

Regularized non-local Total Variation and application in image restoration

Zhi Li · François Malgouyres · Tiejong Zeng

Received: date / Accepted: date

Abstract In the usual non-local variational models, such as the non-local total variations (NLTV), the image is regularized by minimizing an energy term that penalizes gray-levels discrepancy between some specified pairs of pixels, a weight value is computed between these two pixels to penalize their dissimilarity. In this paper, we impose some regularity to those weight values. More precisely, we minimize a function involving a regularization term, analogous to an H^1 term, on weights. Doing so, the finite differences defining the image regularity depend on their environment. When the weights are difficult to define, they can be restored by the proposed stable regularization scheme.

We provide all the details necessary for the implementation of a PALM algorithm with proved convergence. We illustrate the ability of the model to restore relevant unknown edges from the neighboring edges on an image inpainting problem. We also argue on inpainting, zooming and denoising problems that the model better recovers thin structures.

Z. Li
Department of Computational Mathematics, Science and Engineering, Michigan State University, MI, USA
E-mail: zhi_li@life.hkbu.edu.hk. ZL is supported in part by the NSF grant DMS-1621798.

F. Malgouyres
University of Toulouse, CNRS, Toulouse, France,
E-mail: Francois.Malgouyres@math.univ-toulouse.fr.

T. Zeng
Department of Mathematics, Hong Kong Baptist University, Hong Kong, China,
E-mail: zeng@hkbu.edu.hk. TZ is partially supported by NSFC 11271049, RGC 12302714 and RFGs of HKBU.

Keywords non-local regularization · proximal alternating linearized minimization · nonconvex minimization · total variation · image restoration

Mathematics Subject Classification (2000)
49N45 · 65K10 · 90C26

1 Introduction

1.1 The Image Restoration Problem

Given an image domain \mathcal{P} (\mathcal{P} can be a sub-domain of \mathbb{R}^2), we consider a degraded image I as a mapping $\mathcal{P} \rightarrow \mathbb{R}$. We assume that I is obtained from a perfect unknown image \bar{u} by

$$I = H\bar{u} + b, \quad (1)$$

where H is a known linear operator and b is an error term. In the typical situations we have in mind, \mathcal{P} is a lattice of pixels and b is an iid noise with standard deviation $\sigma > 0$.

The purpose of the restoration model studied in this paper is to retrieve an image close to \bar{u} , using only I and H . For instance, in image inpainting, the operator H maps any image u onto an image Hu satisfying

$$(Hu)_p = \begin{cases} 0 & , \text{ if } p \in \mathcal{P}', \\ u_p & , \text{ otherwise,} \end{cases} \quad (2)$$

where \mathcal{P}' is the domain to inpaint.

For deblurring, the operator H is typically the convolution operator. For the purpose of image zooming the operator H is the composition of a convolution and a subsampling operator (the latter can also be written under the form (2)) and so on and so forth. Note that, to simplify notations,

we use the same domain \mathcal{P} for the perfect image \bar{u} and the measure I . This need not be the case in general.

Finally, for simplicity, we assume that \mathcal{P} is periodic, so that for any $(p, q) \in \mathcal{P}^2$ the sum $p + q$ is defined in such a manner that $p + q \in \mathcal{P}$. We also assume that \mathcal{P} is such that $p + \mathcal{P} = \mathcal{P}$, whatever $p \in \mathcal{P}$.

1.2 Review of Image Restoration Models

Total variation (TV) has drawn much attention since Rudin, Osher and Fatemi (ROF) proposed their celebrated model [39] for denoising (i.e. $H = Id$). The variational formulation can be written as

$$\operatorname{argmin}_{u \in \Omega} |\nabla u| + \lambda \|u - I\|^2, \quad (3)$$

where Ω is either $\mathbb{R}^{\mathcal{P}}$ (in the discrete setting) or the space of the functions of bounded variation. When used as a regularization term, the total variation $|\nabla u|$ has the advantage to preserve sharp edges and the drawbacks to create homogeneous regions (staircasing) and to over-smooth the textured regions. This is due to the fact that textures are treated as singularities by the TV term which only sees local information [25]. This regularization has later been used to solve many linear inverse problems such as inpainting [5, 41].

In another direction, generalizing the Yaroslavsky neighborhood filter [47], the non-local means (NLM) of Buades, Coll and Morel [8] takes the following form (we write it in the continuous domain):

$$NLM_I(p) = \frac{1}{C(p)} \int_{\mathcal{B}} v_I(p, p') I(p') dp', \quad (4)$$

where \mathcal{B} is a search window, the weights are provided by

$$v_I(p, p') = \exp\left(-\frac{D(p, p')}{h^2}\right), \quad (5)$$

where, for a patch $\hat{\mathcal{B}}$ (usually a square centered around the origin),

$$D(p, p') = \int_{\hat{\mathcal{B}}} G_a(q) |I(p+q) - I(p'+q)|^2 dq \quad (6)$$

and the normalizing factor

$$C(p) = \int_{\mathcal{B}} v_I(p, p') dp'. \quad (7)$$

The Gaussian kernel G_a with standard deviation a ponderates pixels in the 'patch' area and h is a filtering parameter. Usually in NLM model, the

patch is nothing but a square region which is smaller than the search window \mathcal{B} . The idea behind NLM is to restore the pixel in the contaminated image I by a weighted average of the nearby similar (in the patch sense) pixels. The patch is used to judge the resemblance between two pixels p and p' , in (6), $D(p, p')$ is evaluated based on the difference of the square neighborhoods (patches) $I(p+q)$ and $I(p'+q)$ centered at p and p' , not the intensity difference between the pixels p and p' themselves. Another key point is that the pixels p' used in the average do not have to be close to p , which explains the name 'non-local'.

The idea to use patch was initially known for its ability to do 'texture synthesis' [30, 44], it was later applied to image denoising and inpainting [1]. Patch is widely used in image processing models. For example, Pizarro, et al. [38] used a linear combination of a similarity term and a smoothness term, both of them are defined non-locally in a patch manner to propagate nonlocal information. NL-Bayes [28, 29] bridges between the Bayesian methods and BM3D. It improves the denoising result, when applied to color images and, for gray level images, yields to results comparable to state-of-the-art methods, such as BM3D.

From a weighted graph point of view, Kheradmand and Milanfar [27] proposed an adaptive Laplacian filter using patch, which is mainly for deblurring, and under special circumstances, their method can apply to denoising and sharpening problems. Notice that a similar idea, coming with a precise statistical analysis, has been proposed under the name "Discrete Universal Denoiser" [46].

Considering a thin structure (typically a smooth curve) in a uniform background, we see in the formula that the weights v_I does not differentiate well the pixels in the thin structure from pixel in the background. The thin structure indeed lives in a domain whose measure is small (at the limit, the measure of the domain is 0). It provides a negligible contribution to $D(p, p')$. As a consequence, NLM tends to underestimate the weights of the pixels in the thin structure and, as a consequence, it tends to erase the thin structure. This is especially true for large patch size and may lead to a substantial underestimation of the similarity between two pixels in the thin structure [18].

Moreover, exemplar-based approaches [30, 44] also use patches to produce striking inpainting results. These algorithms have been later interpreted using a probabilistic point of view. In particular, Levina and Bickel [30] provided a theoretical back-

ground to explain their ability to synthesize texture. Aujol, Ladjal and Masnou [4] expressed the exemplar-based methods in a non-local variational framework and studied their ability to retrieve geometric features. However, it is difficult to adapt NLM and other patch based approaches for difficult image restoration tasks involving highly structured noise or (even worse) when the operator H is not invertible and makes the recovery of similar pixels difficult.

Inspired by the advantages of NLM, Gilboa and Osher [23, 24] introduced a non-local variational model to better recover textures. In [24], they considered the non-local total variation (NLTV) model defined (in the continuous domain) by

$$J_{NLTV}(u) + \lambda \|u - I\|^2, \quad (8)$$

where the NLTV term $J_{NLTV}(u)$ is defined by

$$\int_{\mathcal{P}} \sqrt{\int_{\mathcal{B}} (u(p) - u(p+q))^2 v(p,q) dq dp}. \quad (9)$$

In [24], in practical applications, they calculated the weights v according to patch distance and only used the 10 most similar pixels in (the discretized) \mathcal{B} . They also mentioned binary weights. They adapted the celebrated Chambolle's projection algorithm [10] to optimize a discretization of the NLTV model. A more general model, involving p-Laplace operator on graphs, have been proposed in [19]. It also makes connections between the variational approaches and other non-local approaches.

Considering applications such as compressive sensing, when the low sampling rate may lead to a bad initial guess of the weight function, some authors [37, 48] suggested to update the weights during the iterations to improve them. For example, Zhang, et al. [48] employed patches and updated the weights by recomputing the initial step (5) with the current estimate of the clean image. A similar update is used in [1, 37]. To trade off between the accuracy and computational costs, the authors only reevaluate the weights every few steps [48] and in practical implementation, they only use the first few largest weights in a local search window and set the rest of them to zero. Of course, because of these last 'tricks', the algorithms may fail to find the optimal solution of the variational model which is not fully satisfactory. They however provide interesting restoration results.

The model studied in this paper is inspired by the earlier work of Peyre, et al. [36, 37]. In [36], they considered the weights v defined in the NLM and updated v using (5) with the current estimate

of the image. The authors mentioned the difficulty for setting the filtering parameter h [36] since noise level evolves along the iterative process. In [37], the authors extended [36] and [48] by regularizing the graph v . The resulting model has two regularization terms, the first term $J_v(u)$ regularizes the image, while the second term $E(v)$ is the entropy of function v defining the underlying graph. Also, to better recover the textures of the images, the non-local regularization term was chosen to be the patch variation of $J_{NLTV}(u)$ in (9). In this variation, the difference of intensities $u(p) - u(p+q)$, in (9), is replaced by a patch distance. It has also been adopted in [1, 20, 35].

To conclude, non-local variational models [17, 24, 31, 37] have been widely used to restore images, because of their capability to utilize the information from many pixels having a similar neighborhood. However, when treated by these models, thin structures can be overlooked and are sometimes undetectable in a patch of the damaged image. These methods might therefore erase these thin structures.

Also, the construction of the weights between the pixels is usually done according to some pixel similarity criterion in the degraded images I . When some weights are unknown (for instance when inpainting a missing domain or zooming an image), they are updated during the iterative process, according to an ad-hoc procedure [37]. More generally, these methods are difficult to apply and lack precise modeling when the output of the degradation operator H does not look like an image (think about the restoration an image from random measurements or Computerized Tomography.)

When compared to these methods, the model studied in this paper is a regularized non-local total variation (RNLTV) model. It shows the advantages to both: - better restore thin structures; - allow the construction of non-local models for difficult inverse problems.

1.3 The Studied Model

1.3.1 Framework and Model Description

From now on, we discretize the problem and consider a lattice \mathcal{P} . We consider a set $\mathcal{B} \subset \mathcal{P}$. In the typical situation we have in mind, \mathcal{B} is a square centered at the origin with radius¹ r . Throughout the paper, we call patch or window a set $p + \mathcal{B}$ for some $p \in \mathcal{P}$.

¹ We use $r = 5$ in the experiments.

We also denote the cardinality of any finite set C by $|C|$. Moreover, for any finite dimensional vector space V (in practice we have either $V = \mathbb{R}^{\mathcal{P}}$, $V = \mathbb{R}^{\mathcal{P} \times \mathcal{B}}$ or $V = \mathbb{R}^{\mathcal{B}}$) and any positive definite operator L mapping V into itself, we consider the weighted norm [15]

$$\|x\|_L = \langle x, Lx \rangle^{\frac{1}{2}}, \quad \forall x \in V, \quad (10)$$

where $\langle \cdot, \cdot \rangle$ denotes the usual scalar product in V . When L is an identity matrix ($L = Id$), we simply denote $\|x\|$ instead of $\|x\|_{Id}$.

Given these notations, we define, for any $p \in \mathcal{P}$, weights $v^p = (v_q^p)_{q \in \mathcal{B}} \in \mathbb{R}^{\mathcal{B}}$, where each weight v_q^p is the cost for the dissimilarity $|u_p - u_{p+q}|$. Said differently, $v^p \in \mathbb{R}^{\mathcal{B}}$ can be seen as the parameters of a finite difference operator defined at the pixel p . In practice we will force these weights $v^p \in \mathbb{R}^{\mathcal{B}}$ to belong to the set

$$\mathcal{U} = \left\{ v \in \mathbb{R}_+^{\mathcal{B}}, \sum_{q \in \mathcal{B}} v_q = 1 \right\}, \quad (11)$$

where \mathbb{R}_+ are the non-negative reals. This constraint guarantees that every pixel of the image contributes similarly to the overall cost. We denote the collection of all the weights in the image by $v = (v^p)_{p \in \mathcal{P}} \in \mathcal{U}^{\mathcal{P}}$.

