

# Non-local Structure Tensor Functionals for Image Regularization

Stamatios Lefkimmiatis, *Member, IEEE*, and Stanley Osher

**Abstract**—We present a non-local regularization framework that we apply to inverse imaging problems. As opposed to existing non-local regularization methods that rely on the graph gradient as the regularization operator, we introduce a family of non-local energy functionals that involves the standard image gradient. Our motivation for designing these functionals is to exploit at the same time two important properties inherent in natural images, namely the local structural image regularity and the non-local image self-similarity. To this end, our regularizers employ as their regularization operator a novel non-local version of the structure tensor. This operator performs a non-local weighted average of the image gradients computed at every image location and, thus, is able to provide a robust measure of image variation. Further, we show a connection of the proposed regularizers to the Total Variation semi-norm and prove convexity. The convexity property allows us to employ powerful tools from convex optimization in order to design an efficient minimization algorithm. Our algorithm is based on a splitting variable strategy which leads to an augmented Lagrangian formulation. To solve the corresponding optimization problem we employ the alternating-direction methods of multipliers. Finally, we present extensive experiments on several inverse imaging problems, where we compare our regularizers with other competing local and non-local regularization approaches. Our results are shown to be systematically superior, both quantitatively and visually.

**Index Terms**—Image reconstruction, non-local regularization, structure tensor, total variation, convex optimization.

## I. INTRODUCTION

**I**NVERSE problems typically arise in several image processing and computer vision applications, including image restoration, image inpainting, image segmentation, optical flow estimation, stereo and 3D reconstruction, etc. In these cases, the task is to estimate underlying features of interest from partial or indirect measurements. In practice, the majority of inverse imaging problems are ill-posed [1]. This implies that in order to obtain a physically or statistically meaningful solution, some type of prior information about the underlying image must be taken into account.

Among the available strategies that one can follow to deal with inverse problems, the variational approach is one of the most widely used. Under this framework, image recovery is cast as the minimization of an energy functional whose minimizer corresponds to the desired solution. Here, of significant importance is the proper selection of the regularizer, which is responsible to favor certain reconstructions. This has triggered

an increasing research interest in the design of regularization functionals that can accurately model important properties of natural images.

One of the most successful regularization criteria that have been proposed in the literature is the Total variation (TV) semi-norm [2]. TV exploits the *local structural regularity* property, that is natural images are typically smooth everywhere except to edges, and employs the image gradient to penalize the image variation with an  $L_1$ -type of penalty. This leads to reconstructions that feature sharp and well-preserved image edges. Inevitably, TV has also some limitations. The main one is that in certain cases it can over-smooth homogeneous regions and create staircase artifacts [3]. To address this issue several modifications/extensions of TV have been proposed. These involve either first-order functionals whose goal is to provide an improved measure of image variation [4]–[7], or higher-order functionals that favor piecewise-smooth instead of piecewise-constant solutions (see [3], [8], [9] and references therein).

The above functionals are considered as local or semi-local, since they involve a regularization operator that acts on a restricted region of the image domain. Recently, a new regularization paradigm has been introduced in [10]–[12], where non-local operators are used instead for defining energy functionals. These operators are non-local in the sense that they allow interactions between image points that can possibly be located far apart. The motivation behind this approach is that natural images exhibit a *non-local self-similarity* property. This means that images often consist of localized patterns that repeat themselves at distant locations in the image domain. Therefore, non-local regularizers can effectively model long-range dependencies and lead to improved reconstruction results.

### A. Contributions

In this work we combine ideas both from the local and non-local regularization settings and introduce a novel non-local family of regularizers. Our regularizers differ from the existing non-local ones in the sense that they involve a non-local operator which depends on the image gradient rather than the graph gradient. The motivation is that this way we can exploit at the same time both the local structural image regularity and the non-local image self-similarity properties. In detail, our key contributions are the following:

1. We extend our previous work [7] and design penalties that employ a non-local version of the structure tensor as the regularization operator. This operator performs a non-local weighted average of the image gradients

The authors are with the Department of Mathematics, University of California, Los Angeles, United States (email: stamatis@math.ucla.edu; sjo@math.ucla.edu).

SL was supported by the Swiss National Science Foundation (SNF) under grant P300P2\_151325 and SO was partially supported by the Keck Foundation.

computed at every image point and, thus, provides a more robust measure of image variation.

