J Math Imaging Vis
DOI 10.1007/s10851-015-0599-3

Iterative Constrained Minimization for Vectorial TV Image Deblurring

K. Chen[1] · **E. Loli Piccolomini**[2] · **F. Zama**[2]

Received: 18 September 2014 / Accepted: 28 August 2015

Abstract In this paper, we consider the problem of restoring blurred noisy vectorial images where the blurring model involves contributions from the different image channels (cross-channel blur). The proposed method restores the images by solving a sequence of quadratic constrained minimization problems where the constraint is automatically adapted to improve the quality of the restored images. In the present case, the constraint is the Total Variation extended to vectorial images, and the objective function is the ℓ_2 norm of the residual. After proving the convergence of the iterative method, we report the results obtained on a wide set of test images, showing that this approach is efficient for recovering nearly optimal results.

Keywords Vectorial images · Deblurring · Vectorial total variation · Regularization algorithm · Constrained minimization

1 Introduction

Vectorial images can be obtained in various situations, such as from multiple time frames, multiple frequency bands, digital sensors, etc. They occur in many modern digital applications, involving videos, sensors, medical instruments for diagnosis, etc [4,13]. The most widespread vectorial images are certainly multichannel true color images in which the color information of each pixel is represented by the weights of the three primary colors: red, green, and blue (RGB images).

In the continuous setting, the multichannel restoration is modeled by

$$y = Hu + \zeta^{\delta} \tag{1}$$

where H is a Fredholm integral operator of a matrix form, u and y are vectorial image functions defined in a domain $\Omega \subset \mathbb{R}^2$ s.t. $u, y : \Omega \to \mathbb{R}^p$ (p is the number of channels) and ζ^{δ} is the function modeling the noise of level δ. In this paper, we consider the realistic case of cross-channel blur, meaning that each pixel is blurred with the contribution of the pixels in all the image channels, as opposed to within channel blur, where the blur is independent in each channel.

Since (1) is an ill-posed problem, its solution is defined by means of regularization methods which consist in substituting (1) by an optimization problem.

It is quite common to define regularization methods by means of an unconstrained minimization problem of the form:

$$\min_u \frac{1}{2}\|Hu - y\|_2^2 + \lambda R(u) \tag{2}$$

where $\frac{1}{2}\|Hu - y\|_2^2$ is the l2-norm function measuring the fit to the observed data, $R(u)$ is a regularization function controlling the smoothness of the solution, and λ is the regularization parameter.

In this paper, $R(u)$ is the Total variation (TV) function [14,24] extended to vectorial images [6].

The result of the restoration process strongly depends on the parameter λ; a good value of λ is related to some feature of

✉ E. Loli Piccolomini
elena.loli@unibo.it

K. Chen
http://www.liv.ac.uk/cmit

[1] Department of Mathematical Sciences, Centre for Mathematical Imaging Techniques (CMIT), The University of Liverpool, Liverpool, UK

[2] Department of Mathematics, University of Bologna, Piazza di Porta S. Donato 5, 40126 Bologna, Italy

Published online: 14 September 2015

the processed image, such as its scale [23]. Hence, its choice is critical.

There is an extensive literature that treats the choice of the parameter λ in the case of grayscale or multichannel images blurred by within channel blur.

Some methods are completely heuristic and are based on trial-and-error algorithms. Among them, the L-curve [15] and the GCV [18], or GCV based [17] methods are widely used. Their main disadvantages are the high computational complexity, that is prohibitive for multichannel images, and the fact that sometimes they totally fail. A different approach uses the noise norm or its approximation. Examples are the discrepancy principle [19], and some algorithms based on the Lagrange multipliers method applied to the following constrained problem:

$$\min_u R(u) \quad s.t. \quad \|Hu - y\|_2^2 \leq \sigma \tag{3}$$

where σ is the noise norm [2,4,28]. The main drawback of these methods is that they need an estimate of the noise norm. If an accurate estimate of the noise variance is available, the SURE-based methods [21] are very efficient algorithms for the choice of regularization parameters [20]. Another statistical approach for grayscale images is analyzed in [3]. An exhaustive survey of the existing methods for the selection of the regularization parameter in image deblurring applications can also be found in [1,28,30]. Nevertheless, the research in this field is still active, because the existing methods are not completely satisfactory or their use is limited to particular applications.

In the case of multichannel images, blurred by cross-channel blur, some algorithms have been presented for the efficient solution of problem (2) with a predetermined value of λ (see [25,26] and references therein). However, in the case of cross-channel blur, there are far fewer papers that deal with the problem of computing the parameter λ. For what concerns the application of interest in our study, only [10,13] and [1] address this problem. While the former uses Tikhonov regularization function, the latter uses the TV function proposing a multiscale approach where λ is a function of u ([9,10]). They achieve good quality results at the expense of a high computational cost. The method recently proposed in [1] defines a criterion for selecting the regularization parameter based on an estimate of the residual whiteness.

1.1 Contribution

The main contribution of this paper is to propose a regularization algorithm, for deblurring multichannel images affected by cross-channel blur, using only the observed image as data input. The idea is of solving (1) by a finite sequence of constrained optimization problems

$$\min_u \frac{1}{2}\|Hu - y\|_2^2 \ s.t. \ R(u) \leq \gamma_j, \qquad j = 0, 1, \ldots t \tag{4}$$

where the regularization constraints γ_j are adaptively updated. The constrained formulation (4) is equivalent to the unconstrained formulation (2) when $\frac{1}{\lambda}$ is the Lagrange multiplier of (4).

Each problem (4) is solved in Lagrangian dual form extending the idea proposed in [8] for gray-level images. The Lagrange multipliers are computed by the same root-finding procedure, but the starting value of the sequence has been modified, adapting the filtering procedure to the case of multichannel images with cross-channel blur. Finally, compared to [8], the procedure to compute the regularization weights γ_i has changed significantly, and a new one, better suited to multichannel images, is presented.

The proposed algorithm is described in a modular and general form so that it can be easily applied to different fit-to-data and regularization functions satisfying the required hypotheses for the convergence.

In the numerical experiments, we extensively test the method on color images perturbed by cross-channel blur of different types, and we compare our method with the method proposed in [1].

1.2 Outline of the Paper

In Sect. 2, some theory about the vectorial TV model is given; in Sect. 3, the proposed algorithm is presented; in Sect. 4, we show some numerical experiments; and finally in Sect. 5, the conclusions are reported.

2 Theory of the Vectorial TV Model

Consider processing of 2D vectorial images with p channels; where $p = 3$ in case of true color images. Let the given image $y = (y_1, \ldots, y_p)$ and the restored image $u = (u_1, \ldots, u_p)$ be defined in a bounded open domain $\Omega \subset \mathbb{R}^2$, where $y \in L^2(\Omega; \mathbb{R}^p)$. To define the space for u, we have to define a dual variable $q = (q_1, \ldots, q_p)$ in some set $P \subset C_c^1(\Omega)$ of compactly supported functions.

Definition 1 ([6]) A suitable space for u in (7) is the vector space $\mathrm{BV}(\Omega; \mathbb{R}^p)$ of bounded variations which is as a subspace of $L^1(\Omega; \mathbb{R}^p)$ equipped with the norm

$$\|u\|_{BV} = \|u\|_{L^1} + |u|_{BV},$$

where the semi-norm is defined by

$$|u|_{BV} = \int_\Omega |Du| d\Omega = \sup_{q \in P} \left\{ \int_\Omega \langle u, \nabla \cdot q \rangle d\Omega \right\}$$

with $P = \{q \in C_c^1(\Omega) \mid |q|_2 \leq 1\}$.

For $u \in C^1(\Omega)$, the semi-norm $|u|_{BV}$ is simplified to the following vectorial total variation (VTV):

$$\mathrm{VTV}(u) = |u|_{\mathrm{BV}} = \int_\Omega \sqrt{\sum_{i=1}^{p} |\nabla u_i|^2} d\Omega. \tag{5}$$

This is not the only possible extension of the TV function to multichannel images (see [6,7] and references therein); our preference is due to the computational advantages.

In this section, we consider the constrained problem:

$$\min_u \frac{1}{2}\|Hu - y\|_2^2 \ \text{ s.t. } \ \mathrm{VTV}(u) \le \gamma, \tag{6}$$

that can be stated in the equivalent form:

$$\max_\lambda \min_u \left\{ E\big(\lambda, u(\lambda)\big) = \frac{1}{2}\big\|Hu - y\big\|_2^2 + \lambda\,(\mathrm{VTV}(u) - \gamma) \right\} \tag{7}$$

Now we discuss theoretical properties of this model used to explore an optimal choice of λ.

Theorem 1 *([6]) For a fixed λ, the vectorial TV model (7) has a solution in $BV(\Omega; \mathbb{R}^p)$.*

We remark that the presence of a blurring operator does not pose any complications in modifying the proof in [6, Theorem 2.2] to prove Theorem 1.

Theorem 2 *The vectorial TV model (7) has a unique solution in $BV(\Omega; \mathbb{R}^p)$ if the blur operator H is injective.*

The proof can follow that of [8, Lemma 2]. However, for a continuous formulation, H is compact so not injective; practically and fortunately when discretized, H may be regarded as invertible and injective as far as the uniqueness is concerned.

Theorem 3 *For $\lambda \ge 0$, the function $F(\lambda) = \mathrm{VTV}(u(\lambda)) - \gamma$ in (7) is monotonically decreasing in a feasible region, and there exist two distinct parameters λ_1, λ_2 such that $F(\lambda_1) > 0$ and $F(\lambda_2) < 0$. Hence, $F(\lambda)$ has a unique root $\hat{\lambda}$ such that $F(\hat{\lambda}) = 0$, i.e., $\mathrm{VTV}(u(\hat{\lambda})) - \gamma = 0$, where $\hat{\lambda}$ is the optimal parameter for the problem (6).*

Proof For two given parameters λ, μ such that $\lambda > \mu \ge 0$, we first prove that $F(u(\lambda)) < F(u(\mu))$.

