A NONLINEAR MULTIGRID SOLVER WITH LINE GAUSS-SEIDEL-SEMISMOOTH-NEWTON SMOOTHER FOR THE FENCHEL PRE-DUAL IN TOTAL VARIATION BASED IMAGE RESTORATION

Ke Chen

Department of Mathematical Sciences, University of Liverpool
Peach Street, Liverpool L69 7ZL, UK

Yiqiu Dong

Institute of Biomathematics and Biometry
Helmholtz Zentrum München
Ingolstädter Landstrasse 1, 85764 Neuherberg, Germany

Michael Hintermüller

Department of Mathematics, Humboldt-University of Berlin
Unter den Linden 6, 10099 Berlin, Germany
and START-Project "Interfaces and Free Boundaries"
and SFB "Mathematical Optimization and Applications in Biomedical Science"
Institute of Mathematics and Scientific Computing, University of Graz
Heinrichstrasse 36, A-8010 Graz, Austria

(Communicated by Xuecheng Tai)

Abstract. Based on the Fenchel pre-dual of the total variation model, a non-linear multigrid algorithm for image denoising is proposed. Due to the structure of the differential operator involved in the Euler-Lagrange equations of the dual models, line Gauss-Seidel-semismooth-Newton step is utilized as the smoother, which provides rather good smoothing rates. The paper ends with a report on numerical results and a comparison with a very recent nonlinear multigrid solver based on Chambolle's iteration [6].

1. **Introduction.** Image restoration is a fundamental problem in image processing, and over the years various techniques for solving associated variational problems have been developed. Due to the potential non-uniqueness of the solution and the numerical instability from ill-conditioning, image restoration is an ill-posed problem and requires appropriate regularization for stabilization. Among the many techniques, total variation (TV) regularization as proposed by Rudin, Osher and Fatemi [22] is one of the most well-known ones. In this approach, the image u is restored by solving the minimization problem

$$(1) \qquad \min_{u \in BV(\Omega)} \frac{1}{2} \int_\Omega |Ku - z|^2 dx + \alpha \int_\Omega |Du|,$$

where $\Omega \subset \mathbb{R}^2$ is an open and bounded image domain with a Lipschitz-continuous boundary $\partial\Omega$, K is a linear and continuous blurring operator from $L^2(\Omega)$ to $L^2(\Omega)$,

i.e., $K \in \mathcal{L}(L^2(\Omega))$, which we assume to be known, and $\alpha > 0$ is a regularization parameter controlling the trade-off between fitting the degraded image $z \in L^2(\Omega)$ and a smoothness requirement due to the total variation regularization. By $BV(\Omega)$ we denote the space of functions of bounded variation, i.e. $u \in BV(\Omega)$ if $u \in L^1(\Omega)$ and the BV-seminorm

$$\int_\Omega |Du| = \sup \left\{ \int_\Omega u \, \mathrm{div}\vec{v} \, dx : \vec{v} \in (C_0^\infty(\Omega))^2, \|\vec{v}\|_\infty \le 1 \right\}$$

is finite. The space $BV(\Omega)$ endowed with the norm $\|u\|_{BV(\Omega)} = \|u\|_{L^1(\Omega)} + \int_\Omega |Du|$ is a Banach space; see, e.g., [13].

The non-differentiability of the TV-term $\int_\Omega |Du|$ is responsible for preserving edges in images, but at the same time, it poses significant numerical challenges. In the recent past, various research efforts have been undertaken to overcome this difficulty; see, e.g., [10, 5, 8, 24, 4, 21, 20, 23, 16, 17] as well as the monograph [31] and the many references therein.

The first order condition or Euler-Lagrange equation of (1) leads to a differential inclusion which, in a regularized form, becomes a system of nonlinear partial differential equations (PDEs). For their numerical solution, multigrid (MG) methods appear to be ideal candidates as they are known to be efficient numerical techniques for solving partial differential equations; see, e.g., [14, 27, 32]. The convergence behavior of MG techniques usually depends on the type and structure of the differential operator. For linear elliptic second order operators the method has been analyzed and applied successfully, e.g., in [14] and the references therein. Nonlinear variants can be found in [3, 14], for instance. In the multigrid literature, the differential operator is usually assumed to enjoy differentiability properties, whereas the system resulting from the Euler-Lagrange equation of (1) is non-differentiable. This necessitates appropriate reformulations of the PDE system and extensions of smoothing steps in order to account for the non-differentiability.

In [29] a multigrid (MG) solver acting on the primal problem (1) was proposed. In that paper it is noted that a straightforward application of a (nonlinear) MG method to a smoothed version of the Euler-Lagrange system does not work well due to the discontinuity in the diffusion coefficient. Rather the authors propose to resort to a flux-formulation similar to flow through a porous medium with fixed porosity (which amounts to a reformulation of the usual "lagged diffusivity" formulation [30]). It appears that this format improves over the straightforward application of MG. Several discretization and smoothing operations for yet a primal version of multigrid for total variation regularization were investigated in [12]. In [28] multigrid respectively multilevel techniques are used as preconditioners of the conjugate gradient method applied to the subproblem of a "lagged diffusivity" iteration for solving the total variation regularized problem.

Recently, dual formulations of the total variation problem and associated solution algorithms became popular; see, e.g., [4, 7, 16, 17]. Furthermore, the multigrid algorithm is introduced to solve the dual problem which is a quadratic minimization under a convex constraint, see [26, 25]. In [6], a nonlinear multigrid algorithm was proposed to solve the dual formulation of (1) formally given by

$$(2) \qquad -\nabla(\alpha\mathrm{div}(\vec{\omega}) - z) + |\nabla(\alpha\mathrm{div}(\vec{\omega}) - z)|_{l^2}\vec{\omega} = 0,$$

where $\vec{\omega} = (\omega_1, \omega_2)$ is the dual variable. The restored image is obtained as $u = z - \alpha\mathrm{div}(\vec{\omega})$. In this algorithm, the semi-implicit time-marching scheme of [4] (which

we call Chambolle's iteration or method in what follows) is utilized in the smoothing step of the multigrid method. It is known that the convergence rate of the smoothing step is at best linear (in the primal variable) and does not exhibit a resolution independent convergence behavior. Moreover, the dual variable is not necessarily unique at the solution due to the non-trivial kernel of the generalized divergence operator, the adjoint of the derivative in the TV-term. As a consequence of these facts, the efficiency of the multigrid iteration appears to be compromised. Moreover, for practical purposes it turns out that $|\nabla(\alpha\mathrm{div}(\vec{\omega}) - z)|_{l^2}$ close to zero is problematic due to the non-differentiability of the total variation term. This affects the efficiency of the smoother. As a remedy, in [6] (like in the Newton-type scheme [7]) $|\nabla(\alpha\mathrm{div}(\vec{\omega}) - z)|_{l^2}$ is replaced by $\sqrt{|\nabla(\alpha\mathrm{div}(\vec{\omega}) - z)|^2 + \beta}$ with sufficiently large $\beta > 0$ to maintain an acceptable smoothing rate and, thus, turning a non-differentiable term into a C^∞-term. The choice of β implies a decision between a good smoothing rate versus the quality of the reconstruction. In particular, large β leads to artifacts in homogeneous image regions.

In this paper we aim at the introduction of a nonlinear multigrid algorithm in the framework of the dual problem, which overcomes the above mentioned limitations. In fact, the Fenchel pre-dual of (1) is known to be a convex quadratic problem subject to bound-constraints [16]. Since the objective function of the pre-dual problem involves the divergence operator, which has a nontrivial kernel, we propose two models with different regularization terms, respectively, to avoid the nonuniqueness of the dual solution. Then, based on these two models we utilize the nonlinear multigrid algorithm to solve the pertinent Euler-Lagrange equations. Compared to the smoothed model in [6], our two models are much closer to the original nonsmooth total variation formulation. Moreover, one of the models allows for the application of so-called semismooth Newton solvers [15] in the associated function space setting resulting in a resolution independent solution algorithm. Combined with a nonlinear multigrid algorithm, we propose a line Gauss-Seidel-semismooth-Newton step as the smoother, which provides better smoothing rates than the smoother suggested in [6]. Line smoothing is necessary due to the structure of the differential operator involved in the Euler-Lagrange equations.

The outline of the rest of the paper is as follows. In Section 2 we recall some basic facts from convex analysis and state the pre-dual of (1) in the sense of Fenchel. Section 3 is devoted to the description of our nonlinear multigrid algorithm. Section 4 gives numerical results to demonstrate the performance of the new method. Moreover, we compare our method with the one in [6]. Finally, conclusions are drawn in Section 5.

