



# Adaptive regularization of the NL-means: Application to image and video denoising

Camille Sutour, Jean-François Aujol, Charles-Alban Deledalle

## ► To cite this version:

Camille Sutour, Jean-François Aujol, Charles-Alban Deledalle. Adaptive regularization of the NL-means: Application to image and video denoising. 2013. hal-00854830v1

**HAL Id: hal-00854830**

**<https://hal.science/hal-00854830v1>**

Preprint submitted on 28 Aug 2013 (v1), last revised 2 Jun 2014 (v3)

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Adaptive regularization of the NL-means: Application to image and video denoising

Camille Sutour, Jean-François Aujol, and Charles-Alban Deledalle,

**Abstract**—Image denoising is a central problem in image processing and it is often a necessary step prior to higher level analysis such as segmentation, reconstruction or super-resolution. The non-local means (NL-means) perform denoising by exploiting the natural redundancy of patterns inside an image; they perform a weighted average of pixels whose neighborhoods (patches) are close to each other. This allows to reduce significantly the noise while preserving most of the image content. While it performs well on flat areas and textures, it might leave a residual noise around edges and singular structures. We introduce in this paper a variational approach that reduces this residual noise by adaptively regularizing non-local methods with the total variation. The classical minimization of the total variation – the ROF model – leads to restore regular images but it is prone to over-smooth textures and create a staircasing effect on flat areas. The proposed regularized NL-means algorithm combines these methods and reduce both of their respective defaults by minimizing the total variation with a non-local data fidelity term. Besides, this model adapts to different noise statistics and a fast solution can be obtained in the general case of the exponential family. We develop this model for image denoising and then we adapt it to video denoising, where it enables us to use 3D patches.

**Index Terms**—Non-local means, total variation regularization, image and video denoising, adaptive filtering

## I. INTRODUCTION

IMAGE denoising is a central problem in image processing. It is often a necessary step prior to higher level analysis such as segmentation, reconstruction or super-resolution. The goal of the denoising process is to recover a high quality image from a degraded version. Among the main denoising techniques, variational methods minimize an energy that constrains solutions to be regular. One of the most famous variational models used for image denoising is the ROF model [1] that minimizes the total variation (TV) of the image, hence pushing the image towards a piecewise constant solution. This method is quite adapted to denoising while preserving edges, but it presents two major drawbacks: the textures tend to be overly smoothed, and the flat areas are approximated by a piecewise constant surface, resulting in a *staircasing effect*.

Among state-of-the-art denoising methods, the non-local means (NL-means) [2] algorithm performs spatial filtering by using spatial redundancy occurring in an image. Indeed, instead of averaging pixels that are spatially close to each other, it compares patches extracted around each pixel to

perform a weighted average of pixels whose surroundings are close. It has led to many non-local methods and improvements such as [3] or [4]. This technique shows good performance on smooth areas and repetitive textures for which the redundancy is high, but on edges and singular structures it might fail to find enough similar patches and thus performs insufficient denoising. This is referred to as the *rare patch effect*, that has been studied for instance in [5] and [6].

Variational and non-local methods have been put together by Gilboa and Osher in [7] in order to deal separately with smooth and textured areas. They perform a non-local regularization by defining a non-local gradient that allows them to smooth the flat areas while preserving fine structures, resulting in a reduction of the *staircasing effect*. Kindermann et al. [8] have also introduced a non-local regularization term based on the similarity of patches. Non-local regularization has been adapted to many inverse problems, for example in [9] for inpainting or compressive sensing or in [10] for deconvolution problems.

Louchet and Moisan [5] have also proposed to combine the NL-means with TV regularization in their TV-means algorithm. They adapt the total variation in order to turn it into a local filter, and they perform local TV regularization where the NL-means suffers from the *rare patch effect*. They manage to reduce both the *staircasing effect* and the *rare patch effect* with an iterative scheme.

The goal of this paper is also to combine TV minimization with the NL-means, but with a non-local data fidelity term instead of a non-local regularization term as studied above. Protter *et al.* [11] and d’Angelo and Vanderghyest [12] have used the same type of approach in the context of super-resolution. Their technique could be adapted to a denoising context. They use a non-local data-fidelity term based on the normalized weights from the NL-means algorithm. This sums up to computing the solution of the NL-means, then applying TV regularization. In this article, contrary to [11] and [12], we exploit weights that are not normalized. This allows us to use the weights to ponder the TV regularization, hence creating an adaptive model. Our main contribution is to show that the values of the weights reflect the level of denoising performed by the NL-means: on constant areas the NL-means perform efficient denoising free of the *staircasing effect* while on edges adaptive regularization will correct the *rare patch effect*. Besides, we propose a model that adapts to different noise models and we derive a simple resolution scheme in the general case of the exponential family, often encountered in imaging problems.

We have then extended this model to video denoising.

C. Sutour is with IMB and LaBRI, Universit Bordeaux 1, Talence, France, e-mail: camille.sutour@math.u-bordeaux1.fr

J.-F. Aujol and C.-A. Deledalle are with IMB, CNRS – Universit Bordeaux 1, Talence, France, e-mail: {jean-francois.aujol, charles-alban.deledalle}@math.u-bordeaux1.fr

Indeed, Buades *et al.* have shown in [13] that the NL-means allow to denoise image sequences without applying motion compensation. In fact, they even show that it is harmful to do so instead of using all possible candidates in all frames. Besides, in order to guarantee temporal coherence between frames, we use 3D patches, as proposed in [14]. Instead of comparing square windows computed in a single frame in a spatio-temporal neighborhood, we compute 3D patches that take into account temporal consistency in the computing of weights. This selects less (if better) candidates, hence enhancing the *rare patch effect*, that is then corrected thanks to the adaptive spatial TV regularization. Hence, we have adapted our proposed model to video denoising by computing spatio-temporal NL-means combined with spatial TV regularization.

The goal of this paper is to combine the NL-means with variational methods in order to reduce their weaknesses while using their strengths. We thus propose a regularized version of the NL-means algorithm. The main contributions in this paper are the model and its intuitive interpretation, its extension to the exponential family that offers a general model robust to different noise statistics, and the extended framework to video denoising without prior motion compensation. In particular, we can use three-dimensional patches in the NL-means for video denoising. They are responsible for an increased *rare patch effect* that is corrected thanks to the TV regularization, while providing a better temporal consistency.

The organization of the paper is as follows: in Sections II and III we remind the reader of the principle of the NL-means and the ROF model. We give details as to how to solve these problems and how to adapt them to different noise statistics. Section V presents different approaches that combine the NL-means with variational methods in a denoising framework. In Section IV and VI, we present our R-NL model, its generalization and implementation. Finally, Section VII extends the model to video denoising and Section VIII presents results and compares them to state-of-the-art methods.

## II. NL-MEANS

The general problem in denoising is to recover the image  $f \in \mathbb{R}^N$ ,  $N$  being the number of pixels in the image, based on the noisy observation  $g \in \mathbb{R}^N$ . The usual model is the case of additive white Gaussian noise:

$$g = f + \epsilon \quad (1)$$

where  $f$  is the true (unknown) image and  $\epsilon$  is a realization of Gaussian white noise of zero-mean and standard deviation  $\sigma$ . One of the most recent popular denoising methods is the NL-means algorithm described by Buades *et al.* in [2]. It is based on the natural redundancy of the image structures, not just locally but in the whole image. Instead of averaging pixels that are spatially close, the NL-means algorithm compares patches, ie small windows extracted around each pixel, in order to average pixels whose surroundings are similar. Weights are computed in order to reflect how much two noisy pixels are likely to represent the same true gray level, then pixels are averaged according to these weights. For each pixel  $i \in \Omega$ ,

where  $\Omega$  is the domain of the image, the solution of the NL-means is given by the following weighted average:

$$(u_{NL})_i = \frac{1}{Z_i} \sum_{j \in \Omega} w_{i,j} g_j \quad \text{where} \quad Z_i = \sum_{j \in \Omega} w_{i,j}, \quad (2)$$

which is equivalent to the solution of the following minimization problem:

$$u_{NL} = \operatorname{argmin}_{u \in \mathbb{R}^N} \sum_{i \in \Omega} \sum_{j \in \Omega} w_{i,j} (g_j - u_i)^2 \quad (3)$$

where in both cases the weights  $w_{i,j} \in [0, 1]$  are computed in order to select the pixels  $j$  whose surrounding patches are similar to the one extracted around the pixel of interest  $i$ :

$$w_{i,j} = \varphi \left( -\frac{d(g(P_i), g(P_j))}{2|P|h^2} \right) \quad (4)$$

where  $\varphi$  is a kernel decay function,  $d$  a distance function that measures the similarity between the two patches  $P_i$  and  $P_j$  extracted around the pixels  $i$  and  $j$ ,  $|P|$  the size of the patches, and  $h$  a filtering parameter. In [2],  $\varphi$  is a negative exponential and the distance function is an Euclidean norm convolved by a Gaussian kernel of bandwidth  $a$ , with  $a$  controlling the influence of the pixels around the central one. It has been shown that using a flat or trapezoidal kernel achieves lower computational costs with similar performances [15].