Using Theorem 2, denote the respective minimizers for $\min_u E(\lambda, u)$ and $\min_u E(\mu, u)$ by $u(\lambda), u(\mu)$. Then we have

$$E\big(\lambda, u(\lambda)\big) < E\big(\mu, u(\lambda)\big), \quad E\big(\mu, u(\mu)\big) < E\big(\lambda, u(\mu)\big).$$

Here the strict inequality comes from Theorem 2 as Theorem 1 would only give $\le$. The above inequalities read

$$\begin{aligned}
\frac{1}{2}\big\|Hu(\lambda) - y\big\|_2^2 + \lambda F(u(\lambda)) &< \frac{1}{2}\big\|Hu(\lambda) - y\big\|_2^2 \\
&\quad + \mu F\big(u(\lambda)\big), \\
\frac{1}{2}\big\|Hu(\mu) - y\big\|_2^2 + \mu F\big(u(\mu)\big) &< \frac{1}{2}\big\|Hu(\mu) - y\big\|_2^2 \\
&\quad + \lambda F(u(\mu)).
\end{aligned}$$

Subtracting the two inequalities gives the desired result $(\lambda - \mu)(F(u(\lambda)) - F(u(\mu))) < 0$, i.e., $F(u(\lambda)) < F(u(\mu))$, since $(\lambda - \mu) > 0$.

Take $\lambda_1 = 0$ as a lower bound (as $\lambda = 0$ is not the optimal parameter) and the solution $u(\lambda_1) = H^\dagger y$ (where $H^\dagger$ is the pseudoinverse of H) will still be close to the given image f: $\mathrm{VTV}(f) \approx \mathrm{VTV}(u(\lambda_1)) > \gamma$ as γ is chosen to model the restored and the noise-free image. So $F(u(\lambda_1)) = \mathrm{VTV}(u(\lambda_1)) - \gamma > 0$. For the upper bound, from

$$\begin{aligned}
u\big(\lambda_2\big) &= \lim_{\lambda\to\infty} \operatorname*{argmin} E\big(\lambda, u\big) = \operatorname*{argmin}_u \mathrm{VTV}\big(u(\lambda)\big) - \gamma \\
&= \operatorname*{argmin}_u \mathrm{VTV}\big(u(\lambda)\big)
\end{aligned}$$

we see that $\mathrm{VTV}(u(\lambda_2)) = 0$ so $F(u(\lambda_2)) = \mathrm{VTV}(u(\lambda_2)) - \gamma < 0$. Clearly, there exists a positive number $\hat{\lambda}$, lying between λ_1 and λ_2, so that $F(u(\hat{\lambda})) = 0$. □

3 The Multichannel Constrained Least Squares TV algorithm (MCLSTV)

In this section, we describe our `MCLSTV` algorithm for restoring multichannel images. We first define the discrete model of (1) in the particular case considered in this paper, and then we deal with the discretized version of the regularization method (6) for solving the deblurring problem.

In order to estimate a suitable value for the constraint parameter γ, we propose to solve a finite sequence of constrained optimization problems of the form (6), with the parameters $\gamma = \gamma_i, i = 1, \ldots t$, adaptively computed. For the solution of each constrained problem, we extend to multichannel images the algorithm proposed in [8].

3.1 Discretized Problem

The continuous equation (1) is very general.

In this paper, we consider deblurring problems of the form

$$y_n(x, y) = \sum_{k=1}^{p} w_{n,k} \iint h_{n,k}(x - \xi, y - \eta) u_k(\xi, \eta) d\xi d\eta, \quad n = 1, \ldots, p \tag{8}$$

where $0 \le w_{n,k} < \infty$ are the weights with the property

$$\sum_{k=1}^{p} w_{n,k} = 1, \quad \forall n.$$

By taking $N \times M$ samples for the variables (ξ, η) as well as (x, y) and sampling the integrals by means of the rectangle rule with samples spacing equal to 1, we can rewrite (8) as

$$y_n(x_\ell, y_m) = \sum_{k=1}^{p} w_{n,k}\left(\sum_{j=1}^{M}\sum_{i=1}^{N} h_{n,k}(x_\ell - \xi_i, y_m - \eta_j)u_k(\xi_i, \eta_j)\right), \quad n = 1, \ldots, p$$

where $\ell = 1, \ldots, N, m = 1, \ldots, M$.

Reorganizing the data as follows:

$$Y = (Y_1, Y_2, \ldots, Y_p)^T, \quad \text{s.t. } Y_n \in \mathbb{R}^{N\times M}, n = 1, \ldots, p \text{ and } Y_n(\ell, m) \equiv y_n(x_\ell, y_m)$$

$$U = (U_1, U_2, \ldots, U_p)^T, \quad \text{s.t. } U_k \in \mathbb{R}^{N\times M}, k = 1, \ldots, p \text{ and } U_k(i, j) \equiv u_k(\xi_i, \eta_j)$$

we can represent the blurring process as a matrix vector product $Y = \mathcal{H}U$. The blurring matrix $\mathcal{H}$ has $p \times p$ blocks $H_{n,k} \in \mathbb{R}^{N\times M}, (n, k = 1, \ldots, p)$

$$\mathcal{H} = \begin{pmatrix} H_{1,1} & H_{1,2} & \cdots & H_{1,p} \\ H_{2,1} & H_{2,2} & \cdots & H_{2,p} \\ \vdots & \vdots & \ddots & \vdots \\ H_{p,1} & H_{p,2} & \cdots & H_{p,p} \end{pmatrix},$$

$$H_{n,k} \equiv w_{n,k}h_{n,k}(x_\ell - \xi_i, y_m - \eta_j). \tag{9}$$

Within-channel blur is represented by the diagonal blocks $H_{k,k}, k = 1 \ldots, p$. The blocks $H_{n,k}, n \neq k$, represent how the blur on the n-th channel influences the k-channel, i.e., the cross-channel blur. By supposing periodic boundary conditions for U, the matrices $H_{i,j}$ are all block circulant with circulant blocks (BCCB) [26].

For each channel $n, n = 1, \ldots p$, we can compute the n-th channel of the blurred image as a sum of convolutions:

$$Y_n = \sum_{k=1}^{p} H_{n,k} * U_k. \tag{10}$$

Assuming that the observed data Y^δ are affected by additive noise of level δ, we have:

$$Y^\delta = Y + \delta\eta, \tag{11}$$

where $\|\eta\|_2 = 1$.

It is well known that recovering U from (10) and (11) is an ill posed problem and some regularization scheme is necessary. We define the regularization function, multichannel total variation (MTV), by discretizing the VTV function (5):

$$\mathrm{MTV}(U) = \sum_{j=1}^{M}\sum_{i=1}^{N}\left(|\nabla U_1|^2_{i,j} + |\nabla U_2|^2_{i,j} + \cdots + |\nabla U_p|^2_{i,j}\right)^{1/2}$$

where $|\nabla V|^2 = V_x^2 + V_y^2$ is obtained approximating the derivatives by means of forward difference formulas.

We solve the discretized version of (6):

$$\operatorname*{argmin}_U \left\|\mathcal{H}U - Y^\delta\right\|_2^2 \text{ s.t. } \mathrm{MTV}(U) \leq \gamma. \tag{12}$$

With these definitions in mind, we solve the ill posed problem $Y^\delta = \mathcal{H}U$ by a sequence of discrete constrained regularization problems of the form:

$$\min_U \frac{1}{2}\left\|\mathcal{H}U - Y^\delta\right\|_2^2 \text{ s.t. } \mathrm{MTV}(U) \leq \gamma_j, \quad j = 1, 2 \ldots t. \tag{13}$$

3.2 Computation of the Sequence γ_j

In the following, we call V the solution of the noise free problem $\mathcal{H}V = Y$, $U(\gamma_j)$ the solution of the constrained problem (13) and $\tilde{\gamma}$ the solution of the problem:

$$\tilde{\gamma} = \operatorname*{argmin}_\gamma \left\|U(\gamma) - V\right\|_2^2$$

(hence, $\tilde{U} = U(\tilde{\gamma})$). Let's define the residual vector $R(\gamma) = Y^\delta - \mathcal{H}U(\gamma)$, where Y^δ is given in (11), and the residual norm $r(\gamma) = \|R(\gamma)\|$ as a function of γ.

We propose to compute a sequence $\{\gamma_j\}$, $j = 0, 1, \ldots t$ with the aim that γ_t approaches $\tilde{\gamma}$.

Proposition 1 *Given the sequence γ_j s.t. $0 < \gamma_0 \leq \gamma_1 \leq \cdots \leq \tilde{\gamma}$ then*

$$\gamma_j \leq \gamma_{j+1} \leq \tilde{\gamma} \quad \Rightarrow \quad r(\gamma_j) \geq r(\gamma_{j+1}) \tag{14}$$

Proof Let $U(\gamma_j)$ and $U(\gamma_{j+1})$ be the solutions of (13) with $\gamma = \gamma_j$ and $\gamma = \gamma_{j+1}$ and let λ_j and λ_{j+1} be the related Lagrange multipliers. In theorem 3, it is proved that $\mathrm{MTV}(U)$ is monotonically decreasing in a feasible region; hence, in that region if $\gamma_j < \gamma_{j+1}$ then $\lambda_j > \lambda_{j+1}$. On the other hand it is also well known that the residual norm is increasing with λ, i.e., if $\lambda_j > \lambda_{j+1}$ then $r(\gamma_j) > r(\gamma_{j+1})$ [14]. Hence, (14) holds. □

It is experimentally observed that the residual function $r(\gamma)$ is steeply decreasing when $\gamma \ll \tilde{\gamma}$ and becomes flat around $\tilde{\gamma}$. Therefore we assume that in a suitable interval $[\gamma_0, \gamma_\ell]$, $\gamma_\ell < \tilde{\gamma}$, the rate at which the residual curve decreases has the following behavior:

$$\left|r'(\gamma_j)\right| > \left|r'(\gamma_{j+1})\right| \quad \text{if } \gamma_j \leq \gamma_{j+1}$$

Hence, we can define a tolerance parameter $\alpha > 0$ and stop the sequence γ_j, $j = 0, 1, \ldots$ as soon as $|r'(\gamma_j)| < \alpha$.