2. We prove that our regularization criteria are convex and show a connection with the total variation semi-norm.
3. We introduce the concept of the non-local gradient as the non-local extension of the discrete image gradient. We use this operator to re-express our discrete energy functionals in a form that facilitates its efficient minimization.
4. We develop an efficient optimization algorithm that is based on an augmented Lagrangian formulation of the problem.
5. We provide extensive comparisons of our method with alternative regularizers on several inverse imaging problems.

## II. REGULARIZATION OF INVERSE PROBLEMS

### A. Image-Formation Model and Variational Recovery

We will be concerned with the recovery of signals whose measurements are described by a linear observation model of the form:

$$\mathbf{v}(\mathbf{x}) = A\mathbf{u}(\mathbf{x}) + \mathbf{n}(\mathbf{x}). \quad (1)$$

Here,  $\mathbf{u}(\mathbf{x}) = [u^1(\mathbf{x}) \ u^2(\mathbf{x}) \ \dots \ u^C(\mathbf{x})] : \mathbb{R}^d \mapsto \mathbb{R}^C$  represents the generic underlying vector-valued image consisting of  $C$  channels,  $\mathbf{v}$  are the measurements,  $A$  is a linear operator that corresponds to the impulse response of the imaging device, and  $\mathbf{n}$  is a term that accounts for all possible errors during the acquisition. Hereafter, we will assume  $\mathbf{n}$  to be an independent and identically distributed (i.i.d) Gaussian term.

Despite the linear nature of the forward model in (1), the recovery of  $\mathbf{u}$  from  $\mathbf{v}$  is non-trivial. This is due to the measurement noise and the operator  $A$ , which typically is either ill-conditioned or singular. Therefore, a unique and stable solution does not exist. One way to tackle this problem is to cast the estimation of  $\mathbf{u}$  as the minimization of an objective function of the form:

$$\mathcal{E}(\mathbf{u}) = \frac{1}{2} \|\mathbf{v} - A\mathbf{u}\|_2^2 + \tau\psi(\mathbf{u}). \quad (2)$$

This cost function consists of the quadratic *data fidelity* term, which measures the proximity of the estimate to the measurements, and the *regularizer*  $\psi(\mathbf{u})$ , which encodes the regularity assumptions about the underlying image. The role of the latter is to narrow down the set of plausible solutions by favoring those that exhibit the expected properties. The *regularization parameter*  $\tau \geq 0$  balances the influence of both terms to the solution.

### B. Regularization Functionals Revisited

From the discussion above, it is apparent that the regularizer plays a crucial role in image reconstruction. Most of the regularization approaches that have been introduced so far in the literature can be assigned to two main categories, namely the *synthesis-based* and the *analysis-based* regularization. In the synthesis-based framework the reconstruction takes place in a sparsifying-base, such as the wavelet domain, where a

penalty is imposed on the coefficients of the image in this base [13]. Then the final image is obtained by mapping the reconstructed coefficients back to the image domain through an inverse transform. On the other hand, the analysis-based framework involves regularizers that are directly applied on the image one wishes to reconstruct. Such regularizers have been shown to lead to better recovery results (see for example [14]) and, thus, are mostly preferred.

The analysis-based regularizers can be expressed in the following generic form:

$$R(\mathbf{u}) = \int_{\Omega} \Phi(L\mathbf{u}(\mathbf{x})) \, d\mathbf{x}, \quad (3)$$

where  $\Omega \subseteq \mathbb{R}^d$ ,  $L$  is the regularization operator (scalar or multi-component) acting on the image, and  $\Phi(\cdot)$  is the potential function. Typical choices for  $L$  are differential operators such as the Laplacian (scalar operator), the gradient (vectorial operator), the Hessian (matrix-valued operator) or wavelet-like operators (wavelets, curvelets, ridgelets, etc.), while the potential function usually involves a norm distance.