*Adaptation to other noise statistics:* the weighted average can be replaced for other (uncorrelated) noise statistics by the the weighted maximum likelihood estimate [16], [17]:

$$u_{NL} = \operatorname{argmin}_{u \in \mathbb{R}^N} - \sum_{i \in \Omega} \sum_{j \in \Omega} w_{i,j} \log p(g_j | u_i) \quad (5)$$

and the distance between two noisy patches  $g_1$  and  $g_2$  can be adapted using the following likelihood ratio [18]:

$$d(g_1, g_2) = -\log \frac{\sup_{\theta} p(g_1 | \theta_1 = \theta) p(g_2 | \theta_2 = \theta)}{\sup_{\theta} p(g_1 | \theta_1 = \theta) \sup_{\theta} p(g_2 | \theta_2 = \theta)}. \quad (6)$$

where  $\theta_1$  and  $\theta_2$  refers to the underlying noise-free patches.

*Discussion:* non-local methods achieve overall good performances but they suffer from two opposite drawbacks. The algorithm might select too many irrelevant candidates, resulting in the *patch jittering effect*: structures in these areas are overly smoothed due to the combination of candidates with different underlying values. On the contrary, around singular structures or edges, it can be difficult to find enough similar patches so the pixels are not properly denoised, resulting in a *rare patch effect*. These two problems oppose each other; they are controlled by the filtering parameter  $h$  which acts as a bias-variance trade-off, as interpreted in [19].

## III. VARIATIONAL METHODS

The variational methods consist in looking for an image that minimizes a given energy in order to fit the data while respecting some smoothness constraints. Among these methods, the ROF model [1] relies on the total variation (TV), hence forcing smoothness while preserving edges. The restored image  $u_{TV}$  is obtained by minimizing the following energy:

$$u_{TV} = \operatorname{argmin}_{u \in \mathbb{R}^N} \|u - g\|^2 + \lambda \operatorname{TV}(u) \quad (7)$$

The term  $\|u - g\|^2 = \sum (u_i - g_i)^2$  is a data fidelity term,  $\text{TV}(u) = \sum \|(\nabla u)_i\|$  is a regularization term and  $\lambda > 0$  is the parameter that sets the compromise between data fidelity and smoothness. A lot of methods have been developed in order to solve such minimization problems, among which Chambolle's algorithm [20], the forward-backward algorithm [21] or Chambolle-Pock's algorithm [22].

*Adaptation to other noise statistics:* formula (7) is well adapted to Gaussian noise since it can be seen from a Bayesian point of view as a maximum a posteriori with a data fidelity corresponding to the log-likelihood, with a TV *a priori* on the image. This model can thus be extended to other types of (uncorrelated) noise with an energy of the following form:

$$u_{\text{TV}} = \underset{u \in \mathbb{R}^N}{\operatorname{argmin}} - \sum_{i \in \Omega} \log p(g_i | u_i) + \lambda \text{TV}(u) \quad (8)$$

where  $p(g_i | u_i)$  is the conditional likelihood of the true pixel value  $u_i$  given the observation of the noisy value  $g_i$ .

*Discussion:* minimizing the total variation forces the solution to be piece-wise regular, which is well adapted to denoising while preserving edges. However, a compromise has to be found between regularity on flat areas and preservation of textures, based on the choice of the parameter  $\lambda$ . If flat areas are to be properly denoised,  $\lambda$  needs to be high, so fine textures tend to be over-smoothed. On the other hand, preserving small structures requires a smaller  $\lambda$  that will not allow to recover perfectly the flat areas, resulting in a *staircasing effect*.

#### IV. REGULARIZED NL-MEANS (R-NL)

The proposed model combines both the NL-means and the TV minimization in order to reduce the defaults observed in each method, in particular the *rare patch effect* and the *staircasing effect*. We perform a TV minimization with a non-local data fidelity term as follow:

$$u_{\text{R-NL}} = \underset{u \in \mathbb{R}^N}{\operatorname{argmin}} \sum_{i \in \Omega} \sum_{j \in \Omega} w_{i,j} (g_j - u_i)^2 + \lambda \text{TV}(u). \quad (9)$$

With non-local weights defined as in Section II and  $\lambda = 0$ , the solution of (9) is the solution of the NL-means algorithm presented in (2). With  $w_{i,j} = \delta_{i,j}$  (with  $\delta_{i,j} = 1$  if  $i = j$ , 0 otherwise), we recover the solution of the ROF problem (7).

We can show that problem (9) is equivalent to the following:

$$\underset{u \in \mathbb{R}^N}{\operatorname{argmin}} \sum_{i \in \Omega} \left( \sum_{j \in \Omega} w_{i,j} \right) (u_i - (u_{\text{NL}})_i)^2 + \lambda \text{TV}(u). \quad (10)$$

Hence  $u_{\text{R-NL}}$  can be interpreted as a regular solution fitting  $u_{\text{NL}}$  where the sum of the non-local weights is large.

This allows both the NL-means and the TV regularization to complete each other: on areas where the redundancy is high (homogeneous areas for example), the NL-means select many candidates so the sum of the weights is high. In the energy to minimize, the data fidelity term is then prominent over the regularization term, so the solution is close to the NL-means. This provides good smoothing and prevents the *staircasing effect* observed on smooth areas when treated with TV minimization. In singular structures and edges where the redundancy is low, the NL-means select fewer candidates so

the sum of the weights is low. The regularization term becomes prominent over the data fidelity term, so it will cost less to minimize the total variation of the image. The solution tends to a TV solution, preserving edges while reducing the *rare patch effect*.

The method is intuitive since it is based on the strengths and weaknesses of both the NL-means and the TV minimization. In Section VI, we will propose a simple implementation, in a more general framework, derived directly from mixing an NL-means algorithm with a TV minimization solver.

Figure 1 gives an illustration of solutions obtained by the R-NL algorithm with the TV minimization and the NL-means. The corresponding original and noisy images are displayed in Fig. 3-a) and b). Figure 2 highlights the effect of the adaptive regularization in R-NL compared to the NL-means and TV. Figure 2-a) shows a map of the sum of the weights obtained in the NL-means algorithm. The sum of the weights is high in flat areas, due to a large number of good candidates. Around the head of the cameraman, the sum of the weights is lower, revealing fewer candidates in the averaging process. Figure 2-b) shows the absolute difference between the result of R-NL and the one of the NL-means. The influence of the adaptive TV regularization appears clearly on this image: the regularization is ineffective where the sum of the weights is high while it is strongly enforced where NL-means have left residual noise. Figure 2-c) displays the absolute difference between the results of R-NL and TV minimization. The difference is stronger inside the head of the cameraman or inside the camera, revealing a smoother result on these areas.

#### V. RELATED APPROACHES: HYBRID NON-LOCAL VARIATIONAL MODELS

The idea to combine the NL-means to variational methods or to interpret the NL-means from a variational point of view has been studied in different approaches [23], [24]. Besides, since the NL-means provide interesting results in denoising, several authors have adapted the non-local methods to other problems such as deconvolution, inpainting or super-resolution [25]–[27]. This has often been achieved through a minimization framework that relies on non-local properties.

One of the most famous hybrid methods is the non-local TV (NL-TV) proposed by Gilboa and Osher in [7]. Based on the work on graph Laplacian of Zhou and Scholkopf [28] and Bougleux et al. [29], as well as the definition of non-local regularization terms of Kindermann et al. [8], they define a non-local gradient as follows:

$$(\nabla_w u)(x, y) = (u(x) - u(y)) \sqrt{w(x, y)} \quad (11)$$

where  $w(x, y)$  is the weight that measures the similarity between  $x$  and  $y$ . This leads to the definition of a non-local framework, including the non-local ROF model:

$$u_{\text{NL-TV}} = \underset{u}{\operatorname{argmin}} \|u - g\|^2 + \lambda \sum_{i \in \Omega} \|(\nabla_w u)_i\| \quad (12)$$

$$\text{with } \sum_{i \in \Omega} \|(\nabla_w u)_i\| = \sum_{i \in \Omega} \sqrt{\sum_{j \in \Omega} (u_i - u_j)^2 w_{i,j}}.$$



