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 ($e_x = e_y$ 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:

$$q = q_r + q_i i + q_j j + q_k k$$

(1)

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

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

(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].

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

(3)

Therefore, it is difficult to analyze the quaternion LTI system. On the other hand, the representation of RBs [12]–[14] is

$$q = q_r + q_i i + q_j j + q_k k$$

(4)

where

$$i^2 = j^2 = k^2 = -1, \quad ij = -ji = k.$$  

(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

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

(6)
In 1996, Davenport proposed the four-dimensional (4-D) commutative hypercomplex algebras (HCA4), which are isomorphic to the group ring C2, 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 2n dimensions, and they had proven that the 2n-dimensional HCA is isomorphic to C2n−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 C2.

In fact, the difference between the three numbers—RBs (j2 = 1), double-complex algebra (j2 = 1) and HCA4(k2 = 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 e1 = e2 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 e1 = e2 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:

- \( \bar{q} \) RB conjugation.
- |q| Norm of the RBs.
- \( M_q \) Matrix representation of an RB q.
- \(*RB\) RB convolution.

\( \mathbb{C}_{RB} \) RB correlation.
\( \mathbb{C}_{RBFO} \) RB phase-only correlation.
\( H_{RB}(p, s) \) 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 \( j^2 = -1 \)), 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:

\[ u_{2n} = u_n + v_n \varepsilon \]

where \( u_{2n} \) is a 2n-dimensional hypercomplex number, \( u_n \) and \( v_n \) are two n-dimensional hypercomplex numbers, and \( \varepsilon^2 = 1 \) 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:

\[ z = a_1 + a_2 i, \quad q = a + bi + cj + dk = (a + bi) + (c + di)j \equiv z_1 + z_2 j \]

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

\[ q = q_a + q_i a + q_j j + q_k k = (q_a + q_i) + (q_j + q_k)j \]

where \( q_a \) and \( q_b \) are two complex numbers, and

\[ ij = ji = k, \quad jk = kj = i, \quad ik = ki = -j, \quad i^2 = k^2 = -1, \quad j^2 = 1. \]

Although the RBs are isomorphic to the HCA4, we still use the same name, reduced biquaternions as Schitte and Wenzel because they first used the HCA4 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 \( e_1^2 = e_1^{-1} = \ldots = e_1, \quad e_2^2 = e_2^{-1} = \ldots = e_2 \). Therefore, \( e_1 \) and \( e_2 \) are both idempotent elements (\( e_1^2 = e_1, \quad e_2^2 = e_2 \))
and divisors of zero \( (e_1 e_2 = 0) \). (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 \( e_1 e_2 \) or \( e_2 e_1 \) has no inverse element. (\( e_1 \) and \( e_2 \) are any complex numbers.)

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

\[
q = (q_1 + q_i i) + \left( q_j + q_k k \right) j = q_1 e_1 + q_i i e_1 + q_j j e_1 + q_k k e_1 = \left( q_1 e_1 + q_i i e_1 + q_j j e_1 + q_k k e_1 \right) = q_{a+b} i e_1 + q_{b-a} j e_1 + q_{a-b} k e_1 + q_{b+a} \tag{11}
\]

where \( q_{a+b} = q_1 q_2 \), \( q_1 = (q_1 + q_x) + (q_i + q_y) i \), and \( q_{b-a} = q_1 q_2 = (q_1 - q_x) + (q_i - q_y) i \). 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:

\[
q_1 q_2 = q_2 q_1 = (q_{1a} + q_{1b} i) (q_{2a} + q_{2b} i) = (q_{1a} + q_{1b} i) (q_{2a} + q_{2b} i) = q_{a+b} i e_1 + q_{b-a} j e_1 + q_{a-b} k e_1 + q_{b+a} \tag{12}
\]

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 \( q^{-1} = \left( q_{b+a} \right)^{-1} e_1 + \left( q_{b-a} \right)^{-1} e_2 \). Therefore, the inverse of an RB exists if \( (q_{b+a}) \neq 0 \) and \( (q_{b-a}) \neq 0 \).

### C. Matrix Representation of an RB

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

\[
1 \rightarrow I_4 \equiv \begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1
\end{bmatrix}
\]

\[
i \rightarrow M_i \equiv \begin{bmatrix}
0 & -1 & 0 & 0 \\
1 & 0 & 0 & 0 \\
0 & 0 & 0 & -1 \\
0 & 1 & 0 & 0
\end{bmatrix}
\]

\[
j \rightarrow M_j \equiv \begin{bmatrix}
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 \\
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0
\end{bmatrix}
\]

\[
k \rightarrow M_k \equiv \begin{bmatrix}
0 & 0 & 0 & -1 \\
0 & 0 & 1 & 0 \\
0 & -1 & 0 & 0 \\
1 & 0 & 0 & 0
\end{bmatrix}
\]

\[
M_0^2 = I_4, M_1^2 = M_2^2 = M_3^2 = -I_4, M_4 M_1 = M_5 M_3 = M_6 M_2 = M_7, M_4 M_2 = M_5 M_1 = M_6 M_3 = -M_7
\]

where \( M_1, M_2, M_3, M_4, M_5, M_6, M_7 \) are the matrices representing \( e_1, e_2, e_1 e_2, e_1 + e_2, e_1 - e_2, e_1 e_2 + 1, e_1 e_2 - 1 \), respectively.

