Commutative Reduced Biquaternions and Their Fourier Transform for Signal and Image Processing Applications

Soo-Chang Pei, *Fellow, IEEE*, Ja-Han Chang, and Jian-Jiun Ding

***Abstract*—Digital signal and image processing using reduced biquaternions (RBs) are introduced in this paper. RBs are an extension of the complex numbers, following the doubling procedure. Two useful representations of RBs ($e_1 - e_2$ form and matrix representation) are discussed in this paper. Besides, we propose a new representation of RBs (the polar form) to calculate the multiplication and conjugation of RBs easily. Furthermore, we define a unique and suitable RB norm and its conjugate. These definitions are similar and compatible with the complex numbers.**

The efficient algorithms of the discrete reduced biquaternion Fourier transform (DRBFT), convolution (DRBCV), correlation (DRBCR), and phase-only correlation are discussed in this paper. In addition, linear-time-invariant and symmetric multichannel complex systems can be easily analyzed by RBs.

For color image processing, we define a simplified RB polar form to represent the color image. This representation is useful to process color images in the brightness-hue-saturation color space. Many different types of color template matching and color-sensitive edge detection (brightness, hue, saturation, and chromaticity matched edges) can be performed simultaneously by RBs.

***Index Terms*—Color image processing, discrete reduced biquaternion Fourier transform, quaternions, reduced biquaternions.**

I. Introduction

THE well-known concept of quaternions was introduced by Hamilton in 1843 [1], [2]. It has been used for signal and color image processing in recent years [3]–[11]. Quaternions are one of the generalizations of complex numbers. A complex number has two components: the real part and the imaginary part. A quaternion, however, has four components, i.e., one real part and three imaginary parts:

$$q = q_r + q_i i + q_j j + q_k k \tag{1}$$

and i, j, k obey the rules as follows:

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

From (2), we find that the multiplication rule of quaternions is not commutative. This problem restricts the applications of quaternions in signal and image processing. Moreover, in general, the convolution of two quaternion signals $f(x,y)$ and $g(x,y)$ cannot be calculated by the product of their Fourier transform $F(u,v)$ and $G(u,v)$ in the frequency domain [16].

$$f(x,y) *_q g(x,y) \neq \text{IQFT}\left(F(u,v)G(u,v)\right). \tag{3}$$

Therefore, it is difficult to analyze of the quaternion LTI system.

On the other hand, the representation of RBs [12]–[14] is

$$q = q_r + q_i i + q_j j + q_k k \tag{4}$$

where

$$ij = ji = k,\ \ jk = kj = i,\ \ ik = ki = -j,\ \ i^2 = k^2 = -1,\ \ j^2 = 1. \tag{5}$$

The multiplication rule of RBs is commutative. This is the unique advantage over quaternions. The implementations of the discrete reduced biquaternion Fourier transform (DRBFT), convolution (DRBCV), and correlation (DRBCR) and the analysis of RB LTI system are much simpler than the existing implementations of quaternions.

The RBs also have their limitations. The algebra of RBs is not a division algebra, and their geometric meaning is unfamiliar to most engineers. However, these have almost no influence on signal and image processing applications. The details are discussed in Section V—the comparison of quaternions and RBs.

Previous research on RBs are briefly reviewed in the following.

In 1853, Hamilton originally proposed *biquaternions*, i.e., quaternions with complex coefficients. These biquaternions, with eight elements, are not commutative in multiplication, and they do not form a division algebra.

In 1990, Schtte and Wenzel [12] suggested *RBs*, in which they have four elements as conventional quaternions, and proposed their applications for the implementation of a digital filter. They showed that a fourth-order real filter can be realized by means of a first-order RB filter [12]. In 1993, Ueda and Takahashi showed that the first-order digital filter with RB coefficients can implement any real coefficient digital filters with orders less than four [13]. From their research, we find the RBs' potential on digital filter design.

In 1992, Ell defined the *double-complex algebra*, which is similar to quaternions but with commutative multiplication [15]. These double-complex algebra signals satisfy

$$f(x,y) *_q g(x,y) = \text{IQFT}\left(F(u,v)G(u,v)\right). \tag{6}$$

Manuscript received March 22, 2002; revised July 13, 2003. This work was supported by the National Science Council, R.O.C. under Contract NSC 91-2219-E-002-044 and by the Ministry of Education under Contract 89-E-FA06-2-4. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Xiang-Gen Xia.

The authors are with the Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan, 10617, R.O.C. (e-mail: pei@cc.ee.ntu.edu.tw).

Digital Object Identifier 10.1109/TSP.2004.828901

In 1996, Davenport proposed the *four-dimensional (4-D) commutative hypercomplex algebras* ($\mathbf{HCA_4}$), which are isomorphic to the group ring C^2, and gave the matrix representation and some interesting properties of this algebra, which are discussed later [21]. In 1998 and 1999, Sommer *et al.* extended the HCA to any 2^{n} dimensions, and they had proven that the 2^{n}-dimensional HCA is isomorphic to $C^{2^{n-1}}$ [8], [20], [22]. Besides, the 4-D HCA is isomorphic to the two-fold tensor product and the Cartesian product of the complex algebra C^2.

In fact, the difference between the three numbers—RBs ($j^2 = 1$), double-complex algebra ($i^2 = 1$) and HCA_4($k^2 = 1$)—is the choice of the square root of 1. Due to the different choice, the properties of these numbers are still similar, but the equations will be different.

The purpose of this paper is to summarize the properties of RBs from previous works and our discoveries and to use them for signal and image processing.

First, we give the new, suitable, and unique definitions of RB norm and conjugation in Section II. These new definitions have similar properties as the ones of complex numbers and are compatible with the complex numbers.

Second, we propose a new representation of RBs (the polar form) in Section III. From the polar form, we calculate the multiplication and conjugation of RBs easily, get many interesting properties of RBs, and understand the meanings of RBs. This polar form and the $e_1 - e_2$ form and matrix representation are useful tools to analyze RBs from different viewpoints.

For color image processing, we use a simplified polar form to represent the color image because the color space is three dimensional (3-D), but the algebra of RBs is a 4-D algebra. This simplified polar form is useful to process color images in a brightness-hue-saturation color space.

Finally, we use the RBs to do the signal processing and use the simplified polar form to do the color template matching and color-sensitive edge detection.

The rest of this paper is organized as follows. The definitions about RBs, such as the $e_1 - e_2$ form and matrix representation, norm, conjugation, etc., are given in Section II. The polar form of RBs and the simplified polar form for color images are proposed in Section III. The implementations of DRBFT, DRBCV, DRBCR, and RB phase-only correlation are given in Section IV. We make comparisons between quaternions and reduced biquaternions in Section V. Then, we apply the RBs for the symmetric multichannel complex system and symmetric lattice filter system analysis in Section VI. We use RBs and the simplified polar form for color template matching and color-sensitive edge detection in brightness-hue-saturation color space in Section VII. Finally, conclusions are made in Section VIII.

In this paper, unless we give a special explanation, we use the names, RB Fourier transform, RB convolution, and RB correlation to represent the discrete reduced biquaternion Fourier transform, convolution, and correlation in the following context, respectively.

We use the following notation:

$\overline{q}$	RB conjugation.
$\lvert q\rvert$	Norm of the RBs.
M_q	Matrix representation of an RB q.
$*_{\mathrm{RB}}$	RB convolution.
$\otimes_{\mathrm{RB}}$	RB correlation.
$\otimes_{\mathrm{RB,PO}}$	RB phase-only correlation.
$H_{(\mathrm{RB})}(p,s)$	RB Fourier transform of $h(m,n)$, type 1 or 2.

II. Definitions

A. Reduced Biquaternions

Before discussing the definition of the reduced biquaternions, we first introduce two related topics: the generalized complex numbers [32], [33] and the doubling procedure [2].

1) Generalized Complex Number: There are three types of the generalized complex numbers proposed in [32] and [33]. These three types are the *ordinary complex numbers* $a + ib$ (where $i^2 = -1$), the *dual numbers* $a + \varepsilon b$ (where $\varepsilon \neq 0$ and $\varepsilon^2 = 0$), and the *double numbers* $a + jb$ (where $j^2 = 1$).

The geometric meanings of these three types of complex numbers are rotation, shear transformation, and Lorentz transformation in a plane, respectively [33].

2) Doubling Procedure: The doubling procedure is used to construct the $2n$-dimensional hypercomplex numbers from two n-dimensional hypercomplex numbers by the following equation:

$$u_{2n} = u_n + v_n e \tag{7}$$

where $u_{2\mathrm{n}}$ is a $2n$-dimensional hypercomplex number, u_{n} and v_{n} are two n-dimensional hypercomplex numbers, and $e^2 = 1$ or -1. For example, we can use the doubling procedure to construct complex numbers from real numbers and quaternions from complex numbers by the following equations:

$$z = a_1 + a_2 i,\ \ q = a + bi + cj + dk = (a + bi) + (c + di)j \equiv z_1 + z_2 j \tag{8}$$

where a_1, a_2, a, b, c, and d are real numbers, z, z_1, and z_2 are complex numbers, and q is a quaternion number.

Following the doubling procedure and using the double numbers, the definition of RBs is

$$q = q_r + q_i i + q_j j + q_k k = (q_r + q_i i) + (q_j + q_k i)j \equiv q_a + q_b j \tag{9}$$

where q_a and q_b are two complex numbers, and

$$ij = ji = k,\ \ jk = kj = i,\ \ ik = ki = -j,\ \ i^2 = k^2 = -1,\ \ j^2 = 1. \tag{10}$$

Although the RBs are isomorphic to the HCA_4, we still use the same name, reduced biquaternions as Schtte and Wenzel because they first used the HCA_4 for signal processing. Moreover, from [22], we see that the RBs are isomorphic to the two-fold tensor product and the Cartesian product of the complex algebra C^2.

Before discussing the other definitions of RBs, we first review the two useful representations of the RB $e_1 - e_2$ form and matrix representation, which were proposed by Davenport in [21].

B. $e_1 - e_2$ Form of RBs

In [21], Davenport found that there exists two special nonzero numbers e_1 and e_2 in HCA_4 such that $e_1 e_2 = 0$ and $e_1^n = e_1^{n-1} = \ldots\ldots = e_1$, $e_2^n = e_2^{n-1} = \ldots\ldots = e_2$. Therefore, e_1 and e_2 are both idempotent elements ($e_1^2 = e_1$, $e_2^2 = e_2$)

and divisors of zero ($e_1e_2 = 0$). (For complex numbers and quaternions, the idempotent elements are only 0 and 1, and the divisor of zero is only the number 0.) From [20, Ch. 1], the geometric meanings of e_1 and e_2 are the null cone of Minkowsky space. However, we only use the algebraic properties of these two numbers in this paper.

For RBs, $e_1 = (1+j)/2$, and $e_2 = (1-j)/2$. Any RB with the form c_1e_1 or c_2e_2 has no inverse element. (c_1 and c_2 are any complex numbers.)

Any RB can be represented by the linear combination of e_1 and e_2

$$q = (q_r+q_ii)+(q_j+q_ki)j = q_a+q_bj \equiv q_{a+b}e_1+q_{a-b}e_2 \quad (11)$$

where $q_{a+b} = q_a + q_b = (q_r + q_j) + (q_i + q_k)i$, and $q_{a-b} = q_a - q_b = (q_r - q_j)+(q_i - q_k)i$. We name (11) the $e_1 - e_2$ form of RBs. This form is the irreducible representation for RBs.

The complexity of many operations about RBs, such as multiplication and Fourier transform, can be reduced by the use of e_1 and e_2, and the analysis about RBs becomes easier. For example, the multiplication of two RBs q_1 and q_2 can be computed by the following equation:

$$q_1q_2 = q_2q_1 = (q_{1a}+q_{1b})(q_{2a}+q_{2b})e_1+(q_{1a}-q_{1b})(q_{2a}-q_{2b})e_2. \quad (12)$$

We only need two instead of four complex multiplications to calculate the multiplication of two RBs [14]. However, the addition operations are increased from 12 to 16. The result is shown in Table I. Moreover, the inverse of an RB q is $q^{-1} = (q_{\mathrm{a+b}})^{-1}\mathrm{e}_1 + (q_{\mathrm{a-b}})^{-1}\mathrm{e}_2$. Therefore, the inverse of an RB exists if $(q_{\mathrm{a+b}}) \neq 0$ and $(q_{\mathrm{a-b}}) \neq 0$.

C. Matrix Representation of an RB

In [21], Davenport gave the matrix representation for HCA_4. For an RB, it has a similar but different representation because the matrix representation is determined by the multiplication rule:

$$1 \to \mathrm{I}_4 \equiv \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$
$$i \to \mathrm{M}_i \equiv \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$
$$j \to \mathrm{M}_j \equiv \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$
$$k \to \mathrm{M}_k \equiv \begin{bmatrix} 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \quad (13)$$

where $\mathrm{M}_j^2 = \mathrm{I}_4$, $\mathrm{M}_i^2 = \mathrm{M}_k^2 = -\mathrm{I}_4$, $\mathrm{M}_i\mathrm{M}_j = \mathrm{M}_j\mathrm{M}_i = \mathrm{M}_k$, $\mathrm{M}_j\mathrm{M}_k = \mathrm{M}_k\mathrm{M}_j = \mathrm{M}_i$, and $\mathrm{M}_i\mathrm{M}_k = \mathrm{M}_k\mathrm{M}_i = -\mathrm{M}_j$.

TABLE I
COMPLEXITY OF TWO REDUCED BIQUATERNIONS' MULTIPLICATION

	Real Multiplications	Real Additions
Direct Calculation	16	12
Calculation by means of e_1 and e_2	8	16

Therefore, the matrix representation of an RB $q = q_r + iq_i + jq_j + kq_k$ is

$$q = q_r+q_i\cdot i+q_j\cdot j+q_k\cdot k \to \mathrm{M_q} \equiv \begin{bmatrix} q_r & -q_i & q_j & -q_k \\ q_i & q_r & q_k & q_j \\ q_j & -q_k & q_r & -q_i \\ q_k & q_j & q_i & q_r \end{bmatrix}. \quad (14)$$

The first column of $\mathbf{M}_q$ is the four elements of the original RB.

Similar to [21], the four eigenvalues of $\mathbf{M}_q$ are

$$\varepsilon = (q_r+q_j)+(q_i+q_k)i, \quad \varepsilon^* = (q_r+q_j)-(q_i+q_k)i$$
$$\eta = (q_r-q_j)+(q_i-q_k)i, \quad \eta^* = (q_r-q_j)-(q_i-q_k)i. \quad (15)$$

Therefore, from (11) and (15), we find that $q_{\mathrm{a+b}}(=\varepsilon)$ and $q_{\mathrm{a-b}}(=\eta)$ in (11) are the two eigenvalues of $\mathbf{M}_q$. This is the relation between the $e_1 - e_2$ form and matrix representation.

