Efficient Implementation of Quaternion Fourier Transform, Convolution, and Correlation by 2-D Complex FFT

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

***Abstract*—The recently developed concepts of quaternion Fourier transform (QFT), quaternion convolution (QCV), and quaternion correlation, which are based on quaternion algebra, have been found to be useful for color image processing. However, the necessary computational algorithms and their complexity still need some attention. In this paper, we will develop efficient algorithms for QFT, QCV, and quaternion correlation. The conventional complex two-dimensional (2-D) Fourier transform (FT) is used to implement these quaternion operations very efficiently. By these algorithms, we only need two complex 2-D FTs to implement a QFT, six complex 2-D FTs to implement a one-side QCV or a quaternion correlation and 12 complex 2-D FTs to implement a two-side QCV, and the efficiency of these quaternion operations is much improved. Meanwhile, we also discuss two additional topics. The first one is about how to use QFT and QCV for quaternion linear time-invariant (QLTI) system analysis. This topic is important for quaternion filter design and color image processing. Besides, we also develop the spectrum-product QCV. It is the improvement of the conventional form of QCV. For any arbitrary input functions, it always corresponds to the product operation in the frequency domain. It will be very useful for quaternion filter design.**

***Index Terms*—Quaternion convolution, quaternion correlation, quaternion Fourier transform.**

I. Introduction

The concept of the quaternion was introduced by Hamilton in 1843 [1]. It is the generalization of a complex number. A complex number has two components: the real and the imaginary part. The quaternion, however, has four components, i.e., one real part and three imaginary parts:

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

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

$$\begin{aligned} i^2 &= j^2 = k^2 = -1 \\ i\,j &= -\,j\,i = k \\ j\,k &= -\,k\,j = i \\ k\,i &= -\,i\,k = j. \end{aligned} \tag{2}$$

The quaternion can be used for three-entry or four-entry vector analysis [2]. Recently, the quaternion has also been used for color image analysis. In (1), we can use q_i, q_j, and q_k to represent the R, G, and B values of a color image pixel, respectively, and set q_r as 0.

Based on the concept of quaternion, the *quaternion Fourier transform* (QFT) has been introduced recently. There are many different types of QFT. The earliest definition of QFT is the two-side form as follows[3], [4], [15]:

$$H_{(q)}(w,v) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{-iwx} \cdot h(x,y) \cdot e^{-jvy} \cdot dxdy. \tag{3}$$

In fact, the QFT defined above can be generalized as [5]

$$H_{(q)}(w,v) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{-\mu_1 wx} \cdot h(x,y) \cdot e^{-\mu_2 vy} \cdot dxdy \tag{4}$$

where μ_1 and μ_2 are two unit pure quaternions (i.e., the quaternions with unit magnitude having no real part) that are orthogonal to each other:

$$\begin{aligned} \mu_1 &= \mu_{1,i} \cdot i + \mu_{1,j} \cdot j + \mu_{1,k} \cdot k \\ \mu_2 &= \mu_{2,i} \cdot i + \mu_{2,j} \cdot j + \mu_{2,k} \cdot k \\ \mu_{1,i}^2 + \mu_{1,j}^2 + \mu_{1,k}^2 &= \mu_{2,i}^2 + \mu_{2,j}^2 + \mu_{2,k}^2 = 1 \\ &(\text{i.e., } \mu_1^2 = \mu_2^2 = -1) \end{aligned} \tag{5}$$

$$\mu_{1,i} \cdot \mu_{2,i} + \mu_{1,j} \cdot \mu_{2,j} + \mu_{1,k} \cdot \mu_{2,k} = 0. \tag{6}$$

Equation (3) is the special case of (4) in which $\mu_1 = i$, and $\mu_2 = j$. Except for (4), there are also other types of QFT. Recently, the left-side form of QFT was introduced in [5]:

$$H_{(q)}(w,v) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{-\mu_1 (wx+vy)} \cdot h(x,y) \cdot dxdy \tag{7}$$

where μ_1 is any unit pure quaternion. Besides, we can also define the right-side QFT as the transpose transform of (7) [5]. Therefore, there are at least three types of QFT.

- **Type 1 QFT** (two-side):

$$H_{(q1)}(w,v) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{-\mu_1 wx} \cdot h(x,y) \cdot e^{-\mu_2 vy} \cdot dxdy \tag{8}$$

  where μ_1 and μ_2 satisfy (5) and (6).
- **Type 2 QFT** (left-side):

$$H_{(q2)}(w,v) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{-\mu_1 (wx+vy)} \cdot h(x,y) \cdot dxdy. \tag{9}$$

Manuscript received November 21, 2000; revised August 21, 2001. The associate editor coordinating the review of this paper and approving it for publication was Dr. Paulo J. S. G Ferreira.

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

Publisher Item Identifier S 1053-587X(01)09596-4.

- **Type 3 QFT** (right-side):

$$H_{(q3)}(w,v) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h(x,y)\cdot e^{-\mu_1(wx+vy)}\cdot dxdy. \tag{10}$$

Their inverse [i.e., *inverse quaternion Fourier transform* (IQFT)] are as follows.

- **Type 1 IQFT** (two-side):

$$h(x,y) = \left(4\pi^2\right)^{-1}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{\mu_1 wx} H_{(q1)}(w,v) e^{\mu_2 vy}\, dwdv. \tag{11}$$

- **Type 2 IQFT** (left-side):

$$h(x,y) = \left(4\pi^2\right)^{-1}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{\mu_1(wx+vy)} H_{(q2)}(w,v)\, dwdv; \tag{12}$$

- **Type 3 IQFT** (right-side):

$$h(x,y) = \left(4\pi^2\right)^{-1}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} H_{(q3)}(w,v) e^{\mu_1(wx+vy)}\, dwdv. \tag{13}$$

As the case of continuous QFT, there are also at least three types of **discrete quaternion Fourier transform** (**DQFT**) [3], [5], [6], [19].

- **Type 1 DQFT** (two-side):

$$H_{(q1)}(p,s) = \sum_{m=0}^{M-1}\sum_{n=0}^{N-1} e^{-\mu_1 2\pi(pm/M)} h(m,n)\, e^{-\mu_2 2\pi(sn/M)}. \tag{14}$$

- **Type 2 DQFT** (left-side):

$$H_{(q2)}(p,s) = \sum_{m=0}^{M-1}\sum_{n=0}^{N-1} e^{-\mu_1 2\pi((pm/M)+(sn/M))}\, h(m,n). \tag{15}$$

- **Type 3 DQFT** (right-side):

$$H_{(q3)}(p,s) = \sum_{m=0}^{M-1}\sum_{n=0}^{N-1} h(m,n)\, e^{-\mu_1 2\pi((pm/M)+(sn/M))}. \tag{16}$$

QFT and DQFT are useful for color image processing, especially for the color-sensitive smoothing, edge detection, and data compression [7]–[11], but research on the efficient algorithms of QFT and DQFT are not sufficient. This limits the utilities of QFT and DQFT. Although [12] discussed the efficient algorithm of Type 2 DQFT and [17] proposed a method to implement the Type 1 DQFT, neither discussed the efficient algorithms for all types of QFT and DQFT. In [17], although the idea is creative, one must design another algorithm using quaternion arithmetic other than using the well-known FFT algorithm to implement the DQFT.

In Section II of our paper, we will develop the efficient algorithms of all types of QFT, and with our algorithms, we can just use the structure of the original FFT directly to implement the QFT. Because the efficient algorithms of DQFT are very similar to those of the continuous QFT, in this paper, we discuss the efficient algorithms of the continuous QFT.

In Section III, we will discuss the efficient algorithms of QCV, including one-side and two-side QCVs. In Section IV, we will discuss how to use QCV for quaternion linear time-invariant (QLTI) system analysis. We will show that if the impulse responses are all even functions, then the analysis of QLTI systems is as simple as the analysis of conventional LTI systems. In Section V, we will define a new type of QCV. For this type of QCV, the QCV in the time domain always corresponds to the product operation in the frequency domain. This is useful in quaternion filter design. In Section VI, we will discuss the efficient algorithms of quaternion correlation.

In this paper, we will use the following abbreviations:

FT	Fourier transform.
IFT	Inverse Fourier transform.
QFT	Quaternion Fourier transform.
IQFT	Inverse quaternion Fourier transform.
QCV	Quaternion convolution.
QCR	Quaternion correlation.

We will also use the following notation:

- $h_r(x,y)$, $h_i(x,y)$, $h_j(x,y)$, $h_k(x,y)$: Real part, i-part, j-part, and k-part of $h(x,y)$.
- $\overline{}^{\mu}$: Partial conjugation. For example

$$\overline{h(x,y)}^{j,k} = h_r(x,y) + i\cdot h_i(x,y) - j\cdot h_j(x,y) - k\cdot h_k(x,y) \tag{17}$$

$$\overline{h(x,y)}^{i,k} = h_r(x,y) - i\cdot h_i(x,y) + j\cdot h_j(x,y) - k\cdot h_k(x,y) \tag{18}$$

- Q^{μ}: Extracting some parts of original function. For example

$$\begin{aligned} Q^{r,i}(h(x,y)) &= h_r(x,y) + i\cdot h_i(x,y)\\ Q^{j,k}(h(x,y)) &= j\cdot h_j(x,y) + k\cdot h_k(x,y). \end{aligned} \tag{19}$$

- $H_{(q)}(w,v)$: Transform results of QFT for $h(x,y)$ (not specified for which type).
- $H_{(q1)}(w,v)$, $H_{(q2)}(w,v)$, $H_{(q3)}(w,v)$: Transform results of QFT of types 1, 2, and 3 for $h(x,y)$.

II. Efficient Implementation of Quaternion Fourier Transform

A. Implementation of Type 1 QFT

We first discuss the implementation of type 1 QFT. To simplify our discussion, we first discuss the special case that $\mu_1 = i$ and $\mu_2 = j$. We note that if

$$H_c(w,v) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{-iwx}\cdot h(x,y)\cdot e^{-ivy}\cdot dxdy \tag{20}$$

then

$$\begin{aligned} &\frac{[H_c(w,v) + H_c(w,-v)]}{2}\\ &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{-iwx}\cdot h(x,y)\cdot \cos(vy)\cdot dxdy\\ &\frac{[H_c(w,v) - H_c(w,-v)]}{2}\\ &= -\left[\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{-iwx}\cdot h(x,y)\cdot \sin(vy)\cdot dxdy\right]\cdot i \end{aligned} \tag{21}$$

and therefore

$$\frac{H_c(w,v)+H_c(w,-v)}{2}+\frac{H_c(w,v)-H_c(w,-v)}{2}\cdot(-k) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{-iwx}\, h(x,y)\, e^{-jvy}\, dxdy. \quad (22)$$

Thus

$$H_{(q1)}(w,v) = \frac{[H_c(w,v)\cdot(1-k)+H_c(w,-v)\cdot(1+k)]}{2}. \quad (23)$$

If we want to compute the QFT, we can first compute the complex 2-D FT of input function as (20) and then use (23) to compute the QFT.

