A Regularization of Nonlinear Diffusion Equations in a Multiresolution Framework

Gerd Teschke*, Mariya Zhariy*, Joana Soares†

May 4, 2007

Abstract: *We are developing a regularization technique for Perona–Malik diffusion equations that relies on multiresolution techniques. The main result of this paper is to show that the chosen discretization overcomes the ill-posedness of the nonlinear Perona–Malik model. The resulting algorithm is tested and the results are compared with pixel–based methods.*

keywords, phrases: Nonlinear diffusion, regularization, operator adapted refinable functions

1 Introduction

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

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 prominent image smoothing model was introduced by Perona and Malik, see [14], which results in a nonlinear diffusion equation

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

where g is supposed to be a non–negative, smooth, non–increasing function with $g(0) = 1$ and g tending to zero at infinity. Typically, the function g should be chosen such that the diffusion process described by (1) behaves like a linear diffusion for small gradients and does nothing for large gradients. In [14], it is suggested to choose $g(x) = (1 + x/\lambda^2)^{-1}$. When now considering the flow function $\Phi(x) = g(x^2)x$, we have a change of the sign of Φ' when $|x| > \lambda$. This, unfortunately, induces ill-posedness of (1).

There exists now several ways to circumvent the ill–posedness. The very first way is to choose diffusivities g for which (1) remains well–posed, see e.g. [19]. But in principle, such functions will have a very slow decay what implies undesired filtering properties such as edge smoothing. Thus, the second way is

*The first two authors are with Konrad Zuse Zentrum für Informationstechnik Berlin, Takustr.7, D-14195 Berlin, Germany

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

to overcome ill-posedness by a reasonable regularization of (1). This, moreover, allows a more unrestricted choice of smoothing properties of (1).

We shall follow the second way and develop a regularization of (1) which is based on multi–resolution discretization. The involved techniques are in principle similar to Weickert's finite-differences methods , see [10, 19, 20]: discretize spatially problem (1) and obtain a system of ordinary differential equations $du/dt = A(u)u$ that can be solved by simple means. It is shown in [20], when using a finite difference discretization, that under certain conditions on the matrix $A(u)$ ($A(u) \in C^1$, symmetry, non-negative off diagonals, irreducibility) the system of ODE's exhibits well-posedness, continuous dependency on the initial values and on the right hand side, mean-value conservation, extremum principle, and, finally, convergence to the stationary constant state.

As a result, we show that the system of ODE's obtained when discretizing with multi–resolution generator functions yields a *well-posed* problem. Thus, this method represents an alternative method in regularizing nonlinear diffusion equations. But beside the theoretical result (on which is the focus in this paper), the scheme that comes out has a comparable numerical complexity as pixel–based Perona–Malik implementations.

The organization of the paper is as follows: in Section 2 we summarize all mathematical ingredients; in Section 3 we present our multi–resolution–based discretization and prove the main result; Section 4 is concerned with the concrete applications/experiments.

2 Technical Building Blocks

In this section, we declare the setting and the necessary means that are needed for our purposes.

Let $\Omega = [0,1]^2$ be the two dimensional domain of our image u. Then, for $T > 0$, 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 $t \in (0,T)$. This can be relaxed when writing (1) in its variational (weak) formulation. Then it is enough to require $u \in C^1(0,T;H^1(\Omega))$. Note that here we do not focus on applying the classical Galerkin idea to approximate the solution of the continuous problem since we do not have existence of a continuous solution in general, see [10]. Instead, we choose just one particular subspace $V_j \subset H^1(\Omega)$ for some fixed j and aim to solve the related discretized problem. The choice of the subspace V_j is of significant importance when aiming at regularization and thrifty algorithm.

Since we focus on a periodic framework, we aim at ansatz spaces V_j on the torus (instead on Ω itself). When discretizing (1), it would be also desirable to have an ansatz system (spanning the subspace V_j) that is nicely adapted to the application of the ∇–operator and, moreover, that ensures a simple way to compute the remaining inner products and integral terms.