Let's assume that we are given a starting value γ_0 s.t. $0 < \gamma_0 \ll \hat{\gamma}$ and that the derivative of the residual norm $r'(\gamma)$ is continuous. Then assigning the parameters $\alpha > 0$, $P > 0$ and $s \in (0, 1)$, we define the sequence $\{\gamma_j\}_{j=1}^{t}$ s.t. $|r'(\gamma_t)| \leq \alpha$ as follows:

Algorithm 1 (INPUT γ_0; OUTPUT γ_t)

```
j=0;
while r'(γ_j) < −α
        j = j + 1;
        γ_j = γ_{j−1}(1 + P)
end
k = j
if r'(γ_k) < α
        γ_t = γ_k
else
        a = γ_{k−1}, b = γ_k
        compute γ_t =BCKTRK(a, b).
end
```

The steps of the backtracking procedure are reported in Algorithm 2.

Algorithm 2 BCKTRK (input(a,b); OUTPUT(γ,$U(\gamma)$))

```
s = 0.5
repeat
        γ = a + (b − a)s
        Compute U(γ) solving min_U J(U), s.t. MTV(U) ≤ γ
        if r'(γ) < −α
              a = γ
        else
              b = γ
        end
until |r'(γ)| ≤ α
```

Since $\gamma_0 \leq \tilde{\gamma}$ then $r(\gamma_0) > r(\tilde{\gamma})$ (Proposition 1). Assume that $r'(\gamma_j) < -\alpha, \quad j = 1, 2, \ldots, k$ and $r'(\gamma_{k+1}) \geq -\alpha$. Hence, in the first $k-1$ points $\gamma_j = \gamma_{j-1}(1+P)$ we have $r(\gamma_j) < r(\gamma_{j-1})$, i.e., the residual curve $r(\gamma)$ is decreasing with a slope steeper than α.

At the k-th step we may have the following situations:

- $r'(\gamma_k) \leq \alpha$ In this case, we have $\gamma_t \equiv \gamma_{k+1}$ and $|r'(\gamma_t)| \leq \alpha$.
- $r'(\gamma_k) > \alpha$ In this case, $r(\gamma_k) > r(\gamma_{k-1})$, and hence, we compute a new value γ_{k+1} in the interval (γ_{k-1}, γ_k) by means of the backtracking procedure (Algorithm 2).

Now we describe how to compute a suitable starting value γ_0. The idea is to choose

$$\gamma_0 = \mathrm{MTV}\left(U^{(F)}\right)$$

where $U^{(F)}$ is obtained by solving :

$$\min_U \frac{1}{2}\left\|\mathcal{H}U - Y^\delta\right\|_2^2 + \lambda_F \mathrm{MTV}(U). \tag{15}$$

In order to choose a good value for λ_F, we deal with the spectral components of the blurring operator.

We first consider the case of within channel blur, i.e., the matrix $\mathcal{H}$ in (9) is block diagonal with $H_{i,j} \neq 0$ iff $i = j$. Since each blurring submatrix $H_{i,i}$ is a BCCB matrix, it can be diagonalized by a unitary Fourier matrix F

$$H_{i,i} = F^* \Psi^{(i)} F, \quad i = 1, \ldots, p$$

where $\Psi^{(i)} \equiv diag(\psi_k^{(i)}, k = 1, \ldots M \cdot N)$, and $\psi_k^{(i)}$ are the eigenvalues of $H_{i,i}$. We define suitable parameters β_i, $i = 1, \ldots, p$, that filter out the contribution of the small components of $\Psi^{(i)}$,

by undersampling the power spectrum and compute the filter parameters as

$$\beta_i = \min_{1 \leq k \leq N} \left| \frac{\left(F Y_i^\delta\right)_{k \cdot M}}{\psi_{k \cdot M}^{(i)}} \right|, \quad i = 1, \ldots, p. \tag{16}$$

Finally, λ_F is defined as

$$\lambda_F = \min_{1 \leq i \leq p} \beta_i.$$

In the case of cross-channel blur, where $H_{i,j} = F^* \Psi^{(i,j)} F$, $i, j = 1, \ldots p$, we choose to approximate $\mathcal{H}$ by a block diagonal matrix $\bar{\mathcal{H}}$ where each diagonal block is

$$\bar{H}_{i,i} = F^* \bar{\Psi}^{(i)} F, \quad \bar{\Psi}^{(i)} = \frac{1}{p} \sum_{j=1}^{p} \Psi^{(i,j)}, \quad i = 1, \ldots, p,$$

and $\bar{\Psi}^{(i)} \equiv diag(\bar{\psi}_k^{(i)}, k = 1, \ldots M \cdot N)$. We use $\bar{\psi}_k^{(i)}$ in place of $\psi_k^{(i)}$ in (16) and we proceed as before for the computation of $U^{(F)}$.

Finally, the initial value of the parameter γ is computed as follows:

$$\gamma_0 = \mathrm{MTV}\left(U^{(F)}\right). \tag{17}$$

3.3 Solution of the Constrained Minimization Problem

For each value γ_j of the sequence described in the previous section, we solve the constrained minimization problem (13) by solving its Lagrangian dual form:

$$\max_\lambda \min_U \mathcal{L}(U, \lambda),$$
$$\mathcal{L}(U, \lambda) \equiv \frac{1}{2}\left\|\mathcal{H}U - Y^\delta\right\|_2^2 + \lambda\left(\mathrm{MTV}(U) - \gamma_j\right). \tag{18}$$

In the rest of this section, we analyze the algorithm for the solution of a single problem of the form (18) with a fixed value $\gamma \equiv \gamma_j$.

Imposing the first order conditions $\nabla_\lambda \mathcal{L}(U,\lambda) = 0$, we compute the solution $(\hat{\lambda}, \hat{U})$ of (18) as

$$\begin{aligned} &\text{find } \hat{\lambda} \text{ s.t. } \mathrm{MTV}(\hat{U}(\lambda)) - \gamma = 0 \\ &\quad \text{where } \hat{U} \equiv U(\hat{\lambda}) \text{ is the solution of} \\ &\min_U \mathcal{L}\big(U, \hat{\lambda}\big). \end{aligned} \tag{19}$$

The properties of the nonlinear equation $\mathrm{MTV}(U(\lambda)) - \gamma = 0$, proved in theorem 3, allow us to compute a sequence $\{\lambda_k\}$ converging to the root $\hat{\lambda}$. By solving (19) with $\lambda = \lambda_k$, we obtain a sequence $U^{(k)} = U(\lambda_k)$ converging the solution $\hat{U}$. Hence, the solution $(\hat{\lambda}, \hat{U})$ of (18) is the limit of the sequence $(\lambda_k, U^{(k)})$.

3.3.1 Computation of the Sequence λ_k

The nonlinear equation $\mathrm{MTV}(U(\lambda)) - \gamma = 0$ is solved by an iterative method that, starting from a suitable initial value λ_0 s.t. $\mathrm{MTV}(U^{(0)}) - \gamma < 0$, computes an approximation of the root $\hat{\lambda}$ by few iterations of a hybrid bisection+secant method. Therefore the update of λ_k is obtained by the bisection rule in the first k_s steps, i.e., when $k \le k_s$:

$$\begin{aligned} &\lambda_k = \mathcal{F}_b\big(\lambda_{k-1}\big), \quad \mathcal{F}_b\big(\lambda_{k-1}\big) = \lambda_{k-1} \\ &\qquad + \mathrm{sign}\big(\mathrm{MTV}(U^{(k-1)}) - \gamma\big)\frac{\lambda_0}{2^k}, \qquad k = 1, 2 \ldots, k_s \end{aligned} \tag{20}$$

When $k > k_s$ the secant rule is used, giving:

$$\begin{aligned} &\lambda_k = \mathcal{F}_s\big(\lambda_{k-1}\big), \quad \mathcal{F}_s\big(\lambda_{k-1}\big) = \lambda_{k-1} \\ &\quad - \frac{\mathrm{MTV}\big(U^{(k-1)}\big) - \gamma}{\mathrm{MTV}\big(U^{(k-1)}\big) - \mathrm{MTV}\big(U^{(k-2)}\big)}\,(\lambda_{k-1} - \lambda_{k-2}), \\ &k = k_s + 1, \ldots \end{aligned} \tag{21}$$

Usually few bisection iterations are necessary ($k_s \simeq 3$) to guarantee the convergence of the secant iterations, and globally less than 10 iterations are performed by the hybrid method when stopped by the following criterion:

$$\begin{aligned} &\big|\mathrm{MTV}(U^{(k)}) - \gamma\big| < \tau_t \big|\mathrm{MTV}(U^{(0)}) - \gamma\big| + \tau_a \\ &\quad \text{or } \big|\lambda_k - \lambda_{k-1}\big| < \tau_a \quad \text{or} \quad k > \texttt{maxit}. \end{aligned} \tag{22}$$

with $\tau_t = \tau_a = 10^{-3}$ and $\texttt{maxit} = 15$ in our experiments.

3.3.2 Computation of the Sequence $\boldsymbol{U^{(k)}}$

For what concerns the solution of the unconstrained optimization problem (19), the choice of the multichannel TV definition has great influence on its efficiency. Using MTV, an efficient and fast solution is obtained by the fast alternating minimization algorithm proposed in [26]. However, different solvers could be applied to (19), related to different extensions of the TV to multichannel images.

Below we report the algorithm used for solving the constrained minimization problem (13).

