Implicit Image Differentiation and Filtering with Applications to Image Sharpening*

Alexander Belyaev[†]

Abstract. This paper demonstrates potential advantages of using implicit finite differencing and filtering schemes for fast, accurate, and reliable differentiating and filtering of multidimensional signals defined on regular grids. In particular, applications to image enhancement and Gaussian image deblurring are considered. The theoretical contribution of the paper is threefold. The first adapts the Fourier–Padé–Galerkin approximation approach for constructing compact implicit finite difference schemes with desirable spectral resolution properties. The second establishes a link between implicit and explicit finite differences used for gradient estimation. Finally, the third one consists of introducing new implicit finite difference schemes with good spectral resolution properties.

Key words. image derivative estimation, implicit image filtering, implicit finite differences

AMS subject classifications. 68U10, 94A08

DOI. 10.1137/12087092X

1. Introduction. The main goal of this paper consists of demonstrating advantages of using compact implicit finite differencing and filtering schemes for basic image processing applications.

1.1. Previous work on image derivative estimation. Fast and reliable estimation of image derivatives is among the most fundamental tasks of low level image processing. Unfortunately, the image processing literature often provides the reader with a set of recipes given without appropriate mathematical analysis. For example, for image gradient estimation, the image processing textbooks usually refer to the Prewitt and Sobel operators while mentioning that these operators are not so good at preserving the rotation equivariance property of the gradient [22], [39, Chapter 15].

Typically, for a two-dimensional image defined on a regular grid with spacing h, image processing textbooks recommend using a 3×3 kernel

(1.1)
$$D_x = \frac{1}{2h(w+2)} \begin{bmatrix} -1 & 0 & 1 \\ -w & 0 & w \\ -1 & 0 & 1 \end{bmatrix} \equiv \frac{1}{2h} \begin{bmatrix} -1 & 0 & 1 \end{bmatrix} \times \frac{1}{w+2} \begin{bmatrix} 1 \\ w \\ 1 \end{bmatrix}$$

and its $\pi/2$ -rotated counterpart D_y for estimating the x-derivative and y-derivative, respectively. Here w is a parameter: setting w = 1 in (1.1) yields the Prewitt mask and w = 2

http://www.siam.org/journals/siims/6-1/87092.html

660

^{*}Received by the editors March 22, 2012; accepted for publication (in revised form) December 17, 2012; published electronically March 21, 2013.

[†]Institute of Sensors, Signals and Systems, School of Engineering and Physical Sciences, Heriot-Watt University, Edinburgh, EH14 4AS, United Kingdom (a.belyaev@hw.ac.uk).

corresponds to the Sobel mask. Searching for an optimal value of w in (1.1) remains an active research area [9, 18, 2, 21, 42, 25, 3] (see also the references therein).

More than 60 years ago W. G. Bickley, a British applied mathematician, noted that (1.2)

$$D_x\big|_{w=4} \equiv \frac{1}{12h} \left(4 + e^{h\frac{\partial}{\partial y}} + e^{-h\frac{\partial}{\partial y}} \right) \left[e^{h\frac{\partial}{\partial x}} - e^{-h\frac{\partial}{\partial x}} \right] = \left(1 + \frac{h^2}{12}\Delta \right) \frac{\partial}{\partial x} + O(h^4) \text{ as } h \to 0,$$

where Δ is the Laplacian, and, therefore, (1.1) with w = 4 has optimal rotation-invariant properties for small grid spacing h. After its introduction in [9], the Bickley kernel was rediscovered at least twice [29, 35], and was even patented [49], but nevertheless remains little known to the graphics and imaging communities.

Recently Scharr and co-authors [42, 25] suggested using (1.1) with w = 10/3. Their analysis is based on a frequency-based optimization of rotation-invariant properties of (1.1). The Scharr mask works very well in practice [51] and quickly gained popularity among computer vision researchers and practitioners [11, Chapter 6].

Bigger stencils for estimating image derivatives were also widely considered. In particular, Simoncelli and Farid proposed a general frequency-based framework for the design of discrete multidimensional differentiators [43, 19]. Sophisticated gradient estimation filters were studied in [7, 1, 24] in connection with high-quality volume visualization isosurface rendering problems.

1.2. Previous work on compact implicit filters and finite differences. Although implicit finite differences had become known to a general audience of numerical mathematicians and computational physicists after Collatz's book [15], their heyday began after Lele's seminal paper [30], where a remarkable performance of implicit finite differences for computational problems with a range of spatial scales was analyzed and demonstrated. At present compact implicit finite difference schemes constitute advanced but standard tools for accurate numerical simulations of physical problems involving linear and nonlinear wave propagation phenomena [16], [38, section 5.8], [31, Chapter 5].

Implicit filtering schemes were first considered by Raymond [40] and Raymond and Garder [41] in connection with meteorology studies. They were reintroduced in [30] and currently are widely used in computational aeroacoustics [16, 26].

In the signal processing language, compact implicit finite differences and more general implicit filtering schemes are described by infinite impulse response (IIR) filters. In image processing applications, finite impulse response (FIR) filters continue to dominate over IIR image filtering schemes [32, 33]. Notable exceptions include works of Unser and co-workers on B-splines, their generalizations, and applications [47, 48, 10, 45] and very recent studies on accurate visualization of volumetric data [1, 24], where sophisticated implicit gradient estimation schemes were developed.

1.3. Paper contribution. We believe that the potential of implicit finite difference schemes for image processing applications is largely underestimated and consider this work as an attempt to demonstrate the usefulness of implicit finite differences for basic image processing tasks. Other contributions of the paper include the following:

• adapting Fourier–Padé–Galerkin approximations for designing high-quality implicit image differencing schemes (the second half of section 4);

- establishing a link between implicit and explicit finite differences used for gradient estimation (section 3);
- introducing new implicit differencing schemes and evaluating their properties;
- demonstrating usefulness of implicit differencing and filtering schemes for various image processing tasks including image deblurring and sharpening.

All the mathematical derivations presented in this paper are elementary and based on straightforward calculations. The main results of this paper were previously reported by the author in conference proceedings [5, 6].

2. Estimating the derivative for univariate signals. Consider a uniformly sampled signal f(x). Let us recall that the Nyquist, or folding, frequency is the highest frequency that can be represented in the signal. It equals one-half of the sampling rate.

