computers

MDPI

Article

Commutative Quaternion Algebra with Quaternion Fourier Transform-Based Alpha-Rooting Color Image Enhancement [†]

Artyom M. Grigoryan * and Alexis A. Gomez

Electrical and Computer Engineering, The University of Texas at San Antonio, San Antonio, TX 78249, USA; alexis.gomez@utsa.edu

* Correspondence: artyom.grigoryan@utsa.edu

[†] This article is a revised and expanded version of authors' conference paper which was presented at *SPIE 13033 Conference, Defense + Commercial Sensing 2024*, 22 April 2024,National Harbor, Maryland20745, MD, USA.

Abstract: In this paper, we describe the associative and commutative algebra or the (2,2)-model of quaternions with application in color image enhancement. The method of alpha-rooting, which is based on the 2D quaternion discrete Fourier transform (QDFT) is considered. In the (2,2)-model, the aperiodic convolution of quaternion signals can be calculated by the product of their QDFTs. The concept of linear convolution is simple, that is, it is unique, and the reduction of this operation to the multiplication in the frequency domain makes this model very attractive for processing color images. Note that in the traditional quaternion algebra, which is not commutative, the convolution can be chosen in many different ways, and the number of possible QDFTs is infinite. And most importantly, the main property of the traditional Fourier transform that states that the aperiodic convolution is the product of the transform in the frequency domain is not valid. We describe the main property of the (2,2)-model of quaternions, the quaternion exponential functions and convolution. Three methods of alpha-rooting based on the 2D QDFT are presented, and illustrative examples on color image enhancement are given. The image enhancement measures to estimate the quality of the color images are described. Examples of the alpha-rooting enhancement on different color images are given and analyzed with the known histogram equalization and Retinex algorithms. Our experimental results show that the alpha-rooting method in the quaternion space is one of the most effective methods of color image enhancement. Quaternions allow all colors in each pixel to be processed as a whole, rather than individually as is done in traditional processing methods.

Keywords: color image enhancement; quaternion convolution; quaternion Fourier transform; alpha-rooting; quaternion pyramids

Academic Editors: M. Ali Akber Dewan and Selene Tomassini

Received: 28 December 2024
Revised: 22 January 2025
Accepted: 24 January 2025
Published: 26 January 2025

Citation: Grigoryan, A.M.; Gomez, A.A. Commutative Quaternion Algebra with Quaternion Fourier Transform-Based Alpha-Rooting Color Image Enhancement. *Computers* **2025**, *14*, 37. https://doi.org/10.3390/computers14020037

1. Introduction

In recent years, many articles have been published on color image processing, wherein image enhancement plays an important role. Many color images are low quality and require enhancement as the first stage of processing. [1–5]. Examples of such images can be found among underwater images, thermal images, and medical images. Decades ago, we divided methods of image enhancement into two classes, namely spatial and traditional, or complex, frequency domains; now, a new class has been added to them. Here, we mention methods of image enhancement in the quaternion algebras. Color and grayscale images can be processed in the quaternion space with good results not only in image enhancement but in filtration, face recognition, neural networks, and other applications. The first class of methods includes contrast stretching techniques and logarithmic models [4] and the very effective and simple histogram equalization with its different modifications [6–10]. The Retinex algorithm can also be classified into this class [11,12]. In the second class, we should note the Fourier transform-based alpha-rooting [13], which is the most effective method for enhancing grayscale and color images. The advantage of enhancing color images in the quaternion space is in the fact that the primary colors plus the gray are processed as one unit, not separately. Therefore, quaternion image processing does not introduce false color artifacts [14].

In this paper, we focus on the commutative quaternion algebra, or the (2,2)-model. In this model, the concepts of the 1D and 2D QDFT are considered, and their properties are described. This model of quaternion uses the color image enhancement alpha-rooting by the 2D QDFT. A comparison with the traditional quaternion algebra is also given.

The main contributions of this work are the following:

- The separable alpha-rooting method of color image enhancement;
- New two-parameter alpha-rooting methods of color image enhancement;
- The effectiveness of using the 2D QDFT-based alpha-rooting in the (2,2)-model;
- Illustrative examples showing the effectiveness of using the (2,2)-model in color image enhancement.

The rest of the paper is organized in the following way. Section 2 describes two models of quaternions, namely the (2,2)- and (1,3)-models. The first model is commutative, and the second one is not. Section 3 describes the exponential functions of the (2,2)-model. The concepts of the QDFTs are considered in Section 4 for both models. The methods of alpha-rooting in these models are described in Section 5. The comparison of the 2D QDFT-based alpha-rooting methods in the (2,2)- and (1,3)-models are given. Results and illustrative examples of color images are presented in Section 6.

2. Quaternion Numbers: Two Arithmetics

In this section, we describe quaternion numbers in two algebras, non-commutative and commutative. The concept of the quaternion, or quadruple of numbers (a, b, c, d), as a vector in the 4-dimensional (4D) space was introduced by Gauss in 1819 [15]. As complex numbers, quaternions have one real part and one imaginary part. Only the imaginary part presents a triplet of numbers or a 3D vector. Therefore, quaternions can be considered as an extension of complex numbers [16–18]. It is not possible for us to draw quaternions in 4D space, but we will show how such numbers can be embedded in geometric figures in 3D space. There are different types of arithmetic of quadruples of numbers, or quaternions, because they define the main operation—multiplication—differently. We consider two arithmetics, or models, in which the operation of multiplication is commutative and non-commutative. The second arithmetic attracted much attention from researchers in the field of signal and image processing. However, the fact that the multiplication of quaternions is a non-commutative operation leads to large uncertainties in such important operations as the convolution, correlation, and Fourier transform, especially in processing color images. Therefore, we think it is necessary to pay more attention to the commutative operation of the multiplication of quaternions and the corresponding arithmetic, or the commutative algebra of quaternions.

2.1. The (1,3)-Model of Quaternions

Consider three units $i, j,$ and k with the following multiplication laws:

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

A quaternion is defined as the number $q = a + bi + cj + dk$ with real numbers $a, b, c,$ and d. The number $q' = bi + cj + dk$ is the imaginary part q' of the quaternion and can be considered as the vector (a, b, c) in the 3D space. Therefore, we can write $q = a + q' = a + (bi + cj + dk)$. This model of representation of quaternions as $q = (a, q')$ is called the (1,3)-model [14]. A quaternion has one real part, a, and the three-component imaginary part, q'. If the imaginary part $a = 0$, then the quaternion is called a pure quaternion number. If $c = d = 0$, the quaternion $q = a + bi$ is a complex number. Similar to the complex numbers, the conjugate of the quaternion q is defined as $\bar{q} = (a, -q')$, or $\bar{q} = a - q' = a - bi - cj - dk$.

The multiplication of two quaternions $q_1 = a_1 + q_1' = a_1 + (ib_1 + jc_1 + kd_1)$ and $q_2 = a_2 + q_2' = a_2 + (ib_2 + jc_2 + kd_2)$ is defined according to the laws in Equation (1). Thus, the quaternion $q = q_1q_2 = (a + q')$ is calculated by

$$a = a_1a_2 - [b_1b_2 + c_1c_2 + d_1d_2], and\, q' = [a_1q_2' + a_2q_1'] + \begin{vmatrix} i & j & k \\ b_1 & c_1 & d_1 \\ b_2 & c_2 & d_2 \end{vmatrix} \tag{2}$$

It is important to note, that the number $q\bar{q}$ is real and non-negative, $q\bar{q} = a^2 + (b^2 + c^2 + d^2)$; it is denoted by $|q_1|^2$. The number $|q_1|$ is called the length of the quaternion.

The sum of quaternions is calculated component-wise, $q_1 + q_2 = (a_1 + a_2) + (q_1' + q_2')$. In the multiplication of imaginary units, $ij \neq ji,\ jk \neq kj,$ and $ik \neq ki$. The multiplication in the (1,3)-model is not commutative. That is, for different q_1 and q_2, the product $q_1q_2 \neq q_2q_1$ or $q_1q_2 = q_2q_1$.

Considering the quaternions q_1 and q_2 as the 4D vectors, $\boldsymbol{q}_1 = (a_1, b_1, c_1, d_1)'$ and $\boldsymbol{q}_2 = (a_2, b_2, c_2, d_2)'$, the above operation of multiplication can be written in the matrix form as follows:

$$\boldsymbol{q} = \boldsymbol{A}_1 \begin{bmatrix} a_2 \\ b_2 \\ c_2 \\ d_2 \end{bmatrix} = \begin{bmatrix} a_1 - b_1 - c_1 - d_1 \\ b_1\ a_1 - d_1\ c_1 \\ c_1\ d_1\ a_1 - b_1 \\ d_1 - c_1\ b_1\ a_1 \end{bmatrix} \begin{bmatrix} a_2 \\ b_2 \\ c_2 \\ d_2 \end{bmatrix} \tag{3}$$

The determinant of the matrix equals $|q_1|^4 = (a_1^2 + b_1^2 + c_1^2 + d_1^2)^2$. For the case when $|q_1| = 1$, the matrix $\boldsymbol{A}_1$ is unitary and its determinant $\det \boldsymbol{A}_1 = 1$. The coefficients of this matrix are components of the quaternion q_1. The first column of the matrix is the quaternion q_1. A similar matrix of multiplication can be defined by the components of the quaternion q_2 (for details, see [14]).

Unlike traditional arithmetic, where the exponential function is defined uniquely, in the (1,3)-model, the number of such functions is infinite. Given a pure unit quaternion $\mu = im_1 + jm_2 + km_3$, $|\mu| = 1, \mu^2 = -1$, the quaternion exponential function at the angle x is defined as $e^{\mu x} = \cos x + \mu \sin x$. In the next sections, we will discuss the concept of the quaternion discrete Fourier transforms, which are different analogues of the traditional DFT. This transform is defined by the system of basis functions, which are calculated by the single complex exponential function e^{ix}. In the (1,3)-model, we are faced with the problem of which exponential function to use as the base function for the QDFT. In other words, if in the traditional representation each signal or image has the unique representation in the frequency domain, in the (1,3)-model, there are an infinite number of such representations. How to choose, namely which quaternion number μ is best for the QDFT, is unknown today. And it is this model that has been widely used in the last two decades in many applications in signal and image processing [14,19–21].

2.2. The (2,2)-Model of Quaternions

In this section, we consider the arithmetic of quaternions with the associative and commutative operation of multiplication introduced by Grigoryan in 2022 [22]. This is the so-called (2,2)-model of representation of quaternions.

In the (2,2)-model, the complex arithmetic is used in the following way. Given two complex numbers a_1 and a_2, the quaternion q is considered to be a pair of them and is written as

$$q=[a_1,a_2],a_1=(a_{1,1},a_{1,2}),a_2=(a_{2,1},a_{2,2}). \tag{4}$$

Here, the numbers $a_{1,1},a_{1,2},a_{2,1},$ and $a_{2,2}$ are real. We use the round brackets for 2D vectors a_1 and a_2, which represent the complex numbers $(a_{1,1}+ia_{1,2})$ and $(a_{2,1}+ia_{2,2})$, respectively. In this model, the quaternion is a pair of two complex numbers, or the pair of two 2-D vectors.

The quaternions include the complex and real numbers. Indeed, a quaternion $q=[a_1,0]$ is a complex number. If a complex number $a_1=(a_{1,1},0)$, that is, a_1 is real, then $q=[a_1,0]=\left[(a_{1,1},0),(0,0)\right]$ is a real number. We call the quaternion numbers $q=[0,a_2]$ the second complex numbers. Only complex numbers are used with the traditional unit i . The conjugate of the quaternion q is the quaternion $\bar{q}=[\bar{a}_1,\bar{a}_2]=[(a_{1,1},-a_{1,2}),(a_{2,1},-a_{2,2})]$. One can see that the conjugates of the unit quaternions are $\bar{e}_2=-e_2$, $\bar{e}_3=e_3$, and $\bar{e}_4=-e_4$. The second conjugate $\bar{\bar{q}}=q$.

The operation of sum of two quaternions $q_1=[a_1,a_2]$ and $q_2=[b_1,b_2]$ is defined component-wise. That is, the sum $q=q_1+q_2=[a_1+b_1,a_2+b_2]$. The multiplication of quaternions q_1 and q_2 is defined by

$$q=q_1q_2=[a_1,a_2][b_1,b_2]\triangleq[a_1b_1-a_2b_2,a_1b_2+a_2b_1]. \tag{5}$$

Here, the complex numbers are written as $a_1=\left(a_{1,1},a_{1,2}\right),a_2=\left(a_{2,1},a_{2,2}\right),b_1=(b_{1,1},b_{1,2})$, and $b_2=\left(b_{2,1},b_{2,2}\right)$. It should be noted that the similar operation over 4D elements was described by Clyde Davenport [23]; the multiplication was defined by using the complex conjugates as

$$q=q_1q_2\triangleq[a_1b_1-a_2\bar{b}_2,a_1b_2+a_2\bar{b}_1].$$

It directly follows from Equation (5) that if the quaternions are complex numbers, $q_1=[a_1,0]=a_1$ and $q_2=[b_1,0]=b_1$, then the multiplication $q=q_1q_2$ is the multiplication of complex numbers, that is,

$$q=q_1q_2=[a_1,0][b_1,0]=[a_1b_1,0]=a_1b_1.$$

The operation of multiplication in Equation (5) can also be written in the matrix form. For this, we consider the quaternions as 4D vectors $\boldsymbol{q}_1=\left(a_{1,1},a_{1,2},a_{2,1},a_{2,2}\right)'$ and $\boldsymbol{q}_2=\left(b_{1,1},b_{1,2},b_{2,1},b_{2,2}\right)'$. In the matrix form, the product $q=q_1q_2$ can be written as

$$q=\begin{bmatrix}q_{1,1}\\q_{1,2}\\q_{2,1}\\q_{2,2}\end{bmatrix}=\boldsymbol{M}_1\begin{bmatrix}b_{1,1}\\b_{1,2}\\b_{2,1}\\b_{2,2}\end{bmatrix}=\begin{bmatrix}a_{1,1}&-a_{1,2}&-a_{2,1}&a_{2,2}\\a_{1,2}&a_{1,1}&-a_{2,2}&-a_{2,1}\\a_{2,1}&-a_{2,2}&a_{1,1}&-a_{1,2}\\a_{2,2}&a_{2,1}&a_{1,2}&a_{1,1}\end{bmatrix}\begin{bmatrix}b_{1,1}\\b_{1,2}\\b_{2,1}\\b_{2,2}\end{bmatrix}. \tag{6}$$

As in the matrix $\boldsymbol{A}_1$ in the (1,3)-model, the first column of the matrix $\boldsymbol{M}_1$ is the quaternion q_1. This matrix has a block structure, that is,

$$\boldsymbol{M}_1=\begin{bmatrix}A&-B\\B&A\end{bmatrix},A=\begin{bmatrix}a_{1,1}&-a_{1,2}\\a_{1,2}&a_{1,1}\end{bmatrix},B=\begin{bmatrix}a_{2,1}&-a_{2,2}\\a_{2,2}&a_{2,1}\end{bmatrix}. \tag{7}$$

Here, the matrices A and B are matrices of multiplication of complex numbers a_1 and a_2, respectively. The matrix $\boldsymbol{A}_1$ also has the same block structure, but it is orthogonal, and the matrix $\boldsymbol{M}_1$ is not orthogonal.