We note in (20) that the input $h(x,y)$ is a quaternion function and not a complex function; therefore, we cannot just use one complex 2-D FT to implement it. Instead, we can first decompose $h(x,y)$ as

$$\begin{aligned} h(x,y) =& h_a(x,y)+h_b(x,y)\cdot j \\ \text{where } h_a(x,y) =& h_r(x,y)+h_i(x,y)\cdot i \\ h_b(x,y) =& h_j(x,y)+h_k(x,y)\cdot i. \end{aligned} \quad (24)$$

Then, (20) can be rewritten as

$$\begin{aligned} H_c(w,v) =& \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{-iwx}h_a(x,y)\cdot e^{-ivy}\cdot dxdy \\ &+\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{-iwx}h_b(x,y)\cdot je^{-ivy}\cdot dxdy \\ =& \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{-iwx}h_a(x,y)\cdot e^{-ivy}\cdot dxdy \\ &+\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{-iwx}h_b(x,y)\cdot e^{ivy}\cdot j\cdot dxdy \\ H_c(w,v) =& \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{-iwx}e^{-ivy}\cdot h_a(x,y)\cdot dxdy \\ &+\left\lfloor\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{-iwx}e^{-ivy}\right. \\ &\left.\cdot h_b(x,-y)\cdot dxdy\right\rfloor\cdot j. \end{aligned} \quad (25)$$

From (25), we can calculate (20) by two complex 2-D FTs.

From the above discussion, we can compute the type 1 QFT with three steps as follows.

1) Decompose the input function as (24).
2) Calculate $H_c(w,v)$ from (25).
3) Calculate the transform result of QFT from $H_c(w,v)$ by (23).

If we sample the x-, y-, w-, and v-axes as

$$\begin{aligned} x =& m\Delta_x,\ y=n\Delta_y,\ w=p\Delta_w \\ v =& q\Delta_v,\ m,p\in[-M_0,M_0],\ n,q\in[-N_0,N_0] \\ \Delta_x\cdot\Delta_w =& \frac{2\pi}{M},\ \Delta_y\cdot\Delta_v=\frac{2\pi}{N},\ M=2M_0+1 \\ N =& 2N_0+1 \end{aligned} \quad (26)$$

then (25) can be implemented as

$$\begin{aligned} \tilde{H}_{(q1)}&(p\Delta_w,q\Delta_v) \\ =& \sum_{m=-M_0}^{M_0}\sum_{n=-N_0}^{N_0} e^{-i(((2\pi pm)/M)+((2\pi qn)/N))} \\ &\cdot h_a(m\Delta_x,n\Delta_y) \\ &+\left[\sum_{m=-M}^{M_0}\sum_{n=-N_0}^{N_0} e^{-i(((2\pi pm)/M)+((2\pi qn)/N))}\right. \\ &\left.\cdot h_b(m\Delta_x,-n\Delta_y)\right]j. \end{aligned} \quad (27)$$

The QFT can be implemented by two $M\times N$ point 2-D DFTs. Since each 2-D DFT requires $MN\cdot\log_2 MN$ real number multiplications [13], to implement QFT, we totally require

$$2MN\cdot\log_2 MN \quad (28)$$

real number multiplications.

In the above, we only discuss the special case wherein $\mu_1=i$ and $\mu_2=j$. In general cases, we just need to modify the above process a little. Suppose that

$$\begin{aligned} \mu_1 =& \mu_{1,1}\cdot i+\mu_{1,2}\cdot j+\mu_{1,3}\cdot k \\ \mu_2 =& \mu_{2,1}\cdot i+\mu_{2,2}\cdot j+\mu_{2,3}\cdot k \\ \mu_3 =& \mu_1\cdot\mu_2=\mu_{3,1}\cdot i+\mu_{3,2}\cdot j+\mu_{3,3}\cdot k. \end{aligned} \quad (29)$$

Then, we express $h(x,y)$ as

$$\begin{aligned} h(x,y) =& h_r(x,y)+h_1(x,y)\cdot\mu_1 \\ &+h_2(x,y)\cdot\mu_2+h_3(x,y)\cdot\mu_3 \end{aligned} \quad (30)$$

and the relation between $h_1(x,y)$, $h_2(x,y)$, $h_3(x,y)$, and $h_i(x,y)$, $h_j(x,y)$, $h_k(x,y)$ is

$$\begin{bmatrix} h_1(x,y) \\ h_2(x,y) \\ h_3(x,y) \end{bmatrix} = \begin{bmatrix} \mu_{1,1} & \mu_{2,1} & \mu_{3,1} \\ \mu_{1,2} & \mu_{2,2} & \mu_{3,2} \\ \mu_{1,3} & \mu_{2,3} & \mu_{3,3} \end{bmatrix}^{-1} \begin{bmatrix} h_i(x,y) \\ h_j(x,y) \\ h_k(x,y) \end{bmatrix}. \quad (31)$$

Then, we can implement the type 1 QFT from the process as follows.

1) First, decompose the input function as

$$h(x,y) = h_a(x,y)+h_b(x,y)\cdot\mu_2 \quad (32)$$

where

$$\begin{aligned} h_a(x,y) =& h_r(x,y)+h_1(x,y)\cdot\mu_1 \\ h_b(x,y) =& h_2(x,y)+h_3(x,y)\cdot\mu_1. \end{aligned} \quad (33)$$

2) Then, calculate $H_c(w,v)$ from

$$\begin{aligned} H_c(w,v) =& \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{-\mu_1(wx+vy)}\, h_a(x,y)\, dxdy \\ &+\left\lfloor\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{-\mu_1(wx+vy)}\, h_b(x,-y)\, dxdy\right\rfloor\cdot\mu_2. \end{aligned} \quad (34)$$

3) Then, calculate the transform result of QFT by

$$H_{(q1)}(w,v) = H_c(w,v)\,\frac{1-\mu_3}{2}+H_c(w,-v)\,\frac{1+\mu_3}{2}. \quad (35)$$

Although the integration operations in (34) are different from the complex 2-D FT, they can still be implemented by complex 2-D FT. Since all the exponential terms and the input functions $h_a(x,y)$, and $h_b(x,y)$ in (34) contain the real part and the μ_1 part, as well as $\mu_1^2=-1$, there is no problem to use the efficient algorithm of the complex 2-D FT to implement the integration operations in (34), except that all the is are changed to μ_1s. Even when $\mu_1 \neq i$ and $\mu_2 \neq j$, we require two complex 2-D DFTs to implement the type 1 QFT defined as (8), and the amount of real number multiplications required is also about

$$2MN \cdot \log_2 MN. \tag{36}$$

We can also use the similar process as above to implement the type 1 QFT. It also requires $2MN \cdot \log_2 MN$ real multiplications.

B. Implementation of QFT of Types 2, 3

We now discuss the efficient algorithms of QFT of types 2 and 3. The efficient algorithm of the discrete quaternion Fourier transform (DQFT) of type 2 has been introduced in [12]. We can modify their algorithms a little and derive the efficient algorithms of continuous QFT of types 2 and 3. To implement the *type 2 QFT and type 3*, we can implement the following process.

1) Find a unit pure quaternion μ_2 orthogonal to μ_1, that is

$$\begin{aligned}\mu_1 =& \mu_{1,i} \cdot i + \mu_{1,j} \cdot j \\ &+ \mu_{1,k} \cdot k \\ \mu_2 =& \mu_{2,i} \cdot i + \mu_{2,j} \cdot j \\ &+ \mu_{2,k} \cdot k \\ \mu_{1,i} \cdot \mu_{2,i} + \mu_{1,j} \cdot \mu_{2,j} + \mu_{1,k} \cdot \mu_{2,k} =& 0.\end{aligned} \tag{37}$$

We define μ_3 as the product of μ_1 and μ_2. If $\mu_1 = i$, we can choose $\mu_2 = j$ and $\mu_3 = k$ and save this step.

2) By the same method as (31)–(33), we decompose the input function $h(x,y)$ into $h_a(x,y)$ and $h_b(x,y)$.

3) For the *type 2 QFT*, we can calculate the transform result from

$$\begin{aligned}H_{(q2)}(w,v) =& \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{-\mu_1(wx+vy)}\, h_a(x,y)\, dxdy \\ &+ \left[\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{-\mu_1(wx+vy)}\, h_b(x,y)\, dxdy\right] \cdot \mu_2.\end{aligned} \tag{38}$$

For the *type 3 QFT*, we can calculate the transform result from

$$\begin{aligned}H_{(q3)}(w,v) =& \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{-\mu_1(wx+vy)} h_a(x,y) dxdy \\ &+ \left[\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{\mu_1(wx+vy)}\, h_b(x,y)\, dxdy\right] \cdot \mu_2.\end{aligned} \tag{39}$$

Thus, we require two complex 2-D FTs or one 2-D FT and one 2-D IFT to implement the QFT of types 2 and 3. Therefore, the amounts of real multiplications required for the QFT of types 2 and 3 are

$$2MN \cdot \log_2 MN. \tag{40}$$

The complexities of the QFT of types 1–3 are the same.

We can also use the similar way as above to implement the IQFT of types 2 and 3, except that the roles of $h(x,y)$ and $H_{(q2,3)}(w,v)$ are exchanged, and in (39) and (40), $\pm\mu_1(wx+vy)$ are replaced by $\mp\mu_1(wx+vy)$. We also require $2MN \cdot \log_2 MN$ real number multiplications to implement the IQFT of types 2 and 3.

C. Simplified Implementation of QFT for Some Special Cases

We have discussed the implementation of the QFT requiring $2MN \cdot \log_2 MN$ real multiplication operations. In fact, for some special cases, we can further simplify the implementation of the QFT.

We usually use the quaternion to deal with the color image. When we use the quaternion to express a color image, we usually express it as

$$h(m,n) = h_R(m,n) \cdot i + h_G(m,n) \cdot j + h_B(m,n) \cdot k. \tag{41}$$

That is, we use i-, j-, and k-parts to represent the R, G, and B parts of the color image and set the real part to 0. Therefore, for the applications about the color image, the input of the QFT is usually a pure quaternion function (i.e., the real part is 0).

For the complex FT, if the input function is pure real, then we can use the real-valued Fourier transform (RFT) instead of the complex FT. The 2-D complex FT requires $MN \cdot \log_2 MN$ real multiplications, but the 2-D RFT requires $(MN/2) \cdot \log_2 MN$ real number multiplications [13]. Similarly, if the input of QFT is a pure quaternion function, we can also simplify the implementation. In this case, since $h_r(x,y)=0$, then in (34), (38), and (39), $h_a(x,y)$ is a pure imaginary function, and the first integration in these equations is, in fact, the 2-D FT for the pure imaginary input. For example, (34) can be implemented as

$$H_c(w,v) = \mathrm{RFT}^{\mu_1}(h_1(x,y)) \cdot \mu_1 + \mathrm{FT}^{\mu_1}(h_b(x,-y)) \cdot \mu_2 \tag{42}$$

where FT^{μ_1} represents the complex 2-D FT (but all is are changed as μ_1s), and RFT^{μ_1} represents the complex 2-D FT for pure real input. Since RFT^{μ_1} requires $(MN/2) \cdot \log_2 MN$ real number multiplications, the total amount of real number multiplications required is

$$\left(\frac{3MN}{2}\right) \cdot \log_2 MN. \tag{43}$$

Comparing it with (34) or (40), we find that one fourth of the real multiplications are saved.