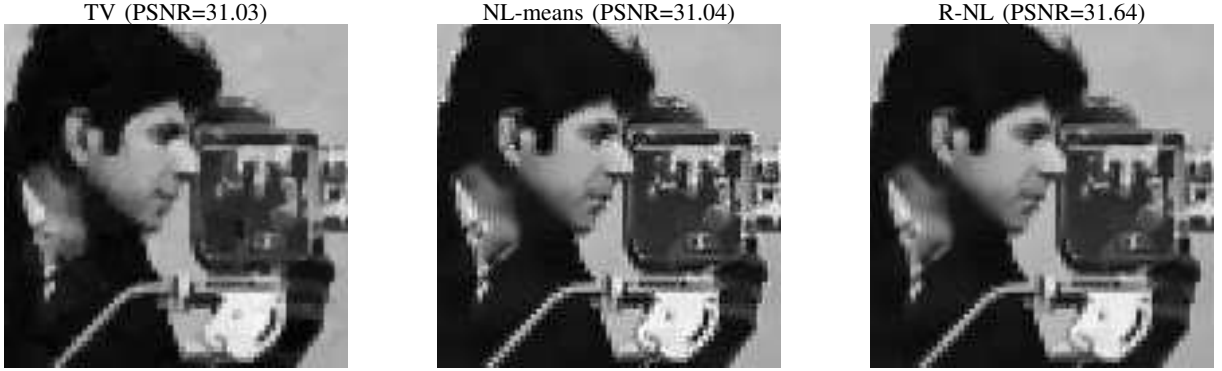


Fig. 1. Denoising of Gaussian noise. We note the ability of R-NL to reduce both the staircasing effect observed on the cameraman with TV minimization and the rare patch effect observed on the NL-means around edges while preserving small structures such as the details on the camera.

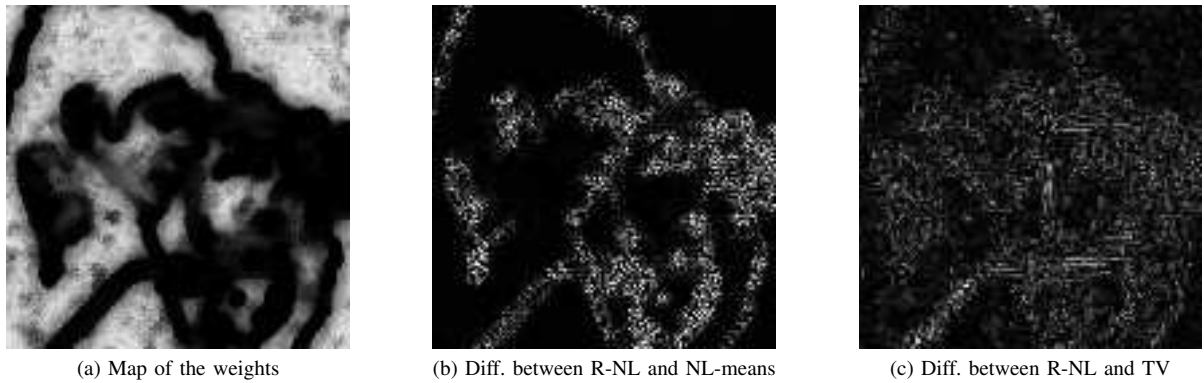


Fig. 2. a) Map of the sum of the non-local weights computed in the NL-means and b) Absolute difference between R-NL and the NL-means. We can see that the weights are lower around the edges, resulting from the rare patch effect. This leads to a stronger TV regularization that compensates this rare patch effect, while no regularization is applied where the weights are higher, preventing the staircasing effect on constant areas. c) Absolute difference between R-NL and TV. We can see that the difference between R-NL and TV are stronger on the flat areas such as the face of the cameraman or the inside of the camera. This shows that the staircasing effect is reduced and that smoothness is better respected with R-NL than with TV.

This model has been introduced to deal separately with textures and smooth areas. It has been adapted to deblurring, inpainting or compressive sensing [9], [10]. Gilboa and Osher have also adapted this non-local regularization to non-local diffusion [30], which offers good denoising results. Figure 3-c) shows the denoising in the case of Gaussian noise with NL-TV. We can see that small structures such as the cables and the writings on the boat are well-preserved while the *staircasing effect* is reduced on flat areas, thanks to the adaptive non-local regularization.

Louchet and Moisan [5] also combine the NL-means with TV minimization in order to reduce both the *rare patch effect* and the *staircasing effect*. First they adapt the gradient to create a local filter and they show that their local TV denoising prevents the staircasing effect. Indeed, on flat areas their filter can be shown to be close to a blur by a low-pass filter, hence avoiding the piecewise approximation in these regions. Then they associate this local TV denoising to the NL-means to deal with textures and edges. TV minimization tends to attenuate textures while the NL-means suffer from the *rare patch effect*. To overcome these effects, Louchet and Moisan [5] adapt the NL-means thanks to local TV minimization: if the NL-means do not select enough candidates to provide sufficient denoising, they apply local TV regularization to the patches

to obtain a sufficient number of candidates. On each pixel of the image TV-means selects the regularization parameter  $\lambda$  that will guarantee a sufficient number of candidates in the NL-means process. This amounts to applying adaptive TV regularization prior to the NL-means in order to ensure that it will find enough candidates in the averaging process. Besides, it involves a threshold controlling the level of denoising wished for. Our idea in the R-NL algorithm actually works the other way, since we first compute the solution of the NL-means and apply adaptive TV regularization afterward on the areas that did not have enough good candidates, based on the sum of the weights obtained from the NL-means. Figure 3-d) shows some results of the TV-means algorithm, that can be compared to the three results presented on Fig. 1. TV-means reduce both the *staircasing effect* compared to the TV minimization and the *rare patch effect* compared to the NL-means.

NL-means have also been associated to TV minimization in a super-resolution context. Protter et al. [11] and d'Angelo and Vandergheynst [12] use a non-local data fidelity term combined to TV minimization in order to obtain a high-quality image from a low-resolution image sequence. Their formulation shows some similarities to our proposed model, but the philosophy behind it remains quite different. Figure 3-e) illustrates this algorithm in a denoising context. This

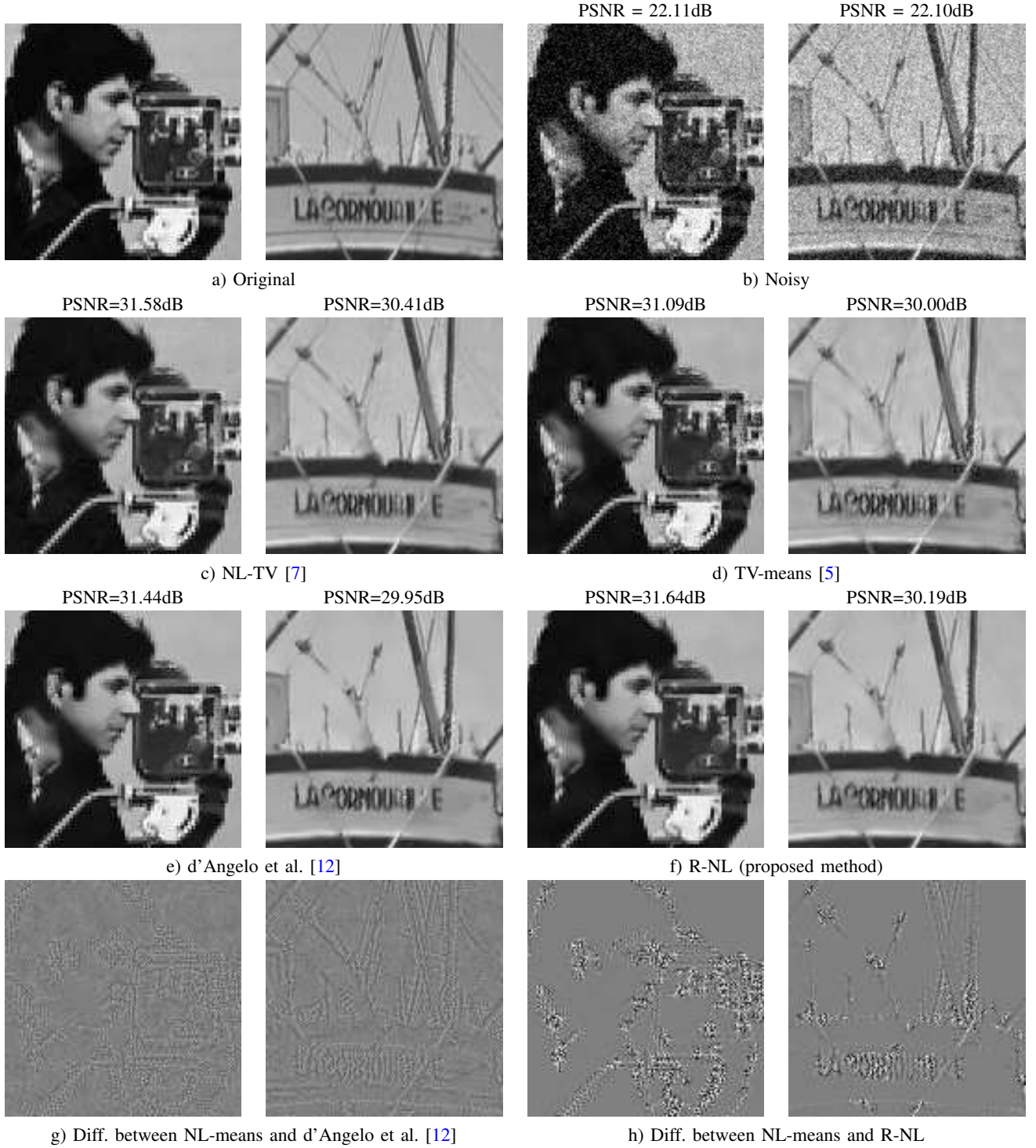


Fig. 3. Denoising of Gaussian noise. From top to bottom, left to right: a) Original and b) noisy images of the Cameraman and the boat, processed with c) NL-TV, d) TV-means, e) d'Angelo et al. and f) R-NL. g) Difference between the solution of the NL-means and the result of e) and h) Difference between the solution of the NL-means and the result of f). We observe on these images the ability of each method to combine variational and non-local methods in order to denoise efficiently while reducing the *staircasing effect* and the *rare patch effect*. See text for more details.