### C. Total Variation

A very popular regularizer of the form (3) is the Total Variation (TV) semi-norm [2], which for a smooth grayscale image  $u$  ( $C = 1$ ) corresponds to the  $L_1$  norm of the gradient magnitude and it is formally defined as

$$\text{TV} = \int_{\Omega} \|\nabla u(\mathbf{x})\|_2 \, d\mathbf{x}. \quad (4)$$

For the past two decades, TV has been extensively applied to several imaging and computer vision problems. One of the main reasons for its success is its ability to allow sharp features (discontinuities) in the solutions. This stems from the  $L_1$ -type behavior of TV that does not over-penalize high intensity variations. For image reconstruction this means that the result will have well-preserved and sharp edges and, thus, it will be visually appealing. Since TV only applies to scalar images, it has been further extended in several ways to cover the case of vector-valued images [15]–[17]. The main requirement for its vectorial variants is that their definition should coincide with the scalar one in (4) when  $C = 1$ .

### D. Semi-Local Regularization Functionals

While TV has been proven a very powerful regularizer, in several cases its applicability can be limited due to the fact that by design it promotes piecewise-constant solutions. Furthermore, the gradient magnitude, which is employed to penalize the image variation at every point in the image domain, is completely localized and thus it is not very informative of the geometric image structures.

To deal with these limitations of TV, in our prior work we have introduced a family of regularization functionals that involve more general descriptors of image variation which take into account information that is available in a local neighborhood of every point in the image domain [7]. Therefore, the resulting functionals exhibit a semi-local behavior and, thus, can provide a more robust measure of image variation. This is

accomplished by employing in their formulation the structure tensor operator [18].

The structure tensor of an image  $u$  evaluated at a spatial location  $\mathbf{x}$ , denoted as  $S_k u(\mathbf{x}) \in \mathbb{S}_+^d$ , is a  $d \times d$  symmetric positive semi-definite (PSD) matrix. It summarizes the dominant directions of the gradient in a neighborhood centered at the point  $\mathbf{x}$  and it is defined as

$$S_k u(\mathbf{x}) = k_\sigma * \left( \nabla u(\mathbf{x}) \nabla u(\mathbf{x})^T \right), \quad (5)$$

where  $k_\sigma$  is a Gaussian convolution kernel of standard deviation  $\sigma$ . When dealing with vector-valued images the definition of the structure tensor can be extended by replacing the gradient in (5) with the Jacobian operator

$$J\mathbf{u}(\mathbf{x}) = [\nabla u^1(\mathbf{x}) \nabla u^2(\mathbf{x}) \dots \nabla u^C(\mathbf{x})], \quad (6)$$

which is the gradient's extension for vector-valued functions.

The importance of the structure tensor lies in its eigenvalues which provide a rich and discriminative description of the local geometry of the image by summarizing the distribution of the image gradients in the neighborhood specified by the support of the kernel  $k_\sigma$ . Let us consider a 2D image and let  $\lambda_1(\mathbf{x})$  and  $\lambda_2(\mathbf{x})$  represent the largest and smallest eigenvalues of the structure tensor at a spatial point  $\mathbf{x}$ , respectively. When both eigenvalues are relatively small there are small intensity variations in the neighborhood around  $\mathbf{x}$ , indicating that the region is homogeneous. When  $\lambda_1(\mathbf{x})$  is large and  $\lambda_2(\mathbf{x})$  is small there are strong variations but only on a dominant orientation. Therefore, the point  $\mathbf{x}$  is located close to an image edge. When both eigenvalues are large there are high variations on both directions specified by the corresponding eigenvectors and, thus, the point  $\mathbf{x}$  is close to an image corner.

Based on the above, it appears that the eigenvalues of the structure tensor are more informative of the geometric structure of the image than the gradient magnitude which is computed completely locally. This observation has led to the introduction of the Structure tensor Total Variation (STV) functionals [7] which are defined as:

$$\text{STV}_p(u) = \int_{\Omega} \left( \sum_{i=1}^d \lambda_i^{p/2}(\mathbf{x}) \right)^{1/p} d\mathbf{x}, \quad (7)$$

where  $\lambda_i(\mathbf{x})$  is the  $i$ th eigenvalue of the structure tensor evaluated at the point  $\mathbf{x}$  and  $p \geq 1$ .

### E. Non-local Regularization Functionals

Non-local regularization functionals have been recently introduced as a means of modeling complex image structures. These regularizers exploit the non-local self-similarity property which is inherent in natural images. The interest for designing such functionals was initiated by the introduction of the non-local means (NLM) filter, which has been used for the task of image denoising [19], [20].