Except for the above cases, if the input is symmetric, then the implementation of QFT can also be simplified. For example, if the input $h(x,y)$ is even-even (i.e., $h(x,y)=h(\pm x, \pm y)$), even-odd, odd-even, or odd-odd, then we can use the 2-D cosine/sine transforms instead of the 2-D FTs in (34), (38), and (39), and three fourths of the real multiplications can be saved. If the input is symmetric in the x-axis (i.e., $h(x,y)=\pm h(-x,y)$) or in the y-axis, we can save half of the real multiplications.

III. Implementation of Quaternion Convolution

There are two different definitions for the QCV.

- **One-side QCV** [3]:

$$g(x,y) = f(x,y) *_q h(x,y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x-\tau, y-\eta)\, h(\tau,\eta)\cdot d\tau d\eta. \quad (44)$$

- **Two-side QCV** [7]:

$$g(x,y) = f(x,y) *_q \{h_1(x,y), h_2(x,y)\} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h_1(\tau,\eta)\, f(x-\tau, y-\eta)\, h_2(\tau,\eta) \cdot d\tau d\eta. \quad (45)$$

For the conventional convolution operation, if $f_3(t)$ is the convolution output of $f_1(t)$ and $f_2(t)$, then $f_3(t)$ can be calculated from

$$f_3(t) = \text{IFT}(\text{FT}(f_1(t)) \cdot \text{FT}(f_2(t))). \quad (46)$$

We can calculate the conventional convolution by the efficient algorithms of the complex FT. Similarly, it is reasonable to think that we can also find some simple relations between the QFT and the QCV and use the efficient algorithms of the QFT to implement the QCV.

In [14], how to use the hypercomplex Wiener–Khintchine theorem to compute the quaternion correlation was discussed, and we can modify their algorithm to implement the one-side QCV, but their algorithm requires eight complex 2D FTs and is not simple enough. In the following, we will show that we can use *six complex 2-D FTs* to implement the one-side QCV. For the two-side QCV, we require *12 complex 2–D FTs* to implement it.

A. Implementation of the One-Side QCV by the Type 2 QFT

To derive the relation, we can first separate $f(x,y)$ in (44) into two parts.

$$\begin{aligned} f(x,y) &= f_a(x,y) + f_b(x,y)\cdot j \\ \text{where } f_a(x,y) &= f_r(x,y) + f_i(x,y)\cdot i \\ f_b(x,y) &= f_j(x,y) + f_k(x,y)\cdot i. \end{aligned} \quad (47)$$

The one-side QCV can be rewritten as

$$\begin{aligned} g(x,y) &= f(x,y) *_q h(x,y) \\ &= f_a(x,y) *_q h(x,y) \\ &\quad + f_b(x,y)\cdot j *_q h(x,y). \end{aligned} \quad (48)$$

Then, if $F_{a(q2)}(w,v)$, $F_{b(q2)}(w,v)$, and $H_{(q2)}(w,v)$ are the type 2 QFT of $f_a(x,y)$, $f_b(x,y)$, and $h(x,y)$, we can calculate the result of the QCV as follows.

- **Relation between the type 2 QFT and the one-side QCV:**

$$g(x,y) = \text{IQFT}^{(2)}\left(F_{a(q2)}(w,v)H_{(q2)}(w,v) + F_{b(q2)}(w,v)\cdot jH_{(q2)}(-w,-v)\right) \quad (49)$$

where IQFT$^{(2)}$ means the type 2 IQFT. Its proof is shown in the Appendix.

By (49), we can use the efficient algorithm of the type 2 QFT to implement the one-side QCV. In Section II-B, we have stated that we can use two 2-D Fourier transforms (FTs) to implement the type 2 QFT or IQFT; therefore, in (49), we need two 2-D FTs to calculate $H_{(q2)}(w,v)$ or IQFT$^{(2)}$. We notice that if the input function of the type 2 QFT has a real part and and i part, then for the algorithm in Section II-B, we can set $\mu_1 = i$, $\mu_2 = j$, and $\mu_3 = k$, and in (38), $h_b(x,y) = h_j(x,y) + i\cdot h_k(x,y) = 0$; thereforfe, in this case, we require one 2-D FT to implement the type 2 QFT. Thus, in (49), we need one 2-D FT to calculate $F_{a(q2)}(w,v)$. Similarly, we also need one 2-D FT to calculate $F_{b(q2)}(w,v)$. Thus, by the method of (49), we totally require *six complex 2-D FTs* to implement the one-side QCV. We remember that for the conventional 2-D convolution, we totally require three complex 2-D FTs to implement it. The complexity of the one-side QCV is twice of that of the conventional 2-D convolution.

In some special conditions, the relation between the one-side QCV and the type 2 QFT can be simplified. In the case where $h(x,y)$ has the even symmetry relation

$$h(x,y) = h(-x,-y) \quad (50)$$

then we can prove that the QFT of $h(x,y)$ also has the same symmetry relation

$$H_{(q2)}(w,v) = H_{(q2)}(-w,-v) \quad (51)$$

and the relation between the QCV and the type 2 QFT in (49) can be simplified

$$\begin{aligned} g(x,y) &= \text{IQFT}^{(2)}\left(\left(F_{a(q2)}(w,v) + F_{b(q2)}(w,v)j\right)H_{(q2)}(w,v)\right) \\ &= \text{IQFT}^{(2)}\left(F_{(q2)}(w,v)H_{(q2)}(w,v)\right) \end{aligned} \quad (52)$$

or

$$G_{(q2)}(w,v) = F_{(q2)}(w,v)\cdot H_{(q2)}(w,v) \quad (53)$$

When $\boldsymbol{h(x,y) = h(-x,-y)}$, *the QCV operation in the space domain corresponds to the product operation in the frequency domain*. This is the same as the case of the conventional convolution. In the case where $h(x,y)$ has the odd symmetry relation as

$$h(x,y) = -h(-x,-y) \quad (54)$$

since $H_{(q2)}(w,v) = -H_{(q2)}(-w,-v)$, (49) can be simplified to

$$G_{(q2)}(w,v) = \overline{F_{(q2)}(w,v)}^{j,k} H_{(q2)}(w,v). \quad (55)$$

However, we must notice that the benefits described above do not exist for the case in which $f(x,y) = \pm f(-x,-y)$. We must remember that the QCV operation is not commutative.

In general, if $h(x,y)$ is neither even nor odd, from (53) and (55), we can conclude that the relation between the inputs and the output of the QCV in the frequency domain can be written as

$$G_{(q2)}(w,v) = F_{(q2)}(w,v)\,H_{(q2)e}(w,v) + \overline{F_{(q2)}(w,v)}^{j,k} H_{(q2)o}(w,v) \quad (56)$$

where $H_{(q2)e}(w,v)$and $H_{(q2)o}(w,v)$ are the even and odd parts of $H_{(q2)}(w,v)$

$$H_{(q2)e}(w,v) = \frac{\lfloor H_{(q2)}(w,v) + H_{(q2)}(-w,-v) \rfloor}{2}$$
$$H_{(q2)o}(w,v) = \frac{\lfloor H_{(q2)}(w,v) - H_{(q2)}(-w,-v) \rfloor}{2}.$$

The relation in (56) is useful for the quaternion linear time invariant system analysis.

B. Implementation of One-Side QCV by Type 1 and Type 3 QFT

In Section III-A, we have discussed how to use the type 2 QFT to calculate the one-side QCV. In fact, we can also use the type 1 and type 3 QFT to calculate the one-side QCV.

We first discuss the case of the type 3 QFT. Suppose $g(x,y)$ is the output of the one-side QCV of $f(x,y)$ and $h(x,y)$, as in (44). We can first separate $h(x,y)$ as $h_a(x,y) + j \cdot h_d(x,y)$, where

$$h_a(x,y) = h_r(x,y) + h_i(x,y) \cdot i$$
$$h_d(x,y) = h_j(x,y) - h_k(x,y) \cdot i. \tag{57}$$

$g(x,y)$ can be calculated from the following.

- **Relation between the type 3 QFT and the one-side QCV:**

$$g(x,y) = \text{IQFT}^{(3)} \left(F_{(q3)}(w,v) H_{a(q3)}(w,v) + F_{(q3)}(-w,-v) \cdot j H_{d(q3)}(w,v) \right) \tag{58}$$

where $\text{IQFT}^{(3)}$ means the type 3 IQFT. The process of proof is similar as the process of proving the relation between the type 2 QFT and one-side QCV. Since we need two complex 2-D FTs to calculate $F_{(q3)}(w,v)$ or $\text{IQFT}^{(3)}$ and just one complex 2-D FT to calculate $H_{a(q3)}(w,v)$ and $H_{d(q3)}(w,v)$, we totally require *six complex 2-D FTs* to compute the QCV by the type 3 QFT. This is the same as the case of computing the QCV by the type 2 QFT.

The relation in (58) can be simplified in some special cases. When

$$f(x,y) = f(-x,-y) \tag{59}$$

since $F_{(q3)}(w,v) = F_{(q3)}(-w,-v)$, (58) can be simplified as

$$G_{(q3)}(w,v) = F_{(q3)}(w,v) H_{(q3)}(w,v). \tag{60}$$

In this case, the QCV corresponds to the product operation in the frequency domain, and when

$$f(x,y) = -f(-x,-y) \tag{61}$$

the relation between the QCV and the type 3 QFT can be simplified to

$$G_{(q3)}(w,v) = F_{(q3)}(w,v) \overline{H_{(q3)}(w,v)}^{j,k}. \tag{62}$$

We can compare (59)∼(62) with (50)∼(55). We find that they are very much alike, except that the roles of $f(x,y)$ and $h(x,y)$ are exchanged. We can conclude that when we want to implement the one-side QCV, then the *type 2 QFT* is suitable for the case in which $\boldsymbol{h(x,y) = \pm h(-x,-y)}$, and the *type 3 QFT* is suitable for the case in which $\boldsymbol{f(x,y) = \pm f(-x,-y)}$.

In general, if $h(x,y)$ is neither even nor odd, then by combining (60) with (62), we obtain

$$G_{(q3)}(w,v) = F_{(q3)e}(w,v) H_{(q3)}(w,v) + F_{(q3)o}(w,v) \overline{H_{(q3)}(w,v)}^{j,k} \tag{63}$$

where $F_{(q3)e}(w,v) = [F_{(q3)}(w,v) + F_{(q3)}(-w,-v)]/2$, and $F_{(q3)o}(w,v) = [F_{(q3)}(w,v) - F_{(q3)}(-w,-v)]/2$. This relation is also useful for the quaternion linear time-invariant system analysis.

We can also use the type 1 QFT to implement the QCV. We can first separate $f(x,y)$ into $f_a(x,y) + f_b(x,y) \cdot j$, where $f_a(x,y) = f_r(x,y) + f_i(x,y)i$ and $f_b(x,y) = f_j(x,y) + f_k(x,y)i$, as (47). Then, the correlation output $g(x,y)$ can be calculated from the following.

