A Wavelet Regularization for Nonlinear Diffusion Equations

Joana Soares
Departamento de Matemática
Universidade de Minho
Campus de Gualtar
4710-057 Braga, Portugal

Gerd Teschke* and Mariya Zhariy
Fachbereich Mathematik
Universität Bremen
PF 33 04 40
28334 Bremen, Germany

September 13, 2004

Abstract

We are concerned with a wavelet–based treatment of nonlinear diffusion equations in the context of image processing. In particular, we focus on the Perona–Malik model as a suitable instrument for smoothing images while preserving edges. We are not exploring a complete new method of solving the Perona–Malik equation but, inspired by Weickert et.al., we develop a new variant, based on wavelet technology, of regularizing this specific equation. By carefully choosing the generators, we are able to derive all inner products and integrals of the weak formulation with high efficiency. We prove that the proposed scheme overcomes the ill-posedness of the nonlinear Perona–Malik diffusion equation and illustrate the obtained results by some numerical experiments.

keywords, phrases: Nonlinear diffusion, regularization of Perona–Malik equations, smoothing, edge preservation, interpolating and operator adapted wavelets

1 Introduction

An important field in image processing is the restoration of the 'true' or the 'cartoon' image from an observation. In almost all applications the observation is a noisy and blurred version of the true image. There are several ways to attack the restoration problem, e.g. by solving a related variational formulation or by solving a partial differential equation.

* *Corresponding author:* email: teschke@math.uni-bremen.de, phone: 0049(0)421 218 9394

In this paper we focus on image restoration methods by means of partial differential equations which induce a smoothing while keeping the edge information. A very proper model for that task was introduced by Perona and Malik, see [13], which results in a nonlinear diffusion equation

$$\partial_t u = \mathrm{div}(g(|\nabla u|^2)\nabla u), \tag{1}$$

where g is supposed to be a smooth, non-increasing function with $g(0) = 1, g(x) \geq 0$ and $g(x)$ tending to zero at infinity. Model (1) might be extended by choosing tensor–valued diffusivities. Here we limit the analysis to the scalar–valued case. In order to solve the restoration problem, the function g should be chosen such that the diffusion process described by (1) behaves like a classical diffusion for small gradients and does nothing for large gradients. In the pioneering paper of Perona and Malik the following model is suggested

$$g(s^2) := \frac{1}{1 + s^2/\lambda^2}. \tag{2}$$

A drawback using this specific function g is that it induces ill-posedness of (1). It is shown e.g. in [19] that there exist also diffusivities g which induce well-posedness of (1). A necessary condition to ensure existence and uniqueness of the solution is that the flow-function $\Phi(s) := g(|s|^2)s$ must be nondecreasing, see [17] (classical theory of ordinary differential equations) or [2] (theory of maximal monotone operators).

The main goal of this paper is to overcome the ill-posedness of equation (1) by a reasonable wavelet–based regularization which is induced by an adequate discretization. In principle, we follow the idea of Weickert et.al., see [9, 20, 19], where the discretization is based on a finite-differences method that provides certain filtering properties, well-posedness, mean conservation etc. Weikert's idea goes as follows: discretize spatially problem (1) by finite differences and obtain a system of ordinary differential equations

$$\frac{du}{dt} = A(u)u. \tag{3}$$

It is shown in [20] that under certain conditions on the matrix $A(u)$ - such as $A(u) \in C^1$, symmetry, non-negative off diagonals, irreducibility - the system (3) exhibits well-posedness, continuous dependency on the initial values and on the right hand side of (3), mean-value conservation, extremum principle, and, finally, convergence to the stationary constant state.

The *main result of this paper* is the proof that the resulting system of ordinary differential equations obtained by a wavelet–based discretization fulfills all the required conditions to circumvent ill-posedness, and, moreover, that all building blocks combined here result in a very thrifty numerical scheme.

The remaining sections of the paper are organized as follows: in Section 2 we summarize all required ingredients; in Section 3 we present our wavelet–based discretization and prove the main result; Section 4 is concerned with the concrete application to the one– and two–dimensional cases; finally, in Section 5 we present numerical experiments.

A remark to the philosophy of this paper – the intention is not to rediscover all the nice techniques to numerically solve the Perona–Malik diffusion equation. This paper mainly aims to show how several buildings blocks taken from wavelet analysis can be reasonably combined to construct a new variant of the Perona–Malik regularization.

2 Preliminaries

Let Ω be a sufficiently smooth bounded domain in $\mathbb{R}^d$ and $T > 0$. In the original setting, the equation (1) is defined for all functions $u : (0,T) \times \Omega \to \mathbb{R}$ with continuous derivative on $(0,T)$ and such that $u(t,.) \in C^2(\Omega)$ for all $t \in (0,T)$. We denote the space of such functions by $C^1(0,T;C^2(\Omega))$. The weak formulation of the problem is defined for functions in $C^1(0,T;H^1(\Omega))$. In order to achieve a spatial discretization of the continuous Perona–Malik equation, we search for a solution in a finite dimensional subspace, let say V_j, of $H^1(\Omega)$. Here we do not focus on applying the classical Galerkin approach to approximate the solution of the continuous problem since we do not have existence of a continuous solution in general, see [9].

Instead of the Galerkin idea we limit the problem to choosing some Ansatz space V_j for some fixed j and to solve the related discretized problem. For our purpose, an adequate space V_j means that we are able to overcome or to circumvent most of the appearing difficulties caused by the weak formulation. This requires test functions that allow a thrifty computation of all the inner products and integrals. All this can be performed by generators and wavelets in a multiresolution framework.

2.1 Generator functions and multiresolution on the torus

We consider a very natural setting for images, consisting of periodic functions on the bounded domain $\Omega := [0,1]^d$. Hence, we need to briefly review the concept of multiresolution analysis on the torus. To this end, instead of the d-dimensional unit interval Ω we take the torus $T^d := \mathbb{R}^d/\mathbb{Z}^d$. Let us now consider the following periodization mapping from $L^2(\mathbb{R}^d)$ to $L^2(T^d)$

$$[f](x) = \sum_{l\in\mathbb{Z}^d} f(x+l). \tag{4}$$

Note that every scaling function ϕ satisfies the so called scaling equation with a filter mask $h = \{h_k\}_{k\in\mathbb{Z}^d}$, as follows:

$$\phi(x) = m^{1/2} \sum_{k\in\mathbb{Z}^d} h_k \phi(Mx-k), \tag{5}$$

where M is a so-called scaling matrix and $m = |\det M|$. Functions that satisfy (5) are also called (M,h)-*refinable*. Assuming that ϕ is a scaling function which generates a multiresolution analysis of $L^2(\mathbb{R}^d)$, the latter mapping can be used to construct periodized scaling functions

$$[\phi_k^j] = \sum_{l\in\mathbb{Z}^d} \phi_k^j(\cdot + l) = m^{j/2} \sum_{l\in\mathbb{Z}^d} \phi(M^j(\cdot + l) - k), \tag{6}$$

which fulfill a rescaling property (5) as well, i.e. we have

$$[\phi](x) = \sum_{l\in\mathbb{Z}^d} \phi(x+l) = \sum_{l\in\mathbb{Z}^d} m^{1/2} \sum_{k\in\mathbb{Z}^d} h_k \phi(M(x+l) - k) = \sum_{k\in\mathbb{Z}^d} h_k [\phi_k^1](x) \tag{7}$$

and for the rescaled versions

$$[\phi_k^j](x) = \sum_{n\in\mathbb{Z}^d} h_n [\phi_{Mk+n}^{j+1}](x) = \sum_{p\in\mathbb{Z}^d} h_{p-Mk} [\phi_p^{j+1}](x). \tag{8}$$

Note that the periodization procedure preserves the compactness property of scaling functions, which is characterized by the finite filter mask, cf. (5) and (7).

Finally, defining $\mathbb{Z}^{d,j} := \mathbb{Z}^d/(M^j\mathbb{Z}^d)$ one can show that the spaces

$$[V_j] := \overline{\mathrm{span}\{[\phi_k^j], k \in \mathbb{Z}^{d,j}\}}, \quad j \in \mathbb{N}_0, \tag{9}$$

form a multiresolution analysis of $L^2(T^d)$, see for details [6]. The construction principle holds for the orthonormal as well as for the biorthormal framework, which is of interest for our approach.

2.2 Interpolating generator functions and quadrature rule

In order to construct easy to implement quadrature rules, the usage of interpolating Ansatz functions is preferred. An interpolating scaling function fulfills by definition

$$\phi(k) = \delta_{0,k}, \quad k \in \mathbb{Z}^d. \tag{10}$$

A simple construction scheme is given, in the one dimensional case, by the so-called Deslauriers-Dubuc scaling functions, see [12, 8],

$$\phi^{2N}(x) := (\varphi^N * \check{\varphi}^N)(x), \tag{11}$$

where $\check{g}(\cdot) = g(-\cdot)$, and φ^N denotes here a Daubechies generator of order N. Throughout the paper we shall use this kind of compactly supported scaling functions. This is reasonable since it is shown in [12, 8] that ϕ^{2N} generate a multiresolution analysis. Moreover, the construction of smooth duals is ensured by [3].

