IPOL IMAGE PROCESSING ON LINE

Published in Image Processing On Line on 2013–12–17.
Submitted on 2013–01–17, accepted on 2013–10–10.
ISSN 2105–1232 © 2013 IPOL & the authors CC–BY–NC–SA
This article is available online with supplementary materials,
software, datasets and online demo at
https://doi.org/10.5201/ipol.2013.61

Chambolle's Projection Algorithm for Total Variation Denoising

Joan Duran[1], Bartomeu Coll[2], Catalina Sbert[3]

[1] Universitat de les Illes Balears, Spain (joan.duran@uib.es)
[2] Universitat de les Illes Balears, Spain (tomeu.coll@uib.es)
[3] Universitat de les Illes Balears, Spain (catalina.sbert@uib.es)

Communicated by Antonin Chambolle *Demo edited by* José-Luis Lisani

Abstract

Denoising is the problem of removing the inherent noise from an image. The standard noise model is additive white Gaussian noise, where the observed image f is related to the underlying true image u by the degradation model $f = u + \eta$, and η is supposed to be at each pixel independently and identically distributed as a zero-mean Gaussian random variable. Since this is an ill-posed problem, Rudin, Osher and Fatemi introduced the total variation as a regularizing term. It has proved to be quite efficient for regularizing images without smoothing the boundaries of the objects.
This paper focuses on the simple description of the theory and on the implementation of Chambolle's projection algorithm for minimizing the total variation of a grayscale image. Furthermore, we adapt the algorithm to the vectorial total variation for color images. The implementation is described in detail and its parameters are analyzed and varied to come up with a reliable implementation.

Source Code

ANSI C source code to produce the same results as the demo is accessible at the IPOL web page of this article[1].

Keywords: denoising, total variation, image restoration

[1] https://doi.org/10.5201/ipol.2013.61

1 Introduction

An important problem in image analysis is the reconstruction of an original image from an observation to extract meaningful information. Such a process, called image restoration, has been studied at length from different points of view [1, 4, 5].

The transformation or degradation to which an image is subjected during formation, transmission and recording processes is in general the result of two phenomena. The first one is deterministic and is related to the mode of image acquisition. The second is a random phenomenon and corresponds to the inherent noise coming from the quantum nature of light emission. *Denoising* is the problem that only takes into account the random phenomenon and it consists of removing noise from an image. The most commonly studied noise model is additive white Gaussian noise, where the observed noisy image f is related to the underlying true image u by the degradation model

$$f = u + \eta, \tag{1}$$

and η is supposed to be at each point in space independently and identically distributed as a zero-mean Gaussian random variable.

Unfortunately, this is an ill-posed inverse problem in the sense of Hadamard [6]. The information provided by f and the model (1) are not sufficient to ensure the existence, uniqueness and stability of a solution u. It is therefore necessary to regularize the problem by adding a constraint on the solution. Observing that quadratic regularization functionals did not allow recovering sharp discontinuities, Rudin, Osher and Fatemi proposed [1] the total variation as regularization. The denoised image u is the solution of the minimization problem

$$\min_{u \in BV(\Omega)} \int_\Omega |\nabla u(x)| dx, \tag{2}$$

subject to the constraints

$$\int_\Omega u(x)dx = \int_\Omega f(x)dx \quad \text{and} \quad \int_\Omega |u(x) - f(x)|^2 dx = \sigma^2 |\Omega|. \tag{3}$$

The first constraint assumes that the noise η has zero mean, whereas the second one uses the a priori information on the noise standard deviation σ. In a posterior paper, Chambolle and Lions [7] proved that (2)-(3) is naturally linked to the following unconstrained minimization problem:

$$\min_{u \in BV(\Omega)} \int_\Omega |\nabla u(x)| dx + \frac{\lambda}{2} \|u - f\|_2^2, \tag{4}$$

for an adequate Lagrange multiplier $\lambda > 0$. This problem is referred to as the *Rudin–Osher–Fatemi (ROF) model*. The first term in (4) is a smoothing term and the second one measures the fidelity to the data. Hence, the parameter λ is a positive weighting constant that controls the trade-off between a good fit to f and an irregular solution.

In view of (4), denoising is performed as an infinite-dimensional minimization problem, where the search space is the set of all images with bounded variation. Let us develop their justification of this choice. We are interested in obtaining a denoised image u in such a way that edges and other singularities are preserved. Classical Sobolev spaces do not authorize such singularities. When u is discontinuous, its gradient has to be understood as a measure, and the space of functions of bounded variation [8] is well adapted for this purpose. Therefore, BV-functions appear as a natural model for images, characterized by the appearance of discontinuous hypersurfaces. Let $\Omega \subset \mathbb{R}^N$ be a bounded domain, $N \geq 2$, then the *Total Variation* (TV) of a function $u \in L^1(\Omega, \mathbb{R})$ is defined by

$$\mathrm{TV}(u) := \sup \left\{ \int_\Omega u \, \mathrm{div}\xi \, dx \ : \ \xi \in C_c^1(\Omega, \mathbb{R}^N), \ \|\xi\|_\infty \leq 1 \right\}. \tag{5}$$

It is well known [8] that $\mathrm{TV}(u)$ is finite if and only if its distributional derivative Du is a finite Radon measure, in which case we have $\mathrm{TV}(u) = |Du|(\Omega)$. If u has a gradient $\nabla u \in L^1(\Omega, \mathbb{R}^2)$, then $\mathrm{TV}(u) = \int_\Omega |\nabla u|$. Without risk of ambiguity, we will adopt this notation even when u is not smooth enough to justify this integral notation. With this notation, the *space of functions of bounded variation* is defined as

$$BV(\Omega) := \left\{ u \in L^1(\Omega) \ : \ \int_\Omega |\nabla u| < \infty \right\}.$$

The TV-term introduced in (4) discourages the solution from having oscillations, yet it does allow the solution to have discontinuities. If $f \in L^2(\Omega, \mathbb{R})$, the minimizer of (4) exists, and it is unique and stable in $L^2(\Omega, \mathbb{R})$ with respect to perturbations in f [9].

Many algorithms for TV-denoising have been developed, especially for the Gaussian noise model. To name a few, there are algorithms based on duality [10, 11], Newton-based methods [12], graph cuts [13], frame shrinkage [14] and operator splitting methods [15, 16, 17, 18], particularly the split Bregman algorithm discussed in the paper by Getreuer [19]. In this paper we discuss the projection algorithm proposed by Chambolle [2] for minimizing the total variation of an image and we adapt it to the vectorial total variation for color images.

2 Chambolle's Projection Algorithm

Chambolle's projection algorithm [2] for TV-denoising is based on a dual formulation and it is related to the works of Chan, Golub and Mulet [20] or of Carter [21]. Chambolle discusses not only its application to the image denoising problem (4) for additive white Gaussian noise, but also to the inverse problem of zooming and to the computation of the mean curvature motion of interfaces. It has also been extended to image deblurring [22] or image inpainting and colorization [23]. For denoising, the algorithm provided in Chambolle's paper [2] has proved to be quite efficient for regularizing noisy images without over-smoothing the boundaries of the objects.

