MULTI-VECTOR COLOR-IMAGE FILTERS

Todd A. Ell

University of Essex
Department of Electronic Systems Engineering
Colchester C04 3SQ, U.K.
Email: t.ell@ieee.org

ABSTRACT

The linear filtering of color images using hypercomplex convolution and Fourier transforms provides a holistic treatment of color by representing pixels as 3-space vector quantities within the quaternion algebra. But, this technique is limited to images with at most three channels of information, e.g., RGB images. Linear filtering of color images by representing color pixels as multi-vectors embedded in a geometric algebra is presented. This multi-vector representation has similar convolution and Fourier transforms as the quaternion based filters, but provides an avenue for multi-spectral images composed of more than three channels.

Index Terms— Geometric Algebra, Multi-vector, Color Image, Convolution Filtering, Spatio-Color Filtering

1. INTRODUCTION

Classic linear filtering, typically implemented by convolution in the spatial domain, can also be implemented in the spatial frequency domain by making use of the Fourier transform. Historically for color image processing this is done by iterating over the color components of the image. This iteration does not do justice to the fact that the information in one color component is usually correlated to the information in the other components. To account for, and exploit, this correlation all three color components should be treated in a holistic fashion. Thornton and Sangwine [1] partially achieved a single spectrum by working with the chromaticity information alone, ignoring the luminance. McCabe, et al. [2] again used a complex Fourier transform to handle chromatic information using an alternate color encoding. Later, Sangwine and Ell [3] used quaternions to encode entire images which leverages the expressive power of quaternion algebra. Recently, it was shown in [4] that the chromatic spectrum is embedded within the hypercomplex (quaternion) spectrum. This work gives yet another holistic encoding. A two-dimensional vector space can be embedded into a four-dimensional *geometric algebra*. By encoding color images as multi-vectors, one gains the same expressive power as quaternion based systems, but obtains an alternative geometric interpretation.

The next section provides a brief introduction to the four dimensional Geometric Algebra G^2. The contents are sufficient to understand the rest of the work. In the subsequent section is presented a technique of encoding color images using multi-vectors taken from G^2. Section 4 shows the construction of chromatic-luminance edge detection filters using a simple technique that encompasses the quaternion based color edge detectors. This is done to show the differences and similarities between this system and the quaternion-based system. Section 5 details the Geometric Fourier Transform used to convert the color images into the frequency domain. Section 6 includes the multi-vector convolution operator formulae, including the bi-convolution and handed-convolution, and spectral versions of the same equations.

The author is a Visiting Fellow at the University of Essex. This work was supported in part by the U.K. Engineering and Physical Sciences Research Council under Grant GR/S58621/01.

2. GEOMETRIC ALGEBRA

The vector space $\mathbb{R}^2$ can be embedded into the geometric algebra G^2. As with quaternions, G^2 has two structures. First, it is a 4-dimensional vector space over $\mathbb{R}$, hence linear matrix theory can be used as an analysis tool. Second, it is an associative non-commutative algebra, whose product is called the geometric product. Members of G^2 are called multi-vectors. Let $\{e_1, e_2\}$ be an ortho-normal basis of $\mathbb{R}^2$. The algebra G^2 is based on two rules:

$$e_i e_i = 1 \quad \text{and} \quad e_i e_j = -e_j e_i, \quad i \neq j \tag{1}$$

The vector space G^2 is 4-dimensional with basis:

1	spans 0-vectors,	scalars,
$\{e_1, e_2\}$	spans 1-vectors,	vectors, and
$e_1 e_2$	spans 2-vectors,	bi-vectors.