- **Relation between the type 1 QFT and one-side QCV:**

$$g(x,y) = \text{IQFT}^{(1)} \Big\lfloor F_{a(q1)}(w,v) Q^{r,j} \left(H_{(q1)}(w,v) \right) + F_{a(q1)}(w,-v) \cdot Q^{i,k} \left(H_{(q1)}(w,v) \right) + F_{b(q1)}(w,-v) j Q^{r,j} \left(H_{(q1)}(-w,v) \right) + F_{b(q1)}(w,v) j Q^{i,k} \cdot \left(H_{(q1)}(-w,v) \right) \Big\rfloor \tag{64}$$

where Q is defined as (19), and $\text{IQFT}^{(1)}$ means the type 1 IQFT.

Since we require one complex 2-D FT to compute $F_{a(q1)}(w,v)$ or $F_{b(q1)}(w,v)$, two complex 2-D FTs to compute $\text{IQFT}^{(1)}$ or $H_{(q1)}(w,v)$, we totally require *six complex 2-D FTs* to implement the QCV by the type 1 QFT. It is the same as the case when using the type 2 and type 3 QFT. However, some problems exist. That is, (64) is more complicated than (49) and (58), and this relation can be simplified into the product operation only in some very special conditions. Although the complexities are the same, we prefer using the type 2 and type 3 QFT rather than using the type 1 QFT to implement the one-side QCV.

C. Implementation of Two-Side QCV

In Section sIII-A and B, we have discussed the implementation of one-side QCV. Here, we will discuss how to implement the two-side QCV, which is defined in (45). We can first convert the two-side QCV into several one-side QCV's, then use the efficient algorithm of one-side QCV to implement it. In (45), we can first separate $h_1(x,y)$ and $f(x,y)$ into $h_{1,a}(x,y) + h_{1,b}(x,y) \cdot j$ and $f_a(x,y) + f_b(x,y) \cdot j$, where

$$h_{1,a}(x,y) = h_{1,r}(x,y) + h_{1,i}(x,y) \cdot i$$
$$h_{1,b}(x,y) = h_{1,j}(x,y) + h_{1,k}(x,y) \cdot i$$
$$f_a(x,y) = f_r(x,y) + f_i(x,y) \cdot i$$
$$f_b(x,y) = f_j(x,y) + f_k(x,y) \cdot i. \tag{65}$$

Then, since

$$\begin{aligned} h_{1,a}(\tau,\eta)\, f_a(x-\tau,y-\eta) =& f_a(x-\tau,y-\eta) \\ &\cdot h_{1,a}(\tau,\eta) \\ h_{1,b}(\tau,\eta)\cdot j\cdot f_a(x-\tau,y-\eta) =& \overline{f_a(x-\tau,y-\eta)} \\ &\cdot h_{1,b}(\tau,\eta)\cdot j, \\ h_{1,a}(\tau,\eta)\, f_b(x-\tau,y-\eta)\cdot j =& f_b(x-\tau,y-\eta)\cdot j \\ &\cdot \overline{h_{1,a}(\tau,\eta)} \\ h_{1,b}(\tau,\eta)\cdot j\cdot f_b(x-\tau,y-\eta)\cdot j =& \overline{f_b(x-\tau,y-\eta)}\cdot j \\ &\cdot \overline{h_{1,b}(\tau,\eta)}\cdot j \end{aligned} \tag{66}$$

(45) can be rewritten as

$$\begin{aligned} g(x,y) =& \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f_a(x-\tau,y-\eta)\; h_3(\tau,\eta)\; d\tau d\eta \\ &+ \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \overline{f_a(x-\tau,y-\eta)}\, h_4(\tau,\eta)\, d\tau d\eta \\ &+ \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f_b(x-\tau,y-\eta)\; jh_5(\tau,\eta)\, d\tau d\eta \\ &+ \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \overline{f_b(x-\tau,y-\eta)} \\ &\cdot jh_6(\tau,\eta)\, d\tau d\eta \end{aligned} \tag{67}$$

where

$$\begin{aligned} h_3(\tau,\eta) =& h_{1,a}(\tau,\eta)\cdot h_2(\tau,\eta) \\ h_4(\tau,\eta) =& h_{1,b}(\tau,\eta)\cdot j\cdot h_2(\tau,\eta) \\ h_5(\tau,\eta) =& \overline{h_{1,a}(\tau,\eta)}\cdot h_2(\tau,\eta) \\ h_6(\tau,\eta) =& \overline{h_{1,b}(\tau,\eta)}\cdot j\cdot h_2(\tau,\eta). \end{aligned} \tag{68}$$

That is, we can use four one-side QCVs to calculate one two-side QCV. Then, with the aid of the efficient algorithms of the one-side QCV introduced in Section III-A and B, we can use the the type 1–3 QFTs to implement the two-side QCV.

When we use the type 2 QFT, then from (49) and (67) and the fact that

$$\begin{aligned} \text{QFT}^{(2)}\left(\overline{f_{a\,\text{or}\,b}(x,y)}\right) =& \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{-i(wx+vy)} \\ &\cdot \overline{f_{a\,\text{or}\,b}(x,y)}\, dxdy \\ =& \overline{F_{a\,\text{or}\,b,(q2)}(-w,-v)} \end{aligned} \tag{69}$$

we can implement the two-side QCV by the type 2 QFT as follows.

- **Relation between the type 2 QFT and two-side QCV:**

$$\begin{aligned} g(x,y) =& \text{IQFT}^{(2)} \\ &\cdot \Big[F_{a(q2)}(w,v)\, H_{3(q2)}(w,v) \\ &\quad + \overline{F_{a(q2)}(-w,-v)}\cdot jH_{4(q2)}(w,v) \\ &\quad + F_{b(q2)}(w,v)\cdot jH_{5(q2)}(-w,-v) \\ &\quad + \overline{F_{b(q2)}(-w,-v)} \\ &\quad \cdot jH_{6(q2)}(-w,-v)\Big]. \end{aligned} \tag{70}$$

Since we require two complex 2-D FTs to compute $\text{IQFT}^{(2)}$, one complex 2-D FT to compute $F_{a(q2)}(w,v)$, or $F_{b(q2)}(w,v)$ and two complex 2-D FTs to compute each of $H_{s(q2)}(w,v)$, $(s=3,4,5,6)$, we totally require *12 complex 2-D FTs* to implement the two-side QCV. The complexity of the two-side QCV is twice of that of the one-side QCV.

Although we can also use the type 1 and type 3 QFTs to implement the two-side QCV, the relation between the type 1 and type 3 QFT and the two-side QCV is rather complicated; therefore, we suggest that it is better to use the type 2 QFT to implement the two-side QCV.

IV. Quaternion Linear Time-Invariant System Analysis

The complex FT is useful for the analysis of linear time-invariant (LTI) systems, especially for the filter design. This is because the conventional convolution corresponds to the product operation in the frequency domain. With the aid of the complex FT, it is easy to analyze the effects of the LTI system.

However, there is little research on how to use the QFT for quaternion linear time-invariant (QLTI) system analysis. This topic was discussed in [3]. The work in [3] is suited to the case where the impulse response of the QLTI system is a pure real function, but it is not general enough. Until now, no simple relation between QCV and QFT has been found; therefore, QLTI system analysis is still a difficult. In this paper, however, we have found some simple and general relations between the QCV and the QFT. With these relations, we can use the QFT to analyze the QLTI system easily.

A. Using QFT for Quaternion Linear Time-Invariant System Analysis

The QLTI can be represented by the one-side QCV, which is defined in (44). If $f(x,y)$ and $g(x,y)$ are the input and output of a QLTI system, $h(x,y)$ is the impulse response, and $f(x,y)$ and $g(x,y)$ should have the relation as

$$\begin{aligned} g(x,y) &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x-\tau,y-\eta)\; h(\tau,\eta)\; d\tau d\eta \\ &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(\tau,\eta)\; h(x-\tau,y-\eta)\; d\tau d\eta. \end{aligned} \tag{71}$$

It matches the definition of one-side QCV. We can use the one-side QCV to represent the QLTI system. Then, from the relations between the QFT and the one-side QCV shown in (49), (58), and (64), we can analyze the QLTI systems easily in the frequency domain. We suggest that using the type 2 or type 3 QFT to analyze the QLTI system is more convenient than using the type 1 QFT.

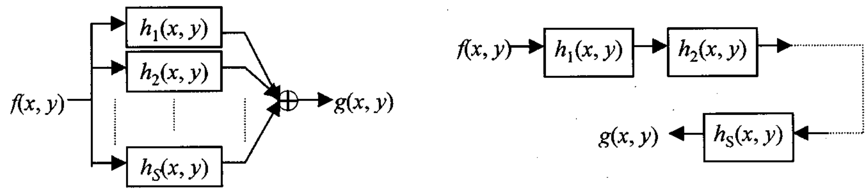
Then, we will discuss how to use the QFT to analyze the *combination of QLTI systems*. In Fig. 1, we show two basic ways to combine the QLTI systems.

When we combine the QLTI systems in *parallel*, as in Fig. 1(a), the relation between the input $f(x,y)$ and the output $g(x,y)$ can be expressed as

$$g(x,y) = f(x,y) *_q h_t(x,y)$$

where

$$h_t(x,y) = \sum_{s=1}^{S} h_s(x,y) \tag{72}$$



(a) (b)

Fig. 1. Combination of QLTI systems. (a) In parallel and (b) in series.

and $*_q$ is the one-side QCV. In the frequency domain, regardless of the type of QFT, the relation between $F_{(q)}(w,v)$ and $G_{(q)}(w,v)$ can be expressed as

$$G_{(q)}(w,v) = F_{(q)}(w,v)\,H_{t(q)}(w,v)$$
where
$$H_{t(q)}(w,v) = \sum_{s=1}^{S} H_{s(q)}(w,v). \tag{73}$$

It is the same as the case of combining the conventional LTI systems in parallel.

When we combine the QLTI systems in *series*, as in Fig. 1(b), the relation between $f(x,y)$ and $g(x,y)$ can be expressed as

$$g(x,y) = f(x,y) *_q h_t(x,y)$$
where
$$h_t(x,y) = h_1(x,y) *_q h_2(x,y) *_q \cdots *_q h_S(x,y). \tag{74}$$

Then, in the frequency domain, when we use the *type 2 QFT*, the relation between $F_{(q2)}(w,v)$ and $G_{(q2)}(w,v)$ is

$$\begin{aligned} G_{(q2)}(w,v) = & \left[F_{(q2)}(w,v) \quad \overline{F_{(q2)}(w,v)}^{j,k} \right] \\ & \cdot \begin{bmatrix} H_{1\,(q2)e}(w,v) & \overline{H_{1\,(q2)o}(w,v)}^{j,k} \\ H_{1\,(q2)o}(w,v) & \overline{H_{1\,(q2)e}(w,v)}^{j,k} \end{bmatrix} \\ & \cdot \begin{bmatrix} H_{2(q2)e}(w,v) & \overline{H_{2(q2)o}(w,v)}^{j,k} \\ H_{2(q2)o}(w,v) & \overline{H_{2(q2)e}(w,v)}^{j,k} \end{bmatrix} \\ & \cdots \begin{bmatrix} H_{S-1,(q2)e}(w,v) & \overline{H_{S-1,(q2)o}(w,v)}^{j,k} \\ H_{S-1,(q2)o}(w,v) & \overline{H_{S-1,(q2)e}(w,v)}^{j,k} \end{bmatrix} \\ & \cdot \begin{bmatrix} H_{S(q2)e}(w,v) \\ H_{S(q2)o}(w,v) \end{bmatrix} \end{aligned} \tag{75}$$

where $H_{s(q2)e}(w,v)$ and $H_{s(q2)o}(w,v)$ are the even part and odd part of $H_{s(q2)}(w,v)$

$$H_{s(q2)e}(w,v) = \frac{\left[H_{s(q2)}(w,v) + H_{s(q2)}(-w,-v)\right]}{2}$$
$$H_{s(q2)o}(w,v) = \frac{\left[H_{s(q2)}(w,v) - H_{s(q2)}(-w,-v)\right]}{2}. \tag{76}$$