### TABLE I

<table>
<thead>
<tr>
<th>Complexity of Two Reduced Biquaternions’ Multiplication</th>
</tr>
</thead>
<tbody>
<tr>
<td>Real Multiplications</td>
</tr>
<tr>
<td>----------------------</td>
</tr>
<tr>
<td>Direct Calculation</td>
</tr>
<tr>
<td>Calculation by means of ( e_1 ) and ( e_2 )</td>
</tr>
</tbody>
</table>

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

\[
q = q_r + iq_i + jq_j + kq_k \rightarrow M_q \equiv \begin{bmatrix}
q_r & -q_i & q_j & -q_k \\
q_i & q_r & -q_k & q_j \\
q_j & q_k & q_r & -q_i \\
q_k & -q_j & q_i & q_r
\end{bmatrix}
\]

The first column of \( M_q \) is the four elements of the original RB. Similar to [21], the four eigenvalues of \( M_q \) are

\[
\begin{array}{c}
\varepsilon = (q_r + q_j) + (q_i + q_k) i \\
\varepsilon^2 = (q_r + q_j) - (q_i + q_k) i \\
\eta = (q_r - q_j) + (q_i - q_k) i \\
\eta^2 = (q_r - q_j) - (q_i - q_k) i
\end{array}
\]

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

Besides, the determinant of \( M_q \) is the product of the above four eigenvalues.

\[
\delta = |q_r - q_i, q_j - q_k, q_i, q_r, q_k, q_j, q_k, q_j, q_i, q_r, q_k, q_k|
\]

\[
= \left( (q_r + q_j)^2 + (q_i + q_k)^2 \right) \left( (q_r - q_j)^2 + (q_i - q_k)^2 \right)
\]

\[
= \left( q_r^2 + q_i^2 + q_j^2 + q_k^2 \right)^2 - 4(q_r q_i + q_j q_k)^2 \geq 0.
\]

The conditions for \( \delta = 0 \) are

\[
(q_r = q_j) \cap (q_i = q_k) \text{ or } (q_r = -q_j) \cap (q_i = -q_k),
\]

If \( \delta = 0 \), then the inverse of \( M_q \) 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, \( (M_q)^{-1} \) is the matrix representation of the inverse of \( q \); therefore, we can calculate the inverse of \( q \) from \( (M_q)^{-1} \).

### 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 \( |q_1 q_2| \neq |q_1| |q_2| \), 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 \( q_2 = (1 - i + j - k) \); then, \( q_1 \overline{q_2} = (2 - i - j - k), q_2 \overline{q_1} = (5 - 2i + 2j - 2k) \notin \mathbb{R} \) and \( q_1 q_2 = 2 + j + 2k, |q_1| = \sqrt{7}, |q_2| = \sqrt{2}, |q_1 q_2| = 3 \neq \sqrt{14} = |q_1||q_2| \).

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:

\[ |q| \equiv \sqrt[8]{\delta} = \left( (q_0^2 + q_1^2 + q_2^2 + q_3^2) - 4(q_1q_2 + q_3) \right)^{1/8} \geq 0 \]  

(18)

where \( \delta \) is the determinant of \( M_q \). We define \(|q| \equiv \sqrt[8]{\delta} \) because \([q_1 q_2] = |q_1||q_2| \) is satisfied for this definition, and \(|q| = (q_0^2 + q_1^2)^{1/2} \) if \( q_j = q_k = 0 \). (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 \( q_0 \neq 0 \) and \( q_2 \neq 0 \), then \( M_q = M_{q_1} M_{q_2} \rightarrow \det(M_{q_3}) = \det(M_{q_1}) \det(M_{q_2}) \)

\[ \rightarrow \delta_3 = \delta_1 \delta_2 \] where \( \delta_i = \det(M_{q_i}) \), \( i = 1, 2, 3 \).

\[ |q_1 q_2| = |q_3| = \sqrt[8]{\delta_3} = \sqrt[8]{\delta_1 \delta_2} = |q_1||q_2| \]

The only property different from complex numbers is \(|q_1 + q_2| > |q_1| + |q_2| \) for some special cases. (This Schwarz triangle inequality is not satisfied.) For example, the two special RB numbers \( c_1 = (1 + j)/2, c_2 = (1 - j)/2 \) have zero norm, but the sum of these two numbers is 1.

• Conjugation of RBs:

\[ \bar{q} \equiv \sqrt[8]{\delta} \cdot q^{-1} = \frac{|q|^2}{q} \]

(19)

The matrix representation of \( \bar{q} \) is \( \sqrt[8]{\delta} \cdot (M_q)^{-1} \), and therefore

\[ \bar{q} = \frac{1}{\sqrt[8]{\delta}} \begin{bmatrix} q_i & q_k & q_j & -q_0 \\ q_j & q_i & q_k & -q_0 \\ q_k & q_j & q_i & -q_0 \\ q_0 & q_k & q_j & q_i \end{bmatrix} - i \begin{bmatrix} q_i & q_k & -q_j \\ q_j & q_i & -q_k \\ q_k & q_i & -q_j \\ q_0 & q_k & q_j \end{bmatrix} + j \begin{bmatrix} q_i & q_k & q_j & -q_0 \\ -k & q_j & q_k & -q_0 \\ q_j & -k & q_k & -q_0 \\ q_k & q_j & -k & -q_0 \end{bmatrix} \]

(20)

We choose \( \bar{q} \equiv \sqrt[8]{\delta} \cdot q^{-1} \) because \( \bar{q} \) is a real number, \( |\bar{q}| = |q|^2 \), and \( \bar{q} \bar{q} = q \bar{q} = q \) for this definition. The proof of \( \bar{q} \bar{q} = q \bar{q} = q \) 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 \( \bar{q} \) exists.

The only property different from complex numbers is \( \bar{q}_1 + \bar{q}_2 \neq \bar{q}_1 + \bar{q}_2 \) in general. (This is because the conjugation of RBs is a nonlinear operation.)