To compare these two algebras visually, namely the operations of multiplication, we consider the following representation of quadruples of numbers in the 3D space. We call this representation the 4-in-3 representation. Any 4D vector can be represented in the form of four triplets, as follows:

$$q = (a, b, c, d) \rightarrow (a, b, c), (b, c, d), (c, d, a), (d, a, b).$$

The geometry of these four coordinates can be described by the quadrangular pyramid. It is clear that not every pyramid can have such a quaternion representation. Therefore, we will call such pyramids the quaternion pyramids *(QP)*. As example, Figure 1 shows the quaternion pyramid, $QP(q)$, for the quaternion $q = (1, -2, 8, 5)$ in part (a) and the pyramid $QP(\bar{q})$, for the conjugate quaternion $\bar{q} = (1, 2, -8, -5)$, in part (b), and the conjugate quaternion $\bar{q} = (1, 2, 8, -5)$ in the (2,2)-model in part (c). The first point (the vertex) of each pyramid is marked as an asterisk, '*'. The vertex of the pyramid should be considered, that is, the $QP(q)$ is the pyramid with the top point $v = (a, b, c)$. Therefore, we consider that $QP(q) = QP(q; v)$. Otherwise, we need to introduce concepts of equivalent pyramids. For instance, the figures of pyramids for four quaternion units, $1 = (1,0,0,0), i = (0,1,0,0), j = (0,0,1,0)$, and $k = (0,0,0,1)$, are the same. Such a vertex can also be considered the point (b, c, d), which corresponds to the imaginary part of the quaternion, $q' = (bi + cj + dk)$. Quaternion pyramids can be added, subtracted, multiplied, and divided, and the inverse pyramids exist. In other words, the set of all quaternion-pyramids is the space with the complete arithmetic as the quaternions.

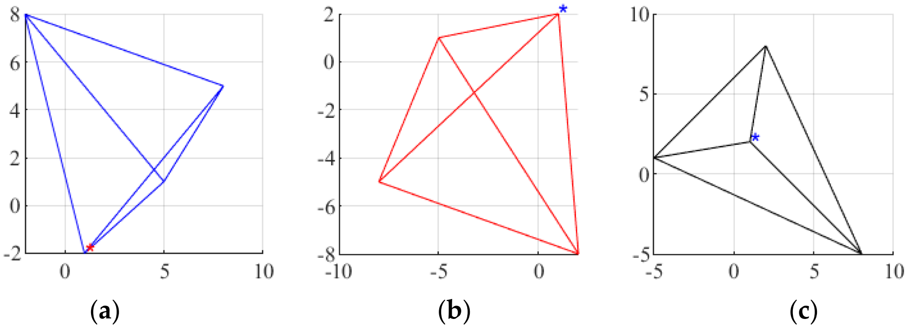


Figure 1. Quaternion-pyramids for (**a**) the quaternion q and its conjugates $\bar{q}$ in (**b**) the (1,3)-model and (**c**) the (2,2)-model.

Figure 2 shows the following four pyramids. Two quaternions are considered, $q_1 = (1,2,8,4)/\sqrt{85}$ and $q_2 = (-2,1,1,2)/\sqrt{10}$. The figure shows two pyramids $QP(q_1)$ and $QP(q_2)$ together with two pyramids for the quaternion multiplications $q = q_1 q_2$. The first pyramid $QP(q)$ is calculated in the non-commutative (1,3)-model, $g = q_2 (A_{q_1})' = (4, -23, -23, 4)/\sqrt{850}$, and another $QP(p)$ in the commutative (2,2)-model, $p = q_2 (M_{q_1})' = (-20, 9, -15, -12)/\sqrt{850}$.

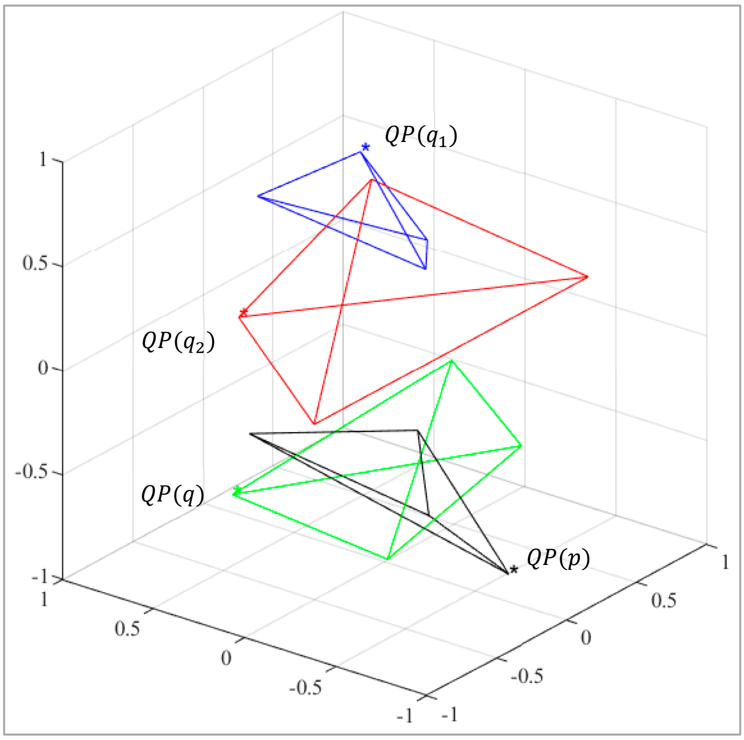


Figure 2. Four quaternion-pyramids.

The following properties hold for the multiplication.

1. The multiplication is commutative, $q_1q_2 = q_2q_1$.
2. The multiplication unit is the quaternion $e_1 = [(1,0),(0,0)] = (1,0) = 1$. For this real unit $e_1q = qe_1 = q$ for any quaternion q.
3. The multiplication rules of four quaternion units e_1, $e_2 = [(0,1),(0,0)], e_3 = [(0,0),(1,0)]$, and $e_4 = [(0,0),(0,1)]$ are given in Table 1. It should be noted that for two quaternion units e_2 and e_3, the square is $-e_1 = -1$. For the other two units e_1 and e_4, the square is $e_1 = 1$.

Table 1. Multiplication table, $T(e_1, e_2, e_3, e_4)$.

	e_1	e_2	e_3	e_4
e_1	e_1	e_2	e_3	e_4
e_2	e_2	$-e_1$	e_4	$-e_3$
e_3	e_3	e_4	$-e_1$	$-e_2$
e_4	e_4	$-e_3$	$-e_2$	e_1

4. The multiplication is associative, that is, $(q_1q_2)q_3 = q_1(q_2q_3)$, for any quaternions q_1, q_2, and q_3.
5. The multiplication is distributive, that is, $q_1(q_2 + q_3) = q_1q_2 + q_1q_3$.
6. The zero quaternion $q = 0$ has "divisors." For instance, the multiplication of two quaternions $q_1 = (1 + e_4)$ and $q_2 = (1 - e_4)$ is equal to $q_1q_2 = 1 - e_4^2 = 0$.
7. The inverse to the non-zero quaternion $q_1 = [a_1, a_2]$ is calculated by

$$q_1^{-1} = \left[\frac{a_1}{a_1^2 + a_2^2}, \frac{-a_2}{a_1^2 + a_2^2}\right] = \frac{1}{a_1^2 + a_2^2}[a_1, -a_2], \text{if } a_1^2 + a_2^2 \neq 0. \tag{8}$$

8. The inverse operation exists for all q, except the quaternions of the form $q = a_1(1 \pm e_4)$. For quaternion exponential numbers, the inverse exists. As mentioned in [24], the absence of some inverse numbers is not an obstacle when using quaternions to process signals and color images.

9. The division $q = q_2/q_1$ of quaternions $q_2 = [b_1, b_2]$ and $q_1 = [a_1, a_2]$ is calculated by $q \triangleq q_2 q_1^{-1}$.
10. The multiplication of a quaternion q on its conjugate $\bar{q}$ is equal to the following quaternion:

$$q\bar{q} = [a_1, a_2][\bar{a}_1, \bar{a}_2] = [|a_1|^2 - |a_2|^2, 2a_1 \cdot a_2] \tag{9}$$

11. In the general case, $q\bar{q}$ is not a real number and cannot be used to define the modulus of the quaternion in the traditional sense. For example, $e_4\bar{e}_4 = e_4(-e_4) = -e_4e_4 = -1$.
12. The length, or modulus, of the quaternion is defined as $|q| = \sqrt{E[q]}$, where the energy of the quaternion number q is calculated by

$$E[q] = E[a_1] + E[a_2] = |a_1|^2 + |a_2|^2 = \left(|a_{1,1}|^2 + |a_{1,2}|^2\right) + \left(|a_{2,1}|^2 + |a_{2,1}|^2\right). \tag{10}$$

Table 2 shows the main properties of quaternion numbers in the (1,3)- and (2,2)-models.

Table 2. Main operations and properties of quaternions in two quaternion models.

	The (2,2)-Model	The Traditional (1,3)-Model
Representation	$q_1 = [a_1, a_2] = [(a_{1,1}, a_{1,2}), (a_{2,1}, a_{2,2})]$	$q_1 = a_1 + q_1' = a_1 + (b_1 i + c_1 j + d_1 k)$
Multiplication q_1q_2	$q_1q_2 = [a_1b_1 - a_2b_2, a_1b_2 + a_2b_1]$	$[a_1q_2' + a_2q_1'] + a_1a_2 - q_1' \cdot q_2' + q_1' \times q_2'$
Multiplication rules	$\begin{array}{\|c\|c\|c\|c\|} \hline e_1 = 1 & e_2 & e_3 & e_4 \\ \hline e_2 & -1 & e_4 & -e_3 \\ \hline e_3 & e_4 & -1 & -e_2 \\ \hline e_4 & -e_3 & -e_2 & 1 \\ \hline \end{array}$	$\begin{array}{\|c\|c\|c\|c\|} \hline 1 & i & j & k \\ \hline i & -1 & k & -j \\ \hline j & -k & -1 & i \\ \hline k & j & -i & -1 \\ \hline \end{array}$
Multiplication matrix	$(a_1 = a_{11}, b_1 = a_{12}, c_1 = a_{21}, d_1 = a_{22})$ $\boldsymbol{M}{\square}_1 = \begin{bmatrix} a_1 & -b_1 & -c_1 & d_1 \\ b_1 & a_1 & -d_1 & -c_1 \\ c_1 & -d_1 & a_1 & -b_1 \\ d_1 & c_1 & b_1 & a_1 \end{bmatrix}$	$\boldsymbol{A}_1 = \begin{bmatrix} a_1 - b_1 - c_1 - d_1 \\ b_1\ a_1 - d_1\ c_1 \\ c_1\ d_1\ a_1 - b_1 \\ d_1 - c_1\ b_1\ a_1 \end{bmatrix}$
Orthogonality	Not	Yes
Commutativity	Yes: $q_1q_2 = q_2q_1$	Not: $q_1q_2 \neq q_2q_1$ or $q_1q_2 = q_2q_1$
Zero "divisors"	Yes: $(1 + e_4)(1 - e_4) = 0$	None: $q_1q_2 = 0 \rightarrow q_1 = 0$, or q_2=0.
Conjugate	$\bar{q}_1 = [(a_{1,1}, -a_{1,2}), (a_{2,1}, -a_{2,2})]$	$\bar{q}_1 = a_1 - b_1 i - c_1 j - d_1 k,$
Quaternion inverse	$q_1^{-1} = \frac{1}{a_1^2 + a_2^2}[a_1, -a_2], a_1^2 + a_2^2 \neq 0$	$q_1^{-1} = \frac{a_1 - b_1 i + c_1 j - d_1 k}{\lvert q_1{\square}\rvert^2}, q_1 \neq 0$
Division $q = \frac{q_1}{q_2}$	$q = q_2[a_1, -a_2]\frac{1}{a_1^2 + a_2^2}, a_1^2 + a_2^2 \neq 0.$	$q = \frac{\bar{q}_2 q_1}{\lvert q_2{\square}\rvert^2}$ (from left) $q = \frac{q_1 \bar{q}_2}{\lvert q_2{\square}\rvert^2}$ (from right)

3. The Quaternion Exponents in the (2,2)-Model

In this section, we describe the exponential functions in the (2,2)-model. For two pairs of quaternions $\mu = \pm e_3$ and $\pm e_2$, the square $\mu^2 = -1$. There are only two pairs of quaternions with the square equal to -1. For each of these quaternions, the exponential function is defined by the following series [22]:

$$\begin{aligned} e^{\mu\varphi} &= 1 + \mu\varphi + \frac{(\mu\varphi)^2}{2!} + \frac{(\mu\varphi)^3}{3!} + \frac{(\mu\varphi)^4}{4!} + \frac{(\mu\varphi)^5}{5!} + \cdots + \frac{(\mu\varphi)^n}{n!} + \cdots \\ &= \left[1 - \frac{\varphi^2}{2!} + \frac{\varphi^4}{4!} - \frac{\varphi^6}{6!} + \cdots\right] + \mu\left[\varphi - \frac{\varphi^3}{3!} + \frac{\varphi^5}{5!} - \frac{\varphi^7}{7!} + \cdots\right] = \cos\varphi + \mu\sin\varphi. \end{aligned} \tag{11}$$

Thus, there are four different exponential functions, or we can say two pair of quaternion exponential functions. The fundamental multiplicative property holds for these exponents, that is,

$$\exp(\mu[\varphi+\vartheta]) = \exp(\mu\varphi)\exp(\mu\vartheta) \tag{12}$$

Now, we consider these two pairs of quaternion exponents.