2.1 Minimization of the ROF Model using a Projection Approach

Let us call $F(u) = \mathrm{TV}(u)$, then the Euler–Lagrange equation associated to (4) is

$$\partial F(u) + \lambda(u - f) \ni 0, \tag{6}$$

where ∂F is called the *sub-differential* of F and is defined by

$$\omega \in \partial F(u) \ \Leftrightarrow \ F(v) \geq F(u) + \langle \omega, v - u \rangle_{L^2(\Omega,\mathbb{R})} \ \forall v \in L^2(\Omega, \mathbb{R}).$$

Note that if F is differentiable at u and $F(u) < \infty$, then $\partial F(u) = \{\nabla F(u)\}$. Now, according to convex analysis [24, 25], (6) can be rewritten as

$$u \in \partial F^* \left(\lambda(f - u) \right),$$

where F^* is the Legendre–Fenchel transform of F defined by

$$F^*(v) = \sup_u \left\{ \ \langle u, v \rangle_{L^2(\Omega,\mathbb{R})} - F(u) \right\}.$$

Equivalently,

$$\lambda f \in \lambda(f - u) + \lambda \partial F^* \left(\lambda(f - u) \right),$$

which implies that $\omega = \lambda(f - u)$ is the minimizer of

$$\frac{1}{2}\|\omega - \lambda f\|_2^2 + \lambda F^*(\omega). \tag{7}$$

It is a standard fact in convex analysis that the Legendre-Fenchel transform of a convex and one-homogeneous[2] functional F is the characteristic function of a closed convex set K_F:

$$F^*(v) = \mathcal{X}_{K_F}(v) = \begin{cases} 0 & \text{if } v \in K_F, \\ +\infty & \text{otherwise,} \end{cases} \tag{8}$$

where K_F is defined by

$$K_F = \left\{ v \in L^2(\Omega, \mathbb{R}) \ : \ \langle v, u \rangle_{L^2(\Omega,\mathbb{R})} \leq F(u) \ \ \forall u \in L^2(\Omega, \mathbb{R}) \right\}.$$

On the other hand, since any closed convex proper function[3] satisfies $F^{**} = F$, we recover

$$F(u) = \mathcal{X}^*_{K_F}(u) = \sup_v \left\{ \langle u, v \rangle_{L^2(\Omega,\mathbb{R})} - \mathcal{X}(v) \right\} = \sup_{v \in K_F} \langle u, v \rangle_{L^2(\Omega,\mathbb{R})}. \tag{9}$$

Therefore, one readily sees from (5) that K_F is the closure of the set

$$\left\{ \text{div}\xi \ : \ \xi \in C_c^1\left(\Omega, \mathbb{R}^2\right), \|\xi\|_\infty \leq 1 \right\}.$$

Finally, since ω is the minimizer of (7) and F^* is given by (8), then ω is necessarily the orthogonal projection of λf on K_F. If π_{K_F} denotes the projection operator onto the set K_F, then the solution of (4) is simply given by

$$u = f - \pi_{\frac{1}{\lambda} K_F}(f). \tag{10}$$

2.2 Computing the Discrete Minimizer

To simplify, we assume that the image is given by a two-dimensional matrix of size $N \times N$, although it would be easy to adapt all arguments for a more general $M \times N$ matrix. By Chambolle's notation, $X = \mathbb{R}^{N^2}$ and $Y = X \times X$. For any $u \in X$, we introduce the discrete gradient operator $Du \in Y$ via forward differences, i.e.

$$Du(i,j) = \left(\partial_x u(i,j), \partial_y u(i,j)\right), \quad \forall\, 1 \leq i, j \leq N, \tag{11}$$

with

$$\partial_x u(i,j) = \begin{cases} u(i+1,j) - u(i,j) & \text{if } i < N, \\ 0 & \text{if } i = N, \end{cases} \tag{12}$$

and

$$\partial_y u(i,j) = \begin{cases} u(i,j+1) - u(i,j) & \text{if } j < N, \\ 0 & \text{if } j = N. \end{cases} \tag{13}$$

Then, the discretization of the standard total variation defined in (5) is

$$F(u) := \sum_{1 \leq i,j \leq N} |Du(i,j)|_Y, \tag{14}$$

[2] $F(\lambda u) = \lambda F(u), \ \forall \lambda > 0, \ \forall u \in L^2(\Omega, \mathbb{R})$.

[3] A proper convex function ($F \neq \infty$) is closed if and only if it is lower semi-continuous.

where the standard Euclidean scalar product on Y reads

$$\langle p, q\rangle_Y = \sum_{1\le i,j\le N} p_1(i,j)q_1(i,j) + p_2(i,j)q_2(i,j),$$

for each $p, q \in Y$, $p = (p_1, p_2)$ and $q = (q_1, q_2)$. Then, we can rewrite (14) as

$$F(u) = \sup\left\{\langle p, Du\rangle_Y \ : \ p \in Y,\ |p(i,j)| \le 1\ \forall i,j\right\} \tag{15}$$

for every $u \in X$. We introduce a discrete divergence operator, $\operatorname{div} : Y \to X$, such that

$$\langle -\operatorname{div} p, u\rangle_X = \langle p, Du\rangle_Y, \quad \forall p \in Y,\ \forall u \in X,$$

where $\langle u, v\rangle_X = \sum_{i,j} u(i,j)v(i,j)$. It is easy to check that, for every $p = (p_1, p_2) \in Y$, div is given by

$$(\operatorname{div} p)(i,j) = \begin{cases} p_1(i,j) & \text{if } i = 1, \\ -p_1(i-1,j) & \text{if } i = N, \\ p_1(i,j) - p_1(i-1,j) & \text{otherwise,} \end{cases} + \begin{cases} p_2(i,j) & \text{if } j = 1, \\ -p_2(i,j-1) & \text{if } j = N, \\ p_2(i,j) - p_2(i,j-1) & \text{otherwise.} \end{cases} \tag{16}$$

From (15) and the definition of the divergence operator, we deduce that

$$F(u) = \sup_{v \in K_F} \langle u, v\rangle_X,$$

with K_F given by

$$\left\{\operatorname{div} p \ : \ p \in Y,\ |p(i,j)| \le 1,\ \forall i,j\right\}.$$

Computing the nonlinear projection $\pi_{\frac{1}{\lambda}K_F}(f)$ amounts to solving the following constrained minimization problem

$$\min\left\{\left\|\frac{1}{\lambda}\operatorname{div} p - f\right\|_X^2 \ : \ p \in Y,\ |p(i,j)|^2 - 1 \le 0,\ \forall i,j\right\}.$$