2. **Pre-dual problem of total variation model for image restoration.** We start by recalling the Fenchel duality concept in infinite dimensional spaces in a form that is convenient for our work; see, e.g., [11] for details. For this purpose let V and Y be Banach spaces with topological duals V^* and Y^*, respectively. Let $\mathcal{F} : V \to \mathbb{R} \cup \{\infty\}$ and $\mathcal{G} : Y \to \mathbb{R} \cup \{\infty\}$ be convex, lower semi-continuous functionals such that there exists $v_0 \in V$ with $\mathcal{F}(v_0) < \infty$, $\mathcal{G}(\Lambda v_0) < \infty$, $\mathcal{G}$ is continuous at Λv_0, and $\Lambda \in \mathcal{L}(V, Y)$ is a continuous linear operator from V to Y. Then we have

$$(3) \qquad \inf_{u \in V} \mathcal{F}(u) + \mathcal{G}(\Lambda u) = \sup_{p \in Y^*} -\mathcal{F}^*(\Lambda^* p) - \mathcal{G}^*(-p),$$

where the conjugate of $\mathcal{F}$ is defined by $\mathcal{F}^* : V^* \to \mathbb{R} \cup \{\infty\}$

$$(4) \qquad \mathcal{F}^* = \sup_{v \in V} \langle v, v^* \rangle_{V,V^*} - \mathcal{F}(v),$$

and analogously for $\mathcal{G}^* : Y^* \to \mathbb{R} \cup \{\infty\}$.

In order to compute the Fenchel dual of (1) formally, we set $\Lambda = \nabla$,

$$\mathcal{F}(u) = \frac{1}{2} \int_\Omega |Ku - z|^2 \, dx \quad \text{and} \quad \mathcal{G}(\vec{p}) = \alpha \int_\Omega |\vec{p}|_{l^2} \, dx.$$

Based on the definition of the Fenchel conjugate (4), we get

$$\mathcal{F}^*(u^*) = \frac{1}{2} \|u^* + K^* z\|_B^2 - \frac{1}{2} \int_\Omega z^2 \, dx \quad \text{and} \quad \mathcal{G}^*(\vec{p}^*) = I_{\{\vec{w} : |\vec{w}|_{l^2} \leq \alpha\}}(\vec{p}^*),$$

where $\|v\|_B^2 = \langle v, B^{-1} v \rangle$ with $B = K^* K$, which is assumed to be invertible, and I_S is the indicator function of the set S. Here, $\langle \cdot, \cdot \rangle$ denotes a duality pairing. Hence, in view of (3) the formal dual of (1) is given by

(5)
$$\min_{\vec{p}} \frac{1}{2} \|\text{div}\vec{p} + K^* z\|_B^2 \qquad \text{subject to (s.t.)} \quad |\vec{p}|_{l^2} \leq \alpha \text{ almost everywhere (a.e.) in } \Omega.$$

Assuming sufficient regularity of u, observe that $|\vec{p}|_{l^2} \leq \alpha$ in Ω results from $\vec{p}(x) \in \alpha B(0,1)$, where $B(0,1)$ denotes the closed unit ball in $\mathbb{R}^2$, whenever $\nabla u(x) = 0$ and $|\vec{p}(x)|_{l^2} = \alpha$ for $|\nabla u(x)|_{l^2} > 0$. In [16], the following result concerning (5) was established.

Theorem 2.1. *The Fenchel dual of*

$$(6) \qquad \min_{\vec{p} \in \mathbf{H}_0(\text{div})} \frac{1}{2} \|\text{div}\vec{p} + K^* z\|_B^2 \qquad \text{s.t.} \quad |\vec{p}|_{l^2} \leq \alpha \quad a.e. \text{ in } \Omega$$

is given by (1)*, where* $\mathbf{H}_0(\text{div}) = \{\vec{v} \in \mathbf{L}^2(\Omega) : \text{div}\vec{v} \in L^2(\Omega), \vec{v} \cdot \vec{n} = 0 \text{ on } \partial\Omega\}$ *with* $\vec{n}$ *is the outer unit normal to* $\partial\Omega$*, and* $\mathbf{L}^2(\Omega) = L^2(\Omega) \times L^2(\Omega)$*. Alternatively,* (6) *can be regarded as the pre-dual of* (1)*.*

Since the kernel of the divergence operator is nontrivial, we add a dual regularization term with associated regularization parameter $\mu > 0$ to enforce uniqueness of the solution of the pre-dual. Here, we consider the following two variants:

$$(7) \qquad \min_{\vec{p} \in \mathbf{H}_0^1(\Omega)} \frac{1}{2} \|\text{div}\vec{p} + K^* z\|_B^2 + \frac{\mu}{2} \int_\Omega |\vec{\nabla}\vec{p}|_{l^2}^2 \, dx \qquad \text{s.t. } |\vec{p}|_{l^2} \leq \alpha \text{ a.e. in } \Omega,$$

and

$$(8) \qquad \min_{\vec{p} \in \mathbf{H}_0(\text{div})} \frac{1}{2} \|\text{div}\vec{p} + K^* z\|_B^2 + \frac{\mu}{2} \int_\Omega |\vec{p}|_{l^2}^2 \, dx \qquad \text{s.t. } |\vec{p}|_{l^2} \leq \alpha \text{ a.e. in } \Omega,$$

where $\mathbf{H}_0^1(\Omega) = H_0^1(\Omega) \times H_0^1(\Omega)$, which is dense in $\mathbf{H}_0(\text{div})$. Note that the regularization in (7) lifts $\vec{p}$ into $\mathbf{H}_0^1(\Omega)$ whereas the second regularization maintains the function space regularity of the original pre-dual problem (6). Like in [17] we are able to find the effect of the dual regularization in (8) in the primal problem. In fact, the total variation term $\int_\Omega |Du|$ in (1) is approximated by $\int_\Omega \Phi_{\mu\alpha}(\nabla u) \, dx$ with

$$\Phi_{\mu\alpha}(\vec{w})(x) := \begin{cases} |\vec{w}(x)|_{l^2} - \frac{\mu\alpha}{2} & \text{if } |\vec{w}(x)|_{l^2} \geq \mu\alpha, \\ \frac{1}{2\mu\alpha} |\vec{w}(x)|_{l^2}^2 & \text{if } |\vec{w}(x)|_{l^2} < \mu\alpha. \end{cases}$$

Observe that $\Phi_{\mu\alpha}(\vec{w})$ is a Huber-type function (see [19]) which smoothes the total variation regularization only locally. For the first regularization the effect in the

primal problem is non-local in the sense that the regularity of the dual variable $\vec{p}$ increases from $\mathbf{H}_0(\mathrm{div})$ to $\mathbf{H}_0^1(\Omega)$.

In [16], see also [17], it was shown that the penalized version of (8) given by

$$(9) \qquad \min_{\vec{p} \in \mathbf{H}_0(\mathrm{div})} \frac{1}{2} \|\mathrm{div}\vec{p} + K^* z\|_B^2 + \frac{\mu}{2} \int_\Omega |\vec{p}|_{l^2}^2 \, dx + \frac{\gamma}{2} \int_\Omega \max(0, |\vec{p}|_{l^2} - \alpha)^2 \, dx,$$

with a penalty parameter $\gamma > 0$ and the max-term penalizing violations of the pointwise a.e. constraints in (8) can be solved by a semismooth Newton iteration acting on the Euler-Lagrange equation. It is well-known that $\gamma \to \infty$ enforces the constraint $|\vec{p}|_{l^2} \le \alpha$. Moreover, it can be shown that $\|\max(0, |\vec{p}_\gamma|_{l^2} - \alpha)\|_{L^2(\Omega)} \le C\gamma^{-\frac{1}{2}}$ for some constant $C > 0$ (independent of γ), where $\vec{p}_\gamma$ denotes the solution of (9). The semismooth Newton iteration converges locally at a superlinear rate in function space and, as a consequence, exhibits an (image) resolution independent convergence behavior in the discrete setting. This suggests to utilize such a semismooth Newton solver as the smoothing iteration for the nonlinear multigrid method given below.