In order to develop a quadrature rule, we introduce an interpolation projector by

$$\pi_j(v) := m^{-j/2} \sum_{k\in\mathbb{Z}^{d,j}} v(M^{-j}k)[\phi_k^j]. \tag{12}$$

For this interpolation projector we quote the following $L_\infty-$ error estimate, see [11],

Theorem 2.1. *Let π_j be the interpolation operator* (12) *associated with a continuous, compactly supported (M, h)-refinable function ϕ, which is l_2-stable and belongs to the Sobolev space $W^n(L_1(\mathbb{R}^d))$ for some $n \in \mathbb{N}$. M is assumed to be an isotropic scaling matrix, i.e. there exist some $A, B \in \mathrm{GL}(d)$ and a diagonal matrix $D = \mathrm{diag}\{\lambda_1, \dots, \lambda_d\}$ with $|\lambda_1| = \dots = |\lambda_d|$, so that $M = ADB$. Then there exists $C > 0$, so that for all $v \in W^{n+1}(L_\infty(T^d))$ the error estimate*

$$\|v - \pi_j(v)\|_{L_\infty(T^d)} \leq C r(M)^{-j(n+1)},$$

holds true. Here $r(M)$ denotes the spectral radius of M.

Now we may define in a usual way a quadrature rule for the integral $\int_\Omega v$ by

$$Q\Big(\int_\Omega v dx\Big) := \int_\Omega \pi_j(v) dx = m^{-j} \sum_{k \in \mathbb{Z}^{d,j}} v(M^{-j}k). \tag{13}$$

As a simple consequence of Theorem 2.1 the following error estimate for this quadrature rule (13) holds

$$\left| \int_\Omega v(x) dx - \int_\Omega \pi_j(v) dx \right| \leq C r(M)^{-j(n+1)}.$$

2.3 Generator functions via integration and differentiation

As a final refinement of our Ansatz system we apply a method that allows a construction of scaling functions by integration and differentiation, see e.g. [10, 16, 4]. This refinement enables us to use differential operator adapted generator functions which 'absorb' the derivatives.

The basic idea is to start with a dual pair of generator functions and to apply some integration and differentiation in order to obtain another dual pair of scaling functions (now equipped this sort of 'derivative absorbing property'). The next quoted theorem states how the new dual pair can be obtained.

Theorem 2.2. *Let $\phi, \tilde{\phi} \in L^2(\mathbb{R})$ be a dual pair of compactly supported scaling functions with symbols $\mathbf{H}$ and $\tilde{\mathbf{H}}$ respectively, which generates a dual MRA of $L^2(\mathbb{R})$. Let $\tilde{\phi} \in H^1(\mathbb{R})$ and $\int_{\mathbb{R}} \phi = \int_{\mathbb{R}} \tilde{\phi} = 1$. Then there exists a dual pair $\varphi, \tilde{\varphi} \in L^2(\mathbb{R})$ of compactly supported scaling functions with*

$$\varphi'(x) = \phi(x+1) - \phi(x), \qquad \tilde{\phi}'(x) = \tilde{\varphi}(x) - \tilde{\varphi}(x-1).$$

The symbols $\mathbf{h}$ and $\tilde{\mathbf{h}}$ of φ and $\tilde{\varphi}$ satisfy the following relations:

$$\mathbf{h}(z) = \frac{1+z}{2} \mathbf{H}(z), \qquad \tilde{\mathbf{h}}(z) = \frac{2z}{1+z} \tilde{\mathbf{H}}(z).$$

The functions φ and $\tilde{\varphi}$ also generate a dual MRA of $L^2(\mathbb{R})$.

3 Wavelet regularization by spatial discretization

This section is devoted to the main result of this paper, namely to show that our wavelet regularization by spatially discretizing the Perona–Malik equation results in a well–posed problem for which we present an implicit scheme that is solved by a fixed point iteration.

3.1 Semi-discrete formulation

Let Ω be as before, $T > 0$ and u, u_0 be sufficiently smooth so that the following boundary value problem (which might be ill-posed for a certain class of functions g) is well-defined

$$\begin{aligned} \partial_t u &= \operatorname{div}(g(|\nabla u|^2)\nabla u) && \text{in } (0,T)\times\Omega && (14)\\ u_{|t=0} &= u_0 && \text{in } \Omega && (15)\\ u_{|x_i=0} &= u_{|x_i=1}, i=1,\ldots,d && \text{in } (0,T). && (16)\end{aligned}$$

According to the periodic boundary conditions we choose $[V_j]$ as the underlying Ansatz space, which is a finite dimensional subspace of $H^1_p(\Omega)$ if the generator function was properly chosen. Note that the existence of boundary values of Sobolev functions follows from the Trace Theorem for Sobolev functions, see [1], and that there exists a $L^2(\partial\Omega)$-valued trace of a $H^1(\Omega)$-function onto $\partial\Omega$ such that every $v \in H^1_p(\Omega)$ is periodic almost everywhere on the boundary $\partial\Omega$. Thus the resulting semi-discrete problem is to find some $u^j \in C^1(0,T;[V_j])$ such that the variational equation

$$\begin{aligned} \big(\partial_t u^j(t), v\big) + \Big(g(|\nabla u^j(t)|^2)\nabla u^j(t), \nabla v\Big) &= 0 && (17)\\ u^j(0) &= u^j_0 && (18)\end{aligned}$$

is fulfilled for all $v \in [V_j]$ and $t \in (0,T)$. In case of discrete initial values u_0 we can choose $[V_j]$ so that $u_0 \in [V_j]$. Then we can use u_0 instead of u^j_0 in (18).

3.2 Well-posedness

Let $\{\phi_\lambda, \lambda \in \Lambda_j\}$, where $\Lambda_j := \{\lambda = (j,k), k \in \mathbb{Z}^{d,j}\}$, be a basis of $[V_j]$ and let N_j be its dimension, i.e. $N_j = \#\Lambda_j$. With this notation, $u^j \in C^1(0,T;[V_j])$ has the following expansion

$$u^j = \sum_{\lambda\in\Lambda_j} c_\lambda \phi_\lambda \tag{19}$$

with coefficients $c_\lambda \in C^1(0,T)$. Furthermore, let $A \in \mathbb{R}^{N_j\times N_j}$ denote the mass matrix associated with the nodal basis of $[V_j]$, i.e.

$$A_{\lambda\mu} := \big(\phi_\lambda, \phi_\mu\big), \quad \lambda,\mu \in \Lambda_j. \tag{20}$$

Theorem 3.1. *Let $u_0 \in [V_j]$ and $T > 0$. Assume the mass matrix A to be regular and let*

$$G(c)(x) := g\Big(\big|\sum_{\nu\in\Lambda_j} c_\nu \nabla\phi_\nu(x)\big|^2\Big)c, \quad x \in \Omega, c \in \mathbb{R}^{N_j} \tag{21}$$

be Lipschitz continuous with respect to c with Lipschitz constant $L > 0$. Then there exists a unique set of coefficients $c_\lambda \in H^1(0,T)$ such that $u^j = \sum_{\lambda\in\Lambda_j} c_\lambda\phi_\lambda$ is the unique solution of the discrete problem (17)-(18).

Proof. Taking as test functions ϕ_μ, $\mu \in \Lambda_j$, and using $\partial_t u^j(t) = \sum_{\lambda\in\Lambda_j} c'_\lambda(t)\phi_\lambda$ and $\nabla u^j(t) = \sum_{\lambda\in\Lambda_j} c_\lambda(t)\nabla\phi_\lambda$, the semi-discrete problem (17)-(18) results in an equivalent ODE system:

$$\begin{aligned}\sum_{\lambda\in\Lambda_j} c'_\lambda(t)\big(\phi_\lambda,\phi_\mu\big) + \sum_{\lambda\in\Lambda_j} c_\lambda(t)\Big(g\big(\big|\sum_{\nu\in\Lambda_j} c_\nu(t)\nabla\phi_\nu\big|^2\big)\nabla\phi_\lambda,\nabla\phi_\mu\Big) &= 0 &(22)\\ c_\lambda(0) &= c_{0,\lambda}, &(23)\end{aligned}$$

which can be rewritten as

$$\begin{aligned}Ac'(t) + B(c(t))c(t) &= 0 &(24)\\ c(0) &= c_0, &(25)\end{aligned}$$

where $c_{0,\lambda} = (u_0, \tilde{\phi}_\lambda)$ and $B : \mathbb{R}^{N_j} \to \mathbb{R}^{N_j\times N_j}$ represents the nonlinear term

$$B_{\lambda\mu}(c) := \Big(g\big(\big|\sum_{\nu\in\Lambda_j} c_\nu\nabla\phi_\nu\big|^2\big)\nabla\phi_\lambda,\nabla\phi_\mu\Big), \quad c \in \mathbb{R}^{N_j}. \tag{26}$$

