requires a channel (gray or color) to be available in advance for estimating the missing channels. In other words, what it does is colorization rather than inpainting. Differing from [31], our method can recover missing pixels whose values of all channels are missing. From this view, our method is superior to that model in image inpainting.

#### D. Single Color Image Super-Resolution

Single image super-resolution refers to the process of obtaining higher-resolution (HR) images  $\hat{\mathbf{X}}_H$  from one lower-resolution (LR) image  $\hat{\mathbf{X}}_L$ . Current image super-resolution methods can be divided into three categories: interpolation-based methods, reconstruction-based methods and example-based methods. Among interpolation-based algorithms, bi-linear and bi-cubic are most commonly used but tend

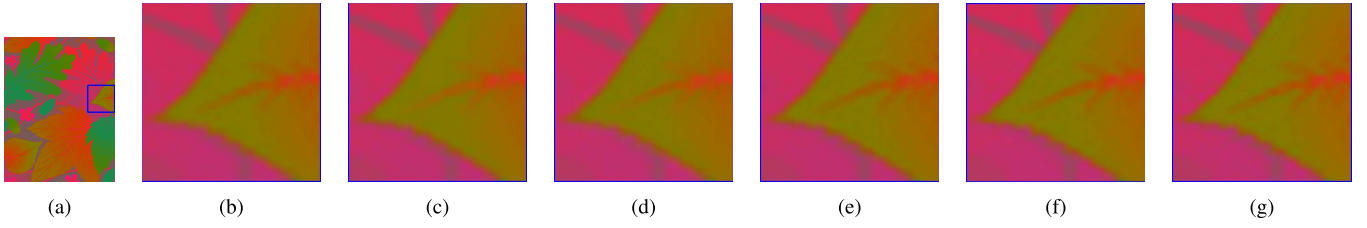


Fig. 10. 3X super-resolution results of leaf ( $110 \times 144$ ) with PSNR(dB) and SSIM. (a) Input. (b) Bi-cubic (31.96/0.891). (c) Shan [58] (32.09/0.888). (d) Yang [20] (31.96/0.888). (e) Zeyede [57] (31.96/0.890). (f) OnlineQ (32.40/0.895). (g) Proposed (32.94/0.910).

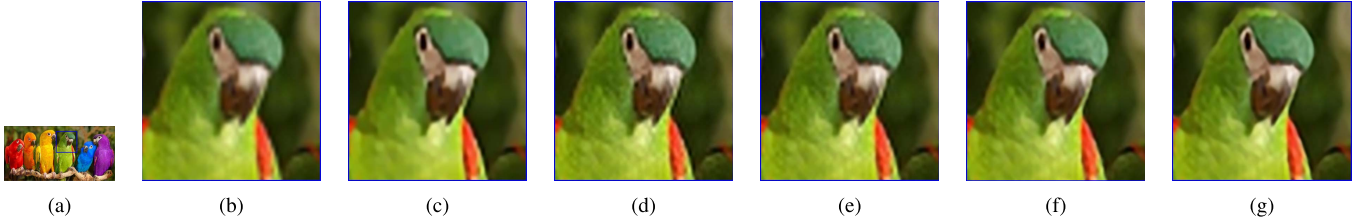


Fig. 11. 3X super-resolution results of birds ( $233 \times 114$ ) with PSNR(dB) and SSIM. (a) Input. (b) Bi-cubic (27.18/0.845). (c) Shan [58] (27.86/0.851). (d) Yang [20] (28.01/0.860). (e) Zeyede [57] (28.08/0.871). (f) OnlineQ (27.86/0.863). (g) Proposed (28.58/0.879).