Given v , we define the non-local total variation of $u \in \mathbb{R}^{\mathcal{P}}$ by

$$\begin{aligned} TV(v, u) \\ = \sum_{p \in \mathcal{P}} \Psi_{\mu} \left(\sqrt{\sum_{q \in \mathcal{B}} v_q^p (u_p - u_{p+q})^2} \right), \end{aligned} \quad (12)$$

where $\mu > 0$ and Ψ_{μ} is the Moreau envelope of the absolute value function, also called Huber function [6, page 368]:

$$\psi_{\mu}(t) = \begin{cases} |t| - \frac{\mu}{2}, & \text{if } |t| \geq \mu, \\ \frac{t^2}{2\mu}, & \text{otherwise.} \end{cases} \quad (13)$$

Note that (9) is a continuous analogue of (12) without the Huber function Ψ_{μ} . The first motivation for using the Huber function is that, once the weights are properly estimated, we expect v_q^p to be small when $(u_p - u_{p+q})^2$ is large. As a consequence, we do not expect edges to be oversmoothed and blurred. As a consequence, the benefit of the ℓ^1 term used to define the total variation might be questioned and it is interesting to use a function Ψ_{μ} , involving a parameter μ , that permit Ψ_{μ} to behave quadratically or linearly. The second motivation for using the Huber function is that it is differentiable. This

is required by the PALM algorithm. Moreover, we consider, for any $v \in \mathcal{U}^{\mathcal{P}}$, the regularity criterion R defined by

$$R(v) = \gamma \sum_{p \in \mathcal{P}} \sum_{p' \in \mathcal{N}} \|v^p - v^{p+p'}\|^2, \quad (14)$$

where γ is a non-negative parameter and \mathcal{N} is a small neighborhood² of 0. When minimized, the term $R(v)$ favors the situation when v^p is similar to $v^{p+p'}$. If we isolate v_q^p in $TV(v, u) + R(v)$, we observe that v_q^p should minimize

$$\Psi_{\mu} \left(\sqrt{\sum_{q' \in \mathcal{B}} v_{q'}^p (u_p - u_{p+q'})^2} \right) + \gamma \sum_{p' \in \mathcal{N}} (v_q^p - v_q^{p+p'})^2.$$

As a consequence, v_q^p must be small when $(u_p - u_{p+q})^2$ is large and must be similar to the weights at its neighbor $v_q^{p+p'}$. Also, since $v \in \mathcal{U}^{\mathcal{P}}$, the $TV(\cdot, u)$ term forces v_q^p to be large for p and q such that $(u_p - u_{p+q})^2$ is small. For instance, when p and $p+q$ are on the same side of an edge or in the same thin structure; we know that $(u_p - u_{p+q})^2$ is small; it does not cost much to have v_q^p large (i.e. close to 1). Although it depends on the local structure in the image, we also often have $(u_{p+p'} - u_{p+p'+q})^2$ small and therefore $v_q^{p+p'}$ large. Altogether, we expect v to end up aligned with such edges and thin structures. This looks like a desirable behavior.

Finally, we consider a data fidelity term D . This term either takes the form

$$D(u) = \lambda \|Hu - I\|^2, \quad (15)$$

for some parameter $\lambda \geq 0$, or, for a parameter $\tau \geq 0$,

$$D(u) = \chi_{\|Hu - I\|^2 \leq \tau}(u), \quad (16)$$

where the indicator function $\chi_{|C|}$ is 0 if the condition C is satisfied and infinity otherwise. The advantage of the latter formulation is that τ can be deduced from the noise standard deviation σ (A standard choice is indeed $\tau = |\mathcal{P}|\sigma^2$).

The restored image we are interested in is the argument u of a minimizer of

$$\operatorname{argmin}_{u \in \mathbb{R}^{\mathcal{P}}, v \in \mathcal{U}^{\mathcal{P}}} R(v) + TV(v, u) + D(u). \quad (17)$$

Although the parameters are not presented explicitly in (17), the model depends on the parameters μ , γ and either λ or τ .

² In the experiments, we consider 4-connectivity: $\mathcal{N} = \{(1, 0), (0, 1)\}$.

Notice that, the model does not make use of the patch distance. It can therefore be categorized as a purely non-local method.

In the problem (17), the constraint on v is compact and, unless in very particular counter-example, we expect the objective function to be coercive in u . The objective function being also lower semi-continuous, the problem (17) has a solution. We will see moreover that TV and R are differentiable and have Lipschitz gradient (see Proposition 1, 2 and 3). The main difficulty concerning the numerical resolution of (17) is that, because of the term $TV(u, v)$, the objective function is not convex.

1.3.2 Insight on the Model

The connectivity principle [26, 32] states that human prefers to see that disjoint edges are connected through the missing domains. It is well-known that the classic inpainting models including TV models [5, 41], have difficulty in connecting the thin objects [11, 12] (see Figure 1, 'TV' inpainting). Thus inpainting methods involving higher order information [11, 40], such as curvature, were proposed to improve the results. However, if the size of the unknown regions separating thin objects are too large, then even the fourth-order PDE-based model [40] will leave the thin structures broken (see Figure 1, 'TV-H⁻¹' inpainting).

Moreover, we see on Figure 1, 'NLinpaint' image, that even non-local methods that do not construct v with enough care [21] have difficulty connecting lines, when there are many large missing areas. One motivation for considering RNLTV is that, because it smoothly inpaints v in the missing regions and since large weights values are concentrated and aligned along the direction of the thin structures, RNLTV has the ability to connect the broken lines.

1.4 Overview of the Paper

In Section 1.3, we present the RNLTV model. Section 2 details the proximal alternating linearized minimization (PALM) algorithm [7] minimizing the proposed model. We also provide in this section the gradient of the different terms, show that the gradients are Lipschitz continuous and provide their Lipschitz constants. The numerical experiments in Section 3 demonstrate the efficiency of the proposed method on inpainting, zooming and denoising experiments.

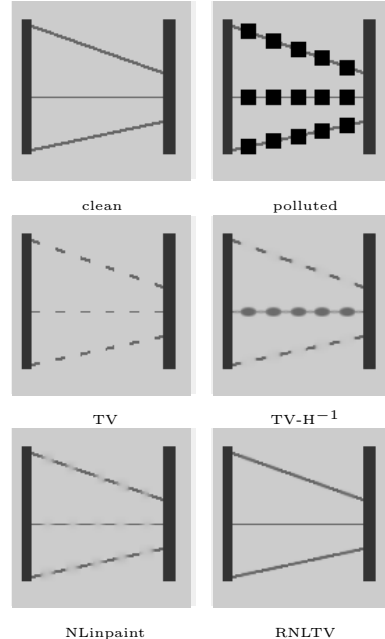


Fig. 1 A synthetic image with thin lines contaminated by 11×11 missing squares. All methods fail to follow the Connectivity Principle properly, but RNLTV can connect the broken thin lines.

2 Implementing the PALM Algorithm

In the last couple of years, alternating minimization algorithms [3, 7, 13, 14, 15, 34] were developed to solve optimization problems over multiple variables. A proof of convergence in the non-convex setting for an alternating variant of the forward-backward algorithm, tagged PALM, has been proposed in [7]. It is perfectly adapted to the structure of (17). A preconditioned version has also been studied in [15].

Below, we provide the details for implementing the PALM algorithm generating a sequence that converges to a stationary point of (17). First, we give a few notations and describe the algorithm in Section 2.1. Next, we derive the components of the algorithm. Section 2.2, 2.3 and 2.4 are devoted to the calculation of the gradients of the regularization terms in (17) and their Lipschitz constants. Then we discuss the proximal operator $\text{prox}_L^D(u')$ in Section 2.5 in detail.

2.1 Notations and Overview of the Algorithm

We denote $\nabla_u TV(v, u)$ and $\nabla_v TV(v, u)$ as the partial gradient of $TV(., .)$ with regards to the variable u and v respectively. The PALM algorithm adapted to (17) is described in Algorithm 4.

In summary, the Algorithm 4 alternates forward-backward iterations for each one of the two block of variables: u and v .

For any $v \in \mathcal{U}^{\mathcal{P}}$ and any $u \in \mathbb{R}^{\mathcal{P}}$, we assume that we know a linear operator L and constants l and l' (for the detailed calculations of these quantities see Section 2.2, 2.3 and 2.4) satisfying the majorization conditions [15] described below:

- For any $v \in \mathcal{U}^{\mathcal{P}}$, L is such that there exists $m > 0$ satisfying

$$\|u\|_L \geq m\|u\|, \quad \forall u \in \mathbb{R}^{\mathcal{P}}, \quad (18)$$

and for any u and $u' \in \mathbb{R}^{\mathcal{P}}$

$$\begin{aligned} TV(v, u') &\leq TV(v, u) \\ &\quad + \langle \nabla_u TV(v, u), u' - u \rangle \\ &\quad + \frac{1}{2} \langle u' - u, L(u' - u) \rangle. \end{aligned} \quad (19)$$

- For any $u \in \mathbb{R}^{\mathcal{P}}$, $l \geq 0$ and $l' \geq 0$ are such that $l + l' > 0$ and for any v and $v' \in \mathcal{U}^{\mathcal{P}}$

$$\begin{aligned} TV(v', u) &\leq TV(v, u) \\ &\quad + \langle \nabla_v TV(v, u), v' - v \rangle \\ &\quad + \frac{l}{2} \|v' - v\|^2, \end{aligned} \quad (20)$$

and

$$\begin{aligned} R(v') &\leq R(v) + \langle \nabla R(v), v' - v \rangle \\ &\quad + \frac{l'}{2} \|v' - v\|^2. \end{aligned} \quad (21)$$

Moreover, the proximal operators used in Algorithm 4 are defined, for any $u' \in \mathbb{R}^{\mathcal{P}}$, by

$$\text{prox}_L^D(u') = \underset{u \in \mathbb{R}^{\mathcal{P}}}{\text{argmin}} D(u) + \frac{1}{2} \|u - u'\|_L^2, \quad (22)$$

and for any $v' \in \mathcal{U}^{\mathcal{P}}$,

$$\text{prox}^{\chi|\mathcal{U}^{\mathcal{P}}}(v') = \underset{v \in \mathcal{U}^{\mathcal{P}}}{\text{argmin}} \|v - v'\|^2, \quad (23)$$

simply is the orthogonal projection onto $\mathcal{U}^{\mathcal{P}}$.

Given the above quantities l , l' and L , the Algorithm 4 has many advantages and is perfectly adapted to the structure of our problem. The latter indeed satisfies the following properties:

1. The objective function is analytical and satisfies the KL property.
2. $R(v)$, $TV(v, u)$ and $D(u)$ are proper, non-negative and lower semi-continuous.
3. The partial gradients $\nabla_u TV(v, u)$ and $\nabla_v TV(v, u)$ are Lipschitz continuous with bounded moduli.

Algorithm 1: Overview of the algorithm

Input:

I : Degraded image
 H : Linear operator to invert
 \mathcal{B} : Support of the weights
 \mathcal{N} : Support for the finite differences defining R
 $\gamma, \mu, \lambda/\tau$: parameters

Output:

u : Restored image
 v : Weights

begin

Initialize the image u and v ;

while *not converged* **do**

step 1: Update u :

$$u = \text{prox}_L^D(u - L^{-1} \nabla_u TV(v, u)) \quad (24)$$

step 2: Update v :

$$v = \text{prox}^{\chi|\mathcal{U}^{\mathcal{P}}}\left(v - \frac{\nabla_v TV(v, u) + \nabla R(v)}{l + l'}\right)$$

4. The sequence generated from Algorithm is bounded since both u and v are belong to compact sets.

It is shown in [7, 15] that the sequence generated by (even an inexact version of) Algorithm 4 has good convergence properties:

- The objective function decays along the iterative process.
- The iterates converge to a stationary point of our problem.
- The sum over the iterative process of the distance between two successive iterates is finite.

Of course, calculating the Kurdyka-Lojasiewicz characterization of our objective function would provide a better understanding of the convergence quality but it is not the purpose of the paper.

Finally, in order to implement the algorithm, we need to

- compute $\nabla_u TV(v, u)$ and construct the operator L satisfying (19) (see Section 2.2);
- compute $\nabla_v TV(v, u)$ and the constant l satisfying (20) (see Section 2.3);
- compute $\nabla R(v)$ and the constant l' satisfying (21) (see Section 2.4);
- compute the proximal operator (22) (see Section 2.5);
- compute the proximal operator (23): Notice that this reduces to multiple projections onto a simplex. The latter projection is a well studied problem for which fast algorithms exists [16].

2.2 Calculation of $\nabla_u TV(v, u)$ and L

For any $v \in \mathcal{U}^{\mathcal{P}}$, we consider the following non-local analogue of the finite difference operator:

$$\begin{aligned} \mathbf{D}_v : \quad \mathbb{R}^{\mathcal{P}} &\longrightarrow \mathbb{R}^{\mathcal{P} \times \mathcal{B}} \\ u = (u_p)_{p \in \mathcal{P}} &\longmapsto \left(\sqrt{v_q^p} (u_p - u_{p+q}) \right)_{(p,q) \in \mathcal{P} \times \mathcal{B}} \end{aligned} \quad (25)$$

We also denote, for $p \in \mathcal{P}$,

$$(\mathbf{D}_v u)_p = ((\mathbf{D}_v u)_{p,q})_{q \in \mathcal{B}} \in \mathbb{R}^{\mathcal{B}}$$

$$\text{and } \|(\mathbf{D}_v u)_p\| = \sqrt{\sum_{q \in \mathcal{B}} (\mathbf{D}_v u)_{p,q}^2}.$$

We obtain after some simple calculations that its adjoint is given, for any $w = (w_{p,q})_{(p,q) \in \mathcal{P} \times \mathcal{B}}$, it is easy to see that

$$(\mathbf{D}_v^* w)_p = \sum_{q \in \mathcal{B}} \left(\sqrt{v_q^p} w_{p,q} - \sqrt{v_q^{p-q}} w_{p-q,q} \right), \quad (26)$$

for all $p \in \mathcal{P}$.