The classical Karush–Kuhn–Tucker optimality conditions [25, 26] imply that

$$-D\left(\frac{1}{\lambda}\operatorname{div} p - f\right)(i,j) + \alpha(i,j)p(i,j) = 0,$$

where $\alpha(i,j) \ge 0$ is the Lagrange multiplier associated with the constraint $|p(i,j)|^2 - 1 \le 0$. This new parameter can be eliminated as follows. If $|p(i,j)| < 1$, then the Lagrange multiplier is not active, i.e. $\alpha(i,j) = 0$ and $D\left(\frac{1}{\lambda}\operatorname{div} p - f\right)(i,j) = 0$. On the other hand, if $|p(i,j)| = 1$, then the Lagrange multiplier becomes active and $\alpha(i,j) = \left|D\left(\frac{1}{\lambda}\operatorname{div} p - f\right)(i,j)\right| > 0$. In any case, we deduce that $\alpha(i,j) = \left|D\left(\frac{1}{\lambda}\operatorname{div} p - f\right)(i,j)\right|$ for each $1 \le i,j \le N$. Finally, we get:

$$-D\left(\frac{1}{\lambda}\operatorname{div} p - f\right)(i,j) + \left|D\left(\frac{1}{\lambda}\operatorname{div} p - f\right)(i,j)\right| p(i,j) = 0.$$

Chambolle thus proposed the semi-implicit gradient descent scheme described in algorithm 1 for computing the discrete minimizer of (4). The criterion for stopping the iteration just consists in checking that the maximum variation between the dual variables $p_{i,j}^n$ and $p_{i,j}^{n+1}$ is less than some given tolerance. Furthermore, Chambolle proved [2, Theorem 3.1] the convergence of the algorithm for $\delta_t < 1/8$. However, Aujol showed [27] that Chambolle's algorithm is a particular case of a previous one proposed by Bermúdez and Moreno [28] that guarantees convergence using a twice larger time-step parameter. Accordingly, the convergence of algorithm 1 is guaranteed for any $\delta_t < 1/4$. Finally, we want to remark that the dual problem with a fixed λ can also be solved using a Nesterov accelerated method in an usually faster way, see for instance the paper by Beck et al. [17].

Algorithm 1 Chambolle's projection algorithm for gray-scale TV-denoising

Input: A noisy image $f(i,j)$ as a $N \times N$ matrix, a trade-off parameter $\lambda > 0$, a time-step parameter $\delta_t > 0$ and an algorithm tolerance $tol > 0$.
Output: The denoised image $u(i,j)$ as a $N \times N$ matrix.

$p \leftarrow 0$
while $\max_{1 \leq i,j \leq N} \left\{ |p^{n+1}(i,j) - p^n(i,j)| \right\} > tol$ **do**
  for all pixel (i,j) in the image **do**
    $$p(i,j) \leftarrow \frac{p(i,j) + \delta_t D\left(\operatorname{div} p - \lambda f\right)(i,j)}{1 + \delta_t |D\left(\operatorname{div} p - \lambda f\right)(i,j)|}$$
  end for
end while
return $u = f - \frac{1}{\lambda} \operatorname{div} p$.

2.3 Chambolle's Projection Algorithm for Color Images

For color images, the standard dual definition of the vectorial total variation introduced by Blomgren and Chan [3] is used in place of the TV:

$$\mathrm{VTV}(\mathbf{u}) = \int_\Omega \left(\sum_{m=1}^{M} |\nabla u_m|^2 \right)^{\frac{1}{2}} dx, \tag{17}$$

where $\mathbf{u} = (u_1, \ldots, u_M)$ and M is the number of channels. From the Euler-Lagrange equation of (17), one can observe that there is a coupling between components in such a way that each channel uses information coming from the others to improve the denoising model.

The so-called *Vectorial Rudin–Osher–Fatemi model* (VROF) for color image denoising is

$$\min_{\mathbf{u} \in BV(\Omega, \mathbb{R}^M)} \int_\Omega |\nabla \mathbf{u}| \, dx + \frac{\lambda}{2} \|\mathbf{u} - \mathbf{f}\|^2_{L^2(\Omega, \mathbb{R}^M)}, \tag{18}$$

where $\int_\Omega |\nabla \mathbf{u}|$ is an alternative notation for the VTV defined by (17), $\mathbf{f} = (f_1, \ldots, f_M)$ is the given noisy vector-valued image and the $L^2(\Omega, \mathbb{R}^M)$-norm is naturally defined as

$$\|\mathbf{u} - \mathbf{f}\|^2_{L^2(\Omega, \mathbb{R}^M)} = \int_\Omega \sum_{m=1}^{M} |u_m - f_m|^2 \, dx.$$

Existence of minimizer for the variational VROF model can be proved by standard arguments [29].

Chambolle's projection algorithm can be easily extended to the vectorial case so that, for any observed color image $\mathbf{f} \in L^2(\Omega, \mathbb{R}^M)$, the minimizer of (18) is computed as

$$\mathbf{u} = \mathbf{f} - \pi_{\frac{1}{\lambda} K_{VTV}}(\mathbf{f}), \tag{19}$$

where $K_{VTV} = \left\{ \mathbf{v} \in L^2(\Omega, \mathbb{R}^M) \, : \, \langle \mathbf{v}, \mathbf{u} \rangle_{L^2(\Omega, \mathbb{R}^M)} \leq \mathrm{VTV}(\mathbf{u}) \ \forall \mathbf{u} \in L^2(\Omega, \mathbb{R}^M) \right\}$. Therefore, algorithm 1 can be easily adapted for computing $\pi_{\frac{1}{\lambda} K_{VTV}}(\mathbf{f})$ in the discrete setting:

$$p_m^{n+1}(i,j) = \frac{p_m^n(i,j) + \delta_t D(\operatorname{div} p_m^n - \lambda f_m)(i,j)}{1 + \delta_t \sqrt{\sum_{m=1}^{M} |D(\operatorname{div} p_m^n - \lambda f_m)(i,j)|^2}}, \quad \forall 1 \leq m \leq M.$$

3 Estimation of λ for Gaussian Noise

The choice of the trade-off parameter λ affects the balance between removing noise and preserving sharp signal content. A straightforward method for parameter tuning is the *discrepancy principle*, according to which λ is selected to match the noise variance σ^2. This method has an observed tendency to overestimate the mean squared error optimal choice of λ to slightly oversmooth the solution [30]. We nevertheless follow it here as a simple automatic selection of the parameter.

For TV-denoising, the discrepancy principle suggests to solve the constrained form of the ROF problem (2)-(3). Note that the first constraint in (3) can be dropped because it is automatically enforced by the evolution procedure (see the paper by Chambolle et al. [7, section §2.1] for more details). Consequently, tuning λ can be approached by solving the following constrained minimization problem:

$$\min_{u\in BV(\Omega)} \int_\Omega |\nabla u(x)|dx \quad \text{subject to} \quad \int_\Omega |u(x)-f(x)|^2 dx = \sigma^2|\Omega|. \tag{20}$$

Let $\langle f\rangle$ denote the mean value of f. Provided that the variance of the observed image is at least as large as the noise level[4],

$$\int_\Omega |f(x)-\langle f\rangle|^2 dx \geq \sigma^2|\Omega|,$$

then it can be shown [7] that there exists a Lagrange multiplier $\lambda > 0$ such that problem (20) has a unique solution that is given by the equivalent unconstrained problem (4). Therefore, there exists a unique value of λ for which the minimizers of the two problems are the same. Unfortunately, the relationship between σ and λ is indirect, there is no closed-form formula to obtain the value of λ corresponding to a particular σ.