1. The first pair of exponents is defined for the conjugate quaternions $\mu = \pm e_2 = [(0,\pm 1),(0,0)]$. The quaternion exponents are the following conjugate functions:

$$e^{\mu\varphi} = \cos\varphi \pm e_2 \sin\varphi = [(\cos\varphi, \pm\sin\varphi), 0] = (\cos\varphi, \pm\sin\varphi). \tag{13}$$

In the matrix form, the multiplication of a quaternion $q = [a_1, a_2]$ by the exponent $q_1 = e^{\mu\varphi}$ is described as follows:

$$qq_1 = qe^{\mu\varphi} = (\cos\varphi, \pm\sin\varphi)q = \begin{bmatrix} c & -s & 0 & 0 \\ s & c & 0 & 0 \\ 0 & 0 & c & -s \\ 0 & 0 & s & c \end{bmatrix} q = \begin{bmatrix} R_\varphi & \mathbf{0} \\ \mathbf{0} & R_\varphi \end{bmatrix} q. \tag{14}$$

Here, we denote $c = \cos\varphi$ and $s = \pm\sin\varphi$. With the operation of the Kronecker product of matrices, the above matrix of multiplication can be written as $A_{q_1} = I_2 \otimes R_\varphi$. The matrix R_φ is the matrix of elementary rotation by the angle $\pm\varphi$. Thus, the operation $qe^{\mu\varphi}$ is reduced to separate rotations of two components of the quaternion, a_1 and a_2, by the same angle.

2. The second pair of exponents is defined by the quaternion $\mu = \pm e_3 = [(0,0),(\pm 1,0)]$. The corresponding pair of quaternion exponential functions is

$$e^{\mu\varphi} = \exp(\mu\varphi) = \cos\varphi \pm e_3 \sin\varphi = [(\cos\varphi, 0), (\pm\sin\varphi, 0)]. \tag{15}$$

These two exponential functions are not conjugate but inverse to each other. The inverse of the exponent is $(e^{\mu\varphi})^{-1} = [(\cos\varphi, 0), (-\sin\varphi, 0)] = e^{-\mu\varphi}$. In the matrix form, the multiplication of the exponent $q_1 = e^{\mu\varphi}$ by a quaternion q can be written as follows:

$$qq_1 = q_1 q = e^{\mu\varphi} q = \begin{bmatrix} c & 0 & -s & 0 \\ 0 & c & 0 & s \\ s & 0 & c & 0 \\ 0 & -s & 0 & c \end{bmatrix} q. \tag{16}$$

The matrix of the multiplication is the tensor product of the rotation matrix and the identity matrix, $A_{q_1} = R_\varphi \otimes I_2$.

It should be noted that if we consider the symmetric matrix $P_{(1,2)}$ of the permutation (1,2), then the above two pairs of quaternion exponents can be derived from each as

$$[(\cos\varphi, 0), (\pm\sin\varphi, 0)] = [(\cos\varphi, \pm\sin\varphi), (0,0)] \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{17}$$

4. Quaternion Discrete Fourier Transforms

In this section, we consider the concept of the quaternion discrete Fourier transform (QDFT) in the (1,3)- and (2,2)-models. In the first model, the N-point QDFT of the quaternion signal $q = \{q_n; n = 0{:}(N-1)\}$ is defined by

$$Q_p = \sum_{n=0}^{N-1} q_n W_\mu^{np} = \sum_{n=0}^{N-1} q_n \left[\cos\left(\frac{2\pi}{N}np\right) - \mu \sin\left(\frac{2\pi}{N}np\right)\right], p = 0{:}(N-1). \tag{18}$$

Here, μ is a pure quaternion unit number, such that $\mu^2 = -1, |\mu| = 1$. As mentioned above, the number of such quaternions is infinite. For instance, this number can be taken as $\mu = i, j, k$, and $(i \pm j \pm k)/3$. The multiplication is not commutative; therefore, this

QDFT is the left-sided transform. The right-sided QDFT is defined as the sum of $W_\mu^{np}q_n$. The inverse N-point QDFT is calculated by

$$q_n = \frac{1}{N}\sum_{p=0}^{N-1} Q_p W_\mu^{-np} = \frac{1}{N}\sum_{p=0}^{N-1} Q_p \left[\cos\left(\frac{2\pi}{N}np\right) + \mu\sin\left(\frac{2\pi}{N}np\right)\right], n = 0{:}\,(N-1). \quad (19)$$

The fast algorithms to calculate the QDFT exist for both types of transform in the 1D and 2D cases. For 2D signals, the QDFT can be defined as the right-, left-, or both-sided transform [14,24]. These transforms do not have one of the basic properties of the traditional Fourier transform, namely, the cyclic convolution of signals is not reduced to the operation of multiplication in the frequency domain. In the 1D case, the cyclic convolution of two periodic quaternion signals $q_n = [f_n, g_n]$ and $h_n = [h_{1,n}, h_{2,n}]$ is defined as

$$y_n = q_n \circledast h_n = \sum_{k=0}^{N-1} q_{n-k}h_k\,, n = 0{:}\,(N-1). \quad (20)$$

Here, we need to consider that $q_n \circledast h_n \neq h_n \circledast q_n$, because the products $q_{n-k}h_k \neq h_k q_{n-k}$. Thus, in the (1,3)-model, two different linear convolutions can be used.

Now, we consider these concepts in the (2,2)-model with two pairs of quaternion exponential functions, namely $e^{\mu\varphi}$, when $\mu = \pm e_2$ and $\pm\, e_3$. Each pair of these functions is used for the direct and inverse QDFTs. Thus, in the (2,2)-model there are only two pairs of the direct and inverse QDFTs. The (2,2)-model is commutative; therefore, the transform of the N-point quaternion signal $[f_n, g_n]$ is defined as

$$Q_p = \sum_{n=0}^{N-1} q_n W_\mu^{np} = \sum_{n=0}^{N-1} W_\mu^{np} q_n\,, W_\mu = \exp\left(-\frac{\mu 2\pi}{N}\right) e^{-\mu\frac{2\pi}{N}}. \quad (21)$$

Two different N-point QDFTs are described in the following way.

1. When the quaternion μ is $e_2 = [(0,1),(0,0)]$ and the angle is $\varphi = 2\pi/N$, the basis exponential functions are

$$\psi_p(n) = W_{e_2}^{np} = \exp(-e_2\varphi np)\,[(\cos np\varphi\,, -\sin np\varphi), (0,0)]\;[e^{-inp\varphi}, 0], \quad (22)$$

$p, n = o = 0{:}\,(N-1)$. The N-point direct QDFT is defined as

$$Q_p = \sum_{n=0}^{N-1} q_n\psi_p(n) = \sum_{n=0}^{N-1} [f_n, g_n]\,[e^{-i\varphi np}, 0] = \sum_{n=0}^{N-1} [f_n e^{-i\varphi np}, g_n e^{-i\varphi np}].$$

or

$$Q_p = \left[\sum_{n=0}^{N-1} f_n e^{-i\varphi np}\,, \sum_{n=0}^{N-1} g_n e^{-i\varphi np}\right] = [F_p, G_p]. \quad (23)$$

Here, F_p and G_p are the traditional N-point DFTs of the complex signal f_n and g_n, respectively,

$$F_p = \sum_{n=0}^{N-1} f_n e^{-i\varphi np}\,, G_p = \sum_{n=0}^{N-1} g_n e^{-i\varphi np}\,, p = 0{:}\,(N-1).$$

This N-point QDFT is called the N-point e_2-QDFT and it requires two N-point DFTs [22]. The inverse N-point e_2-QDFT is calculated by

$$q_n = [f_n, g_n] = \frac{1}{N}\sum_{p=0}^{N-1} Q_p W_\mu^{-np} = \frac{1}{N}\sum_{p=0}^{N-1} [F_p, G_p]\,[e^{inp\varphi}, 0], n = 0{:}\,(N-1). \quad (24)$$

2. In the $\mu = e_3$ case, the basis exponential functions for the QDFT are

$$\psi_p(n) = W_{e_3}^{np} = \exp(-e_3\varphi np) = [(\cos(np\varphi)\,,0),(-\sin(np\varphi)\,,0)], p, n = 0\colon(N-1). \tag{25}$$

The N-point QDFT which is called the *N-point e_3-QDFT* is defined as [22]

$$Q_p = \sum_{n=0}^{N-1}[f_n, g_n]\,W_{e_3}^{np} = \sum_{n=0}^{N-1}[f_n\cos(\varphi np) + g_n\sin(\varphi np)\,, -f_n\sin(\varphi np) + g_n\cos(\varphi np)].$$

In the matrix form, this transform can be written with the rotation matrices as

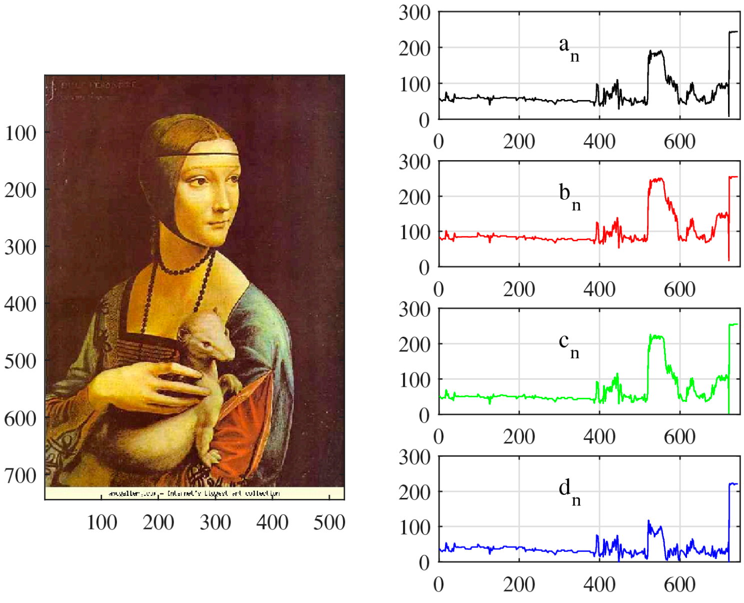
$$Q_p = \sum_{n=0}^{N-1}[f_n, g_n]R_{\varphi np} = \sum_{n=0}^{N-1}[f_n, g_n]\begin{bmatrix}\cos(\varphi np) & -\sin(\varphi np)\\ \sin(\varphi np) & \cos(\varphi np)\end{bmatrix}, p = 0\colon(N-1). \tag{26}$$

The inverse N-point e_3-QDFT $Q_p = [A_p, B_p]$ is calculated by

$$q_n = [f_n, g_n] = \frac{1}{N}\sum_{p=0}^{N-1}Q_pW_\mu^{-np} = \frac{1}{N}\sum_{p=0}^{N-1}[A_p, B_p]R_{-\varphi np}, n = 0\colon(N-1). \tag{27}$$

Thus, in the (2,2)-model, we can work with only two N-point QDFT, namely, e_2-QDFT and e_3-QDFT.

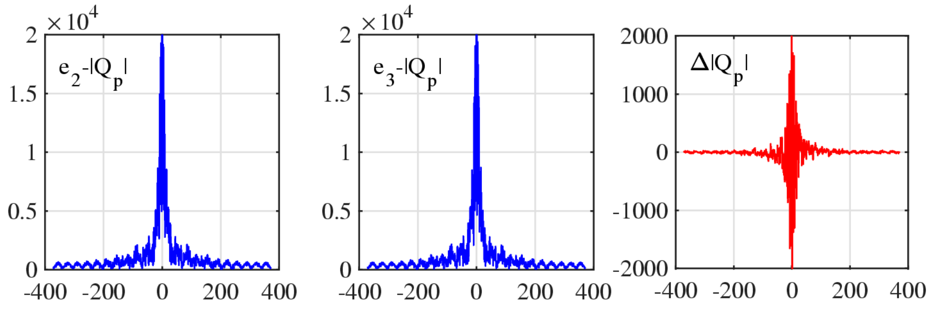
As an example, Figure 3 shows the color image ‘leonardo9.jpg’ of 744 × 526 pixels in part (a) and the quaternion signal composed from column number 101 in part (b). The signals b_n, c_n, and d_n are the red, green, and blue channels of the image column, respectively. The signal a_n is the average of these signals.



(a) (b)

Figure 3. (**a**) The color image and (**b**) the quaternion signal of length 744 composed from one image column.

The e_2-QDFT and e_3-QDFT of this quaternion signal are plotted in absolute scale, $|Q_p|, p = 0\colon 733$, in Figure 4 in parts (a) and (b), respectively. The difference between these two plots is shown in part (c).



(**a**) (**b**) (**c**)

Figure 4. The magnitude of (**a**) the e_2-QDFT, (**b**) the e_3-QDFT, and (**c**) the difference of these transforms.