The regularity of A allows us to transform the initial value problem (24)-(25) into an integral representation

$$c(t) = c_0 - \int_0^t A^{-1}B(c(s))c(s)ds. \tag{27}$$

Note, that $c \in C^1(0,T)^{N_j}$ solves the initial value problem (24)-(25) if and only if $c \in C^0([0,T])^{N_j}$ and c solves the integral equation (27). Introducing the space $X := C^0([0,T],\mathbb{R}^{N_j})$ equipped with the norm $\|c\|_r := \sup_{0\le t\le T} e^{-rt}\|c(t)\|_2$ (this is then a Banach space), the solution c of (27) has to be a fixed point for $S : X \to X$ where S is defined as

$$S(c)(t) := c_0 - \int_0^t A^{-1}B(c(s))c(s)ds. \tag{28}$$

The existence of a fixed point is shown by proving that S is a *contraction* for one r, i.e.

$$\|S(c) - S(\tilde{c})\|_r \le q\|c - \tilde{c}\|_r \tag{29}$$

for some $0 < q < 1$. For $t \in [0,T]$ we have

$$\begin{aligned} \big\|S(c)(t) - S(\tilde{c})(t)\big\|_2 &= \Big\|\int_0^t A^{-1}\big(B(c(s))c(s) - B(\tilde{c}(s))\tilde{c}(s)\big)ds\Big\|_2 \\ &\leq \|A^{-1}\| \int_0^t \Big\|B(c(s))c(s) - B(\tilde{c}(s))\tilde{c}(s)\Big\|_2 ds \\ &\leq \|A^{-1}\| \int_0^t \sum_{\lambda\in\Lambda_j} \Big|\big(B(c(s))c(s) - B(\tilde{c}(s))\tilde{c}(s)\big)_\lambda\Big| ds. \quad (30) \end{aligned}$$

Additionally, for $s \in [0,T]$ we may compute

$$\begin{aligned} &\Big(B(c(s))c(s) - B(\tilde{c}(s))\tilde{c}(s)\Big)_\lambda = \sum_{\mu\in\Lambda_j} B_{\lambda\mu}(c(s))c_\mu(s) - B_{\lambda\mu}(\tilde{c}(s))\tilde{c}_\mu(s) \\ &= \sum_{\mu\in\Lambda_j}\Big(g\big(\big|\sum_{\nu\in\Lambda_j} c_\nu(s)\nabla\phi_\nu\big|^2\big)\nabla\phi_\lambda, \nabla\phi_\mu\Big)c_\mu(s) - \Big(g\big(\big|\sum_{\nu\in\Lambda_j} \tilde{c}_\nu(s)\nabla\phi_\nu\big|^2\big)\nabla\phi_\lambda, \nabla\phi_\mu\Big)\tilde{c}_\mu(s) \\ &= \sum_{\mu\in\Lambda_j}\int_\Omega \Big(G_\mu(c(s))(x) - G_\mu(\tilde{c}(s))(x)\Big)\nabla\phi_\lambda(x)\nabla\phi_\mu(x)dx. \quad (31) \end{aligned}$$

The Lipschitz continuity of G

$$\Big\|G(c)(x) - G(\tilde{c})(x)\Big\|_2 \leq L\big\|c - \tilde{c}\big\| \quad \forall x \in \Omega; c, \tilde{c} \in \mathbb{R}^{N_j} \quad (32)$$

and the estimate

$$\Big|G_\mu(c(t))(x) - G_\mu(\tilde{c}(t))(x)\Big| \leq \Big\|G(c(t))(x) - G(\tilde{c}(t))(x)\Big\|_2 \quad (33)$$

results, for $s \in [0,T]$, in

$$\begin{aligned} &\Big|\Big(B(c(s))c(s) - B(\tilde{c}(s))\tilde{c}(s)\Big)_\lambda\Big| \\ &\leq \sum_{\mu\in\Lambda_j}\int_\Omega \Big|G_\mu(c(s))(x) - G_\mu(\tilde{c}(s))(x)\Big|\big|\nabla\phi_\lambda(x)\nabla\phi_\mu(x)\big|dx \\ &\leq L\big\|c(s) - \tilde{c}(s)\big\|_2 \sum_{\mu\in\Lambda_j}\int_\Omega \big|\nabla\phi_\lambda(x)\nabla\phi_\mu(x)\big|dx. \quad (34) \end{aligned}$$

Combining the latter estimate with estimate (30) we obtain

$$\begin{aligned} \big\|S(c)(t) - S(\tilde{c})(t)\big\|_2 &\leq \|A^{-1}\| \int_0^t L\big\|c(s) - \tilde{c}(s)\big\|_2 ds \sum_{\lambda,\mu\in\Lambda_j}\int_\Omega \big|\nabla\phi_\lambda(x)\nabla\phi_\mu(x)\big|dx \\ &\leq C\|c - \tilde{c}\|_r \int_0^t e^{rs}ds, \quad (35) \end{aligned}$$

where $C = \|A^{-1}\| L \sum_{\lambda,\mu\in\Lambda_j} \int_\Omega \big|\nabla\phi_\lambda(x)\nabla\phi_\mu(x)\big| dx$ is a finite constant. With the help of (35) we finally obtain

$$\begin{aligned} e^{-rt}\big\|S(c)(t) - S(\tilde{c})(t)\big\|_2 &\leq C\|c-\tilde{c}\|_r \int_0^t e^{-r(t-s)} ds \\ &\leq \frac{C}{r}\|c-\tilde{c}\|_r \end{aligned} \tag{36}$$

which shows that, for suitably chosen r ($r > 0$ and $\frac{C}{r} < 1$), S is contractive. The application of Banach's Fixed Point Theorem completes the proof. □

3.3 Conservation of well–posedness after applying quadrature rules

Thanks to the Dahmen & Micchelli scheme, see [5], we have a method for exactly computing the entries of the mass matrix A in (24). The remaining task is to numerically approximate the integral term B in (24). We aim to apply the quadrature rule introduced in Subsection 2.2. This yields the following approximation $\tilde{B}$ of B

$$\begin{aligned} \tilde{B}_{\lambda\mu}(c) &= Q\big(B_{\lambda\mu}(c)\big) = Q\Big(\big(g\big(\big|\sum_{\nu\in\Lambda_j} c_\nu \nabla\phi_\nu\big|^2\big)\nabla\phi_\lambda, \nabla\phi_\mu\big)\Big) \\ &= m^{-j} \sum_{k\in\mathbb{Z}^{d,j}} \Big(g\big(\big|\sum_{\nu\in\Lambda_j} c_\nu\nabla\phi_\nu\big|^2\big)\nabla\phi_\lambda\nabla\phi_\mu\Big)\Big|_{M^{-j}k}, \quad \lambda,\mu\in\Lambda_j, \quad c\in\mathbb{R}^{N_j}. \end{aligned} \tag{37}$$

Hence, we obtain a perturbed system to be solved

$$Ac'(t) + \tilde{B}(c(t))c(t) = 0 \tag{38}$$
$$c(0) = c_0. \tag{39}$$

Thus, we have to ensure that system (38)-(39) is solvable and that its solution approximates the solution of (17)-(18). To this end, we reformulate our problem: let $c(t)$ be a solution of

$$c'(t) = F(t, c(t)) \tag{40}$$
$$c(0) = c_0 \tag{41}$$

for all $t \in (0, T]$, where $F(t,c) := -A^{-1}B(c)c$. Then the problem is to show that the solution continuously depends on the right hand side F. The initial values are not affected by applying the quadrature rule. Thus, by [18], we have to show that $F(t,c)$ is continuous, and that there exists an $\alpha > 0$ such that F is, with respect to c, global Lipschitz continuous on

$$D_\alpha := \{(t,d)|\ t\in[0,T], \|d-c\| \leq \alpha\}.$$

It follows that for each $\varepsilon > 0$ there exists some $\delta > 0$ such that the solution $\tilde{c}(t)$ of the perturbed problem

$$c'(t) = \tilde{F}(t, c(t)) \tag{42}$$
$$c(0) = c_0 \tag{43}$$

with continuous $\tilde{F}$ satisfying

$$\|\tilde{F}(t,d) - F(t,d)\| < \delta \quad \text{for} \quad \|d - c\| < \alpha \tag{44}$$

exists and that the estimate

$$\|\tilde{c}(t) - c(t)\| < \varepsilon$$

holds true.

Note that the global Lipschitz-continuity of F on D_α is a direct consequence of the Lipschitz-continuity of G.

The next proposition verifies condition (44).