Motivated by (9), we approximate (7) by

$$(10) \qquad \min_{\vec{p} \in \mathbf{H}_0^1(\Omega)} \frac{1}{2} \|\mathrm{div}\vec{p} + K^* z\|_B^2 + \frac{\mu}{2} \int_\Omega |\vec{\nabla}\vec{p}|_{l^2}^2 \, dx + \frac{\gamma}{2} \int_\Omega \max(0, |\vec{p}|_{l^2} - \alpha)^2 \, dx,$$

The Euler-Lagrange equations for (10) and (9), respectively, read:

$$(11) \qquad -\mu\vec{\Delta}\bar{\vec{p}} - \nabla(K^*K)^{-1}\mathrm{div}\bar{\vec{p}} + \gamma \max(0, |\bar{\vec{p}}|_{l^2} - \alpha)\bar{\vec{q}} = \nabla(K^*K)^{-1}K^*z,$$

or

$$(12) \qquad \mu\bar{\vec{p}} - \nabla(K^*K)^{-1}\mathrm{div}\bar{\vec{p}} + \gamma \max(0, |\bar{\vec{p}}|_{l^2} - \alpha)\bar{\vec{q}} = \nabla(K^*K)^{-1}K^*z,$$

where

$$\bar{\vec{q}}(x) \in \begin{cases} \{\frac{\bar{\vec{p}}}{|\bar{\vec{p}}|_{l^2}}(x)\} & \text{if } |\bar{\vec{p}}(x)|_{l^2} > 0, \\ B(0,1) & \text{if } |\bar{\vec{p}}(x)|_{l^2} = 0. \end{cases}$$

Here, $\bar{\vec{p}}$ denotes the unique solution of (10) or (9), respectively.

In the next section, we introduce a nonlinear multigrid algorithm to solve (11) and (12) in the denoising case, i.e., for K equal to the identity operator. The restored image is obtained by $\bar{u} = \mathrm{div}\bar{\vec{p}} + z$. Variants of nonlinear multigrid methods in the deblurring case, i.e. for K not the identity, might be conceivable but go beyond the scope of the present work.

3. **Nonlinear multigrid method for the dual problem.** For (11) and (12) in the denoising case, i.e., $K = \mathrm{id}$ yielding

$$(13) \qquad -\mu\vec{\Delta}\vec{p} - \nabla\mathrm{div}\vec{p} + \gamma \max(0, |\vec{p}|_{l^2} - \alpha)\vec{q} = \nabla z,$$

and

$$(14) \qquad \mu\vec{p} - \nabla\mathrm{div}\vec{p} + \gamma \max(0, |\vec{p}|_{l^2} - \alpha)\vec{q} = \nabla z,$$

we introduce a nonlinear multigrid scheme for restoring high resolution images efficiently. From now on we proceed in discrete terms, but, for the sake of simplicity and readability, we keep the notation from the continuous context.

Assume that the discrete image domain Ω contains $(m-1) \times (n-1)$ pixels, and define the discrete gradient operator $\nabla : \mathbb{R}^{(m-1)\times(n-1)} \to \mathbb{R}^{2(m-1)\times(n-1)}$ by

$$\nabla z := \begin{bmatrix} \nabla_x z \\ \nabla_y z \end{bmatrix}$$

for $z \in \mathbb{R}^{(m-1) \times (n-1)}$ with $\nabla_x, \nabla_y : \mathbb{R}^{(m-1) \times (n-1)} \to \mathbb{R}^{(m-1) \times (n-1)}$ corresponding to the discrete derivative in the x- and y-direction, respectively. Furthermore, we have the divergence operator $\mathrm{div} = -\nabla^\top$ and $\vec{\triangle} : \mathbb{R}^{2(m-1) \times (n-1)} \to \mathbb{R}^{2(m-1) \times (n-1)}$ given by

$$\vec{\triangle}\vec{p} := \left[\begin{array}{c} \triangle p_1 \\ \triangle p_2 \end{array} \right]$$

with $\vec{p} = \left[\begin{array}{c} p_1 \\ p_2 \end{array} \right] \in \mathbb{R}^{2(m-1) \times (n-1)}$, $p_1, p_2 \in \mathbb{R}^{(m-1) \times (n-1)}$ and $\triangle : \mathbb{R}^{(m-1) \times (n-1)} \to \mathbb{R}^{(m-1) \times (n-1)}$. For convenience, we denote the discretized versions of (13) and (14) by

(15) $$\mathcal{A}_1(\vec{p}) := -\mu\vec{\triangle}\vec{p} - \nabla\mathrm{div}\vec{p} + \gamma M_\alpha \star \vec{p} = \vec{r},$$

(16) $$\mathcal{A}_2(\vec{p}) := \mu\vec{p} - \nabla\mathrm{div}\vec{p} + \gamma M_\alpha \star \vec{p} = \vec{r},$$

with $\vec{r} = \nabla z$, respectively. Here, M_α is a $2(m-1) \times (n-1)$-matrix with $(M_\alpha)_{i,j} = (M_\alpha)_{m-1+i,j} = \max(0, 1 - \frac{\alpha}{|[\vec{p}]_{i,j}|_{l^2} 2})$ and $|[\vec{p}]_{i,j}|_{l^2} = \sqrt{|(p_1)_{i,j}|^2 + |(p_2)_{i,j}|^2}$, and $\star$ denotes the Hadamard product. Then, the nonlinear full approximation scheme (FAS) multigrid algorithm [32, 27, 9] for solving either (15) or (16) reads as follows.

FAS Multigrid Algorithm.
  1: Initialize $\vec{p}^1 \in \mathbb{R}^{2(m-1) \times (n-1)}$. Set $i \in \{1, 2\}$ and $\vec{r}^1 = \nabla z$. Start from finest level $l = 1$ and do j steps of a nonlinear multigrid V-cycle for solving $\mathcal{A}_i(\vec{p}) = \vec{r}^1$, which we call **NMG**$(\vec{p}^1, \vec{r}^1, 1)$.
  2: One step of a V-cycle of our nonlinear multigrid procedure **NMG**$(\vec{p}^l, \vec{r}^l, l)$ proceeds as follows:
    • if $l = L$, the coarsest level, perform a high accuracy solve of $\mathcal{A}_i^L(\vec{p}) = \vec{r}^L$ and let $\vec{p}^L$ denote the corresponding solution.
    • else, on level l, do
        – Pre-smoothing: $\vec{p}^l = \mathcal{S}_l^{\nu_1}(\vec{p}^l, \vec{r}^l)$.
        – Restrict to the coarser grid: $\vec{p}^{l+1} = \mathcal{R}_l^{l+1}\vec{p}^l$.
        – Compute $\vec{r}^{l+1} = \mathcal{R}_l^{l+1}(\vec{r}^l - \mathcal{A}_i^l(\vec{p}^l)) + \mathcal{A}_i^{l+1}(\vec{p}^{l+1})$.
        – Set the initial solution on level $l+1$ as $\tilde{\vec{p}}^{l+1} = \vec{p}^{l+1}$.
        – Call **NMG**$(\tilde{\vec{p}}^{l+1}, \vec{r}^{l+1}, l+1)$.
        – Interpolation the correction and compute the corrected approximation: $\vec{p}^l = \vec{p}^l + \mathcal{P}_{l+1}^l(\tilde{\vec{p}}^{l+1} - \vec{p}^{l+1})$.
        – Post-smoothing: $\vec{p}^l = \mathcal{S}_l^{\nu_2}(\vec{p}^l, \vec{r}^l)$.
    • end one step of **NMG**$(\vec{p}^l, \vec{r}^l, l)$

Here $\mathcal{P}_{l+1}^l$ denotes an interpolation operator from level $l+1$ to level l, ν_1 and ν_2 are the numbers of the pre-smoothing steps and the post-smoothing steps on each level l, respectively.

Next, we describe each element of the FAS multigrid algorithm for solving (13) and (14) in detail.

3.1. **Line Gauss-Seidel-Newton smoother.** The convergence of the overall multigrid scheme hinges on the performance and reduction rates of the smoothing iteration. For this reason, we first focus on the choice of the smoother in $\mathcal{S}_l^{\nu_i}(\vec{p}^l, \vec{r}^l)$ with $i = 1, 2$. Note that according to [27] the h-ellipticity property of the discrete differential operator is important when constructing a pointwise smoother. Since $\nabla\mathrm{div}$ in (13) and (14) does not obey such a h-ellipticity property we have to proceed differently, i.e. with an alternative to a pointwise smoother. In fact, as a remedy we

propose to use a line Gauss-Seidel-Newton iteration as a smoother in our nonlinear multigrid algorithm.