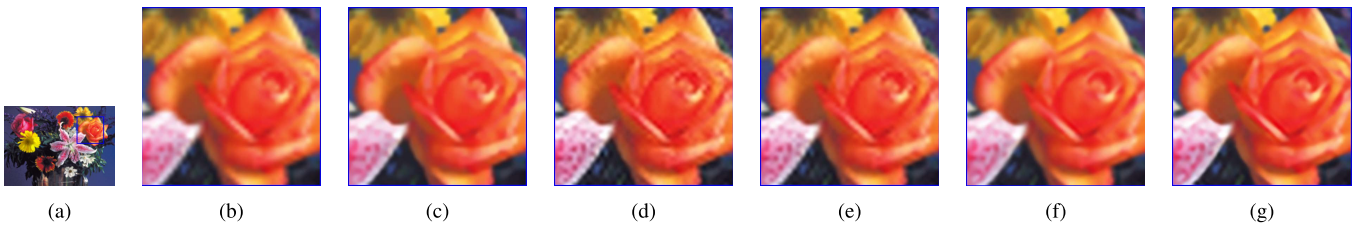


Fig. 12. 3X super-resolution results of flower ( $167 \times 121$ ) with PSNR(dB) and SSIM. (a) Input. (b) Bi-cubic (24.45/0.757). (c) Shan [58] (24.61/0.756). (d) Yang [20] (24.65/0.766). (e) Zeyede [57] (24.63/0.761). (f) OnlineQ (26.02/0.796). (g) Proposed (24.71/0.768).

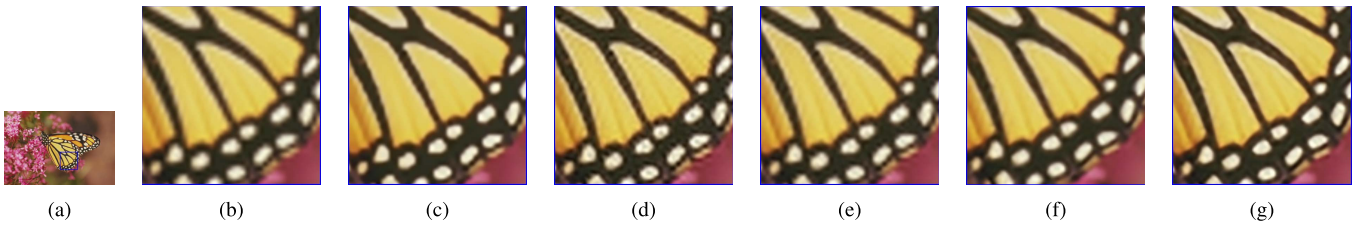


Fig. 13. 3X super-resolution results of monarch ( $256 \times 171$ ) with PSNR(dB) and SSIM. (a) Input. (b) Bi-cubic (27.44/0.899). (c) Shan [58] (28.10/0.907). (d) Yang [20] (28.24/0.907). (e) Zeyede [57] (28.11/0.908). (f) OnlineQ (29.07/0.917). (g) Proposed (28.09/0.909).

to produce blurry and jaggy artifacts. Reconstruction-based methods require the consistency of up-sampled image with the input LR image, where the HR-to-LR degradation process is reversed by various kinds of edge prior models [57]–[59].

More recent researches have focused on the third type, i.e., example-based methods, which reconstruct the high-frequency band of LR image using the provided example database. The works in [60] and [61] exploited the raw patch information from database, whereas our approach finds the sparse representation of the example database, similar to the approach in [57].

We use the general patch samples  $\hat{\mathbf{X}} = \{\hat{\mathbf{X}}_L, \hat{\mathbf{X}}_H\}$  to learn a joint dictionary in (40) and (42), where  $\hat{\mathbf{X}}_L$  and  $\hat{\mathbf{X}}_H$  are obtained by linear low-pass and high-pass filtering of the same image dataset. Dictionary  $\hat{\mathbf{D}}_1$  is for representing the low-resolution ones in the example dataset, which is denoted as  $\mathbf{F}_1 \hat{\mathbf{X}}_L$ ; and dictionary  $\hat{\mathbf{D}}_2$  is for representing the residual high-frequency bands, which is denoted as  $\mathbf{F}_2 \hat{\mathbf{X}}_H$  and  $\mathbf{F}_2 = \mathbf{I}$ .