Proposition 3.1. *Assume all the condition required in Theorem 2.1. Moreover, assume A to be regular and that B and $\tilde{B}$ fulfill a Lipschitz condition. Let $\tilde{F}(t,c) = -A^{-1}\tilde{B}(c)c$. Then there exists some δ such that*

$$\left\|\tilde{F}(t,d) - F(t,d)\right\| < \delta.$$

Proof.

$$\begin{aligned}
\left\|\tilde{F}(t,d) - F(t,d)\right\| &= \left\|A^{-1}\tilde{B}(d)d - A^{-1}B(d)d\right\| \leq \|A^{-1}\|\left\|\tilde{B}(d) - B(d)\right\|\|d\| \\
&\leq \|A^{-1}\|\Big(\|\tilde{B}(d) - \tilde{B}(c)\| + \|\tilde{B}(c) - B(c)\| + \|B(c) - B(d)\|\Big)\|d\| \\
&\leq \|A^{-1}\|\Big(C_1\|d - c\| + C_2 r(M)^{-j(n+1)} + C_3\|d - c\|\Big)\|d\|
\end{aligned}$$

The norm $\|d\|$ is finite on D_α since, by Theorem 3.1, the solution c of (40)-(41) is continuous on $[0,T]$ and therefore bounded on D_α. It follows:

$$\left\|\tilde{F}(t,d) - F(t,d)\right\| \leq \|A^{-1}\|\big((C_1 + C_3)\alpha + C_2 r(M)^{-j(n+1)}\big)C.$$

We have to choose α and j so that the expression on the right side is less then δ. $\square$

3.4 Time discretization

We suggest to use the implicit Euler method for approximating the time derivative. For each time step $t_n = n\tau$, $n = 1, \ldots, n_{max}$, the discretized time derivative of u^j reads as

$$\frac{u_n^j - u_{n-1}^j}{\tau}.$$

Hence we obtain the following implicit problem for $u_n^j \in [V_j]$:

$$\begin{aligned} \Big(\frac{u_n^j - u_{n-1}^j}{\tau}, v\Big) + \Big(g(|\nabla u_n^j(t)|^2)\nabla u_n^j(t), \nabla v\Big) &= 0, \quad \forall v \in [V_j] & (45)\\ u_0^j &= u_0. & (46) \end{aligned}$$

Since the implicit Euler method is absolutely stable for all increments $\tau > 0$, we immediately have that the global discretization error is bounded in time, see e.g. [7, 14, 15]. Expressing $u_n^j = \sum_{\lambda \in \Lambda_j} c_\lambda^n \phi_\lambda$, the discrete problem reads as

$$\begin{aligned} \sum_{\lambda\in\Lambda_j}(c_\lambda^n - c_\lambda^{n-1})(\phi_\lambda, \phi_\mu) + \tau \sum_{\lambda\in\Lambda_j} c_\lambda^n \big(g\big(\big|\sum_{\nu\in\Lambda_j} c_\nu^n \nabla\phi_\nu\big|^2\big)\nabla\phi_\lambda, \nabla\phi_\mu\big) &= 0, \qquad \mu \in \Lambda_j\\ c^0 &= c_0. \end{aligned}$$

After applying the quadrature rule we have

$$\begin{aligned} A(c^n - c^{n-1}) + \tau\tilde{B}(c^n)c^n &= 0 & (47)\\ c^0 &= c_0. & (48) \end{aligned}$$

The system (47)-(48) can be solved with the help of Newton's method. To this end, we define

$$\mathcal{F}(c) := A(c - c^{n-1}) + \tau\tilde{B}(c)c. \tag{49}$$

The First–Order–Newton–Scheme requires the computation $D\mathcal{F}$ with respect to c. Formally we have

$$D\mathcal{F}(c) = A + \tau(D\tilde{B}(c)c + \tilde{B}(c)), \tag{50}$$

where $D\tilde{B}(c)$ denotes the matrix of partial derivatives of $\tilde{B}$. For later convenience we derive the product $D\tilde{B}(c)c$ explicitly

$$(D\tilde{B}(c)c)_{\lambda\mu} = \sum_{\nu\in\Lambda_j} \frac{\partial\tilde{B}_{\lambda\mu}(c)}{\partial c_\nu} c_\nu = m^{-1} \sum_{\nu\in\Lambda_j}\sum_{k\in\mathbb{Z}^{d,j}} \Big(\frac{\partial g(|\nabla u|^2)}{\partial c_\nu}\nabla\phi_\lambda\nabla\phi_\mu)\Big)\Big|_{M^{-j}k} c_\nu, \tag{51}$$

with

$$\frac{\partial g(|\nabla u|^2)}{\partial c_\nu} = g'(|\nabla u|^2)\frac{\partial(|\sum_{\kappa\in\Lambda_j} c_\kappa\nabla\phi_\kappa|^2)}{\partial c_\nu} = 2g'(|\nabla u|^2)\sum_{\kappa\in\Lambda_j} c_\kappa\nabla\phi_\kappa\nabla\phi_\nu \tag{52}$$

and $u = \sum_{\kappa\in\Lambda_j} c_\kappa\phi_\kappa$. Introducing the operator $S : \Omega \to \mathbb{R}^{N_j\times N_j}$, where $S_{\lambda\mu} = \nabla\phi_\lambda\nabla\phi_\mu$, we may write

$$\begin{aligned} (D\tilde{B}(c)c)_{\lambda\mu} &= \sum_{\nu\in\Lambda_j}\sum_{k\in\mathbb{Z}^{d,j}} \Big(2g'(|\nabla u|^2)\sum_{\kappa\in\Lambda_j} c_\kappa\nabla\phi_\kappa\nabla\phi_\nu\nabla\phi_\lambda\nabla\phi_\mu)\Big)\Big|_{M^{-j}k} c_\nu & (53)\\ &= 2\sum_{k\in\mathbb{Z}^{d,j}}\Big(g'(|\sum_{\eta\in\Lambda_j} c_\eta\nabla\phi_\eta|^2)(c^T Sc)S_{\lambda\mu})\Big)\Big|_{M^{-j}k}. & (54) \end{aligned}$$

Note that the appearing terms reduce in complexity if we make use of our special basis functions φ obtained by the integration/differentiation procedure described in Subsection 2.3.

4 The regularization scheme in one and two dimensions

In what follows we show how in the one– and two–dimensional cases all ingredients provided above must be combined.

4.1 The one–dimensional case

Let φ be a generating function calculated as prescribed in Theorem 2.2, i.e.

$$\varphi'(x) = \phi(x+1) - \phi(x) \tag{55}$$

with ϕ satisfying the interpolating property, i.e. $\phi(k) = \delta_{0,k}$. Moreover, let $[\varphi]$ be the periodization of φ as in (4). Note that for the periodization of an interpolating function ϕ we have

$$[\phi_k^j](2^{-j}p) = 2^{j/2} \sum_{l \in \mathbb{Z}} \phi(2^j(2^{-j}p + l) - k) = 2^{j/2} \delta_{k,p}^j \,, \tag{56}$$

with

$$\delta_{k,p}^j = \begin{cases} 1, & k = p + 2^j l \ \text{ for } l \in \mathbb{Z} \\ 0, & \text{otherwise} \end{cases} .$$

Next, we see that

$$\begin{aligned} \frac{d}{dx}[\varphi_k^j](x) &= 2^{j/2} 2^j \sum_{l \in \mathbb{Z}} \varphi'(2^j(x+l) - k) \\ &= 2^j 2^{j/2} \sum_{l \in \mathbb{Z}} (\phi(2^j(x+l) - k + 1) - \phi(2^j(x+l) - k)) \\ &= 2^j([\phi_{k-1}^j](x) - [\phi_k^j](x)). \end{aligned}$$

Thus, the values of the derivative on the grid $\{2^{-j}p, p = 0, \ldots, 2^j - 1\}$ are given by

$$\frac{d}{dx}[\varphi_k^j](2^{-j}p) = 2^j([\phi_{k-1}^j](2^{-j}p) - [\phi_k^j](2^{-j}p)) = 2^{3j/2}(\delta_{k-1,p}^j - \delta_{k,p}^j). \tag{57}$$

The spatial discretization requires the computation of $A^{1,j}$ and $\tilde{B}^{1,j}$. In order to compute $A^{1,j}$ we apply a Dahmen and Micchelli algorithm which ensures exact computation of the inner products. The procedure in the periodic case goes as follows: assuming j to be large enough, we have the following structure of the mass matrix entries

$$A_{kl}^{1,j} = \int_0^1 [\varphi_k^j][\varphi_l^j] = \int_{\mathbb{R}} \varphi_k^j \varphi_l^j + \int_{\mathbb{R}} \varphi_{k-2^j}^j \varphi_l^j + \int_{\mathbb{R}} \varphi_k^j \varphi_{l-2^j}^j .$$

Consequently, the computation can be reduced to the known non-periodic case. The related eigenvalue problem is then obtained by the usual refinement property of φ

$$\int_{\mathbb{R}} \varphi_0^j(x) \varphi_k^j(x) dx = \sum_{p \in \mathbb{Z}} \Big(\sum_{m \in \mathbb{Z}} h_m h_{p+m-2k} \Big) \int_{\mathbb{R}} \varphi_0^j(x) \varphi_p^j(x) dx. \tag{58}$$

