Nonlinear Multilevel Schemes for Solving the Total Variation Image Minimization Problem*

Tony F. Chan† **Ke Chen**‡ and **Xue-Cheng Tai**§

Abstract

The gradient descent approach is the most widely used method in variational modeling of many image processing applications such as image restoration and segmentation. While a user is likely to be content with results obtained after a few time steps, the gradient descent approach can be quite slow in achieving convergence. Among fast iterative solvers, multilevel methods offer the potential of optimal efficiency. This paper first reviews a class of efficient numerical methods for the variational model and then presents our recent work on developing optimization multigrid methods. Advantages of the proposed algorithms over previous results are presented.

Keywords Image restoration, total variation, regularization, subspace correction, fast multilevel solvers.

AMS subject class: 68U10, 65F10, 65K10.

Contents

*To appear: Proc. 1st Int. Conf. PDE-Based Image Processing and Related Inverse Problems, eds. T. F. Chan, X.-C. Tai, K. A. Lie and S. Osher, 2006, Springer-Verlag.

†Department of Mathematics, University of California, Los Angeles, CA 90095-1555, USA. Email: chan@ipam.ucla.edu. Web: http://www.math.ucla.edu/∼chan.

‡Department of Mathematical Sciences, University of Liverpool, Peach Street, Liverpool L69 7ZL, UK. Email: k.chen@liverpool.ac.uk. Web: http://www.liv.ac.uk/∼cmchenke. (For correspondence)

§Department of Mathematics, University of Bergen, Bergen, Norway. Email: xue-cheng.tai@uib.no. Web: http://www.mi.uib.no/∼tai.

1 Introduction

The purpose of this paper is to address the fast solution of a variational model for image processing. To concentrate on the main ideas we consider the standard total variation (TV) based variational model which was proposed by Rudin-Osher-Fatemi (ROF) [55] and studied by various researchers [1, 72, 73, 6, 14, 43]. Other problems are equally important [18, 16, 70, 71]. We remark that improved models have recently been proposed; see [8, 13, 21, 22, 57] and references therein. Our discussion should be applicable to these new models.

The ROF TV model [55] solves the following minimisation problem

$$\min_u \int_\Omega \left(\alpha |\nabla u| + \frac{1}{2}(u-z)^2 \right) dxdy, \tag{1}$$

where $z = z(x,y) \in \mathbb{R}^2$ is an observed image (in practice only a discrete matrix z of $z(x,y)$ is given) that requires restoration, $u = u(x,y)$ will be the restored image, $\alpha > 0$ is a regularization parameter that is necessary for ensuring uniqueness of the inverse problem of image restoration, Ω may be taken as the unit square and $\nabla u = (u_x, u_y)$ so $|\nabla u| = \sqrt{u_x^2 + u_y^2}$. The Euler-Lagrange equation for (1) is

$$-\alpha \nabla \cdot \left(\frac{\nabla u}{|\nabla u|} \right) + u - z = 0, \tag{2}$$

which is a nonlinear partial differential equation (PDE), also known as a curvature equation [52, 77]. One can observe that the 'equivalence' assumes that $|\nabla u| \neq 0$ (which is not a reasonable assumption) while problem (1) is well posed regardless $|\nabla u| \neq 0$ or not. To overcome this 'minor' problem, one normally solves the following equation instead of (2)

$$-\alpha \nabla \cdot \left(\frac{\nabla u}{|\nabla u|_\beta} \right) + u - z = 0, \tag{3}$$

where $|\nabla u|_\beta = \sqrt{|\nabla u|^2 + \beta}$ for some small $\beta > 0$. This equation may be viewed as the Euler-Lagrange equation for the modified problem of (1):

$$\min_u \int_\Omega \left(\alpha |\nabla u|_\beta + \frac{1}{2}(u-z)^2 \right) dxdy. \tag{4}$$

The gradient descent approach proposes to solve, instead of the elliptic PDE (3), the parabolic PDE

$$u_t = \alpha \nabla \cdot \left(\frac{\nabla u}{|\nabla u|_\beta} \right) - (u - z), \tag{5}$$

where $u = u(x,y,t)$ will converge to the solution of (3) when $t \to \infty$, with $u(x,y,0) = z$. The advantage is that various explicit time-marching schemes may be used to solve (5) in a computationally convenient way [55, 52, 48, 71, 47]. For example, the explicit Euler scheme proceeds as follows

$$\frac{u^{k+1} - u^k}{\Delta t} = -\alpha \nabla \cdot \left(\frac{\nabla u^k}{|\nabla u^k|_\beta} \right) + u^k - z,$$

for $k \geq 0$ and $u^0 = z$. Note that if $\Delta t' = \alpha \Delta t$ can be large enough, at $k = 0$, the one-step scheme mimics the nonlinear diffusion type models [53, 42]

$$\frac{u^1 - u^0}{\Delta t'} = -\nabla \cdot \left(\frac{\nabla u^0}{|\nabla u^0|_\beta} \right).$$

As far as fast solvers are concerned, on a single level, the most robust method that we have tested for (3) is the Chan-Golub-Mulet (CGM) algorithm [25, 23] in the primal-dual pair (u, w)

$$\begin{cases} -\alpha\nabla\cdot w + u - z = 0, \\ w|\nabla u|_\beta - \nabla u = 0,\ \|w\|_\infty \le 1 \end{cases} \tag{6}$$

by introducing the new variable $w = \nabla u/|\nabla u|_\beta$ in a mixed formulation as in a mixed finite element method. However we shall be mainly concerned with multilevel methods in this paper for efficiently solving (1). Some numerical comparisons to this CGM algorithm are shown later on. One interesting observation of (6) is the following. Clearly eliminating w reduces it to the original PDE (3). However, if we try to eliminate u in the second equation by using $u = z + \alpha\nabla w$ from the first equation, we obtain (noting $\nabla\cdot w = \text{div } w$)

$$-\nabla\left(\alpha\text{div } w + z\right) + |\nabla\left(\alpha\text{div } w + z\right)|_\beta w = 0$$

which reduces to the same dual formulation [13] for $\beta = 0$. Therefore, if letting $\lambda = \alpha$, the two formulations reproduce each other via their dual variables: $w = -\mathbf{p}$. (Refer to §2 below.)

2 Review of unilevel methods for solving the TV formulation

There is a rather rich literature of related work towards efficient solution of the denoising model (1). Here we give a brief review before we turn to multilevel methods in the next section. Each method attempts to address the non-smoothness and nonlinearity in (1) in a different way.

2.1 The dual formulation

The primal formulation (1) may be indirectly solved via a dual formulation [13, 34]. Define the dual variable $\mathbf{p} = (p_1, p_2)$ s.t. $u = z - \lambda\text{div } \mathbf{p}$. Then the dual formulation takes the form

$$\min_{\mathbf{p}\in Y} \|z - \lambda\text{div } \mathbf{p}\|, \quad |\mathbf{p}_{i,j}|^2 \le 1, \forall\ i, j = 1, \ldots, n \tag{7}$$

where Y is the Euclidean space as specified in [13]. The above problem may be equivalently solved [13] from

$$-\left[\nabla(\lambda\text{div } \mathbf{p} - z)\right]_{i,j} + \left|[\nabla(\lambda\text{div } \mathbf{p} - z)]_{i,j}\right|\mathbf{p}_{i,j} = 0,$$

in which one can observe that the nonlinearity is now present in the 'source' term.

The dual formulation for a related problem to (1)

$$\min_u \int_\Omega \left(\alpha|\nabla u| + \frac{1}{2}(Ku - z)^2 + \frac{\beta}{2}|u|^2\right) dxdy \tag{8}$$

is studied in [35]. Such a formulation leads to a similar dual optimization problem to (7) except that the new dual variable is bilaterally constrained.