The linear filter  $\mathbf{F}_1$  is used to extract discriminant features of the low-frequency band  $\hat{\mathbf{X}}_L$ . Instead of filtering on luminance channel plane alone, we extract dominant features from RGB channels respectively, using four 1D high-pass filters  $[-1, 0, 1]$ ,  $[-1, 0, 1]^T$ ,  $[1, 0, -2, 0, 1]$  and  $[1, 0, -2, 0, 1]^T$  in each channel and then grouping three filtered channels in quaternion form. The concatenation of four high-pass filtered images gives us the final feature representation of low-resolution patch  $\hat{\mathbf{X}}_L$ , and can be embedded into the joint dictionary training procedure of (40) and (41),

$$\{\hat{\mathbf{D}}_1, \hat{\mathbf{A}}\} = \min_{\hat{\mathbf{D}}_1, \hat{\mathbf{A}}} \|\mathbf{F}_1 \hat{\mathbf{X}}_L - \hat{\mathbf{D}}_1 \hat{\mathbf{A}}\|_F^2 + \lambda_1 \|\hat{\mathbf{A}}\|_0, \quad (47)$$

$$\hat{\mathbf{D}}_2 = \min_{\hat{\mathbf{D}}_2} \|\mathbf{F}_2 \hat{\mathbf{X}}_H - \hat{\mathbf{D}}_2 \hat{\mathbf{A}}\|_F^2. \quad (48)$$

We obtain  $\hat{\mathbf{D}}_1$  using K-QSVD method in Table II and calculate  $\hat{\mathbf{D}}_2 = \hat{\mathbf{X}}_H \cdot (\hat{\mathbf{A}})^\dagger$ . By sharing the same sparse coefficient  $\hat{\mathbf{A}}$  and using jointly trained dictionaries  $\hat{\mathbf{D}}_1$  and  $\hat{\mathbf{D}}_2$ , the sparse representation of low-frequency patches can be applied to

its high-frequency ones. Finally, the output HR image is generated by adding  $\hat{\mathbf{X}}_L$  and sparsely reconstructed  $\hat{\mathbf{X}}_H$ . One advantage of quaternion-based sparse model is that it can boost the discriminative power of features as all channel descriptors are now taken into consideration in a vector space, thus improving the prediction accuracy. Besides, the quaternion-based reconstruction is able to present more accurate color structures as mentioned above.

We compare the proposed SR method with other state-of-the-art algorithms on some commonly-used SR testing images. Several representative works are selected from the three SR categories, such as Bi-cubic, deconvolution-based SR [58] and two typical example-based methods using sparse model [20], [57]. Moreover, to further substantiate the advantages of quaternion-based dictionary learning, we also reformulate the online dictionary learning process [3] under quaternion algebra. We call it “OnlineQ” in the experiments, whose implementation details are in [30].

The parameters are set as suggested in these works. We evaluate our experiment using both objective quality metrics and subjective visual quality with the upscaling factor of 3. In Table IV, we compute the PSNR and SSIM values on twelve common images. It is observed that the two quaternion-based SR algorithms obtain better performance than the state-of-the-art works on the global objective evaluations. Especially in the parts with significant inter-channel changes, the two proposed SR algorithms are able to synthesize sharper edges and yield fewer artifacts, which can be seen in Fig. 10-13. It further demonstrates the advantages of quaternion-based sparse model in color feature extraction and color image reconstruction. Meanwhile, it shows the potential of incorporating quaternion system into any sparse model, as well as dictionary learning algorithm for addressing color image problems.

## VI. CONCLUSION

In this paper, we propose a novel sparse model for color image using quaternion matrix analysis. It formulates a color pixel as a vector unit instead of a scalar quantity and consequently overcomes the lack of accuracy describing inter-relationship among color channels. The experiments of reconstruction, denoising, inpainting, and super-resolution on natural color images prove its advantages in effectively accounting for both luminance and chrominance geometry in images.

Currently, the usage of the real part of quaternion seems insufficient: for three-channel color space, the real part is simply set to be zero. We believe that the physically meaningful real part will further help us capture color information. In the future, we will further explore the potential extension of quaternion sparse model to four-channel color space, e.g. CMYK, in which the real part may corresponds to the black channel. Additionally, from the view of algorithm our K-QSVD algorithm does not guarantee global convergence. Recently, a dictionary learning algorithm based on proximal method is proposed in [62], which achieves global convergence. Inspired by this strategy, we plan to further improve our learning algorithm in the future work.