Hence an arbitrary multi-vector $A \in G^2$ can be written as

$$A = a_0 + a_1 e_1 + a_2 e_2 + a_{12} e_1 e_2, \ \ a_0, a_1, a_2, a_{12} \in \mathbb{R}.$$

As shorthand notation, let the pseudo-scalar $e_1 e_2 = I_2$. Since $I_2^2 = e_1 e_2\, e_1 e_2 = -e_2 e_1\, e_1 e_2 = -1$, complex numbers are an algebraic system embedded in G^2. This embedding defines the spinor subalgebra, $\mathcal{S}$,

$$\mathcal{S} = \{\alpha + \beta I_2 | \alpha, \beta \in \mathbb{R}\} \subset G^2.$$

A complex number or spinor is a scalar + bi-vector. Therefore an arbitrary multi-vector $A \in G^2$ can be rewritten as a spinor plus a vector as

$$A = (a_0 + a_{12} I_2) + (a_1 e_1 + a_2 e_2)\,.$$

It is important to note that the pseudo-scalar I_2 commutes with complex numbers but anti-commutes with vectors, i.e.,

$$\begin{aligned}[A]\, I_2 &= [(a_0 + a_{12} I_2) + (a_1 e_1 + a_2 e_2)]\, I_2 \\ &= I_2\, [(a_0 + a_{12} I_2) - (a_1 e_1 + a_2 e_2)]\end{aligned}$$

A rotation of a vector $v \in \mathbb{R}^2$ through an angle θ is represented by the unit spinor (complex number) $e^{I_2 \theta/2}$ in the formula

$$v' = e^{-I_2\theta/2} v e^{I_2\theta/2} = e^{-I_2\theta/2} e^{-I_2\theta/2} v = e^{-I_2\theta} v$$

Unlike complex numbers being used to denote 1-vectors with unit complex numbers being rotation operators, this algebra clearly separates vectors from rotation operators (i.e., the operators are spinors). This issue is discussed, for example, by Douglas Quadling in [5]. The two-sided form of this formula is so that a scalar placed in the same formula is left unchanged as

$$s' = e^{-I_2\theta/2} s e^{I_2\theta/2} = e^{-I_2\theta/2} e^{I_2\theta/2} s = e^0 s = s.$$

Hence the same rotation operator can be used on full multi-vectors, but it only rotates the vector part. The grade projectors $\langle\rangle_k : G^2 \mapsto G^2$ are the maps

$$\langle A\rangle_0 = a_0,\ \ \langle A\rangle_1 = a_1e_1 + a_2e_2,\ \ \langle A\rangle_2 = a_{12}I_2$$

which extract the scalar, vector and bi-vector portion of a multi-vector, respectively. These are the geometric extensions to the real- and imaginary-part operators in complex numbers. Two involutions are of interest. First is the vector conjugation, denoted here with an over-arrow, which is defined as

$$\overrightarrow{A} = (a_0 + a_{12}I_2) - (a_1e_1 + a_2e_2)$$

The second is the bi-vector conjugation, denoted with an over-bar, and is defined as

$$\bar{A} = (a_0 - a_{12}I_2) + (a_1e_1 + a_2e_2)$$

which is exactly the complex conjugate when the multi-vector is from the complex sub-algebra. These involutions allow us to write the following identities for the grade projectors

$$\begin{aligned} &\langle A\rangle_0 = \tfrac{1}{2}\left(\vec{A} + \bar{A}\right),\ \langle A\rangle_1 = \tfrac{1}{2}\left(A - \vec{A}\right),\ \text{and} \\ &\langle A\rangle_2 = \tfrac{1}{2}\left(A - \bar{A}\right). \end{aligned} \tag{2}$$