It is well-known that line relaxation is particularly efficient when solving anisotropic problems due to its effectivity in the direction of strong coupling. As the relevant term in our case is $-\nabla \mathrm{div}\vec{p}$ we clearly find that the directions of strong coupling of p_1 and p_2 are different. In fact, for p_1 the x-direction couples strongly, whereas for p_2 the y-direction is coupling strongly. Consequently, we utilize an alternate line relaxation in our smoothing step. Below we exemplarily discuss the x-line Gauss-Seidel-Newton iteration in detail. The y-direction relaxation can be obtain similarly.

We discretize the Laplace operator by the standard five-point stencil with homogenous Dirichlet boundary conditions and use forward differences for the gradient and corresponding discrete adjoint scheme for the divergence operator. Then, the operators that appear in (13) and (14) can be written as

$$(\nabla v)_{i,j} \approx \left[\begin{array}{c} \frac{v_{i+1,j} - v_{i,j}}{h_x} \\ \frac{v_{i,j+1} - v_{i,j}}{h_y} \end{array} \right],$$

$$(\mathrm{div}\vec{p})_{i,j} \approx \frac{(p_1)_{i,j} - (p_1)_{i-1,j}}{h_x} + \frac{(p_2)_{i,j} - (p_2)_{i,j-1}}{h_y},$$

$$(\Delta v)_{i,j} \approx \frac{v_{i+1,j} + v_{i-1,j} - 2v_{i,j}}{h_x^2} + \frac{v_{i,j+1} + v_{i,j-1} - 2v_{i,j}}{h_y^2},$$

$$(\nabla \mathrm{div}\vec{p})_{i,j} \approx \left[\begin{array}{c} \frac{(p_1)_{i+1,j} + (p_1)_{i-1,j} - 2(p_1)_{i,j}}{h_x^2} + \frac{(p_2)_{i+1,j} + (p_2)_{i,j-1} - (p_2)_{i,j} - (p_2)_{i+1,j-1}}{h_x h_y} \\ \frac{(p_2)_{i,j+1} + (p_2)_{i,j-1} - 2(p_2)_{i,j}}{h_y^2} + \frac{(p_1)_{i,j+1} + (p_1)_{i-1,j} - (p_1)_{i,j} - (p_1)_{i-1,j+1}}{h_x h_y} \end{array} \right],$$

for $1 \leq i \leq m-1$ and $1 \leq j \leq n-1$,

where h_x and h_y are the widths of the subintervals in x-direction and y-direction, respectively. Here (and in our numerics) we consider the image domain *with* boundary given by $\bar{\Omega} = [0,1]^2$. Corresponding to the assumption at the beginning of Section 3, on the discrete level it contains $(m+1) \times (n+1)$ pixels in total. Hence, we have $h_x = \frac{1}{m}$ and $h_y = \frac{1}{n}$, and $i = 0 \wedge i = m$ respectively $j = 0 \wedge j = n$ belong to pixels on the boundary $\partial\Omega$, where the values of v are assumed to be known.

The x-line Gauss-Seidel iterations for the discretized versions of (13) and (14) are denoted as

$$(17) \qquad \mathcal{A}^{\mathrm{GS}}(\vec{p}^{\,k+1}_{\cdot,1}, \cdots, \vec{p}^{\,k+1}_{\cdot,j-1}, \vec{p}^{\,k+1}_{\cdot,j}, \vec{p}^{\,k}_{\cdot,j+1}, \cdots, \vec{p}^{\,k}_{\cdot,n}) = \vec{r}^{\,k},$$

where $\vec{p}_{\cdot,j}$ is the jth row of $\vec{p}$, and the indices k and $k+1$ denote the current and the new approximation, respectively. We note that these nonlinear equations for the unknown $\vec{p}^{\,k+1}_{\cdot,j} = ((p_1^{k+1})_{\cdot,j}, (p_2^{k+1})_{\cdot,j})^\top$ are non-smooth, i.e. not necessarily Fréchet-differentiable. This is due to the presence of the maximum function. But they can be solved efficiently by a semismooth Newton method, which converges locally at a superlinear rate [17]. Applying a generalized Newton step to (17) with initial value $\vec{p}^{\,k}_{\cdot,j}$ yields

$$(18) \qquad \left[\begin{array}{cc} H_{11} & H_{12} \\ H_{21} & H_{22} \end{array} \right] \left[\begin{array}{c} (p_1^{t+1})^\top_{\cdot,j} \\ (p_2^{t+1})^\top_{\cdot,j} \end{array} \right] = \left[\begin{array}{c} F_1^t \\ F_2^t \end{array} \right],$$

where $H_{21} = H_{12}^\top$.

Remark 1. For the nonlinear equation (13), we have

$$
H_{11} = \begin{bmatrix}
(d_1^t)_{1,j} & -\frac{\mu+1}{h_x h_y} & & & \\
-\frac{\mu+1}{h_x h_y} & (d_1^t)_{2,j} & -\frac{\mu+1}{h_x h_y} & & \\
& \ddots & \ddots & \ddots & \\
& & -\frac{\mu+1}{h_x h_y} & (d_1^t)_{m-1,j} & -\frac{\mu+1}{h_x h_y} \\
& & & -\frac{\mu+1}{h_x h_y} & (d_1^t)_{m,j}
\end{bmatrix},
$$

$$
H_{12} = \begin{bmatrix}
(d_2^t)_{1,j} & -\frac{1}{h_x h_y} & & \\
& \ddots & \ddots & \\
& & (d_2^t)_{m-1,j} & -\frac{1}{h_x h_y} \\
& & & (d_2^t)_{m,j}
\end{bmatrix},
$$

$$
H_{22} = \begin{bmatrix}
(d_3^t)_{1,j} & -\frac{\mu}{h_x h_y} & & & \\
-\frac{\mu}{h_x h_y} & (d_3^t)_{2,j} & -\frac{\mu}{h_x h_y} & & \\
& \ddots & \ddots & \ddots & \\
& & -\frac{\mu}{h_x h_y} & (d_3^t)_{m-1,j} & -\frac{\mu}{h_x h_y} \\
& & & -\frac{\mu}{h_x h_y} & (d_3^t)_{m,j}
\end{bmatrix},
$$

$$
\begin{aligned}
F_1^t =& r_1^k + \frac{\mu}{h_x h_y}\big[(p_1^{k+1})_{\cdot,j+1} + (p_1^k)_{\cdot,j-1}\big] + \frac{1}{h_x h_y}\big[(p_2^{k+1})_{\cdot,j-1} - (p_2^k)_{\cdot+1,j-1}\big] \\
& + \gamma \chi_{i,j}^t \frac{\alpha(p_1^t)_{i,j}}{|\vec{p}_{i,j}^t|_{l^2}},
\end{aligned}
$$

$$
\begin{aligned}
F_2^t =& r_2^k + \frac{\mu+1}{h_x h_y}\big[(p_2^{k+1})_{\cdot,j+1} + (p_2^k)_{\cdot,j-1}\big] + \frac{1}{h_x h_y}\big[(p_1^k)_{\cdot,j+1} - (p_1^{k+1})_{\cdot-1,j+1}\big] \\
& + \gamma \chi_{i,j}^t \frac{\alpha(p_2^t)_{i,j}}{|\vec{p}_{i,j}^t|_{l^2}},
\end{aligned}
$$

where $(d_1^t)_{i,j} = \frac{4\mu+2}{h_x h_y} + \gamma\chi_{i,j}^t\big[1 - \frac{\alpha(p_2^t)_{i,j}^2}{|\vec{p}_{i,j}^t|_{l^2}^3}\big]$, $(d_2^t)_{i,j} = \frac{1}{h^2} + \gamma\chi_{i,j}^t\frac{\alpha(p_1^t)_{i,j}(p_2^t)_{i,j}}{|\vec{p}_{i,j}^t|_{l^2}^3}$, $(d_3^t)_{i,j} = \frac{4\mu+2}{h_x h_y} + \gamma\chi_{i,j}^t\big[1 - \frac{\alpha(p_1^t)_{i,j}^2}{|\vec{p}_{i,j}^t|_{l^2}^3}\big]$, and $\chi_{i,j}^t = \begin{cases} 1, & \text{if } |\vec{p}_{i,j}^t|_{l^2} > \alpha, \\ 0, & \text{else.} \end{cases}$