Besides, the determinant of $\mathbf{M}_q$ is the product of the above four eigenvalues.

$$\delta \equiv \begin{vmatrix} q_r & -q_i & q_j & -q_k \\ q_i & q_r & q_k & q_j \\ q_j & -q_k & q_r & -q_i \\ q_k & q_j & q_i & q_r \end{vmatrix}$$
$$= \left\{(q_r+q_j)^2+(q_i+q_k)^2\right\}\left\{(q_r-q_j)^2+(q_i-q_k)^2\right\}$$
$$= \left\{\left(q_r^2+q_i^2+q_j^2+q_k^2\right)^2 - 4(q_rq_j+q_iq_k)^2\right\} \geq 0. \quad (16)$$

The conditions for $\delta = 0$ are

$$(q_r = q_j) \cap (q_i = q_k) \text{ or } (q_r = -q_j) \cap (q_i = -q_k). \quad (17)$$

If $\delta = 0$, then the inverse of $\mathbf{M}_q$ and q do not exist.

This matrix representation is very useful to analyze many concepts of RBs, such as its inverse, addition, multiplication, and norm, etc. For example, $(\mathbf{M}_q)^{-1}$ is the matrix representation of the inverse of q; therefore, we can calculate the inverse of q from $(\mathbf{M}_q)^{-1}$.

D. Norm and Conjugation of RBs

If the norm of an RB q is $|q| = (q_r^2 + q_i^2 + q_j^2 + q_k^2)^{1/2}$, then $|q_1q_2| \neq |q_1||q_2|$, where q_1 and q_2 are two arbitrary RBs, and if the conjugation of an RB q is chosen as $\overline{q} \equiv q_r - iq_i - jq_j - kq_k$, then $q\overline{q}$ is still an RB and not a real number. For example, assume that $q_1 = (2+i+j+k)$ and $q_2 = (1-i+k)$; then, $\overline{q}_1 = (2-i-j-k)$, $q_1\overline{q}_1 = (5-2i+2j-2k) \notin R$, and $q_1q_2 = 2+j+2k$, $|q_1| = \sqrt{7}$, $|q_2| = \sqrt{2}$, $|q_1q_2| = 3 \neq \sqrt{14} = |q_1||q_2|$.

Moreover, another three different RB conjugations have been proposed in [12] and [13]. However, neither the product of an RB nor one of these three conjugations are real.

Therefore, we define the norm and conjugation of RBs as follows:

- *Norm of RBs*:

$$|q| \equiv \sqrt[4]{\delta} = \left(\left(q_r^2 + q_i^2 + q_j^2 + q_k^2\right)^2 - 4(q_r q_j + q_i q_k)^2\right)^{\frac{1}{4}} \geq 0 \tag{18}$$

where δ is the determinant of $\mathbf{M}_q$. We define $|q| \equiv \sqrt[4]{\delta}$ because $|q_1 q_2| = |q_1||q_2|$ is satisfied for this definition, and $|q| = (q_r^2 + q_i^2)^{1/2}$ if $q_\mathrm{j} = q_\mathrm{k} = 0$. (This is compatible with complex numbers.)

$|q_1 q_2| = |q_1||q_2|$ can be proven easily by means of the matrix representation.

Proof: Assuming that $q_3 = q_1 q_2$, then $\mathrm{M}_{q3} = \mathrm{M}_{q1}\mathrm{M}_{q2} \rightarrow \det(\mathrm{M}_{q3}) = \det(\mathrm{M}_{q1})\det(\mathrm{M}_{q2})$

$$\rightarrow \delta_3 = \delta_1\delta_2 \quad \text{where} \quad \delta_i = \det(\mathrm{M}_{qi}), \quad i = 1, 2, 3.$$
$$\therefore \quad |q_1 q_2| = |q_3| = \sqrt[4]{\delta_3} = \sqrt[4]{\delta_1\delta_2} = |q_1||q_2|.$$

The only property different from complex numbers is $|q_1 + q_2| > |q_1| + |q_2|$ for some special cases. (This Schwartz triangle inequality is not satisfied.) For example, the two special RB numbers $e_1 = (1 + j)/2$, $e_2 = (1 - j)/2$ have zero norm, but the sum of these two numbers is 1.

- *Conjugation of RBs*:

$$\overline{q} \equiv \sqrt{\delta} \cdot q^{-1} = \frac{|q|^2}{q}. \tag{19}$$

The matrix representation of $\overline{q}$ is $\sqrt{\delta} \cdot (\mathrm{M}_q)^{-1}$, and therefore

$$\overline{q} = \frac{1}{\sqrt{\delta}}\left\{\begin{vmatrix} q_r & q_k & q_j \\ -q_k & q_r & -q_i \\ q_j & q_i & q_r \end{vmatrix} - i\begin{vmatrix} q_i & q_k & q_j \\ q_j & q_r & -q_i \\ q_k & q_i & q_r \end{vmatrix} + j\begin{vmatrix} q_i & q_r & q_j \\ q_j & -q_k & -q_i \\ q_k & q_j & q_r \end{vmatrix} - k\begin{vmatrix} q_i & q_r & q_k \\ q_j & -q_k & q_r \\ q_k & q_j & q_i \end{vmatrix}\right\}. \tag{20}$$

We choose $\overline{q} \equiv \sqrt{\delta} \cdot q^{-1}$ because $q\overline{q}$ is a real number, $q\overline{q} = |q|^2$, and $\overline{q_1 q_2} = \overline{q_1}\,\overline{q_2}$ for this definition. The proof of $\overline{q_1 q_2} = \overline{q_1}\,\overline{q_2}$ is similar to the one of $|q_1 q_2| = |q_1||q_2|$. However, if $\delta = 0$, then the inverse of neither q nor $\overline{q}$ exists.

The only property different from complex numbers is $\overline{q_1 + q_2} \neq \overline{q_1} + \overline{q_2}$ in general. (This is because the conjugation of RBs is a nonlinear operation.)

Example 1—Calculation of the Two RB Conjugation: 1) $q = 1 + 2\mathrm{i} + \mathrm{j} - 2\mathrm{k}$, and 2) $q = 1 + \mathrm{j}$.

1) $\delta = \{(1+1)^2 + (2-2)^2\}\{(1-1)^2 + (2+2)^2\} = 64$

$$\therefore \quad \overline{q} = \frac{1}{8}\left\{\begin{vmatrix} 1 & -2 & 1 \\ 2 & 1 & -2 \\ 1 & 2 & 1 \end{vmatrix} - i\begin{vmatrix} 2 & -2 & 1 \\ 1 & 1 & -2 \\ -2 & 2 & 1 \end{vmatrix} + j\begin{vmatrix} 2 & 1 & 1 \\ 1 & 2 & -2 \\ -2 & 1 & 1 \end{vmatrix} - k\begin{vmatrix} 2 & 1 & -2 \\ 1 & 2 & 1 \\ -2 & 1 & 2 \end{vmatrix}\right\}$$
$$= 2 - i + 2j + k.$$

2) $\delta = \{(1+1)^2 + (0-0)^2\}\{(1-1)^2 + (0+0)^2\} = 0$. Therefore, conjugation of $(1 + j)$ does not exist.

E. Discrete Reduced Biquaternion Fourier Transform(DRBFT)

There are two possible ways to define the reduced biquaternion Fourier transform [16]–[20]:

- *DRBFT of type 1* (two imaginary unit vectors u_1 and u_2):

$$F_{(\mathrm{RB1})}(p, s) = \sum_{m=0}^{M-1}\sum_{n=0}^{N-1} f(m, n)e^{-2\pi\left(u_1\frac{pm}{M} + u_2\frac{sn}{N}\right)}. \tag{21}$$

- *DRBFT of type 2* (one imaginary unit vector u_1):

$$F_{(\mathrm{RB2})}(p, s) = \sum_{m=0}^{M-1}\sum_{n=0}^{N-1} f(m, n)e^{-2\pi u_1\left(\frac{pm}{M} + \frac{sn}{N}\right)} \tag{22}$$

where $f(m, n)$ is a 2-D reduced biquaternion signal, and $u_1^2 = u_2^2 = -1$. For RBs, only i and k are the square root of -1; therefore, $u_1, u_2 \in \{i, k\}$. The properties of different choices of u_1 and u_2 are similar. In this paper, we choose $u_1 = i$ and $u_2 = k$ because this choice is similar to the conventional complex Fourier transform.

The advantage of the type 1 transform is that the even-even, even-odd, odd-even, and odd-odd components of a real signal can be separated in the frequency domain. On the other hand, the advantage of the type 2 transform is simple and similar to the complex Fourier transform.

III. Polar Form of RBs

Both of the complex numbers and quaternions have a polar form, and these polar forms give the geometric meaning and many useful properties for complex numbers and quaternions. However, the polar form for HCA_4 has yet to be discussed. In this section, we first propose the polar form of RBs and discuss the properties of the polar form. Then, we discuss the relations among the three useful representations of RBs, $e_1 - e_2$ form, matrix representation, and polar form. From these relations, we give the geometric meaning of the RBs. Finally, for color image processing, we define a simplified polar form to represent a color image.

A. Polar Form of RBs

Similar to the quaternion, an RB can be uniquely represented by a polar form if $\delta \neq 0$:

$$q = a + bi + cj + dk = Ae^{i\theta_i}e^{k\theta_k}e^{j\theta_j} \tag{23}$$

where $A = |q| = \sqrt[4]{\delta} \geq 0$, $\theta_i \in (-\pi, \pi]$, $\theta_k \in (-\pi/2, \pi/2]$, $\theta_j \in R$, and δ is defined as (16). The proof of (23) is shown as follows.

Step 1) Let $e^{j\cdot\theta_j} = e + fj$; then, $e = \cosh\theta_j$, $f = \sinh\theta_j$, $e > 0$, $e > f$, $e^2 - f^2 = 1$. Therefore

$$q = a + bi + cj + dk = Ae^{i\theta_i}e^{k\theta_k}e^{j\theta_j}$$
$$\Leftrightarrow qeq^{-j\theta_j} = Ae^{i\theta_i}e^{k\theta_k} \tag{24}$$
$$\Leftrightarrow [(ae - cf) + (be - df)i + (ce - af)j + (de - bf)k]$$
$$= A(\cos\theta_i\cos\theta_k + \sin\theta_i\cos\theta_k i - \sin\theta_i\sin\theta_k j + \cos\theta_i\sin\theta_k k) \tag{25}$$

$$\Leftrightarrow \begin{cases} (ae-cf) = A\cos\theta_i\cos\theta_k \equiv a_1 \\ (be-df) = A\sin\theta_i\cos\theta_k \equiv b_1 \\ (ce-af) = -A\sin\theta_i\sin\theta_k \equiv c_1 \\ (de-bf) = A\cos\theta_i\sin\theta_k \equiv d_1 \end{cases}$$
$$(\text{and} \quad e^2 - f^2 = 1). \tag{26}$$

If the solutions of the five unknowns e, f, A, θ_i, and θ_k in (26) exist, then an RB can be represented by the polar form $q = Ae^{i\theta_i}e^{k\theta_k}e^{j\theta_j}$. In the next step, we give the solutions of (26) if they exist and discuss the conditions that the solutions do not exist.

Step 2) *Solutions of the five unknowns in* (26)

a) *Solutions of* e, f, *and* θ_j. From (26), we get

$$-(ae-cf)(ce-af) = (be-df)(de-bf) = A^2\cos\theta_i\cos\theta_k\sin\theta_i\sin\theta_k. \tag{27}$$

Hence, (27) can be used to calculate e and f:

$$(ae-cf)(ce-af) = -(be-df)(de-bf) \tag{28}$$
$$\rightarrow (e^2+f^2)(ac+bd) = ef(a^2+b^2+c^2+d^2) \tag{29}$$

Substitute $f = \pm\sqrt{e^2-1}$

$$\rightarrow (2e^2-1)(ac+bd) = \pm e\sqrt{e^2-1}(a^2+b^2+c^2+d^2).$$

Taking square at both sides

$$\rightarrow \left[(a^2+b^2+c^2+d^2)^2 - 4(ac+bd)^2\right] \times (e^4-e^2) - (ac+bd)^2 = 0$$
$$\therefore \quad e^2 = \frac{1 \pm \sqrt{1 + \frac{4(ac+bd)^2}{(a^2+b^2+c^2+d^2)^2-4(ac+bd)^2}}}{2} = \frac{1 + \frac{(a^2+b^2+c^2+d^2)}{\sqrt{\delta}}}{2} \quad (\because e^2 > 0)$$

where δ is defined as (16). ($\delta = (a^2+b^2+c^2+d^2)^2 - 4(ac+bd)^2$)

$$\rightarrow \quad e = \sqrt{\frac{1 + \frac{(a^2+b^2+c^2+d^2)}{\sqrt{\delta}}}{2}} \quad (\because e > 0). \tag{30}$$

From (29) and $f^2 = e^2 - 1$

$$f = \frac{(2e^2-1)(ac+bd)}{(a^2+b^2+c^2+d^2)e} = \frac{(ac+bd)}{e\sqrt{\delta}}. \tag{31}$$

Hence

$$\theta_j = \tanh^{-1}\frac{f}{e}. \tag{32}$$

b) *Solutions of* A, θ_1, *and* θ_3: From (26), (30), and (31), we get

$$\begin{aligned} a_1 &= ae - cf = A\cos\theta_i\cos\theta_k \\ b_1 &= be - df = A\sin\theta_i\cos\theta_k \\ c_1 &= ce - af = -A\sin\theta_i\sin\theta_k \\ d_1 &= de - bf = A\cos\theta_i\sin\theta_k \end{aligned} \tag{33}$$

and then

$$\theta_i = \tan^{-1}\frac{b_1}{a_1}, \quad \theta_k = \tan^{-1}\frac{d_1}{a_1}, \quad \text{and} \tag{34}$$
$$\begin{aligned} A &= \sqrt{a_1^2+b_1^2+c_1^2+d_1^2} \\ &= \sqrt{(a^2+b^2+c^2+d^2)(e^2+f^2) - 4(ac+bd)ef} \\ &= \sqrt{\frac{(a^2+b^2+c^2+d^2)^2}{\sqrt{\delta}} - \frac{4(ac+bd)^2}{\sqrt{\delta}}} \\ &= \sqrt[4]{\delta} = |q|. \end{aligned} \tag{35}$$

Because the angle ranges of inverse tangent in (34), $(-\pi/2, \pi/2]$, and inverse cosine function in (33), $(0, \pi]$ are not equal, the solutions of θ_i and θ_k found in (34) may not satisfy (33). However, even if the (33) is not satisfied, the difference is only the minus sign. Therefore, this problem can be solved by adjusting the range of θ_i:

If (33) is satisfied, θ_i is the same as the solution in (34); else, we must adjust θ_i to $\theta_i + \text{pi}$ or $\theta_i - \text{pi}$. Consequently, the ranges of the three polar phases are

$$\theta_i \in (-\pi, \pi], \quad \theta_k \in \left(-\frac{\pi}{2}, \frac{\pi}{2}\right], \quad \text{and} \quad \theta_j \in R. \tag{36}$$

We have proved that the solutions of (26) exist and that these solutions are unique if we limit the range of the three polar phases to be as (36). Therefore, an RB can be represented by the polar form in (23), and this representation is unique.

c) *The condition in which the solutions of (26) do not exist* .

