12-1-2004

Discrete-Time Fractional Differentiation from Integer Derivatives

Hany Farid
Dartmouth College

Discrete-Time Fractional Differentiation from Integer Derivatives

Hany Farid

Abstract

Discrete-time fractional derivative filters (1-D and 2-D) are shown to be well approximated from a small set of integer derivatives. A fractional derivative of arbitrary order (and, in 2-D, of arbitrary orientation) can therefore be efficiently computed from a linear combination of integer derivatives of the underlying signal or image.

I. Introduction

The theory of non-integer order (fractional) derivatives dates back to Leibniz's correspondence with L'Hospital in 1695 [1]. For many decades afterwards, the theory of fractional derivatives was developed primarily as a theoretical field of mathematics (see [2], [3], [4], [5] for general surveys). More recently, however, this branch of mathematics has found applications in a number of different areas ranging from control theory [6] to electrochemistry [7] to neuronal modeling [8], and more – see [9] for both a general theoretical survey and more applications of fractional differential equations.

As fractional derivatives find more applications, it is natural to consider the design of efficient and accurate discrete-time filters The design of discrete fractional derivative filters are described in [10], where the author considers a Taylor series expansion of the underlying function to be differentiated. In [11], the authors take a similar approach by replacing the Taylor series expansion with a Newton series. See also [12] for general numerical methods.

In this paper, I will describe how discrete-time integer derivatives can be used to efficiently compute fractional derivatives of any order. Specifically, for one-dimensional signals, a small linear basis of integer derivatives is shown to approximately span the space of fractional derivatives of arbitrary order. For two-dimensional images, a linear basis is shown to approximately span the space of fractional directional derivatives of arbitrary order and orientation. This approach places no explicit constraints on the form of the underlying signal/image to be differentiated.

II. Fractional Derivatives

Fractional differentiation is formulated in the Fourier domain, for both discrete-time one-dimensional signals and two-dimensional images. This formulation is particularly convenient with respect to the design of discrete-time fractional derivative filters.

H. Farid, 6211 Sudikoff Lab, Computer Science Department, Dartmouth College, Hanover, NH 03755 USA (email: farid@cs.dartmouth.edu; tel/fax: 603.646.2761/603.646.1672). This work was supported by an Alfred P. Sloan Fellowship, a National Science Foundation CAREER Award (IIS-99-83806), and a departmental National Science Foundation Infrastructure Grant (EIA-98-02068).

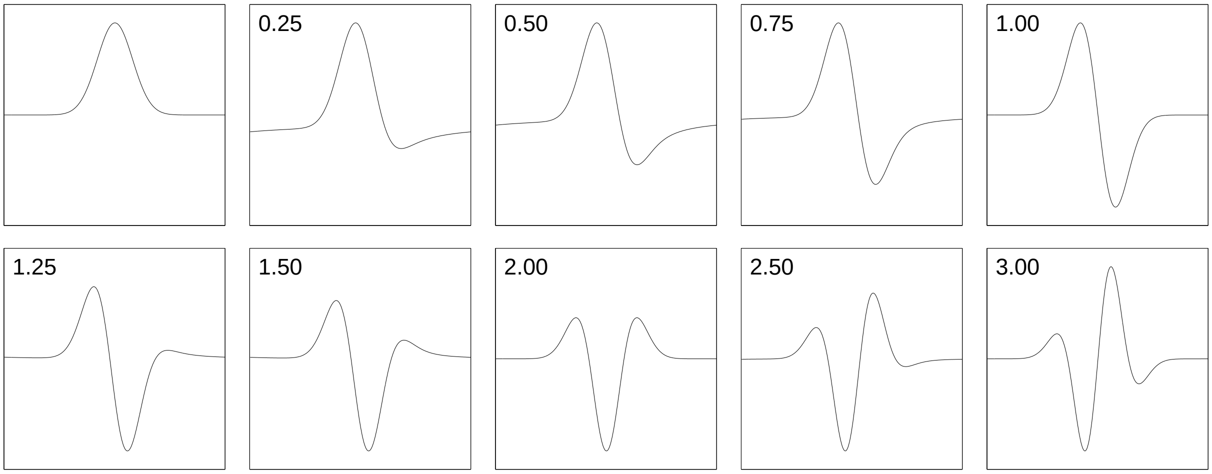

Fig. 1: Fractional and integer derivatives of a 1-D Gaussian (top-left panel). For display, each derivative is normalized to a common scale.