amounts to applying TV regularization on the solution of the NL-means. We can see that the *rare patch effect* can be reduced compared to the solution of the NL-means. However, since the regularization is not adaptive, reducing the *rare patch effect* requires using an excessive smoothing resulting in the loss of small structures. Compared to our model, the

main difference resides in the fact that our weights are not normalized. In fact, we use this very information to adapt the TV minimization step to the performance of the NL-means. The sum of the weights in each pixel reflects the ability of the NL-means to find good candidates and to denoise efficiently. Then the TV minimization completes the denoising

process by smoothing where the NL-means were insufficient while preserving the good properties of the non-local methods elsewhere.

Figure 3 shows the results obtained from the different methods described in this section that combine the NL-means with TV minimization. We can observe that the NL-TV algorithm of Gilboa and Osher does effectively preserve fine structures such as the cables of the boat and it reduces the *staircasing effect*, but the flat areas like the sky around the Cameraman's head are not as smoothed as in the NL-means for example. The TV-means algorithm also reduces the *staircasing effect* and offers smoother areas than NL-TV, but at the cost of a stronger *rare patch effect*. The method based on the super-resolution algorithm of d'Angelo et al. reduces the *rare patch effect* of the NL-means thanks to the TV-regularization, but the compromise is hard to find: for example, the cables of the boat tend to be over-smoothed if the residual noise on the writings is to be efficiently removed. R-NL offers a simple way to deal with this compromise, thanks to the sum of the non-normalized weights that acts as a measure of confidence in the denoising performed by the NL-means. To illustrate further the difference of philosophy between our R-NL method and the algorithm derived from the super-resolution context [12], Fig. 3-g) displays the difference between the solution of the NL-means and the solution of the algorithm derived from [12], while Fig. 3-h) displays the difference between the solution of the NL-means and the solution of our R-NL algorithm. We can observe on Fig. 3-g) that the standard TV regularization applied on the solution of the NL-means smooths the image all over, while the adaptive TV regularization performed in the R-NL method works only on edges and singular structures, based on the confidence in the NL-means provided by the sum of the weights.

## VI. R-NL FOR THE EXPONENTIAL FAMILY

Both TV and NL-means are robust to different kind of noises thanks to possible adaptations described in Sections II and III. The proposed model can then be extended to other types of (uncorrelated) noise with a weighted data fidelity of the form  $-\sum \sum w_{i,j} \log p(g_j|u_i)$ , following the idea in [16], and with weights adapted as presented in Section II. This extended model can be solved efficiently in the general case of the exponential family that includes additive white Gaussian noise, Poisson noise and some multiplicative noises that are encountered frequently in image processing problems such as medical imaging, astronomy, remote sensing applications, etc. A probability law belongs to the exponential family [31] if it can be written under the following form:

$$p(g|u) = h(g) \exp(\eta(u)T(g) - A(u)) \quad (13)$$

where  $h, T, \eta$  and  $A$  are known functions. The extended model is the following:

$$u_{\text{R-NL}} = \underset{u \in \mathbb{R}^N}{\operatorname{argmin}} \sum_{i \in \Omega} Z_i A(u_i) - \eta(u_i) \mu_i + \lambda \text{TV}(u) \\ \text{with } Z_i = \sum_{j \in \Omega} w_{i,j} \quad \text{and} \quad \mu_i = \sum_{j \in \Omega} w_{i,j} T(g_j). \quad (14)$$

---

### Algorithm 1 R-NL

---

**Require:**  $g$ : noisy input image,

$h$ : filtering parameter

$|P|$ : patch size

$\mathcal{N}$ : size and shape of the search neighborhood

$\lambda$ : regularization parameter

#### NL-means step

Computation of the weights:

**for**  $i \in \Omega, j \in \mathcal{N}(i)$  **do**

Compute  $w_{i,j} = \varphi\left(-\frac{d(g(P_i), g(P_j))}{2|P|h^2}\right)$

**end for**

**return**

$$Z = \left( \sum_{j \in \mathcal{N}(i)} w_{i,j} \right)_{i \in \Omega} \\ \mu = \left( \sum_{j \in \mathcal{N}(i)} w_{i,j} T(g_j) \right)_{i \in \Omega}$$

#### Minimization step

Minimize

$$u_{\text{R-NL}} = \underset{u}{\operatorname{argmin}} \sum_{i \in \Omega} Z_i A(u_i) - \eta(u_i) \mu_i + \lambda \text{TV}(u)$$

**return**  $u_{\text{R-NL}}$

---

As in the Gaussian case, it can also be reformulated with a weighted NL-means based fidelity term:

$$u_{\text{R-NL}} = \underset{u \in \mathbb{R}^N}{\operatorname{argmin}} - \sum_{i \in \Omega} Z_i \log p((u_{\text{NL}})_i | u_i) + \lambda \text{TV}(u) \quad (15)$$

where  $\mathcal{L}_w(u_i) = -Z_i \log p((u_{\text{NL}})_i | u_i)$  is the weighted log-likelihood and  $\mu$  and  $Z$  can be calculated with a quick implementation of the NL-means. We refer the interested reader to [32] for a more complete description of a fast way to compute the weights. Then the minimization step is achieved thanks to standard minimization algorithms, depending on the type of noise involved. A general implementation of the R-NL algorithm is given in Algorithm 1. More details regarding the minimization step will be given in the following subsections, according to the type of noise involved.

#### A. Gaussian case

The standard Gaussian model is given in Section IV, equation (9). A lot of methods exist to solve this sort of problem. We have chosen to use Chambolle-Pock's algorithm [22], whose details are given in Algorithm 2. This method deals with minimization problems of the following form:

$$\min_u F(Ku) + G(u) \quad (16)$$

that can be re-written in a primal-dual form:

$$\min_u \max_y \langle Ku, y \rangle - F^*(y) + G(u) \quad (17)$$

More details regarding convex analysis are given in appendix A. In this case, the function  $G$  is the weighted fidelity term, ie the weighted log-likelihood  $\mathcal{L}_w(u_i)$ , and the function  $F(Ku)$  is the total variation  $\|\nabla u\|_1$ , ie  $F = \|\cdot\|_1$  and  $K = \nabla$ .



**Algorithm 2** Chambolle-Pock algorithm [22]

Initialization: choose  $\tau, \sigma > 0$   
 $u_0 = \frac{\mu}{Z}$  the result of the NL-means step  
 $y_0 = \nabla u_0, \bar{u}_0 = u_0$   
Iterations ( $k \geq 0$ ):

$$\begin{cases} y_{k+1} &= (I + \sigma \partial F^*)^{-1} (y_k + \sigma \nabla \bar{u}_k) \\ u_{k+1} &= (I + \tau \partial \mathcal{L}_w)^{-1} (u_k + \tau \nabla^* y_{k+1}) \\ \bar{u}_{k+1} &= 2u_{k+1} - u_k \end{cases}$$

Note that in Algorithm 2,  $\nabla^*$  refers to the adjoint of the gradient, so  $\nabla^* = -\text{div}$ , and the terms  $(I + \sigma \partial F^*)^{-1}$  and  $(I + \tau \partial \mathcal{L}_w)^{-1}$  are the proximal operators associated to the functions  $F^*$  and  $\mathcal{L}_w$ . It is easy to check as in [22] that the proximal operator of  $F^*$  is a soft thresholding and we have:

$$y = (I + \sigma \partial F^*)^{-1}(\tilde{y}) \Leftrightarrow y_i = \frac{\tilde{y}_i}{\max(1, |\tilde{y}_i|)}. \quad (18)$$