2.2 The modified total variation method

If $|\nabla u| \neq 0$, model (1) is easy to solve. For the general case, one idea (quite different from (3)) is to modify the TV-norm [14, 39, 54] to exclude all sets where $|\nabla u| = 0$. As compensation, regularization over these sets is done with smooth norms such as with $|\nabla u|^2$. More specifically in [14], the following problem is considered:

$$\min_u \int_\Omega \frac{1}{2}(u - z)^2 dxdy + \alpha\left(\int_{|\nabla u|>\delta} |\nabla u| dxdy + \frac{1}{\delta}\int_{|\nabla u|\le\delta} |\nabla u|^2 dxdy\right)$$

for a given $\delta > 0$. Although the modified problem is still non-smooth, it is formally differentiable.

Another idea of modifying the TV model is to solve the following minimization problems [5]

$$\min_u \int_\Omega \left(\frac{1}{2}(u-z)^2 + \frac{\alpha}{s}|\nabla u|^s \right) dxdy$$

for $1 \leq s \leq 2$ (see [22, 57] for other models of this type). Numerical solution methods for this model are proposed in [39], where the model was found to give some optimal performance with $s = 1.1$ or 1.2. Incidentally the work of [17] on a different image problem recommends the choice of $s = 1.3$ in similarly modifying the TV norm.

2.3 The active set method

This is a discrete approach [37, 12, 38] for solving the Euler-Lagrange equation of problem (8) which is a related idea to the above modified method i.e. treat inactive sets $|\nabla u| = 0$ differently from active sets $|\nabla u| > 0$. For pixels in the active sets, the problem is smooth while for others, a modified smooth problem is solved by ignoring the TV term.

2.4 The tube method

The discrete solution of (1) can be shown (in one dimension) to lie in a tube, bounded by two known linear splines [36]. As this solution can be interpreted as a taut string in the tube, the taut-string algorithm from statistics can solve the TV model in two dimensions [36]:

$$\begin{cases} \text{Solve } \Phi \text{ from} & \Delta\Phi = z, \quad \frac{\Phi}{n} = 0 \\ \text{Define the vector quantity} & F_z = (F_1, F_2) = \nabla\Phi \\ \text{Solve for two taut-string functions } \omega_1, \omega_2 & \text{from} \\ \min_{\omega_i} \int_\Omega \sqrt{1 + |\nabla\omega_i|^2} dxdy & \text{subject to the tube domain:} \\ F_1 - \alpha \leq \omega_1 \leq F_1 + \alpha, & F_2 - \alpha \leq \omega_2 \leq F_2 + \alpha. \end{cases}$$

Although it may appear that such a formulation is no easier than solving (1), the above method is in fact more amenable to numerical implementation than (1) because the new problem is smooth. Here $\omega = (\omega_1, \omega_2)$ acts like a dual variable but, different from [23], no β is required for (1). Moreover a fixed-point algorithm (outer-loop) is suggested [36] to solve the main nonlinear optimization step. See [58] for connections to bounded variation regularization.

2.5 The second-order cone programming method

To derive a general method for solving (1), we note that an alternative approach is to consider

$$\min_u \int_\Omega |\nabla u| dxdy, \quad s.t. \;\; u + v = z, \;\; \int_\Omega |v|^2 dxdy \leq \sigma^2,$$

where σ^2 is a variance of the noise level in z. In particular, the main TV minimization is a non-smooth problem whose discrete form may be denoted by minimizing

$$T(u_{1,1}, u_{1,2}, \ldots, u_{n,n}) = \sum_{i,j=1}^{n} \sqrt{(u_{i,j} - u_{i+1,j})^2 + (u_{i,j} - u_{i-1,j})^2}$$

subject to the usual adjustment near the image boundaries.

The key observation made in [33] on treating the non-smooth discrete TV-term is the following: the inequality

$$\sqrt{(u_{i,j} - u_{i+1,j})^2 + (u_{i,j} - u_{i-1,j})^2} \leq t_{i,j}$$

defines a well-known second-order cone in optimization theory. The established interior point methods may be used to solve problems with such cone constraints. Therefore the proposal is to replace the minimization of T by minimizing the following equivalent merit function $\widetilde{T}$

$$\widetilde{T}(t_{1,1}, t_{1,2}, \ldots, t_{n,n}) = \sum_{i,j=1}^{n} t_{i,j}, \quad s.t. \quad \sqrt{(u_{i,j} - u_{i+1,j})^2 + (u_{i,j} - u_{i-1,j})^2} \leq t_{i,j} \ \forall \ (i,j).$$

Further the second-order cone programming (SOCP) method [33] is the following

$$\begin{cases} \min\limits_{t_{1,1}, t_{1,2}, \ldots, t_{n,n}} & \sum\limits_{i,j=1}^{n} t_{i,j} & \\ \text{s. t.} & u_{i,j} + v_{i,j} = z_{i,j}, & \text{for } i,j = 1, \ldots, n \\ & -X_{i,j} + (u_{i+1,j} - u_{i,j}) = 0, & \text{for } i = 1, \ldots, n-1; j = 1, \ldots, n \\ & -Y_{i,j} + (u_{i,j+1} - u_{i,j}) = 0, & \text{for } i = 1, \ldots, n; j = 1, \ldots, n-1 \\ & X_{n,k} = Y_{k,n} = 0, & \text{for } k = 1, \ldots, n \\ & \sqrt{X_{i,j}^2 + Y_{i,j}^2} \leq t_{i,j}, & \text{for } i,j = 1, \ldots, n \\ & \sqrt{v_{1,1}^2 + v_{1,2}^2 + \ldots + v_{n,n}^2} \leq \sigma. & \end{cases}$$

Here the extra variables $X_{i,j} = (\frac{u}{x})_{i,j}$ and $Y_{i,j} = (\frac{u}{y})_{i,j}$ (and $u_{i,j}$ may be eliminated to leave $4n^2$ unknowns). See also [76]. To generate a sequence of interior points, an inner loop of iterations is introduced after putting sparsity into consideration [33]. The overall complexity is $O(N\sqrt{N})$ with $N = n^2$ for an $n \times n$ image.

2.6 The additive operator splitting method

Although we have remarked that the time-marching method is widely used (but slow), improved variants also exist. We wish to highlight the semi-implicit approach of an additive operator splitting (AOS) method which is based on classical ideas of dimensional splitting and alternating directions. The AOS method was originally proposed in [44, 45] for Navier-Stokes equations and it was rediscovered independently later in [74] for nonlinear diffusion equations. Different properties of the AOS methods have also been studied intensively recently in [30, 32, 31, 3].

Denote the discretized version of equation (5) from a semi-implicit time-marching scheme by (in matrix vector form)

$$\frac{u^{k+1} - u^k}{\Delta t} = \sum_{\ell=1}^{2} A_\ell(u^k) u^{k+1} \quad \text{i.e.} \quad u^{k+1} = \left(I - \Delta t \sum_{\ell=1}^{2} A_\ell(u^k) \right)^{-1} u^k$$

where A_ℓ denotes the nonlinear coefficient matrix from discretization along the ℓ-coordinate direction. This is well-known. However the above inversion might not be a cheap operation if a direct (or even an iterative) method is used.

The idea of [44, 45, 74] is to make an order $O(\Delta t)$ perturbation so that the new scheme

$$u^{k+1} = \frac{1}{2} \sum_{\ell=1}^{2} \left(I - 2\Delta t A_\ell(u^k) \right)^{-1} u^k$$

is still a semi-implicit method with no essential loss of accuracy but is much easier to solve. The inversion of $\left(I - 2\Delta t A_\ell(u^k)\right)^{-1}$ reduces to the solving of some three diagonal matrices over the lines parallel to the ℓ-coordinate direction, see [44, 45]. More importantly the modified scheme creates a discrete scale-space, see [74].

It should be remarked that there exist other anisotropic diffusion type models [42, 53] that are differential equations based (i.e. not minimisation based) and higher order models [46] for the same image restoration problem.

Although all above ideas might be generalized to a multilevel setting, such generalization work remains to be done. In the remainder of the paper, we shall focus on multilevel methods.