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

1) \( \delta = ((1 + i)^2 + (-2 - 2i)^2)((1 - 1)^2 + (2 + 2)^2) = 64 \)

\[ \bar{q} = \frac{1}{8} \begin{bmatrix} 1 & 2 & 0 & -2 \\ 2 & 1 & 2 & 0 \\ 0 & 2 & 1 & -2 \\ -2 & 0 & 1 & 2 \end{bmatrix} - i \begin{bmatrix} 2 & 1 & 2 & 0 \\ 1 & 2 & 0 & -2 \\ 2 & 1 & 0 & -2 \\ 0 & -2 & 1 & 2 \end{bmatrix} + j \begin{bmatrix} 2 & 1 & 2 & 0 \\ -k & 2 & 1 & 0 \\ 1 & -2 & 1 & 2 \\ 0 & 1 & 2 & -2 \end{bmatrix} \]

\[ = 2 - i + 2j + k \]

2) \( \delta = ((1 + i)^2 + (0 - 0)^2)((1 - 1)^2 + (0 + 0)^2) = 0 \).

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 \( u_2 \)):

\[ F_{R}(p, s) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m, n) e^{-2\pi i (m u_1 + n u_2)} \]  

(21)

• DRBFT of type 2 (one imaginary unit vector \( u_1 \)):

\[ F_{R}(p, s) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m, n) e^{-2\pi i u_1 \cdot (m \hat{u} + n \hat{v})} \]  

(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 needs 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 \( \delta \neq 0 \):

\[ q = a + b i + c j + d k = Ae^{\theta_i}e^{\theta_k}e^{\theta_j} \]  

(23)

where \( |q| = \sqrt[8]{\delta} \geq 0 \), \( \theta_i \in (-\pi, \pi], \theta_k \in (-\pi/2, \pi/2], \theta_j \in R \), and \( \delta \) is defined as (16). The proof of (23) is shown as follows.

Step 1) Let \( e^{\theta_i} = e + f j \); then, \( e = \cos \theta_i, f = \sin \theta_i, e > 0, f > 0, f^2 + e^2 = 1 \). Therefore

\[ q = a + b i + c j + d k = Ae^{\theta_i}e^{\theta_k}e^{\theta_j} \]

\[ \Leftrightarrow q e^{-\theta_j} = Ae^{\theta_i}e^{\theta_k} \]

(24)

\[ \Rightarrow [(a e - cf) + (be - df)i + (ce - af)j + (de - bf)k] \]

\[ = A(\cos \theta_i \cos \theta_k + \sin \theta_i \cos \theta_k) \]

\[ - \sin \theta_i \sin \theta_k j + \cos \theta_i \sin \theta_k k \]  

(25)
\[
\begin{align*}
\begin{cases}
(ace - cf) &= A \cos \theta_i \cos \theta_k \equiv a_1 \\
(be - df) &= A \sin \theta_i \cos \theta_k \equiv b_1 \\
(ce - af) &= -A \sin \theta_i \sin \theta_k \equiv c_1 \\
(de - bf) &= A \cos \theta_i \sin \theta_k \equiv d_1
\end{cases}
\end{align*}
\]

and \( e^2 - f^2 = 1 \). \tag{26}

If the solutions of the five unknowns \( c, f, A, \theta_i, \) and \( \theta_k \) in (26) exist, then an RB can be represented by the polar form \( q = A e^{i\theta_i} e^{i\theta_k} \). 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 \( \theta_j \): From (26), we get

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

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

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

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

\[
(2e^2 - 1)(ac + bd) = \pm e \sqrt{e^2 - 1} (a^2 + b^2 + c^2 + d^2), \tag{29}
\]

Taking square at both sides

\[
\begin{align*}
&\left[ (a^2 + b^2 + c^2 + d^2)^2 - 4(ac + bd)^2 \right] \\
\times (e^4 - e^2) - (ac + bd)^2 = 0
\end{align*}
\]

\[
\Rightarrow e^2 = \frac{1 \pm \sqrt{1 + \frac{4(ac + bd)^2}{(a^2 + b^2 + c^2 + d^2)^2 - 4(ac + bd)^2}}}{2} \tag{30}
\]

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

\[
\Rightarrow e = \sqrt{1 + \frac{(a^2 + b^2 + c^2 + d^2)^2}{2\delta}} \tag{31}
\]

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

\[
f = \frac{(2e^2 - 1)(ac + bd)}{(a^2 + b^2 + c^2 + d^2) e} = \frac{(ac + bd)}{e \sqrt{\delta}}. \tag{32}
\]

Hence

\[
\theta_j = \tan^{-1} \frac{f}{e}. \tag{33}
\]

b) Solutions of \( A, \theta_1, \) and \( \theta_3 \): From (26), (30), and (31), we get

\[
\begin{align*}
a_1 &= ace - cf = A \cos \theta_i \cos \theta_k \\
b_1 &= be - df = A \sin \theta_i \cos \theta_k \\
c_1 &= ce - af = -A \sin \theta_i \sin \theta_k \\
d_1 &= de - bf = A \cos \theta_i \sin \theta_k \tag{34}
\end{align*}
\]

and then

\[
\begin{align*}
\theta_i &= \tan^{-1} \frac{b_1}{a_1}, \quad \theta_k = \tan^{-1} \frac{d_1}{a_1}, \quad \text{and} \quad \theta_j = \tan^{-1} \frac{f}{e}, \tag{35}
A &= \sqrt{a_1^2 + b_1^2 + c_1^2 + d_1^2}
= \sqrt{(a^2 + b^2 + c^2 + d^2)(e^2 + f^2) - 4(ac + bd)e}
= \sqrt{\frac{(a^2 + b^2 + c^2 + d^2)^2 - 4(ac + bd)^2}{\sqrt{\delta}}} \\
\delta &= \sqrt{\delta} = |q|. \tag{36}
\end{align*}
\]

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 \( \theta_i \) and \( \theta_k \) 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 \( \theta_j \):

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

\[
\theta_i \in (-\pi, \pi], \quad \theta_k \in \left( -\frac{\pi}{2}, \frac{\pi}{2} \right], \quad \text{and} \quad \theta_j \in R. \tag{37}
\]

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 \( e, f \to \infty \), 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 \( q = a + bi \) \((j \) and \( k \) components equal to 0); then, the \( \theta_k \) and \( \theta_j \) equal to 0 and the polar form of \( q \) become \( q = a + bi = A e^{i\theta_j} \), where \( A = \sqrt{\delta} = \sqrt{a^2 + b^2} \).