The NLM filter is a patch-based extension of the bilateral filter [21] and it is defined as

$$\text{NLM}(\mathbf{u})(\mathbf{x}) = \frac{1}{Z(\mathbf{x})} \int_{\Omega} e^{-\frac{d_\alpha(\mathbf{u}(\mathbf{x}), \mathbf{u}(\mathbf{y}))}{\beta^2}} \mathbf{u}(\mathbf{y}) d\mathbf{y}, \quad (8)$$

where

$$d_\alpha(\mathbf{u}(\mathbf{x}), \mathbf{u}(\mathbf{y})) = \int_{\Omega} G_\alpha(t) |\mathbf{u}(\mathbf{x} + t) - \mathbf{u}(\mathbf{y} + t)|^2 dt, \quad (9)$$

$G_\alpha$  is a Gaussian kernel of standard deviation  $\alpha$ ,  $\beta$  acts as a filtering parameter, and  $Z(\mathbf{x}) = \int_{\Omega} e^{-d_\alpha(\mathbf{u}(\mathbf{x}), \mathbf{u}(\mathbf{y})) / \beta^2} d\mathbf{y}$  is the normalization factor. According to the definition of NLM, the filtered version of  $\mathbf{u}$  at the spatial coordinates  $\mathbf{x}$  corresponds to a weighted average of the values of all the image points whose Gaussian neighborhood (*image patch*) is similar to the neighborhood of  $\mathbf{x}$ . The weights are determined according to the degree of similarity between two image patches, which is given by the distance metric defined in Eq. (9). This filtering strategy differs significantly from standard filtering approaches that involve convolutions and it exploits the simple but very effective idea that two points centered in similar patches are very likely to have the same intensity values. Since the search for similar points is not restricted in a local region of the image domain, the averaging in Eq. (8) can involve points that are located far from each other. This leads to a non-local filtering which has been shown to be very effective in eliminating noise.

Due to its success, NLM gathered great attention and it was first interpreted in [22] as a non-convex regularizer based on non-local functionals. These functionals are non-local in the sense that the involved operators allow a point to interact with any other point in the image domain. Then, in [23] NL-means was expressed as a non-local quadratic functional. Later, Gilboa and Osher following ideas from graph theory and specifically the gradient and divergence on graphs, which were first introduced in [11] in a discrete setting, they developed in [10] a non-local regularization framework defined in the continuous domain, while a discrete analog was considered in [12]. This framework made possible the extension of the non-local quadratic regularizers to non-smooth functionals.

The best representative of the existing non-local functionals is the non-local total variation (NLTV) [10]. NLTV is a convex regularizer that involves the non-local graph gradient operator. This operator is defined as

$$\nabla_w u(\mathbf{x}) = (u(\mathbf{y}) - u(\mathbf{x})) \sqrt{w(\mathbf{x}, \mathbf{y})}, \quad \forall \mathbf{y} \in \Omega, \quad (10)$$

where  $w(\mathbf{x}, \mathbf{y}) : \Omega \times \Omega \mapsto \mathbb{R}_+$  is a non-negative weighting function that assigns weights between a pair of spatial points  $(\mathbf{x}, \mathbf{y})$  by taking into account their relative distance as well as the similarity of their corresponding values  $u(\mathbf{x}), u(\mathbf{y})$ .

Based on the non-local graph gradient, NLTV can be expressed in the generic form of (3) as

$$\begin{aligned} \text{NLTV}(u) &= \int_{\Omega} \|\nabla_w u(\mathbf{x})\|_2 d\mathbf{x} \\ &= \int_{\Omega} \sqrt{\int_{\Omega} (u(\mathbf{y}) - u(\mathbf{x}))^2 w(\mathbf{x}, \mathbf{y}) d\mathbf{y}} d\mathbf{x}. \end{aligned} \quad (11)$$

From its definition it is clear that NLTV exhibits a non-local behavior, since it permits all the points in the image domain  $\Omega$  to interact with each other. Moreover, the weights  $w(\mathbf{x}, \mathbf{y})$  are chosen to be large for pairs of points whose neighborhoods are similar and smaller for the rest of the pairs. The non-local interactions and the  $L_1$ -type penalty allow NLTV to model

more efficiently complex geometric image structures than the local regularization functionals, such as TV. Therefore, its use can lead to improved image reconstructions. Discrete-domain extensions of NLTV for vector-valued images have been studied in [24], [25].