One natural choice is an ansatz space V_j stemming from a multiresolution analysis, see for conceptual details [6]. Since this concept in its initial shape meets not all our specific requirements, we have to adjust it. The individual building blocks of the proposed adjustment can only be found in the widespread literure. Thus, for sake of coherence and to see them interacting we briefly sketch them here.

Multiresolution analysis on the torus. A detailed introduction can be found in [5, 6]. We limit ourselves to sketching the essential facts that are required for our purpose. As mentioned above, let us consider instead the 2-dimensional unit square Ω the torus $T^2 := \mathbb{R}^2/\mathbb{Z}^2$. For this purpose we have to apply a periodization map from $L_2(\mathbb{R}^2)$ to $L_2(T^2)$ which is given by $[f](x) \mapsto \sum_{l\in\mathbb{Z}^d} f(x+l)$. Let now $M \in \mathbb{Z}^{2\times 2}$ be a given integer matrix where its eigenvalues have modulus larger than one, and let $h = (h_k)_{k\in\mathbb{Z}^2} \in \ell_2(\mathbb{Z}^2)$. Then a function $\phi \in L_2(\mathbb{R}^2)$ is called (M, h)-refinable, if for all $x \in \mathbb{R}^2$

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

with mask $h = \{h_k\}_{k\in\mathbb{Z}^d}$ and scaling matrix M, where $m = |\det M|$. Set $\phi_k^j := m^{j/2}\phi(M^j \cdot - k)$ for $j \in \mathbb{Z}$ and $k \in \mathbb{Z}^2$. Then a refinable function $\phi \in L_2(\mathbb{R}^2)$ with $\int \phi dx \neq 0$ and with ℓ_2-stable system $\{\phi_k^0,\ k \in \mathbb{Z}^2\}$ generates a multiresolution analysis of $L_2(\mathbb{R}^2)$. Such a refinable function is often called a scaling function. The introduced periodization map can now be applied to ϕ in order to construct a multiresolution analysis for $L_2(T^2)$,

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

It can be easily seen that the rescaling property (2) holds also true for the periodized version of ϕ, i.e. we have $[\phi_0^0](x) = \sum_{k\in\mathbb{Z}^2} h_k[\phi_k^1](x)$. Introducing now $\mathbb{Z}^{2,j} := \mathbb{Z}^2/(M^j\mathbb{Z}^2)$, one can define the spaces

$$[V_j] := \overline{\mathrm{span}\{[\phi_k^j], k \in \mathbb{Z}^{2,j}\}}$$

that fullfil all properties of a multiresolution analysis of $L_2(T^d)$, see for details [5]. These spaces will be the basis for our further elaborations.

Interpolating scaling functions and quadrature rule. Solving the variational formulation requires at least the evaluation of the corresponding stiffness matrix and the integral term originating from the nonlinear part. As long as the entries of the stiffness matrix are refinable functions, it is shown that there is an exact way of computing the entries, see [4]. The remaining integral terms need to be approximated. The hope is to construct an easy to implement quadrature rule. Starting point are interpolating scaling functions. An interpolating scaling function fulfills $\phi(k) = \delta_{0,k}$ for all $k \in \mathbb{Z}^d$. A very simple construction scheme is given, in the one dimensional case, by the so–called Deslauriers-Dubuc scaling functions, see e.g. [7, 9],

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

where $*$ stands for convolution, $\check{g}(\cdot) = g(-\cdot)$, and φ^N denotes here a compactly supported Daubechies scaling functions of order N, see e.g. [8, 11, 13]. These Deslauriers-Dubuc scaling functions are again compactly supported and generate a multiresolution analysis with easy to derive duals, see [2]. In order to construct a quadrature rule, we introduce the following interpolation projector

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

for we define a simple quadrature rule for $\int_\Omega v dx$ via

$$Q\Big(\int_\Omega v dx\Big) := \int_\Omega \pi_j(v) dx = m^{-j} \sum_{k\in\mathbb{Z}^{2,j}} v(M^{-j}k).$$

