

An automatic regularization parameter selection algorithm in the total variation model for image deblurring

K. Chen · E. Loli Piccolomini · F. Zama

Received: 17 July 2013 / Accepted: 23 September 2013
© Springer Science+Business Media New York 2013

Abstract Image restoration is an inverse problem that has been widely studied in recent years. The total variation based model by Rudin-Osher-Fatemi (1992) is one of the most effective and well known due to its ability to preserve sharp features in restoration. This paper addresses an important and yet outstanding issue for this model in selection of an optimal regularization parameter, for the case of image deblurring. We propose to compute the optimal regularization parameter along with the restored image in the same variational setting, by considering a Karush Kuhn Tucker (KKT) system. Through establishing analytically the monotonicity result, we can compute this parameter by an iterative algorithm for the KKT system. Such an approach corresponds to solving an equation using discrepancy principle, rather than using discrepancy principle only as a stopping criterion. Numerical experiments show that the algorithm is efficient and effective for image deblurring problems and yet is competitive.

Keywords Constrained/unconstrained problem · Discrepancy principle · Regularization parameter · Image deblurring · Total variation · Lagrange multiplier

Mathematics Subject Classifications (2010) 62H35 · 65N22 · 65N55 · 74G65 · 74G75

K. Chen

Centre for Mathematical Imaging Techniques (CMIT), and Department of Mathematical Sciences, University of Liverpool, Liverpool, UK
e-mail: K.Chen@liverpool.ac.uk

E. Loli Piccolomini (✉) · F. Zama
Department of Mathematics, University of Bologna,
Piazza di Porta San Donato 5, 40126 Bologna, Italy
e-mail: elena.loli@unibo.it

F. Zama
e-mail: fabiana.zama@unibo.it

1 Introduction

Image denoising and deblurring are two fundamental and widely studied problems in image processing. Image restoration can be modelled by the linear equation [12, 19, 28, 31]:

$$y^\delta(z, w) = (Hx)(z, w) + \delta(z, w) = \int_{\Omega} \kappa(u - z, v - w)x(u, v)dudv + \delta(z, w) \quad (1)$$

where κ is the blurring kernel representing the data acquisition process, $y^\delta(z, w) : \Omega \rightarrow \mathbb{R}^+$ is the given data function, $\delta(z, w)$ is the unknown noise and $x(u, v) : \Omega \rightarrow \mathbb{R}^+$ is the image function to be recovered. Here $\Omega \subset \mathbb{R}^2$. The kernel κ defines a spatially invariant operator H as a convolution. There exist other works that assume κ to be also unknown and attempt to restore it in the so-called blind deconvolution. A more general framework than (1) would assume H to be spatially variant which leads to a large scale and non-structured matrix problem. When the convolution operator $H = I$, the problem becomes the denoising case which is easier to deal with. Here we focus on the deblurring case of $H \neq I$.

For simplicity of notation, in the following we use x instead of $x(u, v)$ when it is not necessary to express the dependence from u, v . The problem (1) is ill-posed for finding x and regularization methods are necessary.

A common formulation of the regularization is the following unconstrained minimization:

$$\min_x F(x) + \lambda J(x) \quad (2)$$

where $F(x)$ is a data fit function, $J(x)$ is a regularization function and λ is a positive parameter. The choice of a suitable value of the regularization parameter λ is still a challenge. Moreover, if we consider in (2) λ as a variable, the function

$$\mathcal{L}(x, \lambda) = F(x) + \lambda J(x)$$

is the Lagrangian function of the constrained regularized problem:

$$\min_x F(x), \quad \text{s. t. } \mathcal{R}(x) \leq \gamma \quad (3)$$

with $J(x) = \mathcal{R}(x) - \gamma$ and γ is an estimate of the value of the regularization function in the exact solution x^* , $\mathcal{R}(x^*)$. Using the dual equivalent formulation of (3):

$$\max_{\lambda} \min_x \mathcal{L}(x, \lambda), \quad (4)$$

we propose an algorithm for the estimate of both the exact multiplier λ^* and the exact solution x^* of (3).

In this paper, we focus our attention on the particular case where $F(x)$ is the least squares function

$$F(x) = \frac{1}{2} \|Hx - y^\delta\|_2^2. \quad (5)$$

For what concerns the regularization function, in recent years a number of effective regularization functions have been proposed in the literature. Among them is the total

variation based function by Rudin-Osher-Fatemi (ROF) [28]. The Total Variation (TV) seminorm function is:

$$TV(x) = \int_{\Omega} |\nabla x(u, v)| dudv. \quad (6)$$

With a fixed λ , the well-known ROF TV equation can be derived from (2)

$$H^*(Hx - y^\delta) - \lambda \nabla \cdot \frac{\nabla x}{|\nabla x|_\beta} = 0, \quad (7)$$

where $|\nabla x|_\beta = \sqrt{|\nabla x|^2 + \beta}$ and β is a small and positive parameter to deal with the case $|\nabla x| = 0$; an alternative to using $|\nabla x|_\beta$ is provided by the Huber function as in [17].

The TV function (6) is probably the most popular and widely used for regularization in deblurring [5, 30], denoising [11, 31], and reconstruction [4, 13] imaging applications. The reason is that it allows to recover the edges of the image. Nowadays, the literature on this topic is really huge and many methods have been proposed for the solution of problem (2) with TV regularization. See for example [10, 21, 28, 31] and the references therein. However, while capable of capturing sharp edges using piecewise constant approximations, the TV function (6) can produce undesirable staircasing and is thus unsuitable for restoring smooth images where there are no sharp edges. In the last few years, various efforts were made to use alternative (and often higher order) regularization; see [6–8] and the references therein.

Here we focus on the ROF model. Despite its success when equipped with suitably chosen λ in many applications, it is yet a challenge how to automatically determine this regularization parameter λ . As with other formulations from inverse problems, the generic methods of discrepancy principle (DP) and L-curve may be considered; these lead to practical but non-optimal parameters. Indeed, in almost all works on TV regularization, λ is chosen from DP, L-curves or heuristically fixed. See [27] for the most recent results on these methods. A proposal for choosing λ can be found in [32], where the constrained regularization problem is solved by a primal-dual method and λ is chosen on a discrepancy-principle basis in the update of the dual variables. Another proposal is in [17], where the update of the regularization parameter is made, at each iteration of the solution method, based on the estimate of noise variance and on a hierarchical image decomposition, as proposed in [29]. A different approach is taken in [9, 25, 26, 34] computing λ as the Lagrange multiplier to solve the KKT conditions of the constrained problem (8). In this way, the regularization parameter ensures the constrained and unconstrained problems are equivalent. In particular, [9, 34] solve the constrained problem (3), where the noise information on the data is assumed to be known or estimated first. This work does find the optimal λ but it is only for image denoising (i.e. not applicable to $H \neq I$). In [25, 26], the Tikhonov regularization problem (which is less challenging than the ROF model) is solved but tested only on 1D discretized Fredholm integral equations. The recent study for (2) by [23] addressed the choice of λ for the Tikhonov regularization through DP.

Motivation The aim of this paper is to extend to the more general case of deblurring, where $H \neq I$ in (5), the idea proposed in [9, 34] of computing the regularization

parameter λ via the KKT conditions. We formulate the deblurring problem as in (3) and we solve the equivalent unconstrained form (4). In this case, it is required an estimate of $TV(x^*)$, where x^* is the exact solution.

In literature, an equivalent constrained regularized formulation of (3) is also used [2, 3]:

$$\min_x \mathcal{R}(x), \quad \text{s. t.} \quad F(x) \leq \sigma \quad (8)$$

where σ is an estimate of the noise norm. Although (8) and (3) are equivalent, in some applications where we know the possible content in the true image, it can be easier to estimate $TV(x^*)$ in (3) than to estimate the noise level in (8). For example in medical applications, if we know which organs are being restored, this priori knowledge can help us estimate $TV(x^*)$ quite accurately regardless of what the noise level is.