The proof of (75) is shown in the Appendix. In fact, (75) can also be rewritten as

$$\begin{aligned} G_{(q2)}(w,v) = & F_{(q2)}(w,v)\,H_{t(q2)e}(w,v) \\ & + \overline{F_{(q2)}(w,v)}^{j,k} H_{t(q2)o}(w,v) \end{aligned} \tag{77}$$

where $H_{t(q2)}(w,v)$ is the type 2 QFT of $h_t(x,y)$ [which is defined as (74)], and $H_{t(q2)e}(w,v)$ and $H_{t(q2)o}(w,v)$ are the even and odd parts of $H_{t(q2)}(w,v)$. They can be calculated from

$$\begin{aligned} & \begin{bmatrix} H_{t(q2)e}(w,v) \\ H_{t(q2)o}(w,v) \end{bmatrix} \\ & = \begin{bmatrix} H_{1\,(q2)e}(w,v) & \overline{H_{1\,(q2)o}(w,v)}^{j,k} \\ H_{1\,(q2)o}(w,v) & \overline{H_{1\,(q2)e}(w,v)}^{j,k} \end{bmatrix} \\ & \cdot \begin{bmatrix} H_{2(q2)e}(w,v) & \overline{H_{2(q2)o}(w,v)}^{j,k} \\ H_{2(q2)o}(w,v) & \overline{H_{2(q2)e}(w,v)}^{j,k} \end{bmatrix} \\ & \cdots \begin{bmatrix} H_{S-1,(q2)e}(w,v) & \overline{H_{S-1,(q2)o}(w,v)}^{j,k} \\ H_{S-1,(q2)o}(w,v) & \overline{H_{S-1,(q2)e}(w,v)}^{j,k} \end{bmatrix} \\ & \cdot \begin{bmatrix} H_{S(q2)e}(w,v) \\ H_{S(q2)o}(w,v) \end{bmatrix}. \end{aligned} \tag{78}$$

When we use the *type 3 QFT* instead of the type 2, then (78) is changed to

$$\begin{aligned} G_{(q3)}(w,v) = & \left[F_{(q3)e}(w,v)\, F_{(q3)o}(w,v) \right] \\ & \cdot \begin{bmatrix} H_{1\,(q3)e}(w,v) & H_{1\,(q3)o}(w,v) \\ \overline{H_{1\,(q3)o}(w,v)}^{j,k} & \overline{H_{1\,(q3)e}(w,v)}^{j,k} \end{bmatrix} \\ & \cdot \begin{bmatrix} H_{2(q3)e}(w,v) & H_{2(q3)o}(w,v) \\ \overline{H_{2(q3)o}(w,v)}^{j,k} & \overline{H_{2(q3)e}(w,v)}^{j,k} \end{bmatrix} \\ & \cdots \begin{bmatrix} H_{S-1,(q3)e}(w,v) & H_{S-1,(q3)o}(w,v) \\ \overline{H_{S-1,(q3)o}(w,v)}^{j,k} & \overline{H_{S-1,(q3)e}(w,v)}^{j,k} \end{bmatrix} \\ & \cdot \begin{bmatrix} H_{S(q3)}(w,v) \\ \overline{H_{S(q3)}(w,v)}^{j,k} \end{bmatrix}. \end{aligned} \tag{79}$$

where $F_{(q3)e}(w,v)$ and $F_{(q3)o}(w,v)$ are even/odd parts of $F_{(q3)}(w,v)$. Its proof is similar to the proof of (75). It is hard to use the type 1 QFT to represent the series combination of QLTI systems.

Applying the relation of (73), (78), and (79), we can use the QFT to represent many different combinations of QLTI systems. For example, for the QLTI systems combined as Fig. 2, if we use the type 2 QFT to represent the above system, then from (73) and (78), we have (80), shown at the bottom of the next page, where

$$\begin{aligned} \begin{bmatrix} H_{5(q2)e}(w,v) \\ H_{5(q2)o}(w,v) \end{bmatrix} = & \begin{bmatrix} H_{1(q2)e}(w,v) & \overline{H_{1(q2)o}(w,v)}^{j,k} \\ H_{1(q2)o}(w,v) & \overline{H_{1(q2)e}(w,v)}^{j,k} \end{bmatrix} \\ & \begin{bmatrix} H_{2(q2)e}(w,v) \\ H_{2(q2)o}(w,v) \end{bmatrix}. \end{aligned} \tag{81}$$

With the aid of QFT, especially the QFT of types 2 and 3, the analysis of QLTI systems and their combination becomes much simpler.

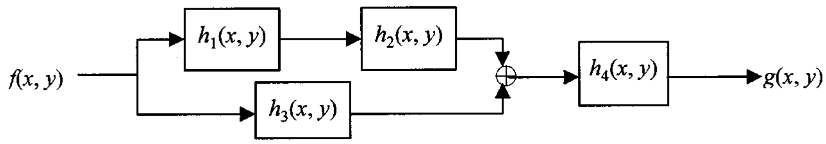


Fig. 2. Combination of quaternion linear time-invariant (LTI) systems.

B. Simplifying the QLTI System Analysis and Quaternion Filter Design

In Section III, we have stated that in some conditions, the relations between the QCV and QFT can be simplified as the product operation in frequency domain. We can restrict the forms of the input or the impulse response of QLTI system to simplify the analysis of the QLTI system.

For example, if we use the type 2 QFT for QLTI system analysis, we can restrict the impulse response of the QLTI system as an even function

$$h(x,y)=h(-x,-y). \tag{82}$$

In this case, the type 2 QFT of $h(x,y)$, $f(x,y)$, and $g(x,y)$ will have the relation as

$$G_{(q2)}(w,v)=F_{(q2)}(w,v)\,H_{(q2)}(w,v). \tag{83}$$

The type 2 QFT is suitable to analyze the QLTI system with even impulse response. In fact, in practical applications, especially for the quaternion filter design, the impulse response of the QLTI system usually satisfies the relation that $h(x,y)=h(-x,-y)$. Similarly, the type 3 QFT is suitable for the case in which the input is an even function

$$f(x,y)=f(-x,-y). \tag{84}$$

It is very convenient to simplify the QLTI system into the product operation in the frequency domain, especially for the *quaternion filter design*. For example, we can use $f(x,y)$ as the received signal and $h(x,y)$ as the impulse response of quaternion filter. If we want to remove the high-frequency noise that interferes with $f(x,y)$, we can first design the transfer function of quaternion lowpass filter as

$$\begin{aligned}H_q(w,v)=&1, \quad \text{for } |w|\le B_w \text{ and } |v|\le B_v\\ H_q(w,v)=&0, \quad \text{otherwise}\end{aligned} \tag{85}$$

and then

$$h(x,y)=\left(\pi^2xy\right)^{-1}\sin\left(\frac{B_wx}{2}\right)\sin\left(\frac{B_vy}{2}\right). \tag{86}$$

We note, in this case, that since $h(x,y)=h(-x,-y)$, the relation between the QCV and the type 2 QFT can be simplified as (83). From (83)

$$\begin{aligned}G_{(q2)}(w,v)=&F_{(q2)}(w,v), \quad \text{for } |w|\le B_w \text{ and } |v|\le B_v\\ G_{(q2)}(w,v)=&0, \quad \text{otherwise.}\end{aligned} \tag{87}$$

When we convolve the input $f(x,y)$ with $h(x,y)$, then the low-frequency components of $f(x,y)$ are preserved, and the high-frequency components of $f(x,y)$ are filtered out. This is the same as the conventional case.

The analysis of the combination of QLTI systems can also be simplified when the impulse responses of all stages are even functions. When we combine QLTI systems in series, as in Fig. 1(b), if

$$h_s(x,y)=h_s(-x,-y) \qquad \text{for } s=1,2,\cdots,S \tag{88}$$

then since $H_{s(q2)e}(w,v)=H_{s(q2)}(w,v)$, $H_{s(q2)o}(w,v)=0$ for all s; therefore, (75) can be simplified to

$$\begin{aligned}G_{(q2)}(w,v)=F_{(q2)}(w,v)\,&H_{1(q2)}(w,v)\\ &\cdot H_{2(q2)}(w,v)\cdots H_{S(q2)}(w,v).\end{aligned} \tag{89}$$

In this case, the combination of QLTI systems corresponds to the continued product of the type 2 QFT of the impulse responses of each stage in the frequency domain. This is the same as the case of series combination of conventional LTI systems. Thus, we can conclude the following.

- When all the impulse responses of QLTI systems are even functions, then the analysis of the combination of QLTI systems by the type 2 QFT is the same as the analysis of the combination of conventional LTI systems.

For example, in Fig. 2, if $h_1(x,y)$, $h_2(x,y)$, $h_3(x,y)$, and $h_4(x,y)$ are all even functions that satisfy $h_s(x,y)=h_s(-x,-y)$, then, similar to the case of the combination of conventional LTI systems, the relation between $G_{(q2)}(w,v)$ and $F_{(q2)}(w,v)$ is

$$\begin{aligned}G_{(q2)}(w,v)=F_{(q2)}(w,v)\,&\left[H_{1(q2)}(w,v)\,H_{2(q2)}(w,v)\right.\\ &\left.+H_{3(q2)}(w,v)\right]H_{4(q2)}(w,v).\end{aligned} \tag{90}$$

$$\begin{aligned}G_{(q2)}(w,v)=&\left[F_{(q2)}(w,v)\quad \overline{F_{(q2)}(w,v)}^{j,k}\right]\\ &\cdot\begin{bmatrix}H_{5(q2)e}(w,v)+H_{3(q2)e}(w,v) & \overline{H_{5(q2)o}(w,v)+H_{3(q2)o}(w,v)}^{j,k}\\ H_{5(q2)o}(w,v)+H_{3(q2)o}(w,v) & \overline{H_{5(q2)e}(w,v)+H_{3(q2)e}(w,v)}^{j,k}\end{bmatrix}\\ &\cdot\begin{bmatrix}H_{4(q2)e}(w,v)\\ H_{4(q2)o}(w,v)\end{bmatrix}\end{aligned} \tag{80}$$

V. Spectrum-Product Quaternion Convolution

A. Definition of Spectrum-Product QCV

In Section III, we have stated in some conditions that the relations between one-side QCV and the QFT of types 1–3 can be simplified as the product operation in the frequency domain. Although, in these cases, the effects of QCV are much easier to analyze, there must be some restrictions for the input of QCV.

In this section, we will reverse the problem and ask for what definitions of QCV, *regardless of what forms of the inputs are, do the QCV in the time domain always correspond to the product operation in the frequency domain*?