We have seen that an arbitrary multi-vector can be written as either a direct sum of a scalar, a vector and a bi-vector, or as a direct sum of a spinor and a vector. Much like the quaternions, there is yet another alternative; the symplectic form. Starting with the spinor-vector form $A = s_0 + v$ where $s_0 = a_0 + a_{12}I_2$ and $v = a_1e_1 + a_2e_2$, we may factor the vector into a unit vector and a spinor in four ways: $v = a_1e_1 + a_2e_2$, $v = (v_1 - v_2I_2)\,e_1 = \bar{s}_1e_1$, $v = (v_1I_2 + v_2)\,e_2 = s_2e_2$, and $v = e_2\,(-v_1I_2 + v_2) = e_2\bar{s}_2$. So that the factors are related as

$$v = e_1s_1 = \bar{s}_1e_1 = s_2e_2 = e_2\bar{s}_2.$$

This allows any arbitrary multi-vector to be written in multiple symplectic forms, e.g., as

$$A = s_0 + e_1s_1 = s_0 + s_2e_2 = s_0 + \bar{s}_1e_1 = s_0 + e_2\bar{s}_2.$$

These multiple choices of factoring the same vector are used extensively to simplify equations. For a complete treatment of G^2 as a system for plane geometry analysis see Calvet [6].

3. MULTI-VECTOR COLOR IMAGES

A color image, f, composed of the red-green-blue triplet $\{r, g, b\}$ can be split into the vector chromaticity image and scalar luminance image using a technique similar to that described by Thornton and Sangwine [1]. Namely, let the vector chromatic image be encoded as

$$v = \left[\tfrac{1}{2}r\,(g+b)\,e_1 + \tfrac{\sqrt{3}}{2}\,(g-b)\,e_2\right] = v_1e_1 + v_2e_2$$

And let the scalar luminance image be given by

$$L = \tfrac{1}{3}\,(r+g+b)$$

Hence the color image, f, is given in G^2 as the multi-vector

$$f = L + v, \quad L \in \mathbb{R}, v \in \mathbb{R}^2$$

Decoding the image back into the rgb-triplet is given as

$$r = L + \tfrac{2}{3}v_1, \quad g = L + \tfrac{1}{\sqrt{3}}v_2 - \tfrac{1}{3}v_1, \quad b = L - \tfrac{1}{\sqrt{3}}v_2 - \tfrac{1}{3}v_1.$$

Although the basic algebra is the same, this encoding is different from that of Thornton and Sangwine in that we have not created a complex chromaticity image and a separate gray-scale image. Instead we have created a vector chromaticity image added to a scalar luminance image. This addition is possible as a multi-vector in the geometric algebra G^2. There is no comparable notion of adding a scalar to a complex without inter-mixing the real-part of the complex image with the scalar image. Obviously, we could have used any luminance-chrominance image decomposition of the rgb-triplet; XYZ, YUV, YIQ, etc. For example, the YUV space image with Y mapped into the scalar and UV mapped into the vector part of a multi-vector would work also. This would then allow us to include the work of McCabe, et al. [2] on spatio-chromatic image processing into our framework.

The encoding of the color image into a scalar + vector multi-vector without the use of the complex sub-algebra is a key step in this work. This allows us to separate operator from operand in the hyper-complex Fourier transform later.

4. COLOR EDGE DETECTION FILTERS

The classic Prewitt filter can be extended in a number of ways using left and right convolution masks with multi-vector coefficients taken from G^2. The following masks, for example, are used to detect horizontal edges in an image f. Each mask filters the luminance and chrominance portions differently (the key to building these filters is to note that -1 can be factored in two ways: $-1 = (+1)(-1) = (I_2)(I_2)$). The following icons depict these filters

$$\mathcal{H}_1 : \frac{1}{6}\begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}\begin{bmatrix} f \end{bmatrix}\begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}$$

$$\mathcal{H}_2 : \frac{1}{6}\begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ I_2 & I_2 & I_2 \end{bmatrix}\begin{bmatrix} f \end{bmatrix}\begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -I_2 & -I_2 & -I_2 \end{bmatrix}$$

$$\mathcal{H}_3 : \frac{1}{6}\begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ I_2 & I_2 & I_2 \end{bmatrix}\begin{bmatrix} f \end{bmatrix}\begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ I_2 & I_2 & I_2 \end{bmatrix}$$