A. One-Dimensional

The Fourier series of a one-dimensional discrete-time function $f[x]$ of length l is given by:

$$f[x] \quad = \quad \frac{1}{l}\sum_{\omega=0}^{l-1} F[\omega]e^{i\omega x/\sigma}, \tag{1}$$

where $\sigma = \frac{l}{2\pi}$, and $F[\omega] = \sum_{x=0}^{l-1} f[x]e^{-i\omega x/\sigma}$ is the discrete Fourier transform. Differentiation in the Fourier domain takes on a particularly simple form (there are, of course, many complicating issues in designing compact and accurate derivative kernels that are not addressed here – see [13] for a discussion of these issues). The first-order derivative, for example, is given by:

$$f^{(1)}[x] \quad = \quad \frac{1}{l}\sum_{\omega=0}^{l-1} \tfrac{i\omega}{\sigma}F[\omega]e^{i\omega x/\sigma}. \tag{2}$$

The n^{th}-order derivative is given by:

$$f^{(n)}[x] \quad = \quad \frac{1}{l}\sum_{\omega=0}^{l-1} \left(\tfrac{i\omega}{\sigma}\right)^n F[\omega]e^{i\omega x/\sigma}. \tag{3}$$

With n taking on any real value, Eq. (3) defines both the integer and fractional derivative. With this formulation, a discrete-time filter, $h[x]$, can be designed by simply inverse Fourier transforming $H(\omega) = (i\omega/\sigma)^n$. The (regularized) fractional or integer derivative of a signal $f[x]$ is then given by $g[x] = f[x] \star h[x]$, where $\star$ denotes the convolution operator. Shown in Fig. 1, for example, are fractional and integer derivatives of a 1-D Gaussian computed in this manner.

B. Two-Dimensional

The Fourier series of a two-dimensional discrete-time function $f[x,y]$, of size $l \times l$, is given by:

$$f[x,y] \quad = \quad \frac{1}{l^2}\sum_{\omega_x=0}^{l-1}\sum_{\omega_y=0}^{l-1} F[\omega_x,\omega_y]e^{i(\omega_x x+\omega_y y)/\sigma}, \tag{4}$$

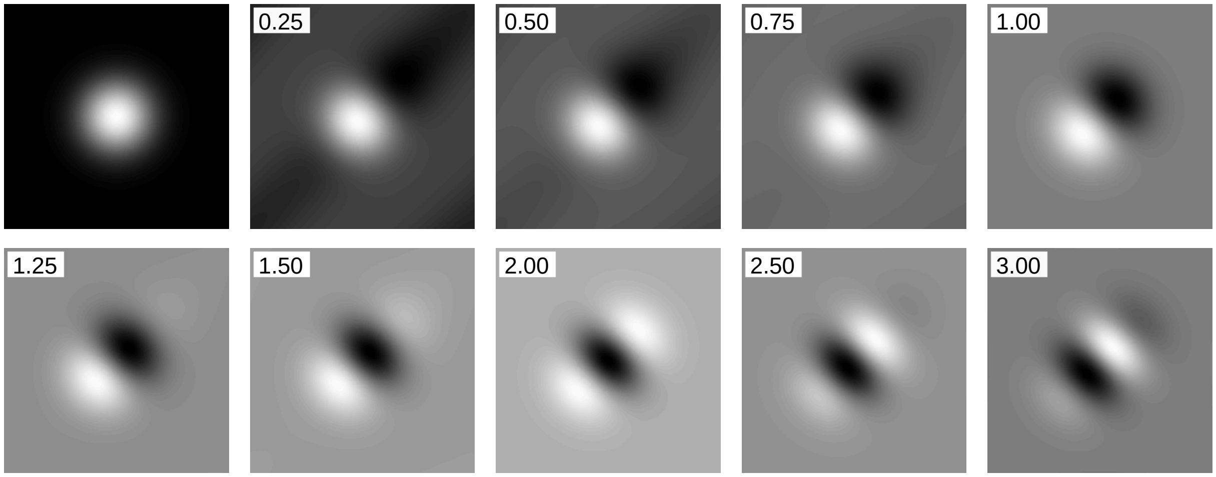

Fig. 2: Fractional and integer derivatives of a 2-D Gaussian (top-left panel) oriented at $\theta = \pi/4$. For display, each derivative is normalized to a common scale.

where $\sigma = \frac{l}{2\pi}$, and $F[\omega_x, \omega_y] = \sum_{x=0}^{l-1} \sum_{y=0}^{l-1} f[x,y] e^{-i(\omega_x x + \omega_y y)/\sigma}$ is the discrete Fourier transform. The first-order directional derivative in the horizontal, $f_0^{(1)}[x,y]$, and vertical, $f_{\pi/2}^{(1)}[x,y]$, directions are given by:

$$f_0^{(1)}[x,y] \quad = \quad \frac{1}{l^2} \sum_{\omega_x=0}^{l-1} \sum_{\omega_y=0}^{l-1} \frac{i\omega_x}{\sigma} F[\omega_x, \omega_y] e^{i(\omega_x x + \omega_y y)/\sigma}, \tag{5}$$

and

$$f_{\pi/2}^{(1)}[x,y] \quad = \quad \frac{1}{l^2} \sum_{\omega_x=0}^{l-1} \sum_{\omega_y=0}^{l-1} \frac{i\omega_y}{\sigma} F[\omega_x, \omega_y] e^{i(\omega_x x + \omega_y y)/\sigma}. \tag{6}$$