The above results cannot be applied when $\delta = 0$. If $\delta = 0$, then e, $f \rightarrow \infty$, and the solutions of (26) do not exist. Consequently, the polar form of an RB does not exist either.

Besides, this RB polar form is compatible with the polar form of complex numbers. Assume an RB $q = a + b\text{i}$ (j and k components equal to 0); then, the θ_k and θ_j equal to 0 and the polar form of q become $q = a + bi = A \cdot e^{i\theta_i}$, where $A = \sqrt[4]{\delta} = \sqrt{a^2+b^2}$.

B. Properties of the Polar Form of RBs

Similar to complex numbers, the polar form of RBs has many useful properties. First, the multiplication of two RBs becomes very easy. Assume that $q_1 = A_1e^{i\theta_{i1}}e^{k\theta_{k1}}e^{j\theta_{j1}}$, $q_2 = A_2e^{i\theta_{i2}}e^{k\theta_{k2}}e^{j\theta_{j2}}$; then, $q_1q_2 = A_1A_2e^{i(\theta_{i1}+\theta_{i2})}e^{k(\theta_{k1}+\theta_{k2})}e^{j(\theta_{j1}+\theta_{j2})}$. Other properties are summarized as follows.

1) *De Moivre's Theorem of RBs*

$$\text{If } q = Ae^{i\theta_i}e^{k\theta_k}e^{j\theta_j}, \text{ then } q^n = A^ne^{i(n\theta_i)}e^{k(n\theta_k)}e^{j(n\theta_j)}. \tag{37}$$

This theorem can be proved directly by the commutative multiplication property of RBs.

2) *Inverse*:

The inverse of an RB can be calculated by the polar form:

$$q^{-1} = A^{-1}e^{-i\theta_i}e^{-k\theta_k}e^{-j\theta_j}. \tag{38}$$

We have introduced three methods to calculate the inverse of an RB: $e_1 - e_2$ form, matrix representation, and polar form. The inverses calculated by these three methods are the same. Besides, the following statements about the inverse of an RB are equivalent.

- The inverse of an RB exists.
- The RB is not equal to c_1e_1 or c_2e_2, where c_1 and c_2 are any complex numbers.
- The inverse of M_q exists (or $\delta \neq 0$).
- The polar form of an RB exists.

3) *Relations between polar Norm of the form of RBs and RB conjugation*

$$\text{If } q = Ae^{i\theta_i}e^{k\theta_k}e^{j\theta_j}, \text{ then } \overline{q} = Ae^{-i\theta_i}e^{-k\theta_k}e^{-j\theta_j}. \tag{39}$$

The conjugation of RBs calculated by (39) is the same as that by (20).

Proof:

a) Letting $q' = Ae^{-i\theta_i}e^{-k\theta_k}e^{-j\theta_j}$, then $qq' = A^2 = |q|^2 = q\overline{q}$. ($\because A = |q|$.)

$$\therefore \quad q(q' - \overline{q}) = 0. \tag{40}$$

b) Because q has a polar form, q must have an inverse. Therefore, the solution of (40) is

$$\overline{q} = q' = Ae^{-i\theta_i}e^{-k\theta_k}e^{-j\theta_j}. \tag{41}$$

4) *Algebraic meaning of $e^{j\theta}$*

Due to $j^2 = 1$, then $e^{j\cdot\theta} \neq \cos\theta + j\cdot\sin\theta$ [32], [33]. In fact

$$e^{j\cdot\theta} = \cosh\theta + j\sinh\theta. \tag{42}$$

Moreover, the following equations are satisfied:

$$\begin{aligned} &e^{j\cdot(-\theta)} = \cosh(-\theta) + j\sinh(-\theta) = \cosh\theta - j\sinh\theta \\ &e^{j\cdot\theta_1}e^{j\cdot\theta_2} = e^{j(\theta_1+\theta_2)} \quad \Rightarrow \quad e^{j\theta}e^{-j\theta} = e^{j(\theta-\theta)} = 1. \end{aligned} \tag{43}$$

5) *Geometric meaning of $e^{j\theta}$*

There are three types of complex numbers proposed in [32] and [33] and is briefly discussed in Section II. Because $e^{j\cdot\theta} = \cosh\theta + j\sinh\theta$, it is the same as the double numbers. Consequently, the geometric meaning of $e^{j\theta}$ is a Lorentz transformation in a plane.

6) *Norm of $e^{j\theta}$*

Because $|q| = A|e^{i\cdot\theta_i}||e^{k\cdot\theta_k}||e^{j\cdot\theta_j}|$ and $A = |q|$, $|e^{j\cdot\theta_2}| = 1 = (\cosh^2\theta_2 - \sinh^2\theta_2)^{1/2}$. This result is the same as the definition of double numbers [32], [33].

C. Relations Among the $e_1 - e_2$ Form, Matrix Representation, and Polar Form

We have discussed the relation between the $e_1 - e_2$ form and matrix representation in Section II. Now, we show the relation between the polar form and the $e_1 - e_2$ form as well as the relation between the polar form and matrix representation.

1) Relations Between Polar Form and $e_1 - e_2$ Form: The $e_1 - e_2$ form of $e^{i\cdot\theta_i}$, $e^{k\cdot\theta_k}$, and $e^{j\cdot\theta_j}$ can be calculated by (11):

$$\begin{aligned} e^{i\theta_i} &= e^{i\theta_i}e_1 + e^{i\theta_i}e_2 \\ e^{k\theta_k} &= e^{i\theta_k}e_1 + e^{-i\theta_k}e_2 \text{ , and} \\ e^{j\theta_j} &= e^{\theta_j}e_1 + e^{-\theta_j}e_2 \end{aligned} \tag{44}$$

respectively. Here, we only prove the result of $e^{i\theta_i}$.

Proof: Let $e^{i\theta_i} = \cos\theta_i + i\sin\theta_i + 0j + 0k \equiv q_a + jq_b$, where $q_a = \cos\theta_i + i\sin\theta_i$, and $q_b = 0$. Then, $q_{a+b} = q_a + q_b = q_a$, and $q_{a-b} = q_a - q_b = q_a$. Hence, from (11), the $e_1 - e_2$ form of $e^{i\cdot\theta_i}$ is

$$e^{i\theta_i} = e^{i\theta_i}e_1 + e^{i\theta_i}e_2.$$

Therefore, the $e_1 - e_2$ form of $q = Ae^{i\theta_i}e^{k\theta_k}e^{j\theta_j}$ is

$$\begin{aligned} Ae^{i\theta_i}e^{k\theta_k}e^{j\theta_j} &= A(e^{i\theta_i}e_1 + e^{i\theta_i}e_2)(e^{i\theta_k}e_1 + e^{-i\theta_k}e_2) \\ &\quad \times (e^{\theta_j}e_1 + e^{-\theta_j}e_2) \\ &= \left[(Ae^{\theta_j})e^{i(\theta_i+\theta_k)}\right]e_1 \\ &\quad + \left[(Ae^{-\theta_j})e^{i(\theta_i-\theta_k)}\right]e_2 \\ &= q_{a+b}e_1 + q_{a-b}e_2 \end{aligned} \tag{45}$$

$$\begin{aligned} &\rightarrow |q_{a+b}| = Ae^{\theta_j}, \quad |q_{a-b}| = Ae^{-\theta_j} \\ &\qquad \text{and} \quad |q_{a+b}||q_{a-b}| = A^2. \end{aligned} \tag{46}$$

2) Relations Between Polar Form and Matrix Representation: The matrix representation of $e^{i\theta_i}$, $e^{k\theta_k}$, and $e^{j\theta_j}$ can be calculated by (14)

$$\begin{aligned} e^{i\theta_i} \longleftrightarrow \mathrm{M}_{\theta_i} &\equiv \begin{bmatrix} \cos\theta_i & -\sin\theta_i & 0 & 0 \\ \sin\theta_i & \cos\theta_i & 0 & 0 \\ 0 & 0 & \cos\theta_i & -\sin\theta_i \\ 0 & 0 & \sin\theta_i & \cos\theta_i \end{bmatrix} \\ &= \begin{bmatrix} \cos\theta_i & -\sin\theta_i & 0 & 0 \\ \sin\theta_i & \cos\theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\ &\quad \times \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \cos\theta_i & -\sin\theta_i \\ 0 & 0 & \sin\theta_i & \cos\theta_i \end{bmatrix}. \end{aligned} \tag{47}$$

$$\begin{aligned} e^{k\theta_k} \longleftrightarrow \mathrm{M}_{\theta_k} &\equiv \begin{bmatrix} \cos\theta_k & 0 & 0 & -\sin\theta_k \\ 0 & \cos\theta_k & \sin\theta_k & 0 \\ 0 & -\sin\theta_k & \cos\theta_k & 0 \\ \sin\theta_k & 0 & 0 & \cos\theta_k \end{bmatrix} \\ &= \begin{bmatrix} \cos\theta_k & 0 & 0 & -\sin\theta_k \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \sin\theta_k & 0 & 0 & \cos\theta_k \end{bmatrix} \\ &\quad \times \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta_k & \sin\theta_k & 0 \\ 0 & -\sin\theta_k & \cos\theta_k & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \end{aligned} \tag{48}$$

$$e^{j\theta_j} \longleftrightarrow \mathrm{M}_{\theta_j} \equiv \begin{bmatrix} \cosh\theta_j & 0 & \sinh\theta_j & 0 \\ 0 & \cosh\theta_j & 0 & \sinh\theta_j \\ \sinh\theta_j & 0 & \cosh\theta_j & 0 \\ 0 & \sinh\theta_j & 0 & \cosh\theta_j \end{bmatrix} = \begin{bmatrix} \cosh\theta_j & 0 & \sinh\theta_j & 0 \\ 0 & 1 & 0 & 0 \\ \sinh\theta_j & 0 & \cosh\theta_j & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cosh\theta_j & 0 & \sinh\theta_j \\ 0 & 0 & 1 & 0 \\ 0 & \sinh\theta_j & 0 & \cosh\theta_j \end{bmatrix}. \quad (49)$$

Therefore, the geometric meaning of $e^{i\theta_i}$ is rotations in both dimensions 1,2 and 3,4, the geometric meaning of $e^{k\theta_k}$ is rotations in both dimensions 1,4 and 2,3, and the geometric meaning of $e^{j\theta_j}$ is the Lorentz transformations in both dimensions 1,3 and 2,4. Hence, the geometric meaning of an RB is one scaling (A), four rotations ($e^{i\theta_i}$ and $e^{k\theta_k}$), and two Lorentz transforms $(e^{j\theta_j})$.

D. Simplified Polar Form—Color Image Representation

The quaternions and RBs are four dimensional algebras, but the color space is a 3-D space. Therefore, if we use a quaternion or an RB to represent a color, the representation is not unique. For quaternions, a color image is generally represented by

$$f(m,n) = f_R(m,n)i + f_G(m,n)j + f_B(m,n)k \quad (50)$$

where $f_R(m,n)$, $f_G(m,n)$, and $f_B(m,n)$ represent the R, G, and B components of the color image, respectively [26]–[28]. However, R-G-B space is not the best choice for all color image processing. For some pattern recognition applications, if the color image is separated into the luminance and chromaticity components (i.e. Intensity-Hue-Saturation I-H-S color space), better performance can be obtained. We will adopt this concept to perform our color image processing by RBs in the I-H-S color space. This I-H-S color space is a well-known and adequate description for human visual color perception. In human vision, brightness represents the perceived light intensity, hue is used to distinguish colors, and saturation is the measurement of the percentage of white light that is added to a pure color.

First, we transform the R,G, and B components $f_R(m,n)$, $f_G(m,n)$, and $f_B(m,n)$, into the I-H-S components $f_I(m,n)$, and $f_H(m,n)$, $f_S(m,n)$ by

$$\begin{bmatrix} f_I(m,n) \\ f_{V_1}(m,n) \\ f_{V_2}(m,n) \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} \\ -\frac{1}{\sqrt{6}} & -\frac{1}{\sqrt{6}} & \frac{2}{\sqrt{6}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & 0 \end{bmatrix} \cdot \begin{bmatrix} f_R(m,n) \\ f_G(m,n) \\ f_B(m,n) \end{bmatrix} \quad (51)$$

$$f_H(m,n) = \tan^{-1}\left(\frac{f_{V_2}(m,n)}{f_{V_1}(m,n)}\right) \text{ and}$$

$$f_S(m,n) = \sqrt{f_{V_1}^2(m,n) + f_{V_2}^2(m,n)}. \quad (52)$$

Second, we use the RBs to represent the color image as follows:

$$f_{\mathrm{RB}}(m,n) = f_A(m,n)e^{if_H(m,n)}e^{kf_\phi(m,n)} \quad (53)$$

where $f_A(m,n)$, $f_H(m,n)$, and $f_\Phi(m,n)$ are the *brightness*, *hue*, and *saturation angle* of the color image, respectively, and

$$f_A(m,n) = \sqrt{f_I^2(m,n) + f_S^2(m,n)} = \sqrt{f_R^2(m,n) + f_G^2(m,n) + f_B^2(m,n)} \quad (54)$$

$$f_\phi(m,n) = \cos^{-1}\left(\frac{f_I(m,n)}{f_A(m,n)}\right) = \sin^{-1}\left(\frac{f_S(m,n)}{f_A(m,n)}\right). \quad (55)$$

The ranges of $f_H(m,n)$ and $f_\Phi(m,n)$ are $-\pi \le f_H(m,n) < \pi$, $-\pi/2 \le f_\phi(m,n) \le \pi/2$.

The geometric meaning of these components in I-H-S color space is shown in Fig. 1.

We call (53) the simplified polar form because the phase of the polar form component $e^{j\theta_j}$ is equal to zero. This representation has the geometric meanings and gives many interesting and beautiful properties, which are summarized as follows.

If an RB $q = q_r + iq_i + jq_j + kq_k$ has the simplified polar form $q = Ae^{i\theta_i}e^{k\theta_k}$, then we have the following.

1) $q_rq_j + q_iq_k = 0$, or $q_r/q_i = (-q_k)/q_j$.
2) The norm of q becomes $|q| = (q_r^2 + q_i^2 + q_j^2 + q_k^2)^{1/2}$.
3) The conjugation of q becomes $\overline{q} = Ae^{-i\theta_i}e^{-k\theta_k} = q_r - q_i + q_j - q_k$.

Properties 2 and 3 come from Property 1. We will use this simplified polar form for the color image processing in Section VII.