3 Review of a class of multigrid methods

As we know, multigrid methods build on two well-known (i.e. old) mathematical ideas: residual (defect) correction and coarse grid approximation. The modern multigrid methods were proposed in the 1970s [27, 66]. The method was casted into a unified framework of multilevel and multidomain subspace correction in the late 1980s (see [75, 65, 61] and the references therein). See [15, 68, 51] for some recent work and refer to [27] for implementation details.

3.1 Linear multigrid approaches

One of the earliest attempts to solve (3) can be seen in [15, 68, 69, 70, 26, 41, 2, 59] where a linear multigrid method is used in conjunction with a linearized PDE. Essentially at the current iteration with $\bar{u}$ (starting initially from $\bar{u} = z$ with the Neumann's boundary condition), multigrid methods are used as an inner (fast) linear solver for

$$-\alpha\nabla\cdot\left(\frac{\nabla u}{|\nabla\bar{u}|_\beta}\right)+u-z=0,$$

but the outer solver of repeating fixed point iterations may not converge very fast. There are also some other approaches using different linearization methods and solve the linearized problem by a multigrid technique.

3.2 The FAS nonlinear multigrid method

One of the well known multigrid method for nonlinear problem is the FAS (Full Approximation Storage) algorithm of Brandt [9, p.346]. The original FAS algorithm for a nonlinear equation

$$N(u)=f \tag{9}$$

needs to use a sequence of nested refined meshes $\mathcal{T}_1^h, \mathcal{T}_2^h, \cdots, \mathcal{T}_L^h$. Assume that $\mathcal{T}_1^h$ is the finest mesh and $\mathcal{T}_L^h$ is the coarsest mesh. For the FAS algorithm, the nonlinear equation (9) also needs to be approximated on the different meshes. Assume that equation (9) is approximated on $\mathcal{T}_k^h$ by

$$N_k(u)=f_k. \tag{10}$$

Thus, the real problem we need to solve is (10) for $k=1$.

Consider two successive meshes on levels $k, k+1$ – a fine and a coarse level. We use R_k^{k+1} to denote the standard restriction operator between $\mathcal{T}_k^h$ and $\mathcal{T}_{k+1}^h$. Let the current approximation on level k be u_k after some smoothing steps. The task is to find a correction quantity e_{k+1} so that $u_{k+1}=R_k^{k+1}u_k+e_{k+1}$ will be the new and improved approximation on the coarser mesh on level $k+1$. The FAS algorithm of [9, p.346] needs to solve the following equation on level $k+1$:

$$N_{k+1}(u_{k+1})=\bar{f}_{k+1}, \tag{11}$$

where $\bar{f}_{k+1}$ is computed recursively through

$$\bar{f}_{k+1}=N_{k+1}(R_k^{k+1}u_k)+R_k^{k+1}(\bar{f}_k-N_k(u_k)).$$

One just needs to use a linearized smoother for equation (11) to get u_{k+1} and the correction value in fact is

$$e_{k+1} = \bar{u}_{k+1} - R_k^{k+1} u_k.$$

For our nonlinear problem (1), the solution u and the data term z are non-smooth and have discontinuities. The coarse mesh problems (10) could not approximate the problem on the finest level. Thus it may not be appropriate to use them to find the correction values over the coarser meshes unless β is sufficiently big [29, 56, 7].

3.3 Nonlinear subspace correction (NSSC) methods

For linear elliptic problems, it is known that the traditional multigrid methods is the same as the subspace correction methods [75]. The subspace correction idea has been extended to nonlinear convex minimization problems in [65]. For constrained convex minimization problems, algorithms and convergence analysis are also available in [61, 63]. The essential ideas used for the nonlinear subspace correction (NSSC) methods in [65, 61, 63] can be traced back to [62, 64]; see also [24]. In the following, we shall outline the NSSC methods of [65, 61, 63] and show its differences from the FAS algorithm [9].

If we apply the NSSC of [65, 61, 63] to linear elliptic problems, it reduces to the standard multigrid method. For nonlinear problems, the essential idea of NSSC can be classified as in the following:

- The NSSC only uses the equation (10) on the finest mesh. It does not need to use the equation (10) over the coarser meshes.
- The NSSC method was formulated for finite element approximations. The functions over the coarser meshes are always regarded as a function defined on the finest mesh using the standard interpolation concept. For convex minimization problems, the corrections values need to minimize the cost functional over the finest mesh. Thus we do not need to construct cost functional over the coarser meshes.
- Nonlinear minimization problems with respect to a scalar which is the nodal value for the coarse mesh nodes need to be solved over all the coarse mesh nodes. We do not need to solve these scalar nonlinear minimization problems exactly, c.f. [62]. If proper linearization methods are used for these scalar minimization problems, the cost for NSSC per iteration can be $O(N)$ where N is the degree of freedom over the finest mesh. Otherwise, the cost is normally $O(N \log N)$ for the NSSC as all the subproblems need to be transformed to a problem over the finest mesh.

The NSSC in [65, 61, 63] was formulated for convex minimization problems. The algorithms can be extended to general nonlinear problems (1), but the convergence analysis may not be extended to (1) under general conditions. For a given reflexive Banach space V, a convex subset $K \subset V$ and a smooth convex functional $J : V \to R$, consider

$$\min_{v \in K} J(v), \qquad K \subset V, \tag{12}$$

In case $K = V$, then (12) is equivalent to (9) with $N(u) = \partial J(u)$. Here ∂J is the Gauteaux differential of J. Note that $N = \partial J$ is a nonlinear mapping which maps V to its dual space V^*. Assume now that we have generated a sequence of nested meshes $\mathcal{T}_k^h$. Let V_h be the finite element approximation space we shall use for (12) over the finest mesh. Then the discretized solution for (12) is the minimizer of

$$\min_{v \in V_h} J(v). \tag{13}$$

Let V_k be the finite element spaces over the meshes $\mathcal{T}_k^h$. Generally, the spaces V_k are spanned by some basis functions, i.e.

$$V_k = \mathrm{span}\big(\{\phi_i^k\}_{i=1}^{n_k}\big) = \sum_{i=1}^{n_k} V_i^k,$$

where

$$V_i^k = \mathrm{span}(\phi_i^k).$$

One essential idea of the NSSC is to regard V_h ($V_h = V_1$) as a decomposition as in the following:

$$V_h = \sum_{k=1}^{L}\sum_{i=1}^{n_k} V_i^k.$$

The NSSC is trying to use all the subspaces V_i^k to find the correction values. Given a current approximation u, the successive NSSC can be written as

- For $k = 1, 2, \cdots L$ and then for $i = 1, 2, \cdots n_k$:

$$\text{Find } c = \mathrm{argmin}_s J(u + s\phi_i^k), \text{ and update } u \text{ as } u := u + c\phi_i^k. \tag{14}$$

- End.

As $N(u) = \partial J(u)$ is the Gateaux differential of a convex functional, the nonlinear scalar minimization problem (14) is equivalent to finding c from

$$\langle N(u + c\phi_i^k), \phi_i^k\rangle = 0.$$

Here $\langle\cdot,\cdot\rangle$ denote the duality pairing between V and V^* in the continuous setting and it is the L^2 inner product for finite element functions in the discrete setting. Thus, the following algorithm can be used for general nonlinear problems (9) and it is equivalent to the algorithm given in (14) if $N(u) = \partial J$:

- For $k = 1, 2, \cdots L$ and then for $i = 1, 2, \cdots n_k$:

$$\text{Solve } c \text{ from } \langle N(u + c\phi_i^k), \phi_i^k\rangle = 0, \text{ and update } u \text{ as } u := u + c\phi_i^k. \tag{15}$$

- End.