Moreover, if we consider, for $w \in \mathbb{R}^{\mathcal{P} \times \mathcal{B}}$, the norm

$$\|w\|_{\infty,2} = \max_{p \in \mathcal{P}} \|w_p\|, \text{ where } w_p = (w_{p,q})_{q \in \mathcal{B}} \in \mathbb{R}^{\mathcal{B}}. \quad (27)$$

We can prove (see appendix 5.1) that for any $u \in \mathbb{R}^{\mathcal{P}}$ and any (fixed) $v \in \mathcal{U}^{\mathcal{P}}$

$$TV(v, u) = \max_{\|w\|_{\infty,2} \leq 1} \langle \mathbf{D}_v u, w \rangle - \frac{\mu}{2} \|w\|^2. \quad (28)$$

From this expression and using the Theorem 1 in [33] (reminded in Appendix 5.2), we can deduce the following proposition whose proof is detailed in Appendix 5.2. Notice that a similar statement and construction is given for the usual total variation in [45].

Proposition 1 *For any $u \in \mathbb{R}^{\mathcal{P}}$, we have*

$$\nabla_u TV(v, u) = \mathbf{D}_v^* w^*(u), \quad (29)$$

where $w^*(u) \in \mathbb{R}^{\mathcal{P} \times \mathcal{B}}$ is the maximizer of (28) and is provided in closed form by:

$$w^*(u)_{p,q} = \begin{cases} \frac{(\mathbf{D}_v u)_{p,q}}{\mu} & , \text{ if } \|(\mathbf{D}_v u)_p\| \leq \mu, \\ \frac{(\mathbf{D}_v u)_{p,q}}{\|(\mathbf{D}_v u)_p\|} & , \text{ otherwise.} \end{cases} \quad (30)$$

Moreover, $u \rightarrow \nabla_u TV(v, u)$ is Lipschitz continuous with Lipschitz constant

$$l^v = \frac{\sqrt{2} \sqrt{|\mathcal{B}| + 1}}{\mu}. \quad (31)$$

As a consequence, we have for any u and $u' \in \mathbb{R}^{\mathcal{P}}$

$$\begin{aligned} TV(v, u') &\leq TV(v, u) \\ &+ \langle \nabla_u TV(v, u), u' - u \rangle \\ &+ \frac{l^v}{2} \|u' - u\|^2. \end{aligned} \quad (32)$$

A straightforward consequence of (32) is that for any diagonalizable linear operator G whose eigenvalues are all larger than 1 (and in particular for $G = Id$), the property (19) holds for $L = l^v G$.

2.3 Calculation of $\nabla_v TV(v, u)$ and l

Let us first define, for any fixed $u \in \mathbb{R}^{\mathcal{P}}$, the linear operator \mathbf{A}_u :

$$\begin{aligned} \mathbb{R}^{\mathcal{P} \times \mathcal{B}} &\longrightarrow \mathbb{R}^{\mathcal{P}}, \\ (v_{p,q})_{(p,q) \in \mathcal{P} \times \mathcal{B}} &\longmapsto \left(\sum_{q \in \mathcal{B}} v_{p,q} (u_p - u_{p+q})^2 \right)_{p \in \mathcal{P}}. \end{aligned} \quad (33)$$

The interest for this operator comes from the fact that for any $v \in \mathcal{U}^{\mathcal{P}}$

$$TV(v, u) = \sum_{p \in \mathcal{P}} \Psi_{\mu} \left(\sqrt{(\mathbf{A}_u v)_p} \right). \quad (34)$$

Notice that, when $v \in \mathcal{U}^{\mathcal{P}}$, we have for any $p \in \mathcal{P}$

$$(\mathbf{A}_u v)_p \geq 0. \quad (35)$$

We easily obtain that the adjoint \mathbf{A}_u^* of \mathbf{A}_u is given for any $w \in \mathbb{R}^{\mathcal{P}}$ by

$$(\mathbf{A}_u^* w)_{p,q} = w_p (u_p - u_{p+q})^2, \quad \forall (p,q) \in \mathcal{P} \times \mathcal{B}. \quad (36)$$

Using these notations, we can prove (see Appendix 5.3) the following proposition.

Proposition 2 *For any fixed $u \in \mathbb{R}^{\mathcal{P}}$, we have for any $v \in \mathcal{U}^{\mathcal{P}}$*

$$\nabla_v TV(v, u) = \mathbf{A}_u^* u^*(v), \quad (37)$$

where for all $p \in \mathcal{P}$

$$u^*(v)_p = \begin{cases} \frac{1}{2\sqrt{(\mathbf{A}_u v)_p}} & , \text{ if } \sqrt{(\mathbf{A}_u v)_p} \geq \mu, \\ \frac{1}{2\mu} & , \text{ if } \mu > \sqrt{(\mathbf{A}_u v)_p} \geq 0. \end{cases} \quad (38)$$

Moreover, the function $v \mapsto TV(v, u)$ is concave over $\mathbb{R}_+^{\mathcal{P} \times \mathcal{B}}$ and (20) holds for $l = 0$.

2.4 Calculation of $\nabla R(v)$ and l'

Let us first define the linear operator B :

$$\begin{aligned} \mathbb{R}^{\mathcal{P} \times \mathcal{B}} &\longrightarrow \mathbb{R}^{\mathcal{P} \times \mathcal{B} \times \mathcal{N}}, \\ (v_{p,q})_{(p,q) \in \mathcal{P} \times \mathcal{B}} &\longmapsto (v_{p,q} - v_{p+p',q})_{(p,q,p') \in \mathcal{P} \times \mathcal{B} \times \mathcal{N}}. \end{aligned} \quad (39)$$

We immediately deduce from (14) that for any $v \in \mathbb{R}^{\mathcal{P} \times \mathcal{B}}$

$$R(v) = \gamma \|Bv\|^2. \quad (40)$$

We also easily obtain that the adjoint B^* of B is provided for any $w \in \mathbb{R}^{\mathcal{P} \times \mathcal{B} \times \mathcal{N}}$ by

$$(B^*w)_{p,q} = \sum_{p' \in \mathcal{N}} w_{p,q,p'} - w_{p-p',q,p'}, \forall (p,q) \in \mathcal{P} \times \mathcal{B}. \quad (41)$$

We can now state the following proposition whose proof is provided in Appendix 5.4.

Proposition 3 *For any $v \in \mathcal{U}^{\mathcal{P}}$, we have*

$$\nabla R(v) = 2\gamma B^*Bv. \quad (42)$$

Moreover, $v \rightarrow \nabla R(v)$ is Lipschitz continuous with Lipschitz constant $6\sqrt{2}\gamma|\mathcal{N}|$. As a consequence, (21) holds for $l' = 6\sqrt{2}\gamma|\mathcal{N}|$.

2.5 Calculation of $\text{prox}_L^D(u')$

Below, we provide closed form expressions that permit to compute $\text{prox}_L^D(u')$ for the two data fidelity terms considered in this paper. In the denoising case, H is a identity; for the purpose of image inpainting, H multiply every entry of the image by either 1 or 0 (see (2)).

– **When** $D(u) = \lambda\|u - I\|^2$ **and** $L = l'' Id$.

In this case, we want to compute

$$\begin{aligned} &\text{prox}_L^{\lambda\|u-I\|^2}(u') \\ &= \underset{u \in \mathbb{R}^{\mathcal{P}}}{\text{argmin}} \lambda\|u - I\|^2 + \frac{l''}{2}\|u - u'\|^2, \\ &= \frac{2\lambda}{2\lambda + l''}I + \frac{l''}{2\lambda + l''}u'. \end{aligned}$$

– **When** $D(u) = \chi_{\|u-I\|^2 \leq \tau}(u)$ **and** $L = l'' Id$.

In that setting, we have

$$\begin{aligned} &\text{prox}_L^{\chi_{\|u-I\|^2 \leq \tau}}(u') \\ &= \underset{u: \|u-I\|^2 \leq \tau}{\text{argmin}} \|u - u'\|_L^2 \\ &= \begin{cases} u' & , \text{ if } \|u' - I\|^2 \leq \tau, \\ I + \frac{\sqrt{\tau}}{\|u' - I\|}(u' - I) & , \text{ otherwise.} \end{cases} \end{aligned}$$

– **When** $D(u) = \lambda\|Hu - I\|^2$

In this case,

$$\begin{aligned} &\text{prox}_L^{\lambda\|Hu-I\|^2}(u') \\ &= \underset{u \in \mathbb{R}^{\mathcal{P}}}{\text{argmin}} \lambda\|Hu - I\|^2 + \frac{l''}{2}\|u - u'\|^2, \\ &= u^*, \end{aligned}$$

where for all $p \in \mathcal{P}$

- if H is defined by (2) for some fixed subset of pixels $\mathcal{P}' \subset \mathcal{P}$

$$u_p^* = \begin{cases} u'_p & , \text{ if } p \in \mathcal{P}', \\ \frac{2\lambda I_p + l'' u'_p}{2\lambda + l''} & , \text{ otherwise.} \end{cases} \quad (43)$$

- if H is a general linear operator (e.g. the subsampling operator used for zooming) then the solution u^* of

$$2\lambda H^T(Hu - I) + l''(u - u') = 0, \quad (44)$$

can be computed by the conjugate gradient (CG) method for a sparse linear equation of the form

$$Au = b, \quad (45)$$

where $A = (2\lambda H^T H + l'' Id)$ and $b = 2\lambda H^T I + l'' u'$. Note that, although we use CG (call Matlab function `pcg(A,b)`) to solve this sub-problem, the elementwise max norm of the residual $\max |Au^* - b|$ is around 10^{-13} .

3 Numerical experiments

In order to illustrate the behavior of the model, we consider in this section three applications: image inpainting, zooming and denoising. We evaluate the ability of the method to restore both synthetic and natural images with a particular emphasis on the restoration of thin structures. We compare the performance of the proposed RNLTV model with the performance of state of the art methods. We also illustrate how the weights v are adjusted once the convergence has been reached using the Isotropy Map described in Section 3.1 and 3.2. Finally, Section 3.3 is devoted to the inpainting experiments, Section 3.4 presents zooming results and Section 3.5 contains the experiments on the denoising problem.

Notice that we have used the different data-fidelity terms mentioned in Section 1.3.1. They all show similar computational cost and convergence properties. For completeness, we summarize the data fidelity term chosen for each experiments in Table 1.

Table 1 The different data-fidelity terms used in the experiments.

	$D(u)$
inpainting and zooming	$\lambda \ Hu - I\ ^2$
denoising (synthetic image)	$\chi_{\ u - I\ ^2 \leq \tau}(u)$
denoising (natural image)	$\lambda \ u - I\ ^2$

(-5,5)	(-4,5)	...	(4,5)	(5,5)
(-5,4)	(-4,4)	...	(4,4)	(5,4)
\vdots	\vdots		\vdots	\vdots
(-5,-4)	(-4,-4)	...	(4,-4)	(5,-4)
(-5,-5)	(-4,-5)	...	(4,-5)	(5,-5)

Fig. 2 The window \mathcal{B} of size 11×11 . Every $q \in \mathcal{B}$ has a coordinate.

3.1 The Isotropy Map

To represent the distribution of the weights v , we introduce the isotropy map $M(v) \in \mathbb{R}^{\mathcal{P}}$. At every pixel $p \in \mathcal{P}$, the feature $M(v)_p$ represents whether the weights in v^p are distributed isotropically or not. Intuitively, the distribution is not isotropic when the significant weights are concentrated along a thin region of the window $p + \mathcal{B}$. The feature therefore needs to reflect the discrepancy between the projections of v^p onto two axes one of which is aligned with the orientation of v^p , the other being orthogonal. Notice that the discrepancy is maximal when the coordinate system is aligned with the orientation of v^p .

3.1.1 Implementation Details

More precisely, given an angular precision parameter $K \in \mathbb{N}^3$ we define the small angle $\theta = \frac{\pi}{2K}$ as shown in Figure 3. Then we consider all the coordinate (x, y) clock-wise rotated by an angle $m\theta$, for $m \in \mathcal{M}$, with $\mathcal{M} = \{0, 1, 2, \dots, K - 1\}$. Doing so, we get several orthonormal bases denoted by

$$\begin{aligned} i_m &= (\sin(m\theta), \cos(m\theta)), \\ j_m &= (\cos(m\theta), -\sin(m\theta)). \end{aligned}$$

³ In the experiments, we use $K = 90$.

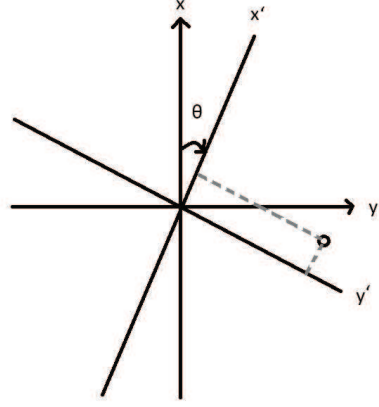


Fig. 3 Coordinate change and projection of the weights on the new coordinate system.