IV. Implementation

A. RB Fourier Transform of Type 1 and 2

Similar to the quaternion Fourier transform [16], [20], [23], [24], the RB Fourier transform can be implemented by decomposing the RBFT into a pair of complex Fourier transforms. For the type 1 RBFT, the first step is to decompose the RB signal into the $e_1 - e_2$ form.

$$f(m,n) = f_a(m,n) + f_b(m,n)j \equiv f_{a+b}(m,n)e_1 + f_{a-b}(m,n)e_2. \quad (56)$$

Then, from (21), the type 1 RBFT is as follows.

- *Type 1*:

$$F_{(\mathrm{RB1})}(p,s) = F_{a+b}(p,s)e_1 + F_{a-b}(p,-s)e_2 \quad (57)$$

where $F_{a+b}(p,s) = \mathrm{DFT}(f_{a+b}(m,n))$, $F_{a-b}(p,s) = \mathrm{DFT}(f_{a-b}(m,n))$. The proof of (57) can be found in [20]. Besides, from (22) and (56), the type 2 RB Fourier transform is as follows.

- *Type 2*:

$$F_{(\mathrm{RB2})}(p,s) = \mathrm{DFT}\left(f_a(m,n)\right) + \mathrm{DFT}\left(f_b(m,n)j\right). \quad (58)$$

Thus, two complex 2-D DFTs can be used to implement the RBFT of types 1 or 2. This is the same complexity as the conventional quaternion Fourier transform [16], [23], [24].

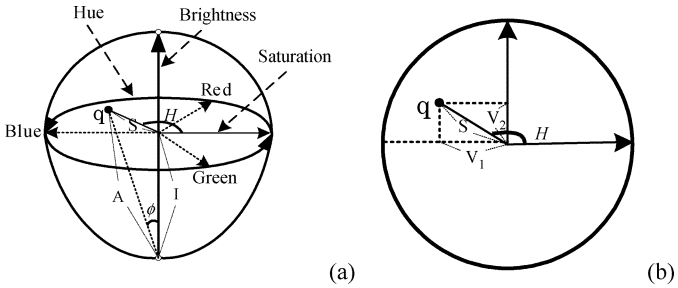

Fig. 1. (a) Geometric meanings of I, H, S, A, and φ in I-H-S color space. (b) Top side view of (a).

We use the same method as above to implement the inverse RBFT of types 1 and 2, except that the roles of $h(m,n)$ and $H_{(\mathrm{RB})}(p,s)$ are exchanged, and in (21) and (22), $-u_1$ and $-u_2$, are replaced by u_1 and u_2.

B. RB Convolution

There are two different definitions for conventional quaternion convolution (QCV): one-side and two-side, as described in [16] and [34]. Nevertheless, these two types reduce to the same one for RBs because the multiplication rule of RBs is commutative. The definition of RB convolution is

$$\begin{aligned} g(m,n) &= f(m,n) *_{\mathrm{RB}} h(m,n) \\ &= \sum_{\tau=0}^{M-1}\sum_{\eta=0}^{N-1} f(m-\tau,n-\eta)h(\tau,\eta). \end{aligned} \tag{59}$$

For quaternions, the quaternion Fourier transform of $g(m,n)$, $G(p,s)$ does not equal to the product of $F(p,s)$ and $H(p,s)$ in the general case. The result of the one-side quaternion convolution by the type 2 quaternion Fourier transform is shown as follows [16]:

$$\begin{aligned} g(m,n) = \mathrm{IQFT}^{(2)} \big(&F_{a(q2)}(p,s)H_{(q2)}(p,s) \\ &+F_{b(q2)}(p,s)H_{(q2)}(-p,-s)j\big). \end{aligned} \tag{60}$$

For RBs, no matter which type of DRBFT is used, the convolution operation in the spatial domain is equal to the product operation in the frequency domain. The proof can be found in [20].

$$g(m,n) = \mathrm{IDRBFT}\left(F_{(\mathrm{RB})}(p,s)H_{(\mathrm{RB})}(p,s)\right). \tag{61}$$

By (57) and (58), we totally require *six complex 2-D FTs* to implement the convolution. The complexity of RB convolution is equal to the one of quaternions [16]. Nevertheless, the algebra of RBs is simpler and more suitable for filter design. RBs satisfy (61) but not for quaternions. This advantage is very useful for the design and analysis of linear time-invariant (LTI) system.

In [16], we have shown that it is difficult to analyze the combination of quaternion filters in cascade series. However, for RBs, the results are very simple and the same as the complex LTI system. The total frequency response of cascade-connection filters is the multiplication of the frequency response of each filter. In Fig. 2, we combine the RB LTI systems in *series*; then, the relation between the input signal $f(x,y)$ and output signal $g(x,y)$ can be expressed as

$$g(m,n) = f(m,n) *_{\mathrm{RB}} h_s(m,n) \tag{62}$$

where

$$h_s(m,n) = h_1(m,n) *_{\mathrm{RB}} h_2(m,n) *_{\mathrm{RB}} \cdots\cdots *_{\mathrm{RB}} h_T(m,n). \tag{63}$$

In the frequency domain, the relation between $F_{(\mathrm{RB})}(p,s)$ and $G_{(\mathrm{RB})}(p,s)$ is

$$G_{(\mathrm{RB})}(p,s) = F_{(\mathrm{RB})}(p,s)H_{S(RB)}(p,s), \quad \text{where}$$
$$H_{S(RB)}(p,s) = \prod_{t=1}^{T} H_{t(RB)}(p,s). \tag{64}$$

The cascade result of quaternion system can be found in [16]. It is very complicated.

On the other hand, if we combine the RB LTI systems in *parallel*, then the relation between $F_{(\mathrm{RB})}(p,s)$ and $G_{(\mathrm{RB})}(p,s)$ is the same as the one of quaternions in [16]:

$$G_{(\mathrm{RB})}(p,s) = F_{(\mathrm{RB})}(p,s)H_{P(\mathrm{RB})}(p,s), \quad \text{where}$$
$$H_{P(\mathrm{RB})}(p,s) = \sum_{t=1}^{T} H_{t(\mathrm{RB})}(p,s). \tag{65}$$

Therefore, using RBs for LTI system analysis and design is much simpler than quaternions.

In [15], Ell derived the LTI analysis using the double complex algebra. The convolution operation of the double complex algebra in the spatial domain is equivalent to a product operation in the frequency domain as RBs. Therefore, the LTI system analysis of double complex algebra is the same as the one of RBs.

C. RB Correlation and Phase-Only Correlation

1) Correlation: The definition of the RB correlation is

$$\begin{aligned} g(m,n) &= f(m,n) \otimes_{\mathrm{RB}} h(m,n) \\ &= \sum_{\tau=0}^{M-1}\sum_{\eta=0}^{N-1} f(\tau,\eta)\overline{h(\tau-m,\eta-n)}. \end{aligned} \tag{66}$$

Correlation can be viewed as a special case of convolution

$$f(m,n) \otimes_{\mathrm{RB}} h(m,n) = f(m,n) *_{\mathrm{RB}} \overline{h(-m,-n)} \tag{67}$$

so we just use the algorithms of convolution to implement correlation.

$$g(m,n) = \mathrm{IDRBFT}\left(F(p,s)H_{c-}(p,s)\right) \tag{68}$$

where $F(p,s)$ and $H_{c-}(p,s)$ are the RB Fourier transform of $f(m,n)$ and $h(-m,-n)$, respectively. (The subscript $_{c-}$ means conjugation and time reverse). Equation (68) is satisfied, no matter which type of DRBFT is used.

2) Phase-Only Correlation [29]: The definition of the RB phase-only correlation is

$$\begin{aligned} \hat{g}(m,n) &\equiv f(m,n) \otimes_{\mathrm{RB,PO}} h(m,n) \\ &\equiv \mathrm{IDRBFT}\left(\frac{F(p,s)H_{c-}(p,s)}{|F(p,s)|\,|H_{c-}(p,s)|}\right). \end{aligned} \tag{69}$$

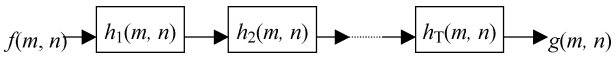

Fig. 2. Cascade combination the RBLTI systems in series.

For color image processing in Section VII, we use the simplified polar form to represent the color image and use (68) and (69) to do the correlation and phase-only correlation, respectively.

V. Comparison of Quaternions and RBs

Before describing the applications of reduced biquaternions for signal and image processing, we make comparisons between quaternions and RBs in Table II. The advantages of quaternions are their clear geometric meaning and division algebra. The advantages of RBs are commutative multiplication and simple convolution equation in the frequency domain.

Both of the quaternions and RBs have unique but different definitions of norm and conjugation and have individual matrix representation and polar form.

The problems of RBs are discussed in the following.

1) Unfamiliar geometric meaning:
   A quaternion can be regarded as a rotation in 3-D space. For RBs, from the polar form and (47)–(49), an RB can be regarded as rotations and Lorentz transforms in 4-D space. This property is related to the "Special theory of Relativity" but is unfamiliar to most engineers.
2) Not a division algebra:
   For RBs, there are no solutions of the variable x in the following equation:

$$ux = 1, \quad \text{if } u = c_1 e_1 \text{ or } c_2 e_2 \tag{70}$$

   and there are infinite solutions of the variable x in the following equation:

$$ux = 0, \quad \text{if } u = c_1 e_1 \text{ or } c_2 e_2. \tag{71}$$

   Consequently, the reduced biquaternion system is not a division system.

Although reduced biquaternions have unfamiliar geometric meaning and are not a division algebra, these disadvantages have almost no influence on the following practical applications. Commutative multiplication and the existence of convolution theorem are very useful for signal and image processing.

VI. Applications for Signal Processing

We can extend RBs to a higher dimension by the doubling procedure as in [4], [20], and [22]. For example, the eight-dimensional commutative hypercomplex algebra HCA_8 is $g_{\mathrm{H},8} = q_1 + q_2 i_4$, where q_1 and q_2 are two RBs, and $i_4^2 = 1$. We use these algebras to analyze the symmetric multichannel system.

A. Using Hypercomplex Algebras to Analyze Symmetric Multichannel System

First, we consider a symmetric 2^{n} channel complex system. Assuming that the input and output complex signals are $f_i(m)$ and $g_i(m)$, respectively, $(\mathrm{i} = 0, 1, \ldots, (2^{\mathrm{n}} - 1))$ and the complex impulse response between channels x and y are $h_{x \oplus y}(m)$, where "$\oplus$" means the binary "exclusive-or" operation. For example, $2 \oplus 6 = (010)_2 \oplus (110)_2 = (100)_2 = 4$, where $()_2$ means the binary bit representation.

- For $\mathrm{n} = 1$ [a symmetric two channel system as Fig. 3(a)]

$$\begin{aligned} g_0(m) &= f_0(m) * h_0(m) + f_1(m) * h_1(m) \\ g_1(m) &= f_0(m) * h_1(m) + f_1(m) * h_0(m). \end{aligned} \tag{72}$$

  We need four conventional convolutions to implement the system.

  However, if we set three RB signals $f_{\mathrm{RB}}(m,n)$, $h_{\mathrm{RB}}(m,n)$, and $g_{\mathrm{RB}}(m,n)$ as follows:

$$\begin{aligned} f_{\mathrm{RB}}(m) &= f_0(m) + f_1(m) j \\ h_{\mathrm{RB}}(m) &= h_0(m) + h_1(m) j \\ g_{\mathrm{RB}}(m) &= g_0(m) + g_1(m) j \end{aligned} \tag{73}$$

  then

$$g_{\mathrm{RB}}(m,n) = f_{\mathrm{RB}}(m,n) *_{\mathrm{RB}} h_{\mathrm{RB}}(m,n). \tag{74}$$

  We just need to use one RB convolution to calculate the output signals. From Section IV-B, an RB convolution equals to two conventional convolutions in implementation. Therefore, we only need two instead of four conventional convolutions to calculate the output signals. Besides, from (72), we get

$$\begin{aligned} g_0(m,n) \pm g_1(m,n) = (f_0(m,n) \pm f_1(m,n)) \\ * (h_0(m,n) \pm h_1(m,n)). \end{aligned} \tag{75}$$

- For $\mathrm{n} = 2$ [a symmetric four channel system as Fig. 3(b)]

$$g_k(m) = \sum_{i=0}^{3} f_i(m) * h_{i \oplus k}(m), \quad (k = 0, 1, 2, 3). \tag{76}$$

  If we set three eight-dimensional commutative hypercomplex number signals, $f_{H,8}(m)$, $h_{H,8}(m)$, $g_{H,8}(m)$ as follows:

$$\begin{aligned} f_{\mathrm{H},8}(m) &= f_0(m) + f_1(m) i_2 + f_2(m) i_4 + f_3(m) i_6 \\ h_{\mathrm{H},8}(m) &= h_0(m) + h_1(m) i_2 + h_2(m) i_4 + h_3(m) i_6 \\ g_{\mathrm{H},8}(m) &= g_0(m) + g_1(m) i_2 + g_2(m) i_4 + g_3(m) i_6 \end{aligned} \tag{77}$$

  then $g_{\mathrm{H},8}(m) = f_{\mathrm{H},8}(m) *_{\mathrm{H},8} h_{\mathrm{H},8}(m)$ where $*_{\mathrm{H},8}$ means the convolution of the eight-dimensional commutative hypercomplex signals. These will be very useful in symmetric communication channel applications such as symmetric coaxial cables and transmission lines, etc.
- Symmetric lattice filter.

  We also use RBs to analysis and design symmetric complex lattice filter system as in Fig. 4. If we set three RB signals f, g, and h_m and six complex singals f_a, f_b, g_a, g_b, h_{ma}, and h_{mb} as follows:

$$\begin{aligned} f &= f_1 + f_2 j, \quad f_a = f_1 + f_2, \quad f_b = f_1 - f_2 \\ g &= g_1 + g_2 j, \quad g_a = g_1 + g_2, \quad g_b = g_1 - g_2 \end{aligned} \tag{78}$$

$$\begin{aligned} h_m &= h_{m,1} + h_{m,2} j, \quad h_{m,a} = h_{m,1} + h_{m,2} \\ h_{m,b} &= h_{m,1} - h_{m,2}, \quad \text{where } (\mathrm{m} = 1, 2, \ldots, \mathrm{n}) \end{aligned} \tag{79}$$

TABLE II
COMPARISON BETWEEN QUATERNIONS AND RBS ON THEIR PROPERTIES

Properties	Quaternions	Reduced biquaternions
1. Geometric meaning	A quaternion means a 3D rotation.	A RB means a 4D transform composed by rotations and Lorentz transforms.
2. Division algebra	Yes	No
3. Definitions of norm and conjugation	$\|q\|=(q_r^2+q_i^2+q_j^2+q_k^2)^{1/2}$ $\overline{q} = q_r - i\, q_i - j\, q_j - k\, q_k$	$\|q\|=\sqrt[4]{\delta}=((q_r^2+q_i^2+q_j^2+q_k^2)^2-4(q_iq_j+q_iq_k)^2)^{1/4}$ $\overline{q} \equiv \sqrt{\delta} q^{-1} = \|q\|^2 / q$
4. Multiplication Rule	Noncommutative	Commutative
5. Convolution	$F(f(t)*g(t)) \neq F(w)G(w)$	$F(f(t)*g(t)) = F(w)G(w)$

and use the results of (74) and (75), then we get

$$\begin{aligned} g &= f *_{\mathrm{RB}} h_1 *_{\mathrm{RB}} h_2 *_{\mathrm{RB}} \ldots *_{\mathrm{RB}} h_n \\ g_a &= f_a * h_{1,a} * h_{2,a} * \ldots * h_{n,a}, \text{ and} \\ g_b &= f_b * h_{1,b} * h_{2,b} * \ldots * h_{n,b}. \end{aligned} \tag{80}$$

Therefore, the system is equivalent to the one in Fig. 4(b), and the output signals are

$$g_1 = \frac{(g_a + g_b)}{2}, \quad g_2 = \frac{(g_a - g_b)}{2}. \tag{81}$$

The complexity is greatly reduced, and the design and analysis become very easy by using RBs.