Let $g(s) = \langle N(u + s\phi_i^k), \phi_i^k\rangle$, then (14) and (15) is to solve

$$g(c) = 0.$$

We normally do one step of gradient descent or Newton iteration. For differentiable J functionals, it is easy to see that $g'(s) = \langle \partial^2 J(u + s\phi_i^k)\cdot\phi_i^k, \phi_i^k\rangle$. For quasilinear problems, we can also use a Picard iteration. The choice of the approximate solver for $g(c) = 0$ depends on the problem. For some problems, it is possible to solve $g(c) = 0$ in a way which only has a cost of $O(N)$ flops per iteration. For (12), we shall give some details later about how to solve (14) and (15).

It is clear that the nonlinear function g depends on i and k. For general nonlinear problems, it is very important that we do not solve (14) and (15) exactly, but replace g by some approximations depending on the problem. A properly approximation for g and a proper implementation technique can improve the numerical efficiency rather a lot. It should be observed that the functions u and ϕ_i^k are regarded as functions defined on the finest mesh. The cost functional $J(u + s\phi_i^k)$ and $N(u + s\phi_i^k)$ shall be evaluated using the values of $u + s\phi_i^k$ over the finest mesh. The duality $\langle N(u + s\phi_i^k), \phi_i^k\rangle$ is an integration involving $N(u + s\phi_i^k)$ over the support set of ϕ_i^k.

For clarity of presentation, we have chosen V_i^k to be spanned by the basis functions ϕ_i^k. For some (linear or nonlinear) problems, it might be necessary to choose the subspaces V_i^k to be spanned by a few related basis functions. We shall not go into much detail about this here.

For several reasons, it is preferable to avoid the use of the coarse mesh equations to get the correction values. The first reason is differentiability. For some non-differentiable problems, the coarse mesh problems may not approximate the fine mesh problem. The analysis of NSSC needs differentiability of the cost functional, c.f. [65, 61, 63]. However, the algorithm given in (14) can be used even for non-differentiable problems. There are also some other problems that the "simple" coarse mesh equations fail to approximate the fine mesh equation. The well-known p-Laplace equation and the equations for convection diffusion process with a non-dominating diffusion term belongs to this class of problems.

3.4 FAS based multigrid methods for minimization

It is possible to use FAS algorithm for convex minimization problems as least when the minimization functional is differentiable. Several related approaches, c.f. [4, 60, 10, 49], tried to design coarse grid problems by using first order conditions (similar to using (2) to measure residuals). Specifically consider a typical setting of 2 levels: a fine level k and a coarse level $k+1$. The 'closeness' of the current approximation u_k on mesh $\mathcal{T}_k^h$ to the true minimizer is measured by its first order condition (i.e. via a discrete version of (2))

$$r_k = \partial J(u_k),$$

where we assume that J must be differentiable. Thus with such a residual information available, it is proposed in [60, 49] to use the following coarse grid solver

$$\min_{u_{k+1}} J(u_{k+1}) - g_{k+1}^T u_{k+1}$$

where

$$g_{k+1} = \partial J(R_k^{k+1} u_k) - R_k^{k+1} \partial J(u_k)$$

represents the residual information projected onto the coarse grid as in a nonlinear multigrid method. As we see, this is rather similar to the FAS algorithm.

3.5 NSSC method for (1)

In order to generalize the multilevel algorithm to optimisation, we have to discuss "local relaxation" algorithms: what is a local relaxation and are local relaxations sufficient for solving (1) as a numerical method? It turns out that a local relaxation for minimisation is simply a local minimisation and local relaxations are not sufficient for solving (1) because only local non-stationary minimizers are found i.e. local relaxations can get 'stuck' before reaching the global minimizer. For (1), Carter [11] appears to be one of the earliest to observe such 'stuck' minimizers and hence would not recommend local relaxations as a standalone method. This may be seen from Figure 1 where the observed image z is denoted by $*$, the global minimizer (using $\alpha = 4$) by $\bigcirc$ and the solution from local relaxation by $\square$; clearly the local non-stationary minimizer $\square$ got stuck as noted by [11]. (Note: *a local non-stationary minimizer is not a local minimizer as the latter is also the global minimizer.*) We remark that [11] proposed hybrid relaxation methods using both the primal and dual formulations, and other ideas to avoid using the primal relaxation alone. For general convex functions, the study of block relaxation can be found in [67], where the problem of 'stuck' minimizers is also discussed. Other nonlinear solvers for relaxation may be found in [39, 40, 50]; in particular, sophisticated optimization methods are tested in [39].

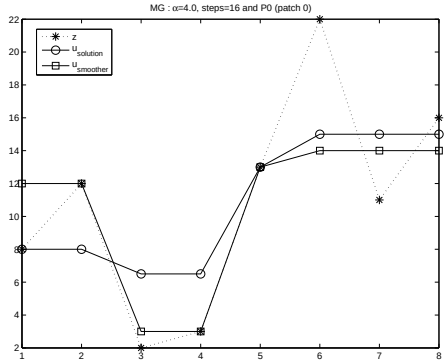

Figure 1: Example to show that relaxations alone □ are not good methods for solution ◯.

4 A piecewise constant NSSC algorithm for (1)

Instead of attempting other relaxation methods or formulations, a different idea was considered in [19] where global minimization was achieved through utilizing multilevels. Some theoretical analysis has also been given in [20]. In the following, we shall cast the standard approaches in [19, 20] as NSSC algorithms for piecewise constant approximation and their suggested approaches as new NSSC algorithms for (1).

Denote an observed image by $z \in \mathbb{R}^{n\times n}$ and let $n = 2^L$. The standard coarsening is used to define $L+1$ levels: $k = 1(\text{finest}), 2, \ldots, L, L+1(\text{coarsest})$. Take $\Omega = [0,1]\times[0,1]$ as the coarsest mesh, we divide each element of a coarse mesh element by connecting the four edge middle points to form four equal rectangles over the fine mesh. The gives us a nested sequences of meshes with uniform mesh sizes $h_k = 1/2^{k-1}$. The grid points for the finest mesh are $x_i = i/2^L, y_j = j/2^L$. Let $n_k = 2^{k-1}$ and $\{\tau_{i,j}^k\}_{i,j=1}^{n_k}$ be the rectangular finite elements for the mesh at level k. Then the functions $\phi_{i,j}^k$ given by

$$\phi_{i,j}^k = 1 \text{ on } \tau_{i,j}^k \text{ else } \phi_{i,j}^k = 0, \tag{16}$$

form a basis for the piecewise constant finite element space over the mesh of level k. On the finest level, the discretized minimization we shall consider is:

$$\min_u J_h(u) \tag{17}$$

where $J_h(u) = \alpha \sum_{i,j=1}^{n_k-1} \sqrt{|D_x^+ u_{i,j}|^2 + |D_y^+ u_{i,j}|^2 + \beta} + \frac{1}{2}\sum_{i,j=1}^{n}(u_{i,j} - z_{i,j})^2$. Here u denotes a piecewise constant function defined on the finest level, $u_{i,j}$ is its value over an element $\tau_{i,j}^k (k=1)$ and D_x^+, D_y^+ are the standard forward finite differences. This minimization problem is widely used for image denoising which normally works on a fixed mesh. We remark that (1) has been discretized by finite differences to give (17) so the function u may be constructed by any piecewise approximation (not restricted to piecewise constants). This is also the reason that the approximated equations on the coarser level are not appropriate to be used for the correction values.

We shall use the NSSC algorithm for solving the discretized problem (17). For this case, the algorithm given in (14) turns out to be:

Algorithm 1 (Piecewise constant NSSC algorithm)

- For $k = 1, 2, \cdots L+1$
  - For $i, j = 1, 2, \cdots n_k$:

$$\text{Find } c = \text{argmin}_s J(u + s\phi_{i,j}^k), \tag{18}$$

$$\text{and update } u \text{ as } u := u + c\phi_{i,j}^k. \tag{19}$$

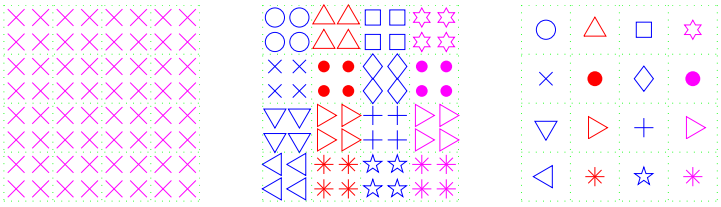

Figure 2: Illustration of the restriction process for a piecewise constant multigrid method from the fine 8×8 grid (left) to the coarse 4×4 grid (right). Here the middle plot shows the level 2 piecewise constants and each symbol denotes a separate constant.