In general, σ is less difficult to estimate than λ. To find a good estimate of λ, Chambolle proposed [2] solving (4) through algorithm 2. The algorithm generates a sequence of values that is proved to converge monotonically to the unique λ such that $\|u-f\|_2^2 = \sigma^2$.

Algorithm 2 Tuning λ for Gaussian noise

Input: The noise standard deviation σ and an arbitrary starting value $\lambda_0 > 0$.
Output: The limit value of the trade-off parameter λ.

$\lambda \leftarrow \lambda_0$
while not convergence **do**
  $u \leftarrow f - \frac{1}{\lambda}\operatorname{div} p$ from algorithm 1
  $\lambda \leftarrow \dfrac{\|u-f\|_X}{N\sigma}\lambda$
end while
return λ.

We initialize the iteration process with the following empirical estimate of λ:

$$\lambda_0 = \frac{2.1237}{M\sigma} + \frac{2.0547}{M\sigma^2}, \tag{21}$$

where M is the number of channels of the image and σ is the noise standard deviation relative to the intensity range $[0, 255]$. To speed up computations, note that the current u can be used as the initial guess in the following iteration. The algorithm converges quickly for most images and noise levels. We perform five iterations to tune λ, which is sufficiently accurate so that $\|u-f\|_2^2 \simeq \sigma^2$.

[4]This assumption is likely to be true since f is supposed to have additive noise of variance σ^2.

4 Experimental Results

In this section, we present experimental results illustrating the behavior of Chambolle's projection algorithm for (vectorial) total variation denoising and we display a comparative performance with some state-of-the-art methods.

4.1 Examples of Gaussian Denoising

We use algorithm 1 for Gaussian denoising with λ provided by algorithm 2. The step size used is 0.248, and the criterion for stopping the process just consists in checking that the maximum variation between two consecutive iterations of the dual variable is less than 10^{-2}.

The first example demonstrates how for VTV-regularized Gaussian denoising Chambolle's projection algorithm behaves. Figure 1 displays a color image which has been corrupted with additive white Gaussian noise of standard deviation $\sigma = 20$. The denoised image reveals that the root-mean-square error (RMSE) has been significantly reduced, whereas the so-called peak signal-to-noise ratio[5] (PSNR) has been improved. We observe that while sharp edges have been preserved, some details have been blurred or have disappeared as is highlighted in figure 2. As the noise level increases, singularities on the image and distortions produced by the random noise become indistinguishable from the gradient point of view. Consequently, the VTV-regularization drives to a smoothing process.

Reference image

Noisy image
RMSE = 18.63
PSNR = 22.73

Denoised image
RMSE = 8.84
PSNR = 29.20

Figure 1: From left to right, reference image, noisy image corrupted with Gaussian noise of standard deviation $\sigma = 20$ and denoised image provided by Chambolle's projection algorithm. We observe that much of the noise has been removed while preserving, more or less, the signal content.

To illustrate how the value of the trade-off parameter λ influences the result for TV-regularized Gaussian denoising, Chambolle's projection algorithm is applied to a grayscale image corrupted with Gaussian noise of standard deviation $\sigma = 20$. Figure 3 displays the results for several values of λ. On the one hand, when λ is very small, the image becomes blurred due to an over-smoothing process. This effect can be observed for $\lambda = 0.01$. On the other hand, if $\lambda \geq 0.1$ then the algorithm is not able to remove all noise from the input image. Algorithm 2 gets a compromise between removing noise and preserving signal content. The optimal choice of λ for the noisy image in figure 3 is 0.052, which provides the best result both visually and in terms of RMSE and PSNR.

[5]The *peak signal-to-noise ratio* (PSNR) is the ratio between the reference signal and the distortion signal in an image, given in decibels. The higher the PSNR, the closer the distorted image is to the original. Its value is defined as $\text{PSNR} = 10 \log_{10} \left(\frac{\text{MAX}^2}{\text{MSE}} \right)$, where MSE is the mean-square error between the original and the distortion image, and MAX is the maximum possible pixel value of the image. In our case, since the pixels are represented using 8 bits per sample, MAX = 255.

Reference image | Noisy image | Denoised image

Figure 2: Zoom in on the bar code located in the lower right-hand side of the images displayed in figure 1. Note that some details have been blurred during the denoising process.

Reference image | Noisy image RMSE = 19.94 PSNR = 22.14 | Denoised image $\lambda = 0.1$ RMSE = 8.48 PSNR = 29.56

Denoised image $\lambda = 0.075$ RMSE = 7.35 PSNR = 30.81 | Denoised image $\lambda = 0.052$ RMSE = 7.24 PSNR = 30.94 | Denoised image $\lambda = 0.01$ RMSE = 11.99 PSNR = 26.55

Figure 3: From left to right and from top to bottom, reference image, noisy image corrupted with Gaussian noise of standard deviation $\sigma = 20$ and denoised images provided by Chambolle's algorithm for several λ values. Notice that the noise is not completely removed for $\lambda = 0.1$, whereas the result for $\lambda = 0.01$ becomes blurred. The optimal choice of λ is 0.052, which provides the best result.

The criterion for stopping the algorithm consists in checking that the maximum variation between two consecutive iterations is less than 10^{-2}, as suggested in Chambolle's paper [2]. Nevertheless, we can impose the tolerance to be smaller, even at the expense of increasing the CPU time. The results obtained in figure 4 reveal that not only the denoised image is visually much better (see how the artifacts in the sky have been reduced), but also the PSNR gain is not at all negligible. Consequently, to decrease the tolerance is worthwhile, but at a serious computational cost.

Reference image

Noisy image
RMSE = 18.92, PSNR = 22.59

Denoised image
tolerance = 10^{-2}
RMSE = 9.94, PSNR = 28.19
CPU time = 0.18 seconds

Denoised image
tolerance = 10^{-3}
RMSE = 9.88, PSNR = 28.24
CPU time = 18.94 seconds

Figure 4: From left to right and from top to bottom, reference image, noisy image corrupted with Gaussian noise of standard deviation $\sigma = 20$ and denoised images provided by Chambolle's algorithm using tolerances 10^{-2} and 10^{-3}, respectively. The algorithm was run on a laptop with one core Processor Intel® Core™ i5 CPU M 430 @ 2.27 GHz × 4, with 3.8 GB RAM. We observe that the result provided by the lowest tolerance is noticeably better since, for instance, artifacts in the sky have been reduced.

As noted and modeled by Nikolova [31], a drawback of the TV-regularization is the *staircase effect*, a tendency to produce flat regions separated by artificial edges. In order to illustrate this phenomenon, we have generated a grayscale dégradé image corrupted with Gaussian noise of standard deviation $\sigma = 20$. Figure 5 displays a cross section of the image to visualize the stair steps.