Proposition 2.1. *Let π_j be the interpolation operator associated with a continuous, compactly supported (M,h)-refinable function ϕ belonging to the Sobolev space $W^n(L_1(\mathbb{R}^d))$ for some $n \in \mathbb{N}$ and for which $\{\phi_k^0, k \in \mathbb{Z}^2\}$ is l_2-stable. Then there exists $C > 0$, such that for all $v \in W^{n+1}(L_\infty(T^d))$ the error estimate*

$$|\int_\Omega v(x)dx - \int_\Omega \pi_j(v)dx| \leq Cr(M)^{-j(n+1)},$$

where $r(M)$ denotes the spectral radius of M.

For a proof of the latter proposition we refer to [12], whereas a detailed discussion on Sobolev spaces can be found, e.g., in [21].

Scaling functions via integration. Finally, the scaling functions should fit with the application of the ∇–operator. This property can be achieved when constructing the scaling functions via integration and differentiation. For a precise description of the construction principles we refer to [3, 11, 17].

The spine of this approach is to start with a dual pair of scaling functions and to apply integration and differentiation. This yields another dual pair of refinable functions that are now equipped with sort of 'derivative absorbing property'. In univariate case the construction process is simple and explained in the following theorem for which a proof can be found in [11]. The bivariate case follows immediately when limiting the construction to the separable situation.

Proposition 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, that generate a dual multiresolution analysis for $L_2(\mathbb{R})$. Moreover, let $\tilde{\phi} \in H^1(\mathbb{R})$ and $\int \phi = \int \tilde{\phi} = 1$. Then there exists a dual pair $\varphi, \tilde{\varphi} \in L_2(\mathbb{R})$ of compactly supported scaling functions with $\varphi'(x) = \phi(x+1) - \phi(x)$ and $\tilde{\phi}'(x) = \tilde{\varphi}(x) - \tilde{\varphi}(x-1)$. The symbols $\mathbf{h}$ of φ and $\tilde{\mathbf{h}}$ of $\tilde{\varphi}$ satisfy $2\mathbf{h}(z) = (1+z)\mathbf{H}(z)$ and $(1+z)\tilde{\mathbf{h}}(z) = 2z\tilde{\mathbf{H}}(z)$. The functions φ and $\tilde{\varphi}$ generate a dual multiresolution analysis of $L_2(\mathbb{R})$ too.*

We summarize our findings. When selecting φ in accordance with Proposition 2.2 to be our primal scaling, we observe that the application of ∇ yields just differences of ϕ. Once this ϕ is interpolating, we may use our easy quadrature rule, and, moreover, since we are able to derive $\mathbf{h}$, the computation of the stiffness matrix can be done exactly.

3 Regularization

Assume, we have chosen the ansatz system to spatially discretize (1) in accordance with the last section. The goal is now to show whether the variational Perona–Malik equation is well–posed.

Semi-discrete formulation. Suppose u, u_0 are sufficiently smooth such that the boundary value problem $\partial_t u = \mathrm{div}(g(|\nabla u|^2)\nabla u)$ on $(0,T) \times \Omega$, $u_{|_{t=0}} = u_0$ in Ω , and u spatially periodic in $(0,T)$ is well-defined.

Let $H^1_p(\Omega)$ denote the Sobolev space of order one with periodic boundary conditions. Choose then $[V_j] \subset H^1_p(\Omega)$ as the underlying Ansatz space. Since the existence of boundary values of Sobolev functions follows from the Trace Theorem for Sobolev functions, see [1], there exists an $L_2(\partial\Omega)$–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 problem is to find some $u^j \in C^1(0,T;[V_j])$ such that

$$\big(\partial_t u^j(t), v\big) + \Big(g(|\nabla u^j(t)|^2)\nabla u^j(t), \nabla v\Big) = 0 \qquad \text{and} \quad u^j(0) = u^j_0 \tag{3}$$