The n^{th}-order derivative at an arbitrary orientation, θ, is given by:

$$f_\theta^{(n)}[x,y] \quad = \quad \frac{1}{l^2} \sum_{\omega_x=0}^{l-1} \sum_{\omega_y=0}^{l-1} (i[\cos(\theta)\omega_x + \sin(\theta)\omega_y]/\sigma)^n F[\omega_x, \omega_y] e^{i(\omega_x x + \omega_y y)/\sigma}. \tag{7}$$

With n taking on any real value, Eq. (7) defines both the integer and fractional directional derivative. As in the one-dimensional case, a discrete-time filter, $h[x,y]$, can be designed by simply inverse Fourier transforming $H(\omega_x, \omega_y) = (i[\cos(\theta)\omega_x + \sin(\theta)\omega_y]/\sigma)^n$. The (regularized) fractional or integer derivative of an image $f[x,y]$ is then given by $g[x,y] = f[x,y] \star h[x,y]$, where $\star$ denotes the convolution operator. Shown in Fig. 2, for example, are fractional and integer derivatives of a 2-D Gaussian.

III. Fractional Differentiation from Integer Derivatives

A. One-Dimensional

Let $h[x]$ be a lowpass filter, and $h^{(n)}[x]$ denote its n^{th}-order derivative. We ask, can an arbitrary fractional derivative filter be approximated from a linear combination of the first N integer derivatives, a lowpass filter, and a constant:

$$h^{(n)}[x] \quad \approx \quad \sum_{k=1}^{N} \alpha_k h^{(k)}[x] + \alpha_{N+1} h[x] + \alpha_{N+2} c[x], \tag{8}$$

where $c[x] = 1$. Note that, unlike a Taylor series expansion, $h^{(n)}[x]$ is being expressed with respect to integer derivatives of the underlying function $h[x]$, and not derivatives of $h^{(n)}[x]$. Re-writing the above in matrix form yields:

$$\vec{h}^{(n)} \;=\; B\vec{\alpha}, \tag{9}$$

where the columns of the matrix B contain the basis filters, $\vec{h}^{(1)}, \ldots, \vec{h}^{(N)}$, $\vec{h}$, $\vec{c}$, and the vector $\vec{\alpha} = \begin{pmatrix} \alpha_1 & \ldots & \alpha_{N+2} \end{pmatrix}$. The unknown basis weights, $\vec{\alpha}$, are determined using standard least-squares to yield $\vec{\alpha} = (B^t B)^{-1} B^t \vec{h}^{(n)}$.

Consider now convolving an arbitrary signal $f[x]$ with each of the filters in the basis set:

$$g^{(1)} = f \star h^{(1)} \quad \ldots \quad g^{(N)} = f \star h^{(N)}, \qquad g_h = f \star h, \qquad g_c = f \star c, \tag{10}$$

where, for notational convenience, the spatial parameter, $[x]$, is dropped. Given the linearity of the convolution operator, convolution with an arbitrary filter, $h^{(n)}[x]$, can be computed from a linear combination of the above filter responses:

$$g^{(n)}[x] \;=\; \sum_{k=1}^{N} \alpha_k g^{(k)}[x] + \alpha_{N+1} g_h[x] + \alpha_{N+2} g_c[x]. \tag{11}$$

While this formulation obviates the need to convolve the signal $f[x]$ with the desired filter $h^{(n)}[x]$, it does require a least-squares solution, for a given n, to obtain the scalar weights $\vec{\alpha}$, Eq. (9). As I will describe below, however, for a specified set of derivative filters, a closed-form approximation for $\vec{\alpha}$, as a function of n, can be determined.

1) Gaussian-Based Filters: Consider a 1-D Gaussian filter $h[x] = e^{-x^2}$ with unit variance. Denote the n^{th}-order derivative as $h^{(n)}[x]$. Shown in the upper panel of Fig. 3 (filled circles) are 11-tap fractional and integer differentiating filters synthesized by inverse Fourier transforming $(i\omega/\sigma)^n H(w)$, Eq. (3), where $H(\omega)$ is the Fourier transform of $h[x]$. Also shown in this figure (open circles) are the same differentiating filters synthesized from a linear combination of five basis filters (first- through third-order derivatives plus a lowpass and constant function), Eq. (8). Notice that the original and synthesized filters are nearly identical – there is, of course, no error in synthesizing the integer derivatives. Shown in the lower panel of Fig. 3 is the $1/2$-order derivative filter approximated with a basis set of size two through six.