As shown in [22], in the (2,2)-model, the aperiodic convolution of quaternion signals can be calculated by multiplying the QDFTs. This statement is valid for both types of QDFT. The convolution of a periodic quaternion signal $q_n = [f_n, g_n]$ with another one $h_n = [h_{1,n}, h_{2,n}]$ is unique,

$$y_n = q_n \circledast h_n = \sum_{k=0}^{N-1} q_{n-k} h_k = \sum_{k=0}^{N-1} q_k h_{n-k}\,, n = 0{:}\,(N-1). \tag{28}$$

Here, the subscripts $n-k$ are considered by modulo N. This convolution is calculated by four complex convolutions as follows:

$$y_n = [y_{1,n}, y_{2,n}], y_{1,n} = f_n \circledast h_{1,n} - g_n \circledast h_{2,n}\; y_{2,n} = f_n \circledast h_{2,n} + g_n \circledast h_{1,n}. \tag{29}$$

For $k = 2$ and 3, the N-point e_k-QDFT of the convolution y_n is calculated by $Y_p = Q_p H_p,\ p = 0{:}\,(N-1)$. Here, Q_p and H_p are components of the corresponding N-point e_k-QDFT of signals q_n and h_n, respectively. What type of QDFT is used for computing the aperiodic convolution is irrelevant. We think that the calculation of the quaternion convolution by the e_2-QDFT is simple. According to the multiplication, the e_2-QDFT of the aperiodic convolution is calculated by

$$Y_p = Q_p H_p = [F_p, G_p][H_{1,p}, H_{2,p}] = [F_p H_{1,p} - G_p H_{2,p}, F_p H_{2,p} + G_p H_{1,p}]. \tag{30}$$

Therefore, the task of calculating the quaternion aperiodic convolution in the frequency domain is solved in the (2,2)-model. In the traditional (1,3)-model of quaternions, this problem does not have such a simple solution—it is unsolvable. Table 3 summarizes the above considerations.

Table 3. Properties of aperiodic convolution and QDFT.

	The (2,2)-Model	**The (1,3)-Model**
Aperiodic convolution	$q = q_1 \circledast q_2 = q_2 \circledast q_1$	$q = q_1 \circledast q_2 \neq q_2 \circledast q_1$
Exponential functions	Only two pairs	Infinite number
The pair of the QDFT	Only two	Infinite number
Convolution property	$Q_p(q_1 \circledast q_2) = Q_p(q_1) \cdot Q_p(q_1)$	$Q_p(q_1 \circledast q_2) \neq Q_p(q_1) \cdot Q_p(q_1)$

5. Processing Images in the (2,2)-Model

In this section, we describe the concept of the 2D QDFT of images, which will be used in color image enhancement, namely, in the method which is called alpha-rooting. A color image in the RGB model will be presented by the quaternion image $q_{n,m} = [f_{n,m}, g_{n,m}]$ and then transformed to the frequency domain. Let $(r_{n,m}, g_{n,m}, b_{n,m})$ be components of the primary colors, red (R), green (G), and blue (B), in the image of $N \times M$ pixels. To compose the quaternion image $q_{n,m}$, we add the real component $a_{n,m}$. Thus, $q_{n,m} =$

$(a_{n,m}, r_{n,m}, g_{n,m}, b_{n,m})$. The real part of this image is usually considered zero, $a_{n,m} = 0$, or the gray-scale component $a_{n,m} = (r_{n,m} + g_{n,m} + b_{n,m})/3$ at each pixel (n, m). The brightness of the image can also be considered, $a_{n,m} = 0.3r_{n,m} + 0.59gr_{n,m} + 0.11b_{n,m}$. In the (2,2)-model, the quaternion image $q_{n,m} = [f_{n,m}, g_{n,m}]$ is the pair of 2D data $f_{n,m} = (a_{n,m}, r_{n,m})$ and $g_{n,m} = (g_{n,m}, b_{n,m})$. In many applications, processing color images in quaternion space is efficient, since at each pixel the color triplet (plus the gray) is treated as one number, quaternion. Note that in the traditional approach, each color component of the image is processed separately. And this causes many unwanted effects on colors in the processed images [5,14].

The two-dimension $N \times M$-point QDFT in the frequency-point (p, s) is calculated by

$$Q_{p,s} = \sum_{n=0}^{N-1}\sum_{m=0}^{M-1} q_{n,m} W_\mu^{np} W_\mu^{ms} = \sum_{n=0}^{N-1}\sum_{m=0}^{M-1} q_{n,m} W_\mu^{np+ms}, \tag{31}$$

where $p, s = 0,1, \dots, (N-1), (M-1)$. In the (1,3)-model, two sums in this equation are different transforms; the first one is called the separable right-sided 2D QDFT [21,24].

We consider the 2D QDFT, which is calculated by the 1D e_2-QDFTs. This 2D transform is called the 2D $N \times M$-point e_2-QDFT—the case when $\mu = e_2$ [22,25]. As in the 1D case, the 2D e_2-QDFT has a simple form, when compared with the 2D e_3-QDFT. The 2D e_2-QDFT of the quaternion image $q_{n,m} = [f_{n,m}, g_{n,m}]$ is calculated by

$$Q_{p,s} = \sum_{n=0}^{N-1}\sum_{m=0}^{M-1} [f_{n,m}, g_{n,m}] W_N^{np} W_M^{ms} = [F_{p,s}, G_{p,s}]. \tag{32}$$

Here, $F_{p,s}$ and $G_{p,s}$ are the $N \times M$-point 2-D DFTs of the complex components $f_{n,m}$ and $g_{n,m}$, respectively,

$$F_{p,s} = \sum_{n=0}^{N-1}\sum_{m=0}^{M-1} f_{n,m} e^{-i\frac{2\pi}{N}np}\, e^{-i\frac{2\pi}{M}ms}, G_{p,s} = \sum_{n=0}^{N-1}\sum_{m=0}^{M-1} g_{n,m} e^{-i\frac{2\pi}{N}np}\, e^{-i\frac{2\pi}{M}ms}.$$

Thus, the calculation of the $N \times M$-point e_2-QDFT is reduced to two 2D DFTs. The inverse $N \times M$-point e_2-QDFT is calculated by

$$q_{n,m} = \mathcal{F}^{-1}[Q]_{n,m} = \frac{1}{NM}\sum_{p=0}^{N-1}\sum_{s=0}^{M-1} [F_{p,s}, G_{p,s}] W_N^{-np} W_M^{-ms}, n, m = 0{:}\,(N-1), (M-1).$$

5.1. Method of Alpha-Rooting by the 2D QDFT

The absolute value, or the module, of the quaternion $Q_{p,s} = [F_{p,s}, G_{p,s}]$ is defined as $|Q_{p,s}| = \sqrt{|F_{p,s}|^2 + |G_{p,s}|^2}$. In the alpha-rooting [26,27], the image is enhanced by changing its absolute value at each frequency point to $|Q_{p,s}| \to |Q_{p,s}|^\alpha$, where the parameter α is from the interval (0,1). Given value α, the 2D e_2-QDFT of the quaternion image $q_{n,m}$ is processed as follows:

$$q_{n,m} \to Q_{p,s} \to \underbrace{V_{p,s} = Q_{p,s}|Q_{p,s}|^{\alpha-1}} \to \underbrace{(q_\alpha)_{n,m} = \mathcal{F}^{-1}[V_{p,s}]_{\mathrm{n,m}}} \to A[q_\alpha]_{n,m}. \tag{33}$$

Here, $A > 1$ is a necessary constant, since the alpha-rooting method reduces the transforms in absolute scale.

The main steps of the algorithm:

1. Compose the quaternion image $q_{n,m}$ from the given RGB color image, $q_{n,m} = (a_{n,m}, r_{n,m}, g_{n,m}, b_{n,m})$.
2. Calculate the 2D e_2-QDFT of the quaternion image, $Q_{p,s} = \mathcal{F}[q]_{p,s} = [F_{p,s}, G_{p,s}]$.

3. Calculate the module of the transform, $|Q_{p,s}|$.
4. Process the transform modules by the alpha-rooting, $V_{p,s} = Q_{p,s}|Q_{p,s}|^{\alpha-1}$.

Thus, the 2D e_2-QDFT of the quaternion image changes by the non-negative coefficients $c(p,s) = |Q_{p,s}|^{\alpha-1}$,

$$Q_{p,s} = [F_{p,s}, G_{p,s}] \rightarrow V_{p,s} = c(p,s)[F_{p,s}, G_{p,s}] = [c(p,s)F_{p,s}, c(p,s)G_{p,s}]. \tag{34}$$

5. Calculate the inverse 2D e_2-QDFT, $(q_\alpha)_{n,m} = \mathcal{F}^{-1}[V]_{n,m}$.
6. Multiply the image by the constant $A > 1$ to raise the range of the image.
   The output of the alpha-rooting is the quaternion image $(v_\alpha)_{n,m} = A(q_\alpha)_{n,m}$. Rounding to integers is required.
7. Compose the new color image, $(v_c)_{n,m}$, as the three-component imaginary part of the quaternion image $(v_\alpha)_{n,m}$.
8. Extract the new grayscale image from the quaternion image $(v_\alpha)_{n,m}$, as its real part. Note that this grayscale image is not the gray or brightness of the new color image $(v_c)_{n,m}$.

The new image $v_{n,m}$ is parameterized by α. Therefore, the question arises as to how to choose the value of this parameter to better enhance the color image. As our preliminary examples have shown, the choice of the best values of α for enhancing color and quaternion images can be based on the known measure of color image enhancement (EMEC) [5,13]. This measure is used before and after image processing. The EMEC is the generalization of the enhancement measure that was used for grayscale images.

A. Enhancement measures for grayscale images

To estimate the quality of grayscale images, we effectively developed and used the concept of the quantitative estimated measure of enhancement (EME). This measure was selected after analyzing the Weber and Fechner laws of the human visual system [28,29]. The measure is defined as the average of the range of image intensity in the logarithm scale when it is divided by blocks of the same size $L_1 \times L_2$, for example, 7×7. Only the full blocks are considered. Therefore, the number of blocks inside a discrete image $f = \{f_{n,m}\}$ of $N \times M$ pixels is calculated as k_1k_2, where $k_1 = \lfloor N/L_1 \rfloor, k_2 = \lfloor M/L_2 \rfloor$, and $\lfloor . \rfloor$ denotes the rounding floor function. The EME of the image is

$$EME(f) = \frac{1}{k_1k_2}\sum_{k=1}^{k_1}\sum_{l=1}^{k_2} 20 \ln\left[\frac{\max\limits_{k,l}(f)}{\min\limits_{k,l}(f)}\right] = \frac{1}{k_1k_2}\sum_{k=1}^{k_1}\sum_{l=1}^{k_2} 20\left[\ln \max_{k,l}(f) - \ln \min_{k,l}(f)\right]. \tag{35}$$

Here, inside the (k,l)th block, the maximum, $\max_{k,l}(f)$, and minimum, $\min_{k,l}(f)$, of the image $f_{n,m}$ are calculated. Thus, the EME of the image is estimated block-wise by using the logarithm range of the image. If all values of the image in a block are 0, this block can be removed from the measure calculation. To avoid such cases, $EME(f+1)$ can be calculated instead. The change $f \rightarrow (f+1)$ does not change the quality of the image unless it is binary.

Together with EME, other contrast measures also can be used, including [14]:

1. The estimated measure of enhancement entropy measure (EMEE)

$$EMEE(f) = \frac{1}{k_1k_2}\sum_{k=1}^{k_1}\sum_{l=1}^{k_2} \frac{\max\limits_{k,l}(f)}{\min\limits_{k,l}(f)} \ln\left[\frac{\max\limits_{k,l}(f)}{\min\limits_{k,l}(f)}\right]. \tag{36}$$

2. The Michelson enhancement measure (MEM)

$$MEM(f) = -\frac{1}{k_1k_2}\sum_{k=1}^{k_1}\sum_{l=1}^{k_2} [MVR_{k,l}(f)] \ln[MVR_{k,l}(f)], \tag{37}$$

where the Michelson visibility ratio is calculated by

$$MVR_{k,l}(f) = \frac{\left|\max\limits_{k,l}(f) - \min\limits_{k,l}(f)\right|}{\min\limits_{k,l}(f) + \min\limits_{k,l}(f)}.$$

3. The signal-noise ratio (or the ratio of the mean of the image and standard deviation)

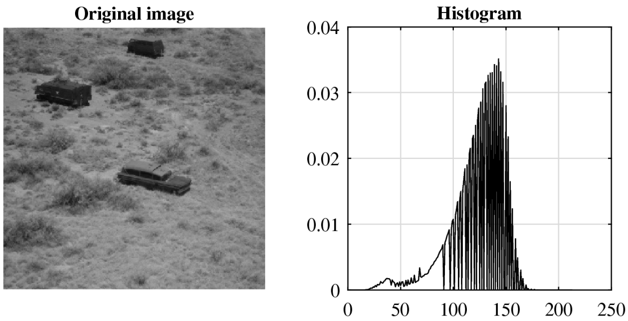
$$SNR(f) = \frac{E[f]}{\sqrt{E[f^2] - E^2[f]}} = \frac{1}{\sqrt{E[f^2]/E^2[f] - 1}}, \tag{38}$$

where

$$E[f] = \frac{1}{NM}\sum_{n=0}^{N-1}\sum_{m=0}^{M-1} f_{n,m} \text{ and } E[f^2] = \frac{1}{NM}\sum_{n=0}^{N-1}\sum_{m=0}^{M-1} f_{n,m}^2 .$$

Our experimental results show that the EME and EMEE measures can be effectively used in enhancing images. After processing the image, $f_{n,m} \to g_{n,m}$, the EME of the enhanced image is calculated and compared with the EME of the original image. The range of the alpha-rooting image is usually smaller than [0,255]. Therefore, the obtained image should be multiplied by a coefficient. The new image and its quality depend on the value of parameter α, i.e., $g = g_\alpha$ and the measure is a function of α, that is, $EME(g_a) = EME(\alpha)$. The parameters of interest for alpha-rooting are in the range $R\{\alpha \in (0,1);\ EME(\alpha) > EME(g_\alpha) > EME(f)\}$. The degree of enhancement is determined by the EME measure. The best or optimal values of the enhancement are considered to be the values α_0, for which $EME\left(g_{\alpha_0}\right) = \max\limits_{a\in R} EME(g_\alpha)$ or $\min\limits_{a\in R} EME(g_\alpha)$.