#### F. Proposed Non-Local Structure Tensor Total Variation

The non-local functionals that have been introduced so far in the literature are based on the notion of the non-local gradient which is defined on graphs. Therefore, under this approach an image is treated as a generic graph and, thus, its underlying structure is not fully exploited. In this work, we follow an alternative approach with our goal being to introduce non-local regularization functionals that employ the standard gradient operator and, thus, can be directly related to the classical Total Variation and provide a measure of non-local image variation. The motivation for designing such non-local functionals is that by combining ideas from local and non-local regularization strategies we can model at the same time two important image properties, namely the local structural regularity and the non-local self similarity.

The underlying idea of our approach is to define a non-local differential operator and use it as the regularization operator in a functional of the form (3). To do so, our starting point is the structure tensor operator which is defined in Eq. (5). As we mentioned earlier, the power of the structure tensor lies in its ability to encode richer information about the image variation than the gradient operator. This is because its computation involves a local neighborhood of a spatial point, as opposed to the gradient which is computed pointwise. Having in mind the non-local principle, a simple but useful observation is that the information encoded by the structure tensor could be further enriched by extending its scope to the entire image domain.

This idea can be formally presented as follows: Let us consider a 2D vector-valued image  $\mathbf{u}$  and let  $\mathbf{r}$  be an arbitrary 2D direction ( $\|\mathbf{r}\|_2 = 1$ ). The *vectorial directional derivative* of  $\mathbf{u}$  in the direction  $\mathbf{r}$  is computed as  $\partial_{\mathbf{r}}\mathbf{u}(\mathbf{x}) = (\mathbf{J}\mathbf{u}(\mathbf{x}))^T \mathbf{r}$  and its magnitude,  $\|\partial_{\mathbf{r}}\mathbf{u}(\mathbf{x})\|_2$ , yields a measure of intensity change in the location  $\mathbf{x}$  of the image  $\mathbf{u}$  at the direction  $\mathbf{r}$ . For a more robust estimation of the intensity change at a specific direction one can instead use the following measure of *local directional variation* [7]

$$V_{\mathbf{r}}(\mathbf{u}(\mathbf{x})) = \sqrt{\left(k_{\sigma} * \|\partial_{\mathbf{r}}\mathbf{u}\|_2^2\right)}(\mathbf{x}) = \sqrt{\mathbf{r}^T S_k \mathbf{u}(\mathbf{x}) \mathbf{r}}, \quad (12)$$

which involves the structure tensor. This measure is more informative, since it captures the behavior of  $\mathbf{u}$  in a local neighborhood of  $\mathbf{x}$  specified by the support of the convolution kernel  $k_{\sigma}$ . This measure of directional variation can be further improved by modifying it so that its range becomes non-local. Specifically, the reasoning here is that if two image patches are similar, then the gradients at the centers of these patches are also expected to be similar. Subsequently, the averaging of similar gradients can lead to a more robust and accurate estimation of image variations. Based on this observation, we consider a *non-local directional variation*, which is computed as the weighted average of the magnitude of the directional

variation over points centered in similar neighborhoods. We define this non-local directional variation as

$$\begin{aligned} V_{\text{NL}}(\mathbf{u}(\mathbf{x})) &= \left( \int_{\Omega} w(\mathbf{x}, \mathbf{y}) \|\partial_{\mathbf{r}}\mathbf{u}(\mathbf{y})\|_2^2 d\mathbf{y} \right)^{1/2} \\ &= \sqrt{\mathbf{r}^T \left( \int_{\Omega} w(\mathbf{x}, \mathbf{y}) \mathbf{J}\mathbf{u}(\mathbf{y}) (\mathbf{J}\mathbf{u}(\mathbf{y}))^T d\mathbf{y} \right) \mathbf{r}}, \end{aligned} \quad (13)$$

where  $w(\mathbf{x}, \mathbf{y})$  is a non-negative weighting function that assigns appropriate weights between pairs of points  $(\mathbf{x}, \mathbf{y})$ . These weights, as opposed to the local directional variation, are computed not only based on the relative distance of the two points but also based on the photometric distance of their neighborhoods (similarity of their intensity values). A potential candidate for this weighting function is the one used in the NLM filter and reads as

$$w(\mathbf{x}, \mathbf{y}) = e^{-\frac{d_{\alpha}(\mathbf{u}(\mathbf{x}), \mathbf{u}(\mathbf{y}))}{\beta^2}}, \quad (14)$$

where  $d_{\alpha}$  is the ‘‘patch distance’’ defined in (9).