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 \( q_1 = A_1 e^{i\theta_1} e^{i\theta_1} \), \( q_2 = A_2 e^{i\theta_2} e^{i\theta_2} \); then, \( q_1 q_2 = A_1 A_2 e^{i(\theta_1 + \theta_2)} e^{-i(\theta_1 + \theta_2)} \). Other properties are summarized as follows.

1) De Moivre’s Theorem of RBs

If \( q = A e^{i\theta_i} e^{i\theta_k} \), then \( q^n = A^n e^{i(n\theta_i)} e^{i(n\theta_k)} \). \tag{38}

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:

\[ q^{-1} = A^{-1} e^{-\rho_1} e^{-k\theta} e^{-j\theta_2} . \]  

(38)

We have introduced three methods to calculate the inverse of an RB: \( c_1 \rightarrow c_2 \) 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 \( c_1 \) or \( c_2 \), where \( c_1 \) and \( c_2 \) are any complex numbers.
- The inverse of \( M_\delta \) 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

If \( q = Ae^{i\theta_1} e^{k\theta} e^{j\theta_2} \), then \( \bar{q} = Ae^{-i\theta_1} e^{-k\theta} e^{-j\theta_2} \).  

(39)

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

Proof:

a) Letting \( q' = Ae^{-i\theta_1} e^{-k\theta} e^{-j\theta_2} \), then \( q' \bar{\bar{q}} = A^2 = |q|^2 = q' \cdot (\cdot \cdot A = |q|) \)

\[ \bar{q} = q' = \begin{vmatrix} A & 0 \\ 0 & 1 \end{vmatrix} \]  

\[ \bar{q} \cdot q' = 0, \]  

(40)

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

\[ \bar{q} = q' = Ae^{-i\theta_1} e^{-k\theta} e^{-j\theta_2} . \]  

(41)

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

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

\[ e^{j\theta} = \cos \theta + j \cdot \sin \theta. \]  

(42)

Moreover, the following equations are satisfied:

\[ e^{j(-\theta)} = \cos(-\theta) + j \cdot \sin(-\theta) = \cos \theta - j \cdot \sin \theta \]

\[ e^{j\theta_1} e^{j\theta_2} = e^{j(\theta_1 + \theta_2)} \Rightarrow e^{j\theta} e^{-j\theta} = e^{j(0\cdot \theta)} = 1. \]  

(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 \( e^{j\theta} = \cos \theta + j \cdot \sin \theta \), 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 \( |q| = \| Ae^{j\theta} \| \| e^{j\theta_2} \| = A = \| q \| \), \( |e^{j\theta_2}| = 1 = (\cos^2 \theta_2 - \sin^2 \theta_2)^{1/2} \). This result is the same as the definition of double numbers [32], [33].

C. Relations Among the \( c_1 \rightarrow c_2 \) Form, Matrix Representation, and Polar Form

We have discussed the relation between the \( c_1 \rightarrow c_2 \) form and matrix representation in Section II. Now, we show the relation between the polar form and the \( c_1 \rightarrow c_2 \) form as well as the relation between the polar form and matrix representation.

1) Relations Between Polar Form and \( c_1 \rightarrow c_2 \) Form: The \( c_1 \rightarrow c_2 \) form of \( e^{j\theta_1}, e^{j\theta_2}, \) and \( e^{j\theta_2} \) can be calculated by (11):

\[ e^{j\theta_1} = e^{j\theta_1} e^{j\theta_2} e^{j\theta_2} , \]

\[ e^{j\theta_2} = e^{j\theta_2} e^{j\theta_2} , \]

\[ e^{j\theta_2} = e^{j\theta_2} e^{j\theta_2} \]

(44)

respectively. Here, we only prove the result of \( e^{j\theta_1} \).

Proof: Let \( e^{j\theta_1} = \cos \theta_1 + i \cdot \sin \theta_1 + j \cdot 0 = \cos \theta_1 + j \cdot 0 \),

where \( q_a = \cos \theta_1 + i \cdot \sin \theta_1, \theta_1 = 0 \). Then, \( q_{a+b} = q_a + q_b = q_a \), and \( q_{a-b} = q_a - q_b = q_b \). Hence, from (11), the \( c_1 \rightarrow c_2 \) form of \( e^{j\theta_1} \) is

\[ e^{j\theta_1} = e^{j\theta_1} e^{j\theta_2} e^{j\theta_2} . \]

Therefore, the \( c_1 \rightarrow c_2 \) form of \( q = Ae^{j\theta} e^{j\theta_2} e^{j\theta_2} \) is

\[ Ac^{j\theta} e^{j\theta_2} e^{j\theta_2} = e^{j\theta_{1+\theta_2}} \]

(45)

\[ |q_{a+b}| = |q| \]

(46)

2) Relations Between Polar Form and Matrix Representation: The matrix representation of \( e^{j\theta_1}, e^{j\theta_2}, \) and \( e^{j\theta_2} \) can be calculated by (14)

\[ e^{j\theta_1} \Leftrightarrow M_{\theta_1} \equiv \begin{bmatrix} 
\cos \theta_1 & -\sin \theta_1 & 0 & 0 \\
\sin \theta_1 & \cos \theta_1 & 0 & 0 \\
0 & 0 & \cos \theta_1 & -\sin \theta_1 \\
0 & 0 & \sin \theta_1 & \cos \theta_1 
\end{bmatrix} \]