is fulfilled for all $v \in [V_j]$ and $t \in (0,T)$. We now rewrite problem (3) by means of the ansatz system. Let $\{\phi_\lambda, \lambda \in \Lambda_j\}$ with $\Lambda_j := \{\lambda = (j,k), k \in \mathbb{Z}^{2,j}\}$ be a basis of $[V_j]$ and let N_j be its dimension, i.e. $N_j = \#\Lambda_j$. Thus, when assuming $u^j \in C^1(0,T;[V_j])$, then u_j has the unique expansion with coefficients $c_\lambda \in C^1(0,T)$. Furthermore, let $A \in \mathbb{R}^{N_j \times N_j}$ denote the stiffness matrix associated with the nodal basis of $[V_j]$, i.e. $A_{\lambda\mu} := \big(\phi_\lambda, \phi_\mu\big)$ with $\lambda, \mu \in \Lambda_j$. Then problem (3) can be rewritten as

$$Ac'(t) + B(c(t))c(t) = 0 \text{ with } c(0) = c_0, \tag{4}$$

where

$$B_{\lambda\mu}(c) = \Big(g\big(\big|\sum_{\nu\in\Lambda_j} c_\nu \nabla\phi_\nu\big|^2\big)\nabla\phi_\lambda, \nabla\phi_\mu\Big).$$

Well–posedness of (3). Now we state our first result. For the proof we refer to the Appendix.

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

$$G(c)(x) := g\Big(\big|\sum_{\nu\in\Lambda_j} c_\nu \nabla\phi_\nu(x)\big|^2\Big)c \tag{5}$$

be Lipschitz continuous with respect to c with Lipschitz constant $L > 0$. Then there exists a unique vector c such that $u^j = \sum_{\lambda\in\Lambda_j} c_\lambda\phi_\lambda$ is the unique solution of problem (3).

Conservation of well–posedness. The question arises whether Theorem 3.1 holds true when numerically solving problem (4). At first, thanks to [4] we may exactly compute the entries of A in (4). The remaining integral term B in (4) will be approximated with the quadrature rule introduced in the previous section, i.e. we obtain the following approximation $\tilde{B}$ for B

$$\begin{aligned}\tilde{B}_{\lambda\mu}(c) &= Q\big(B_{\lambda\mu}(c)\big)\\ &= \frac{1}{m^j}\sum_k \Big(g\big(\big|\sum_{\nu\in\Lambda_j} c_\nu \nabla\phi_\nu\big|^2\big)\nabla\phi_\lambda\nabla\phi_\mu\Big)\Big|_{M^{-j}k}\end{aligned}$$

where $\lambda, \mu \in \Lambda_j$ and $c \in \mathbb{R}^{N_j}$. Consequently, in practice we just solve the perturbed system

$$Ac'(t) + \tilde{B}(c(t))c(t) = 0 \text{ and } c(0) = c_0, \tag{6}$$

i.e. we have to ensure that system (6) remains solvable and that its solution approximates the solution of (3). This can be shown by a perturbation argument. Let us reformulate the problem: for all $t \in (0, T]$, $c(t)$ is a solution of

$$c'(t) = F(t, c(t)) \text{ and } c(0) = c_0, \tag{7}$$

where $F(t, c) := -A^{-1}B(c)c$. We need to show that the solution depends continuously on the right hand side F. By [18], this means we have to show that $F(t, c)$ is continuous and that there exists some $\alpha > 0$ such that F is with respect to c global Lipschitz continuous on $D_\alpha := \{(t, d) | \; t \in [0, T], \|d - c\| \leq \alpha\}$. Then it follows that for each $\varepsilon > 0$ there exists some $\delta > 0$ such that the solution $\tilde{c}(t)$ of the perturbed problem

$$c'(t) = \tilde{F}(t, c(t)) \quad \text{and} \quad c(0) = c_0$$

with continuous $\tilde{F}$ satisfying

$$\|\tilde{F}(t, d) - F(t, d)\| < \delta \tag{8}$$

for $\|d - c\| < \alpha$ 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.

In the next theorem we state that when applying our quadrature rule condition (8) holds true (for the proof we refer to the Appendix).