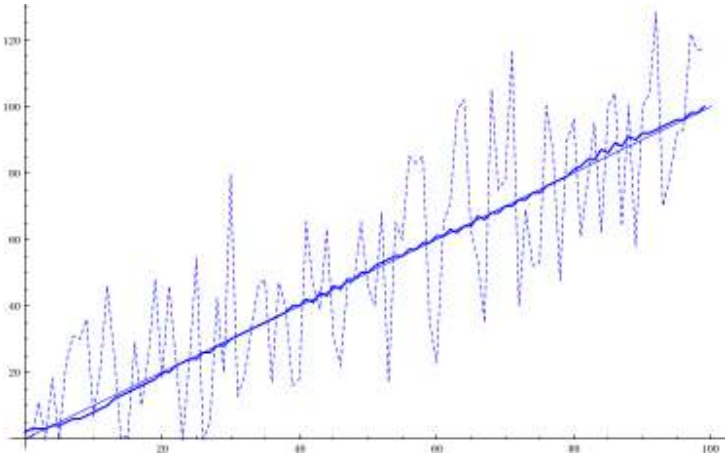


Figure 5: Staircase effect in a grayscale dégradé image. The plot displays the intensity values versus pixel position of the exact (solid line), noisy (dashed line) and denoised (thick line) images.

4.2 Examples of Laplacian and Poisson Denoising

The way the trade-off parameter λ is computed in algorithm 2 for Gaussian noise can be easily adapted to Laplace noise, which fidelity term is $\int_\Omega |u(x) - f(x)|\, dx$. In this case, λ has to be updated at each step of the algorithm as

$$\lambda^{k+1} = \sqrt{\frac{\|u - f\|_X}{N\sigma}}\lambda^k.$$

On the other hand, if the observed image presents inherent Poisson noise, then the fidelity term of the ROF model (4) becomes $\int_\Omega (u(x) - f(x) \log u(x))\, dx$. However, Poisson noise can be approximately reduced to additive white quasi-Gaussian noise through the Anscombe transform [32] and thus algorithm 2 can be used.

Figures 6 and 7 display the denoised images provided by Chambolle's projection algorithm when applied to noisy images corrupted with Laplace and Poisson noise, respectively. The results show that algorithm 1 behaves well in both cases whenever λ is selected appropriately.

Reference image

Noisy image
RMSE = 19.77
PSNR = 22.21

Denoised image
RMSE = 7.43
PSNR = 30.71

Figure 6: From left to right, reference image, noisy image corrupted with Laplace noise of standard deviation $\sigma = 20$ and denoised image provided by Chambolle's algorithm with λ fixed to 0.055.

Reference image

Noisy image
RMSE = 19.93
PSNR = 22.14

Denoised image
RMSE = 7.30
PSNR = 30.87

Figure 7: From left to right, reference image, noisy image corrupted with Poisson noise of standard deviation $\sigma = 20$ and denoised image provided by Chambolle's algorithm with λ fixed to 0.06.

4.3 Comparison with other Denoising Algorithms

The image denoising problem has been intensively studied during the last two decades and numerous algorithms have been proposed and lead to brilliant success. Roughly speaking, the state-of-the-art algorithms can be arranged in three classes: the variational methods, especially those based on the ROF model; methods which try to recover the main structures of the images by using a dictionary; and methods that take advantage of the non-local similarity of patches.

Chambolle's projection approach is a simple, robust and fast algorithm which leads to image results in the same ballpark as, for instance, the split Bregman method [19]. Due to its simplicity and excellent performance, this contribution can be considered in addition as a baseline for comparison and lower bound of performance for newly developed TV-denoising techniques. Nevertheless, other algorithms like split Bregman, which is equivalent to the augmented Lagrangian method introduced by Hestenes [33] and Powel [34] and which convergence was established by Bertsekas [35] and Eckstein [36], are more flexible and can solve more problems than Chambolle's algorithm.

On the other hand, both the dictionary learning and the so-called patch-based methods have much greater computational complexity and are much more slower than most TV-denoising algorithms. To name a few, there are K-SVD [37], NL-Means [38], BM3D [39] and DCT [40]. All of them have public and commented implementations at Image Processing On Line[6].

4.3.1 Comparison with the Split Bregman Method

In order to evaluate the real capacity of Chambolle's approach among TV-denoising algorithms, a precise performance comparison with split Bregman is done. Reliable software on this approach can be found at IPOL webpage [19]. We use the source code provided there and all parameters are selected as given in the corresponding paper. The study has been led on the noise-free color images (the real standard deviation of the noise is $\sigma \ll 1$) displayed in figure 8. Both algorithms have been processed on the noisy images obtained from the original ones after adding white Gaussian noise of several standard deviations $\sigma \in (0, 50]$. All results have been saved in integer values on $[0, 255]$.

Tables 1-2 display the RMSE and the PSNR of the noisy images as well as those corresponding to the denoised images obtained after applying both algorithms for VTV-denoising. By and large,

[6] http://dx.doi.org/10.5201/ipol

Book 774 × 518

Building 704 × 469

Computer 704 × 469

Dice 704 × 469

Flowers 704 × 469

Garden 774 × 518

Hall 704 × 469

Leaves 704 × 469

Traffic 704 × 469

Trees 774 × 518

Valldemossa 774 × 518

Yard 704 × 469

Figure 8: Set of noise-free color images used for the comparison between Chambolle's projection algorithm and the split Bregman method.

we realize that Chambolle's approach provides better results than split Bregman, which is more than evident as the noise level increases. We want to emphasize that the performance of both algorithms on images with outstanding vegetation, such as Trees or Garden, is not as good as on others. This can be easily justified by taking into account that total variation can not distinguish between huge gradients due to noise and huge gradients due to singularities on the image such as vegetation texture.

Image	Noisy image		Chambolle		Split Bregman	
	RMSE	PSNR	RMSE	PSNR	RMSE	PSNR
Book	23.74	22.38	**5.24**	**34.18**	5.50	33.63
Building	22.40	22.83	**9.29**	**29.37**	11.24	27.99
Computer	22.03	22.95	**9.12**	**29.57**	11.72	27.77
Dice	22.70	22.75	**3.29**	**38.39**	5.08	35.23
Flowers	20.57	23.48	**10.29**	**28.46**	14.19	26.18
Garden	21.79	23.02	**13.95**	**26.00**	16.23	25.00
Hall	22.06	22.91	**5.11**	**34.46**	7.28	31.91
Leaves	20.46	23.57	**5.30**	**34.19**	10.75	28.89
Traffic	22.53	22.79	**10.16**	**28.60**	12.03	27.41
Trees	22.77	22.67	**15.28**	**25.23**	16.38	24.82
Valldemossa	23.04	22.64	**10.03**	**28.64**	11.45	27.68
Yard	21.34	23.15	**8.09**	**30.51**	11.11	28.25
Avg.	22.12	22.93	**8.76**	**30.63**	11.08	28.73

Table 1: For each image in figure 8, the averages of both the RMSE and the PSNR over all standard deviations $\sigma \in \{5, 15, 20, 25, 30, 40, 50\}$ are displayed.