Algorithm 3 (Computation of $U(\gamma_j)$: INPUT: $(\gamma_j, \mathcal{H}, Y^\delta)$, λ_0; OUTPUT: U)

compute $U^{(0)}$ *solving* (19) *with* $\lambda = \lambda_0$ *using FTVD4 function*
$k = 0$ *% Solution Computation*
`repeat`
  $k = k + 1$
  `if` $k < k_s$
    $\lambda_k = \mathcal{F}_b(\lambda_{k-1})$ *as in* (20)
  `else`
    $\lambda_k = \mathcal{F}_s(\lambda_{k-1})$ *as in* (21)
  `end`
  compute $U^{(k)}$ *solving* (19) *with* $\lambda = \lambda_k$ *using FTVD4 function*
`until` *exit condition* (22)
$U = U^{(k)}$.

3.4 Algorithm MCLSTV

Finally, we combine Algorithms 1–3 to derive our overall Algorithm MCLSTV. We remind that $r'(\gamma_k)$ is approximated by a first order forward finite difference formula:

$$r'\big(\gamma_k\big) \approx \frac{r\big(\gamma_k\big) - r\big(\gamma_{k-1}\big)}{\gamma_k - \gamma_{k-1}}.$$

Algorithm 4 (MCLSTV: (INPUT $(Y^\delta, \mathcal{H})$; OUTPUT: U)

$j = 0$;
Compute γ_0 *as in* (17)
Compute $U^{(0)}$ *solving* (13) *with* $\gamma = \gamma = 0$ *with Algorithm 3*
$r^{(0)} = \|Y^\delta - \mathcal{H}U^{(0)}\|$,
`repeat` *% as done in Algorithm 1*
  $j = j + 1$;
  $\gamma_j = \gamma_{j-1}(1 + P)$;
  Compute $U^{(j)}$ *solving* (13) *with* $\gamma = \gamma_j$ *with Algorithm 3*
  $r^{(j)} = \|Y^\delta - \mathcal{H}U^{(j)}\|$,
  $D^{(j)} = (r^{(j)} - r^{(j-1)})/(\gamma_j - \gamma_{j-1})$
`until` $D^{(j)} \ge -\alpha$
`if` $|D^{(j)}| < \alpha$;
  $\gamma_t = \gamma_j$; $U = U^{(j)}$
`else`
  $a = \gamma_{j-1}$, $b = \gamma_j$
  Compute $\gamma_t =$ `BCKTRK(a,b)` *using Algorithm 2*
  $U = U(\gamma_t)$
`end`

3.4.1 Comments on the Parameters

The parameter α represents the slope of the residual function at the desired solution γ_t. Recalling the behavior of the residual curve, we know that the residual curve decreases quite steeply when $\gamma_k \le \tilde{\gamma}$ and tends to become flat as $\gamma_k \simeq \hat{\gamma}$. If α is too small, we have $\gamma_t \gg \hat{\gamma}$, and hence a noisy solution.

On the other hand if α is too large we may find $\gamma_t \ll \hat{\gamma}$ giving an oversmoothed solution. Suitable values for α are 10^{-p} with $3 \leq p \leq 5$. In our tests, we set $\alpha = 10^{-4}$. It may appear that we changed the parameter γ with the parameter α with no gain for the algorithm. This is true only in part, since we observe that the new parameter α is an error tolerance, and hence is easier to set, compared to γ which can change greatly, depending on the different images (see column 3 in Tables 4 and 5).

The parameter P is the relative increase of the initial value γ_0; hence, the only requirement is that $P > 0$. However, if P is too large then many backtracking steps may be required. On the other hand, if P is too small, a large number of iterations may be necessary. In our experiments, we found a good compromise by taking $P = 1/4$.

The parameter s is used to define a new value in the interval (a, b) in the backtracking procedure, hence $s \in (0, 1)$. In our experiments, we set $s = 0.5$.

3.4.2 Computational Cost

Each step of MCLSTV requires to solve the constrained problem (13) for a given value γ_j, $j = 1, \ldots t$; hence, the computation workload is mainly given by the cost of **Algorithm 3** multiplied by $(t+nb)$, where t is given in **Algorithm 4** and nb is the total number of backtracking steps in BCKTRK.

4 Numerical Experiments

In this section, we report the results obtained by several tests aiming to show the efficiency and robustness of the MCLSTV algorithm.

The tests are carried out in Matlab R2010a.

We consider some deblurring problems of color images ($p = 3$) on the following test images:

I1 409×409 true color ray tracing image (Fig. 1a),
I2 768×512 true color photographic image (Fig. 1b) ,
I3 1597×1125 true color fractal image (Fig. 1c),
I4 314×513 Matlab true color test image (Fig. 1d) .

In the reported tests, all the images are blurred using two different types of cross-channel blurring kernels:

- $\mathcal{H}1$: One single channel blurring kernel with different weights on each channel (Paragraph 4.1);
- $\mathcal{H}2$: A combination of three different single channel blurring kernels with different weights on each channel Paragraph 4.2).

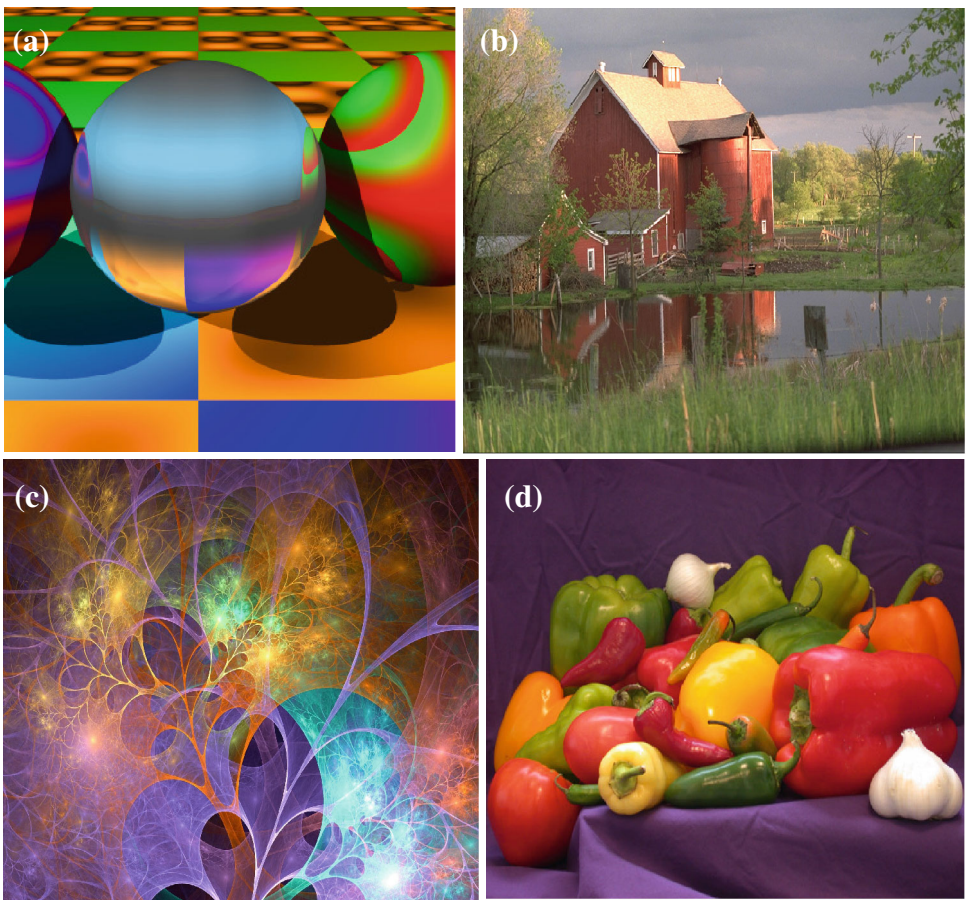

Fig. 1 True color test images. **a** I1: 409×409 ray tracing image, **b** I2: 768×512 photographic image, **c** I3: 1597×1125 fractal image, **d** I4: 314×513 test image

Finally we test the robustness of the method on a large set of color images formed by three image databases available on the web. The aim is to show that the MCLSTV algorithm produces good restoration of color images of different type.

The blurred images Y are obtained by applying the blurring kernel $\mathcal{H}$ to the test image V: $Y = \mathcal{H}V$ where $\mathcal{H} \in \mathbb{R}^{N\times M\times 3}$; the blurred noisy images Y^δ are obtained by adding Gaussian white noise to Y as in (11).

In our experiments, we use the following values of the noise level δ: $\delta_1 = 5\cdot 10^{-3}$, $\delta_2 = 10^{-2}$, $\delta_3 = 5\cdot 10^{-2}$.

In all the reported experiments, unless otherwise specified, we apply MCLSTV with the following parameters: $\alpha = 10^{-4}$, $P = 0.25$, $s = 0.5$. The inner unconstrained minimization problem (19) is solved by the alternating direction method, implemented in the Matlab function ADM2MTVL2, available from http://www.caam.rice.edu/~optimization/L2/ftvd/.

This choice is due to the proven efficiency of the alternating direction method in the case of cross-channel blur [26]. Of course it is always possible to replace it with other favorite black-box solvers.

The quality of the computed image U is evaluated by the Signal to Noise Ratio (SNR) and by the Structural SIMilarity Index (SSIM). The Signal to Noise Ratio is defined as follows:

$$\mathrm{SNR}(U) = 10\log_{10}\left(\frac{\|U-\bar{U}\|}{\|U-V\|}\right)$$

where V is the original image and $\bar{U}$ is a constant image with value equal to the mean intensity value of V.

The Structural SIMilarity Index (SSIM) is defined in [27] to compare two grayscale images. Its value, computed by the function ssim (available online at http://www.cns.nyu.edu/~lcv/ssim/), is in the interval [0, 1] where a value equal to one means that the two images are identical. In this paper, we extend its definition to a p-channel image $U = (U_1, \ldots, U_p)$ by computing

$$\mathrm{SSIM} = \frac{1}{p}\sum_{k=1}^{p} \mathtt{ssim}(U_k)$$

Moreover, the effectiveness of MCLSTV algorithm is evaluated by computing the percentage improvement of the parameters SNR and SSIM, i.e., SNRp and SSIMp:

$$\mathrm{SNRp} = \frac{\mathrm{SNR}(U)}{\mathrm{SNR}(\tilde{U})}\times 100, \qquad \mathrm{SSIMp} = \frac{\mathrm{SSIM}(U)}{\mathrm{SSIM}(\tilde{U})}\times 100 \tag{23}$$

where $\tilde{U} = U(\tilde{\gamma})$ has been defined in Sect. 3 and represents the best possible restoration from model (12) (the experimental value $\tilde{\gamma}$ has been computed through repeated simulations).