Consider the simplest central difference operator

(2.1)
$$f'(x) \approx \frac{1}{2h} \left[f(x+h) - f(x-h) \right]$$

and its corresponding mask

(2.2)
$$\frac{1}{2h}[-1,0,1]$$

defined on a grid with spacing h. For the sake of simplicity we assume that h = 1, which is equivalent to rescaling the x-coordinate: $x \to x/h$. The eigenvalues of the linear operators corresponding to the left- and right-hand sides of (2.1) and subject to periodic boundary conditions are found by setting $f(x) = \exp\{j\omega x\}, j = \sqrt{-1}, -\pi < \omega < \pi$, in (2.1):

$$j\omega e^{j\omega x} \approx \frac{1}{2} \left[e^{j\omega(x+1)} - e^{j\omega(x-1)} \right] \equiv j\sin\omega e^{j\omega x}, \qquad j\omega \approx j\sin\omega$$

The frequency response function (the eigenvalue for eigenfunction $\exp\{j\omega x\}$) $j\sin\omega$ corresponding to the central difference operator in (2.1) delivers a satisfactory approximation of the frequency response function $j\omega$ of the ideal derivative only for sufficiently small frequencies (wavenumbers) ω . See, for example, [23, section 6.4]. One way to improve (2.1) consists of using implicit finite differences.

2.1. Implicit finite differences for univariate signals. One can say that (2.2) adds to the true derivative a certain amount of smoothing applied to nonzero wavenumbers. A natural way to compensate for smoothing introduced by (2.2) consists of adding approximately the same amount of smoothing to the derivative. This simple idea leads us immediately to the concept of implicit finite differences.

For example, the two simplest implicit schemes for approximating the first-order derivative of a function f(x) are based on the following relations [15, p. 538]:

$$(2.3) \quad f'(x-h) + 4f'(x) + f'(x+h) = \frac{3}{h} \left[f(x+h) - f(x-h) \right] + \frac{h^4}{30} f^V(x) + \cdots$$
$$f'(x-h) + 3f'(x) + f'(x+h) = \frac{1}{12h} \left[f(x+2h) + 28f(x+h) - 28f(x-h) - f(x-2h) \right] - \frac{h^6}{420} f^{VII}(x) + \cdots$$

The corresponding implicit finite differences

(2.5)
$$f'_{i-1} + 4f'_i + f'_{i+1} = 3(f_{i+1} - f_{i-1}),$$

(2.6)
$$f'_{i-1} + 3f'_i + f'_{i+1} = \frac{1}{12} \left(f_{i+2} + 28f_{i+1} - 28f_{i-1} - f_{i+2} \right)$$

are often called the fourth- and sixth-order tridiagonal Padé schemes, respectively, since (2.3) and (2.4) are derived using classical Padé rational approximations and since (2.5) and (2.6) lead to tridiagonal systems of linear equations. It is interesting that (2.5) can be obtained if the grid data $\{f_i\}$ is first B-spline interpolated and then processed by the central difference filter (2.2).

Taking into account a similarity between the Bickley mask (1.1) with w = 4 and (2.5) let us call the latter the implicit Bickley scheme. Note that (1.2) can be rewritten as

$$\frac{\partial}{\partial x} = \left(1 + \frac{h^2}{12}\Delta\right)^{-1} D_x\big|_{w=4} + O(h^4),$$

which immediately leads to the implicit Bickley scheme (2.5).

One can observe that the frequency response function

(2.7)
$$S_w(\omega) = \frac{w + 2\cos\omega}{w + 2}$$

corresponds to the smoothing kernel

(2.8)
$$\frac{1}{w+2}[1, w, 1]$$

The frequency response functions associated with (2.5) and (2.6) are given by

(2.9)
$$j \sin \omega \frac{1}{S_4(\omega)}$$
 and $j \sin \omega \frac{S_{28}(\omega)}{S_3(\omega)}$

respectively. It is easy to verify that these two frequency response functions deliver fourth- and sixth-order approximations, respectively, of $j\omega$, the frequency response function corresponding to the ideal derivative, at $\omega = 0$.

Thus the amount of smoothing introduced by (2.2) is compensated in (2.9) by applying (2.8) with w = 4 to the derivative. Finite difference scheme (2.6) delivers a more accurate approximation of the true derivative: smoothing by (2.8) with w = 3 applied to the derivative is balanced by the amount of smoothing introduced by (2.2) plus smoothing by (2.8) with w = 28 applied to the sampled function itself.

It is natural to evaluate the quality of a finite difference approximation by its resolving efficiency, the range of frequencies (wavenumbers) ω over which a satisfactory approximation of the exact differentiation is achieved. Since it is not possible to get a reasonably good approximation when ω is close to π , the frequency range for the optimization is often specified by $0 \leq \omega \leq r\pi$ with some $0 < r \leq 1$.

Although a quantitative study of the resolving efficiency of finite difference schemes is a straightforward task [30], everywhere below we make our judgments based on a visual analysis of graphs of the corresponding frequency response functions.



Figure 1. Left: Resolving efficiency of various finite difference schemes is demonstrated by plotting the graphs of their corresponding frequency response functions. Note the good resolving efficiency of the implicit Scharr scheme. Right: a graphical comparison of the resolving efficiency of various frequency response functions corresponding to implicit finite difference schemes. A high resolving efficiency of both the Fourier-Padé-Galerkin schemes is clearly demonstrated.

In the left image of Figure 1, we provide the reader with a visual comparison of the frequency-resolving efficiencies of $j \sin \omega$ corresponding to central difference (2.1) and frequency response functions (2.9). In addition, we plot the graph of

(2.10)
$$j\sin\omega \frac{1}{S_{10/3}(\omega)} \equiv j\sin\omega \frac{w+2}{w+2\cos\omega} \bigg|_{w=10/3},$$

which we call the *implicit Scharr scheme*, since it can be considered as a counterpart of the original Scharr kernel (1.1) with w = 10/3 introduced in [42, 25]. One can observe that the resolving efficiency of the implicit Scharr scheme is comparable with that of (2.6) in spite of the fact that the latter has a wider stencil.

3. Estimating image gradient. Now it becomes clear in which way (1.1) improves the standard central difference (2.2): smoothing due to the use of the central difference operator instead of the true x-derivative is compensated by adding a certain amount of smoothing in the y-direction. Thus (1.1) and its y-direction counterpart do a better job in estimating the gradient direction than in estimating the gradient magnitude.