For the generalized Newton step for equation (14), we have

$$
H_{11} = \begin{bmatrix}
(d_1^t)_{1,j} & -\frac{1}{h_x h_y} & & & \\
-\frac{1}{h_x h_y} & (d_1^t)_{2,j} & -\frac{1}{h_x h_y} & & \\
& \ddots & \ddots & \ddots & \\
& & -\frac{1}{h_x h_y} & (d_1^t)_{m-1,j} & -\frac{1}{h_x h_y} \\
& & & -\frac{1}{h_x h_y} & (d_1^t)_{m,j}
\end{bmatrix},
$$

$$
H_{12} = H_{12} \text{ for (13)},
$$

$$
H_{22} = \begin{bmatrix}
(d_3^t)_{1,j} & & \\
& \ddots & \\
& & (d_3^t)_{m,j}
\end{bmatrix},
$$

$$F_1^t = r_1^k + \frac{1}{h_x h_y}\left[(p_2^{k+1})_{\cdot,j-1} - (p_2^k)_{\cdot+1,j-1}\right] + \gamma\chi_{i,j}^t \frac{\alpha(p_1^t)_{i,j}}{|\vec{p}_{i,j}^t|_{l^2}},$$

$$F_2^t = r_2^k + \frac{1}{h_x h_y}\left[(p_1^k)_{\cdot,j+1} - (p_1^{k+1})_{\cdot-1,j+1}\right] + \gamma\chi_{i,j}^t \frac{\alpha(p_2^t)_{i,j}}{|\vec{p}_{i,j}^t|_{l^2}},$$

where $(d_1^t)_{i,j} = \mu + \frac{2}{h_x h_y} + \gamma\chi_{i,j}^t\left[1 - \frac{\alpha(p_2^t)_{i,j}^2}{|\vec{p}_{i,j}^t|_{l^2}^3}\right]$, $(d_2^t)_{i,j} = \frac{1}{h^2} + \gamma\chi_{i,j}^t \frac{\alpha(p_1^t)_{i,j}(p_2^t)_{i,j}}{|\vec{p}_{i,j}^t|_{l^2}^3}$, and $(d_3^t)_{i,j} = \mu + \frac{2}{h_x h_y} + \gamma\chi_{i,j}^t\left[1 - \frac{\alpha(p_1^t)_{i,j}^2}{|\vec{p}_{i,j}^t|_{l^2}^3}\right]$.

In order to study the efficiency of the nonlinear multigrid algorithm with these line Gauss-Seidel-Newton smoothers, in the following we use local Fourier analysis to measure the largest amplification factor in the relaxation scheme (18).

3.1.1. *The smoothing property analysis.* The convergence behavior of a multigrid algorithm depends strongly on the smoother. A convenient tool for the study of the expected reduction or smoothing rate is Fourier analysis.

Since the smoothing rate of the nonlinear multigrid algorithm proposed in [6] is mainly limited by the case when $|(\nabla u^{k+1})_{i,j}|_{l^2}$ is close to zero, in this section we mostly focus on this case, i.e., $|\vec{p}_{i,j}^{k+1}|_{l^2} \leq \alpha$ (see the arguments right before Theorem 1). Then, we have $\max\left(0, 1 - \frac{\alpha}{|\vec{p}_{i,j}^{k+1}|_{l^2}}\right) = 0$. In the following, for simplifying the formulas, we assume that the numbers of the intervals in each row and each column are equal to n, i.e., $m = n$ and $h := \frac{1}{n} = h_x = h_y$. The extension to $m \neq n$ is straightforward.

Define the local error functions by $e_1^k = p_1 - p_1^k$ and $e_2^k = p_2 - p_2^k$, where $\vec{p} = (p_1, p_2)$ is the solution of (13) or (14), and $\vec{p}^k = (p_1^k, p_2^k)$ is the result after k steps of the line Gauss-Seidel iteration as defined in (17). Similar to the local Fourier analysis in [32, 27, 9], let

$$e_1^k(i,j) = \sum_{\theta_\alpha, \theta_\beta = -n/2}^{n/2} (c_1^k)_{\theta_\alpha, \theta_\beta} \psi_{\theta_\alpha, \theta_\beta}(i,j),$$

$$e_2^k(i,j) = \sum_{\theta_\alpha, \theta_\beta = -n/2}^{n/2} (c_2^k)_{\theta_\alpha, \theta_\beta} \psi_{\theta_\alpha, \theta_\beta}(i,j),$$

where $\psi_{\theta_\alpha, \theta_\beta}(i,j) = e^{\frac{2\pi I(\theta_\alpha i + \theta_\beta j)}{n}}$ with I the imaginary unit. Then, the so-called amplification matrix $\mathcal{M}_{\theta_\alpha, \theta_\beta}$ is defined as

$$\left[\begin{array}{c} (c_1^{k+1})_{\theta_\alpha, \theta_\beta} \\ (c_2^{k+1})_{\theta_\alpha, \theta_\beta} \end{array}\right] = \mathcal{M}_{\theta_\alpha, \theta_\beta} \left[\begin{array}{c} (c_1^k)_{\theta_\alpha, \theta_\beta} \\ (c_2^k)_{\theta_\alpha, \theta_\beta} \end{array}\right],$$

and the smoothing rate $\max_{\theta_\alpha, \theta_\beta} \rho(\mathcal{M}_{\theta_\alpha, \theta_\beta})$ is the maximum spectral radius of $\mathcal{M}_{\theta_\alpha, \theta_\beta}$ in the high frequency range $(\theta_\alpha, \theta_\beta) \in [-\frac{n}{2}, \frac{n}{2}]^2 \setminus [-\frac{n}{4}, \frac{n}{4}]^2$.

Substituting the local error functions into the x-line Gauss-Seidel iteration (17), we obtain the following amplification matrices for (13) and (14), respectively:

$$\mathcal{M}_{\theta_\alpha, \theta_\beta}^1 = \left[\begin{array}{cc} 4\mu+2-2(\mu+1)\cos\frac{2\pi I\theta_\alpha}{n} - \mu e^{\frac{-2\pi I\theta_\beta}{n}} & 1-e^{\frac{2\pi I\theta_\alpha}{n}} - e^{\frac{-2\pi I\theta_\beta}{n}} + e^{\frac{2\pi I(\theta_\alpha-\theta_\beta)}{n}} \\ 1-e^{\frac{-2\pi I\theta_\alpha}{n}} & 4\mu+2-(\mu+1)e^{\frac{-2\pi I\theta_\beta}{n}} - 2\mu\cos\frac{2\pi I\theta_\alpha}{n} \end{array}\right]^{-1}$$

$$\cdot \left[\begin{array}{cc} \mu e^{\frac{2\pi I\theta_\beta}{n}} & 0 \\ e^{\frac{2\pi I\theta_\beta}{n}} - e^{\frac{2\pi I(\theta_\beta-\theta_\alpha)}{n}} & (\mu+1)e^{\frac{-2\pi I\theta_\beta}{n}} \end{array}\right],$$

μ for $\mathcal{M}^1_{\theta_\alpha,\theta_\beta}$,	Line GS method		Chambolle's method	
μh^2 for $\mathcal{M}^2_{\theta_\alpha,\theta_\beta}$,	$\max\limits_{\theta_\alpha,\theta_\beta} \rho(\mathcal{M}^1_{\theta_\alpha,\theta_\beta})$	$\max\limits_{\theta_\alpha,\theta_\beta} \rho(\mathcal{M}^2_{\theta_\alpha,\theta_\beta})$	Original	Modified
C for [6]				
1	0.4880	0.5	0.8889	0.8000
1/2	0.5885	0.6667	0.9412	0.8889
1/4	0.7044	0.8	0.9697	0.9412
10^{-1}	0.8398	0.9091	0.9877	0.9756
10^{-2}	0.9796	0.9901	0.9988	0.9980
10^{-3}	0.9979	0.9990	0.9999	0.9998

TABLE 1. Comparison of smoothing rates $\max_{\theta_\alpha,\theta_\beta} \rho(\mathcal{M}_{\theta_\alpha,\theta_\beta})$ of the line Gauss-Seidel (GS) iteration (17) with those of Chambolle's iteration (Table 4.1 and 4.2 in [6]).

and

$$\mathcal{M}^2_{\theta_\alpha,\theta_\beta} = \left[\begin{array}{cc} \mu h^2 + 2 - 2\cos\frac{2\pi I\theta_\alpha}{n} & 1 - e^{\frac{2\pi I\theta_\alpha}{n}} - e^{\frac{-2\pi I\theta_\beta}{n}} + e^{\frac{2\pi I(\theta_\alpha-\theta_\beta)}{n}} \\ 1 - e^{\frac{-2\pi I\theta_\alpha}{n}} & \mu h^2 + 2 - e^{\frac{-2\pi I\theta_\beta}{n}} \end{array} \right]^{-1}$$
$$\cdot \left[\begin{array}{cc} 0 & 0 \\ e^{\frac{2\pi I\theta_\beta}{n}} - e^{\frac{2\pi I(\theta_\beta-\theta_\alpha)}{n}} & e^{\frac{2\pi I\theta_\beta}{n}} \end{array} \right].$$

It is easy to see that $\mathcal{M}^1_{\theta_\alpha,\theta_\beta}$ is independent of h, whereas $\mathcal{M}^2_{\theta_\alpha,\theta_\beta}$ is not. If we assume that the image domain Ω is $(0,1)^2$, as we mentioned in the beginning of Section 3.1, increasing the image resolution has the same effect on $\mathcal{M}^2_{\theta_\alpha,\theta_\beta}$ as reducing μ (for fixed h).