## REFERENCES

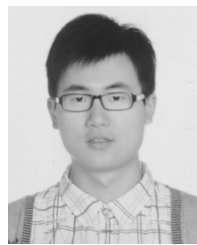
- [1] M. Elad, J.-L. Starck, P. Querre, and D. L. Donoho, “Simultaneous cartoon and texture image inpainting using morphological component analysis (MCA),” *Appl. Comput. Harmon. Anal.*, vol. 19, no. 3, pp. 340–358, Nov. 2005.
- [2] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, “Robust face recognition via sparse representation,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 2, pp. 210–227, Feb. 2009.
- [3] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, “Online dictionary learning for sparse coding,” in *Proc. 26th Annu. Int. Conf. Mach. Learn.*, 2009, pp. 689–696.
- [4] M. Aharon, M. Elad, and A. Bruckstein, “K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation,” *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4311–4322, Nov. 2006.
- [5] H. Lee, A. Battle, R. Raina, and A. Y. Ng, “Efficient sparse coding algorithms,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2006, pp. 801–808.
- [6] Y. Zhang, Z. Jiang, and L. S. Davis, “Learning structured low-rank representations for image classification,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2013, pp. 676–683.
- [7] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, and Y. Ma, “Robust recovery of subspace structures by low-rank representation,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 171–184, Jan. 2013.
- [8] E. Elhamifar and R. Vidal, “Robust classification using structured sparse representation,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2011, pp. 1873–1879.
- [9] Z. Jiang, Z. Lin, and L. S. Davis, “Learning a discriminative dictionary for sparse coding via label consistent K-SVD,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2011, pp. 1697–1704.
- [10] M. Yang, L. Zhang, X. Feng, and D. Zhang, “Fisher discrimination dictionary learning for sparse representation,” in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Nov. 2011, pp. 543–550.
- [11] L. Zelnik-Manor, K. Rosenblum, and Y. C. Eldar, “Dictionary optimization for block-sparse representations,” *IEEE Trans. Signal Process.*, vol. 60, no. 5, pp. 2386–2395, May 2012.
- [12] Y.-T. Chi, M. Ali, A. Rajwade, and J. Ho, “Block and group regularized sparse modeling for dictionary learning,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2013, pp. 377–382.
- [13] Y. Han, F. Wu, Q. Tian, and Y. Zhuang, “Image annotation by input-output structural grouping sparsity,” *IEEE Trans. Image Process.*, vol. 21, no. 6, pp. 3066–3079, Jun. 2012.
- [14] R. M. Figueras i Ventura, P. Vanderghenst, P. Frossard, and A. Cavallaro, “Color image scalable coding with matching pursuit,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, vol. 3, May 2004, pp. iii-53-1–iii-53-6.
- [15] M. Elad and M. Aharon, “Image denoising via sparse and redundant representations over learned dictionaries,” *IEEE Trans. Image Process.*, vol. 15, no. 12, pp. 3736–3745, Dec. 2006.
- [16] F. Chen, H. Yu, and R. Hu, “Shape sparse representation for joint object classification and segmentation,” *IEEE Trans. Image Process.*, vol. 22, no. 3, pp. 992–1004, Mar. 2013.
- [17] S. Agarwal and D. Roth, “Learning a sparse representation for object detection,” in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2006, pp. 97–101.
- [18] J. Mairal, M. Elad, and G. Sapiro, “Sparse representation for color image restoration,” *IEEE Trans. Image Process.*, vol. 17, no. 1, pp. 53–69, Jan. 2008.
- [19] N. L. Bihan and G. Ginolhac, “Subspace methods for 3D arrays,” in *Proc. Workshop Phys. Signal Image Process. (PSIP)*, 2001, pp. 359–364.
- [20] J. Yang, J. Wright, T. S. Huang, and Y. Ma, “Image super-resolution via sparse representation,” *IEEE Trans. Image Process.*, vol. 19, no. 11, pp. 2861–2873, Nov. 2010.
- [21] J. Pang, O. C. Au, K. Tang, and Y. Guo, “Image colorization using sparse representation,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, May 2013, pp. 1578–1582.
- [22] X. Li, B. Gunturk, and L. Zhang, “Image demosaicing: A systematic survey,” *Proc. SPIE*, vol. 6822, p. 68221J, Jan. 2008.
- [23] R. Lukac, B. Smolka, K. Martin, K. N. Plataniotis, and A. N. Venetsanopoulos, “Vector filtering for color imaging,” *IEEE Signal Process. Mag.*, vol. 22, no. 1, pp. 74–86, Jan. 2005.
- [24] A. Koschan and M. Abidi, “Detection and classification of edges in color images,” *IEEE Signal Process. Mag.*, vol. 22, no. 1, pp. 64–73, Jan. 2005.
- [25] K. M. Singh and P. K. Bora, “Adaptive vector median filter for removal impulses from color images,” in *Proc. Int. Symp. Circuits Syst. (ISCAS)*, vol. 2, May 2003, pp. II-396–II-399.

