# Locally Analytic Schemes: a Link between Diffusion Filtering and Wavelet Shrinkage

Martin Welk[1], Gabriele Steidl[2], Joachim Weickert[1].

[1] Mathematical Image Analysis Group, Faculty of Mathematics and Computer Science, Bldg. E11, Saarland University, 66041 Saarbrücken, Germany
{welk|weickert}@mia.uni-saarland.de – http://www.mia.uni-saarland.de

[2] Faculty of Mathematics and Computer Science, D7, 27, University of Mannheim, 68131 Mannheim, Germany
steidl@math.uni-mannheim.de – http://kiwi.math.uni-mannheim.de

**Abstract**

We study a class of numerical schemes for nonlinear diffusion filtering that offers insights on the design of novel wavelet shrinkage rules for isotropic and anisotropic image enhancement. These schemes utilise analytical or semi-analytical solutions to dynamical systems that result from space-discrete nonlinear diffusion filtering on minimalistic images with $2\times2$ pixels. We call them *locally analytic schemes (LAS)* and *locally semi-analytic schemes (LSAS)*, respectively. They can be motivated from discrete energy functionals, offer sharp edges due to their locality, are very simple to implement because of their explicit nature, and enjoy unconditional absolute stability. They are applicable to singular nonlinear diffusion filters such as TV flow, to bounded nonlinear diffusion filters of Perona–Malik type, and to tensor-driven anisotropic methods such as edge-enhancing or coherence-enhancing diffusion filtering. The fact that these schemes use processes within $2 \times 2$-pixel blocks allows to connect them to shift-invariant Haar wavelet shrinkage on a single scale. This interpretation leads to novel shrinkage rules for two- and higher-dimensional images that are scalar-, vector- or tensor-valued. Unlike classical shrinkage strategies they employ a diffusion-inspired coupling of the wavelet channels that guarantees an approximation with an excellent degree of rotation invariance. By extending these schemes from a single scale to a multi-scale setting, we end up at hybrid methods that demonstrate the possibility to realise the effects of the most sophisticated diffusion filters within a fairly simplistic wavelet setting that requires only Haar wavelets in conjunction with coupled shrinkage rules.

**Keywords**

Image processing, diffusion filtering, wavelet shrinkage, finite difference methods, dynamical systems, stability, rotation invariance.

## 1 Introduction

Methods for image denoising that respect discontinuities and other semantically important features are of fundamental importance in digital image processing. Nonlinear diffusion filters [44] and wavelet shrinkage [25] are two widely used classes of methods to achieve this goal.

Diffusion filtering is a continuous concept that relies on partial differential equations (PDEs). It can easily incorporate invariances such as shift and rotation invariance. In order to be applicable to digital images, diffusion filters require discretisations. Typical numerical schemes proceed with a certain time step size and are iterated in order to end up with some result worth striving for. Anisotropic filters with a diffusion tensor instead of a scalar-valued diffusivity have been proposed to allow direction-dependent filtering along image edges and to enhance coherent, flow-like structures [59].

Wavelet shrinkage, on the other hand, uses specific discrete filters with vanishing moments. Usually it is used in a noniterative way that achieves its denoising capabilities by means of a multi-scale setting. Incorporating shift invariance and

approximating rotation invariance requires some extra efforts; see e.g. [18, 35]. Also in the wavelet community many efforts have been made to incorporate anisotropy in order to represent and process oriented structures in a better way, e.g. by *contourlets* [22], *ridgelets* [24], *curvelets* [10, 9] and *shearlets* [36]. In general they form frame elements that exhibit very high directional sensitivity and are highly anisotropic.

Since PDE-based methods and wavelet shrinkage serve the same purpose, it is not surprising that intensive research has been performed on analysing connections between these filters. Most of them analyse the *continuous* framework [4, 7, 13, 14, 17, 16, 40, 51, 52] or focus on designing methods that use wavelet shrinkage and PDE-based denoising methods in combination [11, 15, 19, 28, 27, 38, 39].

Work on the relations between wavelet shrinkage and PDE-based denoising in the *discrete* framework includes a paper by Coifman and Sowa [20] where they proposed total variation (TV) diminishing flows that act along the direction of Haar wavelets. In our previous work [55] we have established connections between space-discrete diffusion filtering and shift-invariant Haar wavelet shrinkage in the *one-dimensional* case. The starting point of this analysis was the consideration of a simplistic signal with only two pixels. It allowed to derive an identical analytic solution for both processes. This local analytic solution formed the building block of a numerical scheme for 1-D signals of arbitrary length. We call such methods *locally analytic schemes (LAS)*.

The goal of the present paper is to derive and investigate locally analytic schemes in the *higher-dimensional* setting. For several reasosns, the higher-dimensional case is significantly more complicated than the 1-D scenario: First of all, in two and more dimensions one has more degrees of freedom in the selection of specific discretisations for diffusion filters. Moreover, specific problems arise that are not present in the 1-D case: They include questions on how to design filters with a high degree of rotation invariance. Another challenge consists in the fact that it is possible to design anisotropic filters that adapt their directional behaviour to the underlying image structure. They are notoriously difficult to discretise in the diffusion setting, since discretisations usually suffer from blurring caused by numerical dissipation, and sometimes even from stability problems.

The locally analytic schemes that we consider in the present paper are based on an analysis of images of size $2 \times 2$ pixels. While it is possible to derive analytical solutions for the corresponding dynamical systems in the isotropic case for specific singular diffusivities such as the total variation diffusivity, this cannot be done for the widely used bounded diffusivities, and it is also not possible for the diffusion tensors of anisotropic diffusion filters. However, if one assumes that the diffusivity or the diffusion tensor does not change in time, it is possible to find an analytic solution for the corresponding dynamical system. We call numerical techniques that use these processes as building blocks *locally semi-analytic schemes (LSAS)*. We shall see that both LAS and LSAS are numerical methods that offer a number of interesting properties.

On the wavelet side, decompositions of an image into four-pixel blocks inspire a way to express a diffusion filter in terms of Haar wavelet shrinkage on the finest scale. In contrast to the 1-D setting, however, higher-dimensional diffusion filtering gives rise to novel shrinkage rules that employ a coupling of the different wavelet channels. Such a coupling provides an elegant way to incorporate rotation invariance into the wavelet shrinkage, since it approximates a rotationally invariant continuous diffusion process. Moreover the shrinkage rules can be incorporated into the typical wavelet multi-scale framework.

**Organisation of the paper.** Our paper is organised as follows. In Section 2 we briefly sketch the basic structure of nonlinear diffusion filters, and we survey the essential properties of the locally analytic two-pixel solutions in the 1-D case. We start our two-dimensional considerations in Section 3, where we derive the dynamical four-pixel systems for space-discrete nonlinear diffusion filters as gradient descent of discretised energy functionals. This is done in the isotropic case with a scalar-valued diffusivity and in the anisotropic setting with a diffusion tensor. In Section 4 we investigate these dynamical systems in detail and derive their solutions that give rise to locally (semi-)analytic schemes for nonlinear diffusion filtering. Section 5 is devoted to establishing formal connections between these numerical schemes and shift-invariant Haar wavelet shrinkage in the single-scale scenario, and it presents experiments in the single scale and multi-scale setting. The paper is concluded with a summary in Section 6.

**Related work.** The idea to split up a diffusion process into local interactions has also proved to be fruitful in other fields. In the context of fluid dynamic problems, related schemes have been formulated by Richardson, Ferrell, and Long [48]. These authors, however, use multiplicative splittings; i.e. they first compute an analytic solution for a given partition into small cells. The result is then used as the initial state for a subsequent diffusion in a shifted cell partitioning. In a general nonlinear setting, such a scheme would not be shift invariant. Our approach computes the diffusion of all four-pixel cells in parallel and averages afterwards. This additive splitting guarantees shift invariance. The splitting into four-pixel interactions also distinguishes our schemes from other additive operator splittings such as the ones in [37, 62]. They use directional splittings along the coordinate axis.

We notice that constructing numerical methods for PDE-based filters from analytic solutions of simpler systems is also a feature of the method of short-time kernels, see e.g. [54], where a locally linearised diffusion equation is solved by Gaussian convolution. For singular diffusion equations, our LAS approaches can also be regarded as alternatives to the finite difference schemes in [8], the finite element scheme in [29], and the level set approach considered in [21].

Early schemes for anisotropic, tensor-driven diffusion filtering did not pay specific attention to the problem of rotation invariance and avoidance of blurring artifacts [33, 47, 59]. Weickert and Scharr [61] addressed these problems by a scheme for coherence-enhancing diffusion filtering that uses optimised, Sobel-like approximations of all first order spatial derivatives. However, no stability theory was presented, and experiments showed only conditional stability. The same holds for the modified scheme of Wang [57] who used Simoncelli's derivative approximations [53] instead. Moreover, both schemes require stencil sizes of at least $5 \times 5$ pixels, while the scheme in the present paper is unconditionally absolutely stable and comes down to a more local $3 \times 3$ stencil.

With respect to novel, diffusion-inspired shrinkage rules for Haar wavelet shrinkage, our work extends results of Mrázek et al. for the 1-D case [42] and the isotropic 2-D framework [41] to the anisotropic 2-D setting. Moreover, our approach is based on absolutely stable locally (semi-)analytic schemes, while the relations in [42, 41] are established for explicit (Euler foward) discretisations that have to respect fairly strict stability limits.

We have reported some preliminary results with locally (semi-)analytic four-pixel schemes in two recent conference papers, one for the isotropic case [63], and one for the more complicated anisotropic framework [64]. In our present paper we extend these conference contributions in several ways: We introduce a unified framework for both isotropic and anisotropic processes. This framework is derived from discrete variational principles and covers a larger class of discretisations than

the ones in [63, 64]. We also generalise them to higher-dimensional and vector-valued data sets, and we consider connections to shift-invariant Haar wavelet shrinkage not only in the single scale case, but extend our experiments to the practically relevant multi-scale setting as well.

# 2 Prerequisites

In this section we sketch the basic structure of nonlinear diffusion filtering, and we review a LAS method in the 1-D case. These concepts are essential for the subsequent considerations.

## 2.1 Nonlinear Diffusion Filtering

**Isotropic nonlinear diffusion.** Let $\Omega \subset \mathbb{R}^2$ denote a rectangular image domain. Then an isotropic nonlinear diffusion filter[1] [44, 12] regards a greyscale image $f : \Omega \to \mathbb{R}$ as initial state of the diffusion equation

$$\partial_t u = \operatorname{div}\left(g(|\nabla u_\sigma|^2)\,\nabla u\right) \qquad \text{on } \Omega \times (0,\infty) \tag{1}$$

with homogeneous Neumann boundary conditions:

$$\partial_n u = 0 \qquad \text{on } \partial\Omega \times (0,\infty)\ . \tag{2}$$

Here $\nabla = (\partial_x, \partial_y)^{\mathrm{T}}$ and div denote the spatial gradient and divergence operators, resp., and $n$ is the outer unit normal to the image boundary $\partial\Omega$. The diffusion time $t$ steers the amount of simplification of the evolving image $u(x, y, t)$. The *diffusivity function* $g(|\nabla u_\sigma|^2)$ is a nonincreasing nonnegative function, and $u_\sigma$ denotes a smoothed image $u_\sigma := K_\sigma * u$ with a Gaussian $K_\sigma$ of standard deviation $\sigma$ (or $u_\sigma = u$ for $\sigma = 0$). If the diffusivity decreases in its gradient magnitude, blurring of edges is inhibited while diffusion within flat areas is encouraged. Incorporating Gaussian smoothing within the diffusivity may be used for rendering the filter more robust under noise [12].

It is not surprising that the choice of a suitable diffusivity function has a strong impact on the result of the diffusion filter. Often one uses bounded diffusivities such as [44, 59]

$$g(|\nabla u_\sigma|^2) = \frac{1}{1 + |\nabla u_\sigma|^2 / \lambda^2}\ , \tag{3}$$

$$g(|\nabla u_\sigma|^2) = \begin{cases} 1\ , & |\nabla u_\sigma|^2 = 0\ , \\ 1 - \exp(-3.31488\,\lambda^8 / |\nabla u_\sigma|^8)\ , & |\nabla u_\sigma|^2 > 0\ , \end{cases} \tag{4}$$

where $\lambda > 0$ serves as contrast parameter: It provides a threshold for distinguishing between edges ($|\nabla u_\sigma| > \lambda$) and the interior of a region ($|\nabla u_\sigma| < \lambda$).

More recently there has been a growing interest in unbounded diffusivities that become singular in zero [3, 21, 29, 34, 45]. This includes the family

$$g(|\nabla u|^2) = \frac{1}{|\nabla u|^p} \qquad (p \geq 0)\ . \tag{5}$$

These diffusivities offer the advantage that they do not require to tune any image specific contrast parameters (note also that no pre-smoothing is involved here, so

[1] Sometimes these processes are already denoted as *anisotropic*; see e.g. [44]. In our nomenclature, this notion is reserved for processes driven by matrix-valued diffusion tensors.

$\sigma = 0$). Moreover, they lead to scale invariant filters [1], for which even some analytical results have been established [56]. For $p = 1$ one obtains the *total variation (TV)* diffusion [3, 21], the diffusion filter that corresponds to TV minimisation [49]. TV diffusion offers a number of interesting properties such as finite extinction time [2] and shape-preserving qualities [5]. For $p > 1$ the diffusion does not only preserve edges but may even enhance them. A diffusivity with $p = 2$ has been considered in [34] for the so-called *balanced forward–backward (BFB)* diffusion filtering.

**Anisotropic nonlinear diffusion.** In a number of applications it is desirable to apply diffusion filters that allow direction-dependent behaviour. Such anisotropic filters require a matrix-valued diffusion tensor instead of a scalar-valued diffusivity. A class of anisotropic nonlinear diffusion equations is given by [59]

$$\partial_t u = \operatorname{div}(D(J) \cdot \nabla u) \,, \tag{6}$$

where $D(J)$ is an anisotropic diffusion tensor that depends on the image via the *structure tensor* [31]

$$J = J_\varrho(\nabla u_\sigma) := K_\varrho * (\nabla u_\sigma \, \nabla u_\sigma^{\mathrm{T}}) \,. \tag{7}$$

The structure tensor is a positive semidefinite matrix whose eigenvectors point in the preferred local image directions, and the eigenvalues measure the contrast along these directions. As before, $K_\varrho$ denotes a Gaussian and $u_\sigma = K_\sigma * u$. By varying the scales $\varrho$ and $\sigma$ and the way that $D$ depends on the structure tensor $J$, this equation can be adjusted to model a large class of anisotropic diffusion processes, including edge-enhancing diffusion (EED) and coherence-enhancing diffusion (CED) [59]. They will be described in more detail in Section 4.

## 2.2 LAS in the One-Dimensional Case

In [55], discretisations of total variation (TV) diffusion on 1-D signals have been investigated. The following results are of particular interest here because the present paper provides 2-D analogs for them.[2]

1. On a very simplistic signal $(f_1, f_2)$ consisting of only two pixels, it is possible to state an analytic solution for the TV flow equation. As opposed to the use of reflecting boundary conditions in [55] we give this solution here for a two-pixel signal $(u_1, u_2)$ with *periodic* boundary conditions. Assuming a spatial grid size of $h = 1$ (as is common in signal processing) the TV flow equation then reduces to $\dot{u}_1 = 2\operatorname{sgn}(u_2 - u_1)$, $\dot{u}_2 = 2\operatorname{sgn}(u_1 - u_2)$ and has the analytic solution

$$u_i(t) = \begin{cases} \mu + (1 - 4t/\left|f_2 - f_1\right|)(f_i - \mu) \,, & 0 \le t < \left|f_2 - f_1\right|/4 \,, \\ \mu \,, & t \ge \left|f_2 - f_1\right|/4 \,, \end{cases} \tag{8}$$

   where $\mu := (f_1 + f_2)/2$. (In fact, in [55] even an analytic solution for $N$-pixel signals with reflecting boundary conditions was given.)

2. An unconditionally absolutely stable and computationally simple numerical scheme for 1-D TV diffusion can be based on these two-pixel analytic solutions. One time step of this scheme for pixel $i$ reads as follows, assuming that $u_j^k$ is the value of pixel $j$ in the old time step $k$:

   - Consider the two-pixel cells $(i-1, i)$ and $(i, i+1)$ containing pixel $i$.

[2] In [55] also 1-D TV regularisation is studied, for which a related algorithm had already been developed in [46].

- For each of these cells, compute the analytic solution (8), with $(u_{i-1}^k, u_i^k)$ and $(u_i^k, u_{i+1}^k)$, resp., as initial values, up to the desired time step size. For pixel $i$, this gives two new values $u_{i,-}^{k+1}$ and $u_{i,+}^{k+1}$.
- Average: $u_i^{k+1} := \frac{1}{2}(u_{i,-}^{k+1} + u_{i,+}^{k+1})$.

3. The novel numerical scheme is equivalent to shift-invariant soft Haar wavelet shrinkage with threshold depending on the diffusion time, if the Haar wavelet shrinkage process is limited to a single (the finest) scale.