If the goal is to achieve an accurate estimation of both the gradient direction and magnitude, we can combine (1.1) and the corresponding 3×3 discrete Laplacian

(3.1)
$$L_w = \frac{1}{h^2(w+2)} \begin{bmatrix} 1 & w & 1\\ w & -4(w+1) & w\\ 1 & w & 1 \end{bmatrix}$$

as follows. Let $\delta = [0 \ 0 \ 0; \ 0 \ 1 \ 0; \ 0 \ 0]$ be the 3×3 identity kernel. Note that

(3.2)
$$\delta + \frac{h^2}{w+2} L_w \equiv \frac{1}{(w+2)^2} \begin{bmatrix} 1 & w & 1 \\ w & w^2 & w \\ 1 & w & 1 \end{bmatrix} \equiv \frac{1}{w+2} \begin{bmatrix} 1 & w & 1 \end{bmatrix} \times \frac{1}{w+2} \begin{bmatrix} 1 \\ w \\ 1 \end{bmatrix},$$

which can be considered as simultaneous smoothing (averaging) with respect to both coordinate directions. Thus, in order to remove smoothing introduced by (1.1), it is natural to use

(3.3)
$$\left(\delta + \frac{h^2}{w+2} L_w\right)^{-1} D_x$$

which combines (1.1) with an implicit Laplacian-based sharpening. The frequency response function corresponding to (3.3) applied to the eigenfunction $\exp(j(\omega_1 x + \omega_2 y))$ is given by

(3.4)
$$H(\omega_1, \omega_2) = j \frac{\sin \omega_1}{S_w(\omega_1)}$$

which, in turn, corresponds to the following implicit finite difference scheme:

(3.5)
$$\frac{1}{w+2} \left(f'_{i-1,j} + w f'_{i,j} + f'_{i+1,j} \right) = \frac{1}{2h} \left(f_{i+1,j} - f_{i-1,j} \right).$$

Although the image gradient (I_x, I_y) is a rotation-invariant quantity, its finite difference approximations are not. For many image processing and computer vision applications, a correct estimation of the gradient direction $\arctan(I_y/I_x)$ is of primary importance.



Figure 2. Gradient direction (top row) and gradient (bottom row) errors calculated using various approximations of the gradient. Cooler color means higher accuracy. (a) Sobel kernel (1.1) with w = 2. (b) Scharr kernel (1.1) with w = 10/3. (c) Implicit Scharr scheme (3.5) with w = 10/3. (d) Explicit filter (1.1) with w = 4. (e) Implicit filter (3.5) with w = 4. The Scharr kernel and implicit Scharr scheme deliver very similar gradient direction accuracy. Notice how good the implicit schemes are in estimating the gradient.

Figure 2 shows results of our numerical experiments with sinusoidal grating

$$I(x,y) = \sin\left(x^2 + y^2\right),$$

where x and y are ranging from -16 to 16 with step-size h = 0.1. It means that the frequencies vary from 0 to 0.51 cycles/pixel. The images of the top row show gradient direction errors

 $\left| \arctan\left(I_{y}^{appr} / I_{x}^{appr} \right) - \arctan\left(I_{y} / I_{x} \right) \right|$

calculated using various approximations of the gradient for each pixel. The images of the bottom row present gradient errors

$$\sqrt{(I_x - I_x^{appr})^2 + (I_y - I_y^{appr})^2}$$

for various approximations of the gradient at each pixel. Comparing the images obtained using the Scharr (w = 10/3) and Bickley (w = 4) kernels (1.1) with their implicit counterparts justifies our theoretical results and demonstrates the advantages of using implicit schemes for gradient estimation.

3.1. Implicit low-pass filtering. Inspired by [30] let us consider a simple one-parameter family of implicit low-pass filters

(3.6)
$$\frac{1}{w+2} \left[\hat{f}_{i-1} + w \hat{f}_i + \hat{f}_{i+1} \right] = \frac{1}{2} f_i + \frac{1}{4} \left(f_{i-1} + f_{i+1} \right).$$

The corresponding frequency response function is given by

(3.7)
$$T_w(\omega) = \frac{w+2}{w+2\cos\omega} \times \frac{1+\cos\omega}{2} \equiv \frac{S_2}{S_w}$$

Here we assume that w > 2. Observe that (3.6) combines smoothing kernel [1/4, 1/2, 1/4] with the inverse of

(3.8)
$$\frac{1}{w+2} [1, w, 1].$$

It turns out that (3.7) belongs to the family of so-called *implicit tangent filters* introduced by Raymond [40] and Raymond and Garder [41] and described by their corresponding frequency response functions

(3.9)
$$T_{\epsilon,p}(\omega) = \left(1 + \epsilon \tan^{2p} \frac{\omega}{2}\right)^{-1}, \quad p = 1, 2, 3, \dots,$$

where ϵ and p are user-specified parameters. Namely, substituting p = 1 and $\epsilon = \frac{w-2}{w+2}$ in (3.9) gives (3.7). Monotonicity of (3.9) for $0 < \omega < \pi$ and asymptotics

$$T_{\epsilon,p}(\omega) = \begin{cases} 1 - \epsilon \left(\frac{\omega}{2}\right)^{2p} + O(\omega^{2p+2}) & \text{as} \quad \omega \to 0, \\ \frac{1}{\epsilon} \left(\frac{\omega - \pi}{2}\right)^{2p} + O\left((\omega - \pi)^{2p+2}\right) & \text{as} \quad \omega \to \pi \end{cases}$$

justify remarkable properties of the implicit tangent filters. The left image of Figure 3 displays the frequency response function (3.9) for p = 2 and various values of ϵ .

Note that

(3.10)
$$\frac{(a+2)(b+2)}{a-b} \left[T_b(\omega) - T_a(\omega) \right] = \omega^2 + \frac{8+a+b-ab}{3(a+2)(b+2)} \omega^4 + O\left(\omega^6\right) \text{ as } \omega \to 0,$$

where the fourth-order term disappears if b = (a+8)/(a-1). Thus (3.10) delivers an accurate implicit approximation of the second-order derivative for low frequencies ω .

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

In particular, the limit case $a \to 4$ (if a = 4, then b = 4 and (3.10) is not properly defined) corresponds to (2.5) applied twice. Setting a = 10 (therefore b = 2) yields the simplest Padé scheme for the second-order derivative

(3.11)
$$f_{i-1}'' + 10f_i'' + f_{i+1}'' = 12(f_{i+1} - 2f_i + f_{i-1}),$$

which can be found in [15, p. 538]).

The right image of Figure 3 displays graphs of the frequency response functions corresponding to (3.10). The main advantage of using (3.10) for estimating the second-order derivative is that it is computationally efficient (solving three-diagonal systems of linear equations) and allows the user to control the amount of smoothing needed for robust differencing of noisy signals.

It is also worth noting that (3.10) resembles the standard Difference-of-Gaussian (DoG) filter and, therefore, can be employed for edge detection and nonphotorealistic rendering purposes (see, for example, [53] and the references therein, where DoG was used for edge-based image enhancement and abstraction).

A different normalization of $T_b(\omega) - T_a(\omega)$ yields a two-parametric family of band-pass filters. The normalization coefficient can easily be found analytically, and the resulting bandpass filters have low-computational complexity.