We note that SNRp and SSIMp are always in the interval [0, 100].

In the following, we define U_t as the image restored with MCLSTV method and γ_t as the final value of γ computed by the algorithm.

4.1 Tests with Kernel $\mathcal{H}1$

We present here the results obtained by the following cross-channel blurring matrix $\mathcal{H} = \mathcal{H}1$:

$$\mathcal{H}1 = \begin{pmatrix} 0.7G(21,11) & 0.15G(21,11) & 0.15G(21,11) \\ 0.1G(21,11) & 0.8G(21,11) & 0.1G(21,11) \\ 0.2G(21,11) & 0.2G(21,11) & 0.6G(21,11) \end{pmatrix}. \tag{24}$$

$G(21, 11)$ is a Gaussian function $G(h, \sigma)$ of size $h = 21$ pixels and variance $\sigma = 11$, computed using the Matlab Image Processing Toolbox fspecial function. The blurred noisy images are obtained by adding Gaussian white noise with $\delta_3 = 0.05$ and $\delta_1 = 0.005$. Figure 2b and d show the images I1 and I3 restored by MCLSTV algorithm in the case $\delta = \delta_3$.

In Table 1, we report the error measures obtained in the four test images considered. In bold fonts, we report the median values of the SNR and SSIM parameters of the computed restorations. We can observe that the proposed method produces very good restorations, in terms of error parameters.

The plots reported in Figs. 3a (for I1) and 4a (for I3) show the behavior of the residual norm of the iterates $U^{(j)}$ as a function of γ_j (in Algorithm 4). The values of γ_j computed by the MCLSTV algorithm are plotted with red circles, the final value γ_t with a red star, and the value $\tilde{\gamma}$ with a black square. In all the cases, the last value γ_t of the sequence is very close to the best possible value $\tilde{\gamma}$, showing that the approximation computed by our algorithm is very good.

Figures 3b (for I1) and 4b (for I3) show, respectively, the SNR and the SSIM (in log scale) of the iterates $U^{(j)}$ as a function of γ_j. Even in these cases, the SNR of the image restored by the MCLSTV method (red star) is very close to the SNR of the image restored by using the value $\tilde{\gamma}$(black square), that is the best possible SNR achievable by the constrained regularization model (13).

4.2 Tests with Kernel $\mathcal{H}2$

In the tests reported in this section, the cross-channel blur is introduced as a weighted combination of motion (M), gaussian (G), and average (A) kernels with different parameters [26] (they all are computed by the fspecial Matlab function).

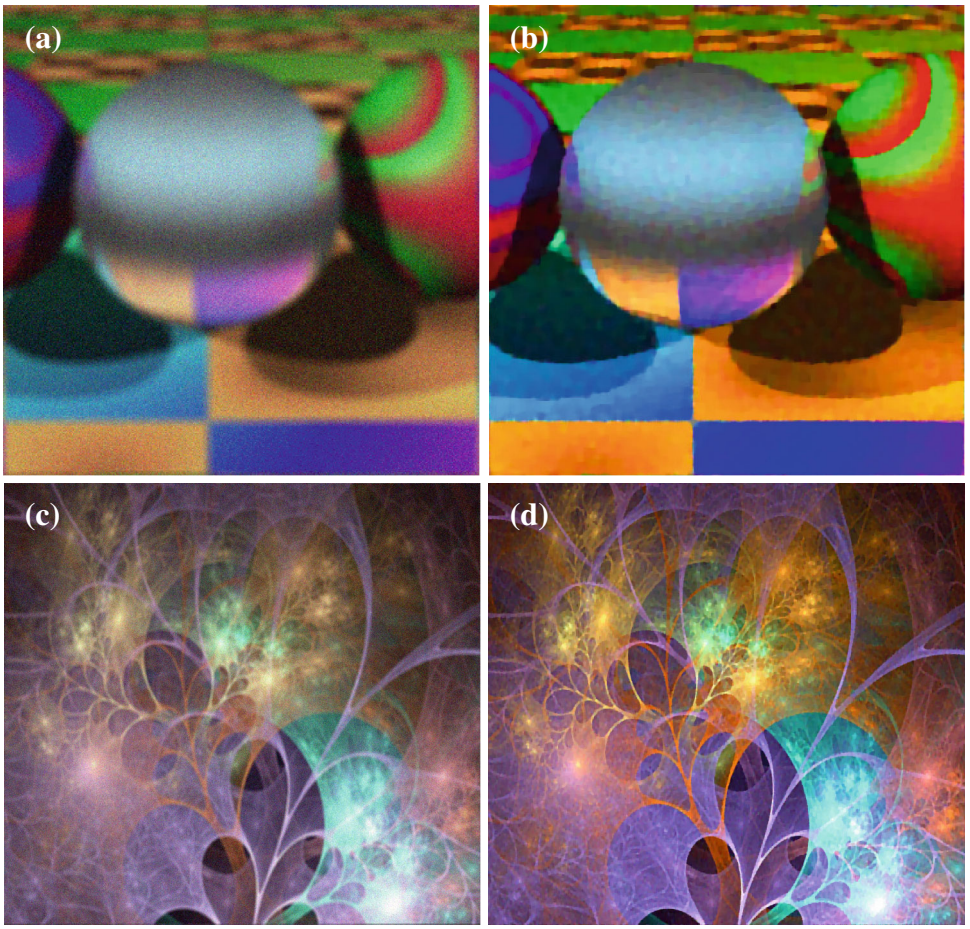

Fig. 2 Images blurred by kernel $\mathcal{H}1$ and restored images. **a** I1 blurred image with H1 and = 0.05, **b** I1 restored image with SNRp = 98.9 %, **c** I3 blurred image with H1 and = 0.05, **d** I3 restored image with SSIMp = 99.5 %

Table 1 Results for $\mathcal{H}1$ blur

Test	SNR (Y^δ)	SSIM (Y^δ)	SNR ($\tilde{U}$)	SSIM ($\tilde{U}$)	SNR (U_t)	SSIM (U_t)
I1	5.690	4.883×10^{-1}	16.339	8.542×10^{-1}	16.168	8.120×10^{-1}
I1	6.080	6.217×10^{-1}	23.196	9.716×10^{-1}	23.047	9.716×10^{-1}
I2	6.150	6.421×10^{-1}	10.577	7.114×10^{-1}	10.414	6.840×10^{-1}
I2	7.765	6.486×10^{-1}	13.993	8.718×10^{-1}	13.454	8.546×10^{-1}
I3	6.871	7.406×10^{-1}	12.890	8.748×10^{-1}	12.775	$\mathbf{8.656 \times 10^{-1}}$
I3	8.149	7.772×10^{-1}	15.618	9.662×10^{-1}	15.529	9.656×10^{-1}
I4	6.551	5.948×10^{-1}	16.697	8.771×10^{-1}	**15.475**	7.915×10^{-1}
I4	7.503	7.744×10^{-1}	20.905	9.505×10^{-1}	20.396	9.364×10^{-1}

For each test problem the first line is with higher noise δ_3, the second line with lower noise δ_1. In bold fonts are reported the results closer to the median values

In this case, the blurred matrix $\mathcal{H}2$ is:

$$\mathcal{H}2 = \begin{pmatrix} 0.7G(9,5) & 0.15G(11,5) & 0.15A(17) \\ 0.1A(13) & 0.8G(7,5) & 0.1M(21,45) \\ 0.2A(15) & 0.2M(41,90) & 0.6M(61,135) \end{pmatrix} \tag{25}$$

The blurred and restored images I2 and I4 are shown in Fig. 5. Table 2 reports the errors obtained for the test images blurred by (25) with noise levels $\delta_3 = 0.05$ and $\delta_1 = 0.005$. From the table, we see that in this case, the results are generally less accurate compared to those in Table 1 (computing the median SNRp and SSIMp, we have 98.7 and 96.9 %, respectively). The deblurring problem with this kind of kernel is more difficult. Comparing the second columns of Tables 1 and 2, we observe that the images blurred by $\mathcal{H}2$ have a lower SNR compared to those blurred by $\mathcal{H}1$. However, good quality results are still obtained.