We remark also that these results possess obvious generalisations to the case of singular 1-D nonlinear diffusion processes with the diffusivities (5).

# 3 Discrete Variational Models

While many diffusion filters can be regarded as steepest descent methods of continuous energy functionals, it is less common to derive their space-discrete approximations from discrete energy functionals. Since this, however, is a conceptually clean way of incorporating a correct treatment of boundary conditions and of showing optimality of the discrete solution, we prefer this approach. First we treat the isotropic case, then we extend the framework to the anisotropic setting. Throughout this section we assume a grid size in $x$- and $y$-direction of $h = 1$.

## 3.1 The Isotropic Case

Let the image $u$ be discretised on a rectangular grid of grid size 1 and with integer coordinates.

It is known from the continuous theory that diffusion processes of type (1) with $\sigma = 0$, i.e. without pre-smoothing of the gradient field, can be represented as gradient descents for energy functionals of the type[3]

$$E[u] = \frac{1}{2} \int_\Omega \Psi(|\nabla u|^2) \, \mathrm{d}x \, \mathrm{d}y \; . \tag{9}$$

Here, $\Psi$ denotes an increasing function on the interval $[0, \infty)$ with $\Psi' = g$.

**Discrete energy function.** We want to generate a *discrete energy function* by discretising (9) in a way that introduces as little numerical blurring as possible, and compute the corresponding discrete gradient descent. The best locations to discretise the integrand most locally are inter-pixel points $(i + \frac{1}{2}, j + \frac{1}{2})$ where $i$, $j$ are integers. In this case, it is possible to approximate the squared gradient magnitude using only the four surrounding pixels. Our discrete energy function will therefore have the form

$$E(u) := \frac{1}{2} \sum_{i,j} \Psi\left( [|\nabla u|^2]_{i+\frac{1}{2}, j+\frac{1}{2}} \right) \; , \tag{10}$$

where $[|\nabla u|^2]_{i+\frac{1}{2}, j+\frac{1}{2}}$ is some discretisation of the squared gradient magnitude $|\nabla u|^2$ within the four-pixel cell $\{i, i+1\} \times \{j, j+1\}$, and the outer sum runs over all such four-pixel cells.

[3]For $\sigma \neq 0$, no energy functional is known that has (1) as gradient descent.

**Discretisation of $|\nabla u|^2$.** It remains to discretise the squared gradient magnitude $|\nabla u|^2$ at cell midpoints $(i+\frac{1}{2}, j+\frac{1}{2})$. We will do this in a twofold way. For the sake of later generalisation, we set $v = u$. To simplify notation, we fix $i = j = 1$.

We have in standard coordinates $|\nabla v|^2 = (\partial_x v)^2 + (\partial_y v)^2$. We can approximate $\partial_x v$ and $\partial_y v$ by arithmetic means of central difference approximations,

$$\begin{aligned} (\partial_x v)_{\frac{3}{2},\frac{3}{2}} &\approx \tfrac{1}{2}(v_{2,2} + v_{2,1} - v_{1,2} - v_{1,1}) \,, \\ (\partial_y v)_{\frac{3}{2},\frac{3}{2}} &\approx \tfrac{1}{2}(v_{2,2} - v_{2,1} + v_{1,2} - v_{1,1}) \end{aligned} \tag{11}$$

which leads to

$$\big(|\nabla v|^2\big)_{\frac{3}{2},\frac{3}{2}} \approx \tfrac{1}{2}\big((v_{2,2} - v_{1,1})^2 + (v_{2,1} - v_{1,2})^2\big) \,. \tag{12}$$

Since, however, computing the gradient magnitude in fact relies on the *squares* of the partial derivatives, we can replace the averaging of central difference approximations of the derivatives themselves in (11) by averaging the squared derivatives, which gives

$$\begin{aligned} \big((\partial_x v)^2\big)_{\frac{3}{2},\frac{3}{2}} &\approx \tfrac{1}{2}\big((v_{2,2} - v_{1,2})^2 + (v_{2,1} - v_{1,1})^2\big) \,, \\ \big((\partial_y v)^2\big)_{\frac{3}{2},\frac{3}{2}} &\approx \tfrac{1}{2}\big((v_{2,2} - v_{2,1})^2 + (v_{1,2} - v_{1,1})^2\big) \,, \end{aligned} \tag{13}$$

and thereby a new approximation for the gradient magnitude,

$$\begin{aligned} \big(|\nabla v|^2\big)_{\frac{3}{2},\frac{3}{2}} \approx \tfrac{1}{2}\big( &\quad (v_{2,2} - v_{1,2})^2 + (v_{2,1} - v_{1,1})^2 \\ &+ (v_{2,2} - v_{2,1})^2 + (v_{1,2} - v_{1,1})^2\big) \,. \end{aligned} \tag{14}$$

Each of the approximations (12), (14) and their convex combinations

$$\begin{aligned} \big(|\nabla v|^2\big)_{\frac{3}{2},\frac{3}{2}} &\approx (G_\alpha(v)_{\frac{3}{2},\frac{3}{2}})^2 \\ &:= \frac{\alpha}{2}\Big((v_{2,2} - v_{1,2})^2 + (v_{2,1} - v_{1,1})^2 \\ &\qquad + (v_{2,2} - v_{2,1})^2 + (v_{1,2} - v_{1,1})^2\Big) \\ &\quad + \frac{1-\alpha}{2}\Big((v_{2,2} - v_{1,1})^2 + (v_{2,1} - v_{1,2})^2\Big) \,, \end{aligned} \tag{15}$$

$\alpha \in [0,1]$, can be used in (10).

**Discrete gradient descent.** Via the chain rule we compute the partial derivative of the energy $E$ with respect to some particular variable $u_{i,j}$:

$$\frac{\partial E}{\partial u_{i,j}} = \frac{1}{2}\sum_{i',j'} \Psi'\left([|\nabla u|^2]_{i'+\frac{1}{2},j'+\frac{1}{2}}\right) \frac{\partial [|\nabla u|^2]_{i'+\frac{1}{2},j'+\frac{1}{2}}}{\partial u_{i,j}} \tag{16}$$

where $[|\nabla u|^2]_{i+\frac{1}{2},j+\frac{1}{2}}$ denotes the chosen discretisation of $|\nabla u|^2$ at $(i+\frac{1}{2}, j+\frac{1}{2})$. Then the corresponding gradient descent is

$$\dot{u}_{i,j} = -\frac{\partial E}{\partial u_{i,j}} \quad \text{for all } i,j \,, \tag{17}$$

where we have denoted differentiation with respect to the time $t$ by a dot. For given $(i,j)$, the sum on the right-hand side has four possibly nonzero contributions belonging to the cells $2 \times 2$-pixel cells

$$\begin{aligned} (--)&: \quad \{i-1,i\} \times \{j-1,j\} \,, \qquad (+-): \quad \{i,i+1\} \times \{j-1,j\} \,, \\ (-+)&: \quad \{i-1,i\} \times \{j,j+1\} \,, \qquad (++): \quad \{i,i+1\} \times \{j,j+1\} \,, \end{aligned} \tag{18}$$

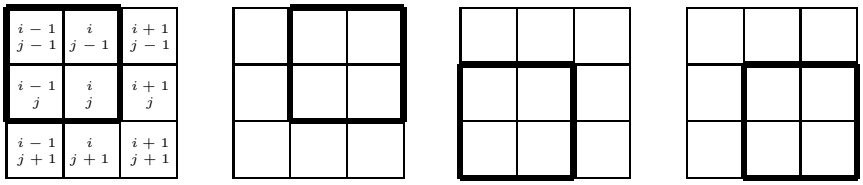



Figure 1: The four-pixel cells contributing to $u_{i,j}$.

which contain the pixel $(i, j)$; see also Figure 1.

It remains to compute $\partial[|\nabla u|^2]_{i'+\frac{1}{2},j'+\frac{1}{2}}/\partial u_{i,j}$. With the discretisation (15) we have

$$\begin{aligned}
\frac{\partial[|\nabla u|^2]_{(--)}}{\partial u_{i,j}} &= \frac{\alpha}{2}(2u_{i,j} - u_{i-1,j} - u_{i,j-1}) + \frac{1-\alpha}{2}(u_{i,j} - u_{i-1,j-1}) \,, \\
\frac{\partial[|\nabla u|^2]_{(+-)}}{\partial u_{i,j}} &= \frac{\alpha}{2}(2u_{i,j} - u_{i+1,j} - u_{i,j-1}) + \frac{1-\alpha}{2}(u_{i,j} - u_{i+1,j-1}) \,, \\
\frac{\partial[|\nabla u|^2]_{(-+)}}{\partial u_{i,j}} &= \frac{\alpha}{2}(2u_{i,j} - u_{i-1,j} - u_{i,j+1}) + \frac{1-\alpha}{2}(u_{i,j} - u_{i-1,j+1}) \,, \\
\frac{\partial[|\nabla u|^2]_{(++)}}{\partial u_{i,j}} &= \frac{\alpha}{2}(2u_{i,j} - u_{i+1,j} - u_{i,j+1}) + \frac{1-\alpha}{2}(u_{i,j} - u_{i+1,j+1}) \,,
\end{aligned} \tag{19}$$

where the indices $(--)$, $(+-)$, $(-+)$, $(++)$ refer to the cells (18), compare Figure 1.

Inserting these expressions into (17), we obtain the dynamical system

$$\begin{aligned}
\dot{u}_{i,j} = \tfrac{1}{2}\Big( \quad & \Psi'_{i-\frac{1}{2},j-\frac{1}{2}} \cdot (\alpha u_{i-1,j} + \alpha u_{i,j-1} + (1-\alpha)u_{i-1,j-1} - (1+\alpha)u_{i,j}) \\
& + \Psi'_{i+\frac{1}{2},j-\frac{1}{2}} \cdot (\alpha u_{i+1,j} + \alpha u_{i,j-1} + (1-\alpha)u_{i+1,j-1} - (1+\alpha)u_{i,j}) \\
& + \Psi'_{i-\frac{1}{2},j+\frac{1}{2}} \cdot (\alpha u_{i-1,j} + \alpha u_{i,j+1} + (1-\alpha)u_{i-1,j+1} - (1+\alpha)u_{i,j}) \\
& + \Psi'_{i+\frac{1}{2},j+\frac{1}{2}} \cdot (\alpha u_{i+1,j} + \alpha u_{i,j+1} + (1-\alpha)u_{i+1,j+1} - (1+\alpha)u_{i,j})\Big)
\end{aligned} \tag{20}$$

with $\Psi'_{i+\frac{1}{2},j+\frac{1}{2}} := \Psi'\big([|\nabla u|^2]_{i+\frac{1}{2},j+\frac{1}{2}}\big)$ as our final gradient descent equation.

**Neumann boundary conditions.** For a rectangular image domain equipped with a regular grid, Neumann boundary conditions are realised by mirroring the pixels of the marginal rows and columns of the image. In this setting, the gradient descent equations (20) can be applied without change to the boundary pixels, guaranteeing that no net flux across the image boundary takes place.

**Extension to multi-channel images.** The space-discrete framework developed here for grey-value images admits a straightforward extension to images whose values are tuples of real numbers, like colour images, vector- or matrix-valued images. This includes particularly tensor-valued images arising from diffusion tensor magnetic resonance imaging and several other applications in e.g. geology and civil engineering.

A discrete energy function for such a *multi-channel image* is given by

$$E(u) := \frac{1}{2} \sum_{i,j} \Psi \left( \sum_k [|\nabla u_k|^2]_{i+\frac{1}{2},j+\frac{1}{2}} \right) \tag{21}$$

where the inner summation extends over the image channels $u_k$, i.e. colour channels, vector components, or matrix entries, compare also [6] for the total variation case. For the discretisation of $|\nabla u_k|^2$ one simply uses (15) with $v = u_k$.

The corresponding gradient descent reads

$$\dot{u}_{k;i,j} = -\frac{\partial E}{\partial u_{k;i,j}} = -\frac{1}{2} \sum_{i',j'} \Psi'\Big( \sum_l [|\nabla u_l|^2]_{i'+\frac{1}{2},j'+\frac{1}{2}} \Big) \frac{\partial [|\nabla u_k|^2]_{i'+\frac{1}{2},j'+\frac{1}{2}}}{\partial u_{k;i,j}} \,, \tag{22}$$

where $u_{k;i,j}$ refers to the value of image channel $k$ at pixel $(i, j)$. Note that the partial derivative $\partial [|\nabla u_k|^2]_{i'+\frac{1}{2},j'+\frac{1}{2}} / \partial u_{k;i,j}$ is computed within the image channel $k$, while the diffusivity $\Psi'(\cdot)$ aggregates in its argument all image channels and thereby couples the evolutions of all channels, compare [32, 58, 50].

## 3.2 The Anisotropic Case

Similar to the link between diffusion and energy minimisation mentioned at the begin of Subsection 3.1, there is also a relation between anisotropic diffusion processes with $\varrho = \sigma = 0$ and energy minimisation[4] [61]. However, for scalar-valued images (6) with $\varrho = \sigma = 0$ degenerates into an isotropic process. In contrast, these parameters still admit a truly anisotropic process for multi-channel images, like colour images, vector or matrix fields.

An energy functional for this case reads [60]

$$E[u] = \frac{1}{2} \int_\Omega \operatorname{tr} \Psi(J(\nabla u)) \, \mathrm{d}x \, \mathrm{d}y \tag{23}$$

with the structure tensor $J(\nabla u)$ that with $\varrho = \sigma = 0$ simplifies to the sum of outer products

$$J(\nabla u) = \sum_k \nabla u_k \nabla u_k^{\mathrm{T}} \,. \tag{24}$$

The summation here extends over all channels (colour channels, vector components or matrix entries) of the multi-channel image. By $\Psi$ we denote an increasing function on $\mathbb{R}_0^+$ which is applied to its symmetric positive semidefinite matrix argument as usual by acting on the eigenvalues and leaving the eigenvector system intact.

**Anisotropic discrete energy function.** Analog to the isotropic case, we want to discretise (23) into a discrete energy function and derive its space-discrete gradient descent. In order to reduce as much as possible numerical blurring effects, we choose similarly as in 3.1 the locations $(i + \frac{1}{2}, j + \frac{1}{2})$ with integers $i$ and $j$ for the discretisation of the structure tensor $\sum_k \nabla u_k \nabla u_k^{\mathrm{T}}$. Our energy function thus reads

$$E(u) := \frac{1}{2} \sum_{i,j} \operatorname{tr} \Psi \left( \sum_k [\nabla u_k \nabla u_k^{\mathrm{T}}]_{i+\frac{1}{2},j+\frac{1}{2}} \right) \tag{25}$$

where $[\nabla u_k \nabla u_k^{\mathrm{T}}]_{i+\frac{1}{2},j+\frac{1}{2}}$ denotes an approximation of the outer gradient product $\nabla u_k \nabla u_k^{\mathrm{T}}$ at $(i + \frac{1}{2}, j + \frac{1}{2})$. The outer sum runs over all these locations while the inner sum again extends over all channels.

[4] For $\varrho \neq 0$ and $\sigma \neq 0$, no energy functional is known that warrants (6) as gradient descent.

**Discretisations of the structure tensor.** It remains to discretise the structure tensor $J$ which we will again do in two ways. For abbreviation we set $v := u_k$ throughout this paragraph, and write down the discretisations for $i = j = 1$. The most direct approach to discretising $J$ is to discretise the gradients $\nabla v$ according to (11) from which we obtain the components of the outer product $\nabla v \nabla v^{\mathrm{T}}$:

$$\begin{aligned}
(\partial_x v)^2_{\frac{3}{2},\frac{3}{2}} &\approx \tfrac{1}{4}(v_{2,2} + v_{2,1} - v_{1,2} - v_{1,1})^2 \\
(\partial_y v)^2_{\frac{3}{2},\frac{3}{2}} &\approx \tfrac{1}{4}(v_{2,2} - v_{2,1} + v_{1,2} - v_{1,1})^2 \\
(\partial_x v \partial_y v)_{\frac{3}{2},\frac{3}{2}} &\approx \tfrac{1}{4}(v_{2,2} + v_{2,1} - v_{1,2} - v_{1,1})(v_{2,2} - v_{2,1} + v_{1,2} - v_{1,1}) \\
&= \tfrac{1}{4}\big((v_{2,2} - v_{1,1})^2 - (v_{2,1} - v_{1,2})^2\big) \; .
\end{aligned} \tag{26}$$

Summation over image channels yields the approximated structure tensor field.