(47)

\[ e^{j\theta_2} \Leftrightarrow M_{\theta_2} \equiv \begin{bmatrix} 
\cos \theta_2 & 0 & 0 & -\sin \theta_2 \\
0 & \sin \theta_2 & \cos \theta_2 & 0 \\
0 & \cos \theta_2 & \sin \theta_2 & 0 \\
0 & 0 & 0 & \cos \theta_2 
\end{bmatrix} \]

(48)
\[ e^{i\theta_j} \leftrightarrow M_{\theta_j} = \begin{bmatrix} \cosh \theta_j & 0 & \sinh \theta_j & 0 \\ 0 & \cosh \theta_j & 0 & \sinh \theta_j \\ \sinh \theta_j & 0 & \cosh \theta_j & 0 \\ 0 & \sinh \theta_j & 0 & \cosh \theta_j \end{bmatrix} \begin{bmatrix} \cosh \theta_j & 0 & \sinh \theta_j & 0 \\ 0 & 1 & 0 & 0 \\ \sinh \theta_j & 0 & \cosh \theta_j & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cosh \theta_j & 0 & \sinh \theta_j \\ 0 & 0 & 1 & 0 \\ 0 & \sinh \theta_j & 0 & \cosh \theta_j \end{bmatrix}. \] (49)

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

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

\[ f(m,n) = f_R(m,n) i + f_G(m,n) j + f_B(m,n) k \] (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 \( f_I(m,n) \), \( f_H(m,n) \), and \( f_S(m,n) \) by

\[ \begin{bmatrix} f_I(m,n) \\ f_H(m,n) \\ f_S(m,n) \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{6}} \\ \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} & -\frac{1}{\sqrt{6}} \\ \frac{1}{\sqrt{2}} & 0 \end{bmatrix} \begin{bmatrix} f_R(m,n) \\ f_G(m,n) \\ f_B(m,n) \end{bmatrix}. \] (51)

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

\[ f_A(m,n) = \sqrt{f_R^2(m,n) + f_S^2(m,n)} \]
\[ = \sqrt{f_R^2(m,n) + f_G^2(m,n) + f_B^2(m,n)} \] (54)
\[ f_H(m,n) = \cos^{-1} \left( \frac{f_I(m,n)}{f_A(m,n)} \right) \]
\[ = \sin^{-1} \left( \frac{f_S(m,n)}{f_A(m,n)} \right). \] (55)

The ranges of \( f_H(m,n) \) and \( f_S(m,n) \) are \(-\pi \leq f_H(m,n) < \pi, -\pi/2 \leq f_S(m,n) \leq \pi/2\).

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 \( e^{j\theta} \) is equal to zero. This representation has the geometric meanings and gives many interesting and beautiful properties, which are summarized as follows.

1) \( q_r q_I + q_I q_R = 0 \), or \( q_r/q_I = (q_R/q_I) \).
2) The norm of \( q \) becomes \( |q| = (q_r^2 + q_I^2 + q_S^2)^{1/2} \).
3) The conjugation of \( q \) becomes \( \overline{q} = A e^{-i\theta} e^{-jb \theta} = q_r - q_i + q_j - q_k \).

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 \( e_1 \) and \( e_2 \) form.

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

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

• Type 1:

\[ F_{(RB1)}(p,s) = F_{a+b}(p,s) e_1 + F_{a-b}(p,s) e_2 \] (57)

where \( F_{a+b}(p,s) = DFT(f_{a+b}(m,n)), F_{a-b}(p,s) = DFT(f_{a-b}(m,n)) \). 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:

\[ F_{(RB2)}(p,s) = DFT(f_a(m,n)) + DFT(-f_b(m,n)) \] (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].
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 \( H_{(RB)}(p,s) \) 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

\[
g(m,n) = f(m,n) \ast_{RB} h(m,n) = \sum_{\tau=-M}^{M-1} \sum_{\eta=-N}^{N-1} f(m-\tau,n-\eta)h(\tau,\eta). \tag{59}\]

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]:

\[
g(m,n) = \text{IQFT}^{(2)} \left( F_{(RB)}(p,s)H_{(RB)}(p,s) + F_{(RB)}(p,s)H_{(RB)}(-p,-s)j \right). \tag{60}\]

For RBs, no matter which type of DFBFT 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].

\[
g(m,n) = \text{IDFBFT} \left( F_{(RB)}(p,s)H_{(RB)}(p,s) \right). \tag{61}\]

By (57) and (58), we totally require six complex 2-D FFTs 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 relationship between the input signal \( f(x,y) \) and output signal \( g(x,y) \) can be expressed as

\[
g(m,n) = f(m,n) \ast_{RB} h_s(m,n) \tag{62}\]

where

\[
h_s(m,n) = h_1(m,n) \ast_{RB} h_2(m,n) \ast_{RB} \cdots \ast_{RB} h_T(m,n). \tag{63}\]

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

\[
G_{(RB)}(p,s) = F_{(RB)}(p,s)H_{(RB)}(p,s), \quad \text{where}\]

\[
H_{(RB)}(p,s) = \prod_{t=1}^{T} H_t(p,s). \tag{64}\]

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 \( F_{(RB)}(p,s) \) and \( G_{(RB)}(p,s) \) is the same as the one of quaternions in [16]:

\[
G_{(RB)}(p,s) = F_{(RB)}(p,s)H_{(RB)}(p,s), \quad \text{where}\]

\[
H_{(RB)}(p,s) = \sum_{t=1}^{T} H_t(p,s). \tag{65}\]