Figures 6 and 7 plot the residual norms, the SNR and the SSIM values as a function of γ, as before. They confirm

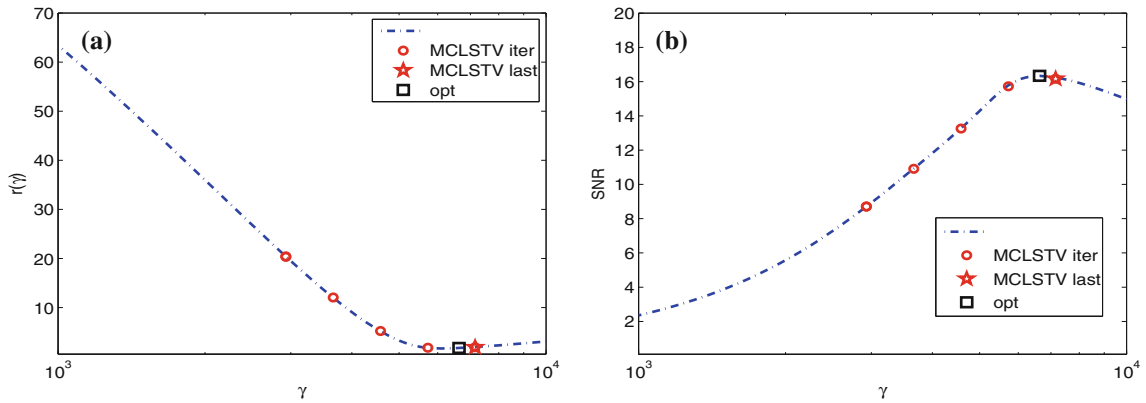

Fig. 3 Test image I1 (blurring kernel $\mathcal{H}1$, $\mathrm{SNR}(Y^\delta) = 5.69$). Plots of the residual norm $r(\gamma)$ and SNR as functions of γ

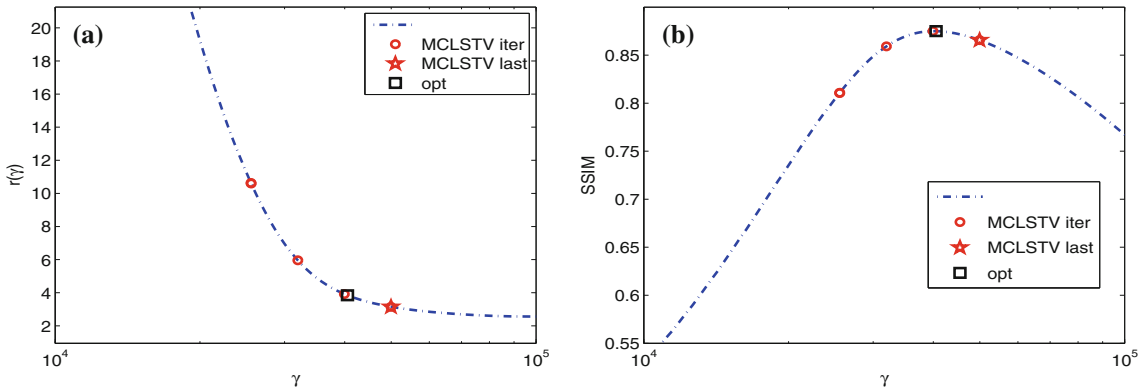

Fig. 4 Test image I3 (blurring kernel $\mathcal{H}1$, $\mathrm{SNR}(Y^\delta) = 6.871$). Plots of the residual norm $r(\gamma)$ and SSIM as functions of γ

that the value γ_t computed by the algorithm sequence is very close to $\tilde{\gamma}$, and the error parameters for the restored images are very good compared to the best we can obtain with the considered model.

4.3 MCLSTV Algorithm Robustness

We tested the behavior of our MCLSTV algorithm assuming that the kernel is not exactly known. Uniform random noise has been added to the weights of the kernel (24) in the reconstruction process, and several reconstructions have been performed with noise level (δ_{ker}) in the interval $[10^{-2}, 2.5\ 10^{-1}]$.

We report in Table 3, as an example, the tests relative to the I4 test image, with noise $\delta_3 = 0.05$ on the blureed image. The parameters of the blurred image are reported in Table 1, first line of I4. The relative error between the original and the corrupted kernel is reported in column $\mathrm{Err}_{\mathrm{Ker}}$, while SNR, SSIM, and PSNR of the reconstructions are reported in the third, fourth, and fifth columns, respectively. Using the exact kernel (24), we obtain SNR = 15.475, SSIM = 0.7915 and PSNR = 27.9197, showing a good robustness of the algorithm.

4.4 MCLSTV Algorithm Efficiency

In Tables 4 and 5, we report some of the parameters for the evaluation of the computational efficiency of the MCLSTV algorithm on test images blurred by $\mathcal{H}1$ and $\mathcal{H}2$, respectively.

In each table, the value γ_t is the approximation of $\tilde{\gamma}$ computed by MCLSTV, t is the number of elements of the sequence γ_j (outer iterations) and N_{AD} the total number of alternating directions iterations for solving problem (19) (inner iterations). We observe that a small number of adaptive steps is required, $5 < t < 15$. As an example of execution time, about 4 seconds are required to perform 718 inner iterations with the image I2 on an Intel i7 3.4 GHz, equipped with 16 Gb of RAM and Windows 7.

4.5 Tests on a Large Set of Images

In order to show the global effectiveness of the method, we run tests on a set of color images from the following images repositories:

S1 Color image set PCD0992 available from the web site http://r0k.us/graphics/kodak.

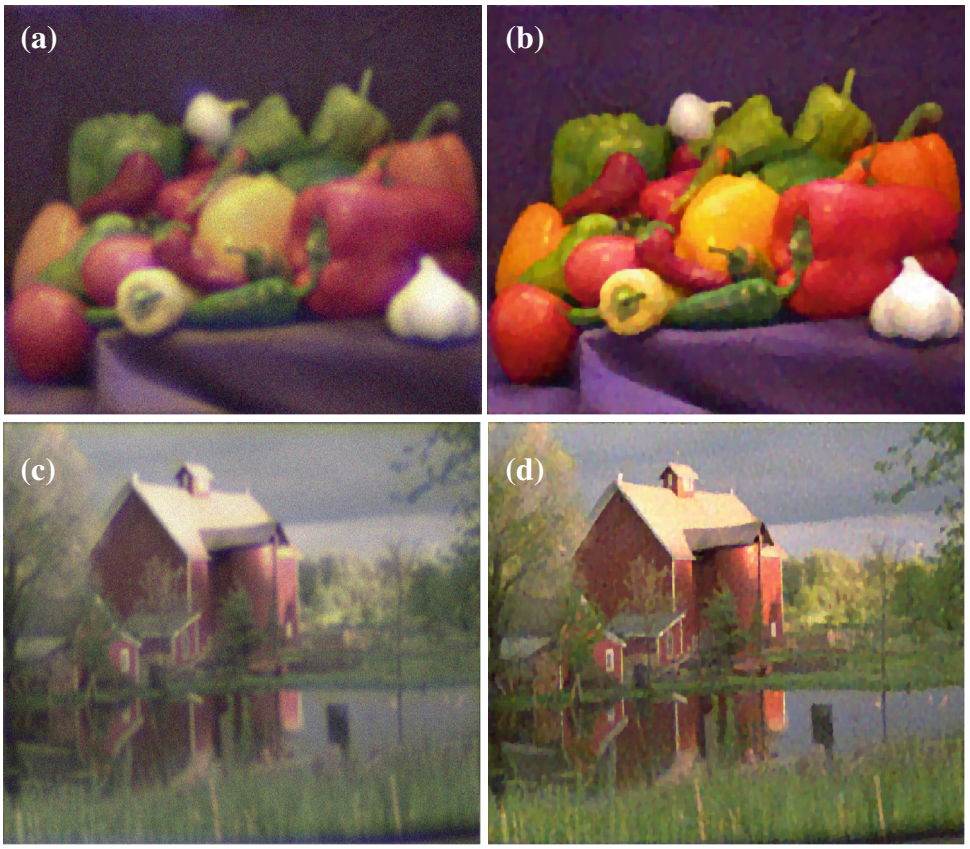

Fig. 5 Images blurred by kernel $\mathcal{H}2$ and restored images. **a** I4 blurred image with H2 and = 0.05, **b** I4 restored image with SNRp = 81.9 %, **c** I2 blurred image with H2 and = 0.05, **d** I2 restored image with SSIMp = 96.9 %

Table 2 Results for $\mathcal{H}2$ blur

Test	SNR (Y^δ)	SSIM (Y^δ)	SNR $(\tilde{U})$	SSIM $(\tilde{U})$	SNR (U_t)	SSIM (U_t)
I1	4.786	4.893×10^{-1}	16.360	8.550×10^{-1}	16.152	7.883×10^{-1}
I1	5.100	6.170×10^{-1}	23.590	9.530×10^{-1}	23.447	9.454×10^{-1}
I2	5.908	5.061×10^{-1}	10.283	7.064×10^{-1}	10.196	6.906×10^{-1}
I2	7.417	6.568×10^{-1}	13.881	8.705×10^{-1}	12.332	8.183×10^{-1}
I3	6.399	6.979×10^{-1}	12.401	8.453×10^{-1}	12.391	$\mathbf{8.455 \times 10^{-1}}$
I3	7.526	7.332×10^{-1}	15.848	9.603×10^{-1}	**14.182**	9.231×10^{-1}
I4	6.170	5.786×10^{-1}	15.657	8.470×10^{-1}	15.531	8.277×10^{-1}
I4	7.033	7.557×10^{-1}	20.864	9.449×10^{-1}	17.085	8.930×10^{-1}

For each test problem the first line is with higher noise δ_3, the second line with lower noise δ_1. In bold fonts are reported the results closer to the median values

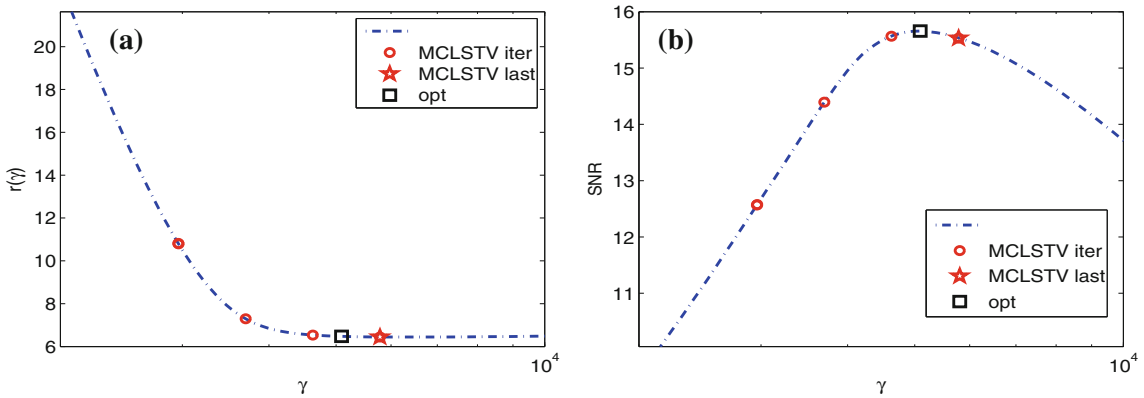

Fig. 6 Test image I4 (blurring kernel $\mathcal{H}2$, SNR$(Y^\delta) = 5.952$). Plots of the residual norm $r(\gamma)$ and SNR as functions of γ