As an alternative, one can use one-sided finite differences at each of the four pixels of the cell for approximation. At $(1,1)$, one obtains the approximations

$$\begin{aligned}
(\partial_x v)_{1,1} &\approx \tfrac{1}{2}(v_{2,1} - v_{1,1}) \; , \\
(\partial_y v)_{1,1} &\approx \tfrac{1}{2}(v_{1,2} - v_{1,1}) \; .
\end{aligned} \tag{27}$$

From these an approximation of the outer product $\nabla v \nabla v^{\mathrm{T}}$ at $(1,1)$ is calculated. At the other three pixels, analogous approximations are made using left- or right-sided differences such as to stay within the given four-pixel cell. Taking the arithmetic mean of the four outer product approximations yields the second-order approximation

$$\begin{aligned}
(\partial_x v)^2_{\frac{3}{2},\frac{3}{2}} &\approx \tfrac{1}{2}\big((v_{2,2} - v_{1,2})^2 + (v_{2,1} - v_{1,1})^2\big) \\
(\partial_y v)^2_{\frac{3}{2},\frac{3}{2}} &\approx \tfrac{1}{2}\big((v_{2,2} - v_{2,1})^2 + (v_{1,2} - v_{1,1})^2\big) \\
(\partial_x v \partial_y v)_{\frac{3}{2},\frac{3}{2}} &\approx \tfrac{1}{4}\big((v_{2,2} - v_{1,1})^2 - (v_{2,1} - v_{1,2})^2\big) \; .
\end{aligned} \tag{28}$$

Note that the two approximations (26) and (28) differ only in the main diagonal entries of the structure tensor.

Once more, one can also use convex combinations of (26) and (28), i.e.

$$\begin{aligned}
(\partial_x v)^2_{\frac{3}{2},\frac{3}{2}} &\approx \tfrac{1}{4}\big((v_{2,2} + v_{2,1} - v_{1,2} - v_{1,1})^2 + \alpha(v_{2,2} - v_{2,1} - v_{1,2} + v_{1,1})^2\big) \; , \\
(\partial_y v)^2_{\frac{3}{2},\frac{3}{2}} &\approx \tfrac{1}{4}\big((v_{2,2} - v_{2,1} + v_{1,2} - v_{1,1})^2 + \alpha(v_{2,2} - v_{2,1} - v_{1,2} + v_{1,1})^2\big) \; , \\
(\partial_x v \partial_y v)_{\frac{3}{2},\frac{3}{2}} &\approx \tfrac{1}{4}\big((v_{2,2} - v_{1,1})^2 - (v_{2,1} - v_{1,2})^2\big)
\end{aligned} \tag{29}$$

for $\alpha \in [0,1]$.

An easy calculation shows that the trace $(\partial_x v)^2_{\frac{3}{2},\frac{3}{2}} + (\partial_y v)^2_{\frac{3}{2},\frac{3}{2}}$ of the so discretised outer product equals the discretised square gradient magnitude (15) from the isotropic case. By summation over channels, the same holds for $\operatorname{tr} J$ and the squared gradient magnitude $\sum_k |\nabla u_k|^2$ of a multichannel image. Even the one-sided difference procedure used to obtain (28) could equally be used in the isotropic case to derive (13) in a different way.

**Gradient descent for anisotropic discrete energy.** According to the chain rule, the partial derivative of the energy $E$ with respect to some particular variable $u_{k;i,j}$ is given by

$$\frac{\partial E}{\partial u_{k;i,j}} = \frac{1}{2} \sum_{i',j'} \operatorname{tr} \left( \Psi' \left( \sum_l [\nabla u_l \nabla u_l^{\mathrm{T}}]_{i'+\frac{1}{2},j'+\frac{1}{2}} \right) \frac{\partial [\nabla u_k \nabla u_k^{\mathrm{T}}]_{i'+\frac{1}{2},j'+\frac{1}{2}}}{\partial u_{k;i,j}} \right) \; , \tag{30}$$

the corresponding gradient descent by

$$\dot{u}_{k;i,j} = -\frac{\partial E}{\partial u_{k;i,j}} \quad \text{for all } i,j,k\,. \tag{31}$$

For given $(i,j)$, the sum on the right-hand side has again four possibly nonzero contributions belonging to the cells (18); compare Figure 1. We need to compute $\partial[\nabla u_k \nabla u_k^{\mathrm{T}}]_{i'+\frac{1}{2},j'+\frac{1}{2}}/\partial u_{k;i,j}$. To simplify the equations, we do this separately for the two discretisations (26), (28). We also abbreviate $v := u_k$ in equations which do not contain explicit interactions of image channels.

First, by discretising the outer product according to (26), we find that

$$\begin{aligned}
\frac{\partial[\nabla v \nabla v^{\mathrm{T}}]_{(--)}}{\partial v_{i,j}} &= \frac{v_{i,j}-v_{i-1,j-1}}{2}\begin{pmatrix}1&1\\1&1\end{pmatrix} + \frac{v_{i,j-1}-v_{i-1,j}}{2}\begin{pmatrix}1&0\\0&-1\end{pmatrix}\,,\\
\frac{\partial[\nabla v \nabla v^{\mathrm{T}}]_{(+-)}}{\partial v_{i,j}} &= \frac{v_{i,j}-v_{i+1,j-1}}{2}\begin{pmatrix}1&1\\1&1\end{pmatrix} + \frac{v_{i,j-1}-v_{i+1,j}}{2}\begin{pmatrix}1&0\\0&-1\end{pmatrix}\,,\\
\frac{\partial[\nabla v \nabla v^{\mathrm{T}}]_{(-+)}}{\partial v_{i,j}} &= \frac{v_{i,j}-v_{i-1,j+1}}{2}\begin{pmatrix}1&1\\1&1\end{pmatrix} + \frac{v_{i,j+1}-v_{i-1,j}}{2}\begin{pmatrix}1&0\\0&-1\end{pmatrix}\,,\\
\frac{\partial[\nabla v \nabla v^{\mathrm{T}}]_{(++)}}{\partial v_{i,j}} &= \frac{v_{i,j}-v_{i+1,j+1}}{2}\begin{pmatrix}1&1\\1&1\end{pmatrix} + \frac{v_{i,j+1}-v_{i+1,j}}{2}\begin{pmatrix}1&0\\0&-1\end{pmatrix}
\end{aligned} \tag{32}$$

where we have used again the cell abbreviations from (18). By plugging this into the right-hand side of (31) and evaluate the trace expression, we obtain

$$\begin{aligned}
\dot{v}_{i,j} = \frac{1}{4}\big( &-(a_{--}+b_{--}+2c_{--})(v_{i,j}-v_{i-1,j-1})\\
&-(a_{--}-b_{--})(v_{i,j-1}-v_{i-1,j})\\
&+(a_{+-}-b_{+-})(v_{i+1,j}-v_{i,j-1})\\
&+(a_{+-}+b_{+-}-2c_{+-})(v_{i+1,j-1}-v_{i,j})\\
&-(a_{-+}-b_{-+})(v_{i,j+1}-v_{i-1,j})\\
&-(a_{-+}+b_{-+}-2c_{-+})(v_{i,j}-v_{i-1,j+1})\\
&+(a_{++}+b_{++}+2c_{++})(v_{i+1,j+1}-v_{i,j})\\
&+(a_{++}-b_{++})(v_{i+1,j}-v_{i,j+1})\big)\,.
\end{aligned} \tag{33}$$

where

$$\begin{aligned}
\Psi'\Big(\sum_k [\nabla u_k \nabla u_k^{\mathrm{T}}]_{i-\frac{1}{2},j-\frac{1}{2}}\Big) &=: \begin{pmatrix}a_{--}&c_{--}\\c_{--}&b_{--}\end{pmatrix}\,,\\
\Psi'\Big(\sum_k [\nabla u_k \nabla u_k^{\mathrm{T}}]_{i+\frac{1}{2},j-\frac{1}{2}}\Big) &=: \begin{pmatrix}a_{+-}&c_{+-}\\c_{+-}&b_{+-}\end{pmatrix}\,,\\
\Psi'\Big(\sum_k [\nabla u_k \nabla u_k^{\mathrm{T}}]_{i-\frac{1}{2},j+\frac{1}{2}}\Big) &=: \begin{pmatrix}a_{-+}&c_{-+}\\c_{-+}&b_{-+}\end{pmatrix}\,,\\
\Psi'\Big(\sum_k [\nabla u_k \nabla u_k^{\mathrm{T}}]_{i+\frac{1}{2},j+\frac{1}{2}}\Big) &=: \begin{pmatrix}a_{++}&c_{++}\\c_{++}&b_{++}\end{pmatrix}\,.
\end{aligned} \tag{34}$$

If we use instead (28) in discretising the structure tensor, (32) is replaced by

$$\begin{aligned}
\frac{\partial[\nabla v \nabla v^{\mathrm{T}}]_{(--)}}{\partial v_{i,j}} &= \frac{1}{2}\begin{pmatrix} 2(v_{i,j}-v_{i-1,j}) & v_{i,j}-v_{i-1,j-1} \\ v_{i,j}-v_{i-1,j-1} & 2(v_{i,j}-v_{i,j-1}) \end{pmatrix}, \\
\frac{\partial[\nabla v \nabla v^{\mathrm{T}}]_{(+-)}}{\partial v_{i,j}} &= \frac{1}{2}\begin{pmatrix} 2(v_{i,j}-v_{i+1,j}) & -v_{i,j}+v_{i+1,j-1} \\ -v_{i,j}+v_{i+1,j-1} & 2(v_{i,j}-v_{i,j-1}) \end{pmatrix}, \\
\frac{\partial[\nabla v \nabla v^{\mathrm{T}}]_{(-+)}}{\partial v_{i,j}} &= \frac{1}{2}\begin{pmatrix} 2(v_{i,j}-v_{i-1,j}) & -v_{i,j}+v_{i-1,j+1} \\ -v_{i,j}+v_{i-1,j+1} & 2(v_{i,j}-v_{i,j+1}) \end{pmatrix}, \\
\frac{\partial[\nabla v \nabla v^{\mathrm{T}}]_{(++)}}{\partial v_{i,j}} &= \frac{1}{2}\begin{pmatrix} 2(v_{i,j}-v_{i+1,j}) & v_{i,j}-v_{i+1,j+1} \\ v_{i,j}-v_{i+1,j+1} & 2(v_{i,j}-v_{i,j+1}) \end{pmatrix}
\end{aligned} \tag{35}$$

which implies as gradient descent

$$\begin{aligned}
\dot{v}_{i,j} = {} & \tfrac{1}{2}\big(-a_{--}(v_{i,j}-v_{i-1,j}) - b_{--}(v_{i,j}-v_{i,j-1}) - c_{--}(v_{i,j}-v_{i-1,j-1})\big) \\
& + \tfrac{1}{2}\big(\ \ a_{+-}(v_{i+1,j}-v_{i,j}) - b_{+-}(v_{i,j}-v_{i,j-1}) - c_{+-}(v_{i+1,j-1}-v_{i,j})\big) \\
& + \tfrac{1}{2}\big(-a_{-+}(v_{i,j}-v_{i-1,j}) + b_{-+}(v_{i,j+1}-v_{i,j}) + c_{-+}(v_{i,j}-v_{i-1,j+1})\big) \\
& + \tfrac{1}{2}\big(\ \ a_{++}(v_{i+1,j}-v_{i,j}) + b_{++}(v_{i,j+1}-v_{i,j}) + c_{++}(v_{i+1,j+1}-v_{i,j})\big)\ .
\end{aligned} \tag{36}$$

For a convex combination (29), the gradient descent will be the corresponding convex combination of (33) and (36).

**Neumann boundary conditions.** As in the isotropic case, Neumann boundary conditions for a rectangular domain can be ensured simply by mirroring boundary rows and columns. Note that the antidiagonal terms in both discretisations (26) and (28) of the structure tensor vanish at the midpoints of cells crossing the domain boundary, aligning the eigensystem of the structure tensor with the boundary tangent and normal directions. As a consequence, the gradient descent equations (33) and (36) admit no net flow across the image boundaries and ensure identical evolution for boundary pixels and their mirrored counterparts. This stands in contrast to other discretisations of anisotropic processes which require a special adjustment for boundary pixels.

# 4 Discretisations for Diffusion Filtering Using Four-Pixel Cells

In this section, we want to use the discretisations derived from the variational models to 2-D nonlinear diffusion processes in more generality. The discretisation of a diffusion process of type (1) or (6) consists of two components. First, there is a discretisation of the divergence form $\operatorname{div}(* \cdot \nabla u)$ of diffusion, where $*$ denotes either the diffusivity field $g$ or the diffusion tensor field $D$, which is treated in this context as "black box". Second, there is the discretisation of $g$ or $D$ itself. This distinction also makes sense since, in the general case, $g$ or $D$ does not depend on the gradient $\nabla u$ itself but on the smoothed version $\nabla u_\sigma$.

## 4.1 Isotropic Nonlinear Diffusion

We start with the isotropic nonlinear diffusion processes (1).

### 4.1.1 Space Discretisation

In accordance with the setting adopted in Subsection 3.1, we assume that the diffusivity field $g$ is discretised in the cell midpoints $(i+\frac{1}{2}, j+\frac{1}{2})$. In the variational model, $\Psi'$ plays the role of the diffusivity field $g$.

**Discretisation of the diffusivity field.** The discretisation of $g$ boils down to discretising the squared gradient magnitude $|\nabla u_\sigma|^2$ and applying the diffusivity function $g(\cdot)$ to this approximation. The equations (15) with $v = u_\sigma$ provide exactly the desired approximation.

**Discretisation of isotropic diffusion with given diffusivity.** With a prescribed diffusivity field, a discretisation of the divergence form results by simply substituting $\Psi'$ in (20) with $g$. For any $\alpha \in [0,1]$ this gives

$$\begin{aligned}\dot{u}_{i,j} = \tfrac{1}{2}\Big( \quad & g_{i-\frac{1}{2},j-\frac{1}{2}} \cdot (\alpha u_{i-1,j} + \alpha u_{i,j-1} + (1-\alpha)u_{i-1,j-1} - (1+\alpha)u_{i,j}) \\ & + g_{i+\frac{1}{2},j-\frac{1}{2}} \cdot (\alpha u_{i+1,j} + \alpha u_{i,j-1} + (1-\alpha)u_{i+1,j-1} - (1+\alpha)u_{i,j}) \\ & + g_{i-\frac{1}{2},j+\frac{1}{2}} \cdot (\alpha u_{i-1,j} + \alpha u_{i,j+1} + (1-\alpha)u_{i-1,j+1} - (1+\alpha)u_{i,j}) \\ & + g_{i+\frac{1}{2},j+\frac{1}{2}} \cdot (\alpha u_{i+1,j} + \alpha u_{i,j+1} + (1-\alpha)u_{i+1,j+1} - (1+\alpha)u_{i,j})\Big)\end{aligned} \tag{37}$$

as an approximation of (1). This discretisation has also been obtained in [63] by directly discretising (1).

Since each summand on the right-hand side contains only quantities from one of the four-pixel cells (18), we can rewrite (37) as an *average* of four dynamical systems each of which contains only interactions within one cell. One such system, written down for simplicity for the cell $\{1,2\} \times \{1,2\}$, reads

$$\begin{aligned}\dot{u}_{1,1} &= 2g_{\frac{3}{2},\frac{3}{2}} \cdot (-(1+\alpha)u_{1,1} + \alpha u_{2,1} + \alpha u_{1,2} + (1-\alpha)u_{2,2}) \,, \\ \dot{u}_{2,1} &= 2g_{\frac{3}{2},\frac{3}{2}} \cdot (\alpha u_{1,1} - (1+\alpha)u_{2,1} + (1-\alpha)u_{1,2} + \alpha u_{2,2}) \,, \\ \dot{u}_{1,2} &= 2g_{\frac{3}{2},\frac{3}{2}} \cdot (\alpha u_{1,1} + (1-\alpha)u_{2,1} - (1+\alpha)u_{1,2} + \alpha u_{2,2}) \,, \\ \dot{u}_{2,2} &= 2g_{\frac{3}{2},\frac{3}{2}} \cdot ((1-\alpha)u_{1,1} + \alpha u_{2,1} + \alpha u_{1,2} - (1+\alpha)u_{2,2}) \,.\end{aligned} \tag{38}$$

We stress that (38) also coincides with the application of (37) to a $2 \times 2$-pixel image with *periodic* boundary conditions, because these boundary conditions surround each pixel with four identical (up to reflections) $2 \times 2$ cells. In this sense, the decomposition is in full analogy with the 1-D case.

To maintain consistency with our variational model, it is important to choose the same parameter $\alpha \in [0,1]$ both in the divergence expression (37) (or (38)) and in the argument (15) of the diffusivity function.

### 4.1.2 Locally Analytic Scheme for Singular Isotropic Diffusion

We strive now to use our four-pixel discretisations for the computation of isotropic diffusion processes. We start with the special case of the singular diffusivities (5), and choose $\alpha = 1/2$ in the discretisations (37) and (15).