We remind that any pixel $q \in \mathcal{B}$ is represented in the basis (i_m, j_m) by the coordinates

$$(\langle q, i_m \rangle, \langle q, j_m \rangle). \quad (46)$$

Therefore, the average distance to the origin of the projections of v^p onto the axes i_m and j_m are defined by

$$P_m^x = \sum_{q \in \mathcal{B}} |v_q^p \langle q, i_m \rangle|, \quad (47)$$

$$P_m^y = \sum_{q \in \mathcal{B}} |v_q^p \langle q, j_m \rangle|. \quad (48)$$

The isotropy map is then defined by

$$M(v)_p = \log \left(\max \left(\left\{ \frac{P_m^x}{P_m^y}, \frac{P_m^y}{P_m^x} \right\}_{m \in \mathcal{M}} \right) \right). \quad (49)$$

Clearly, $M(v)_p$ is non-negative. Also, it is easy to check that the only way to get small $M(v)_p$ is that $P_m^x \sim P_m^y$ for all $m \in \mathcal{M}$. This implies that v^p is somewhat isotropic. On the contrary, if $M(v)_p$ is large, then there exists $m \in \mathcal{M}$ such that P_m^x and P_m^y are dissimilar. From this condition, we conclude that v^p distributes anisotropically. In summary, $M(v)$ can be taken as an indication of the degree of isotropy of v^p , for every pixel $p \in \mathcal{P}$.

3.2 Compare RNLTV with NLTVG

In this section, we compare the proposed RNLTV with NLTVG [37], we use the abbreviation NLTVG, since the authors present this model using a graph defined by v . In the NLTVG, the authors update

the weights v during iterations in order to minimize an entropy term. This is the model most similar to RNLTV. However, NLTVG also uses patch distance. For the considered problems, the best results for NLTVG are obtained for patches of moderate size (the radius of patch equals to 2). Indeed, as already mentioned, patch distance overlooks thin structures when patches are large⁴. To get a fair comparison, we use the same window size for NLTVG and RNLTV, the radius $r = 5$. Also we use the same initial weights for both methods when dealing with the same problem.

To illustrate the difference with the updated weights, we plot the isotropy maps of weights in Figure 5 and 7, and the restored images are shown in Figure 4 and 6 respectively, also the PSNR values are listed below each figure. NLTVG blurs the image and the isotropy maps also show that the weights computed from NLTVG are not as good as the ones got from RNLTV.

In Figure 5, we plot the isotropy map $M(v) \in \mathbb{R}^P$ for the initial and final weights v in RNLTV. This illustrates that RNLTV has the ability to fill in the missing regions with reasonable weights v . As can be seen in the first column of Figure 5, the initial weights are distributed isotropically both in the regions to be inpainted and the homogeneous regions. However, the regularization term $R(v)$ permits to interpolate 'coherent weights' in the missing areas. For example, due to the influence of their neighbors, the inpainted weights located on the lost thin curves are anisotropic and align with their neighbors.

In the denoising cases, we can see that from the second column of Figure 7, the weights value restored by NLTVG is more isotropic than RNLTV results in the last column, thus NLTVG does not regularize image value u well, some thin lines are still vague in Figure 6. In RNLTV model, we can correct the weights iteratively (see the third column in Figure 7). This limits the influence of noise, thus the majority of noise got removed at convergence.

3.3 Inpainting

In this section, we use two masks presented in Figure 8, where the black pixels indicate the region to be inpainted. We set values of parameters in RNLTV as follow: $\lambda = 5000$ and $\mu = 0.8$ where

⁴ Notice that similar properties have been observed in [18] for the Non-Local Mean: small details such as thin lines can fade away when using large patches.

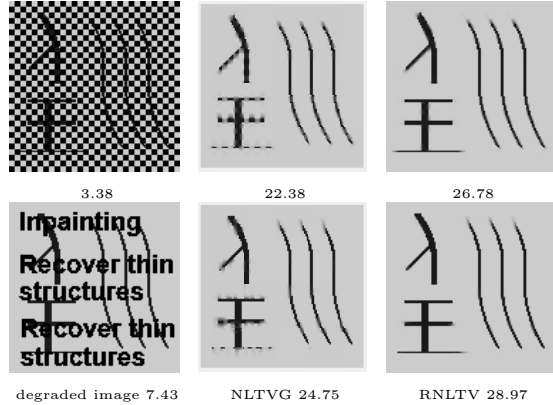


Fig. 4 Results of different methods when restoring the degraded synthetic image with the checkerboard and word masks. (PSNR values are also listed below)

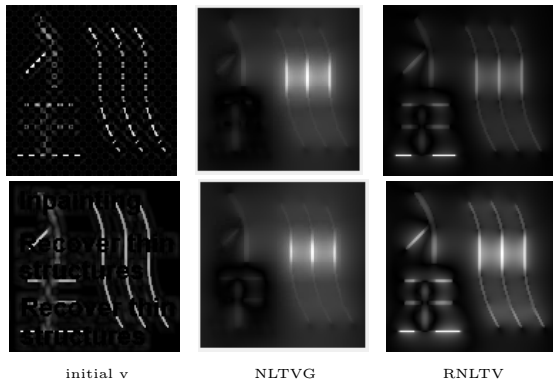


Fig. 5 Plots of the isotropy map $M(v) \in \mathbb{R}^P$ calculated from weights v in RNLTV. The corresponding restored images u are in Figure 10. Top: for the checkerboard mask; Bottom: for the word mask. Left: initial guesses of the weights computed from the damaged image; Middle: weights v after convergence of NLTVG; Right: weights v after convergence of RNLTV.

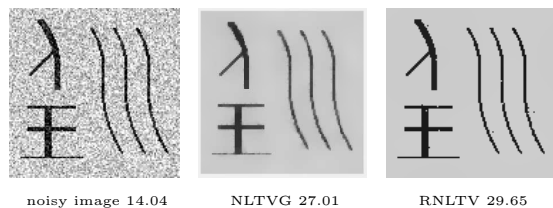


Fig. 6 Denoising results for an additive white Gaussian noise with standard deviation $\sigma = 0.16$. (PSNR values are also listed below)

we remind that the image graylevels range in $[0, 1]$. The number of iterations⁵ is 3000, the radius $r = 5$

⁵ We are aware of the fact that designing a good stopping criterion would permit to save time. However, since the paper presents a new model, we preferred to loose computational time in order to obtain results that truly reflect the behavior of this model.

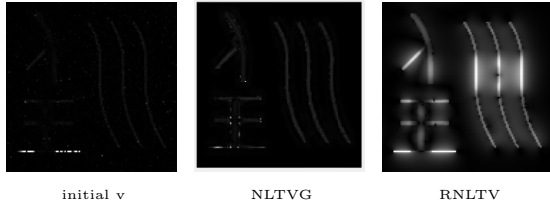


Fig. 7 The isotropy map $M(v) \in \mathbb{R}^P$ calculated from weights v . The corresponding denoised images u are in Figure 14. Left: initial guesses of the weights computed from the noisy image; Middle: weights v after convergence of NLTVG; Right: weights v after convergence of RNLTV.

and the γ values are in Table 2. For most pixel pairs, the initial weight values (before projection) are computed using

$$v_q^p = \exp\left(-\frac{(I_p - I_{p+q})^2}{h^2}\right), \quad (50)$$

where I is the polluted image and for some parameter $h > 0$. If one of the two pixels is in the missing domain, then we set the weight value between them to 0. Then, we project⁶ the weights onto the set defined by (11). We compare the result of the model with some related methods:

- inpaintn: the fully automated MATLAB function inpaintn [22,43] minimizes an energy functional involving a regularization by a squared finite difference and is solved with a Discrete Cosine Transform;
- TV inpainting;
- TV- H^{-1} inpainting [9, 40];
- NLTVG [37];
- NLinpaint [2,21]: NLinpaint implemented three exemplar-based inpainting schemes, namely patch NL-means, patch NL-medians and patch NL-Poisson, under the variational framework suggested by [2]. We provide, in the following experiments, the best result of the three methods for patch sizes between 5 and 15, using the default parameter of the 'Image Processing On Line' [2].

3.3.1 Synthetic Image

In Figure 10, RNLTV outperforms the other methods. There are two strong strokes in the synthetic image: the arc in the character in the top left corner, and the cross in the middle of the character in the bottom left corner. From the first row of Figure 10, we observe that only RNLTV can connect

⁶ Note that, if all the are in the missing domain, then the algorithm assigns weights uniformly equal to $1/|\mathcal{B}|$.

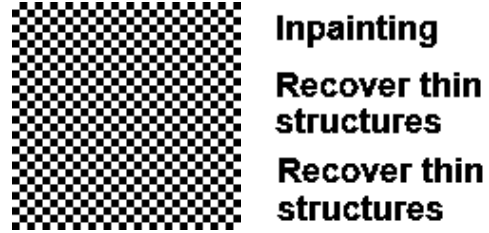


Fig. 8 Inpainting masks with missing data marked in black. Left: checkerboard mask; Right: word mask

Table 2 Values of the parameters γ of RNLTV for the restoration of the images in Figure 9, when corrupted by the checkerboard or word mask (see Figure 8). The other parameters are $\mu = 0.8$, $\lambda = 5000$. The restoration results are presented in Figure 10 and 11.

	checkerboard mask	word mask
	γ	γ
Synthetic	3	2
Boat	3	3
Barbara	4	3
Fingerprint	3	3

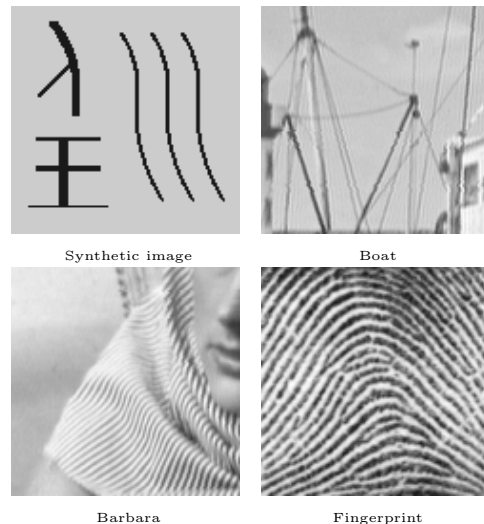


Fig. 9 Test images.

all the thin edges correctly. The TV regularization propagates the background color into the missing domains (see the 3 curved lines on the right). The method inpaintn and TV- H^{-1} cannot clearly inpaint the missing edges in the missing squares on the thin edges. For example, in the middle parts of the 3 curved lines. The method inpaintn creates nonexisting white spots, and blurs the rest of the 3 curved lines. The TV- H^{-1} model connects the thin edge with a wrong shape and a wrong intensity. Taking the character on the bottom left corner for example, we can see that TV- H^{-1} fills larger in-

tensity values into some missing squares located on the top horizontal line, which yields lighter color than it is supposed to be, while puts smaller intensity values into some missing squares located on the bottom horizontal line. The NLTVG and NLinpaint also get blurry results. The RNLTV model connects all the edges correctly except at the end points, see the ends of the 3 curved lines.

3.3.2 Natural Image

In this section, we inpaint natural images in the regions described by the two masks in Figure 8. The test images are Boat, Barbara and Fingerprint (see Figure 9). They contain isolated thin structures or textures where thin structures are grouped. From the PSNR value listed in Table 3 and Figure 11, it is obvious that RNLTV can outperform the other methods in restoring images corrupted by the checkerboard mask, but the PSNR data also indicate that NLinpaint and the proposed RNLTV went into a tie when restoring image corrupted by the word mask, so we zoom-in those restoration results of NLinpaint and RNLTV where thin structure presents, to show that RNLTV can better recover thin structures in Figure 12.

In the first three rows of Figure 11, TV, inpaintn, TV- H^{-1} , NLinpaint can not handle images masked by the checkerboard, the continuous patterns such as the ropes in Boat, stripes in Barbara and friction ridges in Fingerprint can no longer be found in their restoration results. TV regularization takes the color of the background and fills it into the missing areas, see the restored Boat in the first row, second column. Even worse, inpaintn and TV- H^{-1} return blurry restoration results. The method inpaintn creates non-existing white spots around the ropes (the first row, third column). TV- H^{-1} draws fake lines crossing the friction ridges (the third row, fourth column). NLTVG cannot clearly restore stripes on the scarf in Barbara. However, the proposed RNLTV can connect the missing thin curves in those missing squares, see the stripe pattern on the scarf in Barbara (the second row, last column) those well contrasted friction ridges in Fingerprint (the third row, last column). Also, in the last three rows of Figure 11, TV, inpaintn and TV- H^{-1} can not restore those word-shape missing areas, for example, the thin ropes become dashed lines after TV inpainting (the fourth row, second column).