Shown in Fig. 4 are the scalar weights, $\vec{\alpha}$, as a function of derivative order, n, for the filters shown in the upper panel of Fig. 3. Note that these coefficients are well fit with a fifth-order polynomial (solid lines). These functions provide the required coefficients for estimating an arbitrary fractional derivative from a linear combination of integer derivatives, Eq. (11).

Fractional derivatives can be approximated from filters other than the Gaussian-based filters. For example, we consider a truncated sinc filter with a boxcar impulse response, the lowpass filter with a raised cosine impulse response, $H[\omega] = 1/2(1 + \cos(\omega))$, and the derivative filters described in [14]. In each case 11-tap fractional and integer differentiating filters were synthesized as described above. The same differentiating filters were synthesized from a linear combination of the corresponding first- through third-order derivatives plus a lowpass and constant function, Eq. (8). The average root mean square (RMS) error between the actual and approximate filters of order 0.1 to 3.0 (in steps of 0.1) was 0.34 for the sinc-based filters, 0.20 for the cosine filters, 0.12 for the filters of [14] – the Gaussian-based filters have an RMS error of 0.18. As with the Gaussian filters, these other filters are reasonably well approximated with an integer basis, with the sinc-based filters having a slightly higher error.

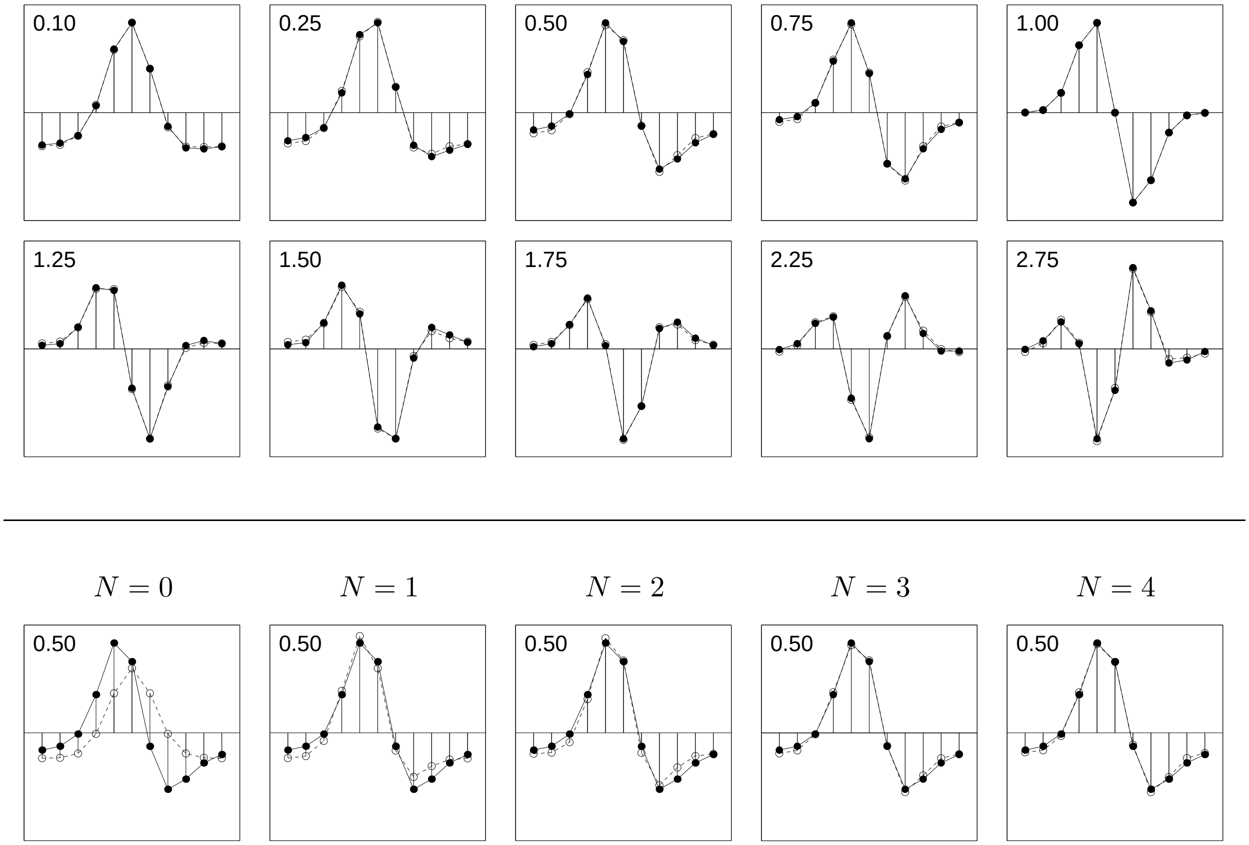

Fig. 3: Shown in the upper panel are 11-tap derivative filters of varying orders (filled circles) and synthesized filters (open circles) from a basis set of size five ($N = 3$). Shown in the lower panel is the $1/2$-order derivative along with synthesized filters with a basis set of size two ($N = 0$) to six ($N = 4$). For display, each derivative order is normalized to a common scale.

B. Two-Dimensional

For integer derivatives, the n^{th}-order directional derivative can be synthesized at any orientation from a linear combination of 1^{st}- through n^{th}-order derivatives (within the Computer Vision community, this property of directional derivatives has been termed steerability [15]). [1] Let $h[x, y]$ be a lowpass filter. The first-order directional derivative at an arbitrary orientation θ can be expressed as a linear combination of the axis derivatives:

$$h_{\theta}^{(1)}[x, y] \quad = \quad \cos(\theta) h_0^{(1)}[x, y] + \sin(\theta) h_{\pi/2}^{(1)}[x, y]. \tag{12}$$

Similarly, the second-order directional derivative is given by:

$$h_{\theta}^{(2)}[x, y] \quad = \quad \cos^2(\theta) h_0^{(2)}[x, y] - 2\cos(\theta)\sin(\theta) h_0^{(1)}[x, y] h_{\pi/2}^{(1)}[x, y] + \sin^2(\theta) h_{\pi/2}^{(2)}[x, y]. \tag{13}$$

[1]While the integer directional derivatives are steerable, the fractional derivatives are not. Consider the Fourier-based formulation of directional differentiation, Eq. (7). For integer values of the derivative order, n, the impulse response $H(\omega_x, \omega_y) = (i[\cos(\theta)\omega_x + \sin(\theta)\omega_y]/\sigma)^n$ can expanded into a finite number of terms, each corresponding to an integer derivative. With non-integer values of n, however, the same expansion yields an infinite number of terms. In practical terms, the fractional derivative is, therefore, not steerable. Note also that, unlike the integer derivatives, the two-dimensional fractional derivatives are not $x - y$ separable.

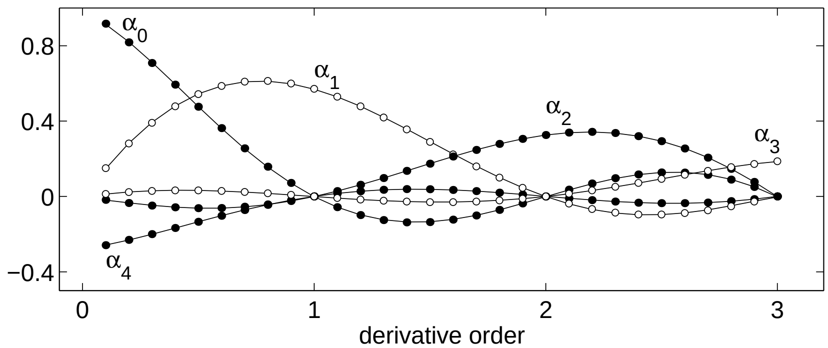

Fig. 4: Shown are the scalar weights, $\alpha_0, \ldots, \alpha_4$, Eq. (11), for the one-dimensional 11-tap filters of Fig. 3 (upper panel). The solid lines correspond to fifth-order polynomial fits to the data points that are computed in $1/10$ increments between a $1/10^{th}$- and 3^{rd}-order derivative.

Given this property of directional derivatives, we now ask, similar to the one-dimensional case, can an arbitrary derivative filter be approximated from a linear combination of integer derivatives, a lowpass filter and a constant:

$$h_\theta^{(n)}[x,y] \approx \sum_{k=1}^{N}\left(\sum_{l=0}^{k}\sum_{m=0}^{k}\alpha_{k,l,m} h_0^{(l)}[x,y] h_{\pi/2}^{(m)}[x,y]\right) + \alpha_{\hat{N}+1} h[x,y] + \alpha_{\hat{N}+2} c[x,y], \tag{14}$$

where the inner pair of summations is over all l and m such that $l+m=k$, $h_0^{(0)}[x,y] = h_{\pi/2}^{(0)}[x,y] = 1$, $c[x,y]=1$, and $\hat{N} = N + N(N+1)/2$. By reshaping each 2-D filter into a 1-D vector, the above can be expressed in matrix form as follows:

$$\vec{h}_\theta^{(n)} = B\vec{\alpha}, \tag{15}$$

where the columns of the matrix B contain the basis filters, Eq. (14), and $\vec{\alpha}$ contains the scalar weights, Eq. (14). For notational convenience, the basis filters will be denoted as $\vec{h}_1 \ldots \vec{h}_{\hat{N}}$, $\vec{h}$, $\vec{c}$, and the scalar weights as $\vec{\alpha} = \left(\alpha_1 \ldots \alpha_{\hat{N}+2}\right)$. As in the one-dimensional case, the basis weights are given by $\vec{\alpha} = (B^t B)^{-1} B^t \vec{h}_\theta^{(n)}$.

Consider now convolving an arbitrary image $f[x,y]$ with each of the filters in the basis set:

$$g_1 = f \star h_1 \quad \ldots \quad g_{\hat{N}} = f \star h_{\hat{N}}, \qquad g_h = f \star h, \qquad g_c = f \star c, \tag{16}$$

where, for notational convenience, the spatial parameters, $[x,y]$, are dropped. Given the linearity of the convolution operator, convolution with an arbitrary filter, $h_\theta^{(n)}[x,y]$, can be computed from a linear combination of the above filter responses:

$$g_\theta^{(n)}[x,y] = \sum_{k=1}^{\hat{N}} \alpha_k g_k[x,y] + \alpha_{\hat{N}+1} g_h[x,y] + \alpha_{\hat{N}+2} g_c[x,y]. \tag{17}$$

As I will describe below, for a specified set of derivative filters, a closed-form approximation for $\vec{\alpha}$, as a function of n and θ, can be determined.

1) Gaussian-Based Filters: Consider a 2-D Gaussian filter $h[x,y] = e^{-(x^2+y^2)}$ with unit variance along both axes. Denote the n^{th}-order directional derivative as $h_\theta^{(n)}[x,y]$. Shown in Fig. 5 are 11-tap fractional and integer differentiating filters synthesized by inverse Fourier transforming $(i[\cos(\theta)\omega_x + \sin(\theta)\omega_y]/\sigma)^n H(\omega_x,\omega_y)$, Eq. (7), where $H(\omega_x,\omega_y)$ is the Fourier transform of $h[x,y]$. Also shown in this figure are the same

differentiating filters synthesized from a linear combination of eleven basis filters (first- through third-order derivatives plus a lowpass and constant function), Eq. (14). In this figure the label in the upper-left corner of each plot corresponds to the derivative order, and the label in the lower-right corner corresponds to the orientation. Note that the synthesized derivative filters of order less than one do not fully capture the desired filter shape. These errors could be relieved by augmenting the basis set with the filters $c_x[x, y] = |x|^{0.5}$ and $c_y[x, y] = |y|^{0.5}$ which capture the "valley" along the directional axes.

Shown in Fig. 6 are a subset of the scalar weights, $\alpha_4, \alpha_5, \alpha_6$, as a function of derivative order, n, and orientation, θ. These and the other coefficients are well fit with an eight-order polynomial surface. These functions provide the required coefficients for estimating a fractional derivative, at any orientation and order, from a linear combination of integer derivatives, Eq. (17).

As with the 1-D filters, fractional derivatives can be approximated from filters other than the Gaussian-based filters. We consider again a truncated sinc filter with a boxcar impulse response, the lowpass filter with a raised cosine impulse response, and the derivative filters described in [14]. Each of these 2-D filters were synthesized from an outer product of the 1-D filters described in Section III-A.1. In each case 11-tap fractional and integer differentiating filters were synthesized as described above. The same differentiating filters were synthesized from a linear combination of the corresponding first- through third-order derivatives plus a lowpass and constant function, Eq. (8). The average root mean square (RMS) error between the actual and approximate filters of order 0.1 to 3.0 (in steps of 0.1) and at orientations $0°$ to $180°$ (in steps of $10°$) was 2.44 for the sinc-based filters, 0.99 for the cosine filters, 1.64 for the filters of [14] – the Gaussian-based filters have an RMS error of 1.51. As with the Gaussian filters, these other filters are reasonably well approximated with an integer basis, with the sinc-based filters having a slightly higher error.

IV. Discussion

Discrete-time fractional derivative filters (1-D and 2-D) can be approximated from a linear combination of integer derivatives. An arbitrary fractional derivative can, therefore, be efficiently computed from a simple linear combination of the result of convloving a signal/image with a fixed set of filters.