σ	Noisy image		Chambolle		Split Bregman	
	RMSE	PSNR	RMSE	PSNR	RMSE	PSNR
5	4.89	34.35	**3.60**	**37.48**	3.96	36.47
10	9.59	28.50	**5.69**	**33.70**	6.15	32.86
15	14.18	25.10	**7.25**	**31.69**	8.05	30.60
20	18.66	22.72	**8.51**	**30.34**	9.82	28.88
25	23.05	20.89	**9.55**	**29.35**	11.51	27.51
30	27.35	19.40	**10.44**	**28.58**	13.14	26.34
40	35.68	17.09	**11.93**	**27.39**	16.39	24.40
50	43.56	15.36	**13.13**	**26.52**	19.63	22.79
Avg.	22.12	22.93	**8.76**	**30.63**	11.08	28.73

Table 2: For each value of σ, the averages of both the RMSE and the PSNR over all color images in figure 8 are displayed.

In addition to RMSE and PSNR values, it is really interesting to compare visually both methods, especially to remark and regret the inherent and characteristic artifacts of each one. For this purpose, we have run both algorithms for the same fixed time and we compare the results at this point. Figure 9 reveals that the result obtained with split Bregman becomes cartoon-like due to an over-smoothing process, which does not happen so much when using Chambolle's projection algorithm.

Finally, we test the speed of both algorithms for VTV-denoising. Computation times are reported using the implementation included with this paper and the codes provided by Pascal Getreuer for the split Bregman method at the IPOL webpage [19]. Both algorithms have been run on a laptop with

Reference image

Noisy image
RMSE = 17.18, PSNR = 23.43

Denoised image by Chambolle
RMSE = 5.05, PSNR = 34.06

Denoised image by split Bregman
RMSE = 8.72, PSNR = 29.32

Figure 9: From left to right and from top to bottom, reference image, noisy image corrupted with Gaussian noise of standard deviation $\sigma = 20$ and denoised images provided by Chambolle's algorithm and split Bregman, respectively. Both methods have been run for the same fixed time and the results are compared at this point. Notice that the split Bregman result becomes cartoon-like, which does not happen so much with Chambolle's approach.

one core[7]. The noisy images have been generated by adding Gaussian noise of standard deviation $\sigma = 20$ to noise-free images of different sizes. The trade-off parameter λ has been taken as the optimal value provided by algorithm 2. We have run both algorithms during the same fixed time and we report the RMSE and the PSNR values of the corresponding denoised images at this point. Table 3 displays the results obtained for several running times. Notice that Chambolle's projection algorithm is significantly more efficient than split Bregman since it can restore corrupted images better than split Bregman for identical stop criterion.

4.3.2 Comparison with the DCT Denoising Method

The Discrete Cosine Transform (DCT) image denoising algorithm is a robust and fast patch-based method, arguably the simplest among all the counterparts but surprisingly effective. Reliable software on this approach is accessible at Image Processing On Line [40]. The comparison has been led on some of the noise-free color images of figure 8, after adding white Gaussian noise of several standard

[7]Processor Intel® Core™ i5 CPU M 430 @ 2.27 GHz × 4, with 3.8 GB RAM.

Image size	Noisy image		CPU time	Chambolle		Split Bregman	
	RMSE	PSNR		RMSE	PSNR	RMSE	PSNR
256 × 256	18.55	22.76	0.050	**7.27**	**30.90**	8.35	29.70
			0.100	**7.17**	**31.02**	8.19	29.87
			0.250	**7.17**	**31.03**	8.21	29.85
			0.500	**7.17**	**31.02**	8.21	29.85
704 × 469	17.18	23.43	0.1	9.16	28.90	**8.71**	**29.33**
			0.3	**7.14**	**31.06**	7.25	30.92
			0.5	**6.75**	**31.54**	7.25	30.92
			1.0	**6.62**	**31.71**	7.26	30.91
1024 × 1024	19.59	22.29	0.5	**9.38**	**28.68**	9.49	28.58
			1.0	**8.89**	**29.15**	9.30	28.76
			2.5	**8.68**	**29.35**	9.29	28.77
			5.0	**8.68**	**29.36**	9.25	28.80

Table 3: For fixed running times, the RMSE and the PSNR of the denoised images provided by Chambolle's projection algorithm and the split Bregman method are displayed. Both algorithms were run on a laptop with one core Processor Intel® Core™ i5 CPU M 430 @ 2.27 GHz × 4, with 3.8 GB RAM. By and large, notice that Chambolle's approach is more efficient than split Bregman.

deviations $\sigma \in (0, 50]$. All results have been saved in integer values on $[0, 255]$. Computation times are reported using the implementation included with this paper on a laptop with one core[7], and the online demo of the DCT algorithm provided at the IPOL webpage.

Table 4 displays the averages of the RMSE, the PSNR and the running times of both algorithms over several noise standard deviations. By and large, DCT provides the best results in terms of both RMSE and PSNR, whereas Chambolle's approach is pretty more efficient. Only in smooth textureless images without significant edges, as Dice, Chambolle's projection algorithm performs better than DCT. Consequently, to use a patch-based method seems to be worthwhile, but at a serious computational cost.

Image	Noisy image		Chambolle			DCT		
	RMSE	PSNR	RMSE	PSNR	CPU time	RMSE	PSNR	CPU time
Dice	22.69	22.76	**3.29**	**38.38**	**0.25**	3.33	38.30	33.29
Leaves	20.44	23.58	5.29	35.44	**0.62**	**4.91**	**35.09**	32.10
Traffic	22.53	22.79	10.16	28.61	**0.25**	**8.69**	**30.08**	32.06
Avg.	21.89	23.04	6.25	34.14	**0.53**	**5.64**	**34.49**	32.15

Table 4: For each image, the averages of the RMSE, the PSNR and of the running times of Chambolle's projection algorithm and DCT method over standard deviations $\sigma \in \{5, 15, 20, 25, 30, 40, 50\}$ are displayed. Chambolle's algorithm was run on a laptop with one core Processor Intel® Core™ i5 CPU M 430 @ 2.27 GHz × 4 with 3.8 GB RAM, whereas the online demo provided at IPOL webpage was used for DCT method. Notice that DCT has the best results in terms of RMSE and PSNR, except for textureless images as Dice. On the contrary, Chambolle's algorithm is more efficient from the computational point of view.

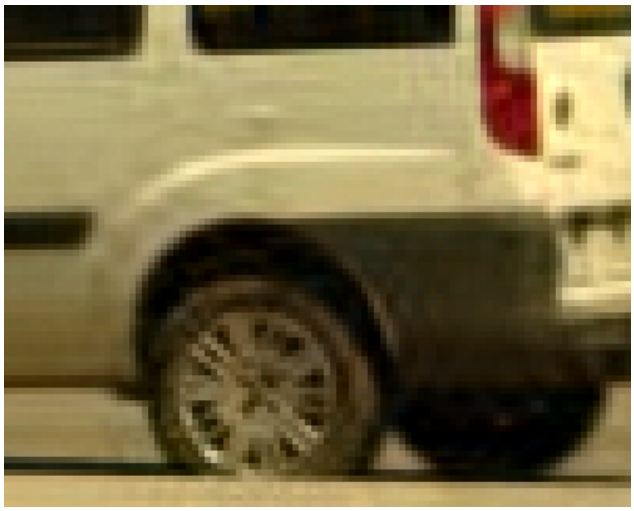
Figure 10 shows the results obtained with both DCT and Chambolle's algorithm on the Traffic image after adding white Gaussian noise of standard deviation $\sigma = 20$. We have zoomed in on the white van that appears on the right-hand side of the corresponding images. On the one hand, having shape adaptive patches in the DCT algorithm definitively improves the performance on regions with singularities. Indeed, edges appear more straight with DCT and some detail has been blurred or has

disappeared in Chambolle's result (see, for instance, the wheel of the van). Furthermore, the result obtained with Chambolle's method becomes cartoon-like due to an over-smoothing process, which does not happen in such considerable way with DCT. On the contrary, one realizes that in quasi-constant areas the DCT denoising algorithm introduces some artifacts that lead to less convincing results.