Figure 3. Graphs of frequency response functions for various implicit filtering schemes. Left: implicit low-pass filters (3.9). Some of them are used in image deblurring experiments conducted in the paper. Right: various approximations of the second-order derivative operator. See the main text for further details.

4. High-resolution schemes for first-order derivatives. Both (2.5) and (2.6) are particular cases of a general seven-point stencil introduced in [30]

$$(4.1) \quad \beta f'_{i-2} + \alpha f'_{i-1} + f'_i + \alpha f'_{i+1} + \beta f'_{i+2} = c \frac{f_{i+3} - f_{i-3}}{6h} + b \frac{f_{i+2} - f_{i-2}}{4h} + a \frac{f_{i+1} - f_{i-1}}{2h},$$

where h = 1 is the grid spacing, and $\{f_i\}$ and $\{f'_i\}$ are the values of a given function f(x)and its derivative at the grid points, respectively. Here the coefficients α , β and a, b, c are

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

determined such that the amount of smoothing introduced by the right-hand side of (4.1) is compensated by averaging the derivatives in the left-hand side of (4.1).

The frequency response function corresponding to (4.1) is given by

(4.2)
$$H(\omega) = j \frac{a \sin \omega + (b/2) \sin 2\omega + (c/3) \sin 3\omega}{1 + 2\alpha \cos \omega + 2\beta \cos 2\omega}$$

The problem of determining the coefficients α , β and a, b, c in (4.2) can now be formulated in a more mathematical way: they are chosen such that $H(\omega)$ delivers a good approximation of ω , the frequency response function of the true derivative.

In his seminal paper [30], Lele used empirical considerations to derive the set of coefficients

(4.3)
$$\begin{array}{c} \alpha = 0.5771439, \quad \beta = 0.0896406, \\ a = 1.302566, \quad b = 0.99355, \quad c = 0.03750245 \end{array} \}$$

which deliver an exceptional resolving efficiency to compare with many other approximations used currently in computational aeroacoustics [16, section 4.1.1] (see also the references therein). Below we use Fourier–Padé–Galerkin approximations for improving Lele's result and establishing a general approach to frequency response function design.

4.1. Fourier–Padé–Galerkin approximations for derivative. Following [34] let us consider the linear space of trigonometric polynomials of degree N

$$\mathcal{V}_N = \operatorname{span}\{e^{jn\omega} \mid -N \le n \le N\}$$

and define a rational Fourier series by

$$R_{kl}(\omega) = P_k(\omega)/Q_l(\omega), \quad P_k \in \mathcal{V}_k, \ Q_l \in \mathcal{V}_l.$$

Given a 2π -periodic function $f(\omega)$, $\omega \in [-\pi, \pi]$, the Fourier–Padé–Galerkin approximation $f(\omega) \approx R_{kl}(\omega)$ determines the unknown coefficients in rational function $R_{kl}(\omega)$ from orthogonality conditions

(4.4)
$$\int_{-\pi}^{\pi} \left[P_k(\omega) f(\omega) - Q_l(\omega) \right] \overline{g(\omega)} W(\omega) \, d\omega = 0 \quad \forall g(\omega) \in \mathcal{V}_{k+l},$$

where $W(\omega)$ is a properly chosen weighting function. In practice, since \mathcal{V}_{k+l} is a linear space, the k+l unknown coefficients are determined from (4.4) with some k+l linearly independent test functions $g_1(\omega), g_2(\omega), \ldots, g_{k+l}(\omega)$. Thus (4.4) leads to a system of k+l linear equations with k+l unknowns.

Pentadiagonal implicit schemes. As a simple application of the above Fourier–Padé–Galerkin approach, let us use (4.4) with $W(\omega) \equiv 1$ to determine the coefficients in (4.1). Since we approximate $f(\omega) = \omega$, which is odd on $[-\pi, \pi]$, we set $\mathcal{V}_5 = \operatorname{span}\{\sin n\omega \mid 1 \leq n \leq 5\}$, use

$$P_3(\omega) = a\sin\omega + (b/2)\sin 2\omega + (c/3)\sin 3\omega \quad \text{and} \quad Q_2(\omega) = 1 + 2\alpha\cos\omega + 2\beta\cos 2\omega,$$

and immediately arrive at

(4.5)
$$\alpha = \frac{3}{5}, \quad \beta = \frac{21}{200}, \quad a = \frac{63}{50}, \quad b = \frac{219}{200}, \quad c = \frac{7}{125}.$$

Our Fourier–Padé–Galerkin approach (4.4) can be considered as a generalization of the least-square fitting procedures used to design explicit [44] and implicit [27] finite differencing schemes.

Note that all the computations involved are very simple. For this particular example, the integrals can be evaluated analytically and the resulting system of five linear equations with five unknowns can be solved by hand. We, however, use Maple for both of these tasks; see the Maple code below.

The right image of Figure 1 demonstrates advantages of (4.1), (4.5). The scheme has a good resolving efficiency but suffers from a slight Gibbs-type phenomenon. A simple yet efficient way to reduce this Gibbs-type artifact consists of choosing an appropriate weight function $W(\omega)$ in (4.4). In particular, setting $W(\omega) = 1$ for $|\omega| \leq 0.9\pi$ and $W(\omega) = 0$ otherwise gives a quite satisfactory result, as shown in the right image of Figure 1.

The right image of Figure 1 provides the reader with a visual comparison of the frequencyresolving efficiencies of (4.3) and both the Fourier–Padé–Galerkin schemes. In addition, we also consider the sixth-order tridiagonal Padé scheme whose frequency response function is also shown in Figure 1 and the tenth-order pentadiagonal Padé scheme

(4.6)
$$\alpha = \frac{1}{2}, \quad \beta = \frac{1}{20}, \quad a = \frac{17}{12}, \quad b = \frac{101}{150}, \quad c = \frac{1}{100}$$

whose frequency response function (4.2), (4.6) delivers the maximal approximation order of $f(\omega) \equiv \omega$ at $\omega = 0$ among the family of five-diagonal schemes (4.1).

Although the right image of Figure 1 suggests that both the Fourier–Padé–Galerkin schemes outperform (4.3) (which, taking into account the importance of an accurate estimation of derivatives for computational aeroacoustics problems, would be an important achievement), we do not claim that. The definition of resolving efficiency depends on a specified error tolerance, and some additional efforts are required to see how accurate (4.2) with (4.5) approximates the ideal derivative. However, we consider the right image of Figure 1 as a

demonstration of the approximation power of the Fourier–Padé–Galerkin approach. A few other applications of the approach are considered below.