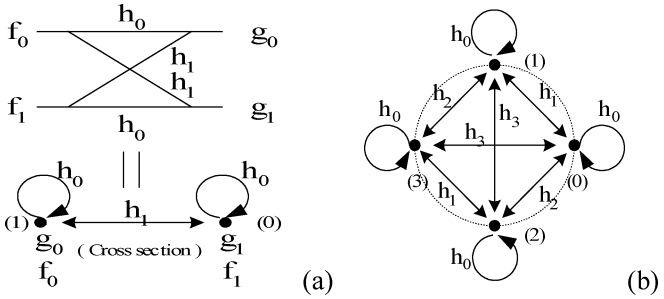

Fig. 3. System block diagram of the simplest two symmetric multichannel complex system. One point represents one channel. (a) Two channels. (b) Four channels.

VII. APPLICATIONS FOR COLOR IMAGE PROCESSING

As the discussion in Section III-D, we perform our color image processing by RBs with the simplified polar form in the I-H-S color space. In general, a color image is digitized in R-G-B format. Therefore, we first transform the color image into I-H-S color space by (51) and (52). Then, we use (53) to represent all the color images we use in this section.

A. Color Template Matching

In [20, Ch. 7], Labunets used quaternions to represent the color image as (50) and calculate the hypercomplex-valued moments of the color image to do pattern recognition. In [11], Pei proposed that five different color template matching results can be done by using quaternion correlation. Here, we use RBs with simplified polar form and RB correlation to further improve the performance in [11]. The color template match steps are summarized as follows:

Step 1) Transform the reference pattern $h_o(m,n)$ and input image $f_o(m,n)$ into I-H-S color space by (51). Then, we use (53) to represent the color images.

$$\begin{aligned} h(m,n) &\equiv A_h(m,n) e^{iH_h(m,n)} \cdot e^{k\phi_h(m,n)} \\ f(m,n) &\equiv A_f(m,n) e^{iH_f(m,n)} \cdot e^{k\phi_f(m,n)} \end{aligned} \tag{82}$$

where $h(m,n)$ and $f(m,n)$ are the RB representation of the reference pattern and input image, respectively.

Step 2) Calculate the energy of reference pattern $h(m,n)$ by (18):

$$E_h = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} (|h(m,n)|)^2 = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} A_h^2(m,n) \tag{83}$$

and divide the reference pattern $h(m,n)$ by E_h : $h_n(m,n) = h(m,n)/E_h$

(After dividing by E_h, if input image fully matches the reference pattern, then the correlation values at the matching positions are equal to 1 $(= 1 + 0\mathrm{i} + 0\mathrm{j} + 0\mathrm{k})$.)

Step 3) Compute the RB Fourier transform of $f(m,n)$ and $h_n(m,n)$

$$\begin{aligned} F(p,s) &= \mathrm{DRBFT}\,(f(m,n)) \text{ and} \\ Hn(p,s) &= \mathrm{DRBFT}\,(hn(m,n)). \end{aligned} \tag{84}$$

Step 4) Compute the RB correlation and phase-only correlation of $f(m,n)$ and $h_n(m,n)$ at the same time by using the RB Fourier transform and (68) and (69):

$$g(m,n) \equiv \mathrm{IDRBFT}\left\{F_{(\mathrm{RB})}(p,s) H_{c-,n(\mathrm{RB})}(p,s)\right\}. \tag{85}$$

$$\hat{g}(m,n) \equiv \mathrm{IDRBFT}\left\{\frac{F_{(\mathrm{RB})}(p,s) H_{c-,n(\mathrm{RB})}(p,s)}{\left|F_{(\mathrm{RB})}(p,s)\right| \left|H_{c-,n(\mathrm{RB})}(p,s)\right|}\right\}. \tag{86}$$

(Requirement 1)

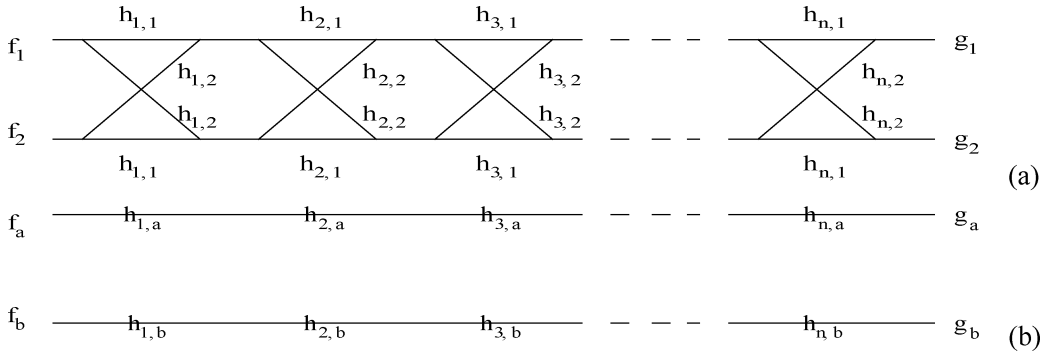

Fig. 4. (a) System block diagram of symmetric lattice filter. (b) Equivalent system block diagram.

Step 5) *Using the result of the phase-only correlation* $\hat{q}(m,n)$ *to find the positions of the objects that have the same shape and size as the reference pattern.*

The phase-only correlation of two images is the inverse Fourier transform of the normalized cross-power spectrum. The heights and positions of the output peaks will measure the similarity and spatial shift between these two images. So, from $\hat{g}(m,n)$, we find the positions of the objects that have the *same shape* and *size* as the reference pattern by selecting appropriate threshold. Assume that these matched positions are at (m_s, n_s).

Step 6) *From the phase-only correlation result* $\hat{g}(m,n)$ *and step 5, we have found the candidate positions* (m_s,n_s) *of shape and size matched objects. Then, to determine if the average of brightness, chromaticity, hue, or saturation of these objects are the same as the one of the reference pattern, we use the values at the candidate positions* (m_s, n_s) *in the correlation result* $g(m,n)$*and the following four requirements.*

- Requirement 2 (for *average brightness* match):

$$c_1 < |g(m_s,n_s)| < c_2 \tag{87}$$

where $c_1 < 1 < c_2$, and c_1, c_2 are all near to 1.

- Requirement 3 (for *average chromaticity* match):

$$\rho_1 \geq d_1 \quad (d_1 \approx 1 \text{ and } d_1 < 1) \tag{88}$$

where

$$\rho_1 = \frac{|g_r(m_s,n_s)|}{|g_r(m_s,n_s)|+|g_i(m_s,n_s)|+|g_j(m_s,n_s)|+|g_k(m_s,n_s)|} \tag{89}$$

and $g_r(m_s,n_s)$, $g_i(m_s,n_s)$, $g_j(m_s,n_s)$ and $g_k(m_s,n_s)$ mean the real-part, i-part, j-part, and k-part of $g(m_s,n_s)$

- Requirement 4 (for *average Hue* match):

$$\rho_2 \geq d_2 \quad (d_2 \approx 1 \text{ and } d_2 < 1) \tag{90}$$

where

$$\rho_2 = \frac{|g_r(m_s,n_s)|+|g_k(m_s,n_s)|}{|g_r(m_s,n_s)|+|g_i(m_s,n_s)|+|g_j(m_s,n_s)|+|g_k(m_s,n_s)|}. \tag{91}$$

- Requirement 5 (for *average saturation* match):

$$\rho_3 \geq d_3, \quad (d_3 \approx 1 \text{ and } d_3 < 1) \tag{92}$$

where

$$\rho_3 = \frac{|g_r(m_s,n_s)|+|g_i(m_s,n_s)|}{|g_r(m_s,n_s)|+|g_i(m_s,n_s)|+|g_j(m_s,n_s)|+|g_k(m_s,n_s)|}. \tag{93}$$

These requirements are only applied to the candidate positions found by the phase-only correlation in step 5 and not for all the positions in the image.

Step 7 *True Match Test*: The results found in step 6 are only for average matches of brightness, chromaticity, hue, or saturation because these average matches will also satisfy the requirements. To find the true match of brightness, hue, and saturation, we need to calculate the exact brightness difference (Δ_{A}), hue difference (Δ_{H}) or saturation difference (Δ_{Φ}) between the reference pattern and the candidate objects with shape and size match.

$$\begin{aligned}
\Delta_A &= \frac{1}{P}\sum_{m,n\in R} |h_A(m,n) - f_A(m+m_s,n+n_s)| \\
\Delta_H &= \frac{1}{P}\sum_{m,n\in R} |h_H(m,n) - f_H(m+m_s,n+n_s)| \\
\Delta_\phi &= \frac{1}{P}\sum_{m,n\in R} |h_\phi(m,n) - f_\phi(m+m_s,n+n_s)|
\end{aligned} \tag{94}$$

where R is the region of the reference pattern, and P is the number of pixels in the region R. If Δ_{A}, Δ_{H}, and Δ_{Φ} is smaller than some threshold $\mathrm{T_A}$, $\mathrm{T_H}$ and T_{Φ}, respectively, then this object at $(m_{\mathrm{s}}, n_{\mathrm{s}})$ has the same brightness, hue, and saturation as the one of the reference pattern, respectively. Otherwise, the object has only the average match but not perfect match.

As in step 6, this step only applies to the positions found by phase-only correlation.

- Explanation of Requirement 2–5
  From Section III-D

$$\overline{h_n(\tau,\eta)} = \frac{A_h(\tau,\eta)e^{-iH_h(\tau,\eta)}e^{-k\phi_h(\tau,\eta)}}{E_h}. \tag{95}$$

Assuming that $f'(\tau,\eta) = f(\tau+m,\eta+n) \equiv A_{f'}(\tau,\eta)e^{iH_{f'}(\tau,\eta)}e^{k\phi_{f'}(\tau,\eta)}$, then the correlation result becomes

$$\begin{aligned} g(m,n) &= \sum_{\tau=0}^{M-1}\sum_{\eta=0}^{N-1} f(\tau,\eta)\overline{h_n(\tau-m,\eta-n)} \\ &= \sum_{\tau=0}^{M-1}\sum_{\eta=0}^{N-1} f(\tau+m,\eta+n)\overline{h_n(\tau,\eta)} \\ &= \frac{1}{E_h}\sum_{\tau=0}^{M-1}\sum_{\eta=0}^{N-1} A_h(\tau,\eta)A_{f'}(\tau,\eta) \\ &\quad \times e^{i\left(H_{f'}(\tau,\eta)-H_h(\tau,\eta)\right)}e^{k\left(\phi_{f'}(\tau,\eta)-\phi_h(\tau,\eta)\right)} \\ &\equiv \frac{1}{E_h}\sum_{\tau=0}^{M-1}\sum_{\eta=0}^{N-1} A_h(\tau,\eta)A_{f'}(\tau,\eta) \\ &\quad \times e^{i\Delta H(\tau,\eta)}e^{k\Delta\phi(\tau,\eta)} \\ &\equiv g_r(m,n)+i\cdot g_i(m,n)+j\cdot g_j(m,n) \\ &\quad + k\cdot g_k(m,n) \end{aligned} \tag{96}$$

where $\Delta H(\tau,\eta) = H_{f'}(\tau,\eta) - H_h(\tau,\eta)$, and $\Delta\phi(\tau,\eta) = \phi_{f'}(\tau,\eta) - \phi_h(\tau,\eta)$.

In the following cases, we assume that $f(m,n)$ has an object that has *the same shape and size* as the reference pattern at the position (m_s, n_s), and $\Delta H(m,n)$ and that $\Delta\phi(m,n)$ are the hue and saturation differences between the object and the reference pattern.

Case 1) *(Average full match)*:

If $f(m,n)$ is just the *space-shift* version of the reference pattern

$$f(m,n) = h(m-m_s, n-n_s) \tag{97}$$

then from (96), the correlation result at (m_s, n_s) is

$$\begin{aligned} g(m_s,n_s) &= \frac{1}{E_h}\sum_{\tau=0}^{M-1}\sum_{\eta=0}^{N-1} A_h^2(\tau,\eta) = 1 \\ &\qquad (\because\ f'(\tau,\eta) = h(\tau,\eta) \text{ at } (m_s,n_s)) \end{aligned} \tag{98}$$

$$\begin{aligned} \Rightarrow\ & g_r(m_s,n_s) = 1 \\ & g_i(m_s,n_s) = g_j(m_s,n_s) = g_k(m_s,n_s) = 0. \end{aligned} \tag{99}$$

Therefore, $|g(m_s,n_s)| = 1$, $\rho_1 = 1$, and (87), and (88) are satisfied.

Case 2) *(Average brightness match)*:

In (96), if $\Delta H(m,n)$ and $\Delta\phi(m,n)$ are constant ($\Delta H(m,n) = \Delta H$ and $\Delta\phi(m,n) = \Delta\phi$), then

$$\begin{aligned} g(m_s,n_s) &= e^{i\cdot\Delta H}e^{k\cdot\Delta\phi}\frac{1}{E_h} \\ &\quad \times \sum_{\tau=0}^{M-1}\sum_{\eta=0}^{N-1} A_{f'}(\tau,\eta)A_h(\tau,\eta). \end{aligned} \tag{100}$$

$$\begin{aligned} |g(m_s,n_s)| &= |e^{i\cdot\Delta H}||e^{k\cdot\Delta\phi}| \\ &\quad \times \left|\frac{1}{E_h}\sum_{\tau=0}^{M-1}\sum_{\eta=0}^{N-1} A_{f'}(\tau,\eta)A_h(\tau,\eta)\right| \\ &= \frac{\left|\sum_{\tau=0}^{M-1}\sum_{\eta=0}^{N-1} A_{f'}(\tau,\eta)A_h(\tau,\eta)\right|}{\sum_{\tau=0}^{M-1}\sum_{\eta=0}^{N-1} A_h^2(\tau,\eta)}. \end{aligned} \tag{101}$$

Moreover, if $f'(\tau,\eta) = ch(\tau,\eta)$, then $|g(m_s,n_s)| = c$. Besides, if $c \approx 1$, then we conclude that the brightness are matched. Consequently, we use $|g(m_s,n_s)|$ to represent the brightness ratio.