**Analytic solution for singular diffusion.** With these settings, we obtain the following dynamical system for the four-pixel cell $\{1,2\} \times \{1,2\}$ which describes the *simultaneous* evolution of the image $u$ and diffusivity field $g$:

$$\begin{aligned}\dot{u}_{1,1} &= g \cdot (-3u_{1,1} + u_{2,1} + u_{1,2} + u_{2,2}) \,, \\ \dot{u}_{2,1} &= g \cdot (u_{1,1} - 3u_{2,1} + u_{1,2} + u_{2,2}) \,, \\ \dot{u}_{1,2} &= g \cdot (u_{1,1} + u_{2,1} - 3u_{1,2} + u_{2,2}) \,, \\ \dot{u}_{2,2} &= g \cdot (u_{1,1} + u_{2,1} + u_{1,2} - 3u_{2,2})\end{aligned} \tag{39}$$

with

$$g = g(t) = (G_{1/2}(u))^{-p} \,. \tag{40}$$

In the following we abbreviate $G_{1/2}(u)$ by $G(u)$. Summation of the four equations (39) yields that the average grey value is constant over time, $\frac{1}{4}(u_{1,1}+u_{2,1}+u_{1,2}+u_{2,2}) = \frac{1}{4}(f_{1,1}+f_{2,1}+f_{1,2}+f_{2,2}) =: \mu$. Therefore the system (39) simplifies to

$$\dot{u}_{i,j} = 4g \cdot (\mu - u_{i,j}) \;, \quad i,j = 1,2 \;. \tag{41}$$

This is a dynamical system with discontinuous right hand side, because of the singularity of $g$ at zero. This discontinuity can be handled in the same way as in [55], see also [30], by considering $(u_{i,j}(t))_{i,j}$ as a solution of (41) if it is absolutely continuous for all $t$, and differentiable whenever the right-hand side is continuous in $t$.

The system (41) can then be solved analytically in the following way. There exists a continuous function $\eta$ with $\eta(0) = 1$, which is differentiable except at discontinuities of the right-hand side of (41), such that, for $i,j = 1,2$,

$$\mu - u_{i,j}(t) = \eta(t) \cdot (\mu - f_{i,j}) \;. \tag{42}$$

Differentiation yields $\dot{u}_{i,j} = -\dot{\eta} \cdot (\mu - f_{i,j})$, which after insertion into (41) leads to

$$-\dot{\eta} \cdot (\mu - f_{i,j}) = 4g \cdot (\mu - u_{i,j}) \tag{43}$$

and by (42) finally to

$$\dot{\eta} = -4\, g\, \eta \;. \tag{44}$$

From (41) and (42) it is clear that $\dot{\eta}$ and $\eta$ always have opposite signs such that the absolute value of $\eta$ can never increase. Consequently, since $\eta(0) = 1$, we see that $\eta(t) \geq 0$ for $t \geq 0$. Taking this into account, we obtain by (15) and (42) that $G(u) = \eta(t) G(f)$. With (5) it follows that

$$\dot{\eta}(t) = -4 (G(f))^{-p}\, \eta(t)^{1-p} \;. \tag{45}$$

This equation has the solution

$$\eta(t) = \begin{cases} \left(1 - \frac{4pt}{(G(f))^p}\right)^{1/p} \;, & 0 \leq t < \frac{(G(f))^p}{4p} \;, \\ 0 \;, & t \geq \frac{(G(f))^p}{4p} \;. \end{cases} \tag{46}$$

Finally, by (42), the analytical solution of (41) is given for $i,j = 1,2$ by

$$u_{i,j}(t) = \begin{cases} \mu + \left(1 - \frac{4pt}{(G(f))^p}\right)^{1/p} (f_{i,j} - \mu) \;, & 0 \leq t < \frac{(G(f))^p}{4p} \;, \\ \mu \;, & t \geq \frac{(G(f))^p}{4p} \;. \end{cases} \tag{47}$$

For the TV diffusion case $p = 1$, particularly, (47) simplifies to

$$u_{i,j}(t) = \begin{cases} \mu + \left(1 - \frac{4t}{G(f)}\right) (f_{i,j} - \mu) \;, & 0 \leq t < \frac{G(f)}{4} \;, \\ \mu \;, & t \geq \frac{G(f)}{4} \;, \end{cases} \tag{48}$$

and shows a linear evolution which can be written in a slightly different form as

$$u_{i,j}(t) = f_{i,j} + \frac{4t}{G(f)} (\mu - f_{i,j}) \, \min\left\{1, \frac{G(f)}{4t}\right\} \;, \qquad i,j = 1,2 \;. \tag{49}$$

For BFB diffusion, $p = 2$, we have the analytical solution

$$u_{i,j}(t) = \begin{cases} \mu + \sqrt{1 - \frac{8t}{(G(f))^2}}\, (f_{i,j} - \mu) \;, & 0 \leq t < \frac{(G(f))^2}{8} \;, \\ \mu \;, & t \geq \frac{(G(f))^2}{8} \;. \end{cases} \tag{50}$$

Let us now use these analytical solutions as building blocks in a numerical scheme.

**Numerical scheme for singular isotropic diffusion.** We recall that the dynamical system (37) in the case under consideration is represented by the average of four dynamical systems of type (39). Using the notations

$$\begin{aligned}
\mu^k_{i,j,--} &:= \tfrac{1}{4}(u^k_{i-1,j} + u^k_{i,j-1} + u^k_{i-1,j-1} + u^k_{i,j}) \,, \\
\mu^k_{i,j,+-} &:= \tfrac{1}{4}(u^k_{i+1,j} + u^k_{i,j-1} + u^k_{i+1,j-1} + u^k_{i,j}) \,, \\
\mu^k_{i,j,-+} &:= \tfrac{1}{4}(u^k_{i-1,j} + u^k_{i,j+1} + u^k_{i-1,j+1} + u^k_{i,j}) \,, \\
\mu^k_{i,j,++} &:= \tfrac{1}{4}(u^k_{i+1,j} + u^k_{i,j+1} + u^k_{i+1,j+1} + u^k_{i,j}) \,,
\end{aligned} \tag{51}$$

we could therefore discretise (37) in time via an explicit Euler scheme and obtain the naive scheme

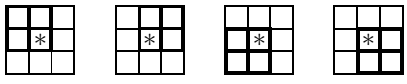
$$\begin{aligned}
u^{k+1}_{i,j} = u^k_{i,j} &+ \tau g_{i-\frac{1}{2},j-\frac{1}{2}} \cdot (\mu^k_{i,j,--} - u^k_{i,j}) + \tau g_{i+\frac{1}{2},j-\frac{1}{2}} \cdot (\mu^k_{i,j,+-} - u^k_{i,j}) \\
&+ \tau g_{i-\frac{1}{2},j+\frac{1}{2}} \cdot (\mu^k_{i,j,-+} - u^k_{i,j}) + \tau g_{i+\frac{1}{2},j+\frac{1}{2}} \cdot (\mu^k_{i,j,++} - u^k_{i,j}) \,.
\end{aligned} \tag{52}$$

Here $\tau$ denotes the time step size and $u^k = (u^k_{i,j})_{i,j}$ the approximate solution at pixel $(i,j)$ and time $k\tau$. Unfortunately, due to the singularity of $g$ at zero, this scheme becomes instable with respect to the maximum-minimum principle for arbitrary small time steps if neighbouring pixel values become arbitrary close. We use therefore a different approximation.

Due to the decomposition of (37) into the average of four $2\times 2$-pixel systems, its solution is approximatively the average of the solutions of the four smaller systems. By (47) we know these analytical solutions. This inspires a simple algorithm to compute one time step of a numerical scheme, in which $u^k$ serves as initial condition for computing (47) within each four-pixel cell up to time $\tau$. Since this scheme is based on composing analytic solutions for small image patches, we will call it a *locally analytic scheme (LAS)*. Its structure is depicted in Figure 2.

***LAS Algorithm for Singular Isotropic Diffusion***

- **For each four-pixel cell $\{i, i+1\} \times \{j, j+1\}$, compute one time step of singular isotropic diffusion according to** (47)**.**
- **For each pixel $(*)$ with coordinates $(i,j)$, consider the four cells**




**which lead to four approximations**

$$u^{k+1}_{i,j,--},\ u^{k+1}_{i,j,+-},\ u^{k+1}_{i,j,-+},\ u^{k+1}_{i,j,++} \,.$$

**Average:**

$$u^{k+1}_{i,j} = \tfrac{1}{4}(u^{k+1}_{i,j,--} + u^{k+1}_{i,j,+-} + u^{k+1}_{i,j,-+} + u^{k+1}_{i,j,++}) \,.$$

Figure 2: One time step of the locally analytic scheme for a singular isotropic diffusion process, where $u^k$, $u^{k+1}$ refer to the old and new time step, respectively.

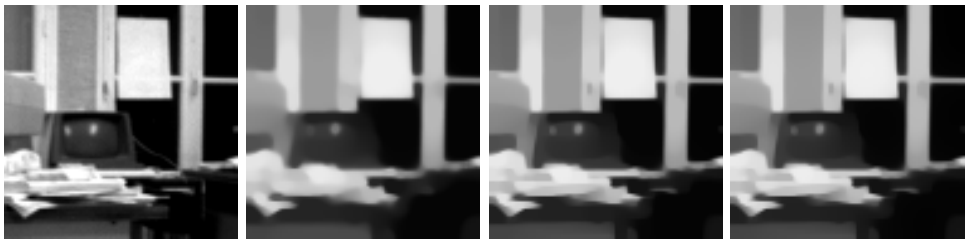


Figure 3: **Left to right: (a)** Original image, $93 \times 93$ pixels. **(b)** TV diffusion with forward Euler scheme, where TV diffusivity is regularised with $\varepsilon = 0.01$, $\tau = 0.0025$, 10000 iterations. **(c)** TV diffusion computed with LAS without regularisation of diffusivity, $\tau = 0.0025$, 10000 iterations. **(d)** LAS with $\tau = 0.1$, 250 iterations.

**Stability Analysis.** The values of the analytical solution (47) at arbitrary time $t \geq 0$ are convex combinations of its initial values. By its construction from the analytical solution (47) the novel scheme in Figure 2 therefore satisfies the maximum–minimum principle. Consequently, it is absolutely stable for each time step size $\tau$.

**Consistency Analysis.** To analyse consistency, let us for simplicity focus on the TV flow, i.e., $p = 1$, but permit an arbitray spatial step size $h$. Then, by (48) our final scheme becomes

$$\begin{aligned} \frac{u_{i,j}^{k+1} - u_{i,j}^k}{\tau} = \quad & \frac{g_{i-\frac{1}{2},j-\frac{1}{2}}}{h^2} \cdot (\mu_{i,j,--}^k - u_{i,j}^k) \, \min\left\{1, \frac{h^2}{4\tau g_{i-\frac{1}{2},j-\frac{1}{2}}}\right\} \\ & + \frac{g_{i+\frac{1}{2},j-\frac{1}{2}}}{h^2} \cdot (\mu_{i,j,+-}^k - u_{i,j}^k) \, \min\left\{1, \frac{h^2}{4\tau g_{i+\frac{1}{2},j-\frac{1}{2}}}\right\} \\ & + \frac{g_{i-\frac{1}{2},j+\frac{1}{2}}}{h^2} \cdot (\mu_{i,j,-+}^k - u_{i,j}^k) \, \min\left\{1, \frac{h^2}{4\tau g_{i-\frac{1}{2},j+\frac{1}{2}}}\right\} \\ & + \frac{g_{i+\frac{1}{2},j+\frac{1}{2}}}{h^2} \cdot (\mu_{i,j,++}^k - u_{i,j}^k) \, \min\left\{1, \frac{h^2}{4\tau g_{i+\frac{1}{2},j+\frac{1}{2}}}\right\}. \end{aligned} \tag{53}$$

This scheme can be considered as a stabilisation of the explicit scheme (52). It coincides with (52), and is therefore a consistent $\mathcal{O}(\tau + h^2)$ approximation for TV diffusion, if each of the four minimum operations on its right-hand side selects the value 1. This consistency condition is fulfilled if

$$0 \leq \tau \leq \min\left\{\frac{h^2}{4g_{i-\frac{1}{2},j-\frac{1}{2}}}, \frac{h^2}{4g_{i+\frac{1}{2},j-\frac{1}{2}}}, \frac{h^2}{4g_{i-\frac{1}{2},j+\frac{1}{2}}}, \frac{h^2}{4g_{i+\frac{1}{2},j+\frac{1}{2}}}\right\} \tag{54}$$

for all $i$ and $j$. For larger $\tau$ it is easy to see that linear diffusion is approximated. This happens in regions where the gradient is already close to zero. In this case, however, the visual differences between linear diffusion and TV diffusion are small. Thus, in practise, it is not a visual problem if this consistency condition is violated in some almost flat areas: Using a fixed, not too large time step size is a pragmatic approach.

Conditional consistency is a characteristic property that can be observed for absolutely stable explicit schemes. A typical example for this behaviour is the DuFort–Frankel method [26].

**Experiments.** We illustrate our LAS approach by three experiments. First, in Figure 3, we contrast the regularisation-free LAS for TV diffusion with a standard explicit discretisation (forward Euler scheme). In the latter, TV diffusivity is approximated by the regularised TV diffusivity $1/\sqrt{|\nabla u|^2 + \varepsilon^2}$. Since the stability

Figure 4: **Left to right: (a)** One quadrant of a rotationally invariant test image, $64 \times 64$ pixels. **(b)** TV diffusion with forward Euler scheme, $\varepsilon = 0.01$, $\tau = 0.0025$, 12000 iterations. **(c)** TV diffusion with our LAS algorithm, $\tau = 0.1$, 300 iterations.

Figure 5: **Left to right: (a)** Original image from Figure 3, $93 \times 93$ pixels. **(b)** Balanced forward–backward diffusion with forward Euler scheme, $\varepsilon = 0.1$, $\tau = 0.0025$, 160000 iterations. **(c)** BFB diffusion with our LAS, $\tau = 0.1$, 4000 iterations.

condition for explicit schemes imposes the upper limit $\varepsilon/4$ on the time step size, a high number of iterations is needed for a reasonably small $\varepsilon$. It can be seen that the LAS based on four-pixel discretisations and the unregularised TV diffusivity – which cannot be used in the explicit scheme – considerably reduce blurring effects caused by the discretisation.

Figure 4 confirms that the LAS approach also performs well in terms of rotational invariance. In spite of the larger time step size and reduced dissipation it keeps at least the same degree of rotational invariance as the explicit scheme, which is an essential requirement in the image processing context.

Figure 5 illustrates the behaviour for balanced forward–backward diffusion. With equal parameters, it can be seen again that the LAS looks sharper by preserving finer details. Moreover, it is worth mentioning that in the LAS experiments we were able to use a time step size that exceeded the largest admissible step size of the explicit scheme by a factor of 40.

#### 4.1.3 Locally Semi-Analytic Scheme for Isotropic Nonlinear Diffusion

**Analytic solutions with frozen diffusivity.** For arbitrary diffusivity functions $g(|\nabla u_\sigma|^2)$, or for the singular diffusivities (5) with $\alpha \neq 1/2$, we do in general not have an analytic solution in the sense of (47) which incorporates the dynamics of $g$. Nevertheless, for a fixed diffusivity field $g$, we can still solve analytically the dynamical system (38) which governs the diffusion process within one four-pixel cell.

To this end it is useful to introduce new variables $w_{i,j}$ by

$$W := HUH \;, \tag{55}$$

where

$$U := \begin{pmatrix} u_{1,1} & u_{2,1} \\ u_{1,2} & u_{2,2} \end{pmatrix} \;, \quad W := \begin{pmatrix} w_{1,1} & w_{2,1} \\ w_{1,2} & w_{2,2} \end{pmatrix} \;, \quad H := \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \;. \tag{56}$$

In terms of the new variables, (38) can be rewritten in a simpler way:

$$\begin{aligned} \dot{w}_{1,1} &= 0 \;, & \dot{w}_{2,1} &= -4g w_{2,1} \;, \\ \dot{w}_{1,2} &= -4g w_{1,2} \;, & \dot{w}_{2,2} &= -8\alpha g w_{2,2} \;, \end{aligned} \tag{57}$$

where we have set $g := g_{\frac{3}{2},\frac{3}{2}}$ for the sole diffusivity value involved. The solution of (57) is clear: While $w_{1,1}$ stays constant, the other variables are subject to independent exponential decays,

$$\begin{aligned} w_{1,1}(t) &= w_{1,1}(0) \;, & w_{2,1}(t) &= \mathrm{e}^{-4gt} w_{2,1}(0) \;, \\ w_{1,2}(t) &= \mathrm{e}^{-4gt} w_{1,2}(0) \;, & w_{2,2}(t) &= \mathrm{e}^{-8\alpha gt} w_{2,2}(0) \;. \end{aligned} \tag{58}$$

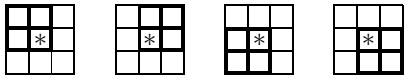
Via the inverse transform of (55),

$$U(t) = H\, W(t)\, H \;, \tag{59}$$

this solution can be rewritten in terms of the original variables.

***LSAS Algorithm for Isotropic Diffusion***

- **Compute the pre-smoothed image $v := K_\sigma * u^k$ by convolution.**
- **For each four-pixel cell $\{i, i+1\} \times \{j, j+1\}$, compute the squared gradient $|\nabla v|^2$ according to (15), and the diffusivity $g = g(|\nabla v|^2)$.**
- **For each four-pixel cell, compute one time step of isotropic diffusion via the analytical solution (55), (58), (59).**
- **For each pixel $(*)$ with coordinates $(i, j)$, consider the four cells**




**which lead to four approximations**

$$u^{k+1}_{i,j,--},\; u^{k+1}_{i,j,+-},\; u^{k+1}_{i,j,-+},\; u^{k+1}_{i,j,++} \;.$$

**Average:**

$$u^{k+1}_{i,j} = \tfrac{1}{4}(u^{k+1}_{i,j,--} + u^{k+1}_{i,j,+-} + u^{k+1}_{i,j,-+} + u^{k+1}_{i,j,++}) \;.$$

Figure 6: One time step of the locally semi-analytic scheme for a nonlinear isotropic diffusion process.