Motivated by the non-local measure of image variation, we introduce a non-local counterpart of the structure tensor which we define as:

$$S_w \mathbf{u}(\mathbf{x}) = \int_{\Omega} w(\mathbf{x}, \mathbf{y}) \mathbf{J}\mathbf{u}(\mathbf{y}) (\mathbf{J}\mathbf{u}(\mathbf{y}))^T d\mathbf{y}. \quad (15)$$

The non-local (NL) structure tensor  $S_w$ , similarly to  $S_k$ , when evaluated at a point  $\mathbf{x}$  corresponds to a symmetric PSD matrix of dimensions  $d \times d$  where its  $(i, j)$  entry,  $1 \leq i, j \leq d$ , is computed as

$$S_w^{(i,j)} \mathbf{u}(\mathbf{x}) = \int_{\Omega} w(\mathbf{x}, \mathbf{y}) \sum_{c=1}^C \partial_{x_i} u_c(\mathbf{y}) \partial_{x_j} u_c(\mathbf{y}) d\mathbf{y}, \quad (16)$$

with  $\partial_{x_i} u_m$  denoting the partial derivative of the  $m$ th channel of the vector-valued image  $\mathbf{u}$  w.r.t the  $i$ -th dimension. To show the potential benefits of using such a non-local operator, in Fig. 1 we present the information captured by the standard structure tensor and its non-local counterpart when they are applied on a grayscale image. From this figure we observe that the edge content of the image is better encoded in the eigenvalues of the NL structure tensor. Indeed the image edges in Fig. 1(c) are sharper and better resolved than in Fig. 1(b).

Having introduced the NL structure tensor, we can now employ it to define our novel family of regularization functionals. Since most of the important information is encoded in the eigenvalues of the NL structure tensor, our non-local energy functionals will be expressed in the generic form

$$R_{\text{NL}}(\mathbf{u}) = \int_{\Omega} \Phi(\varepsilon_1(\mathbf{x}), \dots, \varepsilon_d(\mathbf{x})) d\mathbf{x}, \quad (17)$$

where  $\varepsilon_i$ ,  $i = 1, \dots, d$  are the  $d$  eigenvalues of the NL structure tensor and  $\Phi(\cdot)$  is a potential function. In this work we restrict our attention to potential functions that correspond to  $\ell_p$  norms of the square rooted eigenvalues of the NL structure tensor and can provide a synopsis of the non-local image variation. This leads us to define the family of non-local structure tensor total variation (NLSTV) functionals for  $p \geq 1$  as



Fig. 1. Image edge information captured by the  $\ell_1$ -norm of the square rooted eigenvalues of (b) the structure tensor and (c) the proposed NL structure tensor when applied on image (a).

$$\text{NLSTV}_p(\mathbf{u}) = \int_{\Omega} \left( \sum_{i=1}^d \varepsilon_i^{p/2}(\mathbf{x}) \right)^{1/p} d\mathbf{x}. \quad (18)$$

Next, we prove the following result.

**Theorem 1.** *The NLSTV regularizers for a fixed weighting function  $w(\cdot, \cdot)$  are 1-homogeneous and convex functionals of  $\mathbf{u}$  for all  $p \geq 1$ .*

*Proof:* Our proof follows closely the proof of [7][Theorem A.3]. Let us introduce the compact linear operator  $\mathbf{T}_{\mathbf{x}} = \mathbf{T}(\mathbf{u}, \mathbf{y}; \mathbf{x}) = \sqrt{w(\mathbf{x}, \mathbf{y})} [\nabla u^1(\mathbf{y}), \dots, \nabla u^C(\mathbf{y})] \in \mathcal{H} = L_2(\Omega, \mathbb{R}^{d \times C})$  for arbitrary  $\mathbf{x}, \mathbf{y} \in \Omega$  and a fixed weighting function  $w(\cdot, \cdot)$ . The adjoint of  $\mathbf{T}_{\mathbf{x}}$  is the unique operator  $\mathbf{T}_{\mathbf{x}}^* : \mathcal{H} \mapsto \Omega$  that satisfies

$$\langle \mathbf{f}, \mathbf{T}_{\mathbf{x}} \mathbf{u} \rangle_{\mathcal{H}} = \langle \mathbf{T}_{\mathbf{x}}^* \mathbf{f}, \mathbf{u} \rangle. \quad (19)$$