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

\[
g(m,n) = f(m,n) \otimes_{RB} h(m,n) = \sum_{\tau=-M}^{M-1} \sum_{\eta=-N}^{N-1} f(\tau,\eta)h(\tau-m,\eta-n). \tag{66}\]

Correlation can be viewed as a special case of convolution

\[
f(m,n) \otimes_{RB} h(m,n) = f(m,n) \ast_{RB} \overline{h(-m,-n)} \tag{67}\]

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

\[
g(m,n) = \text{IDFBFT} \left( F(p,s)H_{(RB)}(p,s) \right). \tag{68}\]

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

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

\[
\hat{g}(m,n) \equiv f(m,n) \otimes_{RB,PO} h(m,n) \equiv \text{IDFBFT} \left( \frac{F(p,s)H_{(RB)}(p,s)}{|F(p,s)|} \right). \tag{69}\]
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 Lorenz 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:

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

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

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

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 $g_{H,8} = q_1 + q_2 i_8$, where $q_1$ and $q_2$ are two RBs, and $i_8^2 = 1$. 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, $(i = 0, 1, \ldots, (2^n - 1))$ and the complex impulse response between channels $x$ and $y$ are $h_{x,y}(m)$.

where “⊕” 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 $n = 1$ [a symmetric two channel system as Fig. 3(a)]

$$g_0(m) = f_0(m) * h_0(m) + f_1(m) * h_1(m)$$
$$g_1(m) = f_0(m) * h_1(m) + f_1(m) * h_0(m). \quad (72)$$

We need four conventional convolutions to implement the system.

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

$$f_{RB}(m) = f_0(m) + f_1(m)j$$
$$h_{RB}(m) = h_0(m) + h_1(m)j$$
$$g_{RB}(m) = g_0(m) + g_1(m)j \quad (73)$$

then

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

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

$$g_0(m,n) \pm g_1(m,n) = (f_0(m,n) \pm f_1(m,n)) \ast (h_0(m,n) \pm h_1(m,n)) \quad (75).$$

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

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

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

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

then $g_{H,8}(m) = f_{H,8}(m) *_{H,8} h_{H,8}(m)$ where $*_{H,8}$ 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 $h_{m,n}$ and six complex signals $f_a$, $f_b$, $g_a$, $g_b$, $h_{m,a}$, and $h_{m,b}$ as follows:

$$f = f_1 + f_2 j, \quad f_a = f_1 + f_2, \quad f_b = f_1 - f_2$$
$$g = g_1 + g_2 j, \quad g_a = g_1 + g_2, \quad g_b = g_1 - g_2 \quad (78)$$
$$h_m = h_{m,1} + h_{m,2} j, \quad h_{m,a} = h_{m,1} + h_{m,2}$$
$$h_{m,b} = h_{m,1} - h_{m,2}, \quad \text{where} \quad (m = 1, 2, \ldots, n) \quad (79)$$

Authorized licensed use limited to: National Taiwan University. Downloaded on January 21, 2009 at 22:40 from IEEE Xplore. Restrictions apply.
and use the results of (74) and (75), then we get
\[ g = f *_{RB} h_1 *_{RB} h_2 *_{RB} \cdots *_{RB} h_n \]
\[ g_a = f_a * h_{1,a} * h_{2,a} * \cdots * h_{n,a}, \text{ and} \]
\[ g_b = f_b * h_{1,b} * h_{2,b} * \cdots * h_{n,b}. \]
(80)

Therefore, the system is equivalent to the one in Fig. 4(b), and the output signals are
\[ g_1 = \frac{(g_a + g_b)}{2}, \quad g_2 = \frac{(g_a - g_b)}{2}. \]
(81)

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

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 \( h_0(m,n) \) and input image \( f_0(m,n) \) into I-H-S color space by (51). Then, we use (53) to represent the color images.

\[ h(m,n) \equiv A_h(m,n)e^{i\theta_h(m,n)} \]
\[ f(m,n) \equiv A_f(m,n)e^{i\theta_f(m,n)} \]
(82)
where \( h(m,n) \) and \( f(m,n) \) are the RB representation of the reference pattern and input image, respectively.

---

**TABLE II**

<table>
<thead>
<tr>
<th>Properties</th>
<th>Quaternions</th>
<th>Reduced biquaternions</th>
</tr>
</thead>
<tbody>
<tr>
<td>2. Division algebra</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>3. Definitions of norm and conjugation</td>
<td>(</td>
<td>q</td>
</tr>
<tr>
<td>4. Multiplication Rule</td>
<td>Noncommutative</td>
<td>Commutative</td>
</tr>
<tr>
<td>5. Convolution</td>
<td>(F(f(t)*g(t)) = F(f(t))G(g(t)))</td>
<td>(F(f(t)*g(t)) = F(f(t))G(g(t)))</td>
</tr>
</tbody>
</table>

---

Step 2) Calculate the energy of reference pattern \( h(m,n) \) by (18):
\[ E_h = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} (|h(m,n)|^2) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} A_h^2(m,n) \] (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 \((= 1 + 0i + 0j + 0k)\).)

Step 3) Compute the RB Fourier transform of \( f(m,n) \) and \( h_n(m,n) \)
\[ F(p,s) = DRBFT (f(m,n)) \] and
\[ Hn(p,s) = DRBFT (hn(m,n)). \] (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):
\[ g(m,n) \equiv IDRBFT \left\{ F_{RB}(p,s)H_{C\rightarrow RB}(p,s) \right\}. \] (85)
\[ \hat{g}(m,n) \equiv IDRBFT \left\{ \left| F_{RB}(p,s)H_{C\rightarrow RB}(p,s) \right| \right\}. \] (Requirement 1) (86)
Step 5) Using the result of the phase-only correlation \( \hat{g}(m,n) \) 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):

\[ c_1 < |g(m_s, n_s)| < c_2 \]  

where \( c_1 < c_2 \), and \( c_1, c_2 \) are all near to 1.