**Numerical scheme in the case of arbitrary diffusivity.** Similar as in Subsection 4.1.2, the solution given by (55), (58), (59) can be used as an approximation to the solution of the dynamical system (38). Consequently, four solutions of this kind can again be used to approximate the solution of (37). As a difference to the previous case, it is now necessary to compute in each time step the pre-smoothed image $v = u_\sigma$ and the diffusivity field $g(|\nabla v|^2)$. One time step of the so adapted scheme is given in Figure 6. Note that this scheme can be used for every $\alpha \in [0, 1]$.

In analogy to semi-implicit schemes that keep the diffusivity fixed at the previous time level while discretising the remainder in an implicit fashion, we will call this scheme a *locally semi-analytic scheme (LSAS)*. Its structure is detailed in Figure 6.

**Stability.** Substituting (59) into (58) one checks that, as in the LAS case, the values of the analytical solution for $u$ for any $t \geq 0$ are convex combinations of the initial values. Via the averaging procedure this guarantees a maximum–minimum principle, and absolute stability of our LSAS for arbitrary time step size $\tau$.

**Consistency.** Similar to the LAS case, the LSAS is conditionally consistent, although the situation differs in detail. For simplicity of notation, let us illustrate this for the 1-D case for $\partial_t u = \partial_x(g\,\partial_x u)$ with some bounded diffusivity function $g = g(|\partial_x v|^2)$. Straightforward computations show that a corresponding LSAS with time step size $\tau$ and spatial step size $h$ can be written as

$$\frac{u_i^{k+1} - u_i^k}{\tau} \;=\; \frac{1}{h}\left(\delta_{i+\frac{1}{2}}^k \frac{u_{i+1}^k - u_i^k}{h} - \delta_{i-\frac{1}{2}}^k \frac{u_i^k - u_{i-1}^k}{h}\right) \tag{60}$$

with

$$\delta_{i\pm\frac{1}{2}}^k := \frac{h^2}{4\tau} - \frac{h^2}{4\tau}\exp\left(-\frac{4\tau}{h^2}\, g_{i\pm\frac{1}{2}}^k\right) \tag{61}$$

A Taylor expansion of the exponential function yields

$$\delta_{i\pm\frac{1}{2}}^k = g_{i\pm\frac{1}{2}}^k + \mathcal{O}\left(\frac{\tau}{h^2}\right) \tag{62}$$

implying that the LSAS is an approximation of the underlying PDE of order $\mathcal{O}(\tau + h^2 + \frac{\tau}{h^2})$. Thus, the LSAS is consistent if $\frac{\tau}{h^2} \to 0$ for $\tau,\, h \to 0$. This conditional consistency is uncritical in the practical relevant digital image analysis setting with a fixed grid size $h := 1$: In this case the LSAS is an unconditionally consistent approximation to the dynamical system that is given by the space-discrete PDE.

**Experiments.** We illustrate the performance of the LSAS for isotropic nonlinear diffusion by two experiments. In both cases we use (4) as diffusivity function. Figure 7 shows the denoising capability of nonlinear isotropic diffusion implemented by a standard explicit scheme and by our LSAS method. The enhanced sharpness of the LSAS result is visible.

Figure 8 demonstrates the rotational invariance properties of the LSAS for nonlinear isotropic diffusion and the effect of different choices of the parameter $\alpha$ in our discretisation. The images filtered with $\alpha = 0$, $\alpha = 0.5$ and $\alpha = 1$ are largely comparable and display a good rotational invariance. A closer look reveals that the scheme with $\alpha = 1$ slightly prefers diagonal directions, while for $\alpha = 0$ faint checkerboard artifacts appear in homogeneous regions. Mixing both discretisations as in Figure 8(c) reduces both phenomena.

Figure 7: **Left to right: (a)** Original image from Figure 3, $93 \times 93$ pixels. **(b)** Processed by nonlinear isotropic diffusion, contrast parameter $\lambda = 3$, noise scale $\sigma = 0.5$, with forward Euler scheme, time step size $\tau = 0.25$, $N = 1600$ iterations. **(c)** Processed by nonlinear isotropic diffusion, same parameters as (b), with LSAS, $\alpha = 0.5$, $\tau = 0.25$, $N = 1600$.

#### 4.1.4 Multi-Channel and 3-D Images

**Multi-channel images.** The procedure outlined at the end of Subsection 3.1 admits to directly extend our numerical schemes to colour images, vector- or matrix-valued images. For a multi-channel image $v = (v_{k;i,j})_{i,j,k}$, one uses

$$(G_\alpha(v))^2 := \sum_k (G_\alpha(v_k))^2 \tag{63}$$

with $(G_\alpha(v_k))^2$ given by (15) as squared gradient approximation in the argument of $g$. With the resulting channel-aggregated diffusivity field, one uses then for each channel separately the divergence discretisation (37) or any of the derived schemes.

**Three-dimensional images.** The natural extension of our ideas to three-dimensional images involves dynamical systems for eight-pixel ($2 \times 2 \times 2$) bricks. Neighbours of a pixel within such a brick now come in three types: they can be adjacent via an edge, a face diagonal or a volume diagonal, adding degrees of freedom to the discretisation procedure. For simplicity, we restrict ourselves to generalising the important case $\alpha = 1/2$ which underlies our LAS from Subsection 4.1.2. In this case, we have $(G_{1/2}(v))^2$ which is just the quarter sum of squared differences of all pixel pairs in the four-pixel cell, and the dynamical system (41). It is easy to check that in the case of an eight-pixel cell $(u_{i,j,l})_{i,j,l=1,2}$ the expression

$$(G(v))^2 := \frac{1}{16} \sum_{(i,j,l) \neq (i',j',l')} (v_{i,j,l} - v_{i',j',l'})^2 \tag{64}$$

is a consistent approximation for $\left( |\nabla v|^2 \right)_{\frac{3}{2},\frac{3}{2},\frac{3}{2}}$, while

$$\dot{u}_{i,j,l} = 4g \cdot (\mu - u_{i,j,l}) \,, \quad i,j,l = 1,2 \,, \tag{65}$$

with $\mu$ denoting the arithmetic mean of the eight initial values, consistently approximates $u_t = \operatorname{div}(g \cdot \nabla u)$. Derivation from these approximations of a LSAS or, with the diffusivity (5), even a LAS for 3-D isotropic nonlinear diffusion based on eight-pixel bricks is straightforward.

### 4.2 Anisotropic Nonlinear Diffusion

We turn now to considering the nonlinear anisotropic diffusion equation (6) with anisotropic diffusion tensor $D(J)$ depending on the structure tensor (7).

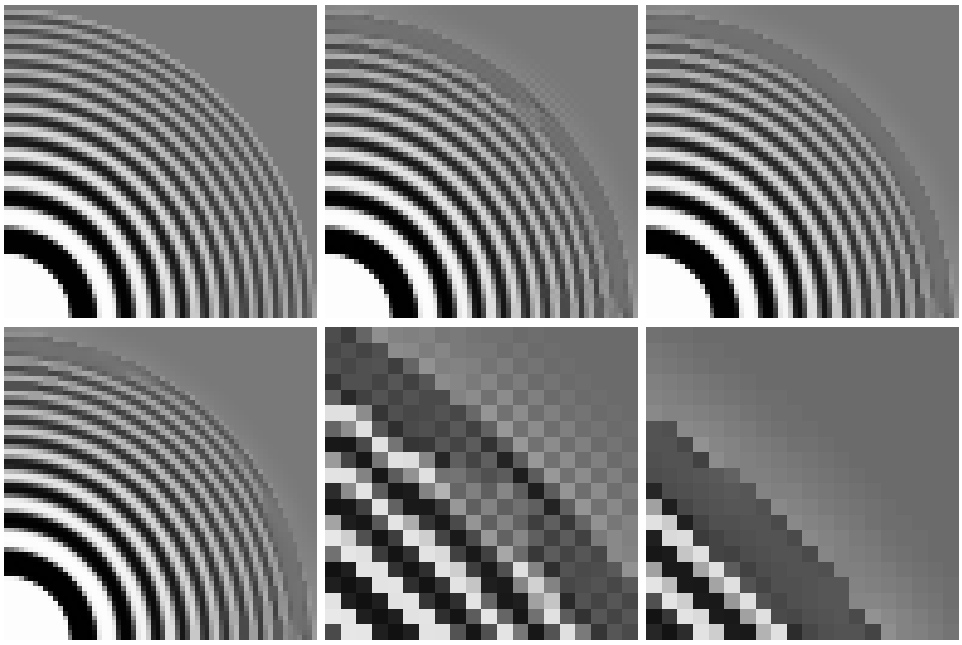


Figure 8: **Top row, left to right: (a)** One quadrant of a rotationally invariant test image, $64 \times 64$ pixels. **(b)** Nonlinear isotropic diffusion, contrast parameter $\lambda = 2.8$, smoothing scale $\sigma = 1$, with LSAS, time step size $\tau = 0.5$ and $N = 20$ iterations. The discretisation parameter in the LSAS is $\alpha = 0$. **(c)** Same as in (b) but with $\alpha = 0.5$. **Bottom row, left to right: (d)** Same as in (b) but with $\alpha = 1$. **(e)** Contrast-enhanced detail ($20 \times 20$ pixels) from (b) with checkerboard artifacts. **(f)** Corresponding contrast-enhanced detail from (c).

#### 4.2.1 Spatial Discretisation

As in the isotropic case, the discretisations developed in Subsection 3.2 give rise to discretisations of both the divergence form and the structure tensor field. Since $\varrho$ and $\sigma$ are no longer bound to be zero, grey-value images can be treated as well as multi-channel images.

**Discretisation of the diffusion tensor field.** Discretising $D$ at the cell midpoints $(i+\frac{1}{2}, j+\frac{1}{2})$ reduces to computing approximations for the structure tensor $J$. Inserting $v = u_\sigma$ into (29) results in a discretisation of the outer product matrices $\nabla u_\sigma \nabla u_\sigma^{\mathrm{T}}$ which by subsequent convolution with $K_\varrho$ yields the discretised structure tensor field.

**Discretisation of anisotropic diffusion with given diffusion tensor field.** To discretise the diffusion process with fixed diffusion tensor field, we have simply to replace (34) with

$$\begin{aligned} D_{i-\frac{1}{2},j-\frac{1}{2}} &=: \begin{pmatrix} a_{--} & c_{--} \\ c_{--} & b_{--} \end{pmatrix}, & D_{i+\frac{1}{2},j-\frac{1}{2}} &=: \begin{pmatrix} a_{+-} & c_{+-} \\ c_{+-} & b_{+-} \end{pmatrix}, \\ D_{i-\frac{1}{2},j+\frac{1}{2}} &=: \begin{pmatrix} a_{-+} & c_{-+} \\ c_{-+} & b_{-+} \end{pmatrix}, & D_{i+\frac{1}{2},j+\frac{1}{2}} &=: \begin{pmatrix} a_{++} & c_{++} \\ c_{++} & b_{++} \end{pmatrix}, \end{aligned} \tag{66}$$

where $D_{i+\frac{1}{2},j+\frac{1}{2}}$ etc. are values of the given diffusion tensor field. With the so defined $a_{\pm\pm}$, $b_{\pm\pm}$, $c_{\pm\pm}$, and $v = u_\sigma$ (in the case of grey-value images), the equations (33), (36) and their convex combinations are the desired approximations.

As before, each of these equations can be split up into the average of four dynamical systems localised in the four-pixel cells (18). Denoting the diffusion tensor discretised in $(\frac{3}{2}, \frac{3}{2})$ simply by $D = \begin{pmatrix} a & c \\ c & b \end{pmatrix}$, we state the four-pixel dynamical system corresponding to $(1-\alpha)\cdot(33) + \alpha\cdot(36)$ for the cell $\{1,2\}\times\{1,2\}$. With the abbreviations

$$\begin{aligned} q_\alpha &:= (1+\alpha)a - (1-\alpha)b \,, \\ r_\alpha &:= -(1-\alpha)a + (1+\alpha)b \,, \\ s_\alpha &:= (1-\alpha)(a+b) + 2c \,, \end{aligned} \tag{67}$$

it reads

$$\begin{aligned} \dot{u}_{1,1} &= q_\alpha(u_{2,1} - u_{1,1}) + r_\alpha(u_{1,2} - u_{1,1}) + s_\alpha(u_{2,2} - u_{1,1}) \,, \\ \dot{u}_{2,1} &= q_\alpha(u_{1,1} - u_{2,1}) + r_\alpha(u_{2,2} - u_{2,1}) - s_\alpha(u_{1,2} - u_{2,1}) \,, \\ \dot{u}_{1,2} &= q_\alpha(u_{2,2} - u_{1,2}) + r_\alpha(u_{1,1} - u_{1,2}) - s_\alpha(u_{2,1} - u_{1,2}) \,, \\ \dot{u}_{2,2} &= q_\alpha(u_{1,2} - u_{2,2}) + r_\alpha(u_{2,1} - u_{2,2}) + s_\alpha(u_{1,1} - u_{2,2}) \,. \end{aligned} \tag{68}$$

A derivation of this result by direct finite-difference discretisation of (6) can easily be devised along the lines of the one given in [64] in the case $\alpha = 0$.

Inserting the *isotropic* diffusion tensor $D = g_{\frac{3}{2},\frac{3}{2}} I$, where $I$ is the $2\times 2$ unit matrix, into (68) yields exactly the system (38) which underlines that our discretisation of anisotropic diffusion is a natural extension of the isotropic case.

### 4.2.2 Locally Semi-Analytic Schemes for Anisotropic Diffusion

**Analytic solution with frozen diffusion tensor field.** Similar to the proceeding in Subsection 4.1.3, we note that the system (68) has a fairly simple analytic solution under the assumption that the diffusion tensors $D$ are kept fixed during the image evolution.

To derive this solution, we rewrite (68) via the variable transform (55) as

$$\begin{aligned} \dot{w}_{1,1} &= 0 \,, & \dot{w}_{2,1} &= -4aw_{2,1} - 4cw_{1,2} \,, \\ \dot{w}_{1,2} &= -4cw_{2,1} - 4bw_{1,2} \,, & \dot{w}_{2,2} &= -4\alpha(a+b)w_{2,2} \,. \end{aligned} \tag{69}$$

This system decomposes into independent evolutions for $w_{1,1}$, which is constant, for $w_{2,2}$ which follows a one-dimensional linear dynamical system with the solution

$$w_{2,2}(t) = \mathrm{e}^{-4\alpha(a+b)t} w_{2,2}(0) \tag{70}$$

(in particular, for $\alpha = 0$ it is constant) and the two-dimensional system

$$\dot{\mathbf{w}} = -4D\mathbf{w} \,. \tag{71}$$

for $\mathbf{w} := (w_{2,1}, w_{1,2})^{\mathrm{T}}$. Let the eigendecomposition of $D$ be given by $D = \lambda_1 \mathbf{e}_1 \mathbf{e}_1^{\mathrm{T}} + \lambda_2 \mathbf{e}_2 \mathbf{e}_2^{\mathrm{T}}$ with eigenvalues $\lambda_{1,2} = \frac{1}{2}(a+b \pm \sqrt{(a-b)^2 + 4c^2})$ and orthonormal eigenvectors $\mathbf{e}_1$, $\mathbf{e}_2$. Then, remembering that $D$ is kept constant, the solution of (71) is

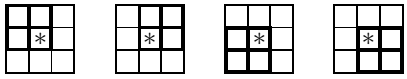
$$\mathbf{w}(t) = \mathrm{e}^{-4\lambda_1 t}(\mathbf{e}_1^{\mathrm{T}}\mathbf{w}(0))\mathbf{e}_1 + \mathrm{e}^{-4\lambda_2 t}(\mathbf{e}_2^{\mathrm{T}}\mathbf{w}(0))\mathbf{e}_2 \,. \tag{72}$$

Via the inverse transform (59) the analytic solution given by (70), (72) can easily be expressed with respect to the original variables.