To illustrate the introduced above measures of image enhancement, we consider the image of 512×512 pixels shown in Figure 5 in part (a). The histogram of the image is given in part (b). The enhancement by the Fourier transform-based alpha-rooting was used when changing the parameter α in the interval [0,1] with a step of 0.01. The graph of the measure of this image, $EME = EME(\alpha)$, as the function of α is shown in part (c). Blocks of size 7×7 were used to calculate the EME. For the original image, the measure of enhancement equals 7.63. The maximum of the function $EME(\alpha)$ is at point $\alpha = 0.83$ and equals $EME(0.83) = 20.69$. The image enhanced by the 0.83-rooting is shown in part (d). It was multiplied by the coefficient 19 to scale the image. In parts (e) and (f), the graph of the measure $EMEE(\alpha)$ and the enhanced image by the 0.83-rooting (and multiplied by 17) are shown, respectively. This measure has the maximum 1071.26 at point $\alpha = 0.84$. The measure of the original image equals $EMEE(1) = 0.68$. The best parameters $\alpha = 0.83$ and 0.84 for these two measures are very close to each other, as well as the results of the enhancement, which are shown in parts (d) and (f). The $EME(\alpha)$ function is much smoother than the $EMEE(\alpha)$ measure, and its graph has a distinct peak. For other images, the optimal values of the parameter α may be very different, but the smoothness of the function $EME(\alpha)$ is preserved and easier to work with.



(**a**) (**b**)

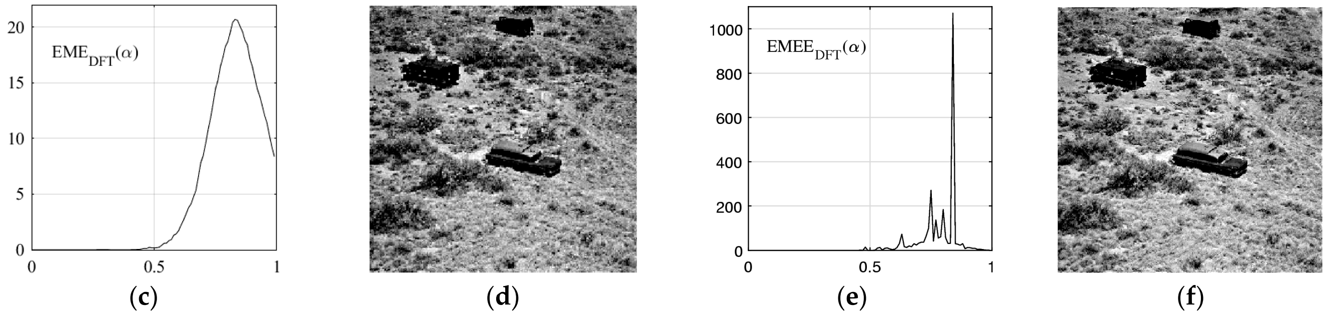
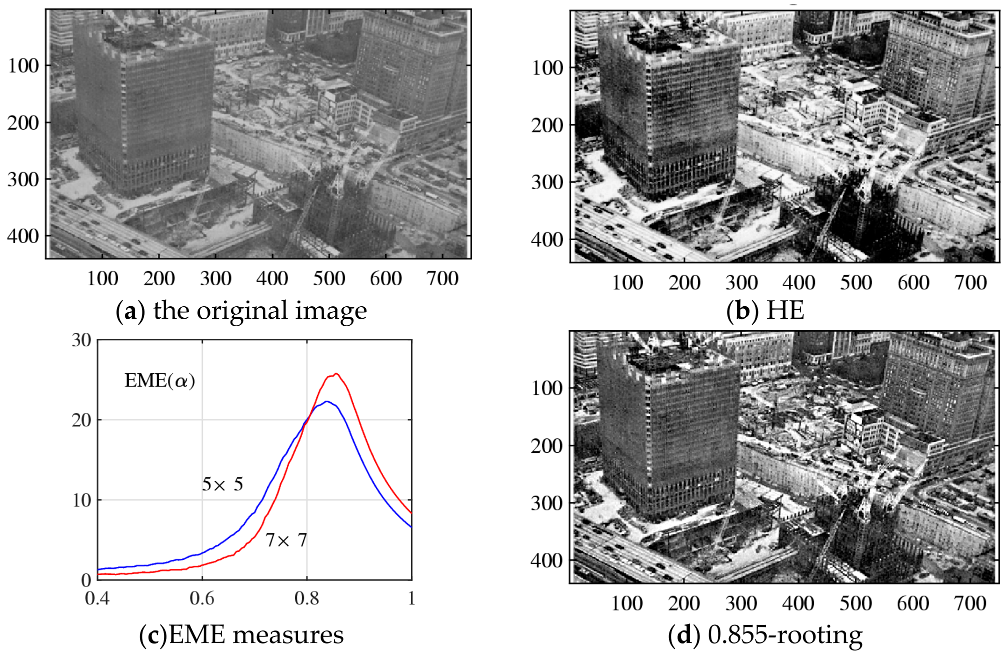


Figure 5. (**a**) The image ‘7.1.10.tiff’ (from http://sipi.usc.edu/database), (**b**) the histogram of the image, (**c**) the EME function, (**d**) the enhanced image by the 0.83-rooting. (**e**) The graph of the EMEE function and (**f**) the image enhanced by the 0.84-rooting method.

In Figure 6a, the image of 440 × 750 pixels is shown, as well as the result of the histogram equalization (HE) of the image in part (b). The graph of the enhancement measure E$ME(\alpha)$, when processing by the alpha-rooting, is given in part (c). The measure function $EME(\alpha)$ was calculated by dividing the image by blocks of sizes 5 × 5 and 7 × 7. The parameter α for the α-rooting method of enhancement varies in the interval [0.4,1] with a step of 0.005. Two graphs of the enhancement measure EME have pikes at the point 0.84 and 0.855, for the 5 × 5 and 7 × 7 block sizes, respectively. These values are almost the same, and we consider $\alpha_0 = 0.855$ for the best visual estimation of the enhancement. The EME of the original image equals 8.30 and 25.76 for the 0.855-rooting enhancement, which is shown in part (d). There, the enhancement can be estimated as $EME(g_{0.855}) - EME(f) = 25.76 - 8.30 = 17.46$. One can note the high quality of the 0.855-rooting image in comparison with the HE image in part (b).



(**a**) the original image

(**b**) HE

(**c**)EME measures

(**d**) 0.855-rooting

Figure 6. (**a**) The original grayscale image and (**b**) enhanced image by (**b**) histogram equalization. (**c**) Two EME measures of alpha-rooting method, and (**d**) the 0.855-rooting of the image.

To estimate the quality of color images, we consider the color image enhancement measure (EMEC). For a color image $f_{n,m} = (r_{n,m}, g_{n,m}, b_{n,m})$ after division by blocks of size $L_1 \times L_2$ each, for instance 7×7, the measure is calculated by

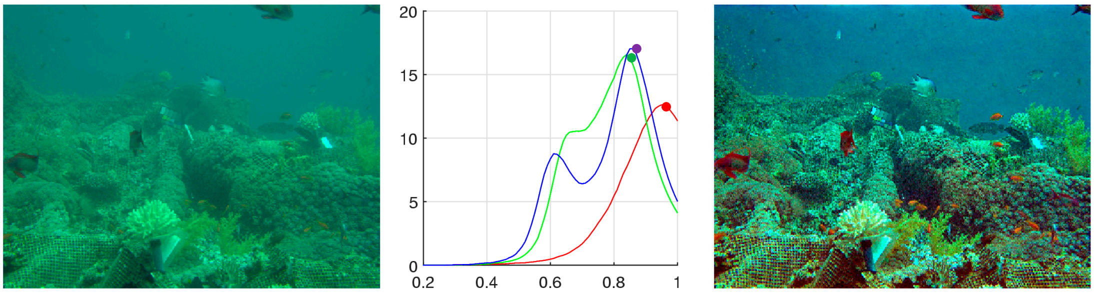
$$EMEC(f)=\frac{1}{k_1k_2}\sum_{k=1}^{k_1}\sum_{l=1}^{k_2}20\log_{10}\left[\frac{\max_{k,l}(r_{n,m},g_{n,m},b_{n,m})}{\min_{k,l}(r_{n,m},g_{n,m},b_{n,m})}\right]. \tag{39}$$

Here, k_1k_2 is the number of blocks, and $\max_{k,l}(\ldots)$ and $\min_{k,l}(\ldots)$ are the maximum and minimum values in the (k,l)-th image block, respectively.

B. Alpha-rooting components-wise

Color images in the RGB color model can be separately processed by red, green, and blue colors. This is the traditional method of processing color images. In the alpha-rooting enhancement, each color component can be processed by alpha-rooting with different or the same values of parameters α_1,α_2, and α_3. We call this method $(\alpha_1,\alpha_2,\alpha_3)$-rooting of the color image For images in the HSI color model, with hue (H), saturation (S), and intensity (I) components, only the last component, intensity, will be only processed by alpha-rooting. The first two components, hue and saturation will stay the same.

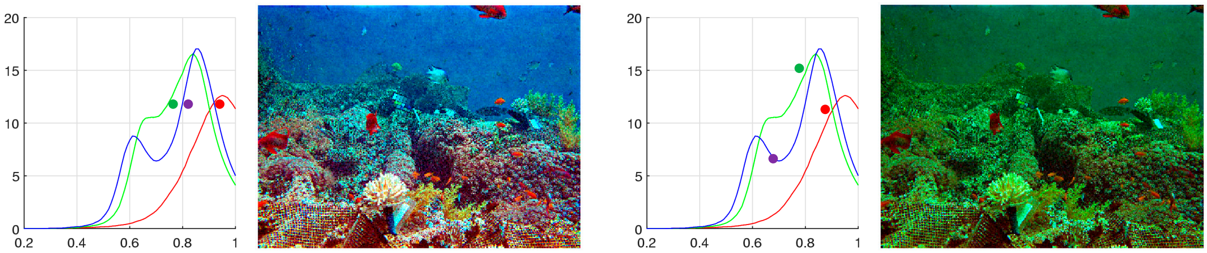
To choose values of these parameters, we can use, for instance, the EME measure. As an example, Figure 7 shows the 1516 × 2012-pixel underwater RGB image in part (a) with EMEC of 38.77, which was calculated by blocks of size 5 × 7. In part (b), the graphs of functions $EME(\alpha)$ of the red, green, and blue channels are shown. The parameter of α runs the interval [0.2,1]. The maximum values of these functions are at points $\alpha=0.94, 0.83$, and 0.84. The color image composed of 0.94-rooting of red, 0.83-rooting of green, and 0.84-rooting of blue components is shown in part (c). The enhancement measure of this image equals EMEC = 44.08.



(**a**) EMEC 38.77 (**b**) (**c**) EMEC 44.08

Figure 7. (**a**) The original image, (**b**) EMEs of red, green, and blue components, and (**c**) enhanced image.

Since the color components are processed separately, it is not possible to state that the above (094,0.84,0.83)-rooting results in the highest enhanced image. It is possible to select other triplets of the vector parameter $\boldsymbol{\alpha}=(\alpha_1,\alpha_2,\alpha_3)$ and obtain images that we can consider the best. As examples, Figure 8 shows two enhanced images together with the graphs of the EMEs of three color channels, R, B, and B. The values of alpha parameters for these channels are marked on the graphs. The case with equal EME for all color channels is shown in part (a). The EMEC of the color image is of 59.62, which is the highest number for all considered cases. Also, a good, enhanced color image is shown in part (b) for the vector parameter (0.9,0.8,0.7).

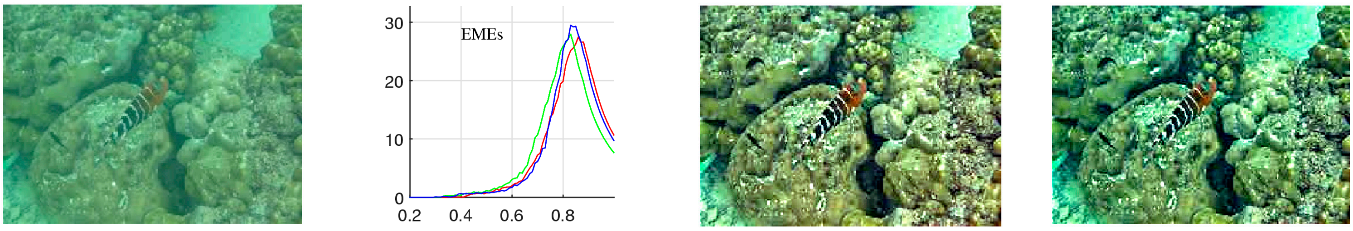


(**a**) (0.92,0.75,0.80)-rooting (EMEC = 59.62) (**b**) (0.9,0.8,0.7)-rooting (EMEC = 55.71)

Figure 8. (**a**) and (**b**) Two enhanced images.

C. Comparison with HE and Retinex

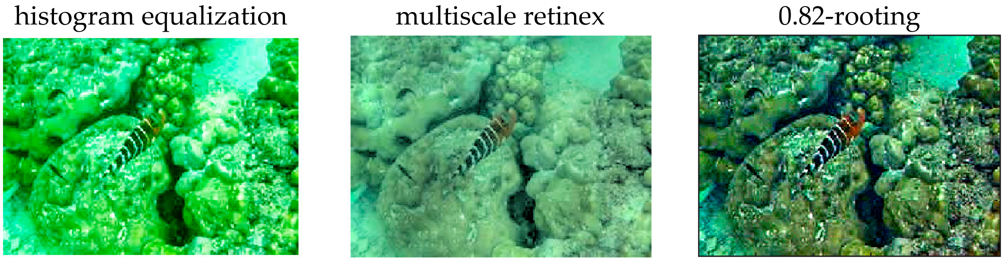
The methods of histogram equalization (HE) [30–32] and Retinex [33–37] are widely used in color image enhancement. We consider these methods together with the method of alpha-rooting. The underwater RGB color image of 192 × 262 pixels is shown in Figure 9 in part (a). This image has a measured EMEC of 11.26, which was calculated by blocks 5 × 5. The graphs of EME of three colors are given in part (b), with maximum values at points 0.85, 0.82, and 0.82, for the red, green, and blue channels, respectively. The corresponding (0.85,0.82,0.82)-rooting of this image with an EMEC of 35.46 is shown in part (c). In part (d), the (0.82,0.82,0.82)-rooting is shown with an EMEC of 36.37.