We first discuss the case of the type 2 QFT. Suppose

$$G_{(q2)}(w,v) = F_{(q2)}(w,v)\,H_{(q2)}(w,v) \tag{91}$$

where $G_{(q2)}(w,v)$, $F_{(q2)}(w,v)$, and $H_{(q2)}(w,v)$ are the type 2 QFT of $g(x,y)$, $f(x,y)$, and $h(x,y)$, respectively. Then, we can prove (the proof is shown in the Appendix)

$$g(x,y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f_a(x-\tau, y-\eta)\,h(\tau,\eta)\,d\tau d\eta + \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f_b(x+\tau, y+\eta)\,jh(\tau,\eta)\,d\tau d\eta \tag{92}$$

where

$$f_a(x,y) = f_r(x,y) + f_i(x,y)\cdot i, \quad f_b(x,y) = f_j(x,y) + f_k(x,y)\cdot i. \tag{93}$$

We can define the new type of QCV [we call it the spectrum-product QCV] as follows.

- **Spectrum-product QCV suitable for type 2 QFT:**

$$g(x,y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} Q^{r,i}\left[f(x-\tau,y-\eta)\right] h(\tau,\eta)d\tau d\eta + \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} Q^{j,k}\left[f(x+\tau,y+\eta)\right] \cdot h(\tau,\eta)d\tau d\eta \tag{94}$$

where Q is defined as (19). It can also be rewritten as

$$g(x,y) = Q^{r,i}(f(x,y)) *_q h(x,y) + Q^{j,k}(f(x,y)) *_q h(-x,-y) \tag{95}$$

where $*_q$ means the one-side QCV defined as (44). For *any inputs* $f(x,y)$, $h(x,y)$, the following relation is always satisfied for the spectrum-product QCV defined as (94)

$$G_{(q2)}(w,v) = F_{(q2)}(w,v)\,H_{(q2)}(w,v). \tag{96}$$

Because the input and output have a very simple relation and the effects of inputs are easier to analyze, the spectrum-product QCV is more suitable for the application of quaternion filter design than the QCVs defined as (44) and (45).

Of course, we also need three QFTs of type 2 (six complex 2-D FTs in total) to implement the spectrum-product QCV.

Similarly, for the type 3 QFT, we can define the spectrum-product QCV as follows.

- **Spectrum-product QCV suitable for type 3 QFT:**

$$g(x,y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x-\tau,y-\eta)\; Q^{r,i}\left[h(\tau,\eta)\right] d\tau d\eta + \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(\tau-x,\eta-y)\; Q^{j,k}\left[h(\tau,\eta)\right] d\tau d\eta. \tag{97}$$

For the type 1 QFT, we can define the spectrum-product QCV as follows.

- **Spectrum-product QCV suitable for type 1 QFT:**

$$g(x,y) = f_{ey}(x,y) *_q h_{ex}(x,y) + f_{oy}(x,y) *_q \overline{h_{ex}(x,y)}^{i,k} + \overline{f_{ey}(x,y)}^{j,k} *_q h_{ox}(x,y) - \overline{f_{oy}(x,y)}^{j,k} *_q \overline{h_{ox}(x,y)}^{i,k} \tag{98}$$

where

$$\begin{aligned} f_{ey}(x,y) &= \frac{[f(x,y)+f(x,-y)]}{2} \\ f_{oy}(x,y) &= \frac{[f(x,y)-f(x,-y)]}{2} \\ h_{ex}(x,y) &= \frac{[h(x,y)+h(-x,y)]}{2} \\ h_{ox}(x,y) &= \frac{[h(x,y)-h(-x,y)]}{2}. \end{aligned} \tag{99}$$

We can prove, for the spectrum-product QCV defined as (97) and (98), regardless of what forms the input functions $f(x,y)$, $h(x,y)$ take, that the following relation is always satisfied:

$$G_{(q)}(w,v) = F_{(q)}(w,v)H_{(q)}(w,v). \tag{100}$$

B. Properties and Applications of Spectrum-Product QCV

We can use the spectrum-product QCV introduced in Section V-A for *quaternion filter design and quaternion system design*. In Section IV, we discussed how to use the one-side QCV defined as (44) for quaternion filter and system design. Only when the impulse response of the system satisfies the following constraint:

$$h(x,y) = \pm h(-x,-y), \tag{101}$$

the relation between the input and the output can be expressed as the product operation in the frequency domain. Under other conditions, the relation is rather complicated, but when we use the spectrum-product QCV introduced in Section V-A for quaternion filter and system design, then the relation between the input and the output can always be expressed as the product operation in the frequency domain, *even when* (101) *is not satisfied*.

By the spectrum-product QCV defined in Section V-A, we can use the conventional method to design the quaternion filter. For example, if we want to design the equalizer filter by QFT, we can set the desired output as $g(x,y) = \delta(x,y)$. Then, since $G_q(w,v) = 1$ and from (96), we can design the equalizer filter as

$$h(x,y) = \text{IQFT}\left[F_q^{-1}(w,v)\right] \qquad \text{for any } f(x,y). \tag{102}$$

It is the same as the conventional case, but when we use the original QCV since the input and output has the relation (49), it is harder to design the equalizer filter $h(x,y)$ to obtain the

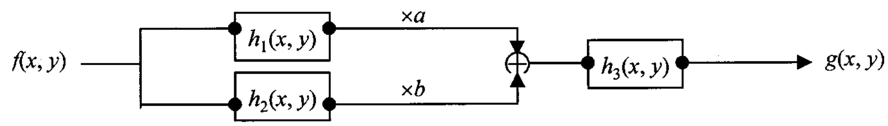


Fig. 3. Combination of spectrum-product QCV systems.

desired output $g(x,y) = \delta(x,y)$. For another example, if we want to design a bandpass filter by QFT with the passband of $\{(w,v)|w_1 < w < w_2, v_1 < v < v_2\}$ and $w_1 > 0$, $v_1 > 0$, as in the conventional case, we can just design the bandpass filter as

$$h(x,y) = \text{IQFT}\,[H_q(w,v)]$$

where

$$\begin{aligned} H_q(w,v) &= 1, \quad \text{for } w_1 < w < w_2, v_1 < v < v_2 \\ H_q(w,v) &= 0, \quad \text{otherwise.} \end{aligned} \tag{103}$$

Then, the QFT of the output $g(x,y)$ and the QFT of the input $f(x,y)$ have the relation

$$\begin{aligned} G_{(q2)}(w,v) &= F_{(q2)}(w,v), \text{ for } w_1 < w < w_2, v_1 < v < v_2 \\ G_{(q2)}(w,v) &= 0, \qquad \text{otherwise.} \end{aligned} \tag{104}$$

If we use the original definition of QCV, and we use (103) as the filter, then the relation between the QFTs of $g(x,y)$ and $f(x,y)$ become (105), shown at the bottom of the page, and it does not satisfy our requirement; therefore, for quaternion filter design, using the spectrum-product QCV is more convenient than using the original definition of QCV.

There is a limitation for the spectrum-product QCV. That is, it does not fully match the definition of linear time-invariant (LTI) system. The spectrum-product QCV defined as (94) is *time-invariant* and *quasilinear*. Suppose $g(x,y)$ is the output of spectrum-product QCV of $f(x,y)$ and $h(x,y)$. If we use $f(x-x_0, y-y_0)$ instead of $f(x,y)$, then the new output $g_0(x,y)$ is

$$\begin{aligned} g_0(x,y) = &\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} Q^{r,i}\left[f(x-x_0-\tau, y-y_0-\eta)\right] \\ &\quad \times h(\tau,\eta)\,d\tau d\eta \\ &+ \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} Q^{j,k}\left[f(x-x_0+\tau, y-y_0+\eta)\right] \\ &\quad \times h(\tau,\eta)\,d\tau d\eta \\ = &g(x-x_0, y-y_0). \end{aligned} \tag{106}$$

If the input has the displacement of (x_0, y_0), the output also has the displacement of (x_0, y_0). Thus, the spectrum-product QCV is *time-invariant*. Besides, it *almost* has the*linearity* property (we call it *quasilinearity*). Suppose that

$$\begin{aligned} g_a(x,y) &= f_a(x,y) *_{q(2)} h(x,y) \\ g_b(x,y) &= f_b(x,y) *_{q(2)} h(x,y). \end{aligned} \tag{107}$$

Here, we use $*_{q(2)}$ to denote the spectrum-product QCV defined as (94). If we use the linear combination of $f_a(x,y)$ and $f_b(x,y)$ as the new input, and the new output is $g_c(x,y)$

$$\begin{aligned} f_c(x,y) &= d \cdot f_a(x,y) + e \cdot f_b(x,y) \\ g_c(x,y) &= f_c(x,y) *_{q(2)} h(x,y) \end{aligned} \tag{108}$$

then we can prove

$$\begin{aligned} g_c(x,y) &= d \cdot g_a(x,y) + e \cdot g_b(x,y) \\ \text{when } Q^{j,k}(d) &= Q^{j,k}(e) = 0 \end{aligned} \tag{109}$$

and the above equation is *not satisfied only when the j- or k-part of d or e is nonzero*. Thus, although the spectrum-product QCV is not a LTI operation, it almost has the property of LTI.

In fact, the problem that spectrum-product QCV does not fully satisfy the LTI property has less effect on its applications. This problem is important only when we want to analyze the linear time-invariant system, but for the applications that are closely related to the spectrum analysis (such as quaternion filter design), whether the inputs and the output have the relation as (96) is more important, and this problem has less of an effect.

In summary, if we want to *design a quaternion system in the time domain*, such as the quaternion linear time-invariant system design, as in Section IV, then it still better to use the *original definition of QCV*. However, if we want to *design a quaternion system in frequency domain*, such as the application of *quaternion filter design*, then it is better to use the *spectrum-product QCV* since, in this case, the QFT of the output function is the product of the QFTs of the input functions.

The analysis of the combination of spectrum-product QCV systems is almost the same as the conventional case. We can see this in the system in Fig. 3. Here, we use block dots in the front and the back to represent the spectrum-product QCV system, i.e., the relations between the input and output of this system is the spectrum-product QCV, which is defined as (94). This is the key difference between Fig. 2 (which consists of original QCV systems) and Fig. 3.

$$\begin{aligned} G_{(q2)}(w,v) &= Q^{r,i}\left(F_{(q2)}(w,v)\right) \text{ for } w_1 < w < w_2, v_1 < v > v_2 \\ G_{(q2)}(w,v) &= Q^{j,k}\left(F_{(q2)}(w,v)\right) \text{ for } -w_1 < w < -w_2, -v_1 < v > -v_2 \\ G_{(q2)}(w,v) &= 0, \quad \text{otherwise} \end{aligned} \tag{105}$$

Suppose in Fig. 3 that the j- and k- parts of a and b are zero; then, as the conventional case, the relation between $f(x,y)$ and $g(x,y)$ has the following relation:

$$G_{(q2)}(w,v) = H_{3(q2)}(w,v)\left[a \cdot H_{1(q2)}(w,v) + b \cdot H_{2(q2)}(w,v)\right] F_{(q2)}(w,v) \tag{110}$$

when

$$Q^{j,k}(a) = Q^{j,k}(b) = 0 \tag{111}$$

regardless of what forms of the system responses $h_1(x,y), h_2(x,y)$ and $h_3(x,y)$ are. The analysis of the combination of spectrum-product QCV is simpler than the analysis of the combination of original definition of QCV. If the requirement of linearity is not necessary, then using the spectrum-product QCV will be more convenient.