We can also show that [22]:

$$u = (I + \tau \partial \mathcal{L}_w)^{-1}(\tilde{u}) \Leftrightarrow u_i = \frac{\tilde{u}_i + 2\tau Z_i(u_{NL})_i}{1 + 2\tau Z_i}. \quad (19)$$

**B. Poisson case**

Poisson noise is also of particular interest since it occurs often in medical imaging or night vision where the number of photons is limited. The probability of the observed (integer) intensity  $g$  according to the true value  $f$  is given by  $p(g|f) = \frac{f^g}{g!} \exp(-f)$ , where  $g$  and  $v$  are both non-negative. The negative log-likelihood writes  $\mathcal{L}(g) = f - g \log f + \log g!$  and the problem becomes:

$$u_{\text{R-NL}} = \underset{u \geq 0}{\operatorname{argmin}} \sum_{i \in \Omega} Z_i u_i - \mu_i \log(u_i) + \lambda \text{TV}(u). \quad (20)$$

This functional is strictly convex, so the uniqueness of the solution is guaranteed. However, the data fidelity term is not differentiable, so we cannot use the forward-backward algorithm. One solution would be to regularize the fidelity term (ie the logarithm). Instead, we use the primal-dual algorithm adapted to the Poisson case. Its general form is the one presented in Algorithm 2, and Anthoine et al. have determined explicitly the proximal operators required in Chambolle-Pock's algorithm in [33]:

$$u = (I + \tau \partial \mathcal{L}_w)^{-1}(\tilde{u}) \Leftrightarrow \quad (21)$$

$$u_i = \begin{cases} \frac{1}{2} \left( \tilde{u}_i - \tau Z_i + \sqrt{(\tilde{u}_i - \tau Z_i)^2 + 4\tau Z_i(u_{NL})_i} \right) & \text{if } (u_{NL})_i > 0, \\ \max(\tilde{u}_i - \tau Z_i, 0) & \text{otherwise} \end{cases}$$

**C. Gamma case**

We also focus on gamma multiplicative noise encountered for example in remote sensing applications in the form of *speckle noise*. The likelihood law is given by  $p(g|v) = \frac{L^L g^{L-1}}{\Gamma(L)v^L} \exp(-\frac{Lg}{v})$ , so the negative log-likelihood is given by  $\mathcal{L}(g|v) = \frac{Lg}{v} + L \log v + \log(\Gamma(L)) - L \log L - (L-1) \log g$ .

**Algorithm 3** FB algorithm [21]

Initialization: choose  $u_0 = \frac{\mu}{Z}$  and  $\gamma > 0$   
Iterations ( $k \geq 0$ ):  
 $u_{k+1} = (I + \gamma \partial F)^{-1} (u_k - \gamma \nabla \mathcal{L}_w(u_k))$

Keeping only the terms that are involved in the minimization step, the problem becomes:

$$u_{\text{R-NL}} = \underset{u > 0}{\operatorname{argmin}} \sum_{i \in \Omega} Z_i \log(u_i) + \frac{\mu_i}{u_i} + \lambda \text{TV}(u) \quad (22)$$

This functional is not convex, so there is no guarantee as to the existence of a unique minimizer. However, we can show that a minimization algorithm will converge towards a stationary point [34]. Since we initialize the algorithm to the solution of the NL-means, it is reasonable to believe that the solution will be close to the global one. We cannot use Chambolle Pock's algorithm since the proximal operator derived from the data-fidelity term is not easy to compute. However, the data fidelity term is differentiable so we can use the forward-backward algorithm [35]. Its purpose is to minimize the following problem:

$$\min_u F(u) + G(u), \quad (23)$$

where  $G$  needs to be differentiable with  $\nabla G$   $1/\beta$ -Lipschitz and  $F$  simple, meaning that its proximal operator is easy to compute. In the present case,  $G$  is the data-fidelity term  $\mathcal{L}_w$ , and  $F$  is the total variation. The algorithm is described in Algorithm 3. Some accelerations such as the FISTA algorithm [36] can also be used, or a generalized version GFB [37] that allows to deal with  $n$  functions. In this algorithm, we cannot compute directly the proximal operator of the total variation  $(I + \gamma \partial F)^{-1}$ , so we have to proceed to an inner loop that will calculate iteratively this proximal operator. The calculation of this proximal operator is achieved through a forward-backward algorithm or its fast version FISTA [38].

Figures 1, 4 and 5 compare the R-NL algorithm with the TV minimization and the NL-means on  $512 \times 512$  images in  $[0, 255]$ . In the Gaussian case, Gaussian noise with standard deviation  $\sigma = 20$  has been added. In the Poisson and gamma case, Poisson noise and gamma noise have been applied respectively to the original images with a noise level set in order to achieve a PSNR of 22dB.

**VII. VIDEO DENOISING USING THE R-NL ALGORITHM**

The NL-means have been adapted in [39] to video denoising, so we would like to adapt our algorithm R-NL to video denoising as well, in order to bring the same advantages. The temporal NL-means achieve spatio-temporal filtering without prior motion-compensation. Indeed, motion estimation is a difficult task that may be nearly impossible to solve in constant areas where the aperture problem is too high. Buades *et al.* have shown that motion compensation is in fact counter-productive in this case: in image sequence denoising, the NL-means use a 3D neighborhood whose third dimension corresponds to the temporal frames. Adding motion



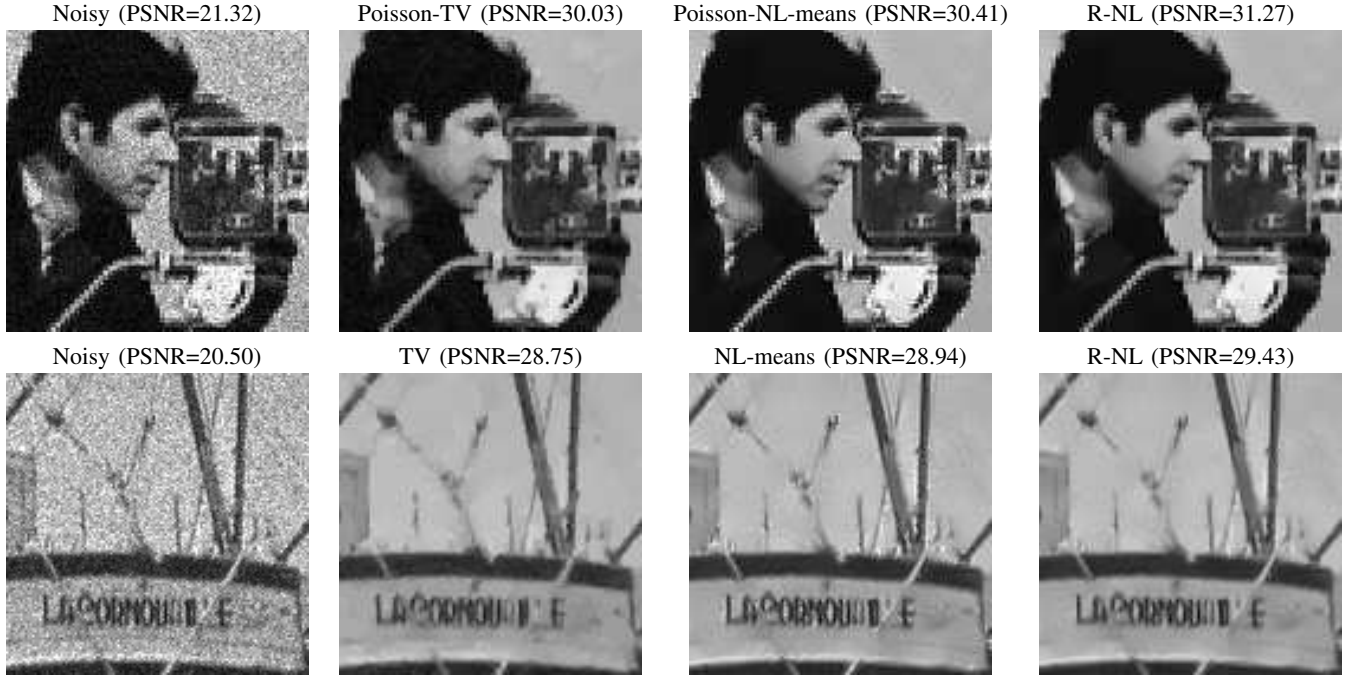


Fig. 4. Denoising of Poisson noise. Small structures such as the writing and the cables of the boat are better preserved with R-NL, while denoising is better achieved on the edges than with NL-means.