Contribution The main contribution of the paper is a new algorithm to automatically compute the optimal parameter λ and the solution x of (2) by solving problem (4) and we propose an automatic procedure for the computation of a suitable value of γ . The algorithm requires, for each iteration, the solution of a problem of the form (2) with λ fixed. We use and compare two different existing algorithms at this aim, hence we test two different versions of our method.

We tested the proposed algorithm on image deblurring applications where the Point Spread Function (PSF) H is applied to an image function supposed to be periodic outside the image domain Ω . In this case, the matrix obtained from the discretization of H has a Block Toeplitz with Toeplitz Blocks (BCCB) structure and it can be diagonalized by Fast Fourier Transforms (FFT). Numerical results obtained on large test problems show that the proposed algorithm converges fast to a good restored image even when γ is unknown.

The rest of the paper is organized as follows. In Section 2 we present the mathematical theory motivating the proposed method; in Section 3 we describe the algorithm and finally in Section 4 some numerical results show the method performance on image deblurring applications. The results obtained are compared with Hintermuller's algorithm in [17]. Finally, Section 5 contains the conclusions.

2 Some mathematical background

In this section, we present some mathematical results concerning the solution of the problem (2), in the general case $H \neq I$. Here, we underline the dependence of the solution of (2) on λ by the notation $x(\lambda)$.

Lemma 1 ([1, 9]) *For $\lambda > 0$, the minimization problem (2) has a minimizer $x(\lambda)$.*

Proof For the existence of a solution in L^p space (with $1 \leq p < 2$), refer to [1]. For the existence of a solution in the BV space, refer to [9], especially when H is a linear but non-compact operator. When $H = I$, see [34]. \square

Lemma 2 *For $\lambda > 0$, the solution $x(\lambda)$ to the minimization problem (2) is unique up to H . It is unique when H is injective.*

Proof We first remark that when $H \neq I$ and κ from (1) is continuous, H is compact with most of its eigenvalues clustered at zero so H does not have an inverse and it is not injective.

To show the uniqueness, let \bar{x}, \tilde{x} be two solutions of (2) satisfying $\min \|Hx - y\|_2^2 \equiv 2\sigma^2$, and $\mathcal{R}(\bar{x}) \leq \gamma, \mathcal{R}(\tilde{x}) \leq \gamma$. Then $(\bar{x} + \tilde{x})/2$ is also a solution. In fact:

$$\left\| H \frac{\bar{x} + \tilde{x}}{2} - y \right\|_2 \leq \frac{1}{2} (\|H\bar{x} - y\|_2 + \|H\tilde{x} - y\|_2) = \sqrt{2}\sigma,$$

and

$$\mathcal{R} \left(\frac{\bar{x} + \tilde{x}}{2} \right) \leq \frac{1}{2} (\mathcal{R}(\bar{x}) + \mathcal{R}(\tilde{x})) \leq \gamma.$$

Because $\left\| H \frac{\bar{x} + \tilde{x}}{2} - y \right\|_2 \geq \sqrt{2}\sigma$, we must have $\left\| H \frac{\bar{x} + \tilde{x}}{2} - y \right\|_2 = \sqrt{2}\sigma$. Thus from

$$\|H\bar{x} - \tilde{x}\|_2 = \|H\bar{x} - y\|_2 = \left\| \frac{1}{2}(H\bar{x} - y) + \frac{1}{2}(H\tilde{x} - y) \right\|_2 = \sqrt{2}\sigma,$$

we have that $\left\| H \frac{\bar{x} - \tilde{x}}{2} \right\|_2 = 0$ or $H\bar{x} = H\tilde{x}$. That is, the uniqueness is up to Hx . When H is injective (or invertible in the discrete case), then $H\bar{x} = H\tilde{x}$ implies $\bar{x} = \tilde{x}$ or the uniqueness is true. □

Once the existence and uniqueness of (2) are known, below is the most important result for designing our computational algorithm. We note that the monotonicity property established in [9, 34] is for the quantity $F(x(\lambda))$ of problem (8), not yet for $J(x(\lambda))$ of problem (2).

Theorem 1 *For $\lambda > 0$, the function $J(x(\lambda))$ in (2) is monotonically decreasing in a feasible region and there exist two distinct parameters λ_1, λ_2 such that $J(x(\lambda_1)) > 0$ and $J(x(\lambda_2)) < 0$. Hence $J(x(\lambda))$ has a unique root λ^* such that $J(x(\lambda^*)) = 0$ which will be the optimal parameter.*

Proof To prove the result, we only need the existence of a minimizer (by Lemma 1). For two given parameters λ, μ such that $\lambda > \mu \geq 0$, denote the respective minimizers for $\min_x \mathcal{L}(x, \lambda)$ and $\min_x \mathcal{L}(x, \mu)$ by $x(\lambda), x(\mu)$. Then clearly $\mathcal{L}(x(\lambda), \lambda) < \mathcal{L}(x(\mu), \lambda)$ and $\mathcal{L}(x(\mu), \mu) < \mathcal{L}(x(\lambda), \mu)$. That is,

$$\begin{aligned} F(x(\lambda)) + \lambda J(x(\lambda)) &< F(x(\mu)) + \lambda J(x(\mu)), \\ F(x(\mu)) + \mu J(x(\mu)) &< F(x(\lambda)) + \mu J(x(\lambda)). \end{aligned}$$

Adding the two inequalities gives the desired result

$$(\lambda - \mu) (J(x(\lambda)) - J(x(\mu))) < 0 \quad \text{i.e.} \quad J(x(\lambda)) < J(x(\mu)).$$

For the interval, we may use $\lambda_1 = 0$ as a lower bound because at $\lambda = 0$ we have to solve

$$x(\lambda_1) = \arg \min_x \frac{1}{2} \|Hx - y\|_2^2$$

which admits at least one least squares solution (where x is non-unique); let $a = \mathcal{R}(x(\lambda_1))$ and, by the choice of γ (i.e. $a < \gamma$), we see that $J(x(\lambda_1)) = a - \gamma > 0$. For the upper bound, we note that

$$\begin{aligned} x(\lambda^+) &= \lim_{\lambda \rightarrow \infty} \arg \min_x \left\{ \frac{1}{2} \|Hx - y\|_2^2 + \lambda (\mathcal{R}(x) - \gamma) \right\} \\ &= \arg \min_x (\mathcal{R}(x) - \gamma) \\ &= \arg \min_x \mathcal{R}(x) \end{aligned}$$

which leads to $\mathcal{R}(x(\lambda^+)) = 0$ since \mathcal{R} is non-negative i.e. $J(x(\lambda^+)) = \mathcal{R}(x(\lambda^+)) - \gamma = -\gamma < 0$. Clearly from $J(x(\lambda_1)) > 0$, $J(x(\lambda^+)) < 0$, there exists a large and positive number $\lambda_2 = M$ (lying between λ_1 and λ^+) at which $J(x(\lambda_2)) = 0$. Hence the stated results hold. \square

3 The CLSTV algorithm

To proceed, we consider the discretization of the above continuous formulation. The images of size $n \times n$ are assumed lexicographically ordered in vectors of size $N = n \times n$ and denoted with bold characters. Using a finite difference method [31], let \mathbf{H} denote the discretized blurring matrix stemming from H after imposing a periodic boundary condition on the image function at $\partial\Omega$.