To understand the filter operation examine the convolution centered on the transition between two horizontal color regions, c_1, and c_2 where $c_i = s_i + v_i$, $\ s_i \in \mathbb{R}$, $v_i \in \mathbb{R}^2$, $i = 1, 2$. Filtered outputs for the each mask at the transition $c_1 \to c_2$ are given by

$$\begin{aligned} \mathcal{H}_1\,[c_1 \to c_2] &= \tfrac{1}{2}\,(c_1 - c_2) &&= \tfrac{1}{2}\,(s_1 - s_2) + \tfrac{1}{2}\,(v_1 - v_2) \\ \mathcal{H}_2\,[c_1 \to c_2] &= \tfrac{1}{2}\,(c_1 - I_2c_2I_2) &&= \tfrac{1}{2}\,(s_1 + s_2) + \tfrac{1}{2}\,(v_1 - v_2) \\ \mathcal{H}_3\,[c_1 \to c_2] &= \tfrac{1}{2}\,(c_1 + I_2c_2I_2) &&= \tfrac{1}{2}\,(s_1 - s_2) + \tfrac{1}{2}\,(v_1 + v_2) \end{aligned}$$

$\mathcal{H}_1$ detects edges in both the luminance and chrominance channels. $\mathcal{H}_2$ smoothes (averages) the luminance but edge detects chrominance. In contrast $\mathcal{H}_3$ smoothes the chrominance but edge detects in luminance. Hence the chromin-edge is given as

$$\langle \mathcal{H}_1 \rangle_1 = \tfrac{1}{2}(v_1 - v_2) = \langle \mathcal{H}_2 \rangle_1$$

and the lumen-edge is given as

$$\langle \mathcal{H}_1 \rangle_0 = \tfrac{1}{2}(s_1 - s_2) = \langle \mathcal{H}_3 \rangle_0 .$$

These grade-projection operators can be replaced with their algebraic equivalents from equation (2). For chrominance edge detectors $\mathcal{H}_1$ and $\mathcal{H}_2$ where the chrominance transition does not change, the result is zero. Where the chrominance transition consists of opposing colors, the resulting edge is maximal and matches the starting chrominance value v_1. This result is comparable to the quaternion based edge detection filter described in [3]. In that work the mask consisted of quaternion rotation operators whose rotation axis was the gray-line and the rotation angle was π radians. The action of the $\mathcal{H}_2$ filter is similar but the response for a maximal transition is different. The other two filters are completely new. The Sobel and Kirsch filters can be generalized in the same way. The type-2 Sobel filter, $\mathcal{H}_{\text{Sobel},2}$, is given by

$$\frac{1}{8}\begin{bmatrix} 1 & \sqrt{2} & 1 \\ 0 & 0 & 0 \\ I_2 & I_2 & I_2 \end{bmatrix}\begin{bmatrix} f \end{bmatrix}\begin{bmatrix} 1 & \sqrt{2} & 1 \\ 0 & 0 & 0 \\ -I_2 & -I_2 & -I_2 \end{bmatrix}$$

And the type-2 Kirsch filter, $\mathcal{H}_{\text{Kirsch},2}$, is given by

$$\begin{bmatrix} \sqrt{3}I_2 & \sqrt{5} & \sqrt{5} \\ \sqrt{3}I_2 & 0 & \sqrt{5} \\ \sqrt{3}I_2 & \sqrt{3}I_2 & \sqrt{3}I_2 \end{bmatrix}\begin{bmatrix} f \end{bmatrix} \times \begin{bmatrix} -\sqrt{3}I_2 & \sqrt{5} & \sqrt{5} \\ -\sqrt{3}I_2 & 0 & \sqrt{5} \\ -\sqrt{3}I_2 & -\sqrt{3}I_2 & -\sqrt{5}I_2 \end{bmatrix}$$

As before for the Prewitt-type filters, the luminance- and chrominance-only filters are given by the 0-grade and 1-grade projectors of the filter outputs.