**Numerical scheme for anisotropic diffusion.** In an analogous manner as done in Subsection 4.1.3 we can employ the explicit solutions (70), (72) to construct locally semi-analytic schemes for anisotropic diffusion. To this end, we use the splitting of $(1-\alpha)\cdot(33) + \alpha\cdot(36)$ into the contributions from four-pixel cells. For

***LSAS Algorithm for Anisotropic Diffusion***

- **Compute the pre-smoothed image** $v := K_\sigma * u^k$ **by convolution.**
- **For each four-pixel cell** $\{i, i+1\} \times \{j, j+1\}$**, compute the approximation of the tensor product** $\nabla v \nabla v^{\mathrm{T}}$ **according to** (29)**.**
- **Compute the structure tensor field** $J = K_\varrho * (\nabla v \nabla v^{\mathrm{T}})$ **by convolution.**
- **For each four-pixel cell, compute the diffusion tensor** $D = D(J)$**.**
- **For each four-pixel cell, compute one time step of anisotropic diffusion via the analytical solution** (55)**,** (70)**,** (72)**,** (59)**.**
- **For each pixel** $(*)$ **with coordinates** $(i, j)$**, consider the four cells**




**which lead to four approximations**

$$u^{k+1}_{i,j,--},\ u^{k+1}_{i,j,+-},\ u^{k+1}_{i,j,-+},\ u^{k+1}_{i,j,++}\ .$$

**Average:**

$$u^{k+1}_{i,j} = \tfrac{1}{4}(u^{k+1}_{i,j,--} + u^{k+1}_{i,j,+-} + u^{k+1}_{i,j,-+} + u^{k+1}_{i,j,++})\ .$$

Figure 9: One time step of the locally semi-analytic scheme for anisotropic diffusion.

these, our analytic solution is evaluated with evolution time equalling the time step size $\tau$.

For the anisotropic diffusion processes in question, the diffusion tensor $D$, whose entries are needed to evaluate (70) and (72), depends on the structure tensor $J_\varrho$. In each time step, the evaluation of the dynamical systems has therefore to be preceded by recomputing $D$ and its eigendecomposition from the current data $u$. This completes our LSAS, one step of which is summarised in Figure 9.

**Stability.** From (59) one easily sees that $\mathrm{tr}(U(t)U(t)^{\mathrm{T}}) = \mathrm{tr}(W(t)W(t)^{\mathrm{T}})$ for all $t$. With the analytical solution (70), (72), it is seen that $\mathrm{tr}(WW^{\mathrm{T}})$ decreases monotonically in $t$ (note that $\lambda_1, \lambda_2 \geq 0$). Since $\mathrm{tr}(UU^{\mathrm{T}})$ is the Euclidean norm of the $2 \times 2$ image, it follows that, within each four-pixel cell, our scheme is $l^2$-stable for any time step size. By the averaging procedure, this $l^2$-stability holds also for the LSAS on the entire grid.

**Consistency.** A consistency analysis in the same way as for the isotropic case shows that also in the anisotropic setting, LSAS is an conditionally consistent approximation of order $\mathcal{O}(\tau + h^2 + \frac{\tau}{h^2})$ to its PDE. Thus, it is consistent if $\frac{\tau}{h^2} \to 0$ for $\tau, h \to 0$. For a fixed spatial grid size $h$, however, it is an unconditionally consistent $\mathcal{O}(\tau)$ approximation to the underlying dynamical system.

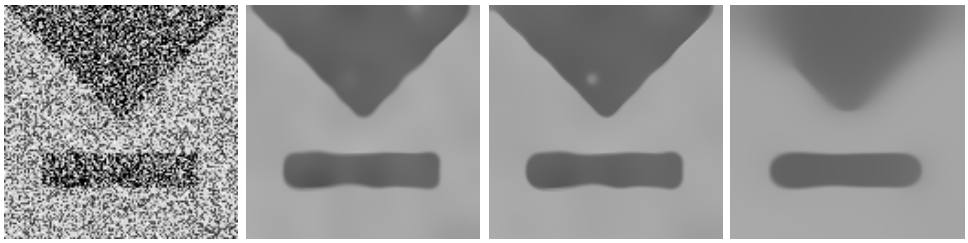


Figure 10: **Left to right: (a)** Test image with impulsive noise, $128 \times 128$ pixels. **(b)** Denoised by edge-enhancing diffusion, $\lambda = 5$, $\sigma = 1.8$, $\varrho = 0$, with forward Euler scheme, $\tau = 0.166$, $N = 200$ iterations. **(c)** Denoised by edge-enhancing diffusion, same parameters as in (b), but with LSAS, $\alpha = 0$, $\tau = 1$, $N = 200$ iterations. **(d)** Denoised by edge-enhancing diffusion with forward Euler scheme, same parameters as in (b) but $N = 1200$ for equal evolution time as in the LSAS example (c).

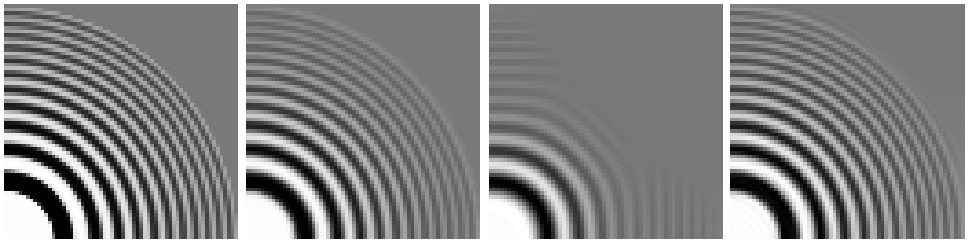


Figure 11: **Left to right: (a)** Test image from Figure 8, $64 \times 64$ pixels. **(b)** Exact solution for coherence-enhancing diffusion, $\varepsilon = 0.001$, $C = 1$, $\sigma = 0.5$, $\varrho = 4$, at time $t = 250$. **(c)** Filtered with the nonnegativity scheme [59] with $\tau = 1/6$, and $N = 1500$ iterations. Average absolute error: 17.99. **(d)** Processed with our LSAS algorithm ($\alpha = 0$), same parameters. Average absolute error: 3.81.

**Experiments.** In Figure 10 we use our scheme to perform edge-enhancing diffusion [59]. This process has been designed to perform diffusion along edges, while reducing smoothing across them. In this case, there is no integration over the outer products, so $\varrho = 0$. The diffusion tensor $D$ has the same eigenvectors as the outer product $J = \nabla v \nabla v^{\mathrm{T}}$, namely $\nabla v$ and its orthogonal $\nabla v^{\perp}$. The eigenvalue in direction $\nabla v = \nabla u_\sigma$ is given by the same function (4) with threshold parameter $\lambda > 0$ that has already been used in Subsection 4.1.3, which means that $g$ is applied to the first eigenvalue of $J$. The eigenvalue of $D$ in direction $\nabla v^{\perp}$ is fixed to 1.

The noisy image from Figure 10(a) is denoised with a standard explicit (forward Euler) scheme with central spatial differences, and with the locally semi-analytic scheme. It is observed that the denoising result with our new scheme is slightly sharper. Moreover, a look at the parameters shows that the effective evolution time used by the new scheme is six times larger than with the explicit scheme which demonstrates how much the latter is indeed dominated by numerical dissipation: Running the explicit discretisation up to the same evolution time (d) already partially destroys the object contours.

In our next experiment, Figure 11, we consider coherence-enhancing diffusion (CED) [59]. It uses an integration scale $\varrho$ that is considerably larger than $\sigma$, thereby introducing into $J$ a smoothing over eigenvector systems. If the structure tensor has the eigendecomposition $J = \mu_1 \mathbf{e}_1 \mathbf{e}_1^{\mathrm{T}} + \mu_2 \mathbf{e}_2 \mathbf{e}_2^{\mathrm{T}}$ with $\mu_1 \geq \mu_2$, then the diffusion

Figure 12: **Left to right: (a)** Fingerprint image, $100 \times 100$ pixels. **(b)** Filtered with CED, $\varepsilon = 0.001$, $C = 1$, $\sigma = 0.5$, $\varrho = 4$, using the nonnegativity scheme [59] with $\tau = 1/6$ and $N = 60$ iterations. **(c)** Processed with our LSAS scheme ($\alpha = 0$) for CED, same parameters. **(d)** LSAS scheme with $\tau = 1$ and $N = 10$ iterations.

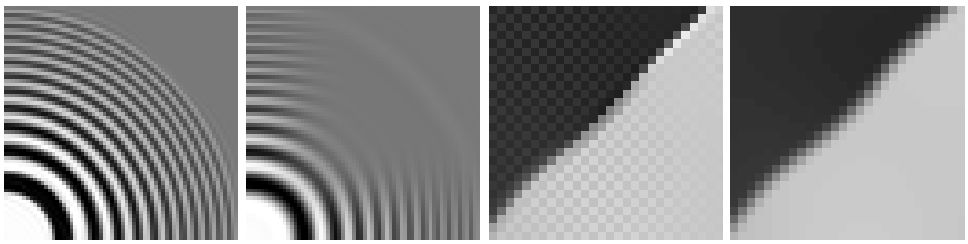


Figure 13: **Left to right: (a)** Test image from Figure 8(a) filtered by coherence-enhancing diffusion, $\varepsilon = 0.001$, $C = 1$, $\sigma = 0.5$, $\varrho = 4$, with LSAS, $\tau = 0.166$ and $N = 300$. The discretisation weight in the LSAS is $\alpha = 0$ **(b)** Same but with discretisation weight $\alpha = 1$. **(c)** Detail ($28 \times 28$ pixels) from Figure 10(c), where the grey-values have been rescaled to raise contrast. **(d)** Corresponding detail from LSAS filtering result for CED with same parameters as in Figure 10(c) except $\alpha = 0.01$.

tensor $D(J)$ has the decomposition $D(J) := \lambda_1 \mathbf{e}_1 \mathbf{e}_1^{\mathrm{T}} + \lambda_2 \mathbf{e}_2 \mathbf{e}_2^{\mathrm{T}}$ with eigenvalues

$$\begin{aligned} \lambda_1 &:= \varepsilon, \\ \lambda_2 &:= \begin{cases} \varepsilon & \text{if } \mu_1 = \mu_2, \\ \varepsilon + (1-\varepsilon) \exp\left(\frac{-C}{(\mu_1 - \mu_2)^2}\right) & \text{else,} \end{cases} \end{aligned} \tag{73}$$

some small regularisation parameter $\varepsilon > 0$ and a contrast parameter $C > 0$. This process smoothes along flow-like structures. For our rotationally invariant test image only radial linear diffusion with diffusivity $\varepsilon$ takes place. Hence, the exact solution at time $t$ is given by a convolution with a Gaussian of standard deviation $\sqrt{2\varepsilon t}$. By comparing the solutions of the so-called nonnegativity discretisation from [59] with our LSAS algorithm and the exact solution, we see that the LSAS does not suffer from visible blurring artifacts. It preserves rotation invariance very well and creates significantly lower errors than the nonnegativity scheme.

The quantitative findings from Fig. 11 are also consistent with the qualitative assessment of the fingerprint example in Figure 12. We observe that the LSAS scheme gives much sharper results, and that it yields still realistic results for time step sizes far beyond the stability limit 1/6 of the nonnegativity scheme.

Finally, we investigate the effect of different discretisation parameters $\alpha$ in our LSAS for anisotropic diffusion. The comparison of the CED filtering results in Figure 13(a) and (b) reveals that, unlike in the isotropic case, the choice $\alpha = 0$ is clearly superior to $\alpha = 1$ in terms of both sharpness and rotational invariance. However, from Figure 13(c) and (d) it can be seen that introducing a small amount

of the second discretisation ($1 \ldots 2$ percent) allows to suppress the checkerboard artifacts that are inherent to the $\alpha = 0$ method.

# 5 Link to Haar Wavelet Shrinkage

In this section we want to interpret our LAS/LSAS schemes for isotropic and anisotropic diffusion in the context of Haar wavelet shrinkage. In the one-dimensional setting, a link between explicit nonlinear diffusion schemes and Haar wavelet shrinkage was shown in [55, 42]. We start by considering four-pixel images.

## 5.1 Haar Wavelet Shrinkage on Four-Pixel Cells

Let us explain how $f := (f_{i,j})_{i,j=1}^2$ changes under two-dimensional Haar wavelet shrinkage. One cycle of Haar wavelet shrinkage consists of the following three steps.

1. In the *analysis step*, the four-pixel image $f$ is transformed by applying low and high pass Haar filters to the rows and columns of $f$, more precisely

$$\hat{f} := HfH \tag{74}$$

   with $H$ defined by (56).

2. In the *shrinkage step*, we modify the high-pass coefficients by reducing the absolute values of some or all of them. To this end, we apply a shrinkage function $S_\theta$ depending on a threshold parameter $\theta$ to the high-pass filtered coefficients, i.e. we compute $S_\theta(\hat{f}_{1,2})$, $S_\theta(\hat{f}_{2,1})$, $S_\theta(\hat{f}_{2,2})$ and let the low-pass coefficient $S_\theta(\hat{f}_{1,1}) = \hat{f}_{1,1}$ fixed.

3. In the *synthesis step*, we perform the inverse transform of step 1 on the shrunken coefficients, and end up with

$$f^{(1)} = HS_\theta(\hat{f})H \; . \tag{75}$$

In conventional wavelet shrinkage, the threshold depends on the individual coefficients $\hat{f}_{i,j}$. For example, *soft shrinkage* [23] shrinks the coefficients towards 0 by an amount that is given by a threshold parameter $\theta$:

$$S_\theta(\hat{f}_{i,j}) := \begin{cases} \hat{f}_{i,j} - \theta \operatorname{sgn}(\hat{f}_{i,j}) & \text{if } |\hat{f}_{i,j}| \geq \theta \; , \\ 0 & \text{otherwise} \; . \end{cases} \tag{76}$$

In the following, we introduce shrinkage rules such that the Haar wavelet shrinkage produces exactly the analytic solutions of our small dynamical systems (41), (57) and (69). In contrast to ordinary wavelet shrinkage our diffusion-inspired shrinkage rules couple the three high-pass components in an appropriate way. We will see that this simple coupling trick will help to improve even the rotation invariance of multi-scale Haar wavelet schemes. To our knowledge, the first coupled isotropic shrinkage rule was proposed in [41].

**Isotropic case with singular diffusivity.** By straightforward computation we see that $G_\alpha^2(v) \approx |\nabla v|^2$ determined in (15) coincides with

$$G_\alpha^2(v) = \hat{v}_{2,1}^2 + \hat{v}_{1,2}^2 + 2\alpha\, \hat{v}_{2,2}^2 \; . \tag{77}$$

First we consider the solution of the dynamical system (41) according to the singular diffusivity (5). Then, with $G(f) := G_{1/2}(f) = \left(\hat{f}_{2,1}^2 + \hat{f}_{1,2}^2 + \hat{f}_{2,2}^2\right)^{1/2}$, the corresponding shrinkage rule is given for $(i,j) \in \{(1,2),(2,1),(2,2)\}$ by

$$S_\theta(\hat{f}_{i,j}) := \begin{cases} \left(1 - 4p\,(G(f))^{-p}\,t\right)^{1/p}\,\hat{f}_{i,j}\,, & G(f) \geq (4p\,\theta)^{1/p}\,, \\ 0\,, & G(f) < (4p\,\theta)^{1/p}\,. \end{cases} \tag{78}$$

In other words, the inverse Haar transform applied to the shrunken coefficients $S_\theta(\hat{f}_{i,j})$ in (78) produces for $\theta = t$ exactly the solution (47) of our dynamical system (41).

**Isotropic case with arbitrary diffusivity.** In case of arbitrary diffusivity functions, the shrinkage rule follows immediately from (58), namely

$$\begin{aligned} S_\theta(\hat{f}_{2,1}) &:= \mathrm{e}^{-4g\theta}\,\hat{f}_{2,1}\,, \\ S_\theta(\hat{f}_{1,2}) &:= \mathrm{e}^{-4g\theta}\,\hat{f}_{1,2}\,, \\ S_\theta(\hat{f}_{2,2}) &:= \mathrm{e}^{-8g\alpha\theta}\,\hat{f}_{2,2}\,, \end{aligned} \tag{79}$$

where $g = g(G_\alpha^2(v))$ and $\theta = t$. We remark that the shrinkage rules proposed in [41] equal the first-order terms of (79).

**Anisotropic case.** First of all, the approximation of the coefficients of the structure tensor $\nabla v \nabla v^{\mathrm{T}}$ in (29) can be written in wavelet terms as

$$(\partial_x v)^2 \approx \hat{v}_{2,1}^2 + \alpha\hat{v}_{2,2}^2\,, \quad (\partial_y v)^2 \approx \hat{v}_{1,2}^2 + \alpha\hat{v}_{2,2}^2\,, \quad \partial_x v\,\partial_y v \approx \hat{v}_{2,1}\hat{v}_{1,2}\,. \tag{80}$$

In accordance with (70) and (72), our shrinkage function couples the reduction of the antidiagonal coefficients $\hat{f}_{1,2}$ and $\hat{f}_{2,1}$ while shrinking $\hat{f}_{2,2}$ independently:

$$\begin{aligned} S_\theta\left(\begin{pmatrix}\hat{f}_{2,1}\\ \hat{f}_{1,2}\end{pmatrix}\right) &:= Q\begin{pmatrix} e^{-4\lambda_1\theta} & 0 \\ 0 & e^{-4\lambda_2\theta}\end{pmatrix} Q^{\mathrm{T}}\begin{pmatrix}\hat{f}_{2,1}\\ \hat{f}_{1,2}\end{pmatrix}\,, \\ S_\theta(\hat{f}_{2,2}) &:= \mathrm{e}^{-4\alpha(a+b)\theta}\,\hat{f}_{2,2}\,, \end{aligned} \tag{81}$$

where $Q := (\mathbf{e}_1, \mathbf{e}_2)$ denotes the eigenvector matrix of $D$ which can be computed from (80). Again, in order to get equivalent schemes, we have to identify the threshold parameter $\theta$ with the diffusion time $t$.