Two special conditions in which $\Delta H(m,n)$ and $\Delta\phi(m,n)$ are constant are the following.

1) The color of the object and the reference pattern are the same. ($\Delta H = \Delta\phi = 0$)
2) Each of the object and the reference pattern has only one color.

If $\Delta H(m,n)$ and $\Delta\phi(m,n)$ are not constant, using $|g(m_s,n_s)|$ to represent the brightness ratio may not be accurate. However, if the variations of $\Delta H(m,n)$ and $\Delta\phi(m,n)$ are not large, the results are still valid.

TABLE III
REQUIREMENTS FOR THE CLASSIFICATION OF COLOR PATTERN RECOGNITION RESULTS

Satisfying Requirement	1	Shape and size match
	1,2 and Step 7	Shape and size match and **true brightness** match
	1,3 and Step 7	Shape and size match and **true chromaticity** match
	1,4 and Step 7	Shape and size match and **true hue match**
	1,5 and Step 7	Shape and size match and **true saturation match**

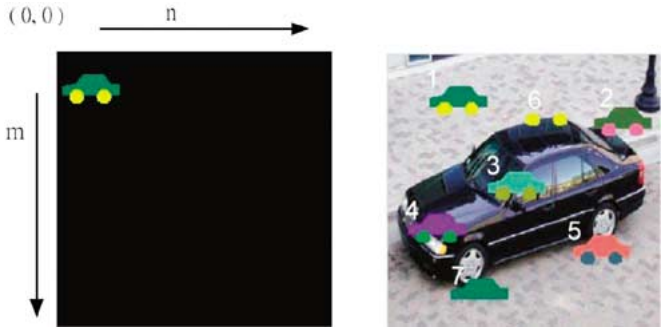

Fig. 5. Reference pattern and input objects of Experiment 1.

TABLE IV
COLOR OF THE CARS IN THE INPUT IMAGE OF EXPERIMENT 1

Numbers of 5 cars	'body color of the car'		'wheel color of the car'	
	(RGB values)	(A,H, ϕ values)	(RGB values)	(A,H, ϕ values)
1	(15, 106, 73)	(**130, -9 π /20, π /6**)	(207, 207, 66)	(**300, π , π /8**)
2	(65, 99, 53)	(**130**, -3 π /4, π /12)	(230, 99, 165)	(**300**, π /2, π /10)
3	(94, 188, 154)	(260, **-9 π /20**, π /12)	(139, 139, 33)	(200, π , π /7)
4	(130, 38, 222)	(260, π /6, **π /6**)	(49, 126, 66)	(150, -3 π /5, **π /8**)
5	(227, 89, 89)	(260, 2 π /3, π /7)	(53, 92, 106)	(150, - π /4, π /12)

Case 3) *(Average chromaticity match)*:

If the object has *the same chromaticity* as the reference pattern, then

$$\Delta H(\tau,\eta) = \Delta\phi(\tau,\eta) = 0 \tag{102}$$

$$g(m_s,n_s) = \frac{1}{E_h}\sum_{\tau=0}^{M-1}\sum_{\eta=0}^{N-1} A_{f'}(\tau,\eta)A_h(\tau,\eta) \tag{103}$$

$$\Rightarrow g_i(m_s,n_s) = g_j(m_s,n_s) = g_k(m_s,n_s) = 0$$

$$\Rightarrow \rho_1 = 1. \tag{104}$$

Therefore, (88) are satisfied, and we use requirement 3 to test the chromaticity match. The *hue and saturation match* can be proved by the same step as chromaticity match.

In fact, the above proof is incomplete because satisfying requirements 2–5 does not guarantee the true match of brightness, chromaticity, hue, and saturation. It only guarantees the average match. Hence, we need another step to judge if the average match is a true match or not.

Therefore, we use the requirements 1–5 and step 7 to do color template match, as shown in Table III. From Table III, we detect the objects that have exactly the same a) shape and size, b) chromaticity, c) brightness, d) hue, or e) saturation, with the reference pattern. When the shape and size are matched, we use whether its brightness, hue, or saturation are matched to obtain the different classification results. These different results can be obtained at the same time.

Experiment 1—Color Template Matching 1 for Artificial Object Match: We use images in Fig. 5 to illustrate our method. We use a car with two different colors as the reference pattern shown in Fig. 5(a), and an input image containing seven cars in a natural background is shown in Fig. 5(b). Five cars (nos. 1–5) have the same shape and size as reference pattern, but the other two cars (nos. 6 and 7) do not. The colors of the five cars (nos. 1–5) are listed in Table IV. Car no. 1 is the same as the reference car. Cars no. 2–4 have the same brightness, hue, and saturation as the reference car, respectively. The color of car no. 5 is different from the one of the reference car. Car no. 6 only has two wheels, and car no. 7 only has the body of the car but without wheels.

- *Fully matched pattern:*

  Then, we do the color template matching in Fig. 5(b) by following the process described in Section VII-B1, and the results of Steps 4 and 5 are shown in Fig. 6(a)–(c), respectively, where ρ_1 is defined as (89). In (87) and (88), if we choose the thresholds close to 1($c_1 = 0.9$, $c_2 = 1.1$, $d_1 = 0.9$), then only one car (at left-upper) is detected, and its *shape, size, brightness, and chromaticity* are all the same as those of the reference pattern. However, if the input image contains the reference object and noise, then there are some small peaks in the result of the phase-only correlation, and we remove them easily by selecting appropriate threshold (an example would be half of the maximal peak).
- *Partially matched pattern (brightness, hue, saturation)*

  If we just use one of the requirements 2, 4, or 5, then the cars with the brightness, or hue, or saturation close to

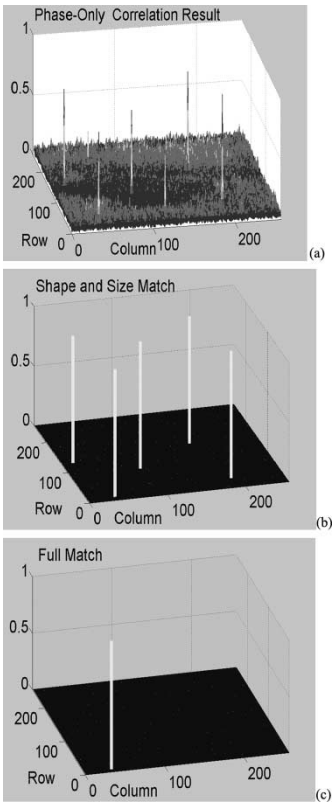

Fig. 6. Searching the objects that fully match the reference pattern by the method in Section VII-B-1. (a) Result of phase-only correlation. (b) Appropriate threshold result of (a). (c) Fully matched result. ($1.1 > |\mathrm{g(ms,ns)}| > 0.9$, and $\mathrm{r1} > 0.9$).

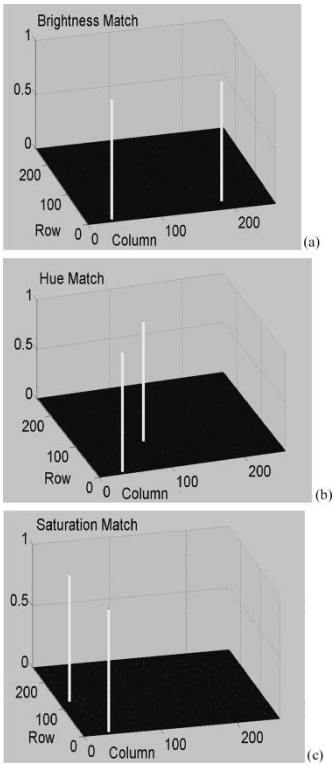

Fig. 7. Searching the objects that *partially match* the reference pattern. (a) $1.1 > |g(m_s,n_s)| > 0.9$, *brightness match*. (b) $\rho_2 > 0.9$, *hue match*. (c) $\rho_2 > 0.9$ *saturation match*.

the reference car can be detected. For example, only cars no. 1 and 2 have the same brightness as the reference case; therefore, only these two cars are detected by requirement 2 in Fig. 7(a). The hue and saturation match results are shown in Fig. 7(b) and (c), respectively.

Experiment 2—Color Template Matching 2 for Average and True Match: In this experiment, following steps 1–7 as discussed above, we will show that objects with the same shape and size, same average brightness, or hue or saturation as the reference pattern will satisfy requirements 2, 4, and 5. However, after step 7 (*true match test*), these average matches are removed. The reference pattern and input image are shown in Fig. 8. The colors of the character objects are listed in Table V. The first character B has a different shape, size, and color from P, the second character—2P with two colors—has the same saturation and average brightness as the reference pattern, the third character 3P is the reference character, the fourth character R has a different shape and size but the same color as P, and the fifth character—5P with two colors—has the average hue as reference pattern.

- *Average Brightness/Hue/Saturation Match*

  We use the color template match steps 1–6 to find the average brightness, hue, and saturation matches. The results of average brightness, hue, and saturation matches are shown in Fig. 9(a)–(c), respectively. 2P and 3P have the same *average brightness* as the reference pattern; therefore, there are two peaks in Fig. 9(a). 3P and 5P have the same *average hue* as the reference pattern; therefore, there are two peaks in Fig. 9(b). 2P and 3P have the *same saturation* as reference pattern; therefore, there are two peaks in Fig. 9(c). The characters B and R in Fig. 8(b) do not have the same shape and size as P so they are removed by the phase-only correlation.
- *True Brightness/Hue/Saturation Match*

  By means of the step 7, the true match test, the average brightness match of second character 2P, and the average hue match of the fifth character 5P are removed. The true brightness, hue, and saturation match with 3P are shown in Fig. 9(d)–(f), respectively. The second character 2P has the same saturation as the reference character; therefore, there are still two peaks in Fig. 9(f).

Experiment 3—Color Template Matching 3 for Real Object Match: In this experiment, using the previous steps and requirements discussed above, we will detect the real objects in the nature image with the same shape and size or the same brightness/hue/saturation as the reference pattern.

The reference pattern is a pink flower shown in Fig. 10(a), and the input image containing five flowers is shown in Fig. 10(b). Flower no. 1 is exactly the same as the reference flower, flower no. 4 is the same as the reference flower with some difference in shape and size, and flower no. 5 is the same as the reference flower in shape and size but not in hue. Flower nos. 2 and 3 have no similar components in shape, size, and hue as the reference flower.

- *Full matched pattern:*

  Using the same steps, requirements and thresholds as in Experiment 1, only flower no. 1 is detected as the fully matched object. The phase-only correlation results, the shape and size match results but not in hue, and the full match result with shape, size, and hue are shown in Fig. 11(a)–(c), respectively.

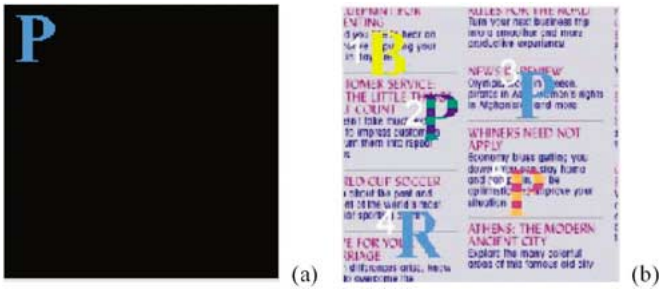

Fig. 8. Reference pattern and input objects of Experiment 2.

TABLE V
COLOR OF THE CHARACTERS IN THE INPUT IMAGE IN FIG. 8(b)

Number of 5 characters	RGB value	(A,H, ϕ values)
1	(220,220,16)	(311,3.14, 0.56)
2	(160,56,108), (244,160,80)	(201,1.57, 0.39), (303, 2.6,0.39)
3 (Reference pattern)	(64,128,192)	(240, -0.52,0.39)
4	(64,128,192)	(240, -0.52,0.39)
5	(60,22,147), (6, 130, 93)	(160, 0.3,0.6), (160, -1.34,0.6)

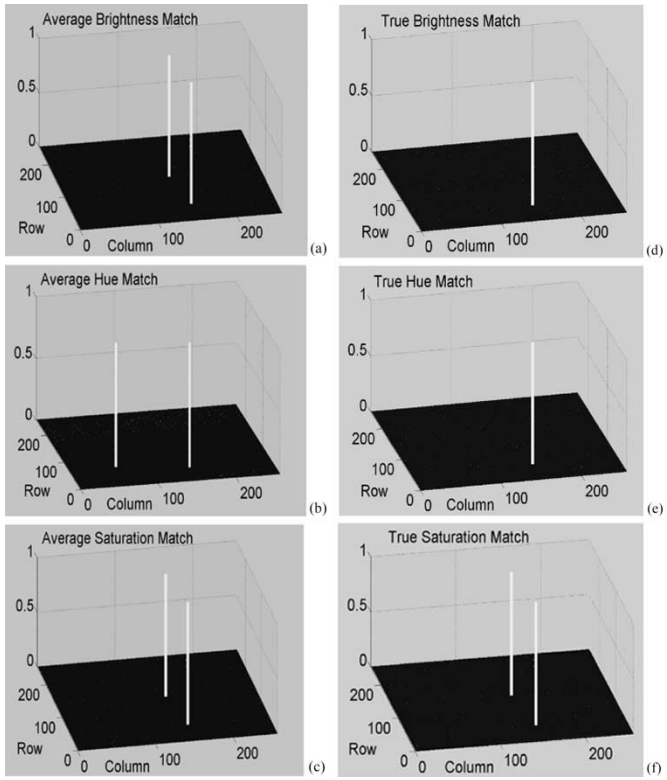

Fig. 9. Searching the objects that *partially match* with the reference pattern. (a)–(c) *Average Brightness, Hue, and Saturation Match*. (d)–(f) *True Brightness, Hue, and Saturation Match*. $(1.1 > |g(m_s, n_s)| > 0.9, \rho_2 > 0.9, \rho_3 > 0.9$. Brightness difference threshold $T_A = 10$. Hue difference threshold $T_H = \pi/6$. Saturation difference threshold $T_\Phi = \pi/12$.

- *Partially matched pattern after the shape and size match:*
  After detecting the objects with the same shape and size as the reference flower in Fig. 11(a), if we just use one of the requirements 2, 4, or 5, the flowers either with the average brightness, hue, or saturation close to the reference flower can be detected, respectively. These results are shown in Fig. 11(d)–(f). Flower no. 1 is detected in all three figures, and flower no. 5 is detected in Fig. 11(d) and (f) with brightness and saturation match but not in hue. Flower no. 4 is not detected because its shape and size is somewhat different from the reference flower.
- *Partially matched pattern without performing the shape and size match first:*
  For the nature image, it is rarely possible to find any two objects with exactly the same shape and size, even if they belong to the same class. Therefore, it is reasonable to relax or remove the shape and size match requirement in the beginning and directly to perform the average brightness, hue, and saturation match. These match results are shown in Fig. 11(g)–(i), respectively. Although the average match results are not the impulse-like peak, we can still locate the candidate positions of the matched objects, or we can find the true match objects by the true match test.