Reference image

Noisy image
RMSE = 18.92, PSNR = 22.59

Denoised image by Chambolle
RMSE = 9.92, PSNR = 28.20



Denoised image by DCT
RMSE = 8.35, PSNR = 29.70

Figure 10: From left to right and from top to bottom, reference image, noisy image corrupted with Gaussian noise of standard deviation $\sigma = 20$ and denoised images provided by Chambolle's algorithm and DCT method, respectively. Notice that edges and other singularities are better preserved on the DCT result than on the Chambolle's one, which is more blurred. On the contrary, we observe that the DCT algorithm introduces some artifacts in homogeneous areas.

5 On Line Demo

An online demo of Chambolle's projection approach is available at the Image Processing On Line webpage[8]. The purpose is to apply VTV-denoising to an input image by means of algorithm 1. The ANSI C source code (documented) used in this on line demo is available from the article web page.

Inputs are a grayscale or color image, a standard deviation $\sigma > 0$ for the Gaussian noise and a value of $\lambda > 0$ if one wants to fix it during the denoising process. One can upload its own image or, otherwise, one can use any of the images available on the demo page as the algorithm input. These proposed images have almost no noise, having been obtained by first taking a good quality snapshot in full daylight, and then zooming out by a 8-factor.

The algorithm can be used in different ways:

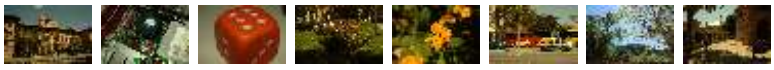
- Upload a noise-free image, add Gaussian noise and then denoise it with dynamic λ computed by algorithm 2. The outputs are the reference image, the noisy image, the denoised image, the difference image (difference between the denoised and the reference images with stretched dynamic range on $[0, 255]$ by an affine contrast change), the RMSE and the PSNR ratio of both noisy and denoised images, and the values of λ tuning.
- Upload a noise-free image, add Gaussian noise and then denoise it with fixed λ. The outputs are the reference image, the noisy image, the denoised image, the difference image, the RMSE and the PSNR of both noisy and denoised images.
- Upload a noisy image, indicate the standard deviation σ and then denoise it with dynamic λ computed by algorithm 2. The outputs are the input noisy image, the denoised image, the residual noise image (difference between the denoised and the noisy images with stretched dynamic range on $[0, 255]$ by an affine contrast change) and the values of λ tuning.
- Upload a noisy image, indicate the standard deviation σ and then denoise it with fixed λ. The outputs are the input noisy image, the denoised image and the residual noise image.

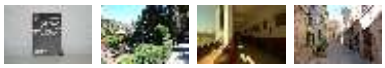
We recall that there exist methods that allow one to estimate the variance of a signal independent additive white Gaussian noise. We propose Ponomarenko's algorithm [41] available at IPOL webpage [42] to estimate the noise level. Finally, the algorithm tolerance is fixed to 10^{-3} for the online demo.

Acknowledgements

The authors were supported by the Ministerio de Ciencia e Innovación under grant TIN2011-27539. During this work, the first author benefited from a fellowship of the Conselleria d'Educació, Cultura i Universitats of the Govern de les Illes Balears for the realization of his Ph.D. thesis, which has been selected under an operational program co-financed by the European Social Fund.

Image Credits

 by M.Colom, CC-BY.

 by A.Buades, CC-BY.

 Lena standard test image.

[8] https://doi.org/10.5201/ipol.2013.61

References

[1] L.I. Rudin, S. Osher, E. Fatemi. "Nonlinear total variation based noise removal algorithms". *Physica D*, vol. 60, pp. 259-268, 1992. http://dx.doi.org/10.1016/0167-2789(92)90242-F

[2] A. Chambolle. "An algorithm for total variation minimization and applications". *Journal of Mathematical Imaging and Vision*, vol. 20, pp. 89-97, 2004. http://dx.doi.org/10.1023/B:JMIV.0000011325.36760.1e

[3] P. Blomgren, T.F. Chan. "Color TV: total variation methods for restoration of vector valued images". *IEEE Transactions on Image Processing*, vol. 7(3), pp. 304-309, 1998. http://dx.doi.org/10.1109/83.661180

[4] J. Weickert. "Anisotropic diffusion in image processing". ECMI Series. Teubner–Verlag, Stuttgart, 1998. ISBN: 9783519026068.

[5] F. Catté, P.-L. Lions, J.-M. Morel, T. Coll. "Image selective smoothing and edge detection by nonlinear diffusion". *SIAM Journal on Numerical Analysis*, vol. 29(1), pp. 182-193, 1992. http://dx.doi.org/10.1137/0729012

[6] V.K. Ivanov. "Conditions for well-posedness in the Hadamard sense in spaces of generalized functions". *Siberian Mathematical Journal*, vol. 28(6), pp. 906-911, 1987. http://dx.doi.org/10.1007/BF00969468

[7] A. Chambolle, P.-L. Lions. "Image recovery via total variation minimization and related problems". *Numerische Mathematik*, vol. 76, pp. 167-188, 1997. http://dx.doi.org/10.1007/s002110050258

[8] L. Ambrosio, N. Fusco, D.Pallara. "Functions of bounded variation and free discontinuity problems". Oxford Mathematical Monographs. Oxford University Press, USA, 2000. ISBN: 9780198502456.