VI. Implementation of Quaternion Correlation

In Section III, we discussed how to implement the QCV. In this section, we will use the result in Section III and discuss the implementation of quaternion correlation. In [14], the efficient algorithm of quaternion correlation was discussed, but in fact, their work can be further simplified. The *quaternion correlation (QCR)* is defined as

$$\begin{aligned} g(x,y) =& f(x,y) \otimes_q h(x,y) \\ =& \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x+\tau, y+\eta) \cdot \overline{h(\tau,\eta)} \cdot d\tau d\eta. \end{aligned} \tag{112}$$

In fact, QCR can be viewed as a special case of one-side QCV

$$f(x,y) \otimes_q h(x,y) = f(x,y) *_q \overline{h(-x,-y)}. \tag{113}$$

We can use the efficient algorithms of the one-side QCV to implement the QCR.

When we use the type 3 QFT to implement the QCR, we can use the efficient algorithm introduced in Section III-B, but $h(x,y)$ is changed as $\overline{h(-x,-y)}$. Because in (113)

$$\overline{h(-x,-y)} = \overline{h_a(-x,-y)} - j\,h_b(-x,-y)$$

where $h_a(x,y) = h_r(x,y) + h_i(x,y)i$ and $h_b(x,y) = h_j(x,y) + h_k(x,y)i$, and

$$\begin{aligned} QFT^{(3)}\left(\overline{h_a(-x,-y)}\right) =& \overline{H_{a(q3)}(w,v)} \\ QFT^{(3)}\left(-h_b(-x,-y)\right) =& -H_{b(q3)}(-w,-v), \end{aligned} \tag{114}$$

from (58), we can implement the QCR from the following.

- **Relation between the type 3 QFT and quaternion correlation:**

$$g(x,y) = \text{IQFT}^{(3)}\left(F_{(q3)}(w,v)\overline{H_{a(q3)}(w,v)} - F_{(q3)}(-w,-v) \cdot jH_{b(q3)}(-w,-v)\right). \tag{115}$$

When we use the type 2 QFT to implement the QCR, because

$$QFT^{(2)}\left(\overline{h(-x,-y)}\right) = \overline{Q^{r,i}\left(H_{(q2)}(w,v)\right)} - Q^{j,k}\left(H_{(q2)}(-w,-v)\right) \tag{116}$$

from (49) and (113), we can implement the QCR by the following.

- **Relation between the type 2 QFT and quaternion correlation:**

$$\begin{aligned} g(x,y) =& \text{IQFT}^{(2)}\Bigg[F_{a(q2)}(w,v)\left(\overline{Q^{r,i}\left(H_{(q2)}(w,v)\right)} - Q^{j,k}\left(H_{(q2)}(-w,-v)\right)\right) \\ &+ F_{b(q2)}(w,v)\,j\left(\overline{Q^{r,i}\left(H_{(q2)}(-w,-v)\right)} - Q^{j,k}\left(H_{(q2)}(w,v)\right)\right)\Bigg] \end{aligned} \tag{117}$$

where $F_{a(q2)}(w,v)$ and $F_{b(q2)}(w,v)$ are the type 2 QFT of $f_a(x,y)$ and $f_b(x,y)$ [which are defined as (47)].

When we use the type 1 QFT, then from (64), we can implement the QCR as follows.

- **Relation between the type 1 QFT and quaternion correlation:**

$$\begin{aligned} g(x,y) =& \text{IQFT}^{(1)}\Big[F_{a(q1)}(w,v)H_s(w,v) \\ &+ F_{a(q1)}(w,-v)H_t(w,v) \\ &+ F_{b(q1)}(w,-v) \cdot jH_s(-w,v) \\ &+ F_{b(q1)}(w,v) \cdot jH_t(-w,v)\Big] \end{aligned} \tag{118}$$

where $F_{a(q1)}(w,v)$ and $F_{b(q1)}(w,v)$ are QFTs of type 1 of $f_a(x,y)$ and $f_b(x,y)$ [which are defined as (47)] and

$$\begin{aligned} H_s(w,v) =& Q^{r,j}\left(QFT^{(1)}\left(\overline{h(-x,-y)}\right)\right) \\ H_t(w,v) =& Q^{i,k}\left(QFT^{(1)}\left(\overline{h(-x,-y)}\right)\right). \end{aligned} \tag{119}$$

When we use the QFT of types 1–3 to implement the quaternion correlation, we all require *six complex 2-D FTs*, but it seems the relation between the type 3 QFT and quaternion correlation is simpler and easier to analyze. Therefore, we suggest that it would be better to use the *type 3 QFT* to analyze the quaternion correlation. Besides, we can also prove that if $f(x,y) = h(x,y)$, i.e., the case of autocorrelation, then (115) can be simplified as follows.

- **Relation between the type 3 QFT and quaternion autocorrelation:**

$$\begin{aligned} g(x,y) = \text{IQFT}^{(3)}\Big[&\left|H_{a(q3)}(w,v)\right|^2 + \left|H_{b(q3)}(w,v)\right|^2 \\ &+ 2j\text{Odd}\left(\overline{H_{a(q3)}(w,v)}H_{b(q3)}(w,v)\right)\Big] \end{aligned} \tag{120}$$

where Odd() means extracting the odd part. That is, $\text{Odd}(f(x,y)) = [f(x,y) + f(-x,-y)]/2$.

VII. Conclusion

In this paper, we have developed efficient algorithms for the quaternion Fourier transform (QFT), the QCV, and the quaternion correlation (QCR). We have shown that for each type of QFT, we require two complex 2-D Fourier transforms (FTs) to

TABLE I
SUMMARY OF THE RELATIONS BETWEEN QFT AND QUATERNION CONVOLUTION AND CORRELATION

Relation between 1-side QCV and QFT-1	$G_{(q1)}(w,v)=F_{a(q1)}(w,v)Q^{r,j}\left(H_{(q1)}(w,v)\right)+F_{a(q1)}(w,-v)Q^{i,k}\left(H_{(q1)}(w,v)\right)+F_{b(q1)}(w,-v)jQ^{r,j}\left(H_{(q1)}(-w,v)\right)+F_{b(q1)}(w,v)jQ^{i,k}\left(H_{(q1)}(-w,v)\right)$, where $f_a(x,y)=f_r(x,y)+f_i(x,y)i$, $f_b(x,y)=f_j(x,y)+f_k(x,y)i$.
Relation between 1-side QCV and QFT-2	$G_{(q2)}(w,v)=F_{a(q2)}(w,v)H_{(q2)}(w,v)+F_{b(q2)}(w,v)\cdot jH_{(q2)}(-w,-v)$, where $f_a(x,y)=f_r(x,y)+f_i(x,y)i$, $f_b(x,y)=f_j(x,y)+f_k(x,y)i$.
Relation between 1-side QCV and QFT-3	$G_{(q3)}(w,v)=F_{(q3)}(w,v)H_{a(q3)}(w,v)+F_{(q3)}(-w,-v)\cdot jH_{d(q3)}(w,v)$, where $h_a(x,y)=h_r(x,y)+h_i(x,y)i$, $h_d(x,y)=h_j(x,y)-h_k(x,y)i$.
Relation between 2-side QCV and QFT-2	$G_{(q2)}(w,v)=F_{a(q2)}(w,v)H_{3(q2)}(w,v)+\overline{F_{a(q2)}(-w,-v)}jH_{4(q2)}(w,v)\ F_{b(q2)}(w,v)jH_{5(q2)}(-w,-v)+\overline{F_{b(q2)}(-w,-v)}jH_{6(q2)}(-w,-v)$, where $f_a(x,y)=f_r(x,y)+f_i(x,y)i$, $f_b(x,y)=f_j(x,y)+f_k(x,y)i$, $h_3(x,y)$, $h_4(x,y)$, $h_5(x,y)$, $h_6(x,y)$, are defined as (68)
Relation between QCR and QFT-3	$G_{(q3)}(w,v)=F_{(q3)}(w,v)\overline{H_{a(q3)}(w,v)}-F_{(q3)}(-w,-v)jH_{b(q3)}(-w,-v)$, where $h_a(x,y)=h_r(x,y)+h_i(x,y)i$, $h_b(x,y)=h_j(x,y)-h_k(x,y)i$.
Relation between QCR and QFT-1, 2	See (117), (118).

implement it. We require six complex 2-D FTs to implement a one-side QCV, 12 complex 2-D FTs to implement a two-side QCV, and six complex 2-D FTs to implement a QCR. With these efficient algorithms, the implementation of QFT, QCV, and QCR are much simplified. The improvement of efficiency is useful to extend the utilities of these quaternion operations.

We also discuss how to use QFT for quaternion linear time-invariant (QLTI) system analysis and filter design. Because some simpler relations between QFT and one-side QCV have been derived, it is easy to use QFT to analyze the QLTI system. When the impulse responses of the QLTI systems are even functions, then using the QFT to analyze the combination of the QLTI systems is the same as the conventional case. We also define a new type of QCV, i.e., spectrum-product QCV. For the spectrum-product QCV, regardless of what forms there are of the input functions, the QCV in the space domain always corresponds to the product operation in the frequency domain. Hence, they are very useful for the quaternion filter design.

APPENDIX

A. Proof of (49) (Relation Between Type 2 QFT and One-Side QCV)

See the equation at the bottom of the page. We have used the fact that $je^{i(wx_2+vy_2)} = e^{-i(wx_2+vy_3)}j$. Then, since

$$(4\pi^2)^{-1}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}e^{i(wx+vy)}e^{-i(wx_1+vy_1)}e^{-i(wx_2+vy_2)}dwdv = \delta(x-x_1-x_2)\,\delta(y-y_1-y_2) \tag{121}$$

$$\begin{aligned}
&\text{IQFT}^{(2)}\left(F_{b(q2)}(w,v)\cdot jH_{(q2)}(-w,-v)\right)\\
&=\frac{1}{4\pi^2}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}e^{i(wx+vy)}e^{-i(wx_1+vy_1)}f_b(x_1,y_1)\\
&\qquad\cdot je^{i(wx_2+vy_2)}h(x_2,y_2)\,dx_1dy_1dx_2dy_2dwdv\\
&=\frac{1}{4\pi^2}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\left[\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}e^{i(wx+vy)}e^{-i(wx_1+vy_1)}e^{-i(wx_2+vy_2)}dwdv\right]\\
&\qquad f_b(x_1,y_1)\,j\,h(x_2,y_2)\,dx_1dy_1dx_2dy_2
\end{aligned}$$

we obtain

$$\begin{aligned}&\text{IQFT}^{(2)}\left(F_{b(q2)}(w,v)\; jH_{(q2)}(-w,-v)\right)\\&=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f_b(x-x_2,y-y_2)\; jh(x_2,y_2)\,dx_2dy_2. \quad (122)\end{aligned}$$

From a similar process, we can prove

$$\begin{aligned}&\text{IQFT}^{(2)}\left(F_{a(q2)}(w,v)\; H_{(q2)}(w,v)\right)\\&=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f_a(x-x_2,y-y_2)\cdot h(x_2,y_2)\,dx_2dy_2. \quad (123)\end{aligned}$$