- Requirement 3 (for average chromaticity match):

\[ \rho_1 \geq d_1 \quad (d_1 \approx 1 \text{ and } d_1 < 1) \]  

where

\[ \rho_1 = \frac{|g_r(m_s, n_s)|}{|g_r(m_s, n_s)| + |g_i(m_s, n_s)| + |g_j(m_s, n_s)| + |g_k(m_s, n_s)|} \]  

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):

\[ \rho_2 \geq d_2 \quad (d_2 \approx 1 \text{ and } d_2 < 1) \]  

where

\[ \rho_2 = \frac{|g_r(m_s, n_s)|}{|g_r(m_s, n_s)| + |g_i(m_s, n_s)| + |g_j(m_s, n_s)| + |g_k(m_s, n_s)|} \]  

- Requirement 5 (for average saturation match):

\[ \rho_3 \geq d_3, \quad (d_3 \approx 1 \text{ and } d_3 < 1) \]  

where

\[ \rho_3 = \frac{|g_r(m_s, n_s)| + |g_i(m_s, n_s)| + |g_j(m_s, n_s)| + |g_k(m_s, n_s)|}{|g_r(m_s, n_s)| + |g_i(m_s, n_s)| + |g_j(m_s, n_s)| + |g_k(m_s, n_s)|} \]  

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 \( \Delta_A \), hue difference \( \Delta_H \) or saturation difference \( \Delta_\phi \) between the reference pattern and the candidate objects with shape and size match.

\[ \Delta_A = \frac{1}{P} \sum_{m,n \in R} |h_A(m,n) - f_A(m + m_s, n + n_s)| \]  

\[ \Delta_H = \frac{1}{P} \sum_{m,n \in R} |h_H(m,n) - f_H(m + m_s, n + n_s)| \]  

\[ \Delta_\phi = \frac{1}{P} \sum_{m,n \in R} |h_\phi(m,n) - f_\phi(m + m_s, n + n_s)| \]  

where \( R \) is the region of the reference pattern, and \( P \) is the number of pixels in the region \( R \). If \( \Delta_A, \Delta_H, \) and \( \Delta_\phi \) is smaller than some threshold \( T_A, T_H \) and \( T_\phi \), respectively, then this object at \((m_s, n_s)\) 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

\[ \bar{h}_\phi(\tau, \eta) = \frac{A_\phi(\tau, \eta)e^{-jH_\phi(\tau, \eta)}e^{-j\phi_\phi(\tau, \eta)}}{\bar{E}_h}. \]  

(95)
Assuming that \( f'(\tau, \eta) = f(\tau + m, \eta + n) \equiv A'_{f}(\tau, \eta) e^{iH'_{f}(\tau, \eta)} e^{i\phi'_{f}(\tau, \eta)} \), then the correlation result becomes

\[
g(m, n) = \sum_{\tau=0}^{M-1} \sum_{\eta=0}^{N-1} f(\tau, \eta) h_{m}(\tau - m, \eta - n) = \sum_{\tau=0}^{M-1} \sum_{\eta=0}^{N-1} f(\tau + m, \eta + n) h_{m}(\tau, \eta)
\]

\[
= \frac{1}{E_{h}} \sum_{\tau=0}^{M-1} \sum_{\eta=0}^{N-1} A_{f}(\tau, \eta) A'_{f}(\tau, \eta)
\]

\[
\times e^{i(H'_{f}(\tau, \eta) - H_{f}(\tau, \eta))} e^{i\phi'_{f}(\tau, \eta) - \phi_{f}(\tau, \eta)}
\]

\[
\equiv \frac{1}{E_{h}} \sum_{\tau=0}^{M-1} \sum_{\eta=0}^{N-1} A_{f}(\tau, \eta) A'_{f}(\tau, \eta)
\]

\[
\times e^{i\Delta H(\tau, \eta)} e^{i\Delta \phi(\tau, \eta)}
\]

\[
\equiv g_{c}(m, n) + i \cdot g_{i}(m, n) + j \cdot g_{j}(m, n) + k \cdot g_{k}(m, n)
\]

(96)

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

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

\[
f(m, n) = h(m - m_{s}, n - n_{s})
\]

(97)

then from (96), the correlation result at \( (m_{s}, n_{s}) \) is

\[
g(m_{s}, n_{s}) = \frac{1}{E_{h}} \sum_{\tau=0}^{M-1} \sum_{\eta=0}^{N-1} A_{f}^{2}(\tau, \eta) = 1
\]

(98)

\[
\Rightarrow g_{c}(m_{s}, n_{s}) = 1
\]

\[
g_{i}(m_{s}, n_{s}) = g_{j}(m_{s}, n_{s}) = g_{k}(m_{s}, n_{s}) = 0.
\]

(99)

Therefore, \( |g(m_{s}, n_{s})| = 1, \rho_{l} = 1, \) 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 \( (\Delta H(m, n) = \Delta H_{0}, \Delta \phi(m, n) = \Delta \phi_{0}) \), then

\[
g(m_{s}, n_{s}) = e^{i\Delta H_{0}} e^{i\Delta \phi_{0}} \frac{1}{E_{h}} \sum_{\tau=0}^{M-1} \sum_{\eta=0}^{N-1} A_{f}(\tau, \eta) A_{f}(\tau, \eta).
\]

(100)

\[
|g(m_{s}, n_{s})| = |e^{i\Delta H_{0}}| |e^{i\Delta \phi_{0}}|
\]

\[
\times \left| \sum_{\tau=0}^{M-1} \sum_{\eta=0}^{N-1} A_{f}(\tau, \eta) A_{f}(\tau, \eta) \right|
\]

Moreover, if \( f'(\tau, \eta) = c h(\tau, \eta) \), 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.
Case 3) (Average chromaticity match):

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

$$\Delta H(\tau, \eta) = \Delta \phi(\tau, \eta) = 0 \quad (102)$$

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

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

$$\Rightarrow \rho_1 = 1. \quad (104)$$

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 $\rho_1$ 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. ($1.1 > |g(m_s, n_s)| > 0.9$, and $r_1 > 0.9$).