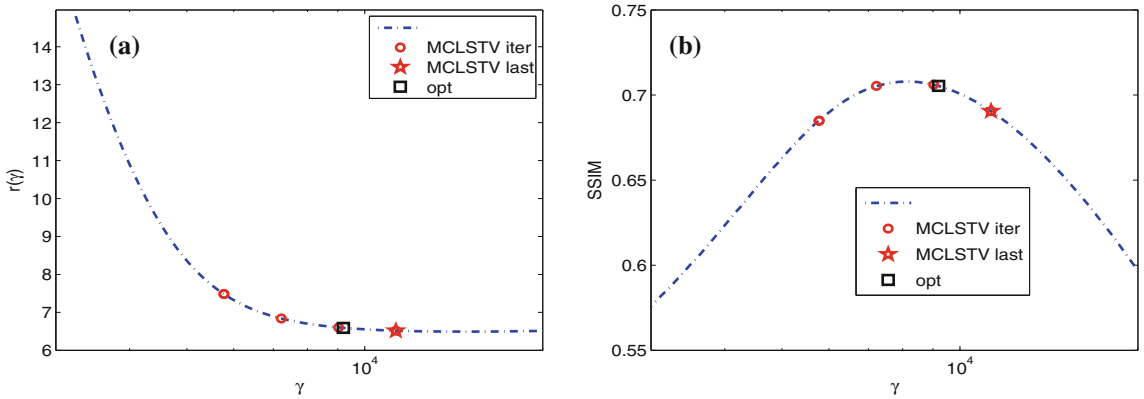

Fig. 7 Test image I2 (blurring kernel $\mathcal{H}2$, SNR$(Y^\delta) = 5.908$). Plots of the residual norm $r(\gamma)$ and SSIM as functions of γ

Table 3 Error values for the restoration of $I4$ image with uniform random noise on the kernel weights

δ_{ker}	Err_{Ker}	SNR	SSIM	PSNR
0.25	1.3687	3.5674	0.6344	16.0121
0.10	0.5475	11.3564	0.7630	23.8010
0.05	0.2737	14.1408	0.7813	26.5855
0.01	0.0547	15.4121	0.7915	27.8567

Table 4 Performance for $\mathcal{H}1$ blur. For each test problem, the first line is with higher noise δ_3, and the second line with lower noise δ_1

Test	SNR (Y^δ)	$\tilde{\gamma}$	γ_t	t	N_{AD}
I1	5.690	6.697×10^3	7.157×10^3	6	1050
I1	6.080	6.715×10^3	7.162×10^3	15	3172
I2	6.150	8.837×10^3	1.157×10^4	6	718
I2	7.765	1.560×10^4	1.138×10^4	6	723
I3	6.871	4.147×10^4	4.994×10^4	5	552
I3	8.149	6.629×10^4	5.872×10^4	6	659
I4	6.551	4.790×10^3	7.013×10^3	5	722
I4	7.503	5.806×10^3	6.963×10^3	5	705

Table 5 Performance for $\mathcal{H}2$ blur. For each test problem, the first line is with higher noise δ_3, and the second line with lower noise δ_1

Test	SNR (Y^δ)	$\tilde{\gamma}$	γ_t	t	N_{AD}
I1	4.786	6.918×10^3	7.599×10^3	4	679
I1	5.100	7.637×10^3	7.575×10^3	4	695
I2	5.908	8.945×10^3	1.129×10^4	5	522
I2	7.417	1.771×10^4	9.610×10^3	6	718
I3	6.399	4.686×10^4	4.334×10^4	7	835
I3	7.526	7.396×10^4	4.368×10^4	5	540
I4	6.170	5.090×10^3	5.785×10^3	5	695
I4	7.033	6.139×10^3	4.383×10^3	5	647

Table 6 Results for image databases

Set	NT	S1	meS1	S2	meS2
S1	144	113	97.04	129	97.89
S2	90	72	98.05	83	98.33
S3	84	69	96.86	82	99.62

NT Number of tests, *S1* number of images with SNRp > 90, *meS1* median value of SNRp, *S2* number of images with SSIMp > 90, *meS2* median value of SSIMp

S2 Color images from raytracing algorithms available at

http://tog.acm.org/resources/SPD/overview.html and http://www.f-lohmueller.de/pov/g_3c.htm.

S3 Set of fractal color images downloaded from http://www.f-lohmueller.de/fractal/gfr_00.htm `fractals_1` and `fractals_5`.

The experiments are performed using the blurring functions $\mathcal{H}1$ and $\mathcal{H}2$; for each blur, noise is added with levels: δ_1, δ_2 and δ_3.

In Table 6, we report the results for each database. As global indicator of success, we count the number of tests where SNRp and SSIMp are greater than 90 % (columns 3, 5) and report the median values of SNRp and SSIMp in columns 4 and 6, respectively. Statistical information about the whole test can be observed in Fig. 8.

In Fig. 8a, we plot the `boxplot` of the data, each containing the median value (the line inside the box), the 0.25, and 0.75 quantiles (the basis of the box); the minimum and maximum of the data (the whiskers); and the outliers (the crosses). In a compact form, they show that the `ssim` values are very good, near the 100 %. In the histogram reported in Fig. 8b, we can see that the largest number of samples achieve the best performance with SNRp and SSIMp close to 100 %.

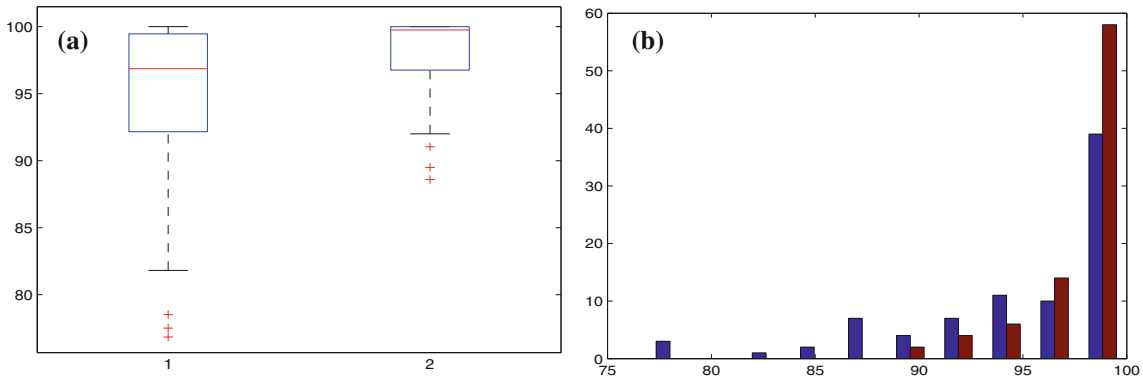

Fig. 8 Statistical information of $S1 \bigcup S2 \bigcup S3$ image sets. **a** Boxplot of SNRp(1) and SSIMp(2), **b** histogram of SNRp (*blue*) and SSIMp (*brown*) (Color figure online)

Table 7 Test problems blurred by kernel $\mathcal{H}_1$ and with added noise of level $\delta = \delta_1(0.05), \delta_3(0.005)$. Best SNRs obtained by RWM and MCLSTV methods

Test	δ	RWM	MCLSTV
I1	0.05	16.337	16.168
	0.005	23.111	23.047
I2	0.05	8.138	10.414
	0.005	13.007	13.454
I3	0.05	11.879	12.775
	0.005	15.603	15.529
I4	0.05	20.867	20.396
	0.005	15.622	15.475

4.6 Comparison with a Different Method

In this paragraph, we apply a method recently presented in the literature to estimate the regularization parameter by estimating the Residual Whiteness Measure (RWM) by Almeida and Figuereido [1]. This method can be extended to color images by converting to grayscale the residual obtained by solving (2) with different values of λ. Using the function `whiteness` from the authors' website, we compute the six measures of whiteness in [1], and we take the maximum.

In Table 7, we report the SNRs obtained on the test problems I1-I4 with $\mathcal{H}1$ blurring matrix and two levels of noise: $\delta_1 = 0.005$ and $\delta_3 = 0.05$, using the methods RWM and MCLSTV.

In Fig. 9, we plot the SNR as a function of λ. The best values obtained by RWM (black star) and MCLSTV (red circle) methods are reported for the best (Fig. 9a) by RWM method and the best (Fig. 9b) obtained by MCLSTV.

The restored images giving the best, median, and worse SNRs are reported in Figs. 10, 11, and 12 together with the reconstructions obtained by MCLSTV in the same cases. We observe that RWM achieves slightly better SNRs in the best and median cases, but seems to obtain worse results in critical cases where MCLSTV still shows better quality reconstruction.