In Figure 12, we zoom-in the restoration results for the word mask in the last three rows of Figure 11 mainly to show that, compare to NLinpaint, we

can better detect thin features from RNLTV results. It is obvious that NLinpaint leaves artifacts around thin ropes in Boat, see the first row, fifth column in Figure 12, which may cause distraction. The well-aligned stripe pattern in the middle is smoothed out by NLTVG and NLinpaint, see the second row, fifth and sixth column in Figure 12, but RNLTV still keeps majority of those stripes on the scarf (the second row, last column)

Table 3 PSNR values of different methods when inpainting with the checkerboard and word masks. Sy: Synthetic image; Bo: Boat; Ba: Barbara; Fi: Fingerprint; cb: the checkerboard mask; wd: the word mask

mask	image	polluted	TV	inpaintn	TV- H^{-1}	NLTVG	NLinpaint	RNLTV
cb	Sy	3.38	18.55	19.61	20.01	22.38	19.45	26.78
	Bo	5.41	25.53	26.82	26.88	27.02	26.73	28.60
	Ba	5.31	24.94	24.50	24.75	26.14	25.41	28.36
	Fi	7.08	16.47	19.60	19.25	19.35	18.11	22.69
wd	Sy	7.43	20.80	21.81	23.64	24.75	25.30	28.97
	Bo	9.24	15.38	15.26	30.79	33.13	32.92	33.79
	Ba	9.24	29.06	28.38	28.64	29.52	34.96	33.71
	Fi	11.04	21.08	10.57	24.36	26.86	28.82	27.73
average		7.27	21.48	20.82	24.79	26.14	26.46	28.83

3.4 Zooming

We compare the zooming results of RNLTV with the results of three methods: the bicubic interpolation, the TV based restoration and NLTVG. In all the experiments, the image data I_d is computed by the Matlab function `imresize` ($I_d = \text{imresize}(\bar{u}, 0.25)$) from an original image \bar{u} with a subsampling of a factor 4, as presented in the first column of Figure 13. The bicubic interpolation is simply implemented by calling the Matlab function `imresize` again (`imresize(ud, 4, 'bicubic')`). The TV based zooming is implemented with a penalty term and a very large parameter $\lambda = 1000$ for which the sampling constraint is satisfied. In the RNLTV model, we set $\lambda = 1000$ and $\mu = 0.6$. We initialize v uniformly with the value $1/|\mathcal{B}|$.

Table 4 PSNR values of the different methods for zooming of images downsampled by a factor of 4.

	bicubic	TV	NLTVG	RNLTV
Vessel	34.42	12.69	34.63	35.15
Ship	24.58	22.90	23.92	24.46

The thin blood vessels are clearly zoomed by RNLTV as presented in the first row, last column of Figure 13. Furthermore, Figure 13 shows a

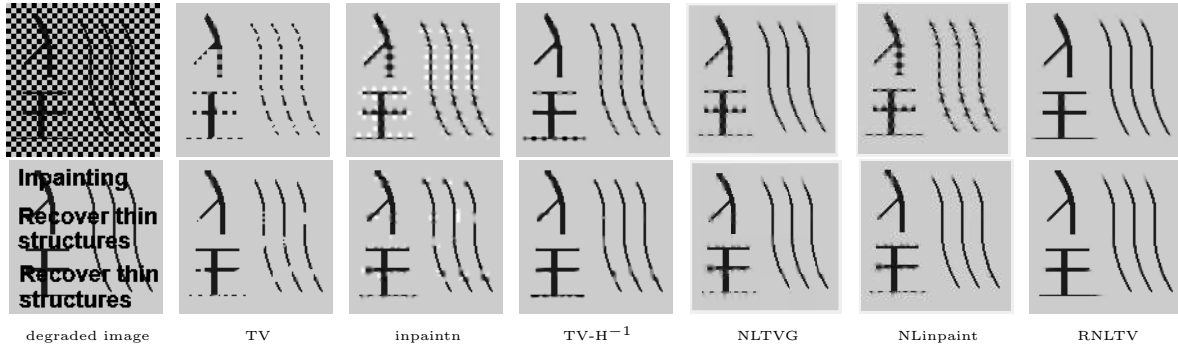


Fig. 10 Results of different methods when restoring the degraded synthetic image with the checkerboard and word masks presented in Figure 8.

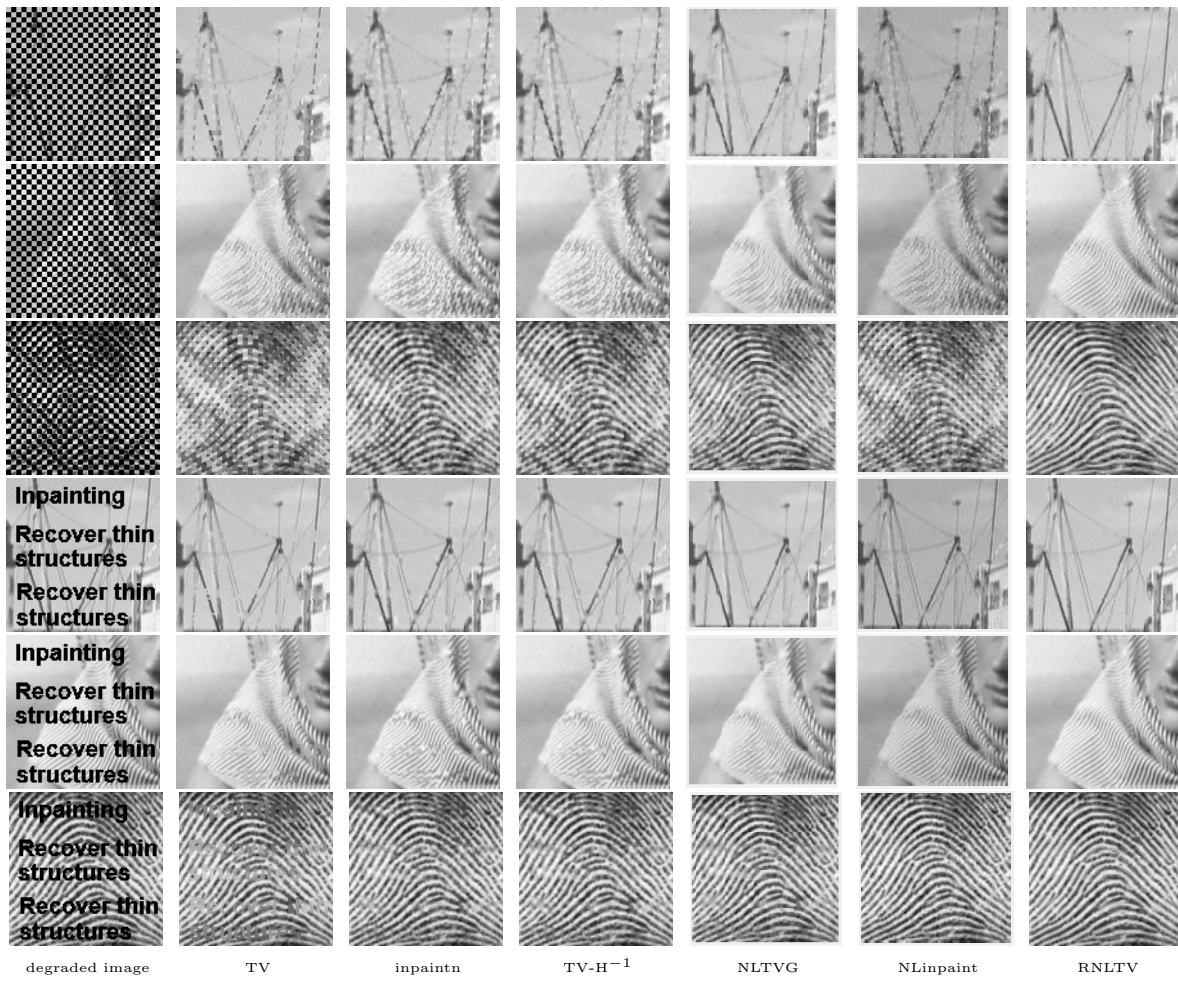


Fig. 11 Restoration results for different methods when inpainting with the checkerboard (the first,second and third rows) and word (the fourth, fifth and sixth rows) masks presented in Figure 8.

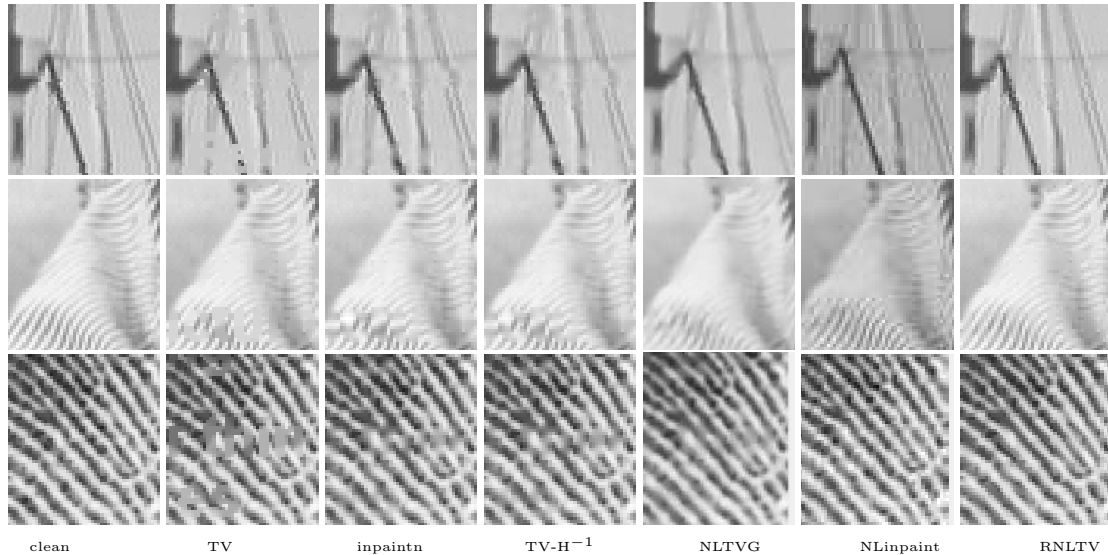


Fig. 12 The zoomed regions of results for the word mask in Figure 11. The first column is the zoomed region of the clean test images; the rest of them are the zoomed region of the restoration results in the fourth, fifth and sixth rows of Figure 11.

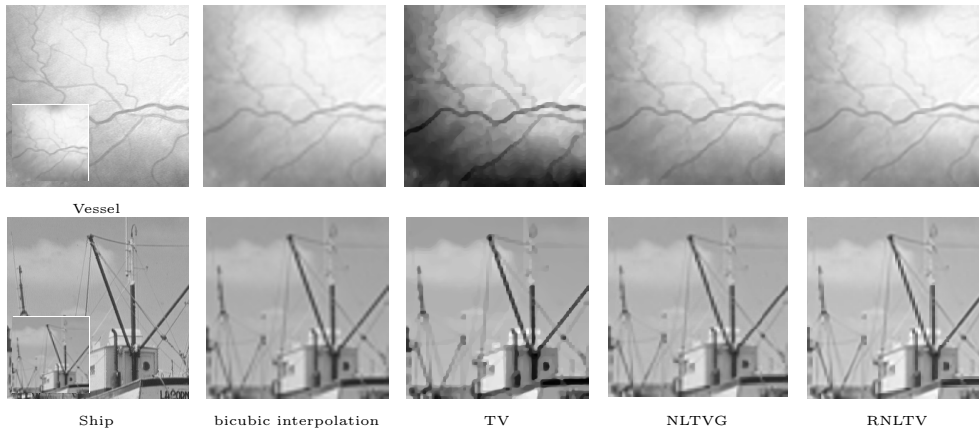


Fig. 13 Image zooming. First column: downsample the original image by a factor of 4.

shortcoming of the TV based restoration, that is, it tends to create piecewise constant zones in the homogenous regions which is the tissue around the vessels and reduces the contrast of thin structures. The PSNR values in Table 4 confirm the observation.

In Table 4, when zooming 'Lacournouaille', although the PSNR value of the result of bicubic is slightly higher, from the second row of Figure 13, one can observe that RNLTV based zooming leads to a better visual quality whereas the bicubic interpolation and NLTVG get more blurry results, see the mast pointing to the one o'clock direction on the right part of Ship.

3.5 Denoising

In this section, we compare the results of RNLTV with the results of BM3D and two standard regularization methods: the TV denoising (ROF [39]) and the NLTV denoising [24]. For the NLTV, we simply set the γ to 0 in our RNLTV code. The NLTV and RNLTV use identical initial weights for the same problem. The initial weights are obtained by applying (5) in which the image I is replaced by the result of the ROF model (i.e. the image presented in the TV column in the same line as the two non-local methods in Figure 14 and 15).

3.5.1 Synthetic Image

We consider additive Gaussian white noise with standard deviation $\sigma = 0.12, 0.14, 0.16$. As mentioned in Table 1, we use the data fidelity term $\chi_{\|u-I\|^2 \leq \tau}(u)$. We set $\gamma = 0.3$, $\tau = |\mathcal{P}|\sigma^2$ and $\mu = 0.6$ in the RNLTV model. The number of iterations⁷ is 600. The initial guess for u is the degraded image. The initial weight values are computed using (50).

In Figure 14, we observe that RNLTV produces sharp edges for all shapes. The contrast is reduced differently in different parts of the thin structures in the ROF results. The black region sometimes 'leak' in the background creating some kind of 'blur-like' artifact. NLTVG and BM3D remove all the noise but at the same time, NLTVG blurs thin curves and BM3D creates artifacts.