Now, we can specify the Gram matrix  $\mathbf{G} = \mathbf{T}_{\mathbf{x}}^* \mathbf{T}_{\mathbf{x}}$  of size  $d \times d$ , which characterizes the spectrum of  $\mathbf{T}_{\mathbf{x}}$ , as

$$\begin{aligned} G_{i,j} &= \langle \sqrt{w(\mathbf{x}, \mathbf{y})} \partial_{x_i} \mathbf{u}(\mathbf{y}), \sqrt{w(\mathbf{x}, \mathbf{y})} \partial_{x_j} \mathbf{u}(\mathbf{y}) \rangle_{\mathcal{H}} \\ &= \int_{\Omega} w(\mathbf{x}, \mathbf{y}) \sum_{c=1}^C \partial_{x_i} u^c(\mathbf{y}) \partial_{x_j} u^c(\mathbf{y}) d\mathbf{y} \\ &= S_w^{(i,j)}(\mathbf{x}). \end{aligned} \quad (20)$$

From (20) it occurs that the singular values of  $\mathbf{T}_{\mathbf{x}}$  are related to the eigenvalues of  $S_w \mathbf{u}(\mathbf{x})$ . Indeed, it holds that the  $i$ -th singular value of the finite rank operator  $\mathbf{T}_{\mathbf{x}}$ , denoted as  $\sigma_i(\mathbf{T}_{\mathbf{x}})$ ,  $1 \leq i \leq d$ , can be computed as  $\sigma_i(\mathbf{T}_{\mathbf{x}}) = \sqrt{\varepsilon_i(\mathbf{x})}$ . Consequently, we have that

$$\left( \sum_{i=1}^d \varepsilon_i^{p/2}(\mathbf{x}) \right)^{1/p} = \left( \sum_{i=1}^d \sigma_i^p(\mathbf{T}_{\mathbf{x}}) \right)^{1/p} = \|\boldsymbol{\sigma}(\mathbf{T}_{\mathbf{x}})\|_p. \quad (21)$$

The NLSTV functionals can now be equivalently written as

$$\text{NLSTV}_p(\mathbf{u}) = \int_{\Omega} \|\boldsymbol{\sigma}(\mathbf{T}_{\mathbf{x}})\|_p d\mathbf{x}. \quad (22)$$

From this reformulation of NLSTV it is straight-forward to show the 1-homogeneity property. Next, to show the convexity of NLSTV we use that for two compact operators  $\mathbf{T}_{\mathbf{x}}, \boldsymbol{\Lambda}_{\mathbf{x}}$  and for any  $1 \leq p, q \leq \infty$  with  $1/p + 1/q = 1$  it holds that [7]

$$\|\boldsymbol{\sigma}(\mathbf{T}_{\mathbf{x}})\|_p = \sup_{\|\boldsymbol{\Lambda}_{\mathbf{x}}\|_q \leq 1} \text{trace}(\boldsymbol{\Lambda}_{\mathbf{x}}^* \mathbf{T}_{\mathbf{x}}). \quad (23)$$

Therefore, for  $t \in [0, 1]$  we can show that:

$$\begin{aligned} & \|\boldsymbol{\sigma}(t\mathbf{T}_{1,\mathbf{x}} + (1-t)\mathbf{T}_{2,\mathbf{x}})\|_p \\ & \leq t \sup_{\|\boldsymbol{\Lambda}_{1,\mathbf{x}}\|_q \leq 1} \text{trace}(\boldsymbol{\Lambda}_{1,\mathbf{x}}^* \mathbf{T}_{1,\mathbf{x}}) + (1-t) \sup_{\|\boldsymbol{\Lambda}_{2,\mathbf{x}}\|_q \leq 1} \text{trace}(\boldsymbol{\Lambda}_{2,\mathbf{x}}^* \mathbf{T}_{2,\mathbf{x}}) \\ & = t \|\boldsymbol{\sigma}(\mathbf{T}_{1,\mathbf{x}})\|_p + (1-t) \|\boldsymbol{\sigma}(\mathbf{T}_{2,\mathbf{x}})\|_p. \end{aligned} \quad (24)$$