Theorem 3.2. *Assume all the conditions required in Proposition* 2.1. *Moreover, assume A to be regular and that B and $\tilde{B}$ fulfill a Lipschitz condition. Then, there exists some δ such that $\left\|\tilde{F}(t, d) - F(t, d)\right\| < \delta$.*

We conclude that with the help of the latter theorem, the solution of the perturbed system is very close to the one of the exact system.

Temporal discretization. It remains to approximate the time derivative. Here we apply the implicit Euler scheme which reads in our case as follows: for each time step $t_n = n\tau$ we reconstruct some sequence c_n by solving

$$\begin{aligned} \mathcal{F}(c^n) := A(c^n - c^{n-1}) + \tau\tilde{B}(c^n)c^n &= 0 \\ c^0 &= c_0. \end{aligned} \tag{9}$$

Since the implicit Euler method is absolutely stable for all increments $\tau > 0$, one has that the global temporal discretization error is bounded, see e.g. [15, 16]. System (9) will be solved with Newton's method. The Newton scheme requires the numerical evaluation of the derivative

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

4 Numerical illustrations

To find a solution of the regularized Perona–Malik equation (1), we finally have to solve several time steps of system (9).

In order to verify the applicability of the proposed scheme, we compare it with several different methods. But before giving the illustrations we briefly sketch how the several terms of our system can be determined. Let us consider at first the univariate situation.

Let φ be a scaling function as prescribed in Proposition 2.2, i.e. $\varphi'(x) = \phi(x+1) - \phi(x)$ with ϕ satisfying the interpolating property. Moreover, let $[\varphi]$ its periodized version. Then we have $[\phi_k^j](2^{-j}p) = 2^{j/2}\delta_{k,p}^j$ and $\frac{d}{dx}[\varphi_k^j](2^{-j}p) = 2^{3j/2}(\delta_{k-1,p}^j - \delta_{k,p}^j)$, where $\delta_{k,p}^j = 1$ for $p = k$ and zero otherwise. Next, the spatial discretization requires the computation of A and $\tilde{B}$. To specify the dimension and the scale we write $A^{1,j}$ and $\tilde{B}^{1,j}$ instead of A and $\tilde{B}$. By periodicity, the stiffness matrix entries are

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

showing that the computation can be reduced to the non–periodic case and can be realized by the algorithm in [4] which ensures an exact computation of the inner products. For the nonlinear term we finally obtain

$$\big(\tilde{B}^{1,j}(c)c\big)_k \quad = \quad 2^{2j}\big(\tilde{G}(c_k - c_{k-1}) - \tilde{G}(c_{k+1} - c_k)\big),$$

where $\tilde{G}(x) := g(2^{3j}x^2)x$. At this point we remark that both the exact and perturbed system ensure gray value conservation (see Appendix for a proof). Finally, the time discretization amounts to an implicit Euler scheme,

$$A^{1,j}(c^n - c^{n-1}) + t_k \tilde{B}^{1,j}(c^n)c^n = 0 \quad \text{and} \quad c^0 = c_0,$$

which will be solved by Newtons method. This requires the explicit structure of the derivative of $\tilde{B}^{1,j}(c)c$,

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

Let us now consider the bivariate case: we generate the scaling functions by tensor products of the univariate scaling functions. Thus, the components of the gradient have the following explicit structure

$$\frac{\partial}{\partial x}[\Phi_k^j](2^{-j}p) = \big(\delta_{p_1,k_1-1} - \delta_{p_1,k_1}\big)[\varphi_{k_2}^j](2^{-j}p_2) \ ,$$

$$\frac{\partial}{\partial y}[\Phi_k^j](2^{-j}p) = [\varphi_{k_1}^j](2^{-j}p_1)\big(\delta_{p_2,k_2-1} - \delta_{p_2,k_2}\big).$$

Since $A_{kl}^{2,j} = A_{k_1 l_1}^{1,j} A_{k_2 l_2}^{1,j}$, the computation of the $A^{2,j}$ can be reduced to the univariate case. Due to the semi–interpolating property of the gradient, the summation over $\mathbb{Z}^{2,j}$ in

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

Figure 1: Left: original image; right: noisy image.

is reduced to the summation over $\mathbb{Z}^{1,j}$. The matrices $B^{2,j}$ and $DB^{2,j}$ are sparse by the compact support 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 computational cost.