Below we define our iterative algorithm: Constrained Least Squares Total Variation (CLSTV), for solving problem (3), when $\mathcal{R}(\mathbf{x}) = TV(\mathbf{x})$, in its equivalent form:

$$\max_{\lambda} \min_{\mathbf{x}} \mathcal{L}(\mathbf{x}, \lambda), \quad \mathcal{L}(\mathbf{x}, \lambda) \equiv \frac{1}{2} \|\mathbf{H}\mathbf{x} - \mathbf{y}^\delta\|_2^2 + \lambda(TV(\mathbf{x}) - \gamma).$$

The algorithm computes two sequences, $\{\lambda_k\}$ and $\{\mathbf{x}_k\}$, converging to the regularization parameter λ^* and to the solution \mathbf{x}^* respectively. By imposing the first order conditions: $\nabla_{\lambda} \mathcal{L}(\mathbf{x}, \lambda) = 0$, we can state the problem as:

find λ s.t. $TV(\mathbf{x}) - \gamma = 0$ where \mathbf{x} solves the following minimization problem : (9)

$$\min_{\mathbf{x}} \frac{1}{2} \|\mathbf{H}\mathbf{x} - \mathbf{y}^\delta\|_2^2 + \lambda(TV(\mathbf{x}) - \gamma), \tag{10}$$

that is exactly the discrete form of problem (2). Using the result of Theorem 1 with $J(\mathbf{x}) \equiv TV(\mathbf{x}) - \gamma$ with \mathbf{x} given by (10), we can solve the nonlinear equation

$$J(\mathbf{x}) = TV(\mathbf{x}) - \gamma = 0 \tag{11}$$

by computing a sequence $\{\lambda_k\}$ that converges to the root λ^* and, solving (10) with $\lambda = \lambda_k$, we can compute the sequence $\{\mathbf{x}_k\}$ that converges to \mathbf{x}^* .

In particular, given a starting value λ_0 s.t. $TV(\mathbf{x}_0) - \gamma < 0$, we can compute the sequence λ_k using the bisection method as follows:

$$\lambda_k = \mathcal{F}_b(\lambda_{k-1}), \quad \mathcal{F}_b(\lambda_{k-1}) = \lambda_{k-1} + \text{sign}(TV(\mathbf{x}_{k-1}) - \gamma) \frac{\lambda_0}{2^k}, \quad k = 1, 2, \dots \tag{12}$$

Or, given two suitable values λ_{k-1} and λ_{k-2} , we can compute λ_k using the secant method:

$$\lambda_k = \mathcal{F}_s(\lambda_{k-1}), \quad \mathcal{F}_s(\lambda_{k-1}) = \lambda_{k-1} - \frac{TV(\mathbf{x}_{k-1}) - \gamma}{TV(\mathbf{x}_{k-1}) - TV(\mathbf{x}_{k-2})} (\lambda_{k-1} - \lambda_{k-2}), \quad k = 1, 2, \dots \tag{13}$$

In this case the values λ_{k-1} and λ_{k-2} must be within the convergence region of the secant method. In this paper, for the solution of (11) we use a hybrid method obtained by starting the secant iterations after a few (k_s) iterations of the bisection method.

The whole algorithm is outlined in Table 1. It has two main steps: feasibility check and solution computation.

Computation of the input value λ_0 In order to compute the required input parameter λ_0 we define a low pass filtered approximate solution $\tilde{\mathbf{x}}$ by a fast and efficient method. Exploiting the properties of the discrete convolution kernel \mathbf{H} it is possible to compute $\tilde{\mathbf{x}}$ by means of a Tikhonov low pass filter. By imposing periodic boundary conditions we obtain that \mathbf{H} is a BCCB matrix and we can diagonalize it by a unitary Fourier matrix \mathbf{F} such that $\mathbf{H} = \mathbf{F}^* \mathbf{S} \mathbf{F}$ where $\mathbf{S} = \text{diag}(s_i)$, $i = 1, 2, \dots$ [20]. Defining

$$\alpha = \min_i \left| \frac{(\mathbf{F} \cdot \mathbf{y})_i}{s_i} \right|,$$

Table 1 Algorithm CLSTV

```

Algorithm 1 (CLSTV( $\tau_a, \tau_r, \text{maxit}, \xi, \lambda_0, \gamma$ )).
  compute  $\mathbf{x}_0$  solving (10) with  $\lambda = \lambda_0$ ;
  if  $\mathcal{R}(\mathbf{x}_0) > \gamma$  % Feasibility check
     $\mu_0 = \lambda_0$ ;  $\ell = 0$ ;
    while  $TV(x_\ell) > \gamma$ 
       $\ell = \ell + 1$ ,  $\mu_\ell = \xi \cdot \mu_{\ell-1}$ ;
      compute  $\mathbf{x}_\ell$  solving (10) with  $\lambda = \mu_\ell$ ;
    end
     $\lambda_0 = \mu_\ell$ 
  end
   $k = 0$  % Solution Computation
  repeat
     $k = k + 1$ 
    if  $k < k_s$ 
       $\lambda_k = \mathcal{F}_b(\lambda_{k-1})$  as in (12)
    else
       $\lambda_k = \mathcal{F}_s(\lambda_{k-1})$  as in (13)
    end
    compute  $\mathbf{x}_k$  solving (10) with  $\lambda = \lambda_k$ 
  until exit condition (24)
  
```

we can compute

$$\tilde{\mathbf{x}} = \mathbf{F}^* \Phi(\alpha) \mathbf{F} \text{ with } \Phi(\alpha) = \text{diag} \left(\frac{|s_i|^2}{|s_i|^2 + \alpha^2} \right). \tag{14}$$

The starting value λ_0 is obtained by imposing that:

$$\lambda_0 = \alpha \frac{\|\mathbf{H}\tilde{\mathbf{x}} - \mathbf{y}\|_2}{TV(\tilde{\mathbf{x}})} \tag{15}$$

where α is a suitable scale parameter.

Computation of the input value γ The parameter γ should be given as close as possible to $TV(\mathbf{x}^*)$ where \mathbf{x}^* is the exact image. In order to obtain a suitable estimate the following heuristic procedure can be used.

The value γ is defined as an intermediate value in the interval $[\gamma_L, \gamma_H]$

$$\gamma = (1 - \theta)\gamma_L + \theta\gamma_H, \quad \theta \in [0, 1]. \tag{16}$$

The bounds of the interval are obtained as follows:

$$\gamma_L = 0.225 TV(\mathbf{x}_\delta), \quad \gamma_H = 1.995 TV(\mathbf{x}_\delta), \text{ where } \mathbf{x}_\delta = \tilde{\mathbf{x}} - \mathbf{y}$$

where \mathbf{y} is the acquired image to be restored and $\tilde{\mathbf{x}}$ is the Tikhonov low pass filtered image (14). Using this procedure we obtained a data dependent range such that $\gamma_L \leq TV(\mathbf{x}_{true}) \leq \gamma_H$ for a very large database of gray levels images.

Feasibility check step In the feasibility step the condition $TV(\mathbf{x}_0) - \gamma < 0$ is checked. If it is not satisfied we define a new starting value λ_0 by defining an increasing sequence $\mu_\ell = \xi \mu_{\ell-1}$, $\mu_0 = \lambda_0$, $\xi > 1$, and by computing \mathbf{x}_ℓ as the solution of (10) with $\lambda = \mu_\ell$. Since $J(\mathbf{x})$ is non increasing w.r. to λ (see Theorem 1) we can find a suitable value $\mu_\ell > \lambda_0$ s.t. $J(\mathbf{x}_\ell) - \gamma < 0$.

Solution computation step The solution computation step calculates the values $(\lambda_k, \mathbf{x}_k)$. The multipliers sequence $\{\lambda_k\}$ is obtained by applying the hybrid update method (12) and (13) for solving equation (11).

Different algorithms exist in literature for solving problem (10), such as the iterative splitting algorithm SP proposed in [22, 24, 33], the fixed point (FP) iteration method [30] or the multilevel algorithm proposed in [14].