(**a**) EMEC 14.28 (**b**) (**c**) EMEC 35.46 (**d**) EMEC 36.37

Figure 9. (**a**) The original image, (**b**) EMEs of red, green, and blue components, and enhanced images by (**c**) (0.85,0.82,0.82)-rooting and (**d**) (0.82,0.82,0.82)-rooting.

Figure 10 shows the result of the histogram equalization with a measured EMEC of 44.29 in part (a). The result of image enhancement by the multi-retinex is shown in part (b). The image was normalized, and sizes of the Gaussian filters were taken 7, 15, and 21 as suggested [33]. The retinex enhancement has an EMEC of 16.96. One can see that the enhancement of the color image was not achieved in these two methods. For comparison, we also add the result of the color image enhancement by the 0.82-rooting. The result is shown in part (c). One can see good enhancement of the image; the color measure of enhancement equals 33.50. Measures of EMEC and EME were calculated by blocks of 5 × 5 pixels.



(**a**) EMEC 44.29 (**b**) EMEC 16.96 (**c**) EMEC 33.50

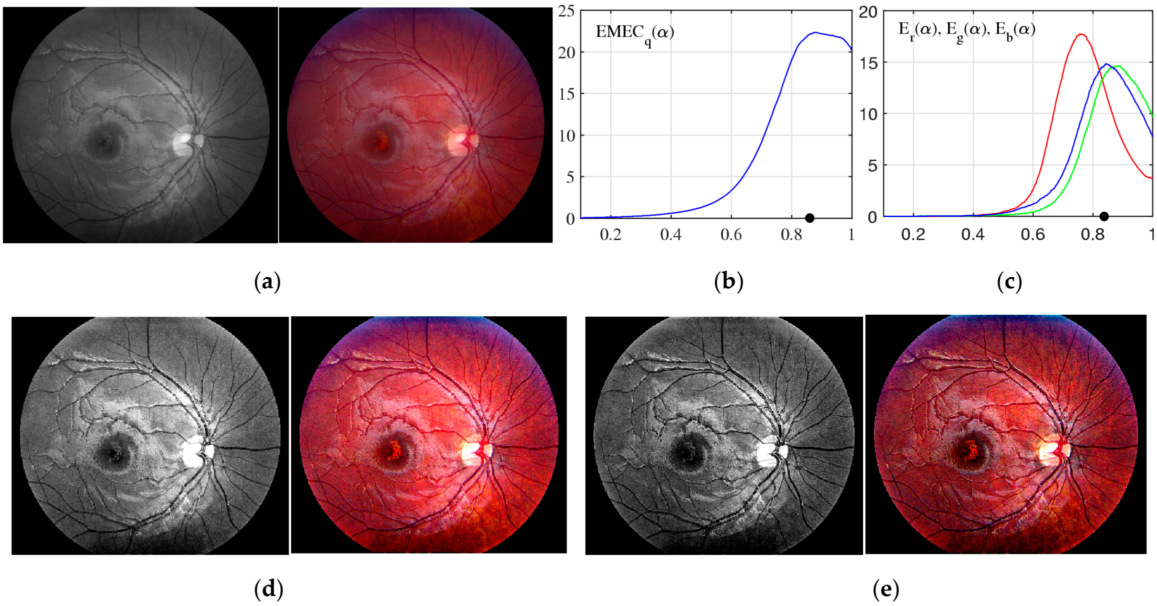
Figure 10. Color image enhancement by (**a**) histogram equalization (MATLAB’s version), (**b**) multi-retinex algorithm (the original version [33]), and (**c**) method of 0.82-rooting.

The quaternion image enhancement (EMEQ) measure for a quaternion image $q_{n,m} = (a_{n,m}, r_{n,m}, g_{n,m}, b_{n,m})$ is calculated similarly [27],

$$EMEQ(q) = \frac{1}{k_1 k_2} \sum_{k=1}^{k_1} \sum_{l=1}^{k_2} 20\log_{10}\left[\frac{\max_{k,l}(a_{n,m}, r_{n,m}, g_{n,m}, b_{n,m})}{\min_{k,l}(a_{n,m}, r_{n,m}, g_{n,m}, b_{n,m})}\right]. \tag{40}$$

This measure includes the real part of the quaternion image. The measure EMEQ is calculated for the input quaternion image $q_{n,m}$ and the processed image $v_{n,m}$. In most cases, the best parameter for color enhancement is considered the value of α with a maximum of $EMEC(q)$ and $EMEQ(v)$ (or minimum). Our experimental results show that the measures EMEC and EMEQ are effective in selecting the best parameters to receive color images with high quality [27]. Other measures for selecting the best values of α and estimating color image quality after image processing can also be used. We mention the color image contrast and quality measures [14].

As an example, Figure 11 shows the quaternion image of 877 × 1024 pixels in part (a). The grayscale image is the real part, and the color image is the imaginary part of the quaternion image. The graph of the EMEC measure as the function of α is shown in part (b). The maximum of this function is at point 0.879. In part (c), the graphs of the measured EME of the color channels are given. The point $\alpha = 0.82$ was selected, at which these graphs roughly intersect. The quaternion images after 0.879 and 0.82-rooting enhancements are shown in parts (d) and (e), respectively.



(**a**) (**b**) (**c**)

(**d**) (**e**)

Figure 11. (**a**) The quaternion fundus image, (**b**) the graph of the EMEC function calculated for α-rooting by the 2-D QDFT, (**c**) the graphs of EME functions calculated for red, green, and blue channels of the α-rooting. The enhanced quaternion after (**d**) 0.879-rooting and (**e**) 0.82-rooting.

5.2. The Separable Alpha-Rooting

The alpha-rooting method by the QDFT can be modified in the following two ways.

1. The separable 1-parameter alpha-rooting of the quaternion image $q_{n,m} = [f_{n,m}, g_{n,m}]$ is the method of processing the 2D e_2-QDFT of the image as

$$Q_{p,s} = [F_{p,s}, G_{p,s}] \to [F_{p,s}|F_{p,s}|^{\alpha-1}, G_{p,s}|G_{p,s}|^{\alpha-1}], \alpha \in (0,1). \tag{41}$$

2. The 2-parameter alpha-rooting of the quaternion image uses two parameters α_1 and α_2 from the interval [0,1] to process the 2D e$_2$-QDFT of the quaternion image as follows:

$$Q_{p,s} = [F_{p,s}, G_{p,s}] \rightarrow [F_{p,s}|F_{p,s}|^{\alpha_1 - 1}, G_{p,s}|G_{p,s}|^{\alpha_2 - 1}]. \quad (42)$$

In the $\alpha_1 = \alpha_2 = \alpha$ case, the 2-parameter alpha-rooting coincides with the 1-parameter alpha-rooting.

5.3. Alpha-Rooting of Color Images and the (1,3)-Model

In the (1,3)-model, we consider one of the 2D QDFTs, namely, the separable right-sided 2D QDFT [27]. This transform of the quaternion image $q_{n,m} = (a_{n,m}, r_{n,m}, g_{n,m}, b_{n,m})$ is calculated by

$$Q_{p,s} = \sum_{n=0}^{N-1} \left(\sum_{m=0}^{M-1} q_{n,m} W_{\mu_1}^{ms} \right) W_{\mu_2}^{np}, p, s = 0: (N-1), (M-1). \quad (43)$$

Here, μ_1 and μ_2 are pure quaternion units. The transform uses N 1D QDFTs by rows and then M 1D QDFT by columns. Given quaternion signal $q_n = (a_n, r_n, g_n, b_n)$ and a pure quaternion $\mu = (0, m_1, m_2, m_3)$, the 1D QDFT, Q_p, with the basis exponential functions $W_\mu^{np} = \cos(2\pi np/N) - \mu \sin(2\pi np/N)$ requires four traditional DFTs since it is calculated by [14]

$$Q_p = \mathrm{Re} \begin{bmatrix} A_p \\ R_p \\ G_p \\ B_p \end{bmatrix} + M_\mu \times \mathrm{Im} \begin{bmatrix} A_p \\ R_p \\ G_p \\ B_p \end{bmatrix} = \mathrm{Re} \begin{bmatrix} A_p \\ R_p \\ G_p \\ B_p \end{bmatrix} + \begin{bmatrix} 0 & m_1 & m_2 & m_3 \\ -m_1 & 0 & -m_3 & m_2 \\ -m_2 & m_3 & 0 & -m_1 \\ -m_3 & -m_2 & m_1 & 0 \end{bmatrix} \mathrm{Im} \begin{bmatrix} A_p \\ R_p \\ G_p \\ B_p \end{bmatrix}. \quad (44)$$

A_p, R_p, G_p, and B_p are the DFTs of the components a_n, r_n, g_n, and b_n, respectively. $\mathrm{Re}(z)$ and $\mathrm{Im}(z)$ denote the operations of real and imaginary parts of the complex number z, respectively. The multiplication of the 4D vector by the matrix M_μ requires a maximum of 12 real multiplications. In the case, when $\mu_1 = (0,0,1,0) = j$ and $\mu_2 = (0,0,0,1) = k$, the exponential basis functions are $W_k^{ms} = \cos(2\pi ms/M) - k \sin(2\pi ms/M)$ and $W_j^{np} = \cos(2\pi np/N) - j \sin(2\pi np/N)$. The matrices of multiplication have simple forms,

$$M_j = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{bmatrix} \text{ and } M_k = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \end{bmatrix}.$$

and the corresponding 1D QDFTs are calculated by

$$Q_p = \begin{bmatrix} Re(A_p) + Im(G_p) \\ Re(R_p) + Im(B_p) \\ Re(G_p) - Im(A_p) \\ Re(B_p) - Im(R_p) \end{bmatrix} \text{ and } Q_p = \begin{bmatrix} Re(A_p) + Im(B_p) \\ Re(R_p) - Im(G_p) \\ Re(G_p) + Im(R_p) \\ Re(B_p) - Im(A_p) \end{bmatrix}.$$

These N-point QDFTs require four 1D DFTs plus $4N$ additions. In this case, the right-sided 2D QDFT of the quaternion image $q_{n,m}$ is calculated by

$$Q_{p,s} = \sum_{n=0}^{N-1} \left(\sum_{m=0}^{M-1} q_{n,m} W_k^{ms} \right) W_j^{np}, p = 0: (N-1), s = 0: (M-1). \quad (45)$$

A total of $4(N+M)$ 1D QDFTs plus $4(NM) + (4M)N = 8NM$ additions are used to calculate the 2D QDFT. The inverse 2D right-sided QDFT is calculated by

$$q_{n,m} = \frac{1}{NM}\sum_{p=0}^{N-1}\left(\sum_{s=0}^{M-1} Q_{p,s} W_k^{-ms}\right) W_j^{-np}, \tag{46}$$

The complexity of the QDFTs in the (1,3) and (2,2)-models for images of $N \times N$ pixels is described in Table 4.

Table 4. Complexity of the calculations for the two algebras.

Model	Transforms	Number of 1D DFTs	Number of Additional Multiplications	Number of Additional Additions
The (1,3)-model:				
General case of μ	1D QDFT	4 (real)	$12N$	$12N$
	2D QDFT	$4(2N) = 8N$	$12N(2N) = 24N^2$	$12N(2N) = 24N^2$
Case $\mu = j, k$	1D QDFT	4 (real)	-	$4N$
	2D QDFT	$4(2N) = 8N$	-	$4N(2N) = 8N^2$
The (2,2)-model:				
1D e_2-QDFT	1D QDFT	2 (complex)	-	-
2D e_2-QDFT	2D QDFT	$2(2N) = 4N$	-	-

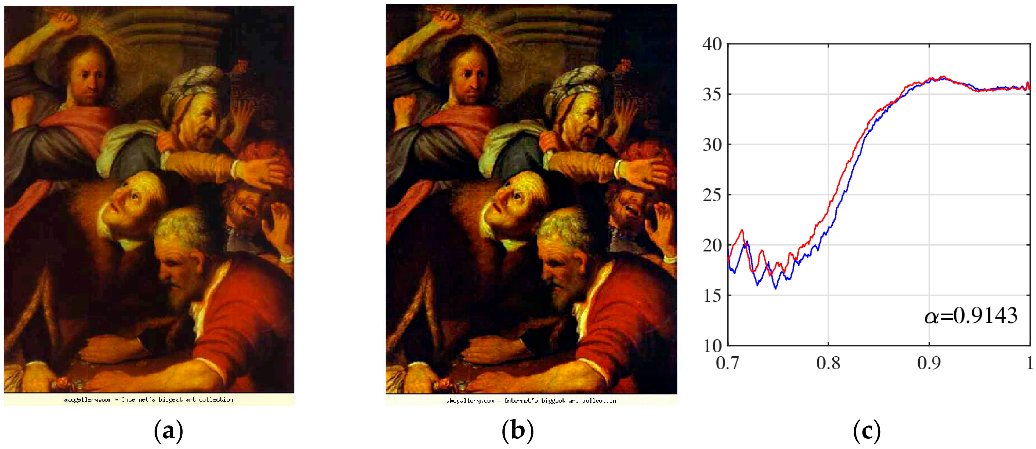
The main steps of the algorithm for α-rooting in the (1,3)-model:

1. Compose the quaternion image $q_{n,m} = (a_{n,m}, r_{n,m}, g_{n,m}, b_{n,m})$ from the color RGB image $(r_{n,m}, g_{n,m}, b_{n,m})$.
2. Calculate the right-sided 2D QDFT, $Q_{p,s}$, of the quaternion image.
3. Given $\alpha \in (0,1)$, calculate the coefficients $c(p,s) = |\ Q_{p,s}|^{\alpha-1}$.
4. Modify the 2D QDFT as $Q_{p,s} \to V_{p,s} = c(p,s)Q_{p,s}$.
5. Calculate the inverse 2D QDFT $v_{n,m} = v_{n,m}(\alpha)$.
6. Select the best value α for color image enhancement by using the measures EMEQ or EMEC.