  Compared with the results in Fig. 11(d)–(f), the only difference is that similar pink flower no. 4 is detected in all of the three match results. The results in Fig. 11(g)–(i) are more reasonable and more close to the human visual appearance.

B. Color-Sensitive Edge Detection

1) Color-Sensitive Edge Detection for One Selected Color: Color-sensitive edge detection of one selected color by quaternions has been developed in [9] and [10]. We increase the types of the color-sensitive edge detection by using RBs and by requirements 2–5 in Section VII-A. By the following method, we detect several color-sensitive edges (brightness matched, hue matched, saturation matched, chromaticity matched, and full matched) at the same time. The detection steps are shown as follows.

Step 1) Transform the color image $f_o(m,n)$ into I-H-S color space by (51). Then, we use (53) to represent the color images.

$$f(m,n) \equiv A_f(m,n)e^{iH_f(m,n)}e^{k\phi_f(m,n)} \tag{105}$$

where $f(m,n)$ is the RB representation of the input image. We do not normalize the image.

Step 2) The color-sensitive edge detection is performed by the following three steps:

a) *Color-selective filter*:

$$h(m,n) = \frac{1}{9}\begin{bmatrix} q_s^{-1} & q_s^{-1} & q_s^{-1} \\ q_s^{-1} & q_s^{-1} & q_s^{-1} \\ q_s^{-1} & q_s^{-1} & q_s^{-1} \end{bmatrix}$$
where $q_s = A_1 e^{iH_1}e^{k\phi_1}$ is the selected color. (106)

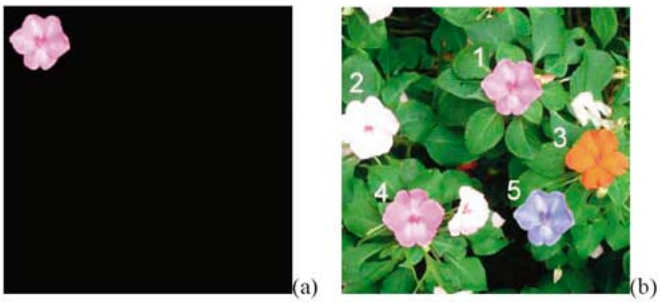

Fig. 10. (a) Reference pattern and (b) input nature image of Experiment 3.

(The size of this filter can be changed for different applications.) First, the color image is convoluted with the filter (106).

$$g(m,n) = f(m,n) *_{\mathrm{RB}} h(m,n). \tag{107}$$

In the color image, if one point and its neighborhood have the same color as q_s, then the output at this point is $1 + 0\mathrm{i} + 0\mathrm{j} + 0\mathrm{k}$. Besides, if only the saturation (or hue) is different from q, then the output will be $e^{k(\Delta\phi)}$ [or $e^{i(\Delta H)}$].

b) *Search the regions composed by the points with the same color as* $\boldsymbol{q_s}$.
From the filtered image $g(m,n)$, to find the points with the *same brightness* and *chromaticity* as q, we apply the requirements 2 and 3 for all the points of $g(m,n)$. If (87) and (88) are satisfied at the same time for some point, then this point has the almost same color as q_{s}. Assume that $g_{\mathrm{m}}(m,n)$ is the matching mask and that

$$g_m(m,n) = \begin{cases} 1, & \text{if } g(m,n) \text{ satisfies (87) and (88)} \\ 0, & \text{else.} \end{cases} \tag{108}$$

then we find the region with the same color as $\boldsymbol{q_s}$ from $g_{\mathrm{m}}(m,n)$.

c) *Search the edges of the matching mask* $\boldsymbol{g_{\mathrm{m}}(m,n)}$
The edges of the matching mask $g_{\mathrm{m}}(m,n)$ are the color-sensitive edges of the color q.

Following the above process, we detect the edges of the regions that are composed by the color q. Besides, using only requirement 2, 4, or 5 in step 2(b), we detect the brightness, hue, or saturation match edges, respectively. Several different color-sensitive edges can be detected at the same time. The requirements used in step 2(b) decide the result of the color-sensitive edge detection.

Experiment 4—Color-Sensitive Edge Detection for One Selected Color: The selected *green* color is (R, G, B) $= (15, 106, 73)$, and Fig. 5(b) is the test image. We do the edge detection by the process as above. The real part of the output resulting by the color-selective filter is shown in Fig. 12(a). The regions whose output values equal to $1 + 0\mathrm{i} + 0\mathrm{j} + 0\mathrm{k}$ are fully matched regions and are shown in Fig. 12(b). The fully matched edges are shown in Fig. 12(c). Besides, partially

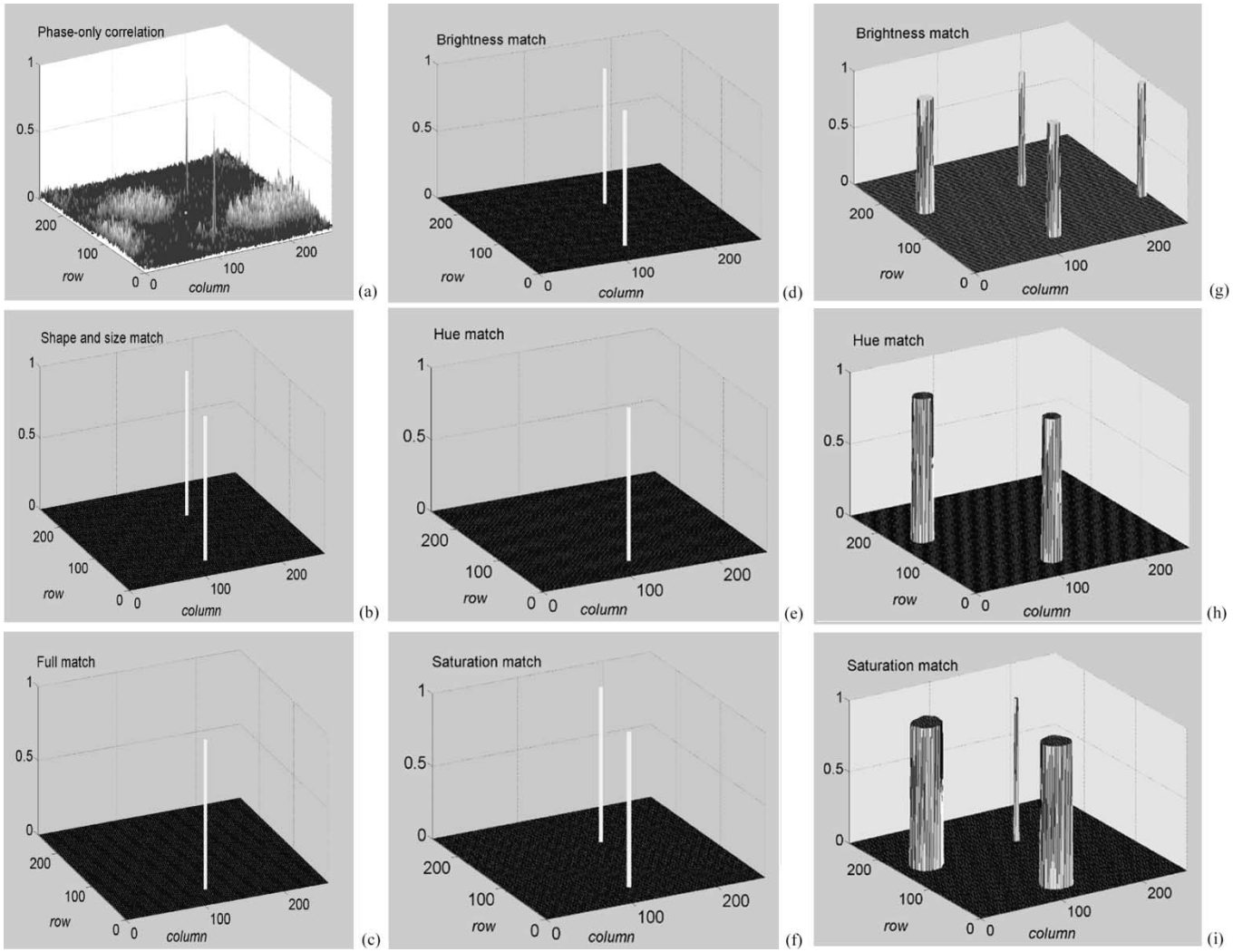

Fig. 11. Searching the real objects that fully or partially match the reference pattern by the method in Section VII-B1. (a) Phase-only correlation. (b) Shape and size match result but not in hue. (c) Full match result with shape, size, and hue. (d)–(f) Either brightness, hue, or saturation match results after shape and size match. (g)–(i) Either average brightness, hue, or saturation match results without performing shape and size match first.

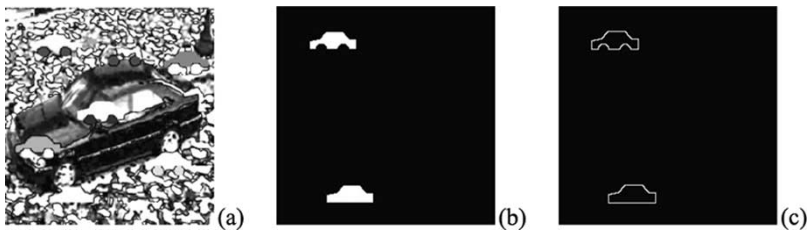

Fig. 12. Color sensitive edge detection—*Fully matched edge*. Selected *green* color is (15, 106, 73). (a) Real part of the output resulting by the color-selective filter. (b) Fully matched regions. (c) Fully matched edges.

matched edges (brightness, hue, and saturation) are plotted in Fig. 13(a)–(c), respectively. The chosen thresholds are $c_1 = d_1 = d_2 = d_3 = 0.95$, $c_2 = 1.05$.

Furthermore, we use our method to detect color-sensitive edges of natural color image [9]. In Fig. 14, we first shift the ranges of R-G-B color from (0, 0, 0)$\sim$(255, 255, 255) to $(-127.5, -127.5, -127.5) \sim (127.5, 127.5, 127.5)$. The selected *brown* color is also shifted to (R, G, B) $= (\mathbf{69} - 127.5, \mathbf{56} - 127.5, \mathbf{33} - 127.5)$. Then, we follow the process as the above. The threshold d_1 in (95) is 0.55. In Fig. 12(b), we find the chromaticity edges of the selected brown color.

In addition to the chromaticity edges, we find the brightness, hue, and saturation edges of the selected color at the same time. There edges are shown in Fig. 14(d)–(f). The chosen thresholds are $c_1 = 0.8$, $c_2 = 1.2$, and $d_2 = d_3 = 0.55$.

2) Color-Sensitive Edge Detection Between Two Selected Colors: Color-sensitive edge detection of two colors by quaternions has been developed in [10]. The authors use two-sided conventional quaternion convolution and a pair of

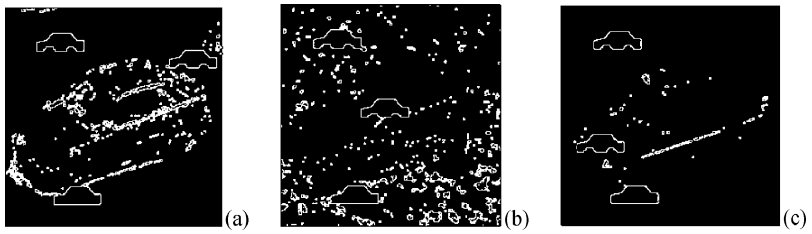

Fig. 13. Color sensitive edge detection—*Partially matched edge*. Selected *green* color is (15, 106, 73). (a) Brightness matched edges. (b) Saturation matched edges. (c) Hue matched edges.

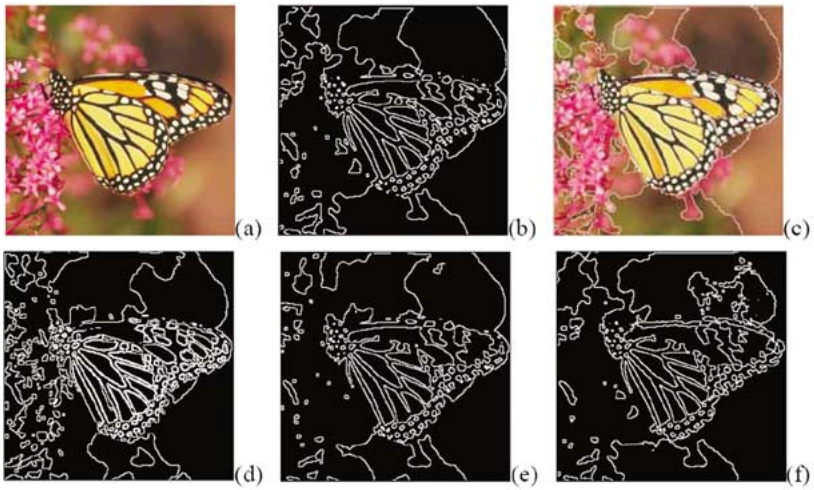

Fig. 14. Color sensitive edge detection of a natural image. The selected *brown* color is (69, 56, 33). (a) Natural image. (b) Chromaticity matched edges of selected brown color. (c) Superposition of (a) and (b). (d)–(f) Brightness matched, hue matched, and saturation matched edges of selected brown color.

hypercomplex filters to find the edges between the two colors. The two left-side and right-side filters they used are

$$L = \frac{1}{\sqrt{6}}\begin{bmatrix} \mu_{c1} & \mu_{c1} & \mu_{c1} \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} \quad R = \frac{1}{\sqrt{6}}\begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ \mu_{c2} & \mu_{c2} & \mu_{c2} \end{bmatrix} \tag{109}$$

where μ_{c1} and μ_{c2} are two pure unit quaternions representing the selected colors C_1 and C_2, respectively.

We improve the above performance by using RBs. First, we do the same edge detection by only using one single RB filter and one-side RB convolution. Therefore, the computation complexity is reduced to half of the one by conventional quaternions in [10]. Second, we do not normalized the color image as in [10] to keep the brightness information; therefore, we detect several color-sensitive edges (brightness matched, hue matched, saturation matched, chromaticity matched, and full matched) at the same time using the method as follows.