## 5.2 Single-Scale Haar Wavelet Shrinkage

Having identified the analytical solutions of our small dynamical systems with appropriate Haar wavelet shrinkage procedures, we can describe their incorporation into the diffusion of the whole $N \times M$ image. In the following, we reformulate our LAS and LSAS schemes in wavelet language. Consider an arbitrary image $f$. The two-dimensional *decimated* Haar wavelet transform acts naturally on subsequent $2 \times 2$-pixel tiles of an image. Unfortunately, this process is not shift invariant. However, it is easy to check that the $(--)$, $(+-)$, $(-+)$, $(++)$ averaging procedures in our LAS/LSAS methods describe exactly the process of cyclic spinning in wavelet shrinkage [18]. Cyclic spinning of the decimated wavelet transform at a single scale is equivalent to applying the *undecimated* wavelet transform. This undecimated transform is shift invariant and we will refer to it as shift invariant Haar transform. More precisely, let

$$H_0(r) := \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & 1 & \dots & 0 & 0 \\ \vdots & & \ddots & & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 & 1 \end{pmatrix}, \; H_1(r) := \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & -1 & 0 & \dots & 0 & 0 \\ 0 & 1 & -1 & \dots & 0 & 0 \\ \vdots & & \ddots & & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 & -1 \end{pmatrix} \tag{82}$$

denote the Toeplitz matrices of size $r \times (r+1)$ corresponding to the undecimated Haar low and high pass filters. In the following, they will be used with different sizes $r$. Since the size is always evident from the matrix multiplications, we will omit the sizes from the notations and write only $H_0$ and $H_1$. Since $\frac{1}{2}\big(H_0(r)H_0(r+1) - H_1(r)H_1(r+1)\big) = I$, we use $\frac{1}{2}H_0(r)$ and $-\frac{1}{2}H_1(r)$ for the inverse shift invariant Haar transform.

In the following, we assume Neumann boundary conditions. Let $U$ denote the $(N+2) \times (M+2)$-pixel image obtained by mirroring the $k$-th time iterated image $U^k \in \mathbb{R}^{N \times M}$ at the boundaries by one pixel.

**LAS for isotropic singular diffusion.** With the preceding considerations the scheme from Figure 2 reads in wavelet fashion as follows:

1. Compute the transformed images $\hat{U}_{i,j}$ of size $(N+1) \times (M+1)$ according to

$$\begin{aligned} \hat{U}_{1,1} &:= H_0\, U\, H_0^{\mathrm{T}}\,, & \hat{U}_{2,1} &:= H_0\, U\, H_1^{\mathrm{T}}\,, \\ \hat{U}_{1,2} &:= H_1\, U\, H_0^{\mathrm{T}}\,, & \hat{U}_{2,2} &:= H_1\, U\, H_1^{\mathrm{T}}\,. \end{aligned} \tag{83}$$

2. Compute $G(U) := \big(\hat{U}_{2,1}^2 + \hat{U}_{1,2}^2 + \hat{U}_{2,2}^2\big)^{1/2}$ and apply the shrinkage function (78) with threshold $\theta = t$ to obtain $S_{i,j} := S_\theta(\hat{U}_{i,j})$ for $(i,j) \in \{(2,1),(1,2),(2,2)\}$. All operations are meant componentwise.

3. Compute the $N \times M$ image

$$U^{k+1} := \frac{1}{4}\Big(H_0\, \hat{U}_{1,1} H_0^{\mathrm{T}} - H_0\, S_{2,1} H_1^{\mathrm{T}} - H_1\, S_{1,2} H_0^{\mathrm{T}} + H_1\, S_{2,2} H_1^{\mathrm{T}}\Big)\,. \tag{84}$$

**LSAS for isotropic diffusion.** In the LSAS case, we have to modify the first and second step of the algorithm. Let $V$ denote the $(N+2) \times (M+2)$-pixel image obtained by mirroring the convolved $k$-th iteration $V^k = K_\sigma * U^k$ at the boundaries by one pixel.

Then the LSAS algorithm for isotropic diffusion computes in step 1 additionally the values $\hat{V}_{1,2}$, $\hat{V}_{2,1}$ and $\hat{V}_{2,2}$. The shrinkage step 2 uses the shrinkage rule (79) with $g := g\big(\hat{V}_{1,2}^2 + \hat{V}_{2,1}^2 + 2\alpha\, \hat{V}_{2,2}^2\big)$.

**LSAS for anisotropic diffusion.** Finally, the LSAS for anisotropic diffusion applies in step 2 the shrinkage rule (81), where $Q$ and the values $\lambda_1, \lambda_2$ and $a, b$ are computed from $\hat{V}_{1,2}$, $\hat{V}_{2,1}$ and $\hat{V}_{2,2}$ based on (80).

## 5.3 Multi-Scale Haar Wavelet Shrinkage

Typically, the wavelet transform has multi-scale character. The shift invariant multi-scale transform can be simply performed by the algorithm à trous [38]. To this end, we have only to introduce $2^s - 1$ zeros between the coefficients of the high and low pass filters at level $s$, i.e. we use

$$(1, \underbrace{0, \dots, 0}_{2^s-1}, 1, 0, \dots, 0) \quad \text{and} \quad (1, \underbrace{0, \dots, 0}_{2^s-1}, -1, 0, \dots, 0) \tag{85}$$

as first column in the Toeplitz transform matrices $H_0^{(s)}$ and $H_1^{(s)}$, respectively. These are $r \times (r + 2^s)$-matrices, again with $r$ determined by the occurring matrix multiplications.

Let $m$ denote the desired decomposition level. Then we mirror $U^k$, respectively $V^k$ in case of an LSAS scheme, at the beginning of the procedure by $2^m - 1$ pixels at the boundaries to obtain the $(N + 2^{m+1} - 2) \times (M + 2^{m+1} - 2)$-pixel images $\hat{U}_{1,1}^{(0)}$ and $\hat{V}_{1,1}^{(0)}$. Now our multi-scale LAS/LSAS algorithm reads as follows:

1. For $s = 1, 2, \ldots, m$ compute

$$\begin{aligned} \hat{U}_{1,1}^{(s)} &= H_0^{(s-1)}\, \hat{U}_{1,1}^{(s-1)} (H_0^{(s-1)})^{\mathrm{T}}\,, \qquad & \hat{U}_{2,1}^{(s)} &= H_0^{(s-1)}\, \hat{U}_{1,1}^{(s-1)} (H_1^{(s-1)})^{\mathrm{T}}\,, \\ \hat{U}_{1,2}^{(s)} &= H_1^{(s-1)}\, \hat{U}_{1,1}^{(s-1)} (H_0^{(s-1)})^{\mathrm{T}}\,, \qquad & \hat{U}_{2,2}^{(s)} &= H_1^{(s-1)}\, \hat{U}_{1,1}^{(s-1)} (H_1^{(s-1)})^{\mathrm{T}}\,. \end{aligned} \tag{86}$$

   In case of an LSAS scheme compute additionally $\hat{V}_{1,1}^{(s)}$, $\hat{V}_{2,1}^{(s)}$, $\hat{V}_{1,2}^{(s)}$ and $\hat{V}_{2,2}^{(s)}$ in the same way.

2. Perform componentwise the appropriate shrinkage rule

   - (78) for LAS,
   - (79) for isotropic LSAS,
   - (81) for anisotropic LSAS

   to obtain $S_{i,j}^{(s)}$ for $(i,j) \in \{(2,1),(1,2),(2,2)\}$.

3. Set $U^{(m)} := \hat{U}_{1,1}^{(m)}$ and compute for $s = m, m-1, \ldots, 1$

$$\begin{aligned} U^{(s-1)} := \frac{1}{4}\Big( & H_0^{(s-1)}\, U^{(s)} (H_0^{(s-1)})^{\mathrm{T}} - H_0^{(s-1)}\, S_{2,1}^{(s)} (H_1^{(s-1)})^{\mathrm{T}} \\ & - H_1^{(s-1)}\, S_{1,2}^{(s)} (H_0^{(s-1)})^{\mathrm{T}} + H_1^{(s-1)}\, S_{2,2}^{(s)} (H_1^{(s-1)})^{\mathrm{T}} \Big)\,. \end{aligned} \tag{87}$$

   As result we obtain the $N \times M$-pixel image $U^{k+1} := U^{(0)}$.

We want to add some remarks concerning this multi-scale algorithm. First of all, in the case of anisotropic diffusion we have to make sure that we use the à trous Gaussian convolution with $K_\varrho^{(s)}$ to compute the structure tensor at level $s$. Here $K_\rho^{(s)}$ is again obtained by introducing $2^s - 1$ zeros between the Gaussian filter coefficients. Next, we may replace the computation of $\hat{V}_{1,1}^{(s)}$ in the LSAS algorithms by the convolution

$$\hat{V}_{1,1}^{(s)} = K_\sigma^{(s)} * \hat{U}_{1,1}^{(s)} \tag{88}$$

of $\hat{U}_{1,1}^{(s)}$ with the à trous Gaussian $K_\sigma^{(s)}$. This requires more effort than the original computation and introduces an additional smoothing. Finally, one may apply scale adapted shrinkage parameters $\theta$, a topic which is not further discussed in this paper.

**Experiments.** In the following, we examine the performance of our multi-scale LAS and LSAS algorithms by numerical examples. We start with multi-scale LAS for isotropic diffusion. More precisely, we want to demonstrate the improved rotation invariance by our coupled shrinkage rule (78) for the TV diffusivity, i.e. $p = 1$. In Figure 14, we compare the performance of our multi-scale LAS with shift invariant soft wavelet shrinkage and with Kingsbury's algorithm with $(9,7)$-tap and 14-tap filters [35] (http://www-sigproc.eng.cam.ac.uk/~ngk) which is known for good rotation invariance. Our experiments confirm the good approximate rotation invariance of the LAS method.

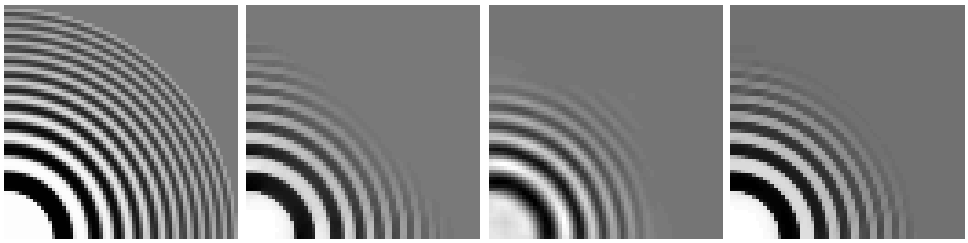


Figure 14: **Left to right: (a)** Original rotationally invariant test image. **(b)** Shift invariant soft Haar wavelet shrinkage with threshold $\theta = 5$, 6 scales and 5 iterations. **(c)** Soft shrinkage in Kingsbury's algorithm with $(9, 7)$-tap and 14-tap filters, 6 scales, threshold $\theta = 15$ and 5 iterations. **(d)** Multi-scale LAS for total variation diffusion with threshold $\theta = 15$, 6 scales and 5 iterations.

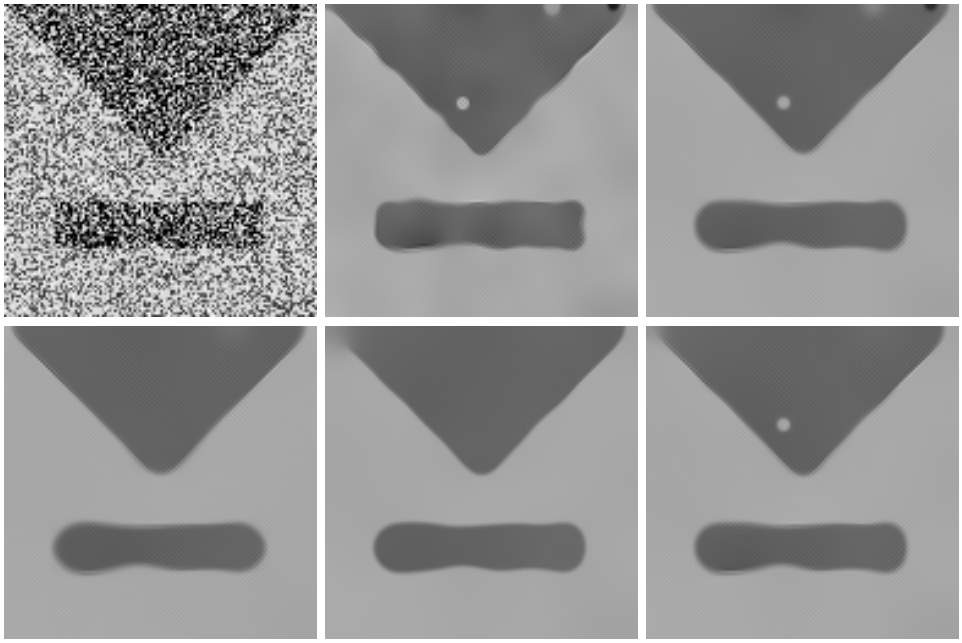


Figure 15: **Top left to bottom right in rows: (a)** Noisy test image from Figure 10. **(b)**–**(d)** Application of multi-scale LSAS for edge-enhancing diffusion with $\lambda = 5$, $\sigma = 1.8$, $\varrho = 0$, $\alpha = 0$, $\tau = 1$, $N = 100$ iterations and 1, 2, 3 scales. **(e)** Single-scale LSAS with $N = 400$ iterations. **(f)** Processed with two-scale LSAS and modified smooth computation (88).

Next we are interested in the effect that the multi-level extension has on our LSAS method for anisotropic diffusion. We use the same experimental settings as in Section 4.2. We start with LSAS for edge-enhancing diffusion and use the same test image as in Figure 10 with the same parameters $\lambda = 5$, $\sigma = 1.8$, $\varrho = 0$, $\alpha = 0$, $\tau = 1$, but only 100 iterations to underline the differences between several numbers of levels. The results are shown in Figure 15. For one scale (b) we have our original LSAS scheme. As expected the images become smoother in flat regions when more scales are used (c, d). At the same time we observe that vertices are increasingly rounded. This is partially due to the fact that the same number of iterations with a higher number of scales realises a larger effective diffusion time, cf. also [43]. This is confirmed by image (e) where 400 iterations were performed on one scale, leading to a comparable rounding as in (c). The last image (f) demonstrates the performance

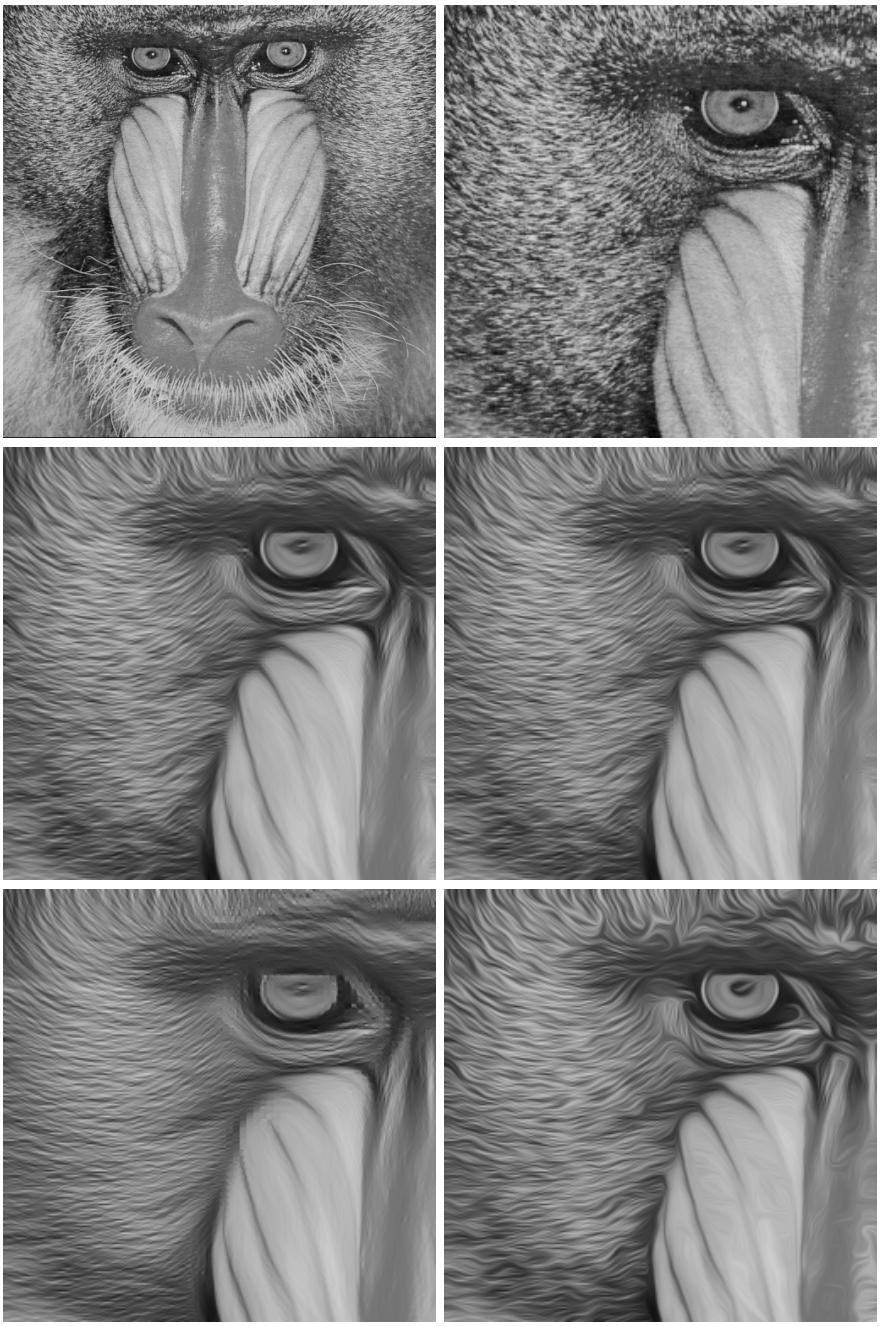


Figure 16: **Top left to bottom right in rows: (a)** Original image, $512 \times 512$ pixels. **(b)** Part of the original image of size $256 \times 256$. **(c)** Coherence-enhancing diffusion by LSAS scheme with $\alpha = 0$, $\varepsilon = 0.001$, $C = 1$, $\sigma = 0.5$, $\varrho = 2$, $\tau = 0.166$, 20 iterations and one scale. **(d)** Processed with two scales. **(e)** Processed with three scales. **(f)** Processed with one scale but 80 iterations.

of the modified algorithm with additional smoothing (88).