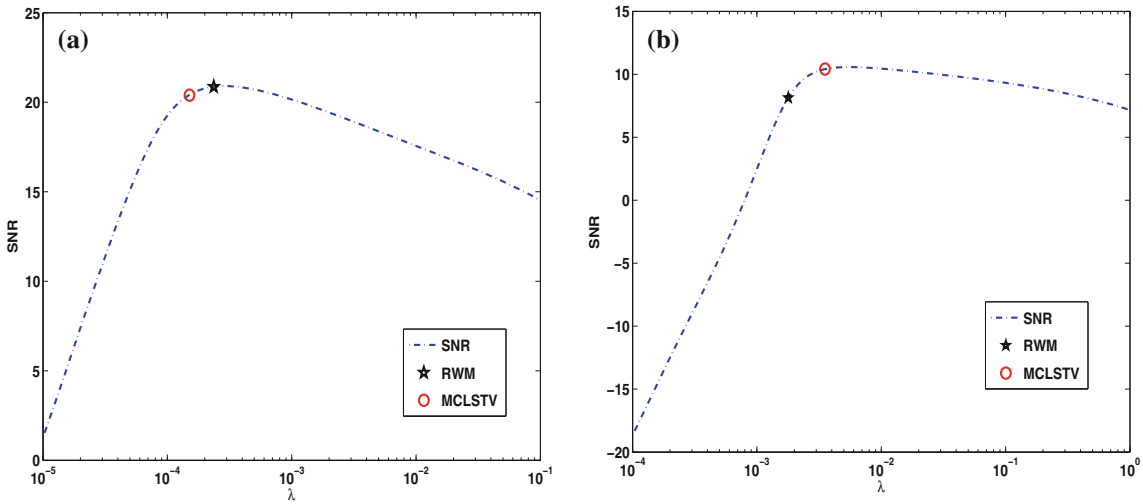

Fig. 9 SNR plots as a function of λ for the best and worse RWM reconstructions. **a** I4 = 0.005, **b** I2 = 0.05

Fig. 10 Best RWM reconstruction: Test I4 $\delta = 0.005$. **a** RWM SNR = 20.867, **b** CLSMTV SNR = 20.396

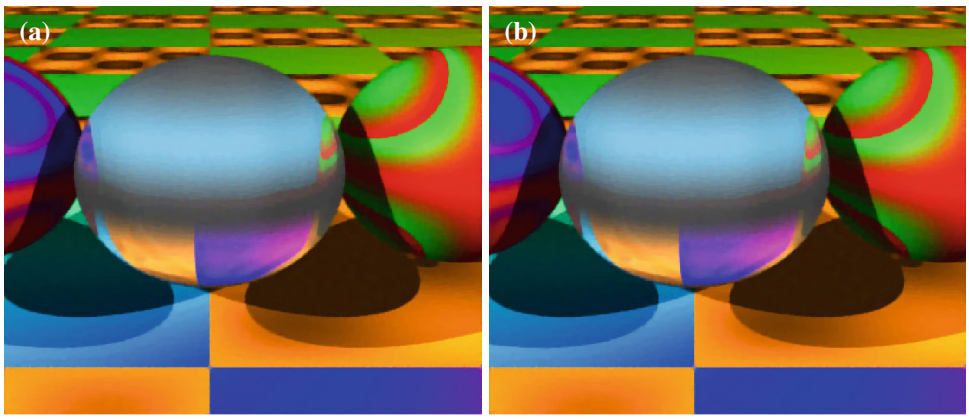

Fig. 11 Median RWM reconstruction: Test I1 $\delta = 0.005$. **a** RWM SNR = 23.111, **b** CLSMTV SNR = 23.047

Fig. 12 Worse RWM reconstruction: Test I2 $\delta = 0.05$. **a** RWM SNR = 8.138, **b** CLSMTV SNR = 10.414

5 Conclusions

In this paper, we proposed an iterative method for automatic restoration of blurred noisy vectorial digital images. The method only requires the blurred noisy image and the blurring kernel as input. The algorithm solves a finite sequence of constrained minimization problems with the objective function being the l2-norm of the residual and the constraint function being a vectorial extension of Total Variation. The experiments on large datasets of color images proved that the method efficiently recovers nearly optimal results in the great majority of the cases.

The present approach is general and can be extended to different types of objective functions and constraints on the basis of the specific application considered. The extension to blind image-restoration problems will be considered in future works.

References

1. Almeida, M.S.C., Figueiredo, M.A.T.: Parameter estimation for blind and non-blind deblurring using residual whiteness measures. IEEE Trans. Image Process. **22**(7), 2751–2763 (2013)
2. Aujol, J., Gilboa, G.: Constrained and SNR-based solutions for TV-Hilbert space image denoising. J. Math. Imaging Vis. **26**, 217–237 (2006)
3. Babacan, S., Molina, R., Katsaggelos, A.: Parameter estimation in TV image restoration using variational distribution approximation. IEEE Trans. Image Process. **17**(3), 326–339 (2008)
4. Blomgren, P., Chan, T.F.: Color TV: total variation methods for restoration of vector-valued images. IEEE Trans. Image Process. **7**(3), 304–309 (1998)
5. Blomgren, P., Chan, T.F.: Modular solvers for image restoration problems using the discrepancy principle. Numer. Linear. Algebra Appl. **9**, 347–358 (2002)
6. Bresson, X., Chan, T.F.: Fast dual minimization of the vectorial total variation norm and applications to color image processing. Inverse Probl. Imaging **2**(4), 455–484 (2008)
7. Brito-Loeza, C., Chen, K.: On high-order denoising models and fast algorithms for vector-valued images. IEEE Trans. Image Process. **19**(6), 1518–1527 (2010)
8. Chen, K., Loli, E.: Piccolomini, and F. Zama. An automatic regularization parameter selection algorithm in the total variation model for image deblurring. Numer. Algorithms **92**, 67–73 (2014)
9. Dong, Y., Hintermuller, M., Rincon-Camacho, M.M.: Automated regularization parameter selection in multi-scale total variation models for image restoration. J. Math. Imaging Vis. **40**(1), 83–104 (2011)
10. Dong, Y., Hintermuller, M., Rincon-Camacho, M.M.: A multi-scale vectorial L^{τ}-TV framework for color image restoration. Int. J. Comput. Vis. **92**, 296–307 (2011)
11. Eldar, Y.: Generalized SURE for exponential families. IEEE Trans. Image Process. **21**(8), 3659–3672 (2012)
12. Fornasier, M., March, R.: Restoration of color images by vector valued BV functions and variational calculus. SIAM J. Appl. Math. **68**(2), 437–460 (2007)
13. Galatsanos, N., Katsaggelos, A., Chin, R., Hillery, A.: Least Squares restoration of multichannel images. IEEE Trans. Signal Process. **39**(10), 2222–2236 (1991)
14. Hansen, P.C., Nagy, J.G., O'Leary, D.P.: Deblurring Images: Matrices, Spectra, And Filtering. SIAM Publications, Philadelphia (2006)
15. Hansen, P.C., O'Leary, D.P.: The use of L-curve in the regularization of of discrete ill-posed problems. SIAM J. Sci. Comput. **14**, 1487–1503 (1993)
16. Hansen, P.C., Kilmer, M.E., Kjeldsen, R.H.: Exploiting residual information in the parameter choice for discrete ill-posed problems. BIT **46**, 41–59 (2006)
17. Liao, H., Li, F., Ng, M.: Selection of regularization parameter in total variation image restoration. J. Opt. Soc. Am. A **26**(11), 2311–2320 (2009)
18. Lukas, M.A.: Robust generalized cross validation for choosing the regularization parameter. Inverse Probl. **22**, 1883–1902 (2006)
19. Morozov, V.A.: Methods for Solving Incorrectly Posed Problems. Springer, New York (1984)
20. Ramani, S., Liu, Z., Rosen, J., Nielsen, J., Fessler, J.: Regularization parameter selection for nonlinear iterative restoration and MRI reconstruction using GCV and SURE-based methods. IEEE Trans. Image Process. **21**(8), 3659–3672 (2012)
21. Stein, C.: Estimation of the mean of a multivariate normal distribution. Ann. Stat. **9**(6), 1135–1151 (1981)
22. Strong, D., Blomgren, P., Chan, T.F.: Spatially adaptive local feature driven total variation minimizing image restoration. CAM Report 97-32, UCLA Math Department (1997)
23. Strong, D., Blomgren, P., Chan, T.F.: Scale recognition regularization parameter selection and Meyer's G norm in Total Variation regularization. CAM Report 05-02, UCLA Math Department (2005)
24. Vogel, C.R., Oman, M.E.: Fast, robust total variation-based reconstruction of noisy, blurred images. IEEE Trans. Image Process. **7**, 813–824 (1998)
25. Yang, J., Zhang, Y., Yin, W.T.: A fast alternating direction method for $tvl1 - l2$ signal reconstruction from partial Fourier data. Selected topics in IEEE. J. Signal Process. **4**(2), 288–297 (2010)
26. Yang, J., Yin, W., Zhang, Y., Wang, Y.: A fast algorithm for edge-preserving variational multichannel image restoration. SIAM J. Imaging Sci. **2**(2), 569–592 (2009)
27. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Perceptual image quality assessment: from error visibility to structural similarity. IEEE Trans. Image Process. **13**(4), 600–612 (2004)
28. Wen, Y., Chan, R.: Parameter selection for total variation based image restoration using discrepancy principle. IEEE Trans. Image Process. **21**(4), 1770–1781 (2012)
29. 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)
30. Zhu, X., Milanfar, P.: Automatic parameter selection for denoising algorithms using a no-reference measure of image content. IEEE Trans. Image Process. **19**(12), 3116–3132 (2010)

K. Chen received his B.Sc., M.Sc., and Ph.D. degrees in Mathematics, respectively, from the Dalian University of Technology, China; the University of Manchester, U.K.; and the University of Plymouth, U.K. He is a computational mathematician specializing in developing nove fast algorithms and has authored over 100 journal publications in Scientific Computing. He is a Professor of Mathematical Sciences of the University of Liverpool, Liverpool, U.K., where he is the head of Applied Mathematics and directs a Multidisciplinary Research Centre for Mathematical Imaging Techniques (CMIT). His current interests include imaging models, theories, and algorithms for image restoration, image segmentation, and registration.

E. Loli Piccolomini received her B.S. degree in 1988. She is working as a Professor of Numerical Analysis at the University of Bologna since 2005. Prior to this period, she was researcher at the same University. Her research is on regularization methods for the numerical solution of inverse problems in imagiing applications, such as restoration or tomographic reconstruction.

F. Zama received the B.S. degree in Mathematics from the University of Bologna in 1987. She continued two years' fellowship on "Analysis and development of parallel algorithms for data and image processing" at the CINECA Consortium of Universities. She subsequently became the Research Assistant (1991) and then Associate Professor (2006), at the Department of Mathematics of the University of Bologna. Her research interests include inverse problems in image processing, regularization methods for ill-posed problems, and parameter identification of nonlinear chemical reaction models.