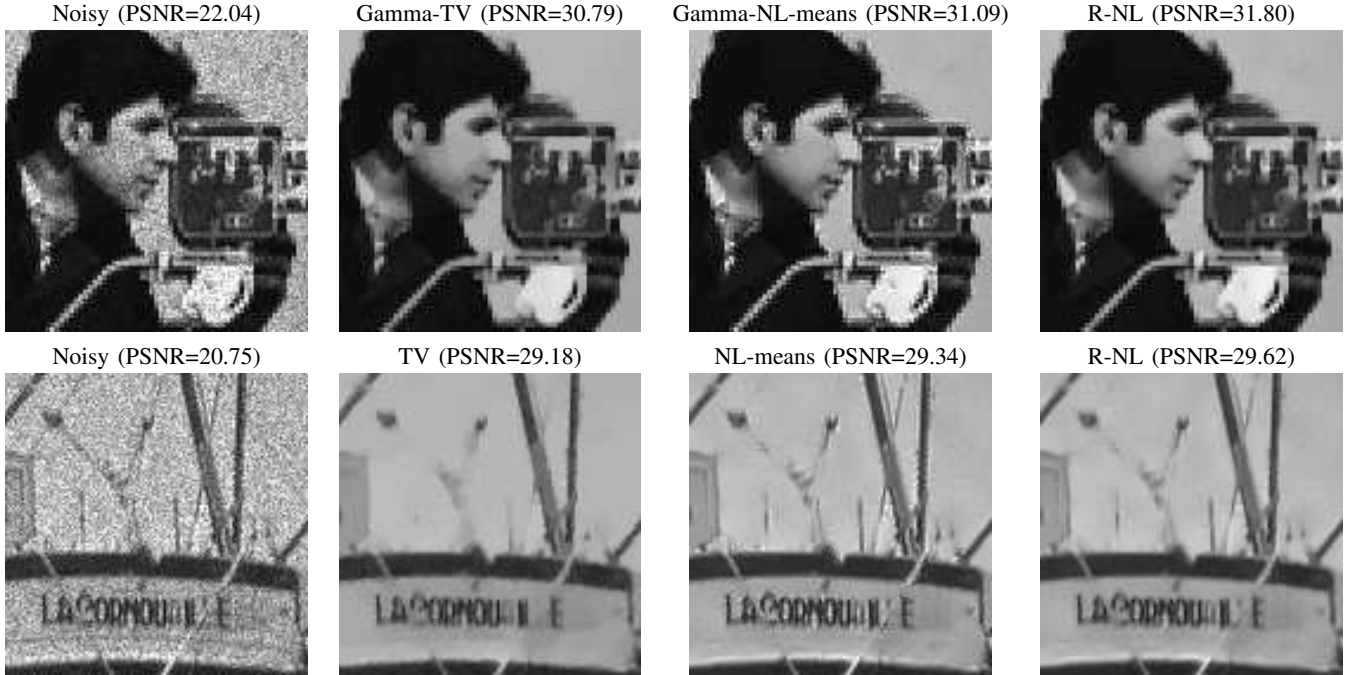


Fig. 5. Denoising of Gamma noise. We observe on the NL-means some residual noise around the face of the cameraman or the writings on the boat, corrected by TV regularization on the R-NL results. The smoother areas are correctly treated by the NL-means, so we do not observe the staircasing effect associated to TV denoising.

compensation reduces the number of eligible candidates, when in the contrary in NL-means the more the merrier.

In the R-NL algorithm, the quantities  $\mu$  and  $Z$  are thus easy to compute in the spatio-temporal domain. However, the TV regularization cannot be performed in the temporal

domain without prior motion compensation, so we apply the regularization spatially only. Hence the R-NL algorithm on image sequences consists in spatio-temporal NL-means followed by spatial TV regularization.

However, on both NL-means and R-NL, if a small number

TABLE I  
PARAMETERS USED FOR IMAGE AND VIDEO DENOISING ACCORDING TO  
THE TYPE OF NOISE INVOLVED

	Gaussian	Poisson	gamma
Image denoising			
Patch size	7		
Search window	13		
Filtering parameter	2	0.06	0.015
Regularization parameter $\lambda$	7	0.06	0.004
Video denoising			
Spatial patch size	7		
Spatial search window	7		
Temporal patch size	5		
Temporal search window	9		
Filtering parameter	0.6	0.025	0.004
Regularization parameter $\lambda$	8	0.04	0.005

of frames is used to compute the solution (which is often the case in order to lower the computational costs), no temporal regularity is guaranteed. Indeed, motion compensation is based on the Lambertian assumption that stipulates that a pixel has the same gray level during its whole trajectory, forcing a temporal regularity. In the case of the NL-means, such a hypothesis is not verified, so a pixel value may change from one frame to another. This results in a glittering effect when looking at the video, even though it is not perceptible when looking at only one frame or at the PSNR. Such a phenomenon is not harmful when the whole image is in motion, but when only a part of the image in the video is in motion while the rest is static, the glittering effect appears like an undesirable consequence of the estimation variability.

To reduce this effect, we use 3D patches instead of 2D patches. Indeed, in the original version on the NL-means, Buades *et al.* compute patches in the current image (in 2D) and compare them to 2D patches in a 3D spatio-temporal search zone. This does not force regularity from one frame to another, resulting in the glittering effect. Thus, we compute 3D patches that compare spatio-temporal neighborhoods in order to force temporal consistency, as in [14]. In the NL-means-3D alone, the use of three-dimensional patches favors the rare patch effect since candidates are harder to find. But using TV regularization afterward balances this drawback, resulting in a more stable video without residual noise or glittering effect.

## VIII. RESULTS AND DISCUSSION

### A. Parameters

The parameters in the different algorithms described below have been optimized in order to offer the best possible results on a set of images or video sequences. The patch size and window size of the NL-means have been maintained but the filtering parameter  $h$  and the regularization parameter  $\lambda$  of the TV regularization are set according to the type of noise. Table I displays the parameters used for image or video denoising, depending on the type of noise.

### B. Image denoising

We present here some numerical results to compare our R-NL algorithm to other approaches relying on non-local and/or variational methods. We can see in Table II that

TABLE II  
PSNR VALUES OF DENOISED IMAGES USING DIFFERENT METHODS FOR  
IMAGES CORRUPTED BY ADDITIVE WHITE GAUSSIAN NOISE, POISSON  
NOISE AND GAMMA NOISE.

	Camerman	Boat	Lena
Gaussian noise, $\sigma = 20$			
Noisy image	22.11	22.10	22.13
TV [1]	31.28	29.42	30.19
NL-means [2]	31.04	29.79	30.94
TV-means [5]	31.09	30.00	31.35
NL-TV [7]	31.58	<b>30.41</b>	<b>31.44</b>
Non-adaptive R-NL [12]	31.43	29.95	31.11
R-NL	<b>31.64</b>	30.19	31.36
BM3D [40]	32.91	31.42	32.34
Poisson noise			
Noisy image	21.32	20.50	21.51
Poisson-TV [33]	30.76	28.61	30.24
Poisson-NL-means [18]	30.41	28.94	30.76
Non-adaptive R-NL	31.19	29.12	31.33
R-NL	<b>31.27</b>	<b>29.43</b>	<b>31.36</b>
Anscombe+BM3D [41]	32.60	30.72	32.55
Gamma noise			
Noisy image	22.04	20.75	22.50
Gamma-TV [34]	30.79	28.10	29.20
Gamma-NL-means	31.09	29.34	31.66
Non-adaptive R-NL	31.27	28.82	31.75
R-NL	<b>31.81</b>	<b>29.62</b>	<b>32.22</b>
log+BM3D [42], [43]	33.24	31.14	33.41

these methods do provide an increase in PSNR compared to the NL-means or the TV minimization, that confirm the visual observations from Figures 1 and 3. In the case of Poisson or gamma noise, we did not include the TV-means or the NL-TV algorithms since they apply only directly to Gaussian noise. Table II illustrates also the benefit of our adaptive regularization (ie using non-normalized weights) compared to the non-adaptive model of d'Angelo *et al.* [12]. Extensions of their model to Poisson and gamma noise have also been included under the name of *Non-adaptive R-NL*. We have furthermore added in this table the results obtained with BM3D [40] in the case of Gaussian noise, and with BM3D applied after variance stabilization using Anscombe transform in the case of Poisson noise [41] and logarithm transform in the case of gamma noise [41]. Since the main goal of this article is to offer an improvement to the NL-means through an interpretation of the sum of the weights as a measure of confidence, we do not claim to offer a better PSNR than BM3D, but we do offer an intuitive interpretation along with a simple and general implementation.

### C. Video restoration

We also study here the behavior of our proposed method to video denoising, on three image sequences: *target*, *tennis* and *bicycle*. The original, noisy and denoised videos are available for download<sup>1</sup>. Table III displays the mean PSNR of the denoised videos using the standard NL-means adapted to video denoising (NL-means 2D), the R-NL algorithm adapted to video denoising (R-NL 2D), the NL-means algorithm with 3-dimensional patches (NL-means 3D), the R-NL algorithm with 3-dimensional patches (R-NL 3D) and the state-of-the-art

<sup>1</sup><http://www.math.u-bordeaux1.fr/~csutour/R-NL/>

TABLE III  
MEAN PSNRs ON DENOISED SEQUENCES WITH NL-MEANS AND R-NL USING 2D PATCHES, NL-MEANS AND R-NL USING 3D PATCHES, AND V-BM3D.