Tridiagonal implicit schemes. If we are interested in less accurate but simpler implicit approximations of the derivatives, we can set

$$(4.7) b = c = 0 = \beta$$

in (4.1), for example. Now choosing

(4.8)
$$g_k(\omega) = \sin(k\omega), \ k = 1, 2, \text{ and } W(\omega) = \begin{cases} 1 & \text{if } |\omega| < 3\pi/4, \\ 0 & \text{otherwise} \end{cases}$$

yields

$$\alpha = \frac{3(22+21\pi)\sqrt{2}}{2(52+78\pi+27\pi^2)} \approx \frac{1}{3} \quad \text{and} \quad a = \frac{(71+96\pi+27\pi^2)\sqrt{2}}{52+78\pi+27\pi^2} \approx \frac{8}{5}.$$

These values of α and a are close to those for the implicit Scharr scheme for which $\alpha = 3/10$ and a = 8/5.

It turns out we still have room for improvement. For example, selecting

(4.9)
$$g_k(\omega) = \sin(k\omega), \ k = 1, 2, \text{ and } W(\omega) = \begin{cases} 1 + e^{3\omega} & \text{if } |\omega| < 3\pi/4, \\ 0 & \text{otherwise} \end{cases}$$

gives closed-form expressions for α and a which can be approximated by

$$\alpha = 0.3581542431$$
 and $a = 1.573767241$.

The left image of Figure 4 compares the frequency-resolving efficiencies of the implicit Scharr scheme and the schemes corresponding to (4.8) and (4.9).



Figure 4. Left: a visual comparison the frequency-resolving efficiencies of three diagonal implicit schemes. Right: a visual evaluation of frequency-resolving properties of implicit cell-centered schemes.

Cell-centered implicit schemes. The Fourier–Padé–Galerkin approach (4.4) is quite general and can be used in many other situations. For example, let us consider finite difference schemes for cell-centered grids. Cell-centered finite differences are closely related to finite volume approximations and widely used for numerical solving of partial differential equations. In general, cell-centered finite differences have better frequency-resolving characteristics than the node-centered schemes considered so far.

The simplest cell-centered approximation of the derivative is given by

(4.10)
$$f'_i = \frac{f_{i+1/2} - f_{i-1/2}}{h}$$

where, as before, h = 1 is the grid spacing. Let us consider a general implicit scheme

(4.11)
$$\beta f'_{i-2} + \alpha f'_{i-1} + f'_i + \alpha f'_{i+1} + \beta f'_{i+2} = b \frac{f_{i+3/2} - f_{i-3/2}}{3h} + a \frac{f_{i+1/2} - f_{i-1/2}}{h}$$

with four coefficients to be determined. The transfer function corresponding to (4.11) is given by

$$H(\omega) = j \frac{2a\sin(\omega/2) + (2b/3)\sin(3\omega/2)}{1 + 2\alpha\cos\omega + 2\beta\cos2\omega}$$

Setting $\beta = 0 = b$ in (4.11) yields a two-parametric family of tridiagonal schemes. The Padé approximation corresponds to $\alpha = 1/22$ and a = 12/11. To demonstrate how the Fourier–Padé–Galerkin approach deals with this simple case, we use the test and weighting functions defined by (4.8) and arrive at

(4.12)
$$\alpha = 0.07155774932$$
 and $a = 1.120033921$

The right image of Figure 4 suggests that set (4.12) delivers better frequency-resolving properties than the Padé approximation.

Now let us consider a four-parametric family of pentadiagonal schemes defined by (4.11). To determine α , β , a, and b in (4.11) we use (4.4), where we simply set $W(\omega) = 1$ everywhere and select four functions $g_k(\omega) = \sin(k\omega)$, k = 1, 2, 3, 4. The coefficients are given by

(4.13)
$$\alpha = \frac{49170}{123803}, \quad \beta = \frac{3255}{247606}, \quad a = \frac{115028235}{507097088}\pi, \quad b = \frac{178520265}{507097088}\pi$$

and, as demonstrated by the right image of Figure 4, cell-centered finite difference scheme (4.11) with the above coefficients has excellent approximation properties.

It is also worth noting here that the Fourier–Padé–Galerkin approach can be easily extended for dealing with second- and higher-order derivatives and for filter design purposes.

5. Potential applications. Below we demonstrate the advantages of implicit image derivatives and filters for two basic image processing tasks: feature detection and image enhancement.

5.1. Feature detection. The classical Canny edge detection scheme is used as our first test for implicit image derivatives. In Figure 5 we compare implicit Scharr scheme (2.10) and Fourier–Padé–Galerkin scheme (4.2), (4.5) with the Sobel mask, Scharr kernel, and Farid–Simoncelli 5-tap filter [19]. We use MATLAB functions from [28] with the same parameter settings for all the schemes tested. As expected, the explicit schemes add unnecessary blur and, therefore, are less sensitive to fine image details to compare with the implicit schemes.



Figure 5. Canny edge detection with explicit and implicit finite differences, $\sigma = 0.75$. (a) Sobel mask; (b) explicit Scharr mask; (c) Farid–Simoncelli 5-tap filter; (d) implicit Scharr scheme; (e) Fourier–Padé–Galerkin scheme (4.1), (4.5). Explicit schemes (a), (b), and (c) add unnecessary blur and, therefore, are less sensitive to fine image edges. For example, the edge of the nose is not sufficiently sharp and appears only with implicit schemes (d) and (e).

5.2. Stabilized inverse diffusion. In our next test, we deal with image deblurring and sharpening. We take an image and add a Gaussian blur to it. Then we start a simple deblurring/sharpening process by the inverse diffusion equation

(5.1)
$$\partial I/\partial t = -\Delta I, \quad I(x, y, t)|_{t=0} = I_0(x, y),$$

which we solve numerically by the forward Euler method

(5.2)
$$I(x,y,t+dt) = I(x,y,t) - dt \Delta_h I(x,y,t),$$

where Δ_h is a discrete approximation of the Laplacian. It is well known that (5.1) and its discrete counterpart (5.2) are highly unstable because they lead to an exponential growth of the nonzero image frequencies. Two discrete Laplacians are tested: the standard five-point Laplacian and the Laplacian whose second-order partial derivatives are estimated via (2.5) applied twice. The latter, as demonstrated in the right image of Figure 3, combines a very accurate approximation of the second-order derivative for the low frequencies with high-quality low-pass filtering. The images of the top row of Figure 6 demonstrate the advantages of using twice-applied (2.5) over the five-point discrete Laplacian for Gaussian deblurring with (5.2).