In this paper, we use the SP and FP methods and compare their results in terms of time and accuracy. We remind, however, that the novelty of the proposed CLSTV algorithm lies in the method for computing a suitable value of λ and a fair solution of the inner problem (10) is only functional to the execution of the external iterations of the CLSTV algorithm.

By using the FP method we solve the nonlinear system $\mathcal{G}(\mathbf{x}) = 0$ where:

$$\mathcal{G}(\mathbf{x}) \equiv \mathbf{H}^T \mathbf{H} \mathbf{x} + \lambda \nabla_x TV(\mathbf{x}) - \mathbf{H}^T \mathbf{y}$$

by means of quasi Newton iterations outlined in Table 2 where $\nabla_x TV(\mathbf{x}(\lambda)) \equiv L_\beta(\mathbf{x})\mathbf{x}$ and

$$L_\beta(\mathbf{x}) = -\nabla \cdot \frac{\nabla \mathbf{x}}{\sqrt{|\nabla \mathbf{x}| + \beta^2}}, \quad \beta > 0.$$

Table 2 Fixed point algorithm for computing \mathbf{x}_k (10)

Algorithm 2 (FP(\mathbf{x}_{k-1} , \mathbf{y} , λ_k , \mathbf{H})).

$\mathbf{u}^0 = \mathbf{x}_{k-1}; j = 0$

repeat

$j=j+1;$

solve $(\mathbf{H}^t\mathbf{H} + \lambda_k L_\beta(\mathbf{u}^{j-1})) \mathbf{d}^j = \mathbf{H}^t\mathbf{y}$

$\mathbf{u}^j = \mathbf{u}^{j-1} + \rho\mathbf{d}^j, \quad \rho \leq 1$

until *exit test* (19)

$\mathbf{x}_k = \mathbf{u}^j$

Provided that \mathbf{d}^j is a descent direction, global convergence can be guaranteed by appropriate control of the stepsize ρ . In our experiments we did not use any stepsize control (i.e. $\rho = 1$) and if \mathbf{d}^j is not a descent direction then the FP iterations are stopped.

The linear system arising from the Newton iterations applied to the equation $\mathcal{G}(\mathbf{x}) = 0$:

$$(\mathbf{H}^T\mathbf{H} + \lambda_k L_\beta(\mathbf{u}^j)) \mathbf{d}^j = \mathbf{H}^T\mathbf{y} \tag{17}$$

is solved by means of Truncated Conjugate Gradients (TCG) iterations of Steihaug [16, 18]. The iterations are stopped when:

$$\|\mathbf{r}_j\| < \zeta\|\mathbf{r}_0\| \quad \text{or} \quad j > \text{maxiter}_{CG} \tag{18}$$

where \mathbf{r}_j is the CG residual at step j and ζ is a given tolerance less than one.

The FP iterations are stopped when the following conditions are satisfied:

$$\mathcal{L}(\mathbf{u}^{j+1}, \lambda_k) \geq \mathcal{L}(\mathbf{u}^j, \lambda_k) \quad \text{or} \quad \frac{\|\nabla\mathcal{G}(\mathbf{u}^j)\|}{\|\nabla\mathcal{G}(\mathbf{u}^0)\|} < \epsilon_1 \quad \text{or} \quad \|\mathbf{d}^j\|^2 < \epsilon_2 \quad \text{or} \quad j \geq \text{maxit}_{FP}, \tag{19}$$

where ϵ_1 and ϵ_2 are two given tolerances and maxit_{FP} is the maximum number of FP iterations.

Using the splitting method SP [33] for the solution of (10), we compute \mathbf{x}_k by decoupling problem (10) as a sequence of problems of the form:

$$\min_{\tilde{\mathbf{u}}} \|\mathbf{H}\tilde{\mathbf{u}} - \mathbf{y}\|_2^2 + \alpha_1 \|\tilde{\mathbf{u}} - \mathbf{u}^{j-1}\|_2^2 \tag{20}$$

$$\min_{\mathbf{u}} \|\mathbf{u} - \tilde{\mathbf{u}}\|_2^2 + \alpha_2(TV(\mathbf{u}) - \gamma) \tag{21}$$

where $\mathbf{u}^0 = \mathbf{x}_k$ and the new iterate \mathbf{u}^j is obtained by solving (21). We underline that (20) is a deblurring problem, while (21) is a denoising problem.

The parameter α_2 is assigned by the CLSTV iterations as the current λ_k while the parameter α_1 can be constant (as proposed in [33]) or can change its value as α_2 , as shown in the numerical results in Section 4.

Problem (20) requires the solution of the linear system:

$$(\mathbf{H}^T\mathbf{H} + \alpha_1\mathbf{I}) \tilde{\mathbf{u}} = \mathbf{H}^T\mathbf{y} + \alpha_1\mathbf{u}^{j-1} \tag{22}$$

Table 3 Splitting method SP for computing \mathbf{x}_k (10)

```

Algorithm 3 (SP( $\mathbf{x}_{k-1}$ ,  $\mathbf{y}$ ,  $\lambda_k$ ,  $\mathbf{H}$ )).
 $\alpha_2 = \lambda_k$ 
 $\alpha_1 = 0.8$ ; % or ( $\alpha_1 = \alpha_2$ )
 $\mathbf{u}^0 = \mathbf{x}_{k-1}$ ;  $j = 0$ 
repeat
     $j=j+1$ ;
    compute  $\tilde{\mathbf{u}}$  by solving (22)
    compute  $\mathbf{u}^j$  by solving (21)
until exit test (23)
 $\mathbf{x}_k = \mathbf{u}^j$ 
    
```

provided a suitable parameter α_1 is given, an efficient and accurate solution is easily computed when \mathbf{H} can be diagonalized by discrete Fourier transforms.

Problem (21) is a denoising problem which can be efficiently solved by several methods such as [10, 15]. In Table 3 the steps of the SP iterative algorithm for solving (10) are reported.

The iterations are stopped when the relative distance between two successive iterates is small or when a maximum number of iterations is reached:

$$\frac{\|\mathbf{u}^j - \mathbf{u}^{j-1}\|}{\|\mathbf{u}^{j-1}\|} \leq \tau_r \text{ or } j > \text{maxit}_{SP} \tag{23}$$

Exit conditions of CLSTV The exit condition used to stop the iterations of CLSTV is the following:

$$|TV(\mathbf{x}_k) - \gamma| < \tau_r |TV(\mathbf{x}_0) - \gamma| + \tau_a \text{ or } |\lambda_k - \lambda_{k-1}| < \tau_a \text{ or } k > \text{maxit}_{CLSTV}. \tag{24}$$

where τ_a, τ_r are relative and absolute tolerance parameters.

4 Numerical results

In this section we present some numerical results to show the effectiveness of the proposed CLSTV method in image deblurring problems. All the tests have been performed in Matlab, R2010a, on 8 Intel i7 processors with 24 GB Ram. In the experiments the blurred noisy image \mathbf{y}^δ is obtained by discrete convolution of the original image \mathbf{x}^* with the kernel \mathbf{H} representing the PSF:

$$\mathbf{y}^\delta = \mathbf{H} * \mathbf{x}^* + \mathbf{e}.$$

We define the noise level NL as:

$$NL = \frac{\|\mathbf{e}\|_F}{\|\mathbf{y}^\delta\|_F} \tag{25}$$

where $\|\cdot\|_F$ is the Frobenius norm.

In our tests the parameter β in (7) has been fixed equal to 10^{-2} . The following blurring kernels are used [20]:

- Gaussian Blur (GB): normalized gaussian kernel \mathbf{H} of size 9×9 and variance 1.
- Motion Blur (MB): approximation of the linear motion of a camera by 10 pixels, with a 45 degrees angle.