When $|\vec{p}^{\,k+1}_{i,j}|_{l^2} > \alpha$, then the γ-terms in (13) and (14) are positive instead of zero, which will increase the diagonal elements in the first matrix in $\mathcal{M}^i_{\theta_\alpha,\theta_\beta}$, $i = 1,2$, and, as a result, the smoothing rate improves, i.e., it becomes smaller.

The amplification matrices of the relaxation in x-direction and the one for the y-direction are related by

$$\mathcal{M}^y_{\theta_\alpha,\theta_\beta} = \left[\begin{array}{cc} 0 & 1 \\ 1 & 0 \end{array} \right] \cdot \mathcal{M}^x_{\theta_\alpha,\theta_\beta} \cdot \left[\begin{array}{cc} 0 & 1 \\ 1 & 0 \end{array} \right],$$

showing that the smoothing rate of the y-line Gauss-Seidel iteration is equal to that of the x-line Gauss-Seidel iteration.

In Table 1 we list the smoothing rate $\max_{\theta_\alpha,\theta_\beta} \rho(\mathcal{M}_{\theta_\alpha,\theta_\beta})$ of the line Gauss-Seidel smoother for the nonlinear equations (13) and (14), respectively. Comparing the results for the two models, we see that when using the line Gauss-Seidel iteration for (13) we obtain a better smoothing rate than for (14). As we mention in Section 2, the dual regularization parameter μ is related to a (local) smoothing of the total variation. It therefore has the same effect as $C := |\nabla(\text{div}\vec{\omega}^k - \frac{z}{\alpha})|_{l^2} = |\nabla u^k|_{l^2}$ in the Fourier analysis of the multigrid method proposed in [6]. Hence, we are able to compare the smoothing rate of our method with the one in Table 4.1 of [6] and find that our method provides a better smoothing rate. Even when compared with the improved method in Table 4.2 of [6] our method still exhibits a smaller rate.

3.2. **Restriction and interpolation.** For the sake of completeness, in this section we provide our choices for the restriction and interpolation operations in our multi-grid algorithm. These operators are merely standard in the multigrid literature.

For the prolongation operator $\mathcal{P}_{l+1}^l$ we use bilinear interpolation [32, 27], i.e.,

$$\vec{p}_{2i-1,2j-1}^{\,l} = \vec{p}_{i,j}^{\,l+1}, \qquad\qquad 1 \le i \le m^{l+1} - 1, 1 \le j \le n^{l+1} - 1$$

$$\vec{p}_{2i-1,2j}^{\,l} = \frac{1}{2}(\vec{p}_{i,j}^{\,l+1} + \vec{p}_{i,j+1}^{\,l+1}), \qquad\qquad 1 \le i \le m^{l+1} - 1, 1 \le j \le n^{l+1} - 2$$

$$\vec{p}_{2i,2j-1}^{\,l} = \frac{1}{2}(\vec{p}_{i,j}^{\,l+1} + \vec{p}_{i+1,j}^{\,l+1}), \qquad\qquad 1 \le i \le m^{l+1} - 2, 1 \le j \le n^{l+1} - 1$$

$$\vec{p}_{2i,2j}^{\,l} = \frac{1}{4}(\vec{p}_{i,j}^{\,l+1} + \vec{p}_{i,j+1}^{\,l+1} + \vec{p}_{i+1,j}^{\,l+1} + \vec{p}_{i+1,j+1}^{\,l+1}), \quad 1 \le i \le m^{l+1} - 2, 1 \le j \le n^{l+1} - 2,$$

where $\vec{p}^{\,l+1} \in \mathbb{R}^{2(m^{l+1}+1) \times (n^{l+1}+1)}$. The restriction operator $\mathcal{R}_l^{l+1}$ is the adjoint of $\mathcal{P}_{l+1}^l$, i.e., the full weighting operator [32, 27] defined by

$$\vec{p}_{i,j}^{\,l+1} = \frac{1}{16}(\vec{p}_{2i-2,2j-2}^{\,l} + 2\vec{p}_{2i-2,2j-1}^{\,l} + \vec{p}_{2i-2,2j}^{\,l} + 2\vec{p}_{2i-1,2j-2}^{\,l} + 4\vec{p}_{2i-1,2j-1}^{\,l} + 2\vec{p}_{2i-1,2j}^{\,l}$$
$$+ \vec{p}_{2i,2j-2}^{\,l} + 2\vec{p}_{2i,2j-1}^{\,l} + \vec{p}_{2i,2j}^{\,l}), \qquad 1 \le i \le m^{l+1} - 1, 1 \le j \le n^{l+1} - 1$$

with homogenous Dirichlet boundary condition.

3.3. **Initialization.** In our smoothing steps, for solving the nonlinear equations (13) and (14) we utilize a semi-smooth Newton method, which converges superlin-early provided that the initial value $\vec{p}^{\,1}$ is sufficiently close to the solution. In our numerics we found the following initialization, which is related to the unconstrained version of the dual problem, to work well: Consider problem (1) with $K = id$ and the associated Euler-Lagrange equation, which formally leads to

$$\bar{u} - z = \text{div}\,\vec{p}$$

with $|\vec{\bar{p}}|_{l^2} \le \alpha$ in Ω. Disregarding the pointwise constraint on $\vec{\bar{p}}$, we initialize our algorithm by $\vec{p}^{\,1} = \nabla\bar{q}$, where $\bar{q} = \bar{q}(\bar{u})$ solves

$$-\Delta q = z - \bar{u} \text{ in } \Omega,$$

$$\frac{\partial q}{\partial \vec{n}} = 0 \text{ on } \partial\Omega.$$

Note that $\vec{p}^{\,1}$ satisfies $\text{div}\,\vec{p}^{\,1} = \bar{u} - z \in L^2(\Omega)$ and $\vec{p}^{\,1} \cdot \vec{n} = 0$ on $\partial\Omega$. Thus, $\vec{p}^{\,1} \in \mathbf{H}_0(\text{div})$; see Theorem 1 for the notations. In practice, we replace $\bar{u}$ by $\tilde{u}$, a smoothed version of z obtained from a convolution of Gaussian kernel and z. In our numerical experiments, we use the 7-by-7 Gaussian kernel with standard variation 1. Then, we set $\vec{p}^{\,1} := \nabla q(\tilde{u})$ yielding $z - \tilde{u} = \text{div}\,\vec{p}^{\,1}$ in Ω and $\vec{p}^{\,1} \cdot \vec{n} = 0$ on $\partial\Omega$.

In Table 2 we compare the above initialization strategy with an initialization uti-lizing the full multigrid approach (i.e. coarse-to-fine mesh refinement only), which is a popular choice in the literature. For a detailed discussion concerning the per-formance of both initialization we refer to the next section.