- [26] S. H. Lee, J. Y. Choi, Y. M. Ro, and K. N. Plataniotis, "Local color vector binary patterns from multichannel face images for face recognition," *IEEE Trans. Image Process.*, vol. 21, no. 4, pp. 2347–2353, Apr. 2012.
- [27] C. E. Moxey, S. J. Sangwine, and T. A. Ell, "Hypercomplex correlation techniques for vector images," *IEEE Trans. Signal Process.*, vol. 51, no. 7, pp. 1941–1953, Jul. 2003.
- [28] L. Hua, Y. Huang, Y. Liu, L. Ding, H. Feng, and J. Gu, "Color medical image registration based on quaternion," *J. Inf. Comput. Sci.*, vol. 10, no. 17, pp. 5607–5617, 2013.
- [29] P. Bas, N. Le Bihan, and J.-M. Chassery, "Color image watermarking using quaternion Fourier transform," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, vol. 3, Apr. 2003, pp. III-521-1–III-521-4.
- [30] M. Yu, Y. Xu, and P. Sun, "Single color image super-resolution using quaternion-based sparse representation," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2014, pp. 5804–5808.
- [31] G. Teschke and R. Ramlau, "An iterative algorithm for nonlinear inverse problems with joint sparsity constraints in vector-valued regimes and an application to color image inpainting," *Inverse Problems*, vol. 23, no. 5, p. 1851, 2007.
- [32] B. Sathyabama, P. Chitra, V. G. Devi, S. Raju, and V. Abhaikumar, "Quaternion wavelets based rotation, scale and translation invariant texture classification and retrieval," *J. Sci. Ind. Res.*, vol. 70, no. 4, pp. 256–263, 2011.
- [33] Ö. N. Subakan and B. C. Vemuri, "A quaternion framework for color image smoothing and segmentation," *Int. J. Comput. Vis.*, vol. 91, no. 3, pp. 233–250, 2011.
- [34] S.-C. Pei, J.-J. Ding, and J.-H. Chang, "Efficient implementation of quaternion Fourier transform, convolution, and correlation by 2D complex FFT," *IEEE Trans. Signal Process.*, vol. 49, no. 11, pp. 2783–2797, Nov. 2001.
- [35] D. Assefa, L. Mansinha, K. F. Tiampo, H. Rasmussen, and K. Abdella, "Local quaternion Fourier transform and color image texture analysis," *Signal Process.*, vol. 90, no. 6, pp. 1825–1835, Jun. 2010.
- [36] W. Lu, Y. Xu, X. Yang, and L. Song, "Local quaternionic Gabor binary patterns for color face recognition," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Mar./Apr. 2008, pp. 741–744.
- [37] M. Yin, W. Liu, J. Shui, and J. Wu, "Quaternion wavelet analysis and application in image denoising," *Math. Problems Eng.*, vol. 2012, Sep. 2012, Art. ID 493976.
- [38] M. Bahri, "Quaternion algebra-valued wavelet transform," *Appl. Math. Sci.*, vol. 5, no. 71, pp. 3531–3540, 2011.
- [39] P. Ginzberg and A. T. Walden, "Matrix-valued and quaternion wavelets," *IEEE Trans. Signal Process.*, vol. 61, no. 6, pp. 1357–1367, Mar. 2013.
- [40] T. A. Ell and S. J. Sangwine, "Quaternion involutions and anti-involutions," *Comput. Math. Appl.*, vol. 53, no. 1, pp. 137–143, Jan. 2007.
- [41] N. Le Bihan and S. J. Sangwine, "Quaternion principal component analysis of color images," in *Proc. Int. Conf. Image Process. (ICIP)*, vol. 1, Sep. 2003, pp. I-809-1–I-809-12.
- [42] N. L. Bihan and J. Mars, "Singular value decomposition of quaternion matrices: A new tool for vector-sensor signal processing," *Signal Process.*, vol. 84, no. 7, pp. 1177–1199, Jul. 2004.