	Target	Tennis	Bicycle
NL-means 2D [39]	29.91	29.31	30.77
R-NL-2D	30.00	29.65	31.72
NL-means 3D	32.02	29.06	29.30
R-NL-3D	<b>32.12</b>	<b>29.93</b>	31.06
V-BM3D [44]	30.21	29.79	<b>32.92</b>

TABLE IV  
TEMPORAL STANDARD DEVIATION ON DENOISED SEQUENCES WITH NL-MEANS AND R-NL USING 2D PATCHES, NL-MEANS AND R-NL USING 3D PATCHES, AND V-BM3D.

	Target	Tennis (1-24)	Tennis (90-148)	Bicycle
NL-means 2D [39]	5.46	4.27	4.69	2.02
R-NL-2D	4.94	3.65	4.11	1.89
NL-means 3D	4.50	5.13	4.82	1.39
R-NL-3D	<b>3.98</b>	<b>3.56</b>	<b>3.91</b>	<b>1.29</b>
V-BM3D [44]	5.25	3.66	4.60	1.76

video denoising algorithm V-BM3D [44]. We can see that our results with the proposed R-NL method with 3 dimensional patches are quite competitive.

We can also show that the video-R-NL algorithm enforces temporal consistency thanks to the use of three-dimensional patches, while limiting the *rare patch effect* thanks to the adaptive spatial TV regularization. We measure temporal variance on different image sequences in order to illustrate the Lambertian assumption: on areas that do not move during a part of the sequence, the pixel value should be unchanged from one frame to another, so the temporal standard deviation should be close to zero. Based on the ground truth of the original sequences, we can select areas that do not vary with time and then calculate the standard deviation obtained on the different denoised versions. Table IV displays the standard deviation computed on such constant areas on the denoised versions of the same three image sequences.

Based on Table IV, we can see that temporal stability is best guaranteed with R-NL-3D, and that the TV regularization corrects some of the glittering effect, compared to the associated NL-means algorithm. Besides, we note that this measurement enlightens two effects: on spatially flat areas, the *glittering effect* is the main default so a higher standard deviation reflects this phenomenon, observed in particular with the NL-means-2D but also V-BM3D. Around edges, if they are supposed to be stable in time, the *rare patch effect* is also responsible for an increase in temporal standard deviation due to the presence of residual noise. The standard deviation reflects both the *glittering effect* and the *rare patch effect*. This is the reason why the standard NL-means offer a lower variance than the NL-means-3D : the *rare patch effect* is stronger with 3D patches due to the difficulty to compare bigger patches.

## IX. CONCLUSION

In this paper, we have presented a new approach that combines the NL-means with TV regularization. We use a weighted non-local data-fidelity term, whose magnitude is

driven by the sum of the weights computed in the NL-means. This sum of the weights offers a measure of confidence in the denoising performed by the NL-means, and the TV regularization is automatically adapted according to this value. This leads to a flexible algorithm that locally uses both the redundant and smooth properties of the images while offering a reduction of the rare patch and staircasing effects that occur respectively in NL-means and TV minimization.

The proposed model has an intuitive interpretation that can be generalized to different noise statistics. Moreover, we have shown that a solution can be achieved quite simply for types of noises belonging to the exponential family.

We also propose an extension to video denoising that does not need motion estimation. The method offers good performances and guarantees temporal stability thanks to the use of three-dimensional patches.

While we have developed our regularized NL-means based on TV regularization, our model naturally extends to other regularization terms adapted to different image priors. Investigating such regularizations is the topic of future work. Besides, we have focused only on the non-local weights provided by NL-means. Beyond non-local denoising, we could also extend this model to other types of algorithms based on weighted averages, hence extending the applicability of this model to a lot of other potential problems.

## APPENDIX A CONVEX ANALYSIS

We give here some elements of convex analysis. We refer to [45] for a complete review on the subject.

Let  $\mathcal{X}$  be a Hilbert space  $\mathcal{C}$  a closed convex subset of  $\mathcal{X}$ .  $\Gamma_0$  is the space of functions  $\mathcal{X} \rightarrow ]-\infty, +\infty]$ , that are convex, proper and lower semi-continuous.

### Legendre-Fenchel conjugate

We define the Legendre-Fenchel conjugate of  $\varphi \in \Gamma_0$  by :

$$\varphi^*(u) = \sup_x \langle u, x \rangle - \varphi(x) \quad (24)$$

We have the following propositions :

*Proposition 1:*  $\varphi^{**} = \varphi \forall \varphi \in \Gamma_0$

*Proposition 2:* If  $\varphi \in \Gamma_0$  is one-homogeneous, that is if  $\varphi(\lambda u) = \lambda \varphi(u) \forall \lambda > 0$ , then:

$$\varphi^*(v) = \chi_K(v) = \begin{cases} 0 & \text{if } v \in K \\ +\infty & \text{otherwise} \end{cases}$$

where  $K$  is a closed convex set, and we have the Legendre-Fenchel relationship:

$$\varphi(x) = \varphi^{**}(x) = \sup_u \langle u, x \rangle - \varphi^*(u) = \sup_{u \in K} \langle u, x \rangle$$

### Sub-gradient

The sub-gradient of  $\varphi \in \Gamma_0$  is given by:

$$\begin{aligned} \partial\varphi(x) &= \{u \in \mathcal{X}, \forall y \in \mathcal{X}, \langle y - x, u \rangle + \varphi(x) \leq \varphi(y)\} \\ &= \{u \in \mathcal{X}, \varphi^*(u) + \varphi(x) \leq \langle x, u \rangle\} \end{aligned}$$

We have the following propositions:

*Proposition 3:*  $x \in \inf \varphi \Leftrightarrow 0 \in \partial\varphi(x)$

*Proposition 4:*  $\forall \varphi \in \Gamma_0, u \in \partial\varphi(x) \Leftrightarrow x \in \partial\varphi^*(u)$ .

### Proximal operator

The proximal operator of  $\varphi \in \Gamma_0$  is defined as:

$$\text{prox}_{\gamma\varphi}x = \underset{y}{\operatorname{argmin}} \varphi(y) + \frac{1}{2\gamma}\|x - y\|^2 \quad (25)$$

We have the following proposition:

$$x - \text{prox}_{\gamma\varphi}x \in \gamma\partial\varphi(\text{prox}_{\gamma\varphi}x) \quad (26)$$

that we write:

$$\text{prox}_{\gamma\varphi}x = (I + \gamma\partial\varphi)^{-1}(x) \quad (27)$$

Using the dual problem, we can show Moreau's identity:

$$x = (I + \gamma\partial\varphi)^{-1}(x) + \gamma \left( I + \frac{1}{\gamma}\partial\varphi^* \right)^{-1} \left( \frac{x}{\gamma} \right) \quad (28)$$

### Forward-backward algorithm

The forward-backward algorithm [35] is a minimization method that solves problems of the form:

$$\min_u F(u) + G(u) \quad (29)$$

with  $F, G \in \Gamma_0$ , with  $G$  differentiable, and  $\nabla G$   $1/\beta$ -Lipschitz. This leads to an iterative resolution:

$$\begin{aligned} x_{n+1} &= \text{prox}_{\gamma_n F}(x_n - \gamma_n \nabla G(x_n)) \\ &= (I + \gamma_n \partial F)^{-1}(x_n - \gamma_n \nabla G(x_n)). \end{aligned} \quad (30)$$

This algorithm is performed, in two steps:

1) Forward (explicit) step:

$$y_n = x_n - \gamma_n \nabla G(x_n)$$

2) Backward (implicit) step:

$$x_{n+1} = \text{prox}_{\gamma_n F}(y_n)$$

The parameter  $\gamma$  needs to verify the condition  $\gamma \leq \beta$  where  $1/\beta$  is the Lipschitz constant of  $\nabla G$ .

### Chambolle-Pock's algorithm

Chambolle-Pock's algorithm [22] minimizes problems of the form

$$\min_x F(Kx) + G(x). \quad (31)$$

with  $F, G \in \Gamma_0$ . We can derive a primal-dual iterative resolution scheme in two steps:

1) Dual step : maximization on  $y$

$$y^{n+1} = (I + \sigma\partial F^*)^{-1}(y^n + \sigma K\bar{x}) \quad (32)$$

2) Primal step : minimization on  $x$

$$x^{n+1} = (I + \tau\partial G)^{-1}(x^n - \tau K^* y^{n+1}) \quad (33)$$

3)  $\bar{x}^{n+1} = x^{n+1} + \theta(x^{n+1} - x^n)$  with  $\theta \in [0, 1]$

### ACKNOWLEDGMENT

C. Sutour would like to thank the DGA and the Aquitaine region for funding her PhD. J.-F. Aujol acknowledges the support of the Institut Universitaire de France. The authors would also like to thank Guy Gilboa for the algorithms of the non-local TV he put at our disposition, and for the advice he gave us. We would also like to thank Cécile Louchet for the results she gave us in order to compare our algorithm to the TV-means.