Like gray-scale edge-detection filters, these color filters provide the location of an edge and, using intensity, the sharpness of the edge, but they also provide, via the response color, the relative orientation of the colors in the transition. This additional information can be used to refine color image segmentation processes.

5. GEOMETRIC FOURIER TRANSFORM (GFT)

This section defines the Geometric Algebra extensions to the Fourier transform. Then, using a decomposition (split) operation, it will be shown that specialized transformation code is unnecessary since all the defined transforms can be encoded using a series of standard complex Fourier transforms.

Definition (Fourier Transforms) The four forms of Geometric Fourier transforms are given by

$$\begin{aligned} \mathcal{F}^{\pm L}[f(n,m)] &\triangleq \frac{1}{k}\sum_{m=0}^{M-1}\sum_{n=0}^{N-1} e^{\mp 2\pi I_2\left(\frac{mv}{M}+\frac{nu}{N}\right)} f(m,n) \\ &= F^{\pm L}[v,u] \\ \mathcal{F}^{\pm R}[f(n,m)] &\triangleq \frac{1}{k}\sum_{m=0}^{M-1}\sum_{n=0}^{N-1} f(m,n)\, e^{\mp 2\pi I_2\left(\frac{mv}{M}+\frac{nu}{N}\right)} \\ &= F^{\pm R}[v,u] \end{aligned} \tag{3}$$

with their corresponding inverses given by changing the sign of the exponential and summing over u and v. The scale factor $k = \sqrt{MN}$. The transforms with positive exponent are referred to as the forward transforms and the negative exponent forms are the reverse transform.

Starting with the image function in symplectic form as $f(n,m) = s_0(n,m) + e_1 s_1(n,m)$, and because the Fourier transform is a linear operator, we may split the right-handed transform across the two spinors as

$$\mathcal{F}^{\pm R}[f(n,m)] = \mathcal{F}^{\pm R}[s_0(n,m)] + e_1\mathcal{F}^{\pm R}[s_1(n,m)]$$

Spinors are equivalent to complex numbers, so the standard complex Fourier transform can be used to write the right-sided transform as

$$F^{\pm R}[u,v] = S_0[u,v] + e_1 S_1[u,v]$$

where $S_k[u,v]$ is given by the standard complex Fourier transform, $\mathcal{F}_{\mathbb{C}}[]$, via the sequence of mappings

$$S_k[u,v] = \mathcal{F}_{\mathbb{C}}\left[s_k(n,m)\big|_{I_2\to i}\right]\Big|_{i\to I_2} \tag{4}$$

where $i = \sqrt{-1}$ is the standard complex operator. This equation says that the spinor is first rewritten as a complex number, then complex Fourier transformed, and finally the resulting complex spectrum is converted back into a spinor spectrum. The left-sided transform is likewise given as

$$F^{\pm L}[u,v] = S_0[u,v] + S_2[u,v]\, e_2$$

which is found by factoring the vector part as $v(n,m) = s_2(n,m)\, e_2$ and then proceeding through the same complex Fourier transform mapping sequence given in (4). Note that s_1 and s_2 are different; the real and imaginary parts of the spinor have swapped roles. The explicit relationships between s_1 and s_2 are: $s_2 = \bar{s}_1 I_2$ and $s_1 = \bar{s}_2 I_2$. The inverse transforms are done by writing the spectra in symplectic form and inverting (4) so that the inverse complex Fourier transform maps the components back into the spatial domain.

6. CONVOLUTION OPERATIONAL FORMULAE

Owing to the non-commutative multiplication of multi-vectors there are three general convolution operators available.