3.5.2 Natural Image

We consider additive Gaussian white noise with standard deviation $\sigma = 0.04, 0.06$ and 0.08 (We add more noise to Fingerprint, because it is more noise-resistant than the rest of test images). We lower the noise level because the thin structures in the natural test images are less contrasted. We use the results of TV denoising to initialize RNLTV and we use the data fidelity term $\lambda\|u-I\|^2$. Again, the initial weight values are according to (50). The setting of the parameters in RNLTV and NLTV is presented in Table 5 and 6 respectively. The number of iterations⁸ are both 600. BM3D does not have any input parameters (except the polluted image). The parameter of the TV model is tuned to find the best restoration quality. The results are in Figure 15 and Table 7. Although BM3D results tend to have highest PSNR values, but we argue that RNLTV can better recover thin structures, see the zoomed image results in Figure 16.

In all the experiments displayed in Figure 15, we observe a similar phenomenon. The TV regularization contains staircasing effects and reduces the contrast or erases thin features, see the ropes in Boat. Some gray stripes on the NLTV denoising result become less notable, see the second row, fourth column in Figure 16. NLTVG denoising results have similar problems because of an excessive blur. BM3D puts distracting artifact around ropes, see the first row, fifth column in Figure 16, and

⁷ Again, we just take a very large number of iteration for which we have convergence.

⁸ Again, we just take a very large number of iteration for which we have convergence.

smooths out some of the textures on the scarf, see the second row, fifth column in Figure 16. RNLTV recovers the majority of the thin stripes on the scarf to their original shapes, and obtains slightly clear friction ridges, see the last column in Figure 16.

Table 5 Values of parameters in RNLTV for restoration of the natural images corrupted by white Gaussian noise. μ is fixed to 0.6, σ is the standard deviation of the Gaussian noise.

	$\sigma=0.04$		$\sigma=0.06$		$\sigma=0.08$	
	γ	λ	γ	λ	γ	λ
Boat	0.05	0.5	0.1	0.5	0.2	0.5
Barbara	0.05	0.5	0.1	0.5	0.2	0.5
	$\sigma=0.1$		$\sigma=0.16$		$\sigma=0.2$	
	γ	λ	γ	λ	γ	λ
Fingerprint	0.05	1	0.1	1	0.2	1

Table 6 Values of parameters in NLTV for restoration of the natural images corrupted by white Gaussian noise. μ is fixed to 0.6, σ is the standard deviation of the Gaussian noise.

	$\sigma=0.04$	$\sigma=0.06$	$\sigma=0.08$
	λ	λ	λ
Boat	1	0.5	0.5
Barbara	1	0.5	0.2
	$\sigma=0.1$	$\sigma=0.16$	$\sigma=0.2$
	λ	λ	λ
Fingerprint	0.2	0.1	0.1

Table 7 PSNR values of different methods for restoration of images corrupted by white Gaussian noise. σ is the standard deviation of the Gaussian noise.

	σ	noisy	TV	NLTV	NLTVG	BM3D	RNLTV
Syn	0.12	16.54	21.76	31.91	32.42	30.58	39.06
	0.14	15.20	21.64	27.20	29.31	29.15	34.04
	0.16	14.04	21.49	23.67	27.01	26.23	29.65
Boat	0.04	28.02	30.73	31.47	31.59	35.92	32.25
	0.06	24.50	28.73	29.27	29.56	34.07	30.26
	0.08	22.00	25.97	27.40	28.17	32.60	28.86
Barbara	0.04	28.02	29.58	30.64	31.68	34.15	32.09
	0.06	24.50	26.76	27.68	28.94	32.07	30.26
	0.08	22.00	23.98	25.48	27.32	30.65	28.79
Fingerprint	0.1	20.06	21.59	21.63	22.84	25.21	24.09
	0.16	15.98	19.42	19.58	20.31	23.22	21.41
	0.2	14.04	18.33	18.34	19.28	22.37	19.98
average		20.41	24.17	26.19	27.37	29.69	29.23

4 Conclusion

In this paper, we have studied a RNLTV model to restore the thin structures through the regularization term on weights v , in a NLTV model. In

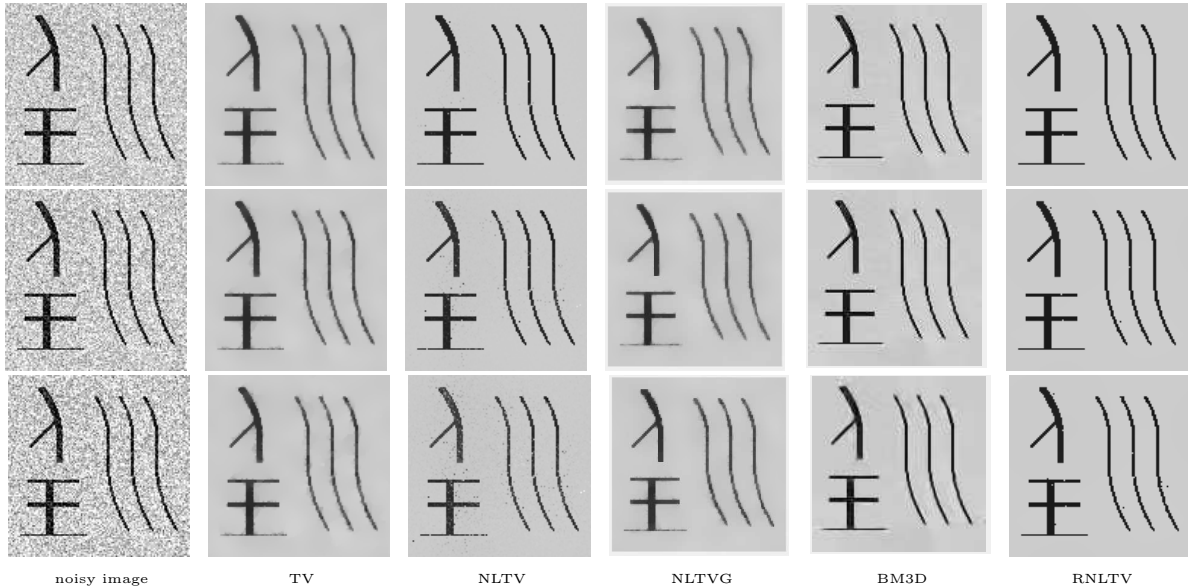


Fig. 14 Denoising results for an additive white Gaussian noise with standard deviation $\sigma = 0.12$ (top row), $\sigma = 0.14$ (middle row) and $\sigma = 0.16$ (bottom row).

order to implement a PALM solver, we compute the gradients and the Lipschitz moduli of the gradient mapping. The performance of the proposed method is demonstrated by numerical experiments on synthetic images and natural images. In particular, we see that the regularization of the weights permit to smooth or interpolate them. This leads to better restoration results.

In the future, we plan to improve the ability of the RNLTv model to recover textures, one possible solution is to use shape-adaptive patches [17]. More precisely, to avoid diminishing thin structures, we choose the shape of the patch based on the local geometry, so that the pixels depicting thin structures in a window will take relatively large portion of the overall, thus will not be overlooked, while patches preserve texture areas. In addition, inspired by the global filtering framework [42], we can use a non-local term to regularize v . Since the model is also better suited to solve inverse problems thus we will extend the model to computerized tomography problems, reconstruction from random measures, etc.

Acknowledgement

François Malgouyres would like to thank Julien Rabin for fruitful discussions on the subject and for teaching him how to efficiently perform the projection on the simplex, and also would like to thank Prof. Gabriel Peyré for providing his code.

5 Appendices

5.1 TV under a max form

We want to prove that for any $u \in \mathbb{R}^{\mathcal{P}}$ and any (fixed) $v \in \mathcal{U}^{\mathcal{P}}$

$$TV(v, u) = \max_{\|w\|_{\infty, 2} \leq 1} \langle \mathbf{D}_v u, w \rangle - \frac{\mu}{2} \|w\|^2, \quad (51)$$

where we remind that, for $w \in \mathbb{R}^{\mathcal{P} \times \mathcal{B}}$, the norm defining the constraint takes the form

$$\|w\|_{\infty, 2} = \max_{p \in \mathcal{P}} \|w_p\|, \text{ for } w_p = (w_{p,q})_{q \in \mathcal{B}} \in \mathbb{R}^{\mathcal{B}}. \quad (52)$$

First, notice that if for all $p \in \mathcal{P}$ we know $w_p^* \in \mathbb{R}^{\mathcal{B}}$ such that

$$w_p^* = \operatorname{argmax}_{w \in \mathbb{R}^{\mathcal{B}}: \|w\| \leq 1} \langle \mathbf{D}_v u_p, w \rangle - \frac{\mu}{2} \|w\|^2, \quad (53)$$

where we denote $\mathbf{D}_v u_p = (\mathbf{D}_v u_{p,q})_{q \in \mathcal{B}} \in \mathbb{R}^{\mathcal{B}}$; we can deduce from the optimality of all its components w_p^* that $w^* = (w_p^*)_{p \in \mathcal{P}} \in \mathbb{R}^{\mathcal{P} \times \mathcal{B}}$ satisfies

$$w^* = \operatorname{argmax}_{w \in \mathbb{R}^{\mathcal{P} \times \mathcal{B}}: \|w\|_{\infty, 2} \leq 1} \langle \mathbf{D}_v u, w \rangle - \frac{\mu}{2} \|w\|^2. \quad (54)$$

In order to calculate w_p^* , for a given $p \in \mathcal{P}$, we first remark that there exists $\alpha_p^* \geq 0$ such that

$$w_p^* = \alpha_p^* \mathbf{D}_v u_p. \quad (55)$$

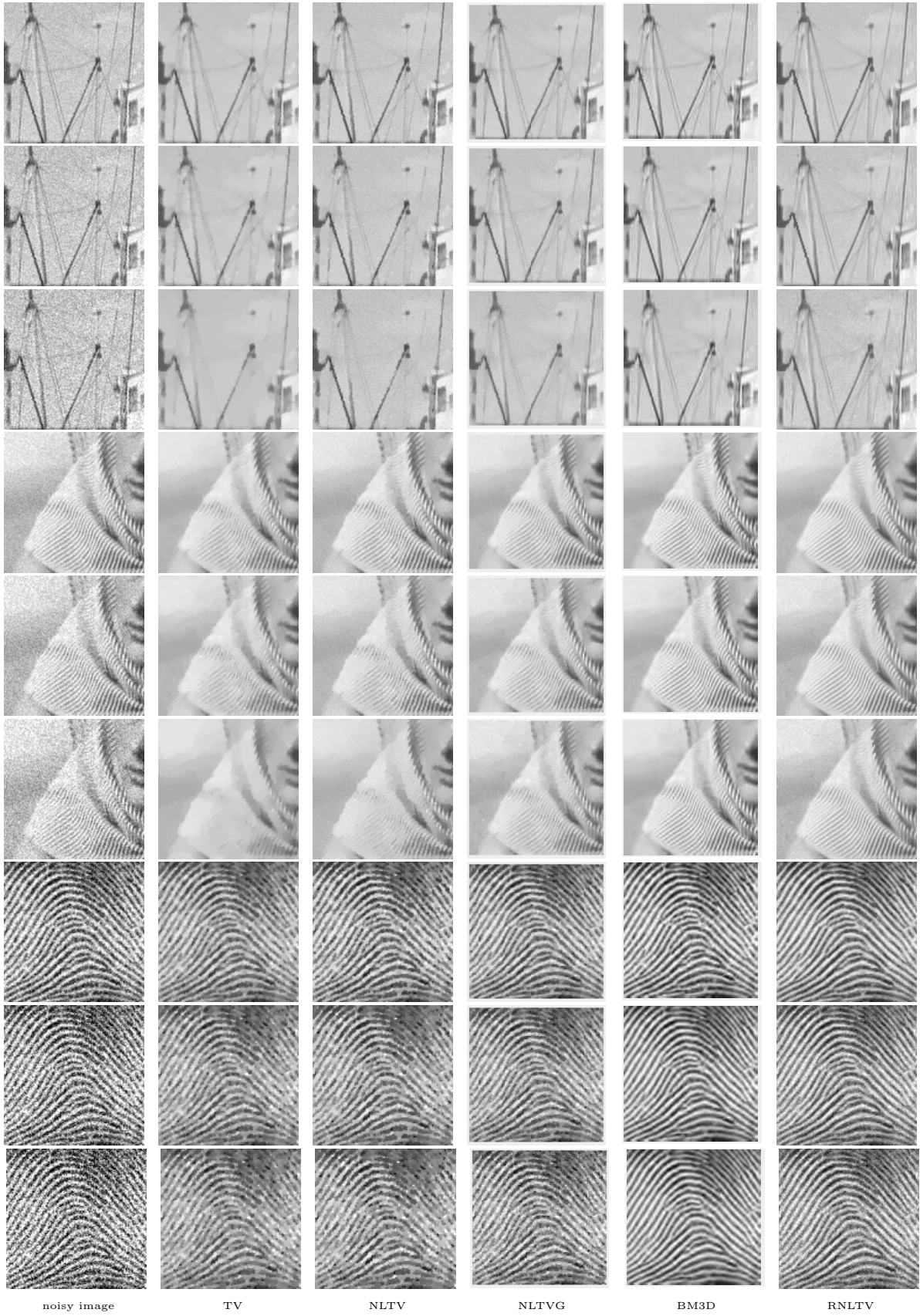


Fig. 15 Denoising results for an additive white Gaussian noise with standard deviation $\sigma = 0.04$ (the first and fourth rows), $\sigma = 0.06$ (the second and fifth rows) and $\sigma = 0.08$ (the third and sixth rows). More Stronger additive white Gaussian noise for Fingerprint with standard deviation $\sigma = 0.1$ (the seventh row), $\sigma = 0.16$ (the eighth row) and $\sigma = 0.2$ (the ninth row).