Therefore

$$\begin{aligned}\text{IQFT}^{(2)}&\left(F_{a(q2)}(w,v)\,H_{(q2)}(w,v)\right.\\&\left.\quad+F_{b(q2)}(w,v)\cdot jH_{(q2)}(-w,-v)\right)\\&=f_a(x,y)*_q h(x,y)+f_b(x,y)\cdot j*_q h(x,y)\\&=f(x,y)*_q h(x,y)\\&=g(x,y).\end{aligned}$$

□

B. Proof of (78) (Series Combination of QLTI Systems Represented by Type 2 QFT)

We use $g_p(x,y)$ to denote the cascade output of $h_p(x,y)$ in Fig. 1(b). Then, since $g_{S-1}(x,y)$ is the input of $h_S(x,y)$ and $g(x,y) = g_S(x,y)$ is the output of $h_S(x,y)$, from (56), $G_{(q2)}(w,v) = G_{S(q2)}(w,v)$ can be expressed as

$$\begin{aligned}G_{(q2)}(w,v)&=G_{S-1,(q2)}(w,v)H_{S(q2)e}(w,v)\\&\quad+\overline{G_{S-1,(q2)}(w,v)}^{j,k}H_{S(q2)o}(w,v)\\&=\left[G_{S-1,(q2)}(w,v)\overline{G_{S-1,(q2)}(w,v)^{j,k}}\right]\\&\quad\cdot\begin{bmatrix}H_{S(q2)e}(w,v)\\H_{S(q2)o}(w,v)\end{bmatrix}. \quad (124)\end{aligned}$$

Similarly, since $g_{p-1}(x,y)$ is the input of $h_p(x,y)$ and $g_p(x,y)$ is the output of $h_p(x,y)$, $G_{p(q2)}(w,v)$ can be expressed as

$$\begin{aligned}G_{p(q2)}(w,v)&=G_{p-1,(q2)}(w,v)H_{p(q2)e}(w,v)\\&\quad+\overline{G_{p-1,(q2)}(w,v)}^{j,k}H_{p(q2)o}(w,v). \quad \text{for } p\geq 2. \quad (125)\end{aligned}$$

Then, from (125) and the fact that $\overline{f(x,y)h(x,y)}^{j,k} = \overline{f(x,y)}^{j,k}\cdot\overline{h(x,y)}^{j,k}$, we obtain

$$\begin{aligned}\overline{G_{p(q2)}(w,v)}^{j,k}&=\overline{G_{p-1,(q2)}(w,v)}^{j,k}\overline{H_{p(q2)e}(w,v)}^{j,k}\\&\quad+G_{p-1,(q2)}(w,v)\,\overline{H_{p(q2)o}(w,v)}^{j,k} \quad \text{for } p\geq 2. \quad (126)\end{aligned}$$

Thus, from (125) and (126), we obtain

$$\begin{aligned}&\left[G_{p(q2)}(w,v)\,\overline{G_{p(q2)}(w,v)}^{j,k}\right]\\&=\left[G_{p-1,(q2)}(w,v)\,\overline{G_{p-1,(q2)}(w,v)}^{j,k}\right]\\&\quad\times\begin{bmatrix}H_{p(q2)e}(w,v)&\overline{H_{p(q2)o}(w,v)}^{j,k}\\H_{p(q2)o}(w,v)&\overline{H_{p(q2)e}(w,v)}^{j,k}\end{bmatrix} \quad (127)\end{aligned}$$

for $p\geq 2$, and when $p=1$

$$\begin{aligned}&\left[G_{1(q2)}(w,v)\,\overline{G_{1(q2)}(w,v)}^{j,k}\right]\\&=\left[F_{(q2)}(w,v)\,\overline{F_{(q2)}(w,v)}^{j,k}\right]\\&\quad\times\begin{bmatrix}H_{1(q2)e}(w,v)&\overline{H_{1(q2)o}(w,v)}^{j,k}\\H_{1(q2)o}(w,v)&\overline{H_{1(q2)e}(w,v)}^{j,k}\end{bmatrix}. \quad (128)\end{aligned}$$

Substituting (127) into (124) iteratively and together with (128), we can obtain (78). □

C. Proof of (92) (Spectrum-Product QCV Suitable for the Type 2 QFT)

From (91), $g(x,y)$ can be expressed as the equation at the bottom of the page. We can separate $f(x,y)$ as

---

$$\begin{aligned}g(x,y)&=\text{IQFT}^{(2)}\left(F_{(q2)}(w,v)\,H_{(q2)}(w,v)\right)\\&=\frac{1}{4\pi^2}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}e^{i(wx+vy)}e^{-i(wx_1+vy_1)}\\&\quad\times f(x_1,y_1)\,e^{-i(wx_2+vy_2)}h(x_2,y_2)\;dx_1dy_1dx_2dy_2dwdv.\end{aligned}$$

$$\begin{aligned}g(x,y)=&\frac{1}{4\pi^2}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}e^{i(wx+vy)}e^{-i(wx_1+vy_1)}e^{-i(wx_2+vy_2)}\\&\times f_a(x_1,y_1)\,h(x_2,y_2)\,dx_1dy_1dx_2dy_2dwdv\\&+\frac{1}{4\pi^2}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}e^{i(wx+vy)}e^{-i(wx_1+vy_1)}e^{i(wx_2+vy_2)}\\&\times f_b(x_1,y_1)\,jh(x_2,y_2)\,dx_1dy_1dx_2dy_2dwdv\\=&\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\delta(x-x_1-x_2,y-y_1-y_2)\\&\times f_a(x_1,y_1)\,h(x_2,y_2)\cdot dx_1dy_1dx_2dy_2\\&+\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\delta(x-x_1+x_2,y-y_1+y_2)\\&\times f_b(x_1,y_1)\,jh(x_2,y_2)\cdot dx_1dy_1dx_2dy_2\end{aligned}$$

$f_a(x,y) + f_b(x,y) \cdot j$, where $f_a(x,y)$, $f_b(x,y)$ are defined as (93). Therefore, we have the second equation at the bottom of the previous page. Therefore

$$\begin{aligned} g(x,y) = &\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f_a(x_1,y_1) \\ &\quad \cdot h(x-x_1, y-y_1)\, dx_1 dy_1 \\ &+ \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f_b(x_1,y_1) \\ &\quad \cdot jh(x_1-x, y_1-y)\, dx_1 dy_1 \\ = &\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f_a(x-\tau, y-\eta)\, h(\tau,\eta)\, d\tau d\eta \\ &+ \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f_b(x+\tau, y+\eta) \\ &\quad \cdot jh(\tau,\eta) d\tau d\eta. \end{aligned}$$ □

D. Summary of the Relations Between QFT, QCV, and QCR

We use Table I to show the relations between QFT and QCV, which is correlation derived in this paper. We use QFT-1, 2, and 3 to denote the QFT of types 1–3.

REFERENCES

[1] W. R. Hamilton, *Elements of Quaternions*. London, U.K.: Longman, 1866.

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

[3] 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.

[4] T. Bülow and G. Sommer, "Multi-dimensional signal processing using an algebraically extended signal representation," in *Lecture Notes in Computer Science*, G. Sommer and J. J. Koenderink, Eds. New York: Springer-Verlag, 1997, vol. 1315, pp. 148–163.

[5] S. J. Sangwine and T. A. Ell, "The discrete fourier transform of a color image," in *Image Processing II Mathematical Methods, Algorithms and Applications*, J. M. Blackledge and M. J. Turner, Eds. Chichester, U.K., 2000, pp. 430–441.

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

[7] ——, "Color image edge detector based on QCV," *Electron. Lett.*, vol. 34, no. 10, pp. 969–971, May 1998.

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

[9] C. J. Evans, T. A. Ell, and S. J. Sangwine, "Color-sensitive edge detection using hypercomplex filters," in *Proc. EUSIPCO*, Tampere, Finland, Sept. 2000.

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

[11] ——, "A novel block truncation coding of color images using a quaternion-moment-preserving principle," *IEEE Trans. Commun.*, vol. 45, pp. 583–595, May 1997.

[12] T. A. Ell and S. J. Sangwine, "Decomposition of 2D hypercomplex fourier transforms into pairs of complex fourier transforms," in *Proc. EUSIPCO*, 2000, pp. 151–154.

[13] P. Duhamel, "Implementation of split-radix FFT algorithms for complex, real and real-symmetric data," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-34, pp. 285–295, Apr. 1986.

[14] T. A. Ell and S. J. Sangwine, "Hypercomplex Wiener-Khintchine theorem with applications to color image correlation," in *Proc. ICIP*, vol. 2, Sept. 2000, pp. 792–795.

[15] T. A. Ell, "Hypercomplex spectral transforms," Ph.D. dissertation, Univ. Minnesota, Minneapolis, 1992.

[16] V. M. Chernov, "Discrete orthogonal transforms with data separation in composition algebras," in *Proc. 9th Scand. Conf. Image Anal.*, June 1995.

[17] M. Felsberg. (1997) Fast-quaternionic-Fourier-transform. Christian-Albrechts Univ. Kiel, Cognitive Syst. Group, Inst. Comput. Sci. Applied Math., Kiel, Germany. [Online]www.informatik.uni-kiel.de~tbl/index.html

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

[19] T. Bülow, "Hypercomplex spectral signal representations for the processing and analysis of images," Ph.D. dissertation, Christian-Albrechts Univ. Kiel, Kiel, Germany, [Online] Available: www.cis.upenn.edu/~thomasbl, 1999.

Soo-Chang Pei (F'00) was born in Soo-Auo,Taiwan, R.O.C., in 1949. He received the B.S.E.E. degree from National Taiwan University (NTU), Taipei, in 1978 and the M.S.E.E. and Ph.D. degrees from the University of California, Santa Barbara, in 1972 and 1975, respectively.

He was an engineering officer in the Chinese Navy Shipyard from 1970 to 1971. From 1971 to 1975, he was a research assistant with the University of California, Santa Barbara. He was Professor and Chairman in the Electrical Engineering Department, Tatung Institute of Technology and NTU, from 1981 to 1983 and 1995 to 1998, respectively. Presently, he is the Professor of Electrical Engineering Department, NTU. His research interests includes digital signal processing, image processing, optical information processing, and laser holography.

Dr. Pei received National Sun Yet-Sen Academic Achievement Award in Engineering in 1984, the Distinguished Research Award from the National Science Council from 1990 to 1998, the Outstanding Electrical Engineering Professor Award from the Chinese Institute of Electrical Engineering in 1998, and the Academic Achievement Award in Engineering from the Ministry of Education in 1998. He has been 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.

Jian-Jiun Ding was born in 1973 in Pingdong, Taiwan, R.O.C. He received both the B.S. and M.S. degree in electrical engineering from National Taiwan University (NTU), Taipei, in 1995 and 1997, respectively. He is currently pursuing the Ph.D. degree under the supervision of Prof. S.-C. Pei in the Department of Electrical Engineering, NTU. His current research areas include the fractional Fourier transform, the linear canonical transform, other fraction transforms, Hilbert transforms, fast algorithms, orthogonal polynomials, integer transforms, quaternion Fourier transforms, pattern recognition, fractals, and filter design.

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, NTU.

His current research areas include image watermarking, image processing, pattern recognition, and the quaternion and quaternion Fourier transform.