The RB filter we use is the combination of the above two filters in (109):

$$h(m,n) = \frac{1}{6}\begin{bmatrix} q_1^{-1} & q_1^{-1} & q_1^{-1} \\ 0 & 0 & 0 \\ q_2^{-1} & q_2^{-1} & q_2^{-1} \end{bmatrix} \tag{110}$$

where q_1 and q_2 are the RBs representing the two selected colors C_1 and C_2. We choose this filter because the convolution result at the matched positions is equal to $1+0\mathrm{i}+0\mathrm{j}+0\mathrm{k}$. Therefore, we use requirements 2 and 3 in Section VII-A to find the match positions, and these positions are the desired edges. Using the filter (110), we only detect the edges between two colors q_1 and q_2, where q_1 is at the top and q_2 is at the bottom. To detect the edges between the same two colors q_1 and q_2 but with q_2 at the top and q_1 at the bottom, we need another filter. This filter is the 180° rotation of (110) (changing the locations of q_1 and q_2)

$$h_{rot180}(m,n) = \frac{1}{6}\begin{bmatrix} q_2^{-1} & q_2^{-1} & q_2^{-1} \\ 0 & 0 & 0 \\ q_1^{-1} & q_1^{-1} & q_1^{-1} \end{bmatrix}. \tag{111}$$

Therefore, we need two filters to detect the *vertical edges* between two colors. For the same reason, we need two filters to detect the *horizontal edges* between two colors. These two filters are the 90 and 270° rotation of (110), respectively. This problem also occurs by using the quaternion filters in [10].

The vertical and horizontal edge detection steps between two colors are shown as follows.

Step 1) Select the desired two colors, and use the simplified polar form to represent the input color image $f(m,n)$ and the filter $h(m,n)$ in (110).

Step 2) First, the color image is convoluted with the filter

$$g(m,n) = f(m,n) *_{\mathrm{RB}} h(m,n). \tag{112}$$

Then, use requirements 2 and 3 to find the positions of matched pattern. These positions are the desired edges between these two selected colors.

Step 3) Repeat the step 2 with filters that are 90, 180 and 270° rotation of $h(m,n)$ in (110).

Fig. 15. Color-sensitive edge detection between two selected colors for a natural image. The selected two *red and white* colors are (237, 157 157) and (255, 255, 255). (a) Natural image. (b) Horizontal and vertical edges detected by RBs. (e) Combination of the natural image and the negative of (b). (c) Horizontal and vertical edges detected by conventional quaternions in [10]. (f) Combination of the natural image and the negative of (c). (d) Diagonal edges detected by RBs. (g) Combination of the natural image and the negative of (d). (Black pixels shown in (e)–(g) have been identified as either horizontal-vertical or diagonal, red-white or white-red, or edge pixels.).

Besides, if we use requirements 2, 4, or 5 in Step 2, then we detect the edges between two selected brightnesses or the edges between two selected hues or the edges between two selected saturations, respectively. Several different color-sensitive edges can be detected at the same time. The requirements used in Step 2(b) decide the result of the color-sensitive edge detection.

Moreover, the diagonal edges can be detected by using the following filters:

$$\frac{1}{6}\begin{bmatrix} q_2^{-1} & q_2^{-1} & 0 \\ q_2^{-1} & 0 & q_1^{-1} \\ 0 & q_1^{-1} & q_1^{-1} \end{bmatrix}, \quad \frac{1}{6}\begin{bmatrix} q_1^{-1} & q_1^{-1} & 0 \\ q_1^{-1} & 0 & q_2^{-1} \\ 0 & q_2^{-1} & q_2^{-1} \end{bmatrix}$$
$$\frac{1}{6}\begin{bmatrix} 0 & q_2^{-1} & q_2^{-1} \\ q_1^{-1} & 0 & q_2^{-1} \\ q_1^{-1} & q_1^{-1} & 0 \end{bmatrix} \text{ and } \frac{1}{6}\begin{bmatrix} 0 & q_1^{-1} & q_1^{-1} \\ q_2^{-1} & 0 & q_1^{-1} \\ q_2^{-1} & q_2^{-1} & 0 \end{bmatrix} \quad (113)$$

Experiment 5—Color-Sensitive Edge Detection Between Two Selected Colors: To be compared with the result in [10], we use the same image and the same selected colors as [10]. The image is shown in Fig. 15(a), and the two selected *red and white* colors are (237 157 157) and (255 255 255). As [10], we shift the ranges of R-G-B color from (0, 0, 0)$\sim$(255, 255, 255) to $(-127.5, -127.5, -127.5) \sim (127.5, 127.5, 127.5)$. The selected red and white colors are also shifted to $(\mathbf{237} - 127.5,\ \mathbf{157} - 127.5,\ \mathbf{157} - 127.5)$ and $(\mathbf{255} - 127.5,\ \mathbf{255} - 127.5,\ \mathbf{255} - 127.5)$. Then, we follow the process as above. The threshold d_1 in (95) is 0.55.

The black pixels shown in Fig. 15(b) have been identified as either vertical or horizontal chromaticity matched edges of the selected color. The result of [10] is shown in Fig. 15(c). Because we use RBs with simplified polar form to represent the color image in I–H–S color space and use different requirements and thresholds to [10], the results of our method and [10] are a little different.

In addition to the vertical or horizontal edges, the diagonal chromaticity-matched edges of the selected two colors are detected by (113) and shown in Fig. 15(f) and (g).

VIII. Conclusion

In this paper, we proposed a new and very important representation of RBs: the polar form. From this polar form, the multiplication of RBs and the calculation of RB inverse and conjugation becomes very easy. Besides, we discuss the relations between the three useful representation of RBs: the $e_1 - e_2$ form, matrix representation, and polar form. From these relations, we understand the geometric meanings of RBs—rotations in the four dimensional space and Lorentz transforms in the subspace of the four-dimensional space. From these three representations, we redefine the new and unique definitions of RB norm and conjugation. The properties of these definitions are almost the same as the ones of the complex numbers and are compatible for the complex numbers.

In digital signal processing, we use RBs to analyze the complex symmetric multichannel system and symmetric lattice filter system. By using RBs, the computation complexity can be greatly reduced.

In digital image processing, we define a simplified polar form of RBs to do color image processing in I-H-S space. We detect efficiently the objects that have the same *shape, size, brightness,* or *chromaticity* with the reference pattern and the edges of the regions that have the same color with the selected color by RB correlation, phase-only correlation, and convolution. Many pattern recognition tasks and color-sensitive edge detection (i.e. brightness, hue, or saturation match) can be done simultaneously. Using RBs to do color-sensitive edge detection between two colors are more efficient and simpler than using conventional quaternions. We believe that the RB is a very useful tool for color image processing.

REFERENCES

[1] W. R. Hamilton, *Elements of Quaternions*. London, U.K.: Longmans, Green and Co., 1866.

[2] I. L. Kantor and A. S. Solodovnikov, *Hypercomplex Number: An Elementary Introduction to Algebras*. New York: Springer-Verlag, 1989.

[3] S. C. Pei and C. M. Cheng, "A novel block truncation coding of color image using a quaternion-moment-preserving principle," *IEEE Trans. Commun.*, vol. 45, pp. 583–595, May 1997.

[4] T. Bülow and G. Sommer, "Multi-dimensional signal processing using an algebraically extended signal representation," in *Proc. Int. Workshop Algebraic Frames for the Perception-Action Cycle*, vol. 1315, G. Sommer and J. J. Koenderink, Eds., 1997, pp. 148–163.

[5] S. J. Sangwine, "Color image edge detector based on quaternion convolution," *Electron. Lett.*, vol. 34, no. 10, pp. 969–971, May 1998.

[6] S. C. Pei and C. M. Cheng, "Color image processing by using binary quaternion-moment preserving thresholding technique," *IEEE Trans. Image Processing*, vol. 8, pp. 614–628, May 1999.

[7] S. J. Sangwine and T. A. Ell, "Hypercomplex auto- and cross-correlation of color images," in *Proc. ICIP*, 1999, pp. 319–323.

[8] T. Bülow, "Hypercomplex spectral signal representations for the processing and analysis of images," Ph.D. dissertaion, Christian-Albrechts Univ., Kiel, Germany, Aug. 1999.

[9] C. J. Evans, S. J. Sangwine, and T. A. Ell, "Hypercomplex color-sensitive smoothing filters," in *Proc. IEEE Int. Conf. Image Process.*, vol. 1, Sep. 2000, pp. 541–544.

[10] ——, "Color-sensitive edge detection using hypercomplex filters," in *Proc. EUSIPCO*, Tampere, Finland, Sept. 2000.

[11] S. C. Pei, J. J. Ding, and J. H. Chang, "Color pattern recognition by quaternion correlation," in *Proc. ICIP*, 2001, pp. 894–897.

[12] H.-D. Schtte and J. Wenzel, "Hypercomplex numbers in digital signal processing," in *Proc. IEEE Int. Symp. Circuits Syst.*, vol. 2, 1990, pp. 1557–1560.

[13] K. Ueda and S.-I. Takahashi, "Digital filters with hypercomplex coefficients," in *Proc. IEEE Int. Symp. Circuits Syst.*, vol. 1, May 1993, pp. 479–482.

[14] V. S. Dimitrov, T. V. Cookler, and B. D. Donevsky, "On the multiplication of reduced biquaternions and applications," *Inform. Process. Lett.*, vol. 43, no. 3, pp. 161–164, 1992.

[15] T. A. Ell, "Quaternion-Fourier transforms for analysis of two-dimensional linear time invariant partial differential systems," in *Proc. 32nd Conf. Decision Contr.*, Dec. 1993, pp. 1830–1841.

[16] S. C. Pei, J. J. Ding, and J. H. Chang, "Efficient implementation of quaternion Fourier transform, convolution, and correlation by 2-D FFT," *IEEE Trans. Signal Processing*, vol. 49, pp. 2783–2797, Nov. 2001.

[17] S. J. Sangwine, "The discrete quaternion Fourier transform," in *Proc. 6th Int. Conf. Image Process. Applicat.*, vol. 2, Jul. 1997, pp. 14–17.

[18] M. Felsberg, "Fast quaternion fourier transform," Tech. Rep., Cognitive Syst. Group, Inst. Comput. Sci. Applied Math., Christian-Albrechts Univ., Kiel, , Sept. 1997.

[19] M. Felsberg and G. Sommer, "Fast algorithms for the hypercomplex Fourier transforms," in *Proc. 2nd Int. Workshop Transforms Filterbanks*, vol. 99, Brandenburg an der Havel, Germany, pp. 295–302.

[20] M. Felsberg, T. Bülow, and G. Sommer, *Geometric Computing With Clifford Algebra*, G. Sommer, Ed. Berlin, Germany: Springer-Verlag, 2001.

[21] C. M.Clyde M. Davenport, "A communitative hypercomplex algebra with associated function theory," in *Clifford Algebra With Numeric and Symbolic Computations*, R. Ablamowicz, Ed. Boston, MA: Birkhauser, 1996, pp. 213–227.

[22] M. Felsberg, "Signal processing using frequency domain methods on Clifford algebra," Master thesis, 1998.

[23] T. A. Ell and S. J. Sangwine, "Decomposition of 2D hypercomplex Fourier transforms into pairs of complex Fourier transforms," in *Proc. EUSIPCO*, pp. 151–154.

[24] ——, "Hypercomplex Wiener-Khintchine theorem with application to color image processing," in *Proc. ICIP*, vol. 2, Sept. 2000, pp. 792–795.

[25] T. A. Ell, "Hypercomplex spectral transform," Ph.D. dissertation, Univ. Minnesota, Minneapolis, MN, 1992.

[26] S. J. Sangwine, "Fourier transforms of color images using quaternion or hypercomplex numbers," *Elect. Lett.*, vol. 32, no. 21, pp. 1979–1980, Oct. 1996.

[27] S. J. Sangwine and T. A. Ell, "The discrete Fourier transforms of a color image," in *Image Process. II Math. Methods, Algorithms, Applicat.*, 2000, pp. 411–430.

[28] ——, "Hypercomplex Fourier transforms of color image," in *Proc. ICIP*, 2001, pp. 137–140.

[29] J. L. Horner and P. D. Gianino, "Phase-only matched filtering," *Appl. Opt.*, vol. 23, pp. 812–816, Mar. 1984.

[30] G. B.G. Baley Price, *An Introduction to Multicomplex Space and Functions*. New York: Marcel Dekker, 1991.

[31] T. Bülow and G. Sommer, "Hypercomplex signals—a novel extension of the analytic signal to the multidimensional case," *IEEE Trans. Signal Processing*, vol. 49, pp. 2844–2852, Nov. 2001.

[32] I. M. Yaglom, *Complex Numbers in Geometry*. New York: Academic, 1968.

[33] J. Rooney, "On the three types of complex number and planar transformations," *Environment Planning*, vol. B5, pp. 89–99, 1978.

[34] C. E. Moxey, S. J. Sangwine, and T. A. Ell, "Hypercomplex correlation techniques for vector images," *IEEE Trans. Signal Processing*, vol. 51, pp. 1941–1953, July 2003.

Soo-Chang Pei (F'00) was born in Soo-Auo, Taiwan, R.O.C., in 1949. He received B.S.E.E. degree from National Taiwan University (NTU), Taipei, Taiwan, in 1970, and the M.S.E.E and Ph.D degrees from the University of California, Santa Barbara (UCSB), in 1972 and 1975, respectively.

He was an engineering officer with the Chinese Navy Shipyard, Ponhu Island, Taiwan, from 1970 to 1971. From 1971 to 1975, he was a Research Assistant at UCSB. He was a Professor and Chairman of the Electrical Engineering Department, Tatung Institute of Technology, Taipei, from 1981 to 1983, and at NTU from 1995 to 1998, where he is currently a Professor and Dean of the Electrical Engineering Department, College of Electrical Engineering and Computer Science. His research interests include digital signal processing, image processing, optical information processing, and laser holography.

Dr. Pei received the National Sun Yet-Sen Academic Achievement Award in Engineering in 1984, the Distinguished Research Award from the National Science Council, R.O.C., from 1990 to 1998, the Outstanding Electrical Engineering Professor Award from the Chinese Institute of Electrical Engineering in 1998, the Academic Achievement Award in Engineering from the Ministry of Education in 1998, the Pan Wen-Yuan Distinguished Research Award in 2002, and the National Chair Professor Award from the Ministry of Education in 2002. He was President of the Chinese Image Processing and Pattern Recognition Society in Taiwan from 1996 to 1998 and is a member of Eta Kappa Nu and the Optical Society of America. He was elected Fellow of the IEEE in 2000 for contributions to the development of digital eigenfilter design, color image coding and signal processing, and electrical engineering education in Taiwan.

Ja-Han Chang was born in Taipei, Taiwan, R.O.C., in 1977. He received the B.S. degree in electrical engineering from National Taiwan University (NTU), Taipei, in 1999. He is currently pursuing the Ph.D. degree under the supervision of Prof. S.-C. Pei in the Department of Electrical Engineering at NTU.

His current research areas include image watermarking, image processing, pattern recognition, quaternion and quaternion Fourier transforms, reduced biquaternions and reduced biquaternion Fourier transforms, etc.

Jian-Jiun Ding was born in 1973 in Pingdong, Taiwan, R.O.C. He received the B.S. degree in 1995, the M.S. degree in 1997, and the Ph.D. degree in 2001, all in electrical engineering from the National Taiwan University (NTU), Taipei, Taiwan.

He is currently a postdoctoral researcher with the Department of Electrical Engineering, NTU. His current research areas include fractional Fourier transforms, linear canonical transforms, orthogonal polynomials, fast algorithms, quaternion algebra, pattern recognition, and filter design.