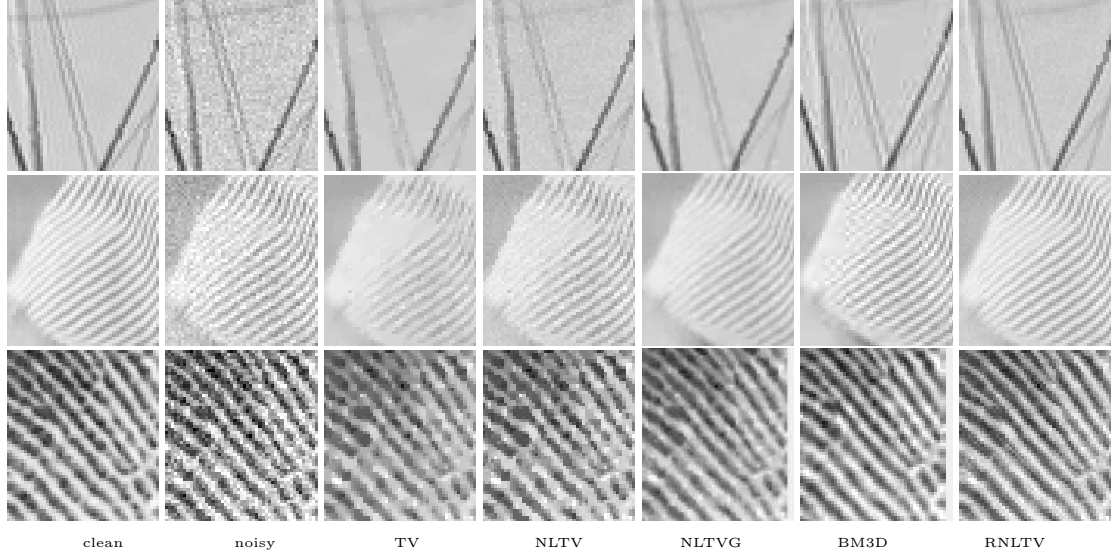


Fig. 16 The zoomed regions of results in Figure 15. The first row is the zoomed regions in the first row of Figure 15; the second row is the zoomed regions in the fourth row of Figure 15; the last row is the zoomed regions in the seventh row of Figure 15.

Then, if we use this expression in (53) we have that

$$\begin{aligned} \alpha_p^* &= \operatorname{argmax}_{\alpha \|\mathbf{D}_v u_p\| \leq 1} \alpha \|\mathbf{D}_v u_p\|^2 - \frac{\mu}{2} \alpha^2 \|\mathbf{D}_v u_p\|^2, \\ &= \operatorname{argmax}_{\alpha \|\mathbf{D}_v u_p\| \leq 1} \alpha \left(1 - \frac{\mu \alpha}{2}\right), \\ &= \begin{cases} \frac{1}{\|\mathbf{D}_v u_p\|}, & \text{if } \frac{1}{\mu} > \frac{1}{\|\mathbf{D}_v u_p\|}, \\ \frac{1}{\mu}, & \text{otherwise.} \end{cases} \end{aligned}$$

We finally obtain that

$$w_p^* = \begin{cases} \frac{\mathbf{D}_v u_p}{\|\mathbf{D}_v u_p\|}, & \text{if } \|(\mathbf{D}_v u)_p\| \leq \mu, \\ \frac{\mu \mathbf{D}_v u_p}{\|(\mathbf{D}_v u)_p\|}, & \text{otherwise.} \end{cases} \quad (56)$$

This corresponds to the expression of $w^*(u)_p$ in Proposition 1.

If we now use the expression for w_p^* to calculate the objective function, we find that

$$\begin{aligned} &\max_{w \in \mathbb{R}^B: \|w\| \leq 1} \langle \mathbf{D}_v u_p, w \rangle - \frac{\mu}{2} \|w\|^2 \\ &= \langle \mathbf{D}_v u_p, w_p^* \rangle - \frac{\mu}{2} \|w_p^*\|^2 \\ &= \begin{cases} \frac{\|\mathbf{D}_v u_p\|^2}{2\mu}, & \text{if } \|\mathbf{D}_v u_p\| \leq \mu, \\ \|\mathbf{D}_v u_p\| - \frac{\mu}{2}, & \text{otherwise.} \end{cases} \\ &= \Psi_\mu(\|\mathbf{D}_v u_p\|). \end{aligned} \quad (57)$$

As a consequence,

$$\begin{aligned} &\max_{\|w\|_{\infty, 2} \leq 1} \langle \mathbf{D}_v u, w \rangle - \frac{\mu}{2} \|w\|^2 \\ &= \sum_{p \in \mathcal{P}} \langle \mathbf{D}_v u_p, w_p^* \rangle - \frac{\mu}{2} \|w_p^*\|^2 \\ &= TV(v, u). \end{aligned} \quad (58)$$

5.2 Proof of Proposition 1

Let us first remind and adapt to the context of this paper the theorem 1 stated in [33]. This result considers finite dimensional real vector spaces V_1 and V_2 , a linear operator $A : V_1 \rightarrow V_2$, a parameter $\mu > 0$ and a function

$$f_\mu(x) = \max_{u \in Q_2} \langle Ax, u \rangle - \frac{\mu}{2} \|u\|^2, \quad \forall x \in V_1 \quad (59)$$

where $Q_2 \subset V_2$ is a closed convex bounded set.

It is stated and proved in [33] that the function f_μ is continuously differentiable at any $x \in V_1$. Moreover, if we denote $u_\mu(x)$ the unique solution of the maximization problem defining f_μ , we have:

$$\nabla f_\mu(x) = A^* u_\mu(x), \quad (60)$$

where A^* is the adjoint of A . Moreover, $x \mapsto f_\mu(x)$ is Lipschitz continuous with constant

$$\frac{\|A\|_{V_1 \rightarrow V_2}}{\mu}, \quad (61)$$

where

$$\|A\|_{V_1 \rightarrow V_2} = \max_{\|x\| \leq 1} \|Ax\|. \quad (62)$$

The Proposition 1 is a straightforward application of this statement to the function

$$TV(v, u) = \max_{\|w\|_{\infty, 2} \leq 1} \langle \mathbf{D}_v u, w \rangle - \frac{\mu}{2} \|w\|^2. \quad (63)$$

In order to compute the Lipschitz constant, we need however to find a bound for $\|\mathbf{D}_v\|_{\mathbb{R}^{\mathcal{P}} \rightarrow \mathbb{R}^{\mathcal{P}} \times \mathcal{B}}$.

In order to compute it, we consider $u \in \mathbb{R}^{\mathcal{P}}$. We have for any $v \in \mathcal{U}^{\mathcal{P}}$

$$\begin{aligned} \|\mathbf{D}_v u\|^2 &= \sum_{p \in \mathcal{P}} \sum_{q \in \mathcal{B}} v_q^p (u_p - u_{p+q})^2 \\ &\leq 2 \sum_{p \in \mathcal{P}} \sum_{q \in \mathcal{B}} v_q^p (u_p^2 + u_{p+q}^2) \\ &\leq 2 \sum_{p \in \mathcal{P}} u_p^2 + 2 \sum_{q \in \mathcal{B}} \sum_{p \in \mathcal{P}} u_{p+q}^2 \\ &\leq 2\|u\|^2 + 2|\mathcal{B}|\|u\|^2, \end{aligned} \quad (64)$$

where $|\mathcal{B}|$ denotes the cardinality of \mathcal{B} . We then deduce⁹ that for any $v \in \mathcal{U}^{\mathcal{P}}$

$$\|\mathbf{D}_v\|_{\mathbb{R}^{\mathcal{P}} \rightarrow \mathbb{R}^{\mathcal{P} \times \mathcal{B}}} \leq \sqrt{2} \sqrt{|\mathcal{B}| + 1}. \quad (65)$$

Finally, it is standard that (32) is a consequence of the fact that $l^\nu = \frac{\sqrt{2} \sqrt{|\mathcal{B}| + 1}}{\mu}$ is a Lipschitz bound for $u \mapsto \nabla_u TV(v, u)$.

5.3 Proof of Proposition 2

Considering $v \in \mathcal{U}^{\mathcal{P}}$, $u \in \mathbb{R}^{\mathcal{P}}$ and a small variation $h \in \mathbb{R}^{\mathcal{P} \times \mathcal{B}}$, we denote $\mathcal{P}_1 = \{p \in \mathcal{P} | (\mathbf{A}_u v)_p \geq \frac{\mu}{2}\}$ and $\mathcal{P}_2 = \mathcal{P} \setminus \mathcal{P}_1$. For h small enough, we have

$$\begin{aligned} TV(v+h, u) &= \sum_{p \in \mathcal{P}_1} \Psi_\mu \left(\sqrt{(\mathbf{A}_u v)_p + (\mathbf{A}_u h)_p} \right) \\ &\quad + \sum_{p \in \mathcal{P}_2} \frac{(\mathbf{A}_u v)_p + (\mathbf{A}_u h)_p}{2\mu}. \end{aligned} \quad (66)$$

Moreover,

$$\begin{aligned} &\sum_{p \in \mathcal{P}_1} \Psi_\mu \left(\sqrt{(\mathbf{A}_u v)_p + (\mathbf{A}_u h)_p} \right) \\ &= \sum_{p \in \mathcal{P}_1} \Psi_\mu \left(\sqrt{(\mathbf{A}_u v)_p} + \frac{(\mathbf{A}_u h)_p}{2\sqrt{(\mathbf{A}_u v)_p}} + o(|(\mathbf{A}_u h)_p|) \right) \\ &= \sum_{p \in \mathcal{P}_1} \Psi_\mu \left(\sqrt{(\mathbf{A}_u v)_p} \right) \\ &\quad + \Psi'_\mu \left(\sqrt{(\mathbf{A}_u v)_p} \right) \frac{(\mathbf{A}_u h)_p}{2\sqrt{(\mathbf{A}_u v)_p}} + o(|(\mathbf{A}_u h)_p|) \end{aligned} \quad (67)$$

Denoting for all $p \in \mathcal{P}_1$

$$(u^*(v))_p = \frac{\Psi'_\mu \left(\sqrt{(\mathbf{A}_u v)_p} \right)}{2\sqrt{(\mathbf{A}_u v)_p}} \quad (68)$$

⁹ Notice that a similar upper bound is provided in [10] for the usual finite differences.

and for all $p \in \mathcal{P}_2$

$$(u^*(v))_p = \frac{1}{2\mu} \quad (69)$$

and using the simple closed form expression for the derivative Ψ'_μ , we get for all $p \in \mathcal{P}$

$$(u^*(v))_p = \begin{cases} \frac{1}{2\sqrt{(\mathbf{A}_u v)_p}}, & \text{if } \sqrt{(\mathbf{A}_u v)_p} \geq \mu, \\ \frac{1}{2\mu}, & \text{if } \mu \geq \sqrt{(\mathbf{A}_u v)_p} \geq 0. \end{cases} \quad (70)$$

Using this notation in the previous calculations we obtain that

$$TV(v+h, u) = TV(v, u) + \langle u^*(v), \mathbf{A}_u h \rangle + o(\|h\|). \quad (71)$$

We finally conclude that

$$\nabla_v TV(v, u) = \mathbf{A}_u^* u^*(v), \quad (72)$$

This proves the first part of Proposition 2.

Let us now show that $v \mapsto TV(v, u)$ is concave over $\mathbb{R}_+^{\mathcal{P} \times \mathcal{B}}$. In order to do so, we rewrite the latter formula under the form $(u^*(v))_p = \varphi((\mathbf{A}_u v)_p)$ where the function φ is defined for all $t \geq 0$ by

$$\varphi(t) = \begin{cases} \frac{1}{2\sqrt{t}}, & \text{if } \sqrt{t} \geq \mu, \\ \frac{1}{2\mu}, & \text{if } \mu \geq \sqrt{t} \geq 0. \end{cases} \quad (73)$$

Notice that the function φ is non-increasing and therefore

$$(\varphi(t_1) - \varphi(t_2))(t_1 - t_2) \leq 0, \quad \forall (t_1, t_2) \in \mathbb{R}^2. \quad (74)$$

We now consider v and $v' \in \mathbb{R}_+^{\mathcal{P} \times \mathcal{B}}$ and $u \in \mathbb{R}^{\mathcal{P}}$. Using Taylor's Theorem, we know there exists $t \in [0, 1]$ such that $v'' = tv' + (1-t)v$ satisfies

$$\begin{aligned} &TV(v', u) - TV(v, u) - \langle \nabla_v TV(v, u), v' - v \rangle \\ &= \langle \nabla_v TV(v'', u) - \nabla_v TV(v, u), v' - v \rangle \\ &= \langle u^*(v'') - u^*(v), \mathbf{A}_u (v' - v) \rangle \\ &= \sum_{p \in \mathcal{P}} (\varphi(\mathbf{A}_u v''_p) - \varphi(\mathbf{A}_u v_p)) (\mathbf{A}_u v'_p - \mathbf{A}_u v_p). \end{aligned} \quad (75)$$

However, for any $p \in \mathcal{P}$, $\mathbf{A}_u v''_p - \mathbf{A}_u v_p = t(\mathbf{A}_u v'_p - \mathbf{A}_u v_p)$ and $\mathbf{A}_u v''_p - \mathbf{A}_u v_p$ and $\mathbf{A}_u v'_p - \mathbf{A}_u v_p$ have the same sign. Using (74), we then get

$$(\varphi(\mathbf{A}_u v''_p) - \varphi(\mathbf{A}_u v_p)) (\mathbf{A}_u v'_p - \mathbf{A}_u v_p) \leq 0 \quad (76)$$

and finally

$$TV(v', u) - TV(v, u) - \langle \nabla_v TV(v, u), v' - v \rangle \leq 0. \quad (77)$$

This concludes the proof.