4. **Numerical results.** In this section, we provide numerical results for studying the behavior of our nonlinear multigrid method with respect to its image denoising capabilities and the speed of convergence. Furthermore, we compare our method with another nonlinear multigrid method based on Chambolle's iteration proposed in [6], the CCC-method for short. In all of our experiments reported on below the image intensity range is [0,255]. Unless otherwise specified in our nonlinear
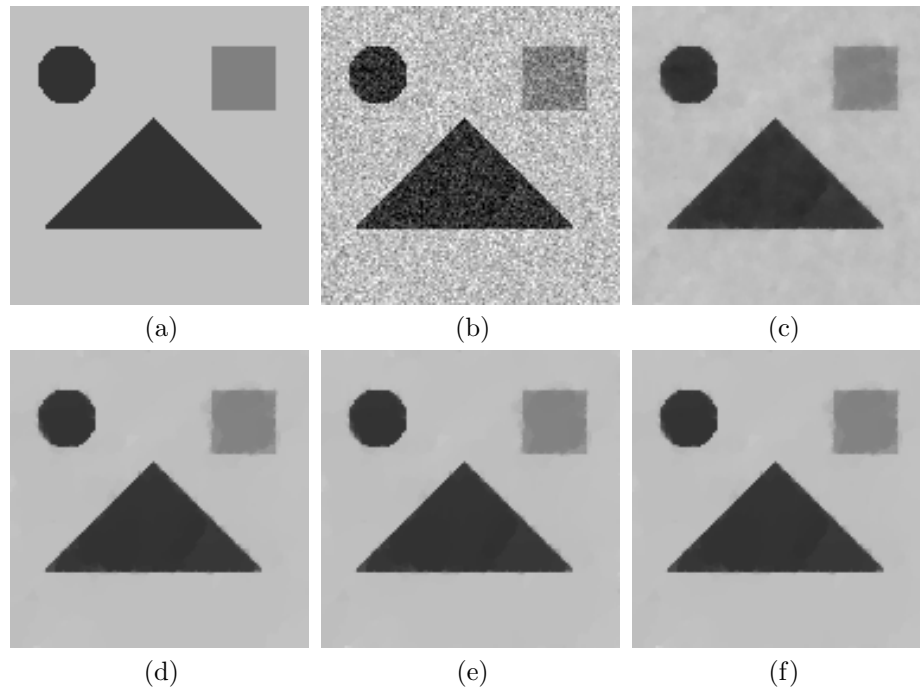
FIGURE 1. Results of the method proposed in [6] and our method for restoring the image corrupted by Gaussian noise with $\sigma = 0.1$: (a) original image, (b) noisy image, (c) by the CCC method in [6] with $\beta = 10^{-2}$, (d) by the CCC-method in [6] with $\beta = 10^{-6}$, (e) by our method for the model (10), (f) by our method for the model (9).

multigrid method, we utilize five complete line Gauss-Seidel iterations, i.e., one x-line and one y-line Gauss-Seidel iteration, as pre-smoothing and post-smoothing on each level, i.e., $\nu_1 = 10$ and $\nu_2 = 10$. The algorithm is stopped as soon as the initial residual is reduced by a factor of 10^{-7}. For comparing fairly, we run the CCC-method under the same parameter selections for (ν_1, ν_2, α) and the same stopping conditions.

Example 1. In the first experiment, we use a simple 127-by-127 gray-level image, which is corrupted by Gaussian white noise with noise level $\sigma = 0.1$; see Figure 1. In Table 2 we list the number of cycles of nonlinear multigrid for reaching the stop criterion (denoted by k), the CPU time (in seconds), and the PSNR-values [2]. Although both methods denoise the image by solving the dual problem of the total variation model (1), they utilize different ways to overcome the difficulty from the constraint in (6). In the CCC-method, a fixed point type projection scheme is utilized within a semi-implicit time-marching scheme [4] used as the smoother. In order to avoid a smoothing rate close to 1 in homogeneous regions (where the gradient is (close to) zero), the CCC-method uses the smoothing approximation $\int_\Omega \sqrt{|\nabla u|_{l^2}^2 + \beta}$ with $\beta > 0$, to avoid the non-differentiable TV-term. As $\beta \downarrow 0$, the solution of the CCC-method approaches the one of (1). Moreover, the larger β becomes the better the CCC-method converges. It is however known (compare

Method		k	CPU time (s)	PSNR (dB)
Method in [6]	$\beta = 10^{-2}$	17	4.47	33.2143
	$\beta = 10^{-4}$	68	15.70	34.6701
	$\beta = 10^{-6}$	147	35.16	34.8515
	$\beta = 10^{-8}$	177	48.50	34.8606
[1] Our method from (10)		12	112.54	34.7986
[1] Our method from (9)		14	115.87	34.7945
Our method from (10)		1	12.35	34.8008
Our method from (9)		1	11.34	34.8000

[1](The full multigrid algorithm is utlized as initialization.)

TABLE 2. Comparison of different methods for denoising the image in Figure 1.

plots (c) and (d) of Figure 1) that the quality of the reconstruction degrades for large values of β.

In the following experiments, we set $\beta = 10^{-6}$ in the CCC-method, which corresponds to the case of $C = 10^{-3}$ in the local Fourier analysis in the Tables 4.1 and 4.2 of [6]. Since the CCC-method relies on the formulation (2) and uses an iterative fixed point algorithm as a smoother, it leads to a very slow convergence for high precision requirements. From Table 2 we find that with $\beta = 10^{-6}$ and 10 pre- and post-smoother iterations the CCC-method still needs almost 150 nonlinear multigrid cycles. However, our method based on a penalization/regularization technique to obtain the unconstrained minimization problems (10) and (9), respectively, uses a line Gauss-Seidel-semismooth-Newton method as a smoother, where the pertinent generalized Newton iteration converges locally at a superlinear rate. It is readily seen that the solutions of (10) or (9) approximate the solution of the total variation model (1) as $\mu \downarrow 0$ and $\gamma \uparrow \infty$. Hence, in our experiments we set $\mu = 10^{-3}$ ($\mu h^2 = 10^{-3}$ for (9)) and $\gamma = 10^{12}$ for our method. In this case, with the initialization introduced in Section 3.3, after one cycle of the nonlinear multigrid, our method reaches the stopping condition. Although in each cycle our method needs more computations, with significantly less iterations it is more efficient. As mentioned earlier, we also provide the results obtained when our method is equipped with the full multigrid initialization. Due to the good approximation quality of the initial point resulting from the new strategy introduced in Section 3.3 and the fast (i.e. superlinear) local convergence of the semismooth Newton iteration, our method requires less nonlinear multigrid cycles and, hence, less CPU-time (compare the figure in the last four rows of Table 2). Because both methods (ours with $\mu = 10^{-3}$ and $\gamma = 10^{12}$ and the CCC-method with $\beta = 10^{-6}$) solve the total variation model (1), from Figure 1 (d), (e) and (f) and the PSNR-values listed in Table 2 we see that the obtained results are very similar.

Example 2. In this example, we compare our method with the CCC-method when restoring the degraded image with the different image resolution. This allows us to study "mesh-(in)dependence" (i.e., resolution (in)dependence) effects. It is interesting that our semismooth Newton solver in the smoothing step is known to be mesh independent as it admits a function space analysis [17, 18], whereas the fixed point iteration of [4] does not have such a behavior.

size	Our method for (10)			Our method for (9)			The CCC-method in [6]		
	k	CPU	PSNR	k	CPU	PSNR	k	CPU	PSNR
127	1	12.01	33.9553	1	9.42	33.9539	100	29.64	33.9500
255	1	44.89	31.8672	1	37.97	31.8662	217	252.23	31.8640
511	1	188.35	31.2693	1	163.00	31.2728	185	1128.3	31.2701

TABLE 3. Comparison of different methods for denoising the image "Boat" with different size.

FIGURE 2. Test image "boat".

The test image is the 511-by-511 gray-level image "Boat", which can be download from [1] (see Figure 2 for reference) and which is corrupted by Gaussian white noise with a noise level of $\sigma = 0.1$. Then this image is resized to 255-by-255 and 127-by-127 by the nearest neighbor scheme. Notice that here we are not aiming at quality aspects of downsampling, which would require an appropriate combination of low-pass filtering and sub-sampling, rather we study algebraic aspect of mesh independence of an iterative solver. As the total variation model (1) is independent on the image resolution for fixed image domain Ω, we adjust α based on the restoration runs for the 127-by-127 image by selecting the α-value giving the best PSNR value. This way lead us to choose $\alpha = 0.04$. This α-value is then kept for restoring the other images at different resolutions. The approximation chosen in the CCC-method exhibits a resolution dependence, i.e. it always takes pixel distance 1 whereas our method chooses $1/n$, where $n-1$-by-$n-1$ is the actual image resolution). Thus, instead of keeping the same α when restoring the images at different resolutions, we need set $\alpha = 0.04(n+1)$. In Table 3 we list the number of cycles of the nonlinear multigrid methods for reaching the stopping condition (denoted by k), the CPU time (in seconds), and the PSNR-values for restoring the images with different image resolutions. We can see that our method reaches the stopping condition within one cycle of the nonlinear multigrid scheme, which is much less than for the CCC-method. As expected, the PSNR-values of the restored images are very similar, which is due to both methods solving the same total variation model.