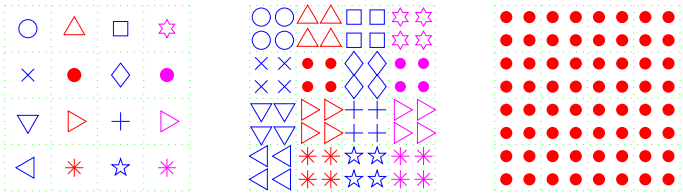

Figure 3: Illustration of the (inherent) interpolation process for a piecewise constant multigrid method from the coarse 4×4 grid (left) to the fine 8×8 grid (right). Here the middle plot shows the interpolated level 2 piecewise constants and each symbol denotes a separate constant.

  - End.

- End.

For $k = 2$, one element $\tau_{i,j}^k$ is split into 4 elements on the finest level. Thus, on the finest level, $s\phi_{i,j}^k$ takes the values:

$$\begin{bmatrix} s & s \\ s & s \end{bmatrix}.$$

For an element $\tau_{i,j}^k$ on a much coarser level, the value of $s\phi_{i,j}^k$ on the finest level looks like:

$$\begin{bmatrix} s & s & \cdots & s \\ \vdots & \vdots & \vdots & \vdots \\ s & s & \cdots & s \end{bmatrix}.$$

To illustrate the setup, we show the restriction process in Figure 2 and the interpolation process in Figure 3. Here each block represents a local constant patch. On the finest level $b = 1$, each pixel is adjusted for adding the best local constant which is the same process of a local minimization (as discussed). The patch size $b \times b$ may be made variable $b_i \times b_j$ if such a set $\{(k, \ell) \mid |u_{k,\ell} - u_{i,j}| < \epsilon\}$, containing indices for a $b_i \times b_j$ block, is non-empty at the current iterate.

In [19], a one step Richardson iteration is used as an approximate solver for (18). We refer to [19] for the details about how to solve (18) in an efficient manner.

5 A new piecewise constant NSSC algorithm with an adaptive subspace

In the last section, the standard multigrid subspaces for a piecewise constant finite element space are used. Due to the nature of problem (17), it was found that such a standard approach is not sufficient to achieve the global convergence. It turns out that using a new coarse mesh produced adaptively

during the iterations provides a solution [19, 20]. Given u defined on the finest mesh and a threshold constant γ, we say the two adjacent grid points (x_i, y_j) and (x_{i+1}, y_j) belong to the same patch if

$$|u_{i+1,j} - u_{i,j}| \leq \gamma.$$

The same is used to classify two points (x_i, y_j) and (x_i, y_{j+1}). In this way, all the grid points (x_i, y_j) can be grouped into a fixed number of patches depending on u and γ. Let $\{\Omega_i\}_{i=1}^{n_u}$ be the patches obtained using u and γ. We then define

$$\psi_i = 1 \text{ on } \Omega_i \text{ else } \psi_i = 0, \quad i = 1, 2, \cdots n_u. \tag{20}$$

We shall add the subspaces spanned by ψ_i to the decomposition for getting the correction values. With these subspaces added, the new NSSC type algorithm is:
Algorithm 2 (Adaptive piecewise constant NSSC algorithm)

- For $k = 1, 2, \cdots L + 1$
  - For $i, j = 1, 2, \cdots n_k$:

$$\text{Find } c = \text{argmin}_s J(u + s\phi_{i,j}^k), \text{ and update } u \text{ as } u := u + c\phi_{i,j}^k. \tag{21}$$

  - End.
  - For $i = 1, 2, \cdots n_u$:

$$\text{Find } c = \text{argmin}_s J(u + s\psi_i), \text{ and update } u \text{ as } u := u + c\psi_i. \tag{22}$$

  - End.
- End.

The new subproblems (22) are solved using similar approximate solvers. This algorithm has been explained and analysed in detail in [20].

Returning to our earlier remark on interpretation the approximation on the finest grid, it has been proven in [12] that piecewise constant finite element functions alone cannot be used to approximate the total variation of bounded variation functions. So our classification of the above algorithm as a NSSC with piecewise constants, useful for understanding the algorithm, is not precise because the minimization (finite difference) functional (17) used in this and the last section is not the total variation of the corresponding piecewise constant function.

6 A piecewise linear type multilevel algorithm

In this section, we shall explain how to use NSSC algorithm for piecewise linear finite element subspaces. For piecewise linear finite element spaces, we need to use triangular mesh over the different levels. The triangular meshes are produced from the rectangular meshes obtained in the last sections by divide each rectangle into two triangles using the diagonal of a negative slope. Let $\phi_{i,j}^k$ be a continuous function which is a linear function over each triangular element on the kth level satisfying

$$\phi_{i,j}^k(x_i, y_j) = 1 \text{ and } \phi_{i,j}^k(x_l, y_m) = 0, l \neq i, m \neq j. \tag{23}$$

Then, $\{\phi_{i,j}^k\}_{i,j=1}^{n_k+1}$ forms a basis for the piecewise linear finite element space over level k. The number n_k is defined as before. Assume that for the given image z on the finest level with $n \times n$ pixels, the desirable image u (discrete) uniquely defines a piecewise linear function u in Ω.

If we use NSSC algorithm for the subspaces spanned by all the basis functions over all the levels as given in (23), we will get:
Algorithm 3: (Piecewise linear NSSC algorithm)

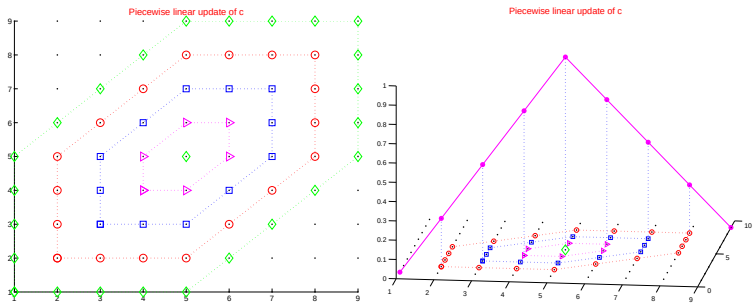

Figure 4: A two dimensional basis function $\phi_{i,j}^k$. Note on the right plot, only the weights v_ℓ along a diagonal are shown. Here $\diamond$ defines the outer boundary of the 2D basis function, $\bigcirc$ shows the nodes where the corresponding weights are $1/4$, $\square$ shows the nodes where the corresponding weights are $1/2$, $\triangleright$ shows the nodes where the corresponding weights are $3/4$ and the central node $\diamond$ defines the weight of 1.

- For $k = 1, 2, \cdots L+1$
  - For $i, j = 1, 2, \cdots n_k + 1$:

$$\text{Find } c = \text{argmin}_s J(u + s\phi_{i,j}^k), \text{ and update } u \text{ as } u := u + c\phi_{i,j}^k. \tag{24}$$

  - End.
- End.

The subproblems (24) are solved by approximate solvers in [28]. The only difference from the piecewise constant case is the evaluation of the values of $s\phi_{i,j}^k$. In Figure 4, the value of a basis function $\phi_{i,j}^k$ on the coarse level $k = 3$ is displayed.

Here it is important to point out that the subproblem in (24) is not expensive to solve due to the compact support of $\phi_{i,j}^k$. In fact, we can simplify the functional $J(w + s\phi_{i,j}^k)$, $s \in \mathbb{R}$, much further for efficient implementation [28]. We remark that [12] shows that piecewise linear finite element functions can be used to approximate a special variant of the standard total variation (as in (1)) of bounded variation functions. However justification for the convergence of Algorithm 3 is not yet available.