5.4 Proof of Proposition 3

For any $v \in \mathcal{U}^{\mathcal{P}}$, given the expression

$$R(v) = \gamma \|Bv\|^2 \quad (78)$$

we immediately have

$$\nabla R(v) = 2\gamma B^* Bv. \quad (79)$$

We only need to calculate the Lipschitz constant l' provided in Proposition 3. In order to do so, we consider v and $v' \in \mathcal{U}^{\mathcal{P}}$ and denote $w = v' - v$. We have

$$\|\nabla R(v') - \nabla R(v)\|^2 = 4\gamma^2 \|B^* Bw\|^2. \quad (80)$$

Moreover, using the formula for B^* , we get

$$\begin{aligned} \|B^* Bw\|^2 &= \\ & \sum_{(p,q) \in \mathcal{P} \times \mathcal{B}} \left| \sum_{p' \in \mathcal{N}} ((Bw)_{p,q,p'} - (Bw)_{p-p',q,p'}) \right|^2. \end{aligned} \quad (81)$$

The term inside the absolute value can be rewritten, using the definition of B , under the form

$$\begin{aligned} & \sum_{p' \in \mathcal{N}} ((Bw)_{p,q,p'} - (Bw)_{p-p',q,p'}) \\ &= \sum_{p' \in \mathcal{N}} (2w_{p,q} - w_{p+p',q} - w_{p-p',q}), \quad (82) \\ &= 2|\mathcal{N}|w_{p,q} - \sum_{p' \in \mathcal{N}} w_{p+p',q} - \sum_{p' \in \mathcal{N}} w_{p-p',q}. \end{aligned}$$

Therefore,

$$\begin{aligned} & \|B^* Bw\|^2 \\ & \leq \sum_{(p,q) \in \mathcal{P} \times \mathcal{B}} 3 \left(4|\mathcal{N}|^2 w_{p,q}^2 + \left(\sum_{p' \in \mathcal{N}} w_{p+p',q} \right)^2 \right. \\ & \quad \left. + \left(\sum_{p' \in \mathcal{N}} w_{p-p',q} \right)^2 \right) \\ & \leq 3 \left(4|\mathcal{N}|^2 \|w\|^2 + |\mathcal{N}| \sum_{(p,q) \in \mathcal{P} \times \mathcal{B}} \sum_{p' \in \mathcal{N}} w_{p+p',q}^2 \right. \\ & \quad \left. + |\mathcal{N}| \sum_{(p,q) \in \mathcal{P} \times \mathcal{B}} \sum_{p' \in \mathcal{N}} w_{p-p',q}^2 \right) \\ & \leq 3 \times 6|\mathcal{N}|^2 \|w\|^2. \end{aligned} \quad (83)$$

Finally,

$$\|\nabla R(v') - \nabla R(v)\|^2 \leq 2 \times 6^2 \gamma^2 |\mathcal{N}|^2 \|v' - v\|^2. \quad (84)$$

We then conclude that $v \mapsto \nabla R(v)$ is Lipschitz with Lipschitz constant $6\sqrt{2}\gamma|\mathcal{N}|$. This concludes the proof.

References

1. Arias, P., Caselles, V., Sapiro, G.: A variational framework for non-local image inpainting. In: *Energy Minimization Methods in Computer Vision and Pattern Recognition*, pp. 345–358. Springer (2009)
2. Arias, P., Facciolo, G., Caselles, V., Sapiro, G.: A variational framework for exemplar-based image inpainting. *Int. J. Comput. Vis.* **93**(3), 319–347 (2011)
3. Aujol, J.F., Dossal, C.: Stability of over-relaxations for the Forward-Backward algorithm, application to FISTA. *SIAM J. Optim.* **25**(4), 2408–2433 (2015)
4. Aujol, J.F., Ladjal, S., Masnou, S.: Exemplar-based inpainting from a variational point of view. *SIAM J. Math. Anal.* **42**(3), 1246–1285 (2010)
5. Ballester, C., Bertalmio, M., Caselles, V., Sapiro, G., Verdera, J.: Filling-in by joint interpolation of vector fields and gray levels. *IEEE Trans. Image Process.* **10**(8), 1200–1211 (2001)
6. Bauschke, H.H., Burachik, R., Combettes, P.L., Elser, V., Luke, D.R., Wolkowicz, H.: *Fixed-point algorithms for inverse problems in science and engineering*, vol. 49. Springer Science & Business Media (2011)
7. Bolte, J., Sabach, S., Teboulle, M.: Proximal alternating linearized minimization for nonconvex and nonsmooth problems. *Math. Program., Serie A* **146**(1-2), 459–494 (2014)
8. Buades, A., Coll, B., Morel, J.M.: A review of image denoising algorithms, with a new one. *Multiscale Model. Simul.* **4**(2), 490–530 (2005)
9. Burger, M., He, L., Schönlieb, C.B.: Cahn-hilliard inpainting and a generalization for grayvalue images. *SIAM J. Imaging Sci.* **2**(4), 1129–1167 (2009)
10. Chambolle, A.: An algorithm for total variation minimization and applications. *J. Math. Imaging Vision* **20**(1-2), 89–97 (2004)
11. Chan, T.F., Shen, J.: Nontexture inpainting by curvature-driven diffusions. *J. Vis. Comm. Image Represent.* **12**(4), 436–449 (2001)
12. Chen, A., Bertozzi, A.L., Ashby, P.D., Getreuer, P., Lou, Y.: Enhancement and recovery in atomic force microscopy images. In: *Excursions in Harmonic Analysis, Volume 2*, pp. 311–332. Springer (2013)
13. Chen, C., Chan, R.H., Ma, S., Yang, J.: Inertial proximal admm for linearly constrained separable convex optimization. *SIAM J. Imaging Sci.* **8**(4), 2239–2267 (2015)
14. Cho, M., Mishra, K.V., Cai, J.F., Xu, W.: Block iterative reweighted algorithms for super-resolution of spectrally sparse signals. *IEEE Signal Process. Lett.* **22**(12), 2319–2313 (2015)
15. Chouzenoux, E., Pesquet, J.C., Repetti, A.: A block coordinate variable metric forward-backward algorithm. *Journal of Global Optimization* pp. 1–29 (2016)
16. Condat, L.: Fast projection onto the simplex and the ℓ_1 ball. *Math. Program.* **158**(1), 575–585 (2016)

17. Deledalle, C.A., Duval, V., Salmon, J.: Non-local methods with shape-adaptive patches (NLM-SAP). *J. Math. Imaging Vision* **43**(2), 103–120 (2012)
18. Duval, V., Aujol, J.F., Gousseau, Y.: On the parameter choice for the non-local means. *CMLA Preprint* (2010)
19. Elmoataz, A., Lezoray, O., Bogleux, S.: Nonlocal discrete regularization on weighted graphs: a framework for image and manifold processing. *IEEE Trans. Image Process.* **17**(7), 1047–1060 (2008)
20. Facciolo, G., Arias, P., Caselles, V., Sapiro, G.: Exemplar-based interpolation of sparsely sampled images. In: *Energy Minimization Methods in Computer Vision and Pattern Recognition*, pp. 331–344. Springer (2009)
21. Fedorov, V., Facciolo, G., Arias, P.: Variational framework for non-local inpainting. *Image Processing On Line* **5**, 362–386 (2015)
22. Garcia, D.: Robust smoothing of gridded data in one and higher dimensions with missing values. *Comput. Statist. Data Anal.* **54**(4), 1167–1178 (2010)
23. Gilboa, G., Osher, S.: Nonlocal linear image regularization and supervised segmentation. *Multiscale Modeling & Simulation* **6**(2), 595–630 (2007)
24. Gilboa, G., Osher, S.: Nonlocal operators with applications to image processing. *Multiscale Model. Simul.* **7**(3), 1005–1028 (2008)
25. Jung, M., Vese, L.A.: Nonlocal variational image deblurring models in the presence of gaussian or impulse noise. In: *Scale Space and Variational Methods in Computer Vision*, pp. 401–412. Springer (2009)
26. Kanizsa, G.: *Organization in vision: Essays on Gestalt perception*. Praeger Publishers (1979)
27. Kheradmand, A., Milanfar, P.: A general framework for regularized, similarity-based image restoration. *IEEE Trans. Image Process.* **23**(12), 5136–5151 (2014)
28. Lebrun, M., Buades, A., Morel, J.M.: Implementation of the "Non-Local Bayes" (NL-Bayes) image denoising algorithm. *Image Processing On Line* **3**, 1–42 (2013)
29. Lebrun, M., Buades, A., Morel, J.M.: A nonlocal Bayesian image denoising algorithm. *SIAM J. Imaging Sci.* **6**(3), 1665–1688 (2013)
30. Levina, E., Bickel, P.J.: Texture synthesis and nonparametric resampling of random fields. *Ann. Statist.* pp. 1751–1773 (2006)
31. Lou, Y., Zhang, X., Osher, S., Bertozzi, A.: Image recovery via nonlocal operators. *J. Sci. Comput.* **42**(2), 185–197 (2010)
32. Mumford, D., Nitzberg, M., Shiota, T.: Filtering, segmentation and depth. *Lecture Notes in Comput. Sci.* **662** (1993)
33. Nesterov, Y.: Smooth minimization of non-smooth functions. *Math. Program., Serie A* **103**(1), 127–152 (2005)
34. Ouyang, Y., Chen, Y., Lan, G., Pasiliao Jr, E.: An accelerated linearized alternating direction method of multipliers. *SIAM J. Imaging Sci.* **8**(1), 644–681 (2015)
35. Peyré, G.: Sparse modeling of textures. *J. Math. Imaging Vision* **34**(1), 17–31 (2009)
36. Peyré, G., Bogleux, S., Cohen, L.: Non-local regularization of inverse problems. In: *The 10th European Conference on Computer Vision*, pp. 57–68. Springer (2008)
37. Peyré, G., Bogleux, S., Cohen, L.D.: Non-local regularization of inverse problems. *Inverse Probl. Imaging* **5**(2), 511–530 (2011)
38. Pizarro, L., Mrázek, P., Didas, S., Grewenig, S., Weickert, J.: Generalised nonlocal image smoothing. *Int. J. Comput. Vis.* **90**(1), 62–87 (2010)
39. Rudin, L.I., Osher, S., Fatemi, E.: Nonlinear total variation based noise removal algorithms. *Phys. D* **60**(1), 259–268 (1992)
40. Schönlieb, C.b., Bertozzi, A.: Unconditionally stable schemes for higher order inpainting. *Commun. Math. Sci.* **9**(2), 413–457 (2011)
41. Shen, J., Chan, T.F.: Mathematical models for local nontexture inpaintings. *SIAM J. Appl. Math.* **62**(3), 1019–1043 (2002)
42. Talebi, H., Milanfar, P.: Global image denoising. *IEEE Trans. Image Process.* **23**(2), 755–768 (2014)
43. Wang, G., Garcia, D., Liu, Y., De Jeu, R., Dolman, A.J.: A three-dimensional gap filling method for large geophysical datasets: Application to global satellite soil moisture observations. *Environ. Modell. Softw.* **30**, 139–142 (2012)
44. Wei, L.Y., Levoy, M.: Fast texture synthesis using tree-structured vector quantization. In: *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pp. 479–488. ACM Press/Addison-Wesley Publishing Co. (2000)
45. Weiss, P., Blanc-Feraud, L., Aubert, G.: Efficient schemes for total variation minimization under constraints in image processing. *SIAM J. Sci. Comput.* **31**(3), 2047–2080 (2009)
46. Weissman, T., Ordentlich, E., Seroussi, G., Verdú, S., Weinberger, M.J.: Universal discrete denoising: Known channel. *IEEE Trans. Inf. Theory* **51**(1), 5–28 (2005)
47. Yaroslavsky, L.P., Yaroslavskij, L.: *Digital picture processing: An introduction*. Springer-Verlag **1** (1985)
48. Zhang, X., Burger, M., Bresson, X., Osher, S.: Bregmanized nonlocal regularization for deconvolution and sparse reconstruction. *SIAM J. Imaging Sci.* **3**(3), 253–276 (2010)