Example 3. Using the 127-by-127 test image of Example 2, we study the convergent rates of our methods and the CCC-method. In order to study the influence of the smoother on the convergent rate, we set the numbers of both the pre- and post-smoothing steps equal to 30, and, for the ease of accessibility, calculate the ratios of the residual after $k + 2$ cycles and the one after k cycles for $k = 1, 3, \ldots, 13$;

| | residual$_{k+2}$/residual$_k$ | | |
k	Our method for (10)	Our method for (9)	The CCC-method in [6]
1	0.3075	0.3548	0.1842
3	0.4690	0.5994	0.4377
5	0.4983	1.1289	0.5728
7	0.4937	0.7484	0.6597
9	0.4873	0.7147	0.7186
11	0.4855	0.8024	0.7599
13	0.5102	0.8128	0.7883

TABLE 4. Comparison of different methods for convergent rate.

see the results in Table 4. We find that the ratio obtained by our method based on (10) stays around 0.5. But the ratio of the CCC-method increases quickly as the multigrid cycle continues. Therefore, the CCC-method quickly slows down which is a clear drawback when rather high precision is needed. This behavior can be attributed to the linear convergent rate of iteration of [4] in the smoothing steps. However note that the ratios of our method based on (9) are not always less than 1. This is due to the fact that the identity operator coming from the L^2-norm regularization in (9) does not yield a smoothing operation (in contrast to the Laplacian obtained from the regularization in (10)). This adversely affects the multigrid scheme.

5. **Conclusion.** Nonlinear multigrid methods for total variation based image denoising are challenged by the non-smooth character of the Euler-Lagrange equations, or better differential inclusion, associated with the pertinent minimization problem. An appropriate numerical treatment depends on a reformulation of the PDE system as a non-smooth equation and the choice of an appropriate smoother. For this purpose and due to the type of the resulting PDE system, a dual formulation of the TV-problem appears better suited than a purely primal formulation, since the latter suffers from oscillatory and even discontinuous coefficients in the partial differential operator. The dual problem involves a vector-field (rather than a scalar-value function as in the primal problem) with a PDE-operator with a component-dependent strong coupling. The latter requires appropriate smoothing in the multigrid scheme. In our case, a line Gauss-Seidel smoother turns out to be efficient yielding satisfactory smoothing rates. In addition, the line smoother has to operate on non-differentiable, but still semismooth, equations. There are several options for solving these non-smooth problems like Chambolle's iteration or semismooth Newton methods. The mesh independence properties of semismooth Newton solvers make the latter solver class appealing. In our numerics we found that while the number of multigrid cycles deteriorates for Chambolle's method as the image resolution is refined, our semismooth Newton iteration based multigrid cycle remains stable.

<div align="center">

REFERENCES

</div>

[1] USC-SIPI image database, In A. Weber, editor,
    http://sipi.usc.edu/services/database/Database.html, University of Southern California.
[2] A. Bovik, "Handbook of Image and Video Processing," Academic Press, 2000.
[3] A. Brandt, *Guide to multigrid development*, In "Multigrid Methods" (Cologne, 1981), volume **960** of Lecture Notes in Math., pages 220–312. Springer, 1982.

[4] A. Chambolle, *An algorithm for total variation minimization and application*, Journal of Mathematical Imaging and Vision, **20** (2004), 89–97,.

[5] A. Chambolle and P-L. Lions, *Image recovery via total variation minimization and related problems*, Numerische Mathematik, **76** (1997), 167–188.

[6] T. Chan, K. Chen and J. L. Carter, *Iterative methods for solving the dual formulation arising from image restoration*, Electronic Transactions on Numerical Analysis, **26** (2007), 299–311.

[7] T. F. Chan, G. H. Golub and P. Mulet, *A nonlinear primal-dual method for total variation-based image restoration*, SIAM J. Sci. Comput., **20** (1999), 1964–1977.

[8] Q. Chang and I-L. Chern, *Acceleration methods for total variation-based image denoising*, SIAM J. Applied Mathematics, **25** (2003), 982–994.

[9] K. Chen, "Matrix Preconditioning Techniques and Applications," Cambridge University Press, 2005.

[10] D. C. Dobson and C. R. Vogel, *Convergence of an iterative method for total variation denoising*, SIAM J. Numer. Anal., **34** (1997), 1779–1791.

[11] I. Ekeland and R. Témam, "Convex Analysis and Variational Problems," Classics Appl. Math. 28, SIAM, Philadelphia, 1999.

[12] C. Frohn-Schauf, S. Henn and K.Witsch, *Nonlinear multigrid methods for total variation image denoising*, Comput. Vis. Sci., **7** (2004), 199–206.

[13] E. Giusti, "Minimal Surfaces and Functions of Bounded Variation," Birkhäuser, Boston, 1984.

[14] W. Hackbusch, "Multigrid Methods and Applications," volume **4** of Springer Series in Computational Mathematics, Springer-Verlag, 1985.

[15] M. Hintermüller, K. Ito and K. Kunisch, *The primal-dual active set strategy as a semismooth newton method*, SIAM J. Optim., **13** (2002), 865–888.

[16] M. Hintermüller and K. Kunisch, *Total bounded variation regularization as bilaterally constrained optimization problem*, SIAM J. Appl. Math., **64** (2004), 1311–1333.

[17] M. Hintermüller and G. Stadler, *An infeasible primal-dual algorithm for total bounded variation-based inf-convolution-type image restoration*, SIAM Journal on Scientific Computing, **28** (2006), 1–23.

[18] M. Hintermüller and M. Ulbrich, *A mesh-independence result for semismooth Newton methods*, Math. Program., **101** (2004), 151–184.

[19] P. J. Huber, *Robust regression: Asymptotics, conjectures, and Monte Carlo*, Annals of Statistics, **1** (1973), 799–821.

[20] S. Osher, M. Burger, D. Goldfarb, J. Xu and W. Yin, *An iterative regularization method for total variation-based image restoration*, SIAM Multiscale Model. and Simu., **4** (2005), 460–489.

[21] L. Rudin, "MTV-Multiscale Total Variation Principle for a PDE-Based Solution to Nonsmooth Ill-Posed Problem," Technical report, Cognitech, Inc. Talk presented at the Workshop on Mathematical Methods in Computer Vision, University of Minnesota, 1995.

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

[23] D. Strong and T. Chan, "Spatially and Scale Adaptive Total Variation Based Regularization and Anisotropic Diffusion in Image Processing," Technical report, UCLA, 1996.

[24] D. Strong and T. Chan, *Edge-preserving and scale-dependent properties of total variation regularization*, Inverse Problems, **19** (2003), 165–187.

[25] X.-C. Tai, "Rate of Convergence for Some Constraint Decomposition Methods for Nonlinear Variational Inequalities," Numerische Mathematik, 2003.

[26] X.-C. Tai, B. Heimsund and J. Xu, *Rate of convergence for parallel subspace correction methods for nonlinear variational inequalities*, In "Thirteenth International Domain Decomposition Conference" (Barcelona, Spain, 2002).

[27] U. Trottenberg, C. W. Oosterlee and A. Schüller, "Multigrid," Academic Press, 2001.

[28] P. S. Vassilevski and J. G. Wade, *A comparison of multilevel methods for total variation regularization*, Electron. Trans. Numer. Anal., **6** (1997), 255–270. Special issue on multilevel methods (Copper Mountain, CO, 1997).

[29] C. R. Vogel, *A multigrid method for total variation-based image denoising*, In "Computation and Control, IV" (Bozeman, MT, 1994), volume **20** of Progr. Systems Control Theory, pages 323–331. Birkhauser Boston, 1995.

[30] C. R. Vogel and M. E. Oman, "Iterative Methods for Total Variation Denoising," SIAM Journal on Scientific Computing, 1996.

[31] C. R. Vogel, "Computational Methods for Inverse Problems," Frontiers Appl. Math. 23, SIAM Philadelphia, 2002.

[32] P. Wesseling, "An Introduction to Multigrid Methods," John Wiley and Sons, 1992.

E-mail address: k.chen@liverpool.ac.uk
E-mail address: yiqiu.dong@helmholtz-muenchen.de
E-mail address: hint@math.hu-berlin.de