We report the results obtained with two test images (I1 and I2) having different characteristics:

- I1 Cameraman test image (Fig. 1a), with 256×256 pixels in range $[0, 1]$.
- I2 Test image with 500×500 pixels in range $[0,1]$, (Fig. 1b).

The accuracy obtained by the restoration algorithm is measured by comparing the Signal to Noise Ratio (SNR) of the noisy image ($SNR(\mathbf{y}^\delta)$) and of the reconstructed image ($SNR(\mathbf{x}_k)$), where the SNR is defined as:

$$SNR(\mathbf{z}) = 20 \log \left(\frac{\|\mathbf{x}^*\|_F}{\|\mathbf{z} - \mathbf{x}^*\|_F} \right).$$

The following parameters have been used in the algorithm implementation.

The iterations of the CLSTV algorithm (Table 1) are stopped using the convergence rule (24) with tolerances $\tau_r = 10^{-4}$ and $\tau_a = 10^{-5}$ and $maxit_{CGLS} = 20$. The number of bisection iterations before starting the secant iterations in the hybrid method is $k_s = 2$.

The FP iterations (Table 2) are stopped using the convergence rule (19) with tolerances $\epsilon_1 = \epsilon_2 = 10^{-2}$ and $maxit_{FP} = 2$. The parameters for the TCG stopping criteria (18) are: $\zeta = 10^{-4}$ and $maxit_{CG} = 15$.

4.1 Algorithm accuracy

Aim of this paragraph is to show the results that can be obtained by CLSTV when γ is estimated by the formula (16). We report the results obtained with the two test images I1 and I2, blurred with both the GB and the MB kernels with different values of NL, by using the FP algorithm.

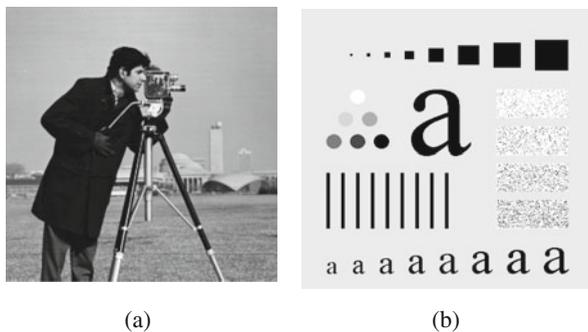


Fig. 1 Test images. **a** I1 : 256×256 test Image. **b** I2 : 500×500 test image

Table 4 Restoration results

Test	Blur	$SNR(\mathbf{y}^\delta)$	$SNR(\mathbf{x}_k)$	γ	λ_k	$k(it_{FP})$
I1	GB	18.8	25.7	3.337e+3	1.956e-4	5(12)
		17.6	21.2	2.879e+3	5.876e-3	6(10)
	MB	14.6	23.6	3.571e+3	1.379e-4	7(12)
		14.1	18.1	2.535e+3	4.943e-3	10(18)
I2	GB	17.4	21.4	1.8953e+4	1.2340e-4	13(24)
		16.4	18.7	1.2164e+4	1.2307e-2	8(14)
	MB	14.6	23.2	2.3007e+4	1.5010e-4	14(26)
		14.0	17.3	1.4429e+04	6.4490e-3	8(14)

The results obtained by adding noise with $NL=0.01$ and $NL = 0.07$ are reported in Table 4. By comparing the columns $SNR(\mathbf{y}^\delta)$ and $SNR(\mathbf{x}_k)$ it is possible to evaluate the improvement in the restored images (see Figs. 2, 3, 4 and 5 for GB and MB examples). The computational efficiency can be measured by the data in column $k(it_{CLSTV})$ reporting the CLSTV iterations (k) and the total number of FP iterations required by FP algorithm (it_{FP}). We observe that for the two different types of blur it is possible to efficiently obtain accurate solutions performing a low number of iterations in all the considered cases.

More insight in the algorithm convergence can be given by observing the behavior of the relative error at each iteration k :

$$\|\mathbf{x}_k - \mathbf{x}^*\|_F / \|\mathbf{x}^*\|_F. \quad (26)$$

In Fig. 6 we observe that the relative error obtained at the exit iteration k (black dot) is very close to the optimal (red star). This confirms the effectiveness of the stopping criteria (24) with the chosen values of the parameters τ_r and τ_a . In Fig. 7 we plot the λ_k sequence as a function of the iteration k . We observe that the value of the regularization parameter λ_k obtained at the exit iteration (black dot in Fig. 7) is close to that of minimum error (red star in Fig. 7). The plot of $\mathcal{R}(\mathbf{x}_k)$ vs. k is reported in Fig. 8 for the two blurring kernels. It shows that the starting value λ_0 together with (9) allows us to obtain a fast non monotone convergence of the $\{\lambda_k\}$ sequence.

Fig. 2 Test image I1: Gaussian blur. **a** Noisy blurred Image $SNR = 17.6$. **b** Restored Image $SNR = 21.2$



(a)

(b)

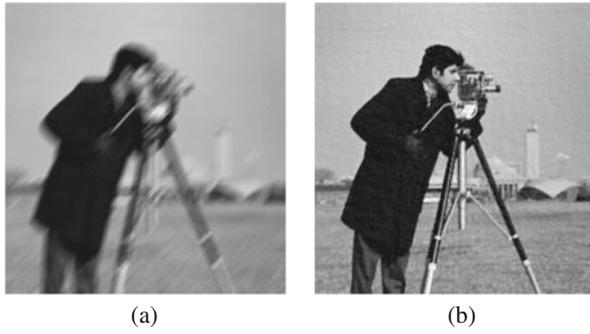


Fig. 3 Test image I1: motion blur. **a** Noisy blurred Image $SNR = 14.6$. **b** Restored Image $SNR = 23.6$

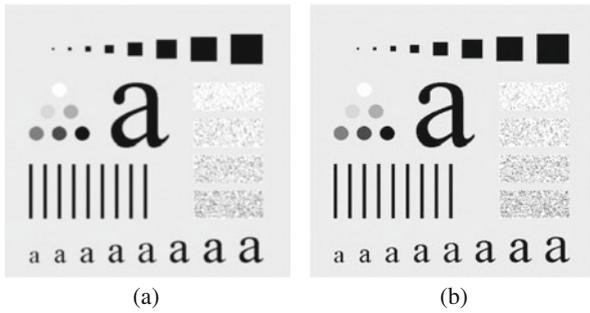


Fig. 4 Test image I2: Gaussian blur. **a** Noisy blurred Image $SNR = 17.4$. **b** Restored Image $SNR = 21.4$

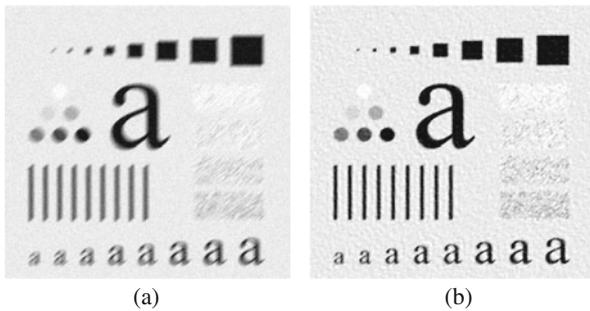


Fig. 5 Test image I2: motion blur. **a** Noisy blurred Image $SNR = 14.0$. **b** Restored Image $SNR = 17.3$