[9] A. Haddad. "Stability in a class of variational methods". *Applied and Computational Harmonic Analysis*, vol. 23, pp. 57-73, 2007. http://dx.doi.org/10.1016/j.acha.2006.10.005

[10] M. Zhu, S.J. Wright, T.F. Chan. "Duality-based algorithms for total-variation-regularized image restoration". *Journal Computational Optimization and Applications* vol. 47(3), pp. 377-400, 2010. http://dx.doi.org/10.1007/s10589-008-9225-2

[11] C. Couprie, L. Grady, H. Talbot, L. Najman. "Combinatorial continuous maximum flows". *SIAM Journal on Imaging Sciences*, vol. 4, pp. 905-930, 2011. http://dx.doi.org/10.1137/100799186

[12] M.K. Ng, L. Qi, Y.-F. Yang, Y.-M. Huang. "On semismooth Newton's methods for total variation minimization". *Journal of Mathematical Imaging and Vision*, vol. 27(3), pp. 265-276, 2007. http://dx.doi.org/10.1007/s10851-007-0650-0

[13] A. Chambolle, J. Darbon. "On total variation minimization and surface evolution using parametric maximum flows". *International Journal of Computer Vision*, vol. 84(3), pp. 288-307, 2009. http://dx.doi.org/10.1007/s11263-009-0238-9

[14] A. Beck, M. Teboulle. "A fast iterative shrinkage-thresholding algorithm for linear inverse problems".*SIAM Journal on Imaging Sciences*, vol. 2(1), pp. 183-202, 2009. http://dx.doi.org/10.1137/080716542

[15] P.-L. Lions, B. Mercier, "Splitting algorithms for the sum of two nonlinear operators". *SIAM Journal on Numerical Analysis*, vol. 16, pp. 964-979, 1979. http://dx.doi.org/10.1137/0716071

[16] C. Chaux, P.L. Combettes, J.-C. Pesquet, V.R. Wajs. "A variational formulation for frame-based inverse problems". *Inverse Problems*, vol. 23(4), 2007. http://dx.doi.org/10.1088/0266-5611/23/4/008

[17] A. Beck, M. Teboulle. "Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems". *IEEE Transactions on Image Processing*, vol. 18(11), pp. 2419-2434, 2009. http://dx.doi.org/10.1109/TIP.2009.2028250

[18] T. Goldstein, S. Osher. "The split Bregman method for L^1 regularized problems". *SIAM Journal on Imaging Sciences*, vol. 2(2), pp. 323-343, 2009. http://dx.doi.org/10.1137/080725891

[19] P. Getreuer. "Rudin-Osher-Fatemi total variation denoising using split Bregman". *Image Processing On Line*, 2012 http://dx.doi.org/10.5201/ipol.2012.g-tvd

[20] T.F. Chan, G.H. Golub, P. Mulet. "A nonlinear primal-dual method for total variation-based image restoration". *SIAM Journal on Scientific Computing*, vol. 20(6), pp. 1964-1977, 1999. http://dx.doi.org/10.1137/S1064827596299767

[21] J.L. Carter. "Dual methods for total variation-based image restoration". PhD. Thesis, U.C.L.A., 2001.

[22] J. Bect, L. Blanc-Féraud, G. Aubert, A. Chambolle. "A l^1-unified variational framework for image restoration". In proceedings of *European Conference on Computer Vision*, Lecture Notes in Computer Science, pp. 1-13, 2004. http://dx.doi.org/10.1007/978-3-540-24673-2_1

[23] F. Li, Z. Bao, R. Liu, G. Zhang. "Fast image inpainting and colorization by Chambolle's dual method". *Journal of Visual Communication and Image Representation*, vol. 22(6), pp. 529-542, 2011. http://dx.doi.org/10.1016/j.jvcir.2011.06.006

[24] I. Ekeland, R. Temam. "Convex analysis and variational problems". Classics in applied mathematics. Society for Industrial and Applied Mathematics, Amsterdam, 1976. ISBN: 9780444108982.

[25] J.-B. Hiriart–Urruty, C. Lemaréchal. "Convex analysis and minimization algorithms I, II". Volumes 305-306 of *Fundamental Principles of Mathematical Sciences*. Springer-Verlag, Berlin, 1993. ISBN: 9780387568508 (I), 9783540568506 (II).

[26] P.G. Ciarlet. "Introduction à l'analyse numérique matricielle et à l'optimisation". Collection of Applied Mathematics for the Master's Degree. Masson, Paris, 1990. ISBN: 9782100041671.

[27] J.-F. Aujol. "Some first-order algorithms for total variation based image restoration". *Journal of Mathematical Imaging and Vision*, vol. 34(3), pp. 307-327, 2009. http://dx.doi.org/10.1007/s10851-009-0149-y

[28] A. Bermúdez, C. Moreno. "Duality methods for solving variational inequalities". *Computer and Mathematics with Applications*, vol. 7(1), pp. 43-58, 1981. http://dx.doi.org/10.1016/0898-1221(81)90006-7

[29] X. Bresson, T.F. Chan. "Fast dual minimization of the vectorial total variation norm and applications to color image processing". *Inverse Problems and Imaging*, vol. 2(4), pp. 455-484, 2008. http://dx.doi.org/10.3934/ipi.2008.2.455

[30] N.P. Galatsanos, A.K. Katsaggelos. "Methods for choosing the regularization parameter and estimating the noise variance in image restoration and their relation". *IEEE Transactions on Image Processing*, vol. 1(3), pp. 322-336, 1992. http://dx.doi.org/10.1109/83.148606

[31] M. Nikolova. "Local strong homogeneity of a regularized estimator". *SIAM Journal on Applied Mathematics*, vol. 61(2), pp. 633-658, 2000. http://dx.doi.org/10.1137/S0036139997327794

[32] F.J. Anscombe. "The transformation of Poisson, binomial and negative-binomial data". *Biometrika*, vol. 35(3), pp. 246-254, 1948. http://dx.doi.org/10.2307/2332343

[33] M.R. Hestenes. "Multiplier and gradient methods". *Journal of Optimization Theory and Applications*, vol. 4, pp. 303-320, 1969. http://dx.doi.org/10.1007/BF00927673

[34] M.J.D. Powell. "A method for nonlinear constraints in minimization problems". In *Optimization*, Academic Press, New York, pp.283-298, 1969.

[35] D.P. Bertsekas. "Constrained optimization and Lagrange multiplier methods". *Academic Press*, New York, 1982.

[36] J. Eckstein. "Approximate iterations in Bregman-function-based proximal algorithms". *Mathematical Programming*, vol. 83(1-3), pp. 113-123, 1998. http://dx.doi.org/10.1007/BF02680553

[37] M. Lebrun, A. Leclaire. "An implementation and detailed analysis of the K-SVD image denoising algorithm". *Image Processing On Line*, 2012. http://dx.doi.org/10.5201/ipol.2012.llm-ksvd

[38] A. Buades, B. Coll, J.-M. Morel. "Non-local means denoising". *Image Processing On Line*, 2011. http://dx.doi.org/10.5201/ipol.2011.bcm_nlm

[39] M. Lebrun. "An analysis and implementation of the BM3D image denoising method". *Image Processing On Line*, 2012. http://dx.doi.org/10.5201/ipol.2012.l-bm3d

[40] G. Yu, G. Sapiro. "DCT image denoising: a simple and effective image denoising algorithm". *Image Processing On Line*, 2011. http://dx.doi.org/10.5201/ipol.2011.ys-dct

[41] N.N. Ponomarenko, V.V. Lukin, S.K. Abramov, K.O. Egiazarian, J.T. Astola. "Blind evaluation of additive noise variance in textured images by nonlinear processing of block DCT coefficients". *Image Processing: Algorithms and Systems II*, SPIE Proceedings, vol. 5014, pp.178-189, 2003. http://dx.doi.org/10.1117/12.477717

[42] M. Colom, A. Buades. "Analysis and extension of the Ponomarenko et al. method, estimating a noise curve from a single image". *Image Processing On Line*, 2013. http://dx.doi.org/10.5201/ipol.2013.45