6. Experimental Results with Color Images

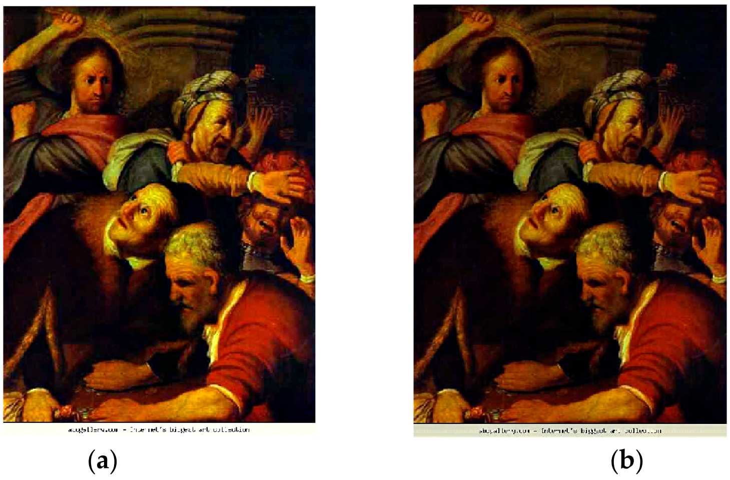
In this section, a few illustrative examples of the 2D QDFT-based alpha-rooting are presented. Many color images of art in this paper are from Olga's Gallery—Free Art Print Museum by address https://www.freeart.com/gallery/ with permission to use them in our research. Figure 12 shows the RGB color image ‘rembrandt195.jpg’ in part (a) and the enhanced image in part (b). The enhanced image was calculated by the alpha-rooting with e_2-QDFT, when the parameter $\alpha = 0.9143$. This value of the parameter is considered optimal, or best, according to the EMEC measure calculated by Equation (39) with block size 7 × 7. This measure as the function EMEC(α) has a maximum of 36.54 at this point. The measure of the original image is EMEC(1) = 34.74.

Two EMEC functions are shown in Figure 12 in part (c); they are close to each other, and both achieve the maximum at the same point. The first graph (which is a little higher than the other one) was calculated by the 2D e_2-QDFT-based alpha-rooting described in Section 5.1, when the transform is modified as $Q_{p,s} = [F_{p,s}, G_{p,s}] \to |Q_{p,s}|^{\alpha-1}[F_{p,s}, G_{p,s}]$, $\alpha \in [0.7,1]$. The second graph is for the EMEC measure calculated from the 1-parameter alpha-rooting described in Section 5.2, when the e_2-QDFT of the images is processed as follows: $Q_{p,s} = [F_{p,s}, G_{p,s}] \to [F_{p,s}|F_{p,s}|^{\alpha-1}, G_{p,s}|G_{p,s}|^{\alpha-1}]$, $\alpha \in [0.7,1]$. Figure 13 shows the enhanced image by 1-parameter 0.9143-rooting in part (a). For comparison, the 0.9143-rooting of the image by the 2D QDFT in the (1,3)-model is shown in part (b).



(**a**) (**b**) (**c**)

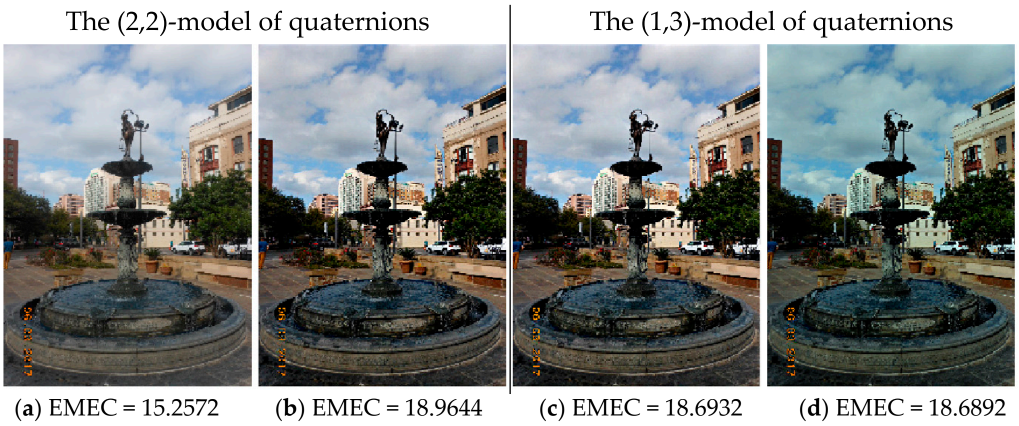
Figure 12. (**a**) The original image, (**b**) 2D e2-QDFT based 0.9143-rooting (with the scaling factor of $A = 4$), and (**c**) the two curves of the EMEC.



(**a**) (**b**)

Figure 13. The enhanced images of the 0.9143-rooting: (**a**) in the (2,2)-model and (**b**) in the (1,3)-model.

Below are a few results of processing other color images by the alpha-rooting and separate algorithms of the alpha-rooting in the commutative (2,2)-model. The results of image enhancement by the alpha-rooting in the non-commutative (1,3)-model are also shown. Figure 14 shows the results of the 0.92-rootings, when processing the image of San Antonio. The values of the color image enhancement EMEC are shown.

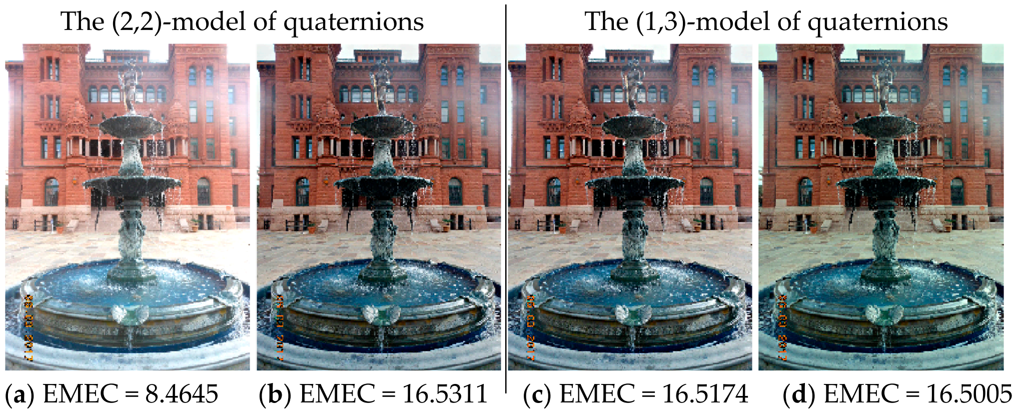


The (2,2)-model of quaternions The (1,3)-model of quaternions

(**a**) EMEC = 15.2572 (**b**) EMEC = 18.9644 (**c**) EMEC = 18.6932 (**d**) EMEC = 18.6892

Figure 14. (**a**) The original color image. The enhanced images in the (2,2)-model by (**b**) the main 0.92-rooting and (**c**) separable 1-parameter 0.92-rooting. (**d**) The 0.92-rooting in the (1,3)-model.

Figure 15 shows the results of the same methods of the 0.92-rootings, when processing another image of San Antonio. One can note that the images processed in the (2,2)-model have higher values of EMEC.

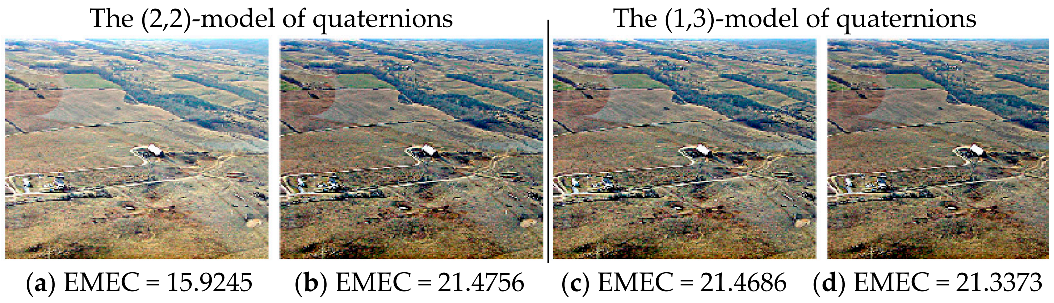

The (2,2)-model of quaternions

The (1,3)-model of quaternions

(**a**) EMEC = 8.4645 (**b**) EMEC = 16.5311 (**c**) EMEC = 16.5174 (**d**) EMEC = 16.5005

Figure 15. (**a**) The original color image. The enhanced images in the (2,2)-model by (**b**) the main 0.92-rooting and (**c**) separable 1-parameter 0.92-rooting. (**d**) The 0.92-rooting in the (1,3)-model.

Figure 16 shows the results of processing image ‘image13-2.jpg.’ The method of alpha-rooting works well in both models for many images. It means that the (2,2)-model does not perform any worse but in fact better than another model, that is, the (1,3)-model.

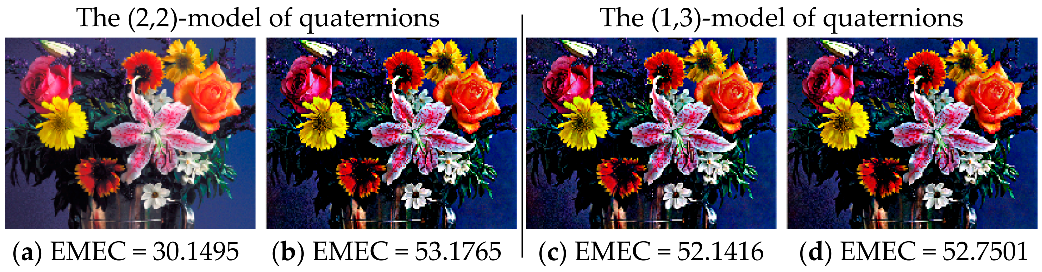

The (2,2)-model of quaternions

The (1,3)-model of quaternions

(**a**) EMEC = 15.9245 (**b**) EMEC = 21.4756 (**c**) EMEC = 21.4686 (**d**) EMEC = 21.3373

Figure 16. (**a**) The original color image. The enhanced images in the (2,2)-model by (**b**) the main 0.92-rooting and (**c**) separable 1-parameter 0.92-rooting. (**d**) The 0.92-rooting in the (1,3)-model.

The results of processing the well-known “flowers” image are shown in Figure 17 in parts (a)–(d).


The (2,2)-model of quaternions

The (1,3)-model of quaternions

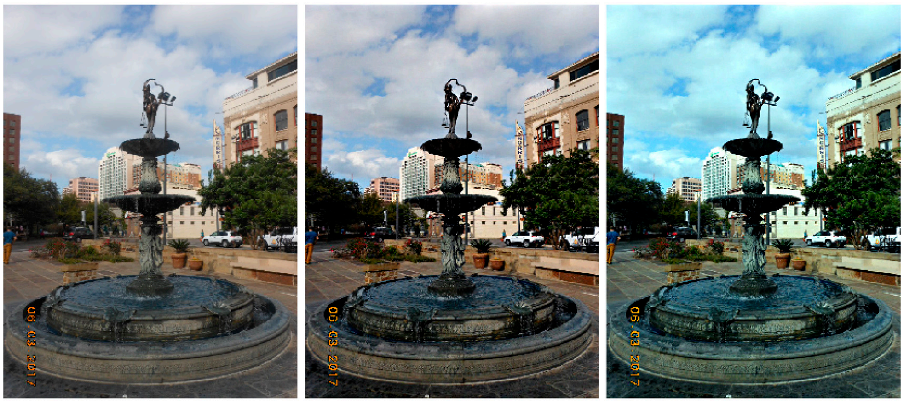
(**a**) EMEC = 30.1495 (**b**) EMEC = 53.1765 (**c**) EMEC = 52.1416 (**d**) EMEC = 52.7501

Figure 17. (**a**) The original color image. The enhanced images in the (2,2)-model by (**b**) the main 0.92-rooting and (**c**) separable 1-parameter 0.92-rooting. (**d**) The 0.92-rooting in the (1,3)-model.

Now we apply the method of alpha-rooting in the (2,2)-model, when two parameters α_1 and α_2 are used and the 2D QDFT of the color image is processed as

$$Q_{p,s} = [F_{p,s}, G_{p,s}] \to [F_{p,s}|F_{p,s}|^{\alpha_1 - 1}, G_{p,s}|G_{p,s}|^{\alpha_2 - 1}], \alpha_1, \alpha_2 \in (0,1]. \quad (47)$$

Figure 18 shows results of the color image enhancement processing by the 2-parameter alpha-rooting with different sets of parameters α_1 and α_2. In part (b), the image of San Antonio was processed by the parameters $\alpha_1 = \alpha_2 = 0.92$. The enhancement by parameters $\alpha_1 = 0.92$ and $\alpha_2 = 0.93$ is shown in part (c).

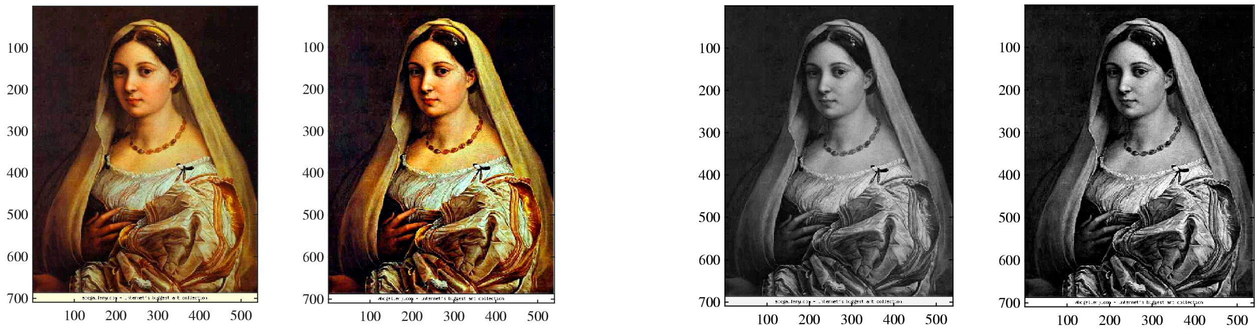


(**a**) EMEC = 15.2572 (**b**) EMEC = 18.6932 (**c**) EMEC = 18.8947

Figure 18. (**a**) The original color image, and (**b**) the [0.92,0.92]-rooting, and (**c**) the [0.92,0.93]-rooting in the (2,2)-model.