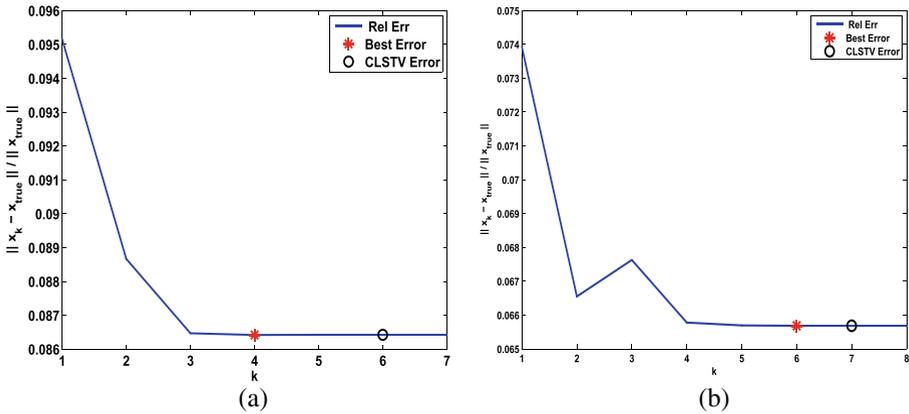


Fig. 6 I1 test image: relative error vs iterations k . **a** Gaussian blur $SNR = 17.6$. **b** Motion blur $SNR = 14.6$

4.2 Robustness of γ estimate

In this section we examine the effectiveness of the heuristic rule proposed for estimating γ , by comparing the results presented in the previous paragraph, where γ is computed with formula (16), with those obtained by using an *optimal* parameter γ (γ_{opt}). We heuristically found γ_{opt} as the value of γ minimizing the relative error (26).

For the sake of brevity we only show the results for the test image I1. In Table 5 are reported the parameters SNR_{opt} and λ_{opt} relative to the reconstructed images obtained from γ_{opt} . Comparing the columns SNR_{opt} and λ_{opt} , obtained with the optimal γ_{opt} , with the corresponding columns $SNR(x_k)$, and λ in Table 4, obtained

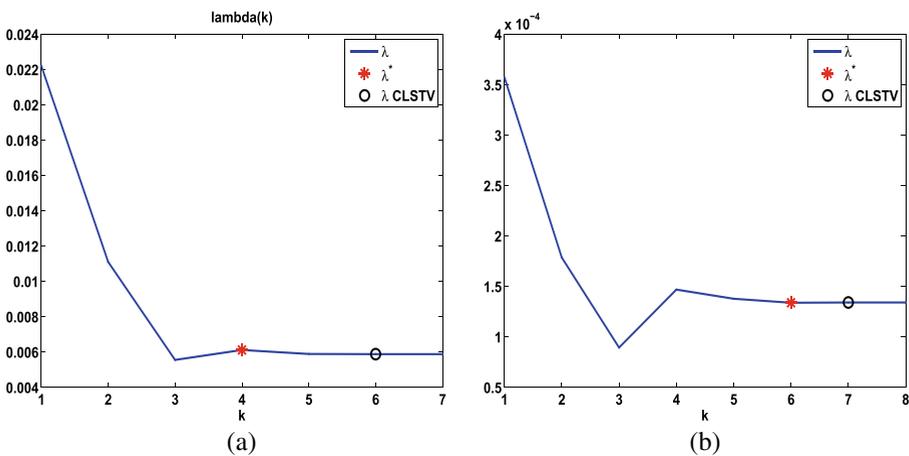


Fig. 7 I1 test image: regularization parameter λ_k vs iterations k . **a** Gaussian blur $SNR = 17.6$. **b** Motion blur $SNR = 14.6$

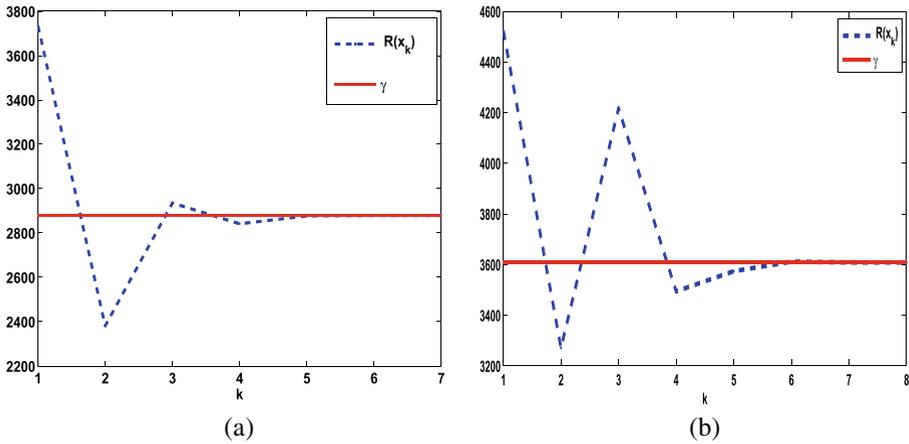


Fig. 8 11 test image: values of γ and $R(\mathbf{x}_k)$ at each iteration. **a** Gaussian blur $SNR = 17.6$. **b** Motion blur $SNR = 14.6$

with the computed γ , we can observe that the solutions computed with γ_{opt} and with the computed γ are very close.

In Fig. 9 we plot the SNR values of the reconstructions obtained with the different values of γ for the two blurring kernels, in the case of low noise. The red dots represent the SNR values of the solutions computed using the values of γ estimated by (16). The dots are in both cases very close to the optimal SNR values, i.e. the maxima of the plotted curves; hence our estimates of γ are accurate. In order to validate the formula (16) for the estimation of γ , we report in Table 6 the values of γ_L and γ_H and the parameter θ used to obtain the results in Table 4. Finally we observe that the interval $[\gamma_L, \gamma_H]$ always includes the optimal value γ_{opt} reported in Table 5 and that larger values of the noise in the data require smaller values of the weight θ . We remind that we computed the values γ_L and γ_H by estimating an interval containing the value γ_{opt} for a data base of grayscale images.

4.3 Estimate of the regularization parameter γ

In this test, we solve the problem

$$\min_x F(\mathbf{x}) + \lambda J(\mathbf{x}) \tag{27}$$

Table 5 Test image 11, restoration with optimal parameters

Blur	$SNR_{\gamma^{\delta}}$	SNR_{opt}	γ_{opt}	λ_{opt}
GB	18.8	25.7	3.4182e+3	1.6681e-4
	17.6	21.3	2.8852e+3	5.8570e-3
MB	14.6	23.7	3.6299e+03	1.3219e-04
	14.1	18.2	2.8043e+03	3.5112e-03

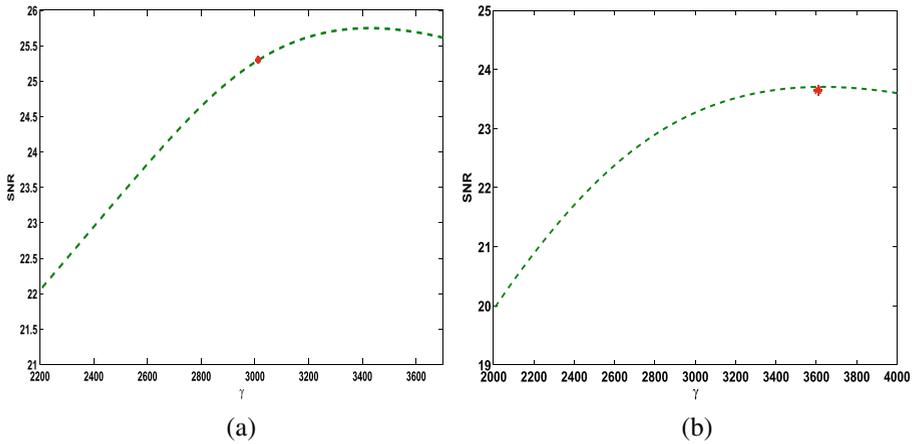


Fig. 9 Test image I1: SNR vs γ (red dots: CLSTV algorithm). **a** Gaussian Blur ($SNR = 18.8$). **b** Motion blur ($SNR = 14.6$)

for several given values of the regularization parameter λ and choose the solution that minimizes the relative error (26). By solving (27) with 50 values in the interval $[10^{-4}, 10^{-1}]$ for illustration purpose we obtain a SNR curve that reaches its maximum value very close to the value obtained by CLSTV method (see Fig. 10a and b). The value of the regularization parameters computed by the CLSTV algorithm are $\lambda = 0.0139$ with $SNR = 17.3$ and $\lambda = 0.0383$ with $SNR = 13.3$. The values of the regularization parameters that minimize the relative error are $\lambda_{opt} = 0.0121$ and $\lambda_{opt} = 0.0429$ respectively.