Another way to stabilize (5.1) and its discrete approximation (5.2) consists of directly suppressing high frequencies. Let us combine (5.2) with low-pass filtering

(5.3)
$$I(x, y, t + dt) = (\text{low-pass}) \left[I(x, y, t) - dt \Delta_h I(x, y, t) \right].$$

The low-pass filtering scheme we use here employs implicit tangent filters defined by (3.9). As shown in [40, 41], for p = 2 the filters have a particularly simple form

(5.4)
$$(S + \epsilon L) \begin{bmatrix} \hat{f}_{i-2} \\ \hat{f}_{i-1} \\ \hat{f}_{i} \\ \hat{f}_{i+1} \\ \hat{f}_{i+2} \end{bmatrix} = S \begin{bmatrix} f_{i-2} \\ f_{i-1} \\ f_{i} \\ f_{i+1} \\ f_{i+2} \end{bmatrix}$$



Figure 6. Top row: (a) the original Trui image; (b) the image is Gaussian blurred; (c) the blurred image is restored and even sharpened by 45 iterations of (5.2) with the Laplacian approximated by (2.5) applied twice to each dimension (use zoom to see how well small-scale image details are restored); (d) an attempt to restore the blurred image by (5.2) with the standard five-point discrete Laplacian; small image defects are already observed after 37 iterations of (5.2), while the blur is not eliminated; (e) after one more iteration of (5.2): the blur is not reduced and the defects become much more apparent. Bottom row: (a) Gaussian blur with $\sigma = 5$ is added; (b)–(e) deblurring of (a) by (5.3); see the main text for details.

with $S = \begin{bmatrix} 1 & 4 & 6 & 4 & 1 \end{bmatrix}$ and $L = \begin{bmatrix} 1 & -4 & 6 & -4 & 1 \end{bmatrix}$. Despite its simplicity, (5.4) maintains a relatively sharp cut-off transition.

The bottom row of Figure 6 demonstrates our experiments with (5.3) and (5.4). We add blur using a discrete 7×7 kernel

$$K = \frac{1}{1003} \begin{bmatrix} 0 & 0 & 1 & 2 & 1 & 0 & 0 \\ 0 & 3 & 13 & 22 & 13 & 3 & 0 \\ 1 & 13 & 59 & 97 & 59 & 13 & 1 \\ 2 & 22 & 97 & 159 & 97 & 22 & 2 \\ 1 & 13 & 59 & 97 & 59 & 13 & 1 \\ 0 & 3 & 13 & 22 & 13 & 3 & 0 \\ 0 & 0 & 1 & 2 & 1 & 0 & 0 \end{bmatrix}$$

approximating the Gaussian with $\sigma = 1$. The original image was convolved 25 times with kernel K. This is equivalent to adding Gaussian blur with $\sigma = 5$ and some noise due to discretely approximating the Gaussian kernel.

We set dt = 0.2 in (5.2) and find out that $\epsilon = 0.14$ (see the middle image of Figure 3, where the thick line visualizes the graph of the frequency response function corresponding to (5.4) with $\epsilon = 0.14$) is a good choice for low-pass filtering with (5.4). In theory, $\sigma^2/dt = 62.5$ iterations are needed to recover the original image. However, discrete Laplacian Δ_h introduces some smoothing, and, therefore, additional iterations of (5.2) are needed.

In Figure 7, we demonstrate how the root mean square error (RMSE) and the structural similarity (SSIM) [50] between the original and deblurred images vary with the number of

iterations of (5.2). The RMSE achieves its minimal value after 66 iterations of (5.2) and SSIM attains its maximal value after 67 iterations. The RMSE and SSIM optimal images seem underdeblurred, and more iterations of (5.2) are needed.

In our experiments shown in the bottom row of Figure 6, we use (3.11) for approximating the second-order derivatives. The standard explicit approximations of the Laplacian lead to slightly worse results, since these approximations are less accurate for low and middle range frequencies ω to compare with the implicit discrete Laplacian obtained from (3.11). A visually good deblurring result is achieved after 75 iterations of (5.2); see the middle image of the bottom row of Figure 6.

The RMSE and SSIM graphs suggest that the instability develops quickly after the total number of iterations goes beyond the RMSE/SSIM optimal values. In practice, however, (5.2) demonstrates that the visual quality of the the overdeblurred images decreases relatively slow, as shown by the two right images of the bottom row of Figure 6.

Figure 7 also demonstrates that (5.2) has a low sensitivity to the selection of a discrete approximation of the Laplacian and choice of parameter ϵ in (5.4). Very similar results are obtained if implicit filters proposed recently by Kim [26] are used instead of (5.4).



Figure 7. Graphs of RMSE error (L^2 -error) and SSIM index for Gaussian image deblurring with (5.2).

It is also worth noting here that Sobolev gradient flows [13, 14, 12] and the closely related screened Poisson equation approach [8] used for image sharpening are also based on suppressing high-order image frequencies.

5.3. Unsharp masking. Our other example is also devoted to image sharpening. Possibly the simplest image sharpening scheme consists of using one iteration (5.2) and can be considered as a variant of unsharp masking. However, unsharp masking amplifies high frequencies and, therefore, may oversharpen image texture and other small-scale image details. In order to reduce such oversharpening we use (5.3) instead of (5.2).

This time we employ a one-parametric family of low-pass implicit filters proposed recently by Kim [26]. Figure 8 displays the frequency response functions of three filters from the family. Each frequency response function from the family has a nice monotonically decreasing profile with an inflection point at $\omega = k\pi$, where k parameterizes the family.

Figure 9 demonstrates the advantages of using (5.3) for image sharpening purposes.



Figure 8. The frequency response functions of three filters from the family of low-pass filters proposed in [26]. Each frequency response function from the family has a nice monotonically decreasing profile with an inflection point at $\omega = k\pi$, where k parameterizes the family.



Figure 9. (a) Original Barbara image. (b) High-frequency image details extracted using the k = 0.5 (half the Nyquist frequency) low-pass filter from Figure 8. (c) The original image is enhanced by the Laplacian subtraction filter (unsharp masking); high-frequency texture details are severely oversharpened. (d) The sharpened image is smoothed by the k = 0.5 low-pass filter from [26]; oversharpening is suppressed.

5.4. Perona–Malik diffusion. In our last example, we deal with a variant of the Perona–Malik nonlinear diffusion [36]

(5.5)
$$\partial I/\partial t = \operatorname{div}\left[g(\nabla I)\nabla I\right], \quad I(x,y,t)|_{t=0} = I_0(x,y), \quad g(\nabla I) = \exp\left\{-\|\nabla I\|/\lambda\right\},$$

where λ is a properly chosen positive constant. A standard finite difference approximation of the nonlinear diffusion operator

(5.6)
$$\operatorname{div}\left[g(\nabla I)\nabla I\right]$$

is given by (see, for example, [4])

(5.7)
$$\frac{1}{h^2} \left[g_{+0}(I_{i+1,j} - I_{i,j}) + g_{-0}(I_{i-1,j} - I_{i,j}) + g_{0+}(I_{i,j+1} - I_{i,j}) + g_{0-}(I_{i,j-1} - I_{i,j}) \right],$$

where $g_{\pm 0}$ and $g_{0\pm}$ denote the values of $g(\nabla I)$ at locations $((i \pm 1/2)h, jh)$ and $(ih, (j \pm 1/2)h)$, respectively. Thus only cell-centered finite differences are used in (5.7). Usually (see, for

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

example, [37]), the simplest cell-centered approximation of the derivative (4.10) is also used for estimating $g_{\pm 0}$ and $g_{0\pm}$.