There is, of course, no inherent reason why the integer derivatives are necessarily the best basis for the fractional derivatives. One may wonder if a principle components analysis (PCA), for example, would yield a smaller and more accurate basis. When applied to the one-dimensional 11-tap fractional derivative filters of Section III-A.1, PCA yields a linear basis of size five that captures nearly all of the variance; this is the same size basis as that used in Fig. 3. And when applied to the two-dimensional 11-tap fractional directional derivative filters of Section III-B.1, PCA yields a basis of size eleven that captures nearly all of the variance; this is the same size basis as that used in Fig. 5. There are, nevertheless, some benefits to using the integer derivatives as a basis: (1) an integer basis ensures that synthesized filters at and around integer values will be accurate; (2) the two-dimensional integer derivatives are $x - y$ separable, and thus computationally efficient; (3) with an integer basis the interpolating scalar weights, Eq. (11) and (17), are more likely to vary smoothly, thus lending themselves to closed-form approximation, Fig. 4 and 6; and (4) there is a large literature on designing derivative filters [13] that can be leveraged in creating a filter basis set for fractional derivatives.

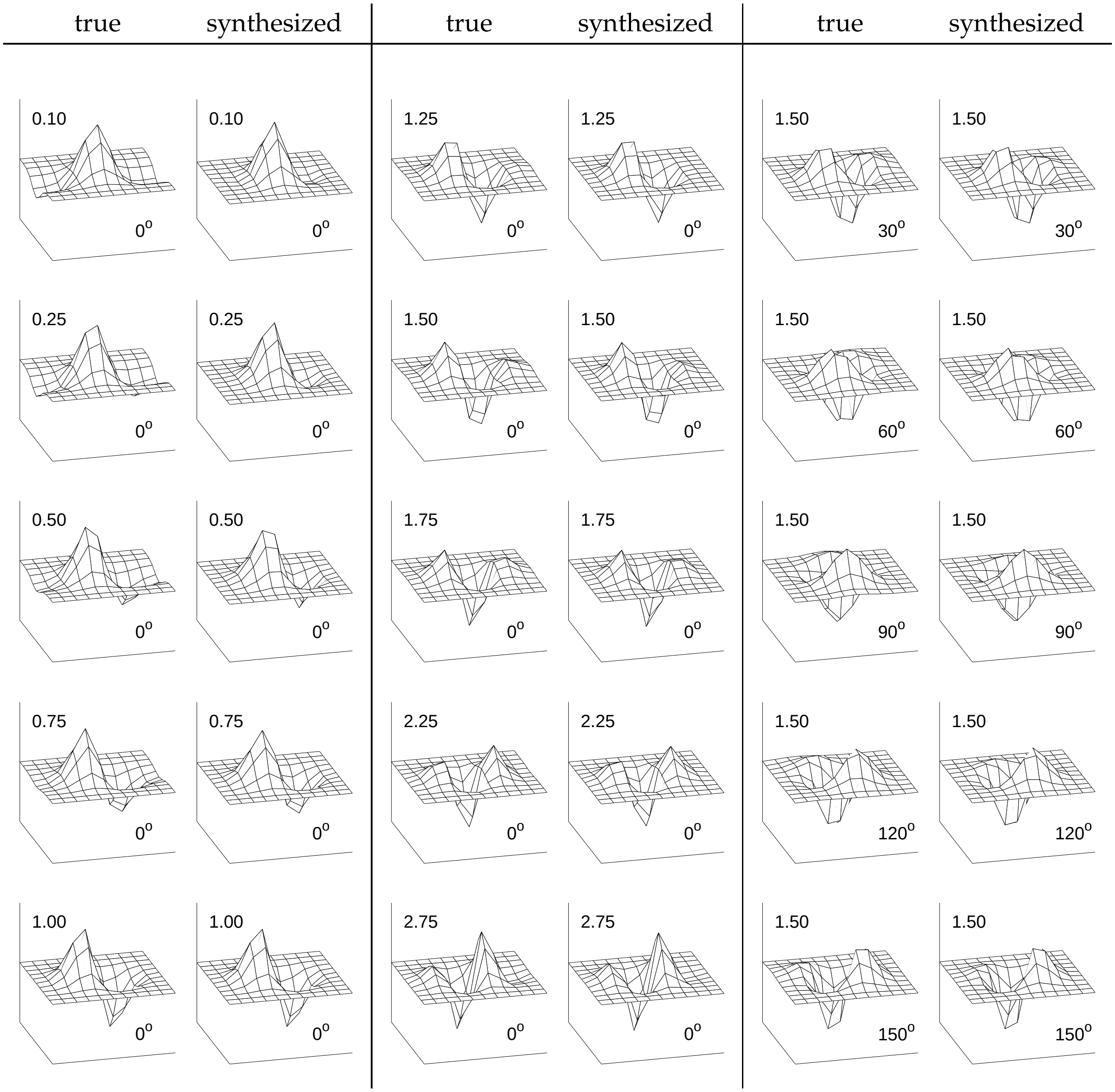

Fig. 5: Shown are two-dimensional 11-tap directional derivative filters of varying orders and orientations (columns 1, 3, 5). Also shown in columns 2, 4, 6, are synthesized filters from a basis set of size eleven (1^{st}- through 3^{rd}-order integer derivatives). For display, each derivative order is normalized to a common scale.