For the nonlinear term we obtain

$$\begin{aligned}
\tilde{B}^{1,j}_{kl}(c) &= 2^{-j} \sum_{p\in\mathbb{Z}^{1,j}} \Big(g\Big(\Big| \sum_{m\in\mathbb{Z}^{1,j}} c_m[\varphi^j_m]'\Big|^2\Big)[\varphi^j_k]'[\varphi^j_l]'\Big)\Big|_{2^{-j}p} \\
&= 2^{-j} \sum_{p\in\mathbb{Z}^{1,j}} \Big(g\Big(2^{3j}\left(c_{p+1}-c_p\right)^2\Big)\Big) 2^{3j/2}(\delta^j_{p,k-1}-\delta^j_{p,k})2^{3j/2}(\delta^j_{p,l-1}-\delta^j_{p,l})\Big) \qquad (59)
\end{aligned}$$

and consequently, $\tilde{B}^{1,j}(c)c$ becomes

$$\begin{aligned}
\left(\tilde{B}^{1,j}(c)c\right)_k &= \sum_{l\in\mathbb{Z}^{1,j}} \tilde{B}^{1,j}_{kl}(c)c_l \\
&= 2^{2j} \sum_{p\in\mathbb{Z}^{1,j}} \Big(g\Big(2^{3j}\left(c_{p+1}-c_p\right)^2\Big)\Big)(\delta_{p,k-1}-\delta_{p,k})(c_{p+1}-c_p)\Big) \\
&= 2^{2j}\big(\tilde{G}(c_k-c_{k-1})-\tilde{G}(c_{k+1}-c_k)\big), \quad k\in\mathbb{Z}^{1,j}, \qquad (60)
\end{aligned}$$

with $\tilde{G}(x) := g(2^{3j}x^2)x$.

Next, we show that the resulting systems (exact and perturbed)

$$A^{1,j}c'(t) + B^{1,j}(c)c = 0 \qquad (61)$$

$$A^{1,j}c'(t) + \tilde{B}^{1,j}(c)c = 0 \qquad (62)$$

fulfill all properties that ensure gray value conservation. In order to prove that result, we shall follow techniques provided in [20, 19].

Proposition 4.1. *The average gray value γ of u_0 defined by*

$$\gamma := \frac{1}{N_j}\int_\Omega u_0(x)dx = \frac{1}{N_j}\sum_{\lambda\in\Lambda_j} c_{0,\lambda}$$

remains unchanged after applying the semi-discrete diffusion filtering described by (61) *or* (62).

In other words, if u is a solution of the semi-discrete problem (61) *or* (62)*, then for all $t\in[0,T]$ we have*

$$\frac{1}{N_j}\int_\Omega u(x)dx = \frac{1}{N_j}\sum_{\lambda\in\Lambda_j} c_\lambda(t) = \gamma.$$

Proof. First, we observe that the coefficients of $A^{1,j}$, $B^{1,j}$, $\tilde{B}^{1,j}$ satisfy

- all row sums of $A^{1,j}$ are equal; this follows by the fact that

$$\sum_{l\in\mathbb{Z}^{1,j}} A^{1,j}_{kl} = \sum_{l\in\mathbb{Z}^{1,j}} \int_0^1 [\varphi^j_k][\varphi^j_l] = \sum_{l\in\mathbb{Z}^{1,j}} \int_0^1 [\varphi^j_0][\varphi^j_{l-k}] = \sum_{l\in\mathbb{Z}^{1,j}} \int_0^1 [\varphi^j_0][\varphi^j_l] \qquad (63)$$

  do not depend on k,

- all row sums of $B^{1,j}$ as well as of $\tilde{B}^{1,j}$ are equal to zero; this can be seen by considering

$$\sum_{l\in\mathbb{Z}^{1,j}} [\varphi_l^j]' = 2^j \sum_{l\in\mathbb{Z}^{1,j}} ([\phi_{l-1}^j] - [\phi_l^j]) = 0,$$

where in particular for $p \in \mathbb{Z}^{1,j}$ holds

$$\sum_{l\in\mathbb{Z}^{1,j}} [\varphi_l^j]'(2^{-j}p) = 0.$$

It follows then

$$\sum_{l\in\mathbb{Z}^{1,j}} B_{kl}^{1,j}(c) = \sum_{l\in\mathbb{Z}^{1,j}} \int_0^1 g\Big(\big|\sum_{m\in\mathbb{Z}^{1,j}} c_m[\varphi_m^j]'(x)\big|^2\Big)[\varphi_k^j]'(x)[\varphi_l^j]'(x)dx = 0 \tag{64}$$

and

$$\sum_{l\in\mathbb{Z}^{1,j}} \tilde{B}_{kl}^{1,j}(c) = \sum_{l\in\mathbb{Z}^{1,j}} 2^{-j} \sum_{p\in\mathbb{Z}^{1,j}} \Big(g\Big(\big|\sum_{m\in\mathbb{Z}^{1,j}} c_m[\varphi_m^j]'\big|^2\Big)[\varphi_k^j]'[\varphi_l^j]'\Big)\Big|_{2^{-j}p} = 0. \tag{65}$$

Hence, we obtain

$$\sum_{\lambda\in\Lambda_j} \big(Ac'(t)\big)_\lambda = -\sum_{\lambda\in\Lambda_j} (B(c)c)_\lambda = -\sum_{\lambda\in\Lambda_j}\sum_{\mu\in\Lambda_j} B_{\lambda\mu}(c)c_\mu = -\sum_{\mu\in\Lambda_j}(\sum_{\lambda\in\Lambda_j} B_{\lambda\mu})c_\mu = 0. \tag{66}$$

It follows that

$$\sum_{\lambda\in\Lambda_j} \big(Ac\big)_\lambda = const. \tag{67}$$

Since all row sums of A are equal, we may conclude

$$const = \sum_{\lambda\in\Lambda_j} \big(Ac\big)_\lambda = \sum_{\lambda\in\Lambda_j}\sum_{\mu\in\Lambda_j} A_{\lambda\mu}c_\mu = \sum_{\mu\in\Lambda_j}(\sum_{\lambda\in\Lambda_j} A_{\lambda\mu})c_\mu = (\sum_{\lambda\in\Lambda_j} A_{\lambda\mu})\sum_{\mu\in\Lambda_j} c_\mu = C\sum_{\mu\in\Lambda_j} c_\mu, \tag{68}$$

i.e. the average gray value remains constant in time. □

Finally, we consider the time discretization and the resulting Newton scheme. To this end, we have to choose some time step $\tau > 0$. As illustrated before, our discretization in time results then in an implicit Euler scheme. Let c^n denote the n-th iterate of the coefficient vector corresponding to the time step $t_n = n\tau$, $n = 1, \dots, n_{max} = T/\tau$. Then the scheme to be solved reads as

$$\begin{aligned} A^{1,j}(c^n - c^{n-1}) + t_k\tilde{B}^{1,j}(c^n)c^n &= 0, \quad n = 1, \dots, n_{max} & (69)\\ c^0 &= c_0. & (70)\end{aligned}$$

The explicit structure of the derivative of $\tilde{B}^{1,j}(c)c$ required for the application of Newton's scheme is given by

$$D(\tilde{B}^{1,j}(c)c)_{kl} = \frac{\partial\big(\tilde{B}^{1,j}(c)c\big)_k}{\partial c_l} = \begin{cases} 2^{2j}\big(\tilde{G}'(c_l - c_{l-1}) + \tilde{G}'(c_{l+1} - c_l)\big), & l = k \\ -2^{2j}\big(\tilde{G}'(c_l - c_{l-1})\big), & l = k-1 \\ -2^{2j}\big(\tilde{G}'(c_{l+1} - c_l)\big), & l = k+1 \\ 0, & \text{else} . \end{cases} \tag{71}$$

Since the whole scheme is solved in the nodal generator basis it might happen that the final Newton system to be solved is ill–conditioned. In this case we suggest to apply wavelet preconditioning techniques in order to stabilize the problem.

4.2 Two–dimensional case

The two–dimensional case is treated exactly the same way as the one–dimensional one. The only difference is that we are not able to generate by the integrating/differentiating method a fully interpolating generator function.