7 Algorithmic complexities

For linear problems, the cost per iteration for the multigrid iteration is typically $O(N)$ flops (floating point operations), where N is the total number of degrees of freedom. For our nonlinear problems [19, 28], the cost per iteration by our Algorithm 1 is

$$2(L+1)N + (2 + 4\kappa/3)N \approx O(N \log N)$$

and by Algorithm 3 is

$$(5N + 32\kappa N)(L+1) \approx O(N \log N),$$

where we assume that $\kappa = O(1)$ steps are needed for a typical inner iteration. Here the reason why Algorithm 3 appears slightly more expensive than Algorithm 1 is that for a typical block of pixels the former only involves boundary pixels interaction while all pixels in a block in the latter interact with each other.

By way of comparisons, the second order cone method [33] costs $O(N\sqrt{N})$ while most time-marching methods (including the AOS method) cost $O(\kappa N)$ where κ is the number of iterations. In the explicit Euler method, $\Delta t \approx h^2 \approx 1/N$. So the complexity for marching to $t = O(1)$ with

$\kappa = O(N)$ will be $O(N^2)$ while the AOS method [74] is known to be 10 times faster than this. As mentioned, the cost of a fixed iteration method may not be easy to estimate as the inner solver is efficient and the outer iteration can be quite slow.

8 Numerical experiments

To demonstrate the effectiveness of our Algorithms 1 and 3, we now present some experimental results. We remark that the above proposed algorithms have not been applied to the image minimisation problem (1) before [19, 28], although attempts on solving (4) have been made.

Effectiveness testing. We have tested the algorithms' effectiveness by solving many image denoising problems. It appears that usually a few multigrid cycles (typically 4) are sufficient to obtain an acceptable and converged result. However, readers may be more interested in comparisons with existing algorithms. Below we shall focus on this aspect. It should be remarked that some comparisons of multigrid methods with non-multigrid methods such as the fixed point iterations and time marching schemes may be found in [56]; the result is not surprising in that the former is faster whenever it converges.

Comparisons with an established method. There are many aspects of the discussed algorithms that could be compared with other methods. Here we choose to compare with the well-known method (perhaps the best but there are strong competitors from §2 not compared yet) of Chan-Golub-Mulet (CGM) [23]. However our task of comparing with CGM becomes somewhat easier because the CGM method 'fails' in 2 cases: (i) when the image size N becomes large (due to ill-conditioning); (ii) when $\beta \leq 10^{-32}$ (due to singularity). Here (i), not (ii), may be fixable by finding a better preconditioner (a non-trivial task) but no such work is available. In both cases, our method would converge although the local solvers take a few more iterations.

It may be of interest to show some results from parameter ranges where the CGM performs well: we take $\beta = 10^{-10}, 10^{-20}$ and 3 test examples in Figs. 5, 6 and 7. Here we mainly compare the solution's visual quality and the PSNR value which is the peak signal-to-noise ratio (PSNR) defined by (see e.g. [17])

$$\mathrm{PSNR}(u, w) = 10 \log_{10} \frac{255^2}{\frac{1}{mn} \sum_{i,j} (u_{i,j} - w_{i,j})^2},$$

where $w_{i,j}$ and $u_{i,j}$ denote the pixel values of the restored and the original images respectively. One observes that our multilevel methods only require 3-5 cycles to obtain a comparable result.

Clearly as displayed in the vertical labels of the plots, the PSNR values of the results from our algorithms are quite close to the CGM results. Comparing CPUs is a harder task on the MATLAB platform; a more convincing test would be to use C or Fortran in some optimal implementation. Nevertheless, our observation for the relatively small 256×256 examples is that Algorithm 1 is about 3 faster than the CGM [23] while Algorithm 3 is about as fast as [23]. This may be predicted by the complexity results shown above.

However our new algorithms are evidently more robust (without having to concern about what parameters to use) and as multilevel methods they have a scope to achieve even better performance with large images and future parallelization.

9 Conclusions

This paper first surveyed various solution techniques for the image denoising problem, then discussed multigrid methods for solving total variation minimization via the differential equation approach, and finally presented two related multilevel algorithms for solving total variation minimization directly.

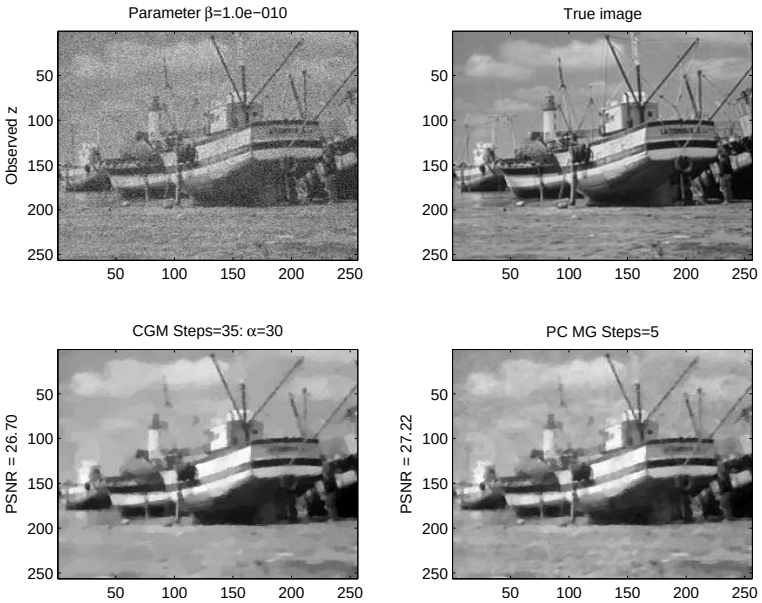

Figure 5: Comparison of Algorithm 1 with the CGM method [23] for test example P_1: $\alpha = 30$ and $\beta = 10^{-10}$

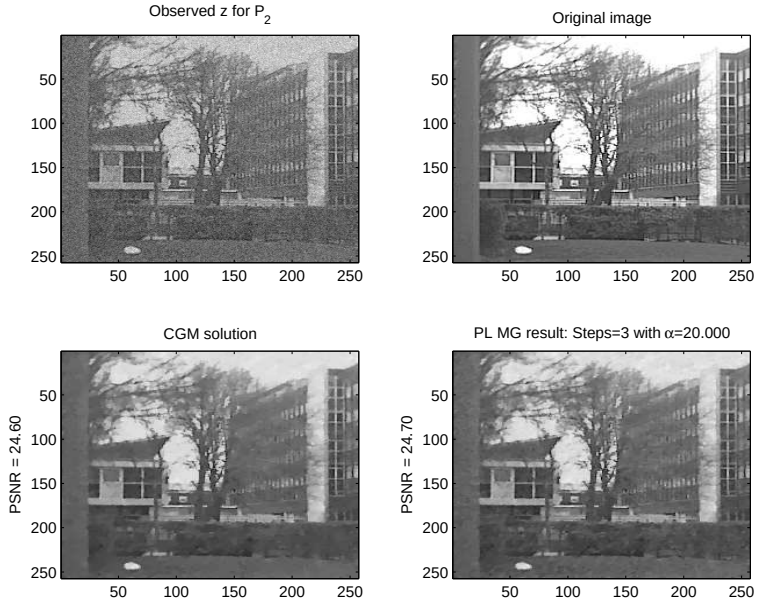

Figure 6: Comparison of Algorithm 3 with the CGM method [23] for test example P_2: $\alpha = 20$.