4.4 Comparison with other algorithms

In this paragraph we compare some aspects of the CLSTV algorithm with different deblurring algorithms given in the literature.

In the first test we compare the performance of our algorithm with the method proposed in [17] called SATV. It is a sophisticated method that produces very good quality deblurred images by automatically computing spatially adapted regularization parameter. For this reason, we use the code available at <http://www.uni-graz.at/imawww/ifb/sa-tv/index.html> implementing the SATV method, and we use the same

Table 6 Parameters used in (16) for estimating γ in the I1 test problem

$SNR (y^\delta)$	γ_L	γ_H	θ
18.8	2.021093e+2	7.168144e+3	0.45
17.6	5.252854e+2	1.863012e+4	0.13
14.6	3.149450e+2	1.117005e+4	0.3
14.1	5.704799e+2	2.023302e+4	0.1

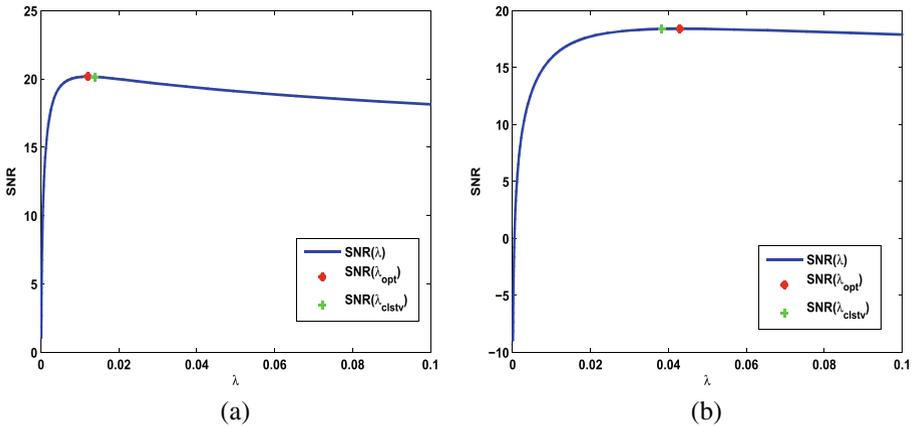


Fig. 10 Test image I1: SNR curve for fixed regularization parameters in the interval $[10^{-4}, 10^{-1}]$. **a** $SNR_{blurred} = 17.6$. **b** $SNR_{blurred} = 14.6$

test problem implemented there. The test image is I1, previously described in Fig. 1, while the blurring kernel is a Gaussian function, generated by SATV code. The noise is added with absolute perturbation $\delta = \|\mathbf{y} - \mathbf{y}^\delta\|_F = 0.05, 0.1$, obtaining severely blurred noisy images with $SNR = 17.3$ and $SNR = 13.3$ respectively.

In Table 7 we report the results obtained with different noise levels. We observe that SATV and CLSTV algorithms give close qualitative results in terms of SNR (columns 4 and 6 in Table 7). Although the images Fig. 11a and b are very close together, by observing the error images in Fig. 12a and b we can see a smoother behavior of CLSTV. The great difference between the two algorithms is given by the computation time. In Table 7 (columns 5 and 7) we report the elapsed time. The ratio between the elapsed time of algorithm CLSTV and SATV is less than 2 %.

In the second test we wish to compare the effectiveness of the CLSTV procedure for determining the regularization parameter when different methods are used to solve the inner problem (10). In particular we compare the Fixed Point method of Table 2 (CLSTV + FP) with the splitting method given in Table 3 (CLSTV + SP). In the reported experiments the denoising step (21)

Table 7 Comparison between CLSTV (algorithm 1) and SATV (Dong et al. [17]) algorithms. Results computed with test image I1 blurred with gaussian kernel and different noise values. Times obtained on 8 intel i7 processor and 24 GB Ram

Test	Noise		CLSTV		SATV	
	δ	$SNR(\mathbf{y}^\delta)$	$SNR(\mathbf{x}_k)$	sec	$SNR(\mathbf{x}_k)$	sec.
I1	0.05	17.3	20.3	11	20.2	828
	0.1	13.3	18.6	12	18.5	745

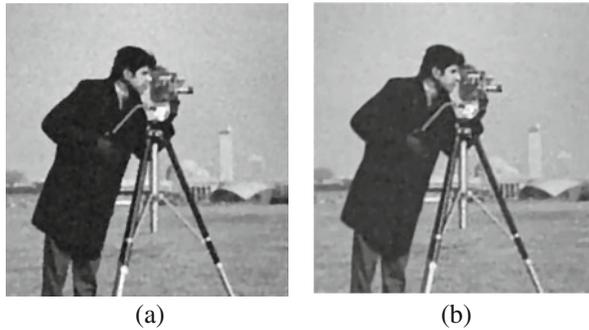


Fig. 11 Test image I1: restored images. **a** CLSTV. **b** SATV

is performed by the Chambolle's algorithm [10]. The computed results are obtained using the function `tvdenoise.m` of the TVreg code available at <http://www.mathworks.com/matlabcentral/fileexchange/29743-tvreg-variational-image-restoration-and-segmentation>.

In Table 8 we report the qualitative (SNR) and performance (computation times) results obtained with test image I1 blurred with gaussian and motion blur as in Table 4 (column 3). The estimated parameter γ is not affected by the solution of (10) as shown in Table 8 (columns 3 and 6). In the (CLSTV + SP) experiment the maximum number of iterations $maxit_{SP} = 10$ is used. We noticed that a larger value of $maxit_{SP}$ would give better SNR values and increase the computation times. From the results reported in Table 8 we can conclude that good quality restorations (columns 2 and 6) can be achieved by both methods at comparable computation times (columns 5 and 9). Therefore CLSTV can be considered a suitable general framework for computing the regularization parameter of different deblurring algorithms. We observe that the number of inner iterations of the denoising algorithm it_j (column 8) cannot be compared with the number of FP inner iterations (column 4) since the latter have a much higher computational cost.

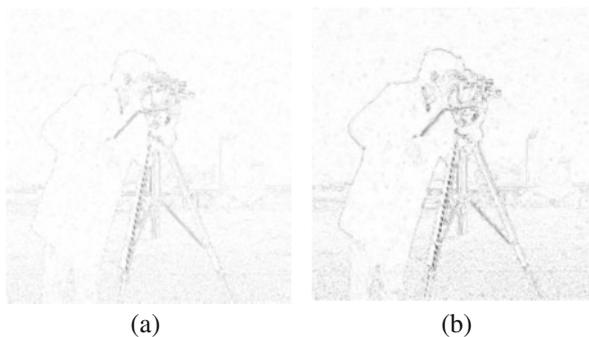


Fig. 12 Test image I1: negative error images. **a** CLSTV. **b** SATV

Table 8 Comparison between CLSTV+ FP and CLSTV+ SP. Results computed with test image I1 blurred with gaussian and motion kernels and different noise values. Times obtained on intel i5 processor and 6 GB Ram

Blur	CLSTV+ FP				CLSTV+ SP			
	SNR(\mathbf{x}_k)	γ	$k(it_{FP})$	sec.	SNR(\mathbf{x}_k)	γ	$k(it_j)$	sec.
GB	25.7	3.337e+3	5(12)	30.01	25.44	3.337e+3	7(4273)	47.14
	21.2	2.879e+3	6(10)	20.85	21.09	2.879e+3	10(2277)	28.58
MB	23.6	3.571e+3	7(12)	36.38	22.14	3.571e+3	10(3099)	35.23
	17.3	2.535e+3	10(18)	25.80	18.14	2.535e+3	10(3099)	29.66

5 Conclusions

The paper presents the CLSTV algorithm in image deblurring applications to compute both the restored image and an optimal regularization parameter simultaneously. The method is based on the solution of a constrained optimization problem, where the TV function is the smoothing constraint. The algorithm estimates the smoothness level γ from the given data.