We generate the nodal basis function by tensor products of the one-dimensional scaling functions

$$[\Phi](x, y) = [\varphi](x)[\varphi](y), \quad (x, y) \in [0, 1]^2. \tag{72}$$

The components of the gradient applying Ansatz (55) are given by

$$\frac{\partial}{\partial x}[\Phi^j_k](x, y) = \frac{\mathrm{d}}{\mathrm{d}x}[\varphi^j_{k_1}](x)[\varphi^j_{k_2}](y) = 2^j\big([\phi^j_{k_1-1}](x) - [\phi^j_{k_1}](x)\big)[\varphi^j_{k_2}](y) \tag{73}$$

$$\frac{\partial}{\partial y}[\Phi^j_k](x, y) = [\varphi^j_{k_1}](x)\frac{\mathrm{d}}{\mathrm{d}y}[\varphi^j_{k_2}](y) = 2^j[\varphi^j_{k_1}](x)\big([\phi^j_{k_2-1}](y) - [\phi^j_{k_2}](y)\big), \tag{74}$$

where the gradient evaluated on $\{2^{-j}p, p \in \mathbb{Z}^{2,j}\}$ has the following explicit structure

$$\frac{\partial}{\partial x}[\Phi^j_k](2^{-j}p) = \big(\delta_{p_1,k_1-1} - \delta_{p_1,k_1}\big)[\varphi^j_{k_2}](2^{-j}p_2) \tag{75}$$

$$\frac{\partial}{\partial y}[\Phi^j_k](2^{-j}p) = [\varphi^j_{k_1}](2^{-j}p_1)\big(\delta_{p_2,k_2-1} - \delta_{p_2,k_2}\big). \tag{76}$$

Note that now all values of $[\varphi^j_k]$ on $\{2^{-j}p, p = 0, \dots, 2^j - 1\}$ are required.

The computation of the mass matrix $A^{2,j}$ can be reduced to the one–dimensional case

$$\begin{aligned} A^{2,j}_{kl} &:= \int_\Omega [\Phi^j_k][\Phi^j_l]d\Omega = \int_0^1\int_0^1 [\varphi^j_{k_1}](x)[\varphi^j_{k_2}](y)[\varphi^j_{l_1}](x)[\varphi^j_{l_2}](y)dxdy \\ &= \int_0^1 [\varphi^j_{k_1}](x)[\varphi^j_{l_1}](x)dx \int_0^1 [\varphi^j_{k_2}](y)[\varphi^j_{l_2}](y)dy = A^{1,j}_{k_1l_1}A^{1,j}_{k_2l_2}. \end{aligned} \tag{77}$$

For $B^{2,j}$ as well as $DB^{2,j}$ we use expressions (75)-(76) and the general properties of $B^{d,j}$ and $DB^{d,j}$. Due to the semi–interpolating property of the gradient the summation over $\mathbb{Z}^{2,j}$ in

$$B_{kl}^{2,j}(c) \quad := \quad 2^{-2j} \sum_{p\in\mathbb{Z}^{2,j}} \Big(g\Big(\big| \sum_{m\in\mathbb{Z}^{2,j}} c_m \nabla[\Phi_m^j] \big|^2 \Big) \nabla[\Phi_k^j] \nabla[\Phi_l^j] \Big) \Big|_{2^{-j}p} \tag{78}$$

is reduced to the summation over $\mathbb{Z}^{1,j}$. The matrices $B^{2,j}$ and $DB^{2,j}$ are sparse by the compactness of φ. Their computation is therefore reduced to the computation of the non-zero entries. Since the values of the product $\nabla[\Phi_k^j]\nabla[\Phi_l^j]$ do not depend on the solution c they can be pre-computed, which results in a reduction of the complexity of the numerical scheme to be implemented.

5 Numerical experiments

This section is devoted to demonstrate the performance of the presented wavelet–based scheme to solve the Perona–Malik equation. Firstly, we show how the discretized nonlinear diffusion process acts on one–dimensional synthetic/simulated data, and secondly, on a two–dimensional 'eye'–image. In order to highlight and to compare the nonlinear smoothing effect, we compare the results obtained with several diffusivity functions (ranging from linear to nonlinear ones). For the computation of the nonlinear term we choose, as described in Subsection 2.2, the interpolating Deslauriers–Dubuc functions of order larger or equal than two, i.e. the lowest order Ansatz function is the hat function. This in turn induces our finite dimensional Ansatz space $[V_j]$ to start with.

5.1 The one–dimensional example

The data were simulated by compositing piecewise constant or smooth functions and adding some i.i.d. Gaussian noise, see e.g., the upper left image in Figure 1.

In the examples the resolution was chosen such that it was fitting to the scale $j = 9$. Moreover, we have picked the diffusivity function g as in (2). Then, one important observation is that the computational cost for one Euler iteration (one time step) requires at most two to three Newton iterations.

As we can observe in Figure 1, the scheme produces reasonable results. Nevertheless, in some regions which are initially smooth, see again Figure 1: lower left image, the scheme creates certain 'stair' irregularities. But this effect is not that unusual; it is the so–called *stair–casing* and stems from instabilities in the discretization of the Perona–Malik equation, see [9].

In Figure 2, one Euler iteration of both the nonlinear and linear scheme to different kinds of data is shown (time step $t = 0.001$). It becomes clearly visible that the nonlinear scheme preserves the edges very well, as expected.

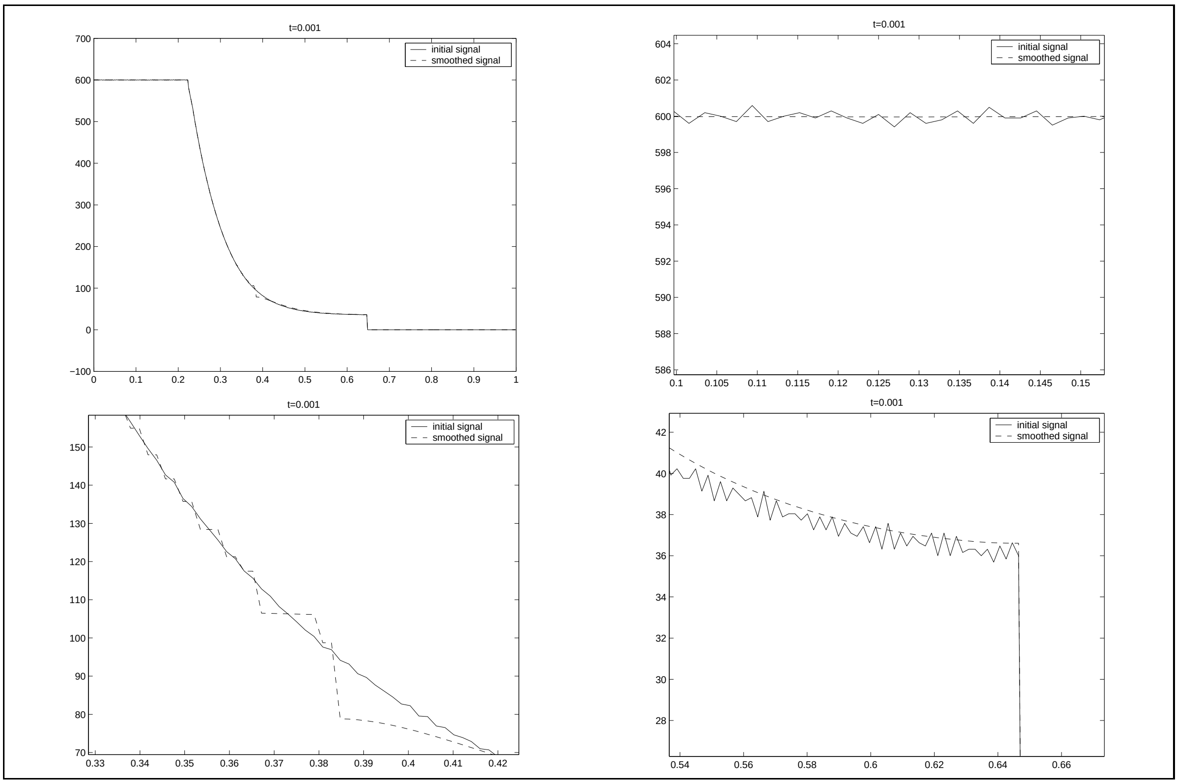

Figure 1: Wavelet–based variant of nonlinear smoothing with the Perona–Malik model after one time step $t = 0.001$. Upper left: complete initial data (solid line) and smoothed data (dotted line); Upper left, lower left and right: zoom in different subintervals.

Finally, in Figure 3 results for increasing values of t are visualized. The scheme produces stable results tending to a step function with a small number of jumps.

We summarize that the wavelet–based variant of the Perona–Malik discretization behaves like other used approaches (e.g. pixel–based implementations), and is therefore an alternative tool to efficiently solve this sort of problems.

5.2 The two–dimensional example

The initial 'eye' image is shown in Figure 4. To this image we have added i.i.d. Gaussian noise. Contrary to the previous example we have chosen here the following diffusivity function:

$$g_{\gamma,\delta}(x) = \begin{cases} 1, & x \leq \delta \\ \frac{1}{2}(\cos(\pi(x-\delta)/\gamma)+1), & \delta < x < \gamma+\delta \\ 0, & x \geq \gamma+\delta \end{cases} \tag{79}$$

where $\gamma > 0$, $\delta \geq 0$ such that $\gamma + \delta \leq 1$. The function $g_{\gamma,\delta}$ decreases from 1 to 0 inside the interval $(\delta, \delta+\gamma)$ and is properly smooth inside $(0,1)$. For $\gamma \to 0$ the function g tends to the characteristic function with jump of size one in δ, i.e. smoothing barrier is

δ. The behavior of the smoothing scheme depends significantly on these two parameters δ and γ. The underlying function (here cos) is less relevant, it has just to be smooth enough. The smoothing rate of the scheme increases with δ and γ controls the region were non–uniform smoothing appears.