Definition (Convolution Operators) The left-, right- and bi-convolution operators are defined, respectively, as:

$$\begin{aligned} h_L \circ f &\triangleq \sum_{i=0}^{N-1}\sum_{j=0}^{M-1} h_L(i,j)\, f(n-i, m-j) \\ f \circ h_R &\triangleq \sum_{i=0}^{N-1}\sum_{j=0}^{M-1} f(n-i, m-j)\, h_R(i,j) \\ h_L \prec f \succ h_R &\triangleq \sum_{i=0}^{N-1}\sum_{j=0}^{M-1} h_L(i,j)\, f(n-i, m-j)\, h_R(i,j) \end{aligned}$$

where f is the input image and h_L and h_R are convolution masks. Using symplectic forms the bi-convolution equation can always be converted to a handed-convolution. We demonstrate this by focusing on the left-most terms in the bi-convolution equation to convert the equation to a right-handed convolution. Starting with the left-mask and image function write each in symplectic form as $h_L = h_0 + e_1 h_1$ and $f = f_0 + f_2 e_2$, then their product is given by

$$\begin{aligned}
h_L f &= (h_0 + e_1 h_1)(f_0 + f_2 e_2) \\
&= h_0 f_0 + e_1 (h_1 f_0) + (h_0 f_2) e_2 + e_1 (h_1 f_2) e_2 \\
&= f_0 h_0 + e_1 f_0 h_1 + f_2 h_0 e_2 + e_1 f_2 (h_1 e_2) \\
&= f_0 h_0 + e_1 f_0 h_1 + f_2 (h_0 e_2) + e_1 (f_2 e_2) \bar{h}_1 \\
&= f_0 h_0 + e_1 f_0 h_1 + (f_2 e_2) \bar{h}_0 + (e_1 e_1) f_1 \bar{h}_1 \\
&= f_0 h_0 + e_1 f_0 h_1 + e_1 f_1 \bar{h}_0 + f_1 \bar{h}_1
\end{aligned}$$

where at each step the terms in the parentheses are modified on the next line. Hence the triple product of the bi-convolution is given by

$$h_L \, f \, h_R = f_0 (h_0 h_R) + e_1 f_1 \left(\bar{h}_0 h_R\right) + f_1 \left(\bar{h}_1 h_R\right) + e_1 f_0 (h_1 h_R)$$

Notice that the left mask terms have been commuted with the image function. Each of these terms, when placed back into the bi-convolution equation is of the form of the right-handed convolution equation so that

$$\begin{aligned}
h_L \prec f \succ h_R = {} & f_0 \circ (h_0 h_R) + e_1 f_1 \circ \left(\bar{h}_0 h_R\right) \\
& + f_1 \circ \left(\bar{h}_1 h_R\right) + e_1 f_0 \circ (h_1 h_R)
\end{aligned}$$

Note that this equation only includes four terms, much like the involution based quaternion version of the same formula. What remains for us to determine is the spectral form of the handed-convolution equation. Before doing this let's re-examine the Prewitt filter to check these results. By definition $\mathcal{H}_2$ has

$$h_L = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ +I_2 & +I_2 & +I_2 \end{bmatrix}, \quad h_R = \frac{1}{6} \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -I_2 & -I_2 & -I_2 \end{bmatrix}$$

so that $h_0 = h_L$ and $h_1 = 0$ which reduces the bi-convolution to two terms, i.e., $h_L \prec f \succ h_R = f_0 \circ (h_0 h_R) + e_1 f_1 \circ \left(\bar{h}_0 h_R\right)$. Combining the mask terms we obtain

$$\begin{aligned}
h_0 h_R &= \frac{1}{6} \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ +I_2 & +I_2 & +I_2 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -I_2 & -I_2 & -I_2 \end{bmatrix} \\
&= \frac{1}{6} \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}
\end{aligned}$$

and

$$\begin{aligned}
\bar{h}_0 h_R &= \frac{1}{6} \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -I_2 & -I_2 & -I_2 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -I_2 & -I_2 & -I_2 \end{bmatrix} \\
&= \frac{1}{6} \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}
\end{aligned}$$

hence the image's luminance scalar-part, f_0, is smoothed and the image's chrominance vector-part, $e_1 f_1$, is differenced which is consistent with our earlier direct analysis of this filter.