Finally, Figure 16 gives a clue how our multi-scale LSAS behaves for coherence-enhancing diffusion. We have transformed the whole "mandrill" image and clipped to the top left $256 \times 256$ pixels afterwards to make the smoothing process better visible.

# 6 Conclusions

In the present paper we have introduced a novel class of schemes for nonlinear isotropic and anisotropic diffusion filters. They average over analytical or semi-analytical solutions to the dynamical systems of $2 \times 2$-pixel image patches. Since these methods are highly local and less dissipative than most other finite difference schemes, they hardly suffer from blurring artifacts at image edges. Their implementation is as simple as for explicit schemes, but they offer unconditional absolute $l^\infty$-stability in the isotropic case, and unconditional absolute $l^2$-stability in the anisotropic setting. While they are conditionally consistent, one can use significantly larger time step sizes than for the widely used conditionally stable explicit schemes. Compared to well-performing implicit methods such as AOS schemes, they are less complicated to implement and they can be applied directly to a larger class of filters, in particular anisotropic ones and singular diffusion processes with unbounded diffusivities. Their locality makes them attractive for parallel computing and their simple structure would even suggest to implement them in hardware.

Another interesting applications of these schemes is their interpretation in terms of Haar wavelet shrinkage. This has led to novel shrinkage rules with coupling of the coefficients and a high degree of rotation invariance. Our results also demonstrate that sophisticated concepts such as ridgelets and curvelets are not the only way to perform advanced anisotropic wavelet-based shrinkage. Even the most elementary class of wavelets, namely Haar wavelets, are sufficient for implementing highly anisotropic filters in a rotationally invariant fashion. We hope that this novel connection can help to fertilise further research on simple, structure-adaptive anisotropic wavelet concepts and to gain new insights into the design of coupled shrinkage rules.

**Acknowledgements.** Partial funding by the *Deutsche Forschungsgemeinschaft (DFG)* is gratefully acknowledged. The first author also thanks Emory University, Atlanta, and the Institute for Mathematics and its Applications, Minneapolis, for their hospitality during part of his work on this paper.

# References

[1] L. Alvarez, F. Guichard, P.-L. Lions, and J.-M. Morel. Axioms and fundamental equations in image processing. *Archive for Rational Mechanics and Analysis*, 123:199–257, 1993.

[2] F. Andreu, V. Caselles, J. I. Diaz, and J. M. Mazón. Qualitative properties of the total variation flow. *Journal of Functional Analysis*, 188(2):516–547, Feb. 2002.

[3] F. Andreu–Vaillo, V. Caselles, and J. M. Mazon. *Parabolic Quasilinaer Equations Minimizing Linear Growth Functionals*, volume 223 of *Progress in Mathematics*. Birkhäuser, Basel, 2004.

[4] Y. Bao and H. Krim. Towards bridging scale-space and multiscale frame analyses. In A. A. Petrosian and F. G. Meyer, editors, *Wavelets in Signal and Image Analysis*, volume 19 of *Computational Imaging and Vision*, chapter 6. Kluwer, Dordrecht, 2001.

[5] G. Bellettini, V. Caselles, and M. Novaga. The total variation flow in $R^N$. *Journal of Differential Equations*, 184(2):475–525, 2002.

[6] P. Blomgren and T. F. Chan. Color TV: total variation methods for restoration of vector valued images. *IEEE Transactions on Image Processing*, 7(3):304–309, Mar. 1998.

[7] K. Bredies, D. A. Lorenz, P. Maaß, and G. Teschke. A partial differential equation for continuous non-linear shrinkage filtering and its application for analyzing MMG data. In F. Truchetet, editor, *Wavelet Applications in Industrial Processing*, volume 5266 of *Proceedings of SPIE*, pages 84–93. SPIE Press, Bellingham, 2004.

[8] M. Breuß, T. Brox, A. Bürgel, T. Sonar, and J. Weickert. Numerical aspects of TV flow. *Numerical Algorithms*, 41(1):79–101, Jan. 2006.

[9] E. Candès, L. Demanet, D. Donoho, and L. Ying. Fast discrete curvelet transform. *Multiscale Modeling and Simulation*, 2006. To appear.

[10] E. J. Candès and D. L. Donoho. Curvelets: A surprisingly effective nonadaptive representation of objects with edges. In A. Cohen, C. Rabut, and L. L. Schumaker, editors, *Curve and Surface Fitting*, Saint-Malo, 2000. Vanderbilt University Press.

[11] E. J. Candès and F. Guo. New multiscale transforms, minimum total variation synthesis: Applications to edge-preserving image reconstruction. *Signal Processing*, 82(11):1519–1543, 2002.

[12] F. Catté, P.-L. Lions, J.-M. Morel, and T. Coll. Image selective smoothing and edge detection by nonlinear diffusion. *SIAM Journal on Numerical Analysis*, 32:1895–1909, 1992.

[13] A. Chambolle, R. A. DeVore, N. Lee, and B. L. Lucier. Nonlinear wavelet image processing: variational problems, compression, and noise removal through wavelet shrinkage. *IEEE Transactions on Image Processing*, 7(3):319–335, Mar. 1998.

[14] A. Chambolle and B. L. Lucier. Interpreting translationally-invariant wavelet shrinkage as a new image smoothing scale space. *IEEE Transactions on Image Processing*, 10(7):993–1000, 2001.

[15] T. F. Chan and H. M. Zhou. Total variation improved wavelet thresholding in image compression. In *Proc. Seventh International Conference on Image Processing*, volume II, pages 391–394, Vancouver, Canada, Sept. 2000.

[16] A. Cohen, W. Dahmen, I. Daubechies, and R. DeVore. Harmonic analysis in the space BV. *Revista Matematica Iberoamericana*, 19:235–262, 2003.

[17] A. Cohen, R. DeVore, P. Petrushev, and H. Xu. Nonlinear approximation and the space $\mathrm{BV}(\mathrm{R}^2)$. *American Journal of Mathematics*, 121:587–628, 1999.

[18] R. R. Coifman and D. Donoho. Translation invariant denoising. In A. Antoine and G. Oppenheim, editors, *Wavelets in Statistics*, pages 125–150. Springer, New York, 1995.

[19] R. R. Coifman and A. Sowa. Combining the calculus of variations and wavelets for image enhancement. *Applied and Computational Harmonic Analysis*, 9(1):1–18, July 2000.

[20] R. R. Coifman and A. Sowa. New methods of controlled total variation reduction for digital functions. *SIAM Journal on Numerical Analysis*, 39(2):480–498, 2001.

[21] F. Dibos and G. Koepfler. Global total variation minimization. *SIAM Journal on Numerical Analysis*, 37(2):646–664, 2000.

[22] M. N. Do and V. Vetterli. Contourlets. In J. Stöckler and G. Welland, editors, *Beyond Wavelets*, pages 1–27. Academic Press, New York, 2003.

[23] D. L. Donoho. De-noising by soft thresholding. *IEEE Transactions on Information Theory*, 41:613–627, 1995.

[24] D. L. Donoho. Orthonormal ridglets and linear singularities. *SIAM Journal on Mathematical Analysis*, 31(5):1062–1099, 2000.

[25] D. L. Donoho and I. M. Johnstone. Ideal spatial adaptation by wavelet shrinkage. *Biometrica*, 81(3):425–455, 1994.

[26] E. C. DuFort and S. P. Frankel. Stability conditions in the numerical treatment of parabolic differential equations. *Mathematical Tables and Other Aids to Computation*, 7:135–152, 1953.

[27] S. Durand and J. Froment. Reconstruction of wavelet coefficients using total-variation minimization. *SIAM Journal on Scientific Computing*, 24(5):1754–1767, 2003.

[28] S. Durand and M. Nikolova. Restoration of wavelet coefficients by minimizing a specially designed objective function. In O. Faugeras and N. Paragios, editors, *Proc. Second IEEE Workshop on Geometric and Level Set Methods in Computer Vision*, Nice, France, Oct. 2003. INRIA.

[29] X. Feng and A. Prohl. Analysis of total variation flow and its finite element approximations. *ESAIM: Mathematical Models and Methods in the Applied Sciences*, 37(3):533–556, 2002.

[30] A. F. Filippov. *Differential Equations with Discontinuous Righthand Sides*. Kluwer, Dordrecht, 1988.

[31] W. Förstner and E. Gülch. A fast operator for detection and precise location of distinct points, corners and centres of circular features. In *Proc. ISPRS Intercommission Conference on Fast Processing of Photogrammetric Data*, pages 281–305, Interlaken, Switzerland, June 1987.

[32] G. Gerig, O. Kübler, R. Kikinis, and F. A. Jolesz. Nonlinear anisotropic filtering of MRI data. *IEEE Transactions on Medical Imaging*, 11:221–232, 1992.

[33] B. Jawerth, P. Lin, and E. Sinzinger. Lattice Boltzmann models for anisotropic diffusion of images. *Journal of Mathematical Imaging and Vision*, 11:231–237, 1999.

[34] S. L. Keeling and R. Stollberger. Nonlinear anisotropic diffusion filters for wide range edge sharpening. *Inverse Problems*, 18:175–190, Jan. 2002.

[35] N. G. Kingsbury. Complex wavelets for shift invariant analysis and filtering of signals. *Journal of Applied and Computational Harmonic Analysis*, 10(3):234–253, 2001.

[36] D. Labate, W. Lim, G. Kutyniok, and G. Weiss. Sparse multidimensional representation using shearlets. In M. Papadakis, A. F. Laine, and M. A. Unser, editors, *Wavelets XI*, volume 5914 of *Proceedings of SPIE*, pages 254–262, Bellingham, 2005. SPIE Press.

[37] T. Lu, P. Neittaanmäki, and X.-C. Tai. A parallel splitting up method and its application to Navier–Stokes equations. *Applied Mathematics Letters*, 4(2):25–29, 1991.

[38] F. Malgouyres. Combining total variation and wavelet packet approaches for image deblurring. In *Proc. First IEEE Workshop on Variational and Level Set Methods in Computer Vision*, pages 57–64, Vancouver, Canada, July 2001. IEEE Computer Society Press.

[39] F. Malgouyres. Mathematical analysis of a model which combines total variation and wavelet for image restoration. *Inverse Problems*, 2(1):1–10, 2002.

[40] Y. Meyer. *Oscillating Patterns in Image Processing and Nonlinear Evolution Equations*, volume 22 of *University Lecture Series*. AMS, Providence, 2001.

[41] P. Mrázek and J. Weickert. Rotationally invariant wavelet shrinkage. In B. Michaelis and G. Krell, editors, *Pattern Recognition*, volume 2781 of *Lecture Notes in Computer Science*, pages 156–163, Berlin, 2003. Springer.

[42] P. Mrázek, J. Weickert, and G. Steidl. Diffusion-inspired shrinkage functions and stability results for wavelet denoising. *International Journal of Computer Vision*, 64(2/3):171–186, Sept. 2005.

[43] P. Mrázek, J. Weickert, G. Steidl, and M. Welk. On iterations and scales of nonlinear filters. In O. Drbohlav, editor, *Proc. Eighth Computer Vision Winter Workshop*, pages 61–66, Valtice, Czech Republic, Feb. 2003. Czech Pattern Recognition Society.

[44] P. Perona and J. Malik. Scale space and edge detection using anisotropic diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12:629–639, 1990.

[45] A. Petrovic, O. Divorra Escoda, and P. Vandergheynst. Multiresolution segmentation of natural images: From linear to non-linear scale-space repesentations. *IEEE Transactions on Image Processing*, 13(8):1104–1114, 2004.

[46] I. Pollak, A. S. Willsky, and H. Krim. Image segmentation and edge enhancement with stabilized inverse diffusion equations. *IEEE Transactions on Image Processing*, 9(2):256–266, Feb. 2000.

[47] T. Preußer and M. Rumpf. An adaptive finite element method for large scale image processing. *Journal of Visual Communication and Image Representation*, 11(2):183–195, June 2000.

[48] J. L. Richardson, R. C. Ferrell, and L. N. Long. Unconditionally stable explicit algorithms for nonlinear fluid dynamics problems. *Journal of Computational Physics*, 104:69–74, 1993.

[49] L. I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D*, 60:259–268, 1992.

[50] G. Sapiro and D. L. Ringach. Anisotropic diffusion of multivalued images with applications to color filtering. *IEEE Transactions on Image Processing*, 5(11):1582–1586, 1996.

[51] J. Shen. A note on wavelets and diffusion. *Journal of Computational Analysis and Applications*, 5(1):147–159, 2003.

[52] J. Shen and G. Strang. On wavelet fundamental solutions to the heat equation – heatlets. *Journal of Differential Equations*, 161(2):403–421, 2000.

[53] E. P. Simoncelli. Design of multidimensional derivative filters. In *Proc. 1994 IEEE International Conference on Image Processing*, volume 1, pages 790–793, Austin, TX, Nov. 1994.

[54] N. Sochen, R. Kimmel, and F. Bruckstein. Diffusions and confusions in signal and image processing. *Journal of Mathematical Imaging and Vision*, 14(3):195–210, May 2001.

[55] G. Steidl, J. Weickert, T. Brox, P. Mrázek, and M. Welk. On the equivalence of soft wavelet shrinkage, total variation diffusion, total variation regularization, and SIDEs. *SIAM Journal on Numerical Analysis*, 42(2):686–713, 2004.

[56] V. I. Tsurkov. An analytical model of edge protection under noise suppression by anisotropic diffusion. *Journal of Computer and Systems Sciences International*, 39(3):437–440, 2000.

[57] W. Wang. On the design of optimal derivative filters for coherence-enhancing diffusion filtering. In *Proc. 2004 International Conference on Computer Graphics, Imaging and Visualization*, pages 35–40, Penang, Malaysia, July 2004. IEEE Computer Society Press.

[58] J. Weickert. Scale-space properties of nonlinear diffusion filtering with a diffusion tensor. Technical Report 110, Laboratory of Technomathematics, University of Kaiserslautern, Germany, Oct. 1994.

[59] J. Weickert. *Anisotropic Diffusion in Image Processing.* Teubner, Stuttgart, 1998.

[60] J. Weickert and T. Brox. Diffusion and regularization of vector- and matrix-valued images. In M. Z. Nashed and O. Scherzer, editors, *Inverse Problems, Image Analysis, and Medical Imaging*, volume 313 of *Contemporary Mathematics*, pages 251–268. AMS, Providence, 2002.

[61] J. Weickert and H. Scharr. A scheme for coherence-enhancing diffusion filtering with optimized rotation invariance. *Journal of Visual Communication and Image Representation*, 13(1/2):103–118, 2002.

[62] J. Weickert, B. M. ter Haar Romeny, and M. A. Viergever. Efficient and reliable schemes for nonlinear diffusion filtering. *IEEE Transactions on Image Processing*, 7(3):398–410, Mar. 1998.

[63] M. Welk, J. Weickert, and G. Steidl. A four-pixel scheme for singular differential equations. In R. Kimmel, N. Sochen, and J. Weickert, editors, *Scale-Space and PDE Methods in Computer Vision*, volume 3459 of *Lecture Notes in Computer Science*, pages 585–597, Berlin, 2005. Springer.

[64] M. Welk, J. Weickert, and G. Steidl. From tensor-driven diffusion to anisotropic wavelet shrinkage. In H. Bischof, A. Leonardis, and A. Pinz, editors, *Computer Vision – ECCV 2006*, volume 3951 of *Lecture Notes in Computer Science*, pages 391–403, Berlin, 2006. Springer.