### REFERENCES

- [1] L. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D*, vol. 60(1):259-268, 1992.
- [2] A. Buades, B. Coll, and J.-M. Morel, "A review of image denoising algorithms, with a new one," *Multiscale Modeling and Simulation*, vol. 4(2), pp. 490-530, Sep 2005.
- [3] M. Mahmoudi and G. Sapiro, "Fast image and video denoising via non-local means of similar neighborhoods," *IEEE Signal Processing Letters*, vol. 12(12), 2005.
- [4] C. Kervrann and J. Boulanger, "Optimal spatial adaptation for patch-based image denoising," *IEEE Trans. Image Processing*, vol. 15(10):2866-2878, 2006.
- [5] C. Louchet and L. Moisan, "Total variation as a local filter," *SIAM Journal on Imaging Sciences*, vol. 4(2):651-694, 2011.
- [6] C.-A. Deledalle, V. Duval, and J. Salmon, "Non-local methods with shape-adaptive patches (nlm-sap)," *Journal of Mathematical Imaging and Vision*, pp. 1-18, 2011.
- [7] G. Gilboa and S. Osher, "Nonlocal operators with applications to image processing," *Multiscale Modeling and Simulation*, vol. 7(3):1005-1028, 2008.
- [8] S. Kindermann, S. Osher, and P. Jones, "Deblurring and denoising of images by nonlocal functionals," *SIAM Journal Multiscale Model. Simul.*, vol. 4(4):1091-1115, 2005.
- [9] G. Peyré, S. Bougleux, and L. D. Cohen, "Non-local regularization of inverse problems," *Inverse Problems and Imaging*, vol. 5, no. 2, pp. 511-530, 2011.
- [10] X. Zhang, M. Burger, X. Bresson, and S. Osher, "Bregmanized non-local regularization for deconvolution and sparse reconstruction," *SIAM Journal Imaging Sciences*, vol. 3(3), pp. 253-276, 2010.
- [11] M. Protter, M. Elad, H. Takeda, and P. Milanfar, "Generalizing the non-local means to super-resolution reconstruction," *IEEE Transactions on image processing*, vol. 18(1):36-51, 2009.
- [12] E. d'Angelo and P. Vanderghynst, "Fully non-local super-resolution via spectral hashing," *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1137-1140, 2011.
- [13] A. Buades, B. Coll, and J.-M. Morel, "Denoising image sequences does not require motion compensation," *Proc. IEEE Conf. Advanced Video and Signal based Surveillance*, pp. 70-74, 2005.
- [14] Y. Wexler, E. Shechtman, and M. Irani, "Space-time completion of video," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 29(3):463-476, March 2007.
- [15] B. Goossens, H. Luong, A. Pizurica, and W. Philips, "An improved non-local denoising algorithm," *Local and non-local approximation (LNLA)*, pp. 143-156, 2008.
- [16] J. Polzehl and V. Spokoiny, "Propagation-separation approach for local likelihood estimation," *Probability Theory and Related Fields*, vol. 135, no. 3, pp. 335-362, 2006.
- [17] C.-A. Deledalle, L. Denis, and F. Tupin, "Iterative weighted maximum likelihood denoising with probabilistic patch-based weights," *IEEE Trans. on Image Processing*, vol. 18(12):2661-2672, 2009.
- [18] —, "How to compare noisy patches ? patch similarity beyond gaussian noise," *International Journal of Computer Vision*, vol. 99(1):86-102, 2012.
- [19] V. Duval, J.-F. Aujol, and Y. Gousseau, "A biais-variance approach for the non-local means," *SIAM Journal on Imaging Sciences*, vol. 4(2):760-788, 2011.
- [20] A. Chambolle, "An algorithm for total variation minimization and applications," *Journal of Mathematical Imaging and Vision*, vol. 20:89-97, 2004.
- [21] P. Combettes and J. Pesquet, "Proximal splitting methods in signal processing," January 2010.
- [22] A. Chambolle and T. Pock, "A first-order primal-dual algorithm for convex problems with applications to imaging," *Journal of Mathematical Imaging and Vision*, vol. 40(1):120-145, 2011.
- [23] S. P. Awate and R. T. Whitaker, "Unsupervised, information-theoretic, adaptive image filtering for image restoration," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 28, no. 3, pp. 364-376, 2006.
- [24] T. Brox and D. Cremers, "Iterated nonlocal means for texture restoration," in *Scale Space and Variational Methods in Computer Vision*, vol. 4485. Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, 2007, pp. 13-24.
- [25] A. Buades, B. Coll, and J.-M. Morel, "Image enhancement by non-local reverse heat equation," *Technical report 22, CMLA, ENS-Cachan, Cachan, France*, 2006.



- [26] M. Mignotte, "A non-local regularization strategy for image deconvolution," *Pattern recognition letters*, vol. 29, pp. 2206–2212, 2008.
- [27] P. Arias, V. Caselles, and G. Facciolo, "Analysis of a variational framework for exemplar-based image inpainting," *Multiscale Modeling & Simulation*, vol. 10, no. 2, pp. 473–514, 2012.
- [28] D. Zhou and B. Scholkopf, "A regularization framework for learning from graph data," *Workshop on Statistical Relational Learning and its Connections to other fields*, 2004.
- [29] S. Bougleux, A. Elmoataz, and M. Melkemi, "Discrete regularization on weighted graphs for image and mesh filtering," in *1st International Conference on Scale Space and Variational Methods in Computer Vision (SSVM)*, vol. 4485. Lecture Notes in Computer Science, 2007, pp. 128–139.
- [30] G. Gilboa and S. Osher, "Nonlocal linear image regularization and supervised segmentation," *SIAM Multiscale Modeling and Simulation*, vol. 6(2):595-630, 2007.
- [31] M. Collins, S. Dasgupta, and R. E. Schapire, "A generalization of principal components analysis to the exponential family," *NIPS*, pp. 617–624, 2002.
- [32] J. Darbon, A. Cunha, T. Chan, S. Osher, and G. Jensen, "Fast non-local filtering applied to electron cryomicroscopy," *IEEE Int. Symposium on Biomedical Imaging (ISBI)*, pp. 1331–1334, 2008.
- [33] S. Anthoine, J.-F. Aujol, Y. Boursier, and C. Mélot, "Some proximal methods for poisson intensity CBCT and PET," *Inverse Problems and Imaging*, vol. 6(4):565-598, 2012.
- [34] G. Aubert and J.-F. Aujol, "A variational approach to remove multiplicative noise," *SIAM Journal on Applied Mathematics*, vol. 68(4):925-946, 2008.
- [35] P. Combettes and V. Wajs, "Signal recovery by proximal forward-backward splitting," *Multiscale Modeling and Simulation*, vol. 4(4):1168-1200, 2005.
- [36] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM J. on Imag. Sci.*, vol. 2(1), 2009.
- [37] H. Raguét, J. Fadili, and G. Peyré, "Generalized forward-backward splitting," *SIAM Journal of Imaging Sciences*, 2012.
- [38] A. Beck and M. Teboulle, "Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems," *IEEE TIP*, vol. 18(11), 2009.
- [39] A. Buades, B. Coll, and J.-M. Morel, "Nonlocal image and movie denoising," *International Journal of Computer Vision*, vol. 76(2), pp. 123–139, 2008.
- [40] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3d transform-domain collaborative filtering," *IEEE Trans. on Image Processing*, vol. 16(8):2080-2095, 2007.
- [41] M. Mäkitalo and A. Foi, "Poisson-gaussian denoising using the exact unbiased inverse of the generalized anscombe transformation," *ICASSP*, pp. 1081–1084, 2012.
- [42] H. Xie, L. Pierce, and F. Ulaby, "Statistical properties of logarithmically transformed speckle," *IEEE Trans. Geosci. Remote Sens.*, vol. 40(3):721-727, 2002.
- [43] M. Mäkitalo, A. Foi, D. Fevrale, and V. Lukin, "Denoising of single-look sar images based on variance stabilization and nonlocal filters," in *Mathematical Methods in Electromagnetic Theory (MMET)*, 2010 International Conference on. IEEE, 2010, pp. 1–4.
- [44] K. Dabov, A. Foi, and K. Egiazarian, "Video denoising by sparse 3d transform-domain collaborative filtering," in *Proc. 15th European Signal Processing Conference. EUSIPCO 2007*, Poznan, Poland, 2007.
- [45] R. Rockafellar, *Convex analysis*. Princeton, NJ, 1970.