References

[1] G. Leibniz, *Mathematische Schiften*. Georg Olms Verlagsbuchhandlung, Hildesheim, 1962.

[2] K. Oldham and J. Spanier, *The Fractional Calculus*. New York – London: Academic Press, 1974.

[3] K. Miller and B. Ross, *An Introduction to the Fractional Calculus and Fractional Differential Equations*. New York: John Wiley & Sons Inc., 1993.

[4] M. Ortigueira, "Introduction to fractional linear systems. Part 1: Continuous-time case," *IEE Proceedings Vision, Image, Signal Processing*, vol. 147, no. 1, pp. 62–70, 2000.

[5] ——, "Introduction to fractional linear systems. Part 2: Discrete-time case," *IEE Proceedings Vision, Image, Signal Processing*, vol. 147, no. 1, pp. 71–78, 2000.

[6] A. Oustaloup, "From fractality to non integer derivation through recursivity, a property common to these two concepts: A fundamental idea for a new process control strategy," in *Proceedings of the 12th IMACS World Congress*, 1988, pp. 203–208.

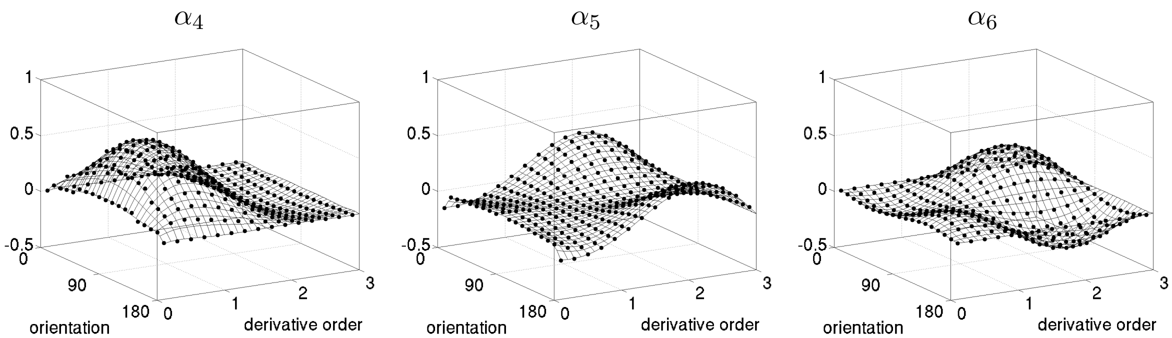

Fig. 6: Shown are a subset of the eleven scalar weights, Eq. (17), for the two-dimensional 11-tap filters of Fig. 5. The meshes correspond to eight-order polynomial surfaces fit to the data points that are computed in $1/10$ increments between a $1/10^{th}$- and 3^{rd}-order derivative, and in $10°$ increments between derivatives oriented from $0°$ to $180°$.

[7] K. Oldham, "A signal-independent electroanalytical method," *Anal. Chem.*, vol. 44, no. 1, pp. 196–198, 1972.

[8] T. Anastasio, "The fractional-order dynamics of brainstem vestibulo-oculomotor neurons," *Biological Cybernetics*, vol. 72, pp. 69–79, 1994.

[9] I. Podlubny, *Fractional Differential Equations*. San Diego: Academic Press, 1999.

[10] C. Tseng, "Design of fractional order digital FIR differentiators," *IEEE Signal Processing Letters*, vol. 8, no. 3, pp. 77–79, 2001.

[11] S. Samadi, M. O. Ahmad, and M. Swamy, "Exact fractional-order differentiators for polynomial signals," *IEEE Signal Processing Letters*, vol. 11, no. 6, pp. 529–532, 2004.

[12] G. Gorenflo, *Fractional calculus: some numerical methods*. New York: Springer-Verlag, 1997, pp. 227–290.

[13] A. Oppenheim and R. Schafer, *Discrete-Time Signal Processing*. Prentice Hall, 1989.

[14] H. Farid and E. Simoncelli, "Differentiation of multi-dimensional signals," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 496–508, 2004.

[15] W. T. Freeman and E. H. Adelson, "The design and use of steerable filters," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 9, pp. 891–906, 1991.