Let us now consider a concrete two–dimensional example, see Figure 1 for the original noiseless 256×256 *eye*–image and its noisy version. In order to compare the capabilities of the proposed scheme, we compare it with the pixel–based implementation[1] for solving the nonlinear Perona–Malik diffusion equation. The setup for the method is now as follows: since the dimension is 256×256, the resulting resolution level is $j = 7$ (when assuming that the domain of the image is $\Omega = [0, 1]^2$). As the dual interpolating scaling function, we start with a linear B-spline of order two, where the primal filter mask is derived via Proposition 2.2. The scaling matrix M has in this separable setup diagonal structure with diagonal $(2, 2)$. Increasing the order of the interpolating scaling functions achieves a better approximation quality but also increases the computational cost (since the stiffness matrix becomes non–sparse).

For both schemes the pixel–based and our proposed multiscale implementation 40 iteration are computed with time step size 0.24. For both we have used the same diffusivity g (see above) with $\lambda = 0.005$. The approximation quality is measured by the signal-to-noise ratio (SNR), which is here defined by $SNR = 10 \log_{10}(\|f\|^2/\|f - u\|^2)$ where f is the noiseless image. see Figure 2. We finally observe that both schemes are able to achieve the same SNR within a similar number of iterations: multiscale method after 16 iterations and the pixel–based approach after 30 iterations) regime, see Figures 2 and 3.

5 Appendix

Proof of Theorem 3.1.
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 (3) results in an equiva-

[1]Matlab code taken from http: / / staff.science.uva.nl / ∼rein / nldiffusionweb / material.html

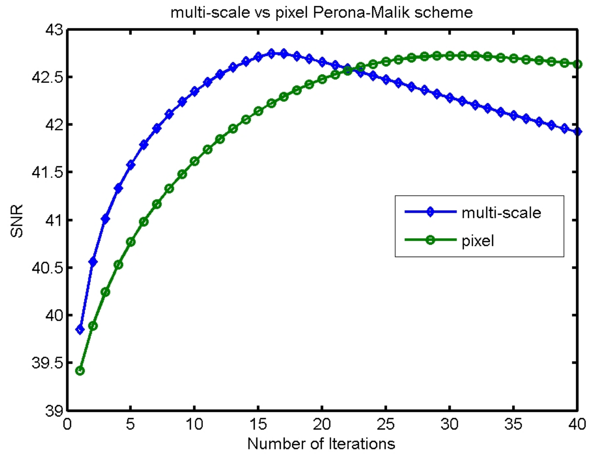

Figure 2: Signal–to–noise ratios per iteration.

lent ODE system:

$$\sum_{\lambda\in\Lambda_j} c_\lambda'(t)\big(\phi_\lambda,\phi_\mu\big)+$$
$$\sum_{\lambda\in\Lambda_j} c_\lambda(t)\Big(g\big(\big|\sum_{\nu\in\Lambda_j} c_\nu(t)\nabla\phi_\nu\big|^2\big)\nabla\phi_\lambda,\nabla\phi_\mu\Big)=0$$
$$c_\lambda(0)=c_{0,\lambda},$$

which can be rewritten as

$$Ac'(t)+B(c(t))c(t) \quad = \quad 0 \text{ with } c(0)=c_0, \tag{10}$$

where $c_{0,\lambda}=(u_0,\tilde{\phi}_\lambda)$ and B represents the nonlinear term,

$$B_{\lambda\mu}(c)=\Big(g\big(\big|\sum_{\nu\in\Lambda_j} c_\nu\nabla\phi_\nu\big|^2\big)\nabla\phi_\lambda,\nabla\phi_\mu\Big).$$