The numerical experiments show the effectiveness of CLSTV: good quality restored images are obtained with different blurring kernels and noise levels. These images are comparable with those obtained by more sophisticated algorithms but the CLSTV algorithm is much faster (saving up to 98 % of the computational time with respect to SATV), as shown in Section 4.4. The computational efficiency of the algorithm makes it suitable to large scale problems such as large size multichannel images.

We observe that CLSTV defines a general for computing a good regularization parameter for deblurring images using different methods for solving (10).

In future works, the method could be extended to the other types of linear inverse problems, such as the reconstruction of tomographic data and blind deconvolution as well as to nonquadratic discrepancy functionals such as, for example, the Kullback-Leibler (or Csiszàr I-divergence).

References

1. Acar, R., Vogel, C.R.: Analysis of bounded variation penalty methods for ill-posed problems. *Inverse Problems* **10**, 1217–1229 (1994)
2. Afonso, M., Boiucas-Dias, J., Figueiredo, M.: An augmented lagrangian approach to the constrained optimization Formulation of Imaging Inverse Problems. *IEEE Trans. Image Process.* **20**(3), 681–695 (2011)
3. Afonso, M., Boiucas-Dias, J., Figueiredo, M.: Fast image recovery using variable splitting and constrained optimization. *IEEE Trans. Image Process.* **19**(9), 2345–2356 (2010)
4. Bardsley, J., Goldes, J.: Regularization parameter selection and an efficient algorithm for total variation-regularized positron emission tomography. *Num. Alg.* **57**, 255–271 (2011)
5. Blomgren, P., Chan, T., Color, T.V.: Total variation methods for restoration of vector-valued images. *IEEE Trans. Image Process.* **7**(3), 304–309 (1998)

6. Bredies, K., Kunisch, K., Pock, T.: Total generalized variation. *SIAM J. Imaging Sci.* **3**(3), 492–526 (2010)
7. Brito, C., Chen, K.: Multigrid algorithm for high order denoising. *SIAM J. Imaging Sci.* **3**(3), 363–389 (2010)
8. Buades, A., Coll, B., Morel, J.: A review of image denoising algorithms with a new one. *SIAM Multiscale Modeling Simul.* **4**(2), 490–530 (2005)
9. Chambolle, A., Lions, P.L.: Image recovery via total variation minimization and related problems. *Numer. Math.* **76**, 167–188 (1997)
10. Chambolle, A.: An algorithm for total variation minimization and applications. *J. Math. Imag. Vis.* **20**, 89–97 (2004)
11. Chan, T., Osher, S.: The digital TV filter and nonlinear denoising. *IEEE Trans. Image Process.* **10**(2), 231–241 (2001)
12. Chan, T.F., Shen, J.: *Image processing and analysis: variational, PDE, wavelet and stochastic methods.* SIAM (2005)
13. Chan, T., Tai, X.: Level set and total variation regularization for elliptic inverse problems with discontinuous coefficients. *J. Comp. Phys.* **193**, 40–66 (2003)
14. Chan, R., Chen, K.: A multilevel algorithm for simultaneously denoising and deblurring images. *Siam J. Sci. Comp.* **32**(2), 1043–1063 (2010)
15. Chan, T., Chen, K.: An optimization-based multilevel algorithm for total variation image denoising. *Multiscale Model. Simul.* **5**, 615–645 (2006)
16. Dembo, R., Steihaug, T.: Truncated-Newton algorithms for large-scale unconstrained optimization. *Math. Program.* **26**, 190–212 (1983)
17. Dong, Y., Hintermüller, M., Rincon-Camacho, M.M.: Automated regularization parameter selection in multi-scale total variation models for image restoration. *J. Math. Image Vis.* **40**(1), 83–104 (2011)
18. Hanke, M.: Regularizing properties of a truncated Newton-CG algorithm for nonlinear inverse problems. *Num. Func. Anal. Optim.* **18**, 971–993 (1997)
19. Hanke, M., Nagy, J.G., Plemmons, R.J.: Preconditioned iterative regularization for ill-posed problems. In: Reichel, L., Ruttan, A., Varga, R.S. (eds.) *Numerical Linear Algebra*, pp. 141–163. de Gruyter, Berlin (1993)
20. Hansen, P.C., Nagy, J.G., O’Leary, D.P.: *Deblurring Images: Matrices, Spectra, and Filtering.* SIAM publications, Philadelphia (2006)
21. Hansen, P.C. et al.: Algorithms and software for total variation image reconstruction via first-order methods. *Num. Alg.* **53**(1), 67–92 (2010)
22. Huang, Y., Ng, M., Wen, Y.: A fast total variation minimization method for image restoration. *Multiscale Model. Simul.* **7**, 774–795 (2008)
23. Martin, D.R., Reichel, L.: Projected Tikhonov regularization of large-scale discrete ill-posed problems. *J. Sci. Comput.* **56**, 471–493 (2013)
24. Abad, J.O., Morigi, S., Reichel, L., Sgallari, F.: Alternating Krylov subspace image restoration methods. *J. Comput. Appl. Math.* **236**, 2049–2062 (2012)
25. Loli Piccolomini, E., Zama, F.: A descent method for the regularization of ill-posed problems. *Optim. Methods Softw.* **20**(4–5), 615–628 (2005)
26. Loli Piccolomini, E., Zama, F.: An iterative algorithm for large size least-squares constrained regularization problems. *Appl. Math. Comput.* **217**, 10343–10354 (2011)
27. Reichel, L., Rodriguez, G.: Old and new parameter choice rules for discrete ill-posed problems. *Num. Alg.* **63**, 65–87 (2013)
28. Rudin, L., Osher, S., Fatemi, E.: Nonlinear total variation based noise removal algorithms. *Physica D* **60**, 259–268 (1992)
29. Tadmor, E., Nezzen, S., Vese, L.: Multiscale hierarchical decomposition of images with applications to deblurring, denoising and segmentation. *Comm. Math. Sci.* **6**, 1–26 (2008)
30. Vogel, C.R., Oman, M.E.: Fast, robust total variation-based reconstruction of noisy, blurred images. *IEEE Trans. Image Processing* **7**, 813–824 (1998)
31. Vogel, C.R.: *Computational Methods for Inverse Problems.* SIAM Publications, USA (2002)
32. Wen, Y., Chan, R.: Parameter selection for total variation based image restoration using discrepancy principle. *IEEE Trans. Image Proc.* **21**(4), 1770–1781 (2012)
33. Wen, Y., Ng, M., Ching, W.K.: Iterative algorithms based on decoupling of deblurring and denoising for image restoration. *Siam J. Sci. Comp.* **30**, 2655–2674 (2008)
34. Zhang, J.P., Chen, K., Yu, B.: An iterative lagrange multiplier method for constrained total-variation-based image denoising. *SIAM J. Numer. Anal.* **50**(3), 983–1003 (2012)