Fig. 7. Searching the objects that partially match the reference pattern. (a) $1.1 > |g(m_s, n_s)| > 0.9$, brightness match. (b) $p_2 > 0.9$, hue match. (c) $p_2 > 0.9$ 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 P so they are removed by the phase-only correlation.

• **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.
• 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 \( f_c(m,n) \) into I-H-S color space by (51). Then, we use (53) to represent the color images.

   \[
   f(m,n) = A_f(m,n)e^{iH_f(m,n)}e^{i\phi_f(m,n)} \quad (105)
   \]

   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:

   \[
   h(m,n) = \frac{1}{9} \begin{bmatrix}
   q_1^{-1} & q_3^{-1} & q_5^{-1} \\
   q_1^{-1} & q_3^{-1} & q_5^{-1} \\
   q_1^{-1} & q_3^{-1} & q_5^{-1}
   \end{bmatrix}
   \]

   where \( q_s = A_1e^{iH}e^{i\phi_s} \) is the selected color, \((106)\)

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

   \[
   g(m,n) = f(m,n) \ast_{RB} h(m,n). 
   \]

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

   b) Search the regions composed by the points with the same color as \( q_s \):

   From the filtered image \( g(m,n) \), to find the points with the same brightness and chromaticity as \( q_s \), 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 \( g_{match}(m,n) \) is the matching mask and that

   \[
   g_{match}(m,n) = \begin{cases} 
   1, & \text{if } g(m,n) \text{ satisfies (87) and (88)} \\
   0, & \text{else,} 
   \end{cases} \quad (108)
   \]

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

   c) Search the edges of the matching mask \( g_{match}(m,n) \)

   The edges of the matching mask \( g_{match}(m,n) \) are the color-sensitive edges of the color \( q_s \).

   Following the above process, we detect the edges of the regions that are composed by the color \( q_s \). 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 \( 1 + 0i + 0j + 0k \) are fully matched regions and are shown in Fig. 12(b). The fully matched edges are shown in Fig. 12(c). Besides, partially
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 $(0, 0, 0)$ to $(255, 255, 255)$ to $(-127.5, -127.5, -127.5)$ to $(127.5, 127.5, 127.5)$. The selected brown color is also shifted to $(R, G, B) = (69 - 127.5, 56 - 127.5, 33 - 127.5)$. 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
hybercomplex filters to find the edges between the two colors. The two left-side and right-side filters they used are

\[
L = \frac{1}{\sqrt{6}} \begin{bmatrix}
\mu c_1 & 0 & 0 \\
0 & 0 & 0 \\
1 & 1 & 1
\end{bmatrix}
\quad R = \frac{1}{\sqrt{6}} \begin{bmatrix}
1 & 1 & 1 \\
0 & 0 & 0 \\
\mu c_2 & \mu c_2 & \mu c_2
\end{bmatrix}
\]

(109)

where \(\mu c_1\) and \(\mu c_2\) 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):

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

(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 \(1 + 0l + 0j + 0k\). 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 \(q_2\))

\[
h_{\text{rot180}}(m, n) = \frac{1}{6} \begin{bmatrix}
q_1^{-1} & q_1^{-1} & q_1^{-1} \\
0 & 0 & 0 \\
q_2^{-1} & q_2^{-1} & q_2^{-1}
\end{bmatrix}
\]

(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

\[
g(m, n) = f(m, n) *_{\text{RB}} h(m, n).
\]

(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).
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:

\[
\begin{align*}
\frac{1}{6} \begin{bmatrix} q_{z1}^{-1} & q_{z2}^{-1} & 0 & 0 \\ 0 & 0 & q_{z1}^{-1} & q_{z2}^{-1} \\ q_{x1}^{-1} & 0 & 0 & q_{x2}^{-1} \\ 0 & q_{x1}^{-1} & q_{x2}^{-1} & 0 \end{bmatrix}, & \quad \frac{1}{6} \begin{bmatrix} q_{z1}^{-1} & 0 & 0 & q_{z2}^{-1} \\ q_{z1}^{-1} & q_{x1}^{-1} & 0 & q_{x2}^{-1} \\ 0 & 0 & q_{x1}^{-1} & q_{x2}^{-1} \\ q_{z1}^{-1} & 0 & q_{x1}^{-1} & q_{x2}^{-1} \end{bmatrix} \\
\frac{1}{6} \begin{bmatrix} 0 & q_{z1}^{-1} & q_{z2}^{-1} & 0 \\ q_{x1}^{-1} & 0 & 0 & q_{x2}^{-1} \\ q_{x1}^{-1} & q_{x2}^{-1} & 0 & 0 \\ q_{x1}^{-1} & 0 & q_{x2}^{-1} & 0 \end{bmatrix} & \quad \frac{1}{6} \begin{bmatrix} 0 & 0 & q_{z1}^{-1} & q_{z2}^{-1} \\ q_{x1}^{-1} & q_{x2}^{-1} & 0 & 0 \\ q_{x1}^{-1} & 0 & q_{x2}^{-1} & 0 \\ q_{x1}^{-1} & 0 & q_{x2}^{-1} & 0 \end{bmatrix}
\end{align*}
\]

(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) to (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 \((237 - 127.5, 157 - 127.5, 157 - 127.5, 255 - 127.5, 255 - 127.5)\). Then, we follow the process as above. The threshold \(d_4\) 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 \(e_1 \sim e_2\) 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

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.