The first step in writing the spectral form of the convolution operator is to prove spatial shift formulas. If an image, $f(n, m)$, is shifted as $f(n + n_0, m + m_0)$, then the two left-sided Fourier transforms (forward and reverse) are given by:

$$\mathcal{F}^{\pm L} \left[f (m + m_0, n + n_0) \right] = e^{\pm I_2 2\pi \left(\frac{m_0 v}{M} + \frac{n_0 u}{N} \right)} F^{\pm L} [v, u]$$

And the right-transforms yield:

$$\mathcal{F}^{\pm R} \left[f (m + m_0, n + n_0) \right] = F^{\pm R} [v, u] \, e^{\pm I_2 2\pi \left(\frac{m_0 v}{M} + \frac{n_0 u}{N} \right)}$$

Using the Geometric Fourier transform, the shifted transform and one-sided convolution definitions, the spectral-domain formula can be derived as:

$$\begin{aligned}
\mathcal{F}^{\pm L} \left[f \circ h (n, m) \right] = \sqrt{MN} \Big(& F_0^{\pm L} [v, u] \, H^{\pm L} [u, v] \\
& + e_1 F_1^{\pm L} [v, u] \, H^{\mp L} [u, v] \Big)
\end{aligned}$$

where $F[v, u]$ is written in symplectic form as

$$F^{\pm L} [v, u] = F_0^{\pm L} [v, u] + e_1 F_1^{\pm L} [v, u] \, .$$

The right-hand convolution follows a similar formula. Note the use of both the forward and reverse left transform on the mask.

7. CONCLUSIONS

The representation of color pixels as multi-vectors in the geometric algebra G^2 provides the starting point for a system of holistic image filters. In particular a G^2 based linear convolution equation was shown useful in extending standard scalar image filters, e.g., the Prewitt edge-detector. By leaving the pseudo-scalar out of the color encoding a simple Fourier transform was demonstrated and used to derive spectral versions of the convolution equations. This work focuses on the mechanics of the G^2 algebra as a tool for color image processing, but leaves unexplored the geometric insights that can be gained and applied in the design of new classes of filters. Also untouched are higher dimensional geometric algebras which will lend themselves to multi-spectral image filtering and analysis techniques.

8. REFERENCES

[1] A. L. Thornton and S. J. Sangwine, "Colour object recognition using complex coding in the frequency domain," in *Proceedings 5th International Conference on Image Processing and its Applications*, Heriot-Watt University, Edinburgh, UK, 3–6 July 1995, pp. 820–824, Institution of Electrical Engineers, IEE Conference Publication 410.

[2] A. McCabe, T. Caelli, G. West, and A. Reeves, "Encoding and processing spatio-chromatic image information using complex fourier transform methods," Tech. Rep. 4, Curtin University of Technology, Perth, Western Australia, June 1997.

[3] S. J. Sangwine and T. A. Ell, "Colour image filters based on hypercomplex convolution," *IEE Proceedings – Vision, Image and Signal Processing*, vol. 147, no. 2, pp. 89–93, Apr. 2000.

[4] T. A. Ell and S. J. Sangwine, "Hypercomplex Fourier transforms of color images," *IEEE Transactions on Signal Processing*, vol. 16, no. 1, pp. 22–35, Jan. 2007.

[5] D. Quadling, "Q is for quaternion," *Mathematical Gazette*, vol. 63, no. 424, pp. 98–110, June 1979.

[6] R. G. Calvet, *Treatise of plane geometry through the geometric algebra*, To be published in 2007, ISBN 84-699-3197-0, 2007, Online version at www.xtec.es/˜rgonzal1/treatise.htm.