- [43] P. R. Girard, *Quaternions, Clifford Algebras and Relativistic Physics*. New York, NY, USA: Springer-Verlag, 2007.
- [44] W. R. Hamilton, "On quaternions; or on a new system of imaginaries in algebra," *London, Edinburgh, Dublin Philosoph. Mag. J. Sci.*, vol. 25, no. 163, pp. 10–13, 1844.
- [45] S. L. Altmann, *Rotations, Quaternions, and Double Groups*. New York, NY, USA: Dover, 2005.
- [46] J. Vía, D. Ramírez, and I. Santamaría, "Properness and widely linear processing of quaternion random vectors," *IEEE Trans. Inf. Theory*, vol. 56, no. 7, pp. 3502–3515, Jul. 2010.
- [47] S. J. Sangwine and T. A. Ell, "Hypercomplex auto- and cross-correlation of color images," in *Proc. Int. Conf. Image Process. (ICIP)*, vol. 4, Oct. 1999, pp. 319–322.
- [48] P. M. Kroonenberg, *Three-Mode Principal Component Analysis: Theory and Applications*, vol. 2. Leiden, The Netherlands: DSWO Press, 1983.
- [49] M. E. Kilmer and C. D. Martin, "Factorization strategies for third-order tensors," *Linear Algebra Appl.*, vol. 435, no. 3, pp. 641–658, Aug. 2011.
- [50] J. T. Barron and J. Malik, "Color constancy, intrinsic images, and shape estimation," in *Computer Vision*. Berlin, Germany: Springer-Verlag, 2012, pp. 57–70.
- [51] S. F. Cotter and B. D. Rao, "Sparse channel estimation via matching pursuit with application to equalization," *IEEE Trans. Commun.*, vol. 50, no. 3, pp. 374–377, Mar. 2002.
- [52] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM J. Sci. Comput.*, vol. 20, no. 1, pp. 33–61, 1998.
- [53] Y. C. Pati, R. Rezaifar, and P. S. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," in *Proc. Conf. Rec. 27th Asilomar Conf. Signals, Syst. Comput.*, Nov. 1993, pp. 40–44.
- [54] R. Rubinstein, M. Zibulevsky, and M. Elad, "Efficient implementation of the K-SVD algorithm using batch orthogonal matching pursuit," *Tech. Rep.-CS Tech.*, vol. 40, no. 8, pp. 1–15, 2008.
- [55] K. Skretting and K. Engan, "Recursive least squares dictionary learning algorithm," *IEEE Trans. Signal Process.*, vol. 58, no. 4, pp. 2121–2130, Apr. 2010.
- [56] A. M. Bruckstein, D. L. Donoho, and M. Elad, "From sparse solutions of systems of equations to sparse modeling of signals and images," *SIAM Rev.*, vol. 51, no. 1, pp. 34–81, 2009.
- [57] R. Zeyde, M. Elad, and M. Protter, "On single image scale-up using sparse-representations," in *Curves and Surfaces*. Berlin, Germany: Springer-Verlag, 2012, pp. 711–730.
- [58] Q. Shan, Z. Li, J. Jia, and C.-K. Tang, "Fast image/video upsampling," *ACM Trans. Graph.*, vol. 27, no. 5, p. 153, Dec. 2008.
- [59] J. Sun, Z. Xu, and H.-Y. Shum, "Gradient profile prior and its applications in image super-resolution and enhancement," *IEEE Trans. Image Process.*, vol. 20, no. 6, pp. 1529–1542, Jun. 2011.
- [60] W. T. Freeman, T. R. Jones, and E. C. Pasztor, "Example-based super-resolution," *IEEE Comput. Graph. Appl.*, vol. 22, no. 2, pp. 56–65, Mar./Apr. 2002.
- [61] Y. HaCohen, R. Fattal, and D. Lischinski, "Image upsampling via texture hallucination," in *Proc. IEEE Int. Conf. Comput. Photography (ICCP)*, Mar. 2010, pp. 1–8.
- [62] C. Bao, H. Ji, Y. Quan, and Z. Shen, "L0 norm based dictionary learning by proximal methods with global convergence," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2014, pp. 3858–3865.