Our finite difference approximation of (5.6) differs from (5.7) by using our cell-centered implicit finite difference scheme (4.11), (4.12) instead of (4.10). In addition, the same highly accurate scheme is employed for estimating $g_{\pm 0}$ and $g_{0\pm}$.

In Figure 10, we compare our finite difference approximation of (5.6) with the standard one (for the latter, we use the MATLAB source code from [37]). As expected, our scheme does a better job in preserving high frequencies.



Figure 10. A circular sinusoidal pattern corrupted by noise is filtered by Perona–Malik nonlinear diffusion (5.5). Left: the original noisy image. Middle: standard finite difference scheme (5.7) is used for approximating the Perona–Malik diffusion. Right: Perona–Malik diffusion operator (5.6) is approximated using highly accurate cell-centered implicit finite difference scheme (4.11), (4.12). The same parameter settings are applied to both implementations.

6. Discussion and conclusion. Implicit finite differencing and filtering schemes offer much higher spectral resolving efficiency compared to explicit schemes, while the computational effort increases only slightly. The resulting narrow-banded systems of linear equations can be efficiently solved by direct solvers (see, for example, [20]).

In this paper, we have focused on demonstrating advantages of implicit schemes for basic picture processing tasks. The main mathematical contributions of the paper include the following:

- adapting Fourier–Padé–Galerkin approximations for designing implicit image differentiation and filtering schemes with good spectral resolution properties;
- establishing a simple link between implicit and explicit finite differences used for gradient estimation.

An alternative way to construct implicit differencing and filtering schemes consists of using a generalized interpolation method [10] combined with B-splines and their generalizations [46]. In particular, it leads to schemes capable of delivering high-quality estimates of derivatives at arbitrary locations and not only at the sample points [1, 24]. Although our Fourier– Padé–Galerkin approach is less general, it seems more flexible if highly accurate estimates of derivatives are required at regularly sampled points.

One interesting application of our approach consists of using implicit finite differences within the highly popular histograms of oriented gradients (HOG) approach [17], where it was reported that smoothing introduced by common gradient estimation schemes decreases the performance of the HOG descriptor. In particular, our experiments on applying HOG to gait energy images (GEIs) for action recognition purposes [52] clearly demonstrate the advantages of using our high-resolution finite difference scheme (4.1), (4.5) over the standard central difference approximation (2.2) employed in [17].

To conclude, implicit image differentiation and filtering schemes deserve to be among the standard computational tools used by image processing researchers and practitioners.

Acknowledgment. The author would like to thank the anonymous reviewers of this paper for their valuable and constructive comments.

REFERENCES

- U. ALIM, T. MÖLLER, AND L. CONDAT, Gradient estimation revitalized, IEEE Trans. Visual. Comput. Graph., 16 (2010), pp. 1494–1503.
- [2] L. ALVAREZ, Images and PDE's, in ICAOS '96, Lecture Notes in Control and Inform. Sci. 219, Springer, London, 1996, pp. 3–14.
- [3] S. ANDO, Consistent gradient operators, IEEE Trans. Pattern Anal. Mach. Intell., 22 (2000), pp. 252–265.
- [4] G. AUBERT AND P. KORNPROBST, Mathematical Problems in Image Processing, 2nd ed., Springer, New York, 2006.
- [5] A. BELYAEV, On implicit image derivatives and their applications, in Proceedings of the 22nd British Machine Vision Conference (BMVC 2011), Dundee, UK, 2011.
- [6] A. BELYAEV AND H. YAMAUCHI, Implicit filtering for image and shape processing, in Proceedings of the 16th International Workshop on Vision, Modeling and Visualization (VMV 2011), Berlin, Germany, 2011, pp. 277–283.
- [7] M. J. BENTUM, T. MALZBENDER, AND B. B. LICHTENBELT, Frequency analysis of gradient estimators in volume rendering, IEEE Trans. Visual. Comput. Graph., 2 (1996), pp. 242–254.
- [8] P. BHAT, B. CURLESS, M. COHEN, AND C. L. ZITNICK, Fourier analysis of the 2D screened Poisson equation for gradient domain problems, in Proceedings of the 10th European Conference on Computer Vision (ECCV 2008), Vol. 2, 2008, pp. 114–128.
- [9] W. G. BICKLEY, Finite difference formulae for the square lattice, Quart. J. Mech. Appl. Math., 1 (1948), pp. 35–42.
- [10] T. BLU, P. THÉVENAZ, AND M. UNSER, Generalized interpolation: Higher quality at no additional cost, in Proceedings of the IEEE International Conference on Image Processing (ICIP 1999), Kobe, Japan, 1999, pp. 667–671.
- G. BRADSKI AND A. KAEHLER, Learning OpenCV: Computer Vision with the OpenCV Library, O'Reilly, Sebastopol, CA, 2008.
- J. CALDER AND A. MANSOURI, Anisotropic image sharpening via well-posed Sobolev gradient flows, SIAM J. Math. Anal., 43 (2011), pp. 1536–1556.
- [13] J. CALDER, A. MANSOURI, AND A. YEZZI, Image sharpening via Sobolev gradient flows, SIAM J. Imaging Sci., 3 (2010), pp. 981–1014.
- [14] J. CALDER, A. MANSOURI, AND A. YEZZI, New possibilities in image diffusion and sharpening via highorder Sobolev gradient flows, J. Math. Imaging Vision, 40 (2011), pp. 248–258.
- [15] L. COLLATZ, The Numerical Treatment of Differential Equations, 3rd ed., Springer, Berlin, 1960.
- [16] T. COLONIUS AND S. K. LELE, Computational aeroacoustics: Progress on nonlinear problems of sound generation, Progr. Aero. Sci., 40 (2004), pp. 345–416.
- [17] N. DALAI AND B. TRIGGS, Histograms of oriented gradients for human detection, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2005), San Diego, CA, 2005, pp. 886–893.
- [18] E. R. DAVIES, Circularity a new principle underlying the design of accurate edge orientation operators, Image Vis. Comput., 2 (1984), pp. 134–142.