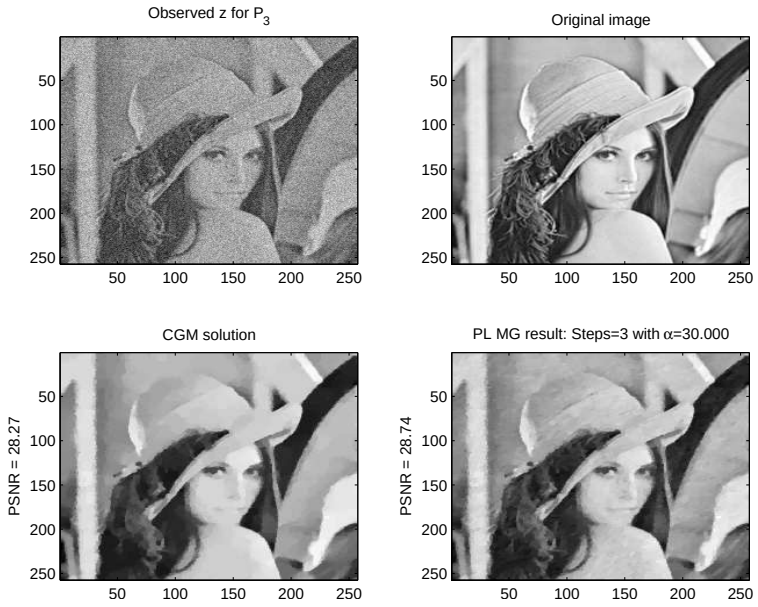

Figure 7: Comparison of Algorithm 3 with the CGM method [23] for test example P_3: $\alpha = 30$.

The subspace correction based algorithms differ from previous attempts for solving similar optimization problems. Numerical tests show that firstly and most importantly the new multilevel algorithms are robust and fast, and secondly they compare favorably with the well-known CGM algorithm [23], which is not a multilevel method.

Acknowledgements

The authors thank the anonymous referees for making helpful suggestions and suggesting relevant references. This work is supported in parts by the Office of Naval Research ONR N00014-03-1-0888, the National Institutes of Health NIH U54-RR021813 and the Leverhulme Trust RF/9/RFG/2005/0482.

References

[1] R. Acar and C. R. Vogel. *Analysis of total variation penalty method for ill-posed problems*, Inverse Probs., 10:1217–1229, 1994.

[2] S. T. Acton. *Multigrid anisotropic diffusion*, IEEE Trans. Imag. Proc., 3 (3):280–291, 1998.

[3] I. Albarreal, M. C. Calzada, J. L. Cruz, E. Fernández-Cara, J. Galo, and M. Marín. Convergence analysis and error estimates for a parallel algorithm for solving the Navier-Stokes equations. *Numer. Math.*, 93(2):201–221, 2002.

[4] E. Arian and S. Ta'asan. *Multigrid one-shot methods for optimal control problems*, ICASE technical report No. 94-52, USA, 1994.

[5] P. Blomgren, T. F. Chan and P. Mulet, *Extensions to Total Variation Denoising*, Proc. SPIE 97, San Diego, USA, 1997.

[6] P. Blomgren, T. F. Chan, P. Mulet, L. Vese, and W. L. Wan. *Variational PDE models and methods for image processing*, in Research Notes in Mathematics, 420:43–67. Chapman & Hall/CRC, 2000.

[7] A. Bruhn, J. Weickert, T. Kohlberger and C. Schnörr, *A multigrid platform for real-time motion computation with discontinuity-preserving variational methods*, Technical Report No. 136, Department of Mathematics, Saarland University, Saarbrücken, Germany, May 2005.

[8] M. Burger, S. Osher, J. Xu and G. Gilboa. *Nonlinear inverse scale space methods for image restoration*, Comm. Math. Sci., 4 (1), pp.179-212, 2006. (See also UCLA CAM report 05-34, 2005).

[9] A. Brandt, *Multilevel adaptive solutions to boundary value problems*, *Math. Comp.*, pp.333-190, 1977.

[10] A. Brandt, *Multigrid solvers and multilevel optimization strategies*, In J. Cong and J. R. Shinnerl, editors, *Multiscale Optimization and VLSI/CAD*, pp.1–68. Kluwer Academic (Boston), 2000.

[11] J. L. Carter. *Dual method for total variation-based image restoration*, CAM report 02-13, PhD thesis, University of California at LA, USA; see http://www.math.ucla.edu/applied/cam/index.html.

[12] E. Casas, K. Kunisch and C. Pola, *Regularization of functions of bounded variation and applications to image enhancement*, Appl. Math. Optim., 40:229-257, 1999.

[13] A. Chambolle. *An algorithm for total variation minimization and applications*, J. Math. Imag. Vis., 20:89–97, 2004.

[14] A. Chambolle and P.L. Lions. *Image recovery via total variation minimization and related problems*, Numer. Math., 76 (2):167–188, 1997.

[15] R. H. Chan, T. F. Chan, and W. L. Wan. *Multigrid for differential convolution problems arising from image processing*, in R. Chan, T. F. Chan, and G. H. Golub, editors, *Proc. Sci. Comput. Workshop.* Springer-Verlag, see also CAM report 97-20, UCLA, USA, 1997.

[16] R. H. Chan, Q. S. Chang, and H. W. Sun., *Multigrid method for ill-conditioned symmetric Toeplitz systems*, SIAM J. Sci. Comput., 19:516–529, 1998.

[17] R. H. Chan, C. W. Ho, and M. Nikolova. *Salt-and-pepper noise removal by median-type noise detectors and detail-preserving regularization*, IEEE Trans. Image Proc., to appear, 2005.

[18] R. H. Chan and C. K. Wong. *Sine transform based preconditioners for elliptic problems*, Numer. Linear Algebra Applic., 4:351–368, 1997.

[19] T. F. Chan and K. Chen. *On a nonlinear multigrid algorithm with primal relaxation for the image total variation minimisation*, to appear: Numer. Alg., 2006.

[20] T. F. Chan and K. Chen. *An optimization-based multilevel algorithm for total variation image denoising*, to appear: SIAM J. Multiscale Mod. Sim., 2006.

[21] T. F. Chan and S. Esedoglu. *Aspects of total variation regularized L^1 function approximation*, UCLA CAM report 04-07, 2004.

[22] T. Chan, S. Esedoglu, F. Park and A. Yip, *Recent Developments in Total Variation Image Restoration*, in The Handbook of Math. Models in Computer Vision, eds. N. Paragios, Y. M. Chen and O. Faugeras, Springer-Verlag, pp.17-32, 2005. (See also CAM report 05-01, UCLA, USA.)

[23] 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 (6):1964–1977, 1999.

[24] T. F. Chan and T. P. Mathew. *Domain decomposition algorithms*, in: Acta Numerica, ed. A. Iserles, pp.61-143, 1994.

[25] T. F. Chan and P. Mulet. *Iterative methods for total variation restoration*, CAM report 96-38, UCLA, USA, 1996;
see http://www.math.ucla.edu/applied/cam/index.html.

[26] Q. S. Chang and I. L. Chern. *Acceleration methods for total variation-based image denoising*, SIAM J. Sci. Comput., 25:982–994, 2003.

[27] K. Chen. *Matrix Preconditioning Techniques and Applications*. Cambridge Monographs on Applied and Computational Mathematics (No. 19). Cambridge University Press, UK, 2005.

[28] K. Chen and X.-C. Tai. *A nonlinear multigrid method for total variation minimization from image restoration*, see UCLA CAM report 05-26, USA, 2005.

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

[30] J. R. Galo, I. Albarreal, M. C. Calzada, J. L. Cruz, E. Fernández-Cara, and M. Marín. Stability and convergence of a parallel fractional step method for the solution of linear parabolic problems. *AMRX Appl. Math. Res. Express*, (4):117–142, 2005.

[31] José R. Galo, Isidoro I. Albarreal, M. Carmen Calzada, José Luis Cruz, Enrique Fernández-Cara, and Mercedes Marín. A simultaneous directions parallel algorithm for the Navier-Stokes equations. *C. R. Math. Acad. Sci. Paris*, 339(3):235–240, 2004.

[32] José R. Galo, Isidoro I. Albarreal, M. Carmen Calzada, José Luis Cruz, Enrique Fernández-Cara, and Mercedes Marín. Simultaneous directions parallel methods for elliptic and parabolic systems. *C. R. Math. Acad. Sci. Paris*, 339(2):145–150, 2004.