Here we limit the choice of the parameters δ and γ to the point–wise maximum μ of $|\nabla u_0|^2$, where u_0 is the noised initial image. In the first example we set $\gamma = \mu/32$ and $\delta = 3\mu/64$, so that for all gradient values greater then $\sqrt{5\mu}/8$ no smoothing occurs. In the second and third examples we set $\gamma = \mu/5$ and $\delta = 0$ and $\delta = 2\mu/5$ respectively.

The corresponding diffusivity functions $g(s^2)$, where s is the absolute value of the gradient, are shown in the Figure 5.

In Figure 6, we have marked the pixels which were not affected through the smoothing process (for one time step $t = 0.1$), and visualized the results of the denoising corresponding to the three diffusivities.

Finally, in the Figure 7 we compare nonlinear with linear smoothing. For the nonlinear smoothing we have again used the diffusivity $g_{\mu/32,3\mu/64}$, whereas $g(s^2) = 1$ in the linear case.

As in the previous example we summarize that also in the two–dimensional case the wavelet–based variant of Perona–Malik regularization results in a thrifty algorithm and yields good results.

References

[1] Hans Wilhelm Alt. *Lineare Funktionalanalysis, 4. Auflage.* Springer, Berlin, 2002.

[2] Felix E. Browder. Nonlinear maximal monotone operators in Banach space. *Math. Annalen*, 175:89–113, 1968.

[3] S. Dahlke, P. Maass, and G. Teschke. Interpolating scaling functions with duals. *Journal of Computational Analysis and Applications*, 5(3):361–373, 2003.

[4] W. Dahmen, A. Kunoth, and K. Urban. A wavelet-Galerkin method for the Stokes equations. Technical Report ISC-95-05-MATH, Institute for Scientific Computation, Texas A&M University, August 1995.

[5] W. Dahmen and C.A. Micchelli. Using the refinement equation for evaluating integrals of wavelets. *SIAM J. Numer. Anal.*, 30:507–537, April 1993.

[6] W. Dahmen, S. Prößdorf, and R. Schneider. Multiscale methods for pseudodifferential equations on smooth manifolds. In *C.K. Chui, L. Montefusco, and L. Puccio, editors, Proceedings of the International Conference on Wavelets: Theory, Algorithms, and Applications*, pages 385–424, 1995.

[7] P. Deuflhard and A. Hohmann. *Numerische Mathematik I.* Gruyter, Stuttgart, 1993.

[8] David L. Donoho. Interpolating wavelet transforms. Technical report, Department of Statistics, Stanford University, 1992.

[9] S. Kichenassamy. The Perona-Malik paradox. *SIAM Journal of Applied Mathematics*, 57(5):1328–1342, 1997.

[10] P.G. Lemarié-Rieusset. Analyses multi-résolutions non orthogonales, commutation entre projecteurs et dérivations et ondelettes vecteurs à divergence nulle. *Revista Mat. Iberoamer*, 8:221–237, 1992.

[11] Mathias Lindemann. Wavelet Expansions and Besov Spaces, 2004. PhD Thesis.

[12] S. Mallat. *A Wavelet Tour of Signal Processing*. Academic Press, San Diego, 1998.

[13] P. Perona and J. Malik. Scale space and edge detection using anisotropic diffusion. *IEEE Trans. Pattern Anal. Mach. Intell.*, 12:629–639, 1990.

[14] H. R. Schwarz. *Numerische Mathematik*. Teubner, Stuttgart, 1997.

[15] J. Stoer. *Numerische Mathematik II*. Springer, Berlin, 2000.

[16] Karsten Urban. On divergence-free wavelets. *Advances in Computational Mathematics*, 4(1,2):51–82, 1995.

[17] W. Walter. *Differential and Integral Inequalities*. Springer, Berlin, 1970.

[18] W. Walter. *Gewöhnliche Differentialgleichungen*. Springer, Berlin, 1976.

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

[20] J. Weickert and B. Benhamouda. A semidiscrete nonlinear scale-space theory an its relation to the Perona-Malik paradox. In *Advances in Computer Vision*, pages 1–10. Springer, 1997.

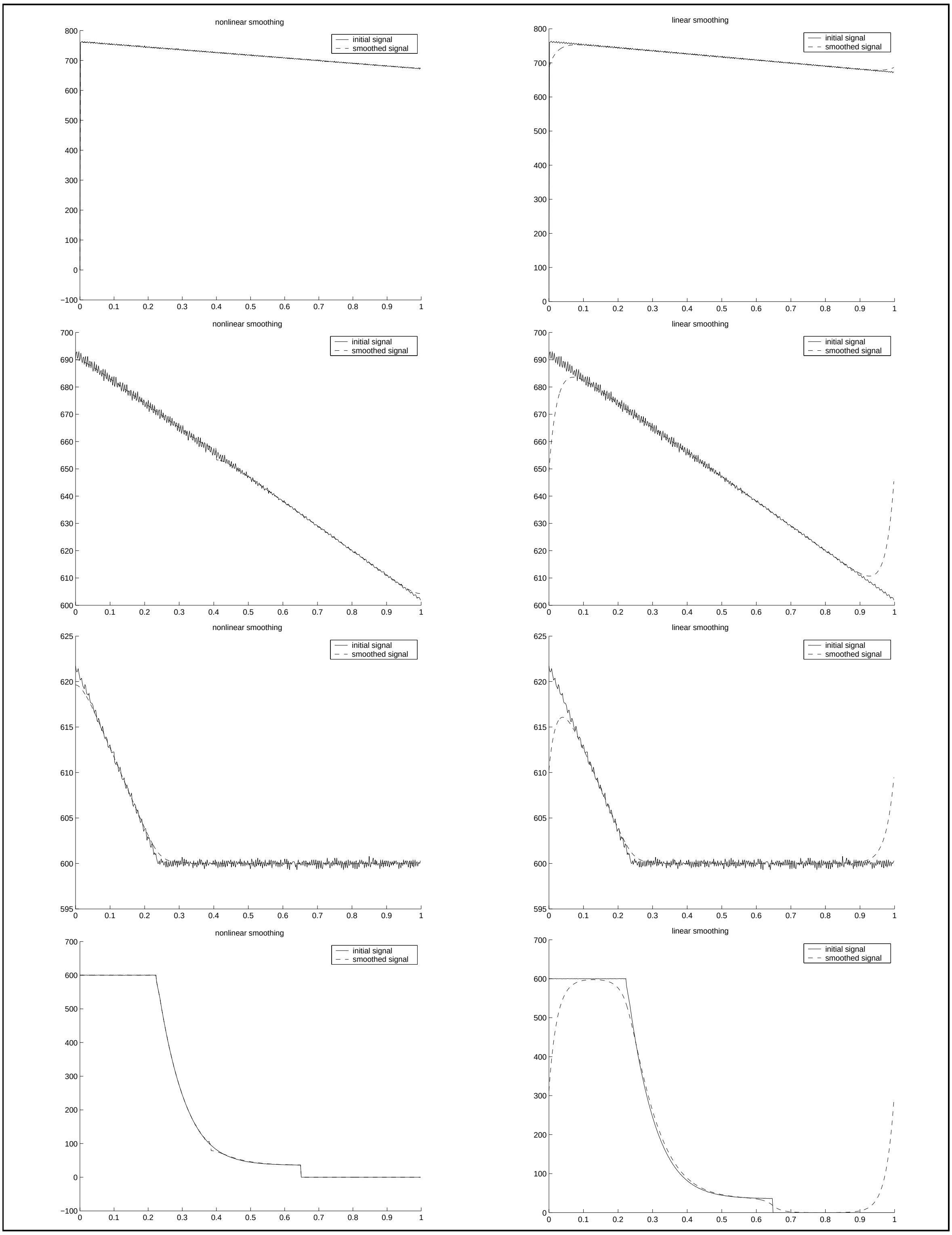

Figure 2: Wavelet–based variant of nonlinear (left column) and linear (right column) smoothing with the Perona–Malik model after one time step $t = 0.001$ for different kinds of initial data.