- [19] H. FARID AND E. P. SIMONCELLI, Differentiation of multi-dimensional signals, IEEE Trans. Image Process., 13 (2004), pp. 496–508.
- [20] G. E. FORSYTHE AND C. B. MOLER, Computer Solution of Linear Algebraic Systems, Prentice-Hall, Englewood Cliffs, NJ, 1967.
- [21] W. FREI AND C. C. CHEN, Fast boundary detection: A generalisation and a new algorithm, IEEE Trans. Comput., C-26 (1997), pp. 988–998.
- [22] R. C. GONZALEZ AND R. E. WOODS, *Digital Image Processing*, 3rd ed., Prentice–Hall, Englewood Cliffs, NJ, 2008.
- [23] R. W. HAMMING, Digital Filters, 3rd ed., Dover, New York, 1998.
- [24] Z. HOSSAIN, U. R. ALIM, AND T. MÖLLER, Toward high-quality gradient estimation on regular lattices, IEEE Trans. Visual. Comput. Graph., 17 (2011), pp. 426–439.
- [25] B. JÄHNE, H. SCHARR, AND S. KÖRKEL, Principles of filter design, in Handbook of Computer Vision and Applications, Vol. 2, Academic Press, New York, 1999, pp. 125–151.
- [26] J. W. KIM, High-order compact filters with variable cut-off wavenumber and stable boundary treatment, Comput. & Fluids, 39 (2010), pp. 1168–1182.
- [27] J. W. KIM AND D. J. LEE, Optimized compact finite difference schemes with maximum resolution, AIAA J., 34 (1996), pp. 887–893.
- [28] P. D. KOVESI, MATLAB and Octave Functions for Computer Vision and Image Processing, School of Computer Science & Software Engineering, The University of Western Australia, Perth; available online from http://www.csse.uwa.edu.au/~pk/research/matlabfns/.
- [29] A. KUMAR, Isotropic finite-differences, J. Comput. Phys., 201 (2004), pp. 109–118.
- [30] S. K. LELE, Compact finite difference schemes with spectral-like resolution, J. Comput. Phys., 103 (1992), pp. 16–42.
- [31] J. LI AND Y.-T. CHEN, Computational Partial Differential Equations Using MATLAB, CRC Press, Boca Raton, FL, 2009.
- [32] J. S. LIM, Two-Dimensional Signal and Image Processing, Prentice-Hall, Englewood Cliffs, NJ, 1990.
- [33] W.-S. LU AND A. ANTONIOU, Two-Dimensional Digital Filters, CRC Press, Boca Raton, FL, 1992.
- [34] M. S. MIN, S. M. KABER, AND W. S. DON, Fourier-Padé approximations and filtering for spectral simulations of an incompressible Boussinesq convection problem, Math. Comp., 76 (2007), pp. 1275– 1290.
- [35] C. MULAT, M. DONIAS, P. BAYLOU, G. VIGNOLES, AND C. GERMAIN, Optimal orientation estimators for detection cylindrical objects, Signal, Image and Video Processing, 2 (2008), pp. 51–58.
- [36] P. PERONA AND J. MALIK, Scale-space and edge detection using anisotropic diffusion, IEEE Trans. Pattern Anal. Mach. Intell., 12 (1990), pp. 629–639.
- [37] P. PERONA, T. SHIOTA, AND J. MALIK, Anisotropic diffusion, in Geometry-Driven Diffusion in Computer Vision, Kluwer Academic Publishers, Dordrecht, The Netherlands, 1994, pp. 73–92.
- [38] T. PETRILA AND D. TRIF, Basics of Fluid Mechanics and Introduction to Computational Fluid Dynamics, Springer, New York, 2005.
- [39] W. K. PRATT, Digital Image Processing, 3rd ed., John Wiley & Sons, New York, 2001.
- [40] W. H. RAYMOND, High-order low-pass implicit tangent filters for use in finite area calculations, Mon. Weather Rev., 116 (1988), pp. 2132–2141.
- [41] W. H. RAYMOND AND A. GARDER, A review of recursive and implicit filters, Mon. Weather Rev., 119 (1991), pp. 477–495.
- [42] H. SCHARR, S. KÖRKEL, AND B. JÄHNE, Numerische isotropieoptimierung von fir-filtern mittels querglättung, in Proceedings of Mustererkennung 1997 (DAGM 1997), 1997, pp. 367–374.
- [43] E. P. SIMONCELLI, Design of multi-dimensional derivative filters, in Proceedings of the IEEE International Conference on Image Processing (ICIP 1994), Vol. I, 1994, pp. 790–794.
- [44] C. K. W. TAM AND J. C. WEBB, Dispersion-relation-preserving finite difference schemes for computational acoustics, J. Comput. Phys., 107 (1993), pp. 262–281.
- [45] P. THÉVENAZ, T. BLU, AND M. UNSER, Image interpolation and resampling, in Handbook of Medical Imaging, Processing and Analysis, Academic Press, New York, 2000, pp. 393–420.
- [46] M. UNSER AND A. ALDROUBI, Polynomial spline signal processing algorithms, in Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 1992), Vol. 3, 1992, pp. 177–180.

- [47] M. UNSER, A. ALDROUBI, AND M. EDEN, B-spline signal processing. I. Theory, IEEE Trans. Signal Process., 41 (1993), pp. 821–833.
- [48] M. UNSER, A. ALDROUBI, AND M. EDEN, B-spline signal processing. II. Efficiency design and applications, IEEE Trans. Signal Process., 41 (1993), pp. 834–848.
- [49] Y. E. WANG, Y. LUO, AND M. N. ALFARAJ, Enhanced Isotropic 2D and 3D Gradient Method, U.S. Patent 7,653,258, 2010.
- [50] Z. WANG, A. C. BOVIK, AND H. R. SHEIKH, Image quality assessment: From error visibility to structural similarity, IEEE Trans. Image Process., 13 (2004), pp. 600–612.
- [51] J. WEICKERT AND H. SCHARR, A scheme for coherence-enhancing diffusion filtering with optimized rotation invariance, J. Vis. Comm. Image Represent., 13 (2002), pp. 103–118.
- [52] T. WHYTOCK, A. BELYAEV, AND N. ROBERTSON, GEI + HOG for action recognition, in 4th UK Computer Vision Student Workshop (BMVC 2012 Student Workshop), Surrey, UK, 2012; available online from http://www.bmva.org/bmvc/2012/WS/paper2.pdf.
- [53] H. WINNEMÖLLER, XDoG: Advanced image stylization with eXtended Difference-of-Gaussians, in Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Non-Photorealistic Animation and Rendering, 2011, pp. 147–156.