[33] D. Goldfarb and W. T. Yin, *Second-order cone programming methods for total variation-based image restoration*, SIAM J. Sci. Comput., 27 (2):622-645, 2005.

[34] M. Hintermüller and K. Kunisch, *Total bounded variation regularization as a bilaterally constrained optimization problem*, SIAM J. Appl. Math., 64:1311-1333, 2004.

[35] M. Hintermüller and G. Stadler, *An infeasible primal-dual algorithm for TV-based infconvolution- type image restoration*, Technical Report TR04-15, CAAM Dept. Rice University, USA, 2004.

[36] W. Hinterberger, M. Hintermüller, K. Kunisch, M. von Oehsen and O. Scherzer, *Tube Methods for BV Regularization*, J. Math. Imaging Vis., 19:219-235, 2003.

[37] K. Ito and K. Kunisch, *An active set strategy based on the augmented Lagrangian formulation for image restoration*, Math. Mod. Numer. Anal. (M2AN), 33 (1):1-21, 1999.

[38] T. Kärkkäinen and K. Majava, Nonmonotone and monotone active set methods for image restoration II. numerical results, J. Optim. Theory Appl., 106:81-105, 2000.

[39] T. Kärkkäinen, K. Majava and M. M. Mäkelä, *Comparison of formulations and solution methods for image restoration problems*, Series B Report No. B 14/2000, Department of Mathematical Information Technology, University of Jyväskylä, Finland, 2000.

[40] C. T. Kelley. *Iterative Methods for Solving Linear and Nonlinear Equations.* SIAM publications, USA, 1995.

[41] R. Kimmel and I. Yavneh. *An algebraic multigrid approach for image analysis*, SIAM J. Sci. Comput., 24(4):1218–1231, 2003.

[42] S. H. Lee and J. K. Seo, *Noise removal with Gauss curvature driven diffusion*, IEEE Trans. Image Proc., 14 (7):904-909, 2005.

[43] Y. Y. Li and F. Santosa. *A computational algorithm for minimizing total variation in image restoration*, IEEE Trans. Image Proc., 5 (6):987-995, 1996.

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

[45] T. Lu, P. Neittaanmäki, and X.-C. Tai. A parallel splitting-up method for partial differential equations and its applications to Navier-Stokes equations. *RAIRO Modél. Math. Anal. Numér.*, 26(6):673–708, 1992.

[46] M. Lysaker, A. Lundervold and X. C. Tai. *Noise removal using fourth-order partial differential equation with applications to medical magnetic resonance images in space and time*, IEEE Trans. Imag. Proc., 12 (12):1579–1590, 2003.

[47] F. Malgouyres. *Minimizing the total variation under a general convex constraint for image restoration*, *IEEE Trans. Imag. Proc.*, 11 (12):1450–1456, 2002.

[48] A. Marquina and S. Osher. *Explicit Algorithms for a new time dependant model based on level set motion for nonlinear deblurring and noise removal*, SIAM J. Sci. Comput., 22(2):387–405, 2000.

[49] S. Nash. *A multigrid approach to discretized optimisation problems*, J. Opt. Methods Softw., 14:99–116, 2000.

[50] M. K. Ng, L. Q. Qi, Y. F. Yang and Y. M. Huang, *On semismooth Newton's methods for total variation minimization*, Technical Rep. 413, Dept of Math., Honk Kong Baptist Univ., China, 2005.

[51] M. V. Oehsen, *Multiscale Methods for Variational Image Denoising*, Logos Verlag, Berlin, 2002.

[52] S. Osher and R. Fedkiw. *Level Set Methods and Dynamic Implicit Surfaces.* Springer, 2003.

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

[54] E. Radmoser, O. Scherzer and J. Schöberl, A cascadic algorithm for bounded variation regularization, SFB-Report No. 00-23, Johannes Kepler University of Linz, Austria, 2000.

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

[56] J. Savage and K. Chen, *An improved and accelerated nonlinear multigrid method for total-variation denoising*, Int. J. Comput. Math., 82 (8):1001–1015, 2005.

[57] J. Savage and K. Chen. *On multigrids for solving a class of improved total variation based PDE models*, in this proceeding, 2006.

[58] O. Scherzer. *Taut-String Algorithm and Regularization Programs with G-Norm Data Fit*, J. Math. Imaging and Vision, 23 (2):135–143, 2005

[59] K. Stuben. *An introduction to algebraic multigrid*, in Appendix A of [66]. Also appeared as GMD report 70 from `http://www.gmd.de` and
`http://publica.fhg.de/english/index.htm`, 2000.

[60] S. Ta'asan. *Lecture note* 4 *of Von-Karman Institute Lectures*, Belgium,
`http://www.math.cmu.edu/~shlomo/VKI-Lectures/lecture4`, 1997.

[61] X. C. Tai. Rate of convergence for some constraint decomposition methods for nonlinear variational inequalities. *Numer. Math.*, 93:755–786, 2000.

[62] X. C. Tai and M. Espedal. *Rate of convergence of some space decomposition methods for linear and nonlinear problems*, SIAM. J. Numer. Anal., 35:1558–1570, 1998.

[63] Xue-Cheng Tai, Bjorn-ove Heimsund and Jin Chao Xu. Rate of convergence for parallel subspace correction methods for nonlinear variational inequalities. In *Thirteenth international domain decomposition conference*, pages 127–138. CIMNE, Barcelona, Spain, 2002. Available online at: http://www.mi.uib.no/7%Etai/.

[64] X. C. Tai and P. Tseng. *Convergence rate analysis of an asynchronous space decompostion method for convex minimization*, Math. Comp., 71:1105–1135, 2001.

[65] X. C. Tai and J. C. Xu. *Global and uniform convergence of subspace correction methods for some convex optimization problems*, Math. Comp., 71:105–124, 2001.

[66] U. Trottenberg, C. W. Oosterlee and A. Schuller. *Multigrid*, Academic Press, London, UK, 2000.

[67] P. Tseng, *Convergence of a block coordinate descent method for nondifferentiable minimization*, J. Optim. Theory and Applics., 109 (3):475-494, 2001.

[68] P. S. Vassilevski and J. G. Wade, *A comparison of multilevel methods for total variation regularization*, Elec. Trans. Numer. Anal., 6:255-270, 1997.

[69] C. R. Vogel. *A multigrid method for TV-based image denoising*, in *Computation and Control IV, 20, Progress in Systems and Control Theory*, eds. K. Bowers and J. Lund, Birkhauser, 1995.

[70] C. R. Vogel. *Negative results for multilevel preconditioners in image deblurring*, in *Scale-Space Theories In Computer Vision*, eds. M. Nielson et al, pp.292–304. Springer, 1999.

[71] C. R. Vogel. *Computational methods for inverse problems.* SIAM publications, USA, 2002.

[72] C. R. Vogel and M. E. Oman. *Iterative methods for total variation denoising*, SIAM J. Sci. Statist. Comput., 17:227–238, 1996.

[73] C. R. Vogel and M. E. Oman. *Fast, robust total variation-based reconstruction of noisy, blurred images*, IEEE Trans. Image Proc., 7:813–824, 1998.

[74] J. Weickert, B. M. ter Haar Romeny and M. A. Viergever, *Efficient and reliable schemes for nonlinear diffusion filtering*, IEEE Trans. Image Proc., 7:398-410, 1998.

[75] J. C. Xu. *Iteration methods by space decomposition and subspace correction*, SIAM Rev., 4:581–613, 1992.

[76] W. T. Yin, D. Goldfarb and S. Osher, *Image cartoon-texture decomposition and feature selection using the total variation regularized L1 functional*, CAM report CAM05-47, 2005, UCLA, USA.

[77] A. M. Yip and F. Park. *Solution dynamics, causality, and critical behavior of the regularization parameter in total variation denoising problems*, CAM report 03-59, UCLA, USA, 2003.