The regularity of A allows to transform the initial value problem (10) into an integral representation

$$c(t)=c_0-\int_0^t A^{-1}B(c(s))c(s)ds. \tag{11}$$

Note that $c\in C^1(0,T)^{N_j}$ solves the initial value problem (10) if and only if $c\in C^0([0,T])^{N_j}$ and c solves the integral equation (11). 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 (11) has to be a fixed point for $S:X\to X$ where S is defined as

$$S(c)(t):=c_0-\int_0^t A^{-1}B(c(s))c(s)ds. \tag{12}$$

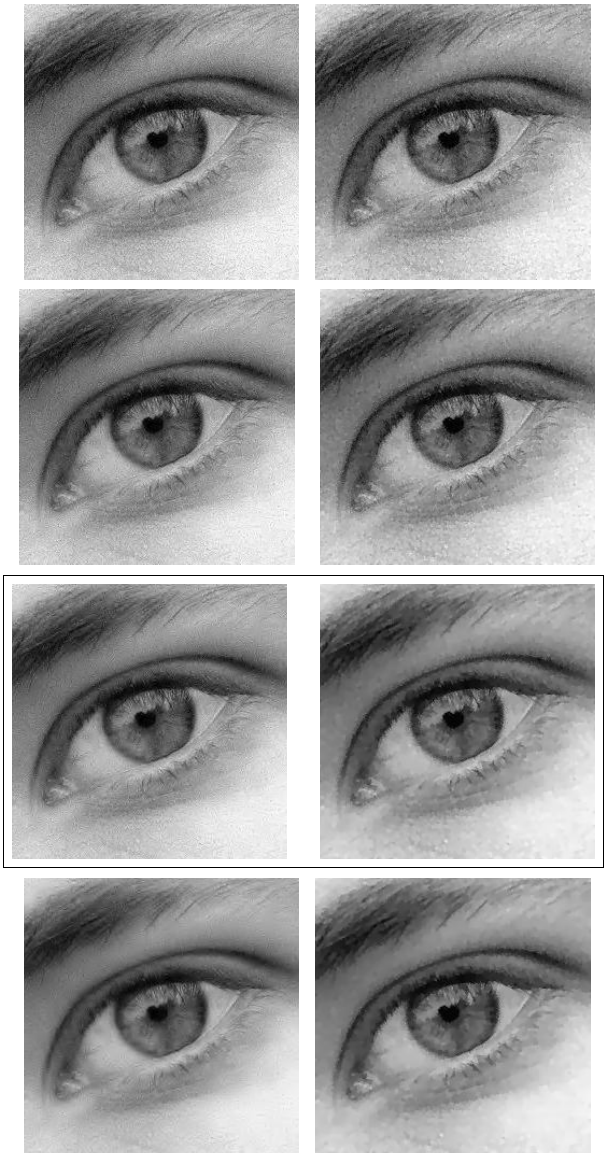

Figure 3: Left column: suggested multiscale iterative approach; right column: pixel–based iterative approach. 1st row: after 5 iterations; 2nd row: after 10 iterations; 3rd row: iteration with maximum SNR; 4th row: after 40 iterations.

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

$$\|S(c) - S(\tilde{c})\|_r \leq q\|c - \tilde{c}\|_r$$

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

$$\big\|S(c)(t) - S(\tilde{c})(t)\big\|_2 \leq \|A^{-1}\| \times \int_0^t \sum_{\lambda\in\Lambda_j} \Big|\big(B(c(s))c(s) - B(\tilde{c}(s))\tilde{c}(s)\big)_\lambda\Big| ds \tag{13}$$

and, moreover, for $s \in [0,T]$ we have

$$\Big(B(c(s))c(s) - B(\tilde{c}(s))\tilde{c}(s)\Big)_\lambda = \sum_{\mu\in\Lambda_j} \int_\Omega \Big(G_\mu(c(s))(x) - G_\mu(\tilde{c}(s))(x)\Big) \times \nabla\phi_\lambda(x)\nabla\phi_\mu(x)dx.$$