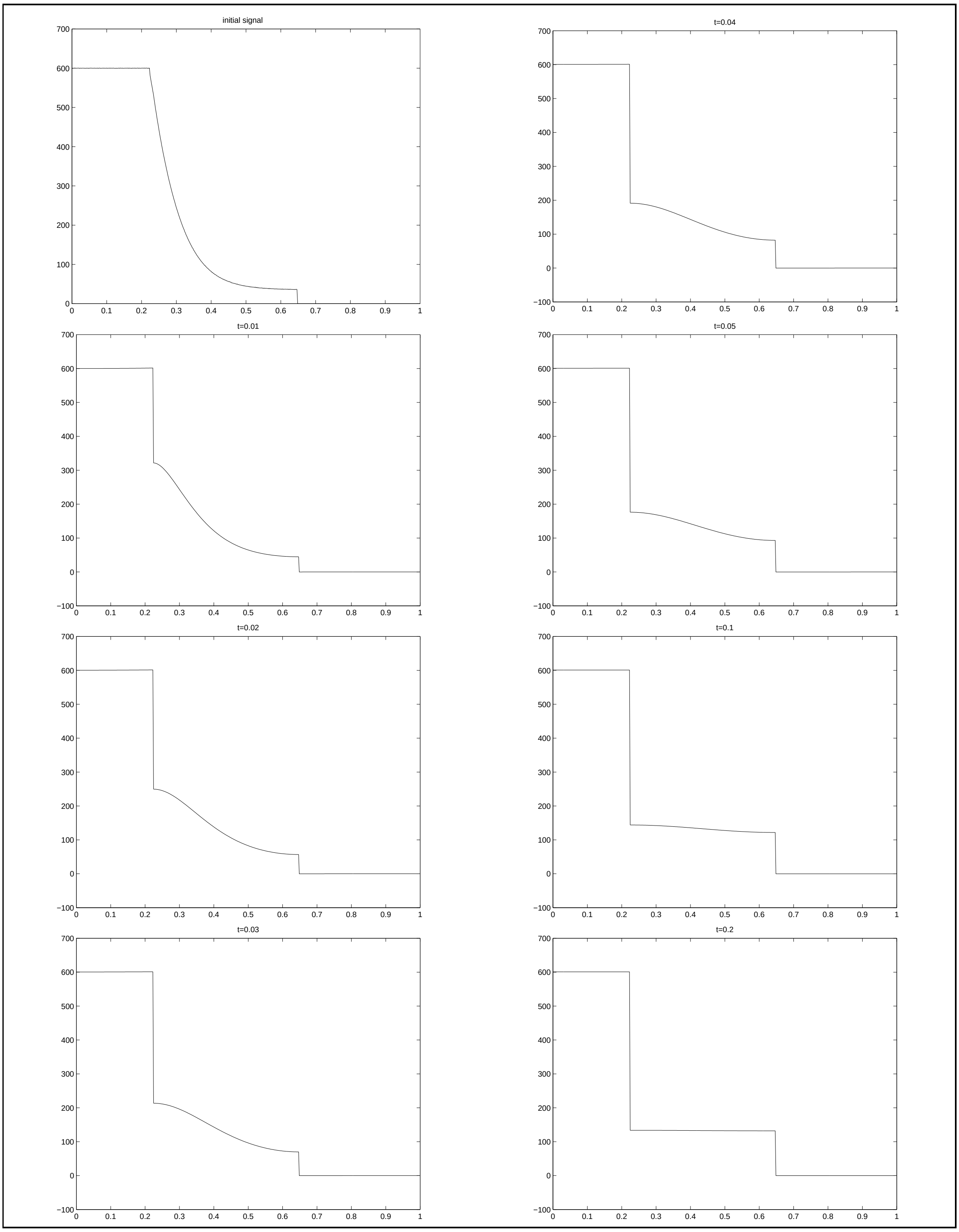

Figure 3: Wavelet–based variant of nonlinear smoothing with the Perona–Malik model for increasing values of t.

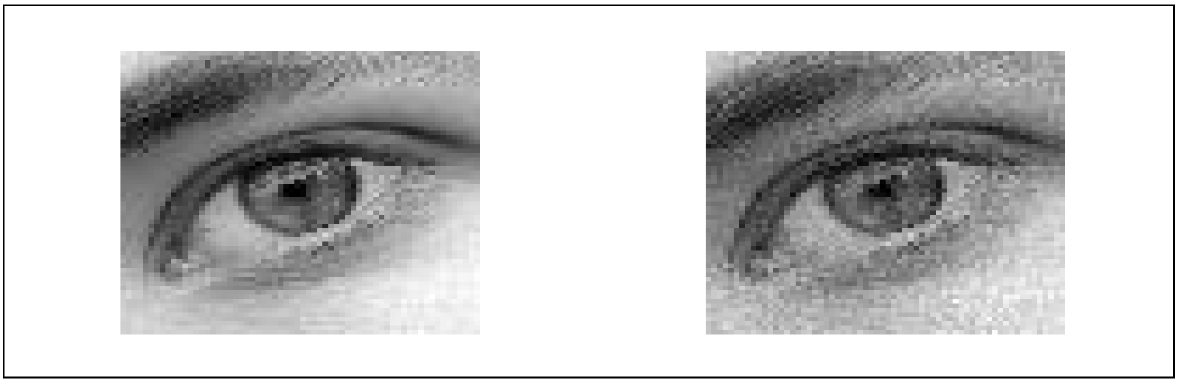

Figure 4: Left: initial image, right: noisy image.

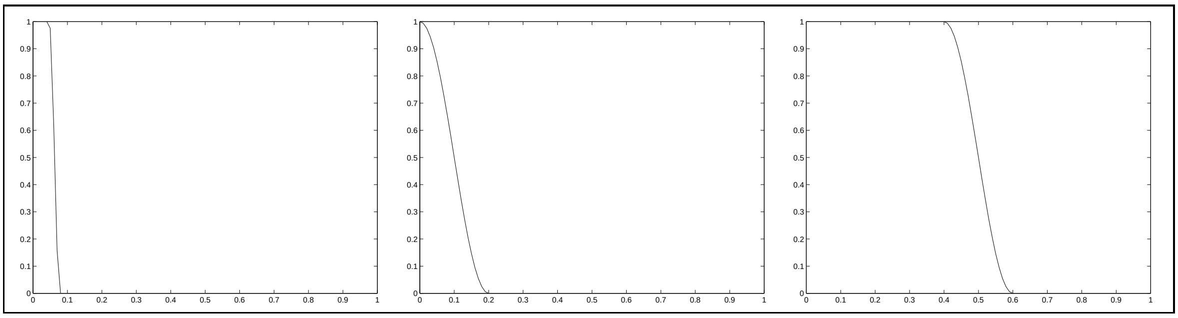

Figure 5: Diffusivity functions with different values of γ and δ.

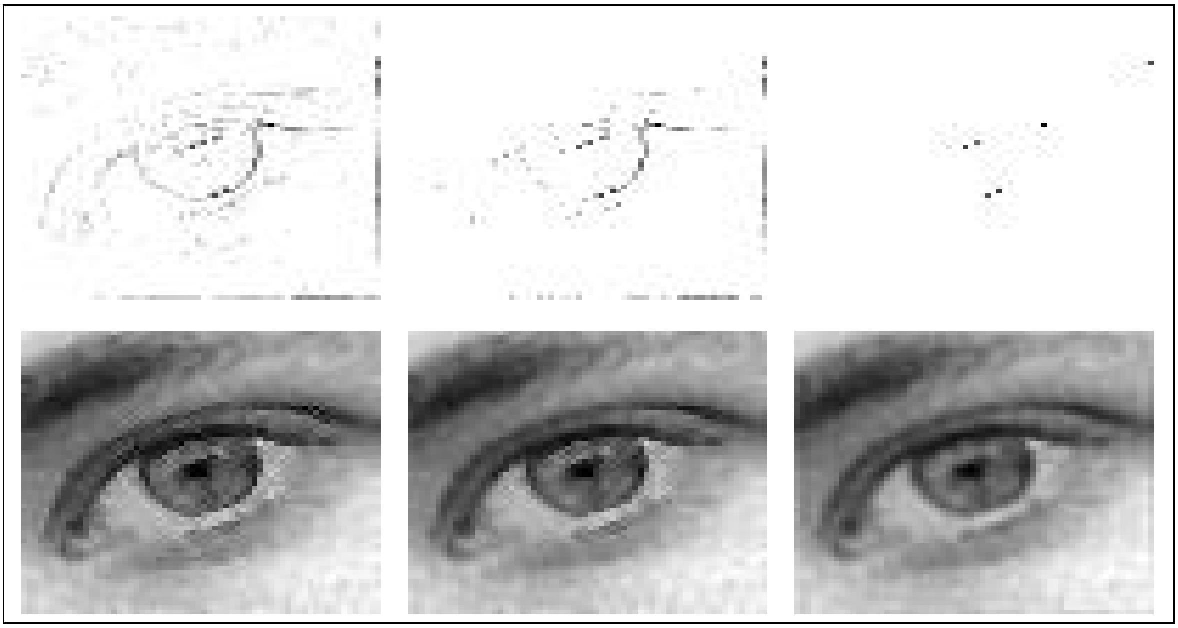

Figure 6: Upper row: regions (pixels) where no smoothing appears; Lower row: nonlinear smoothing, one time step $t = 0.1$. From left to right in accordance with the three different diffusivity functions.

Figure 7: Evolution in time of nonlinear (left) and linear smoothing (right).