and application.

**Yi Xu** received the B.S. and M.S. degrees from the Nanjing University of Science and Technology, Nanjing, China, in 1996 and 1999, respectively, and the Ph.D. degree from Shanghai Jiao Tong University, Shanghai, China, in 2005. She is currently an Associate Professor with the Department of Electronic Engineering, Institute of Image Communication and Network Engineering, Shanghai Jiao Tong University. Her research interests include image processing, intelligent video analysis, and quaternion wavelet theory



**Licheng Yu** received the B.S. degree in electronics engineering from Shanghai Jiao Tong University, Shanghai, China, in 2011, and the M.S. degrees in electrical and computer engineering from Shanghai Jiao Tong University and the Georgia Institute of Technology, Atlanta, GA, USA, in 2014. His research interests are in the field of computer vision, in particular, sparse representation, image classification, and egocentric vision.



**Hongteng Xu** received the B.S. degree from Tianjin University, Tianjin, China, in 2010. In Fall 2010, he joined the Dual-Master Program, Georgia Institute of Technology, Atlanta, GA, USA, and Shanghai Jiao Tong University, Shanghai, China, and received the master's degree in Spring 2013. He is currently pursuing the Ph.D. degree with the School of Electrical and Computer Engineering, Georgia Institute of Technology. His current research interests include image processing, computer vision, and machine learning.



**Hao Zhang** received the B.S. degree in electrical engineering from Shanghai Jiao Tong University (SJTU), Shanghai, China, in 2010, and the M.S. degree in computer science from Colorado State University (CSU), Fort Collins, CO, USA, in 2013, where he is currently pursuing the Ph.D. degree in computer science under the supervision of Dr. R. Beveridge. Since 2010, he has been a Research Assistant with the Computer Vision Laboratory, CSU. His research interest includes computer vision and machine learning with an emphasis on face recognition. His awards and honors include the Outstanding Student Scholarship at SJTU and the Fellowship at CSU.



**Truong Nguyen** is currently a Professor with the Department of Electrical and Computer Engineering, University California at San Diego, La Jolla, CA, USA. His research interests are in 3D video processing and communications and their implementation. He has co-authored (with Prof. Gilbert Strang) a popular textbook entitled *Wavelets and Filter Banks* (Wellesley-Cambridge Press, 1997) and authored several MATLAB-based toolboxes on image compression, electrocardiogram compression, and filter bank design. He has authored over 400 publications. He received the IEEE TRANSACTIONS ON SIGNAL PROCESSING Paper Award (Image and Multidimensional Processing area) for the paper, he co-written with Prof. P. P. Vaidyanathan on linear-phase perfect-reconstruction filter banks (1992). He received the NSF Career Award in 1995 and is currently the Series Editor of *Digital Signal Processing* for Academic Press. He served as an Associate Editor of the IEEE TRANSACTIONS ON SIGNAL PROCESSING from 1994 to 1996, the IEEE SIGNAL PROCESSING LETTERS from 2001 to 2003, the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS from 1996 to 1997 and 2001 to 2004, and the IEEE TRANSACTIONS ON IMAGE PROCESSING from 2004 to 2005.