Now the Lipschitz continuity of G and the estimate

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

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

$$\bigg|\Big(B(c(s))c(s) - B(\tilde{c}(s))\tilde{c}(s)\Big)_\lambda\bigg| \leq L\big\|c(s) - \tilde{c}(s)\big\|_2 \sum_{\mu\in\Lambda_j} \int_\Omega \big|\nabla\phi_\lambda(x)\nabla\phi_\mu(x)\big|dx.$$

Combining the latter estimate with estimate (13) we obtain

$$\big\|S(c)(t) - S(\tilde{c})(t)\big\|_2 \quad\leq\quad C\|c-\tilde{c}\|_r \int_0^t e^{rs}ds, \tag{14}$$

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 (14), we finally obtain

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

showing 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. □

Proof of Theorem 3.2.
We observe that

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

The norm $\|d\|$ is finite on D_α since, by Theorem 3.1, the solution c of (7) is continuous on $[0,T]$ and therefore bounded on D_α. We have to choose α and j such that the expression on the right hand side is less than δ. $\square$

Proof of gray value conservation.
At first, we observe that $A^{1,j}$, $B^{1,j}$, $\tilde{B}^{1,j}$ satisfy the following properties: All row sums of $A^{1,j}$ are equal; this follows by the fact that $\sum_{l\in\mathbb{Z}^{1,j}} A^{1,j}_{kl}$ do not depend on k. Next, all row sums of $B^{1,j}$ as well as of $\tilde{B}^{1,j}$ are equal to zero; this can be deduced by observing that

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

where for $p \in \mathbb{Z}^{1,j}$ we have $\sum_{l\in\mathbb{Z}^{1,j}} [\varphi_l^j]'(2^{-j}p) = 0$. It follows then that $\sum_{l\in\mathbb{Z}^{1,j}} B^{1,j}_{kl}(c) = 0$ and that $\sum_{l\in\mathbb{Z}^{1,j}} \tilde{B}^{1,j}_{kl}(c) = 0$. Consequently,

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

It follows then that $\sum_{\lambda\in\Lambda_j} \big(Ac\big)_\lambda = \mathit{constant}$, and the proof is complete. $\square$

References

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

[2] S. Dahlke and P. Maass and G. Teschke, Interpolating Scaling Functions with Duals, J. Comput. Anal. Appl. 6(1), 2004, 19-29.

[3] W. Dahmen and A. Kunoth and K. Urban, A wavelet-Galerkin method for the Stokes equations, Computing 56, 1996, 259-301.

[4] W. Dahmen and C.A. Micchelli, Using the refinement equation for evaluating integrals of wavelets, SIAM J. Numer. Anal. 30, 1993, 507-537.

[5] W. Dahmen and S. Prößdorf and R. Schneider, Multiscale methods for pseudodifferential equations on smooth manifolds. In: C.K. Chui, L. Montefusco, and L. Puccio (eds), *Wavelets: Theory, Algorithms, and Applications*, Academic Press, Boston, 1995, 385–424.

[6] I. Daubechies, Ten Lectures on Wavlets, SIAM, Philadelphia, 1992.

[7] Deslaurier, G. and Dubuc, S., Interpolation dyadique, Fractals. Dimensions non entières et applications, 1051, 1987, 44–55.

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

[9] S. Dubuc, Interpolation through an iterative scheme, J. Math. Anal. Appl., 114, 1986, 185– 204.

[10] S. Kichenassamy, The Perona-Malik paradox, SIAM Journal of Applied Mathematics, 57, 5,1997, 1328–1342.

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

[12] Mathias Lindemann, Wavelet Expansions and Besov Spaces, Bremen University, PhD Thesis, 2005.

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

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

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

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

[17] Karsten Urban, On divergence-free wavelets, Advances in Computational Mathematics, 4, 1,2, 1995, 51-82.

[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: F. Solina, W.G. Kropatsch, R. Klette, and R. Bajcsy (eds), *Advances in Computer Vision*, Springer, Wien, 1997, 1-10.

[21] K. Yosida, Functional Analysis, Springer, Berlin, 1974.