It should be noted that when processing color images in the quaternion models, the color image is only the imaginary part of the quaternion image. The enhancement of quaternion image includes two images, the color one and the gray one. They are processed together. The first component of the quaternion image, which is referred to as the grayscale image is not the grayscale image of the processed color image. The enhancement of quaternion image results in the enhancement of both images. As examples, we consider a few color images processed in the (2,2)-model by the 2-D e$_2$-QDFT-based alpha-rooting.

Figure 19 shows the color image ‘raphael155.jpg’ in part (a), which was embedded in the quaternion image as its imaginary part. The imaginary component (the new color image) of the enhanced quaternion image by the 0.92-rooting is shown in part (b). The grayscale image of the original color image is shown in part (c). The real part of the processed quaternion image is shown in part (d). This image is not the average of colors in the image in part (b). Thus, both grayscale and color images were enhanced when processing the quaternion image.



(**a**) (**b**) (**c**) (**d**)

Figure 19. (**a**) The original color image and (**c**) its grayscale image. The processed (**b**) imaginary and (**d**) real components of the enhanced quaternion image by the e_2-QDFT 0.92-rooting.

Figures 20 and 21 show the results of enhancement of the quaternion images when the color images ‘raphael155.jpg’ and ‘flowers’ were used, respectively.

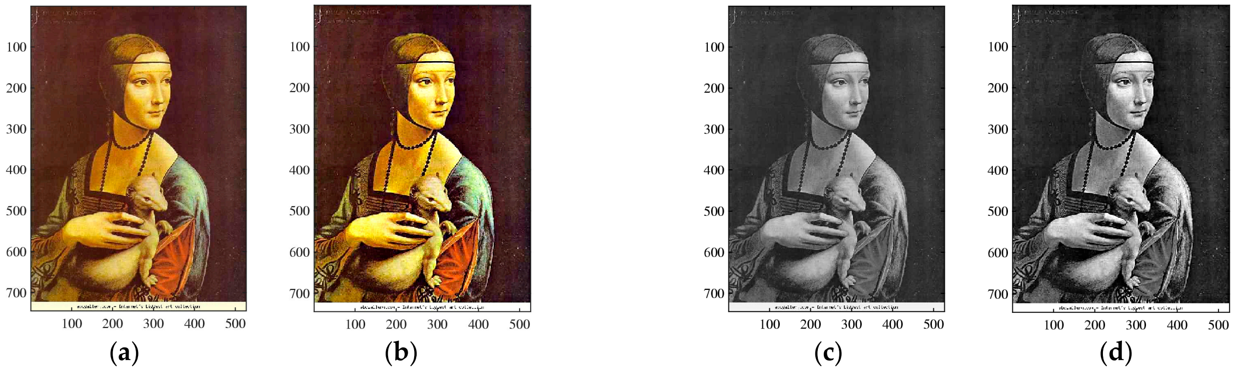


Figure 20. (**a**) The original color image ‘leonardo9.jpg’ and (**c**) its grayscale image. The processed (**b**) imaginary and (**d**) real components of the enhanced quaternion image by the e_2-QDFT 0.92-rooting (×4).

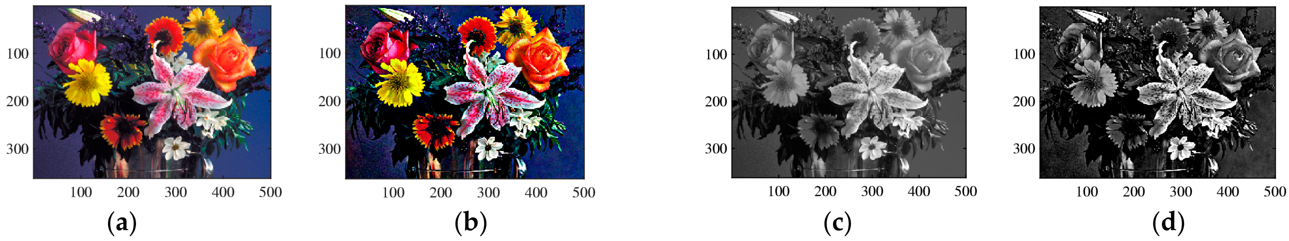


Figure 21. (**a**) The original color flowers image and (**c**) its grayscale image. The processed (**b**) imaginary and (**d**) real components of the enhanced quaternion image by the e_2-QDFT 0.80-rooting (× 20).

7. Conclusions

New quaternion algebra, the (2,2)-model, was presented, and new methods of alpha-rooting by the quaternion discrete Fourier transform (QDFT) were described and analyzed in this model. The main properties of this model were considered. This model of quaternions is commutative and associative and allows to calculate the aperiodic convolution of quaternion images in the frequency domain. The results of the image enhancement of color images in this model were compared with the alpha-rooting in the traditional (1,3)-model. The comparison with the known methods of histogram equalization and Retinex is also provided with examples. The preliminary experimental examples show the effectiveness of the proposed methods for color image enhancement by the 2D QDFT. We believe that the commutative (2,2)-model together with the non-commutative (1,3)-model can be effectively used in color image enhancement, as well as other areas of color imaging. The proposed methods of alpha-rooting are fast, because of fast 1D and 2D QDFTs, and do not require much memory, as well as machine learning algorithms, which require much time and memory and do not work well on many images presented in this work.

Author Contributions: Conceptualization, A.M.G.; methodology, A.M.G.; software, A.M.G. and A.A.G.; validation, A.M.G.; formal analysis, A.M.G. and A.A.G.; investigation, A.M.G.; resources, A.M.G.; data curation, A.M.G.; writing—original draft preparation, A.M.G.; writing—review and editing, A.M.G. and A.A.G.; visualization, A.M.G. and A.A.G.; supervision, A.M.G.; project administration, A.M.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author(s).

Acknowledge: This article is a revised and expanded version of Ref. [25], which was presented at *SPIE 13033 Conference, Defense + Commercial Sensing 2024,* 22 April 2024, Maryland, USA.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Harris, J.L. Constant variance enhancement: A digital processing technique. *Appl. Opt.* **1977**, *16*, 1268.
2. Wang, D.C.; Vagnucci, A.H.; Li, C.C. Digital image enhancement: A survey. *Comput. Vis. Graph. Image Process.* **1983**, *24*, 363–381.
3. Trussell, H.J.; Saber, E.; Vrhel, M. Color image processing [basics and special issue overview]. *IEEE Signal Process. Mag.* **2005**, *22*, 14–22.
4. Gonzalez, R.C.; Woods, R.E. *Digital Image Processing*, 4th ed.; Pearson: New York, NY, USA, 2018.
5. Grigoryan, A.M.; Agaian, S.S. Image processing contrast enhancement. In *Wiley Encyclopedia of Electrical and Electronics Engineering*; Webster, J.G., Ed.; Wiley: Hoboken, NJ, USA, 2017; pp. 1–22.
6. Nithyananda, C.R.; Ramachandra, A.C.; Preethi. Survey on histogram equalization method-based image enhancement techniques. In Proceedings of the 2016 International Conference on Data Mining and Advanced Computing (SAPIENCE), Ernakulam, India, 16–18 March 2016; IEEE: New York, NY, USA, 2016; pp. 150–158.
7. Han, J.-H.; Yang, S.; Lee, B.-U. A novel 3-D color histogram equalization method with uniform 1-D gray scale histogram. *IEEE Trans. Image Process.* **2011**, *20*, 506–512.
8. Trahanias, P.E.; Venetsanopoulos, A.N. Color image enhancement through 3-D histogram equalization. In Proceedings of the 11th IAPR International Conference on Pattern Recognition, The Hague, The Netherlands, 30 August–3 September 1992; IEEE Comput. Soc. Press: Washington, DC, USA, 1992; Volume IV, pp. 545–548.
9. Pitas, I.; Kiniklis, P. Multichannel techniques in color image enhancement and modeling. *IEEE Trans. Image Process.* **1996**, *5*, 168–171.
10. Mlsna, P.A.; Zhang, Q.; Rodriguez, J.J. 3-D histogram modification of color images. In Proceedings of the 3rd IEEE International Conference on Image Processing, Lausanne, Switzerland, 16–19 September 1996; IEEE: New York, NY, USA, 1996; Volume 3, pp. 1015–1018.
11. Land, E.H.; McCann, J.J. Lightness and retinex theory. *J. Opt. Soc. Am.* **1971**, *61*, 1–11.
12. Land, E. An alternative technique for the computation of the designator in the retinex theory of color vision. *Proc. Natl. Acad. Sci. USA* **1986**, *83*, 3078–3080.
13. Agaian, S.S.; Panetta, K.; Grigoryan, A.M. Transform-based image enhancement algorithms with performance measure. *IEEE Trans. Image Process.* **2001**, *10*, 367–382.
14. Grigoryan, A.M.; Agaian, S.S. *Quaternion and Octonion Color Image Processing with MATLAB*; SPIE Press: Bellingham, WA, USA, 2018.
15. Gauss, C.F. Mutationen des raumes [transformations of space] (c. 1819). In *Carl Friedrich Gauss Werke*, 8th ed.; Brendel, M., Ed.; Teubner: Stuttgart, Germany, 1900; pp. 357–361.
16. Hamilton, W.R. On a new species of imaginary quantities connected with a theory of quaternions. *Proc. R. Ir. Acad.* **1844**, *2*, 424–434. Available online: https://www.emis.de/classics/Hamilton/Quatern1.pdf (accessed on 27 December 2024).
17. Hamilton, W.R. *Elements of Quaternions*; Longmans, Green & Co.: London, UK, 1866.
18. Kantor, I.L.; Solodovnikov, A.S. *Hypercomplex Numbers*; Nauka: Moscow, Russia, 1973.
19. Bülow, T. *Hypercomplex Spectral Signal Representations for the Processing and Analysis of Images*; Christian-Albrechts-Univ.: Kiel, Germany, 1999; Volume 9903, p. 171.
20. Yin, Q.; Wang, J.; Luo, X.; Zhai, J.; Jha, S.K.; Shi, Y.-Q. Quaternion convolutional neural network for color image classification and forensics. *IEEE Access* **2019**, *7*, 20293–20301.
21. Sangwine, S.J. Fourier transforms of colour images using quaternion, or hypercomplex, numbers. *Electron. Lett.* **1996**, *32*, 1979–1980.
22. Grigoryan, A.M.; Agaian, S.S. Commutative quaternion algebra and DSP fundamental properties: Quaternion convolution and Fourier transform. *Signal Process.* **2022**, *196*, 108533.
23. Davenport, C. Commutative hypercomplex mathematics. Unpublished work, 2008. https://swissenschaft.ch/tesla/content/T_Library/L_Theory/EM%20Field%20Research/Hypercomplex%20Commutative%20Mathematics.pdf
24. Ell, T.A.; Sangwine, S.J. Hypercomplex Fourier transforms of color images. *IEEE Trans. Image Process.* **2007**, *16*, 22–35.
25. Grigoryan, A.M.; Gomez, A.A. Quaternion Fourier transform-based alpha-rooting color image enhancement in 2 algebras: Commutative and non-commutative. In Proceedings of the SPIE 13033 Conference, Defense + Commercial Sensing 2024, National Harbor, MD, USA, 21–25 April 2024; SPIE: Bellingham, WA, USA, 2024; p. 12. https://doi.org/10.1117/12.3017692.
26. McClellan, J.H. Artifacts in alpha-rooting of images. In Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, Denver, CO, USA, 9–11 April 1980; pp. 449–452.
27. Grigoryan, A.M.; Jenkinson, J.; Agaian, S.S. Quaternion Fourier transform-based alpha-rooting method for color image measurement and enhancement. *Signal Process.* **2015**, *109*, 269–289.
28. Fechner, G.T. *Elements of Psychophysics*; Rinehart & Winston: New York, NY, USA, 1960; Volume 1.
29. Gordon, I.E. *Theory of Visual Perception*; John Wiley & Sons: New York, NY, USA, 1989.
30. Zuiderveld, K. Contrast limited adaptive histogram equalization. In *Graphics Gems IV*; Academic Press: San Diego, CA, USA, 1994; pp. 474–485.

31. Kim, Y.T. Contrast enhancement using brightness preserving bi-histogram equalization. *IEEE Trans. Consum. Electron.* **1997**, *43*, 1–8.
32. Zhu, H.; Chan, F.H.; Lam, F.K. Image contrast enhancement by constrained local histogram equalization. *Comput. Vis. Image Underst.* **1999**, *73*, 281–290.
33. Jabson, D.J.; Rahmann, Z.; Woodell, G.A. A multiscale retinex for bridging the gap between color images and the human observations of scenes. *IEEE Trans. Image Process.* **1997**, *6*, 897–1056.
34. Chen, S.; Beghdadi, A. Nature rendering of color image based on Retinex. In Proceedings of the IEEE International Conference on Image Processing, Cairo, Egypt, 7–10 November 2009; pp. 1813–1816.
35. Huang, K.-Q.; Wang, Q.; Wu, Z.-Y. Natural color image enhancement and evaluation algorithm based on human visual system. *Comput. Vis. Image Underst.* **2006**, *103*, 52–63.
36. Struc, V.; Pavei, N. Photometric normalization techniques for illumination invariance. In *Advances in Face Image Analysis: Techniques and Technologies*; Zhang, Y.J., Ed.; IGI Global: Hershey, PA, USA, 2011; pp. 279–300.
37. Struc, V.; Pavei, N. Gabor-based kernel-partial-least-squares discrimination features for face recognition. *Informatica* **2009**, *20*, 115–138.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.