Since the operator  $\mathbf{T}_{\mathbf{x}}$  is linear in  $\mathbf{u}$ , it turns out that  $\|\boldsymbol{\sigma}(\mathbf{T}_{\mathbf{x}})\|_p$  is a convex functional of  $\mathbf{u}$ . This further implies that the NLSTV functional in (22) is also a convex functional of  $\mathbf{u}$ . ■

Finally, we can also show that our non-local functionals are directly related to the TV semi-norm. Specifically, if we consider a grayscale image  $u$ , then by choosing the weighting function so that it is completely localized, i.e.,

$$w(\mathbf{x}, \mathbf{y}) = \begin{cases} 0, & \text{if } \mathbf{x} \neq \mathbf{y} \\ 1, & \text{if } \mathbf{x} = \mathbf{y}, \end{cases} \quad (25)$$

we get  $\|\sqrt{\varepsilon(\mathbf{x})}\|_p = \|\nabla u(\mathbf{x})\|_2$  for any  $p \geq 1$ . It is then clear that in this degenerate case our regularizer in (18) reduces to the classical TV. In this sense, our regularizers can be considered as a non-local extension of TV.

### III. DISCRETE NON-LOCAL STRUCTURE TENSOR TOTAL VARIATION

Typically, for most of the inverse imaging problems of interest we have to deal with discrete measurements. Therefore, in the rest of this paper we focus on the discrete problem formulation of (1) and its treatment. In this case  $\mathbf{v}$  and  $\mathbf{u}$  are discretized versions of the measurements and the underlying image, respectively, while the discrete analog of the operator  $A$  is the system matrix  $\mathbf{A}$  that behaves as the impulse response of the imaging device.

#### A. Discrete Non-local Gradient

In this section we introduce a novel discrete non-local gradient operator. We will use this operator later to re-express the discrete version of our NLSTV functionals in an alternative form that will allow us to employ robust tools from convex optimization so that we can design an efficient minimization strategy. It is important to note that our non-local operator differs significantly from the non-local gradient that has been introduced in [10], [11]. The main difference is that in our case

the non-local gradient operator is not based on the concept of derivatives on graphs but instead it involves the discrete counterpart of the standard gradient operator.

We assume that the discretized  $d$ -dimensional vector-valued image  $\mathbf{u} = (\mathbf{u}^1, \dots, \mathbf{u}^C)$  is defined on a rectangular grid with unary steps and consists of  $N = N_1 \cdot N_2 \cdot \dots \cdot N_d$  pixels. Each channel  $c$  of  $\mathbf{u}$  ( $c=1, \dots, C$ ) is rasterized in a vector  $\mathbf{u}^c \in \mathbb{R}^N$  and all the image channels are stacked together to form the single vector  $\mathbf{u} \in \mathbb{R}^{NC}$ . We use the notation  $\mathbf{u}_n^c$  to refer to the value of the  $n$ -th pixel of the grid for the  $c$ -th channel of  $\mathbf{u}$  and we use  $\mathbf{u}_n$  to refer to the tuple  $(\mathbf{u}_n^1, \dots, \mathbf{u}_n^C)$ .

Now, let us define the discrete non-local gradient of a grayscale image  $\mathbf{u}^c$  as the linear mapping  $\mathbf{D}_w : \mathbb{R}^N \mapsto \mathcal{X} \triangleq \mathbb{R}^{N \times d \times N}$ . The non-local gradient  $\mathbf{D}_w$ , when applied on the  $n$ -th pixel of  $\mathbf{u}^c$ , corresponds to a matrix of the form<sup>1</sup>:

$$\mathbf{D}_w \mathbf{u}_n^c = [\sqrt{w_{n,1}} \nabla \mathbf{u}_1^c \dots \sqrt{w_{n,N}} \nabla \mathbf{u}_N^c] \in \mathbb{R}^{d \times N}, \quad (26)$$

where  $\nabla \mathbf{u}_n^c$  is the discrete gradient of  $\mathbf{u}^c$  evaluated at pixel  $n$  and  $w_{n,k}$  is the discrete weight that has been assigned to the pair of pixels  $(n, k)$ . Based on the above definition, the non-local gradient evaluated at pixel  $n$  is constructed by : (1) evaluating the discrete gradient of the image at all pixel locations in the image domain, (2) weighting these gradients with the square root of the weights  $w_{n,k}$  with  $1 \leq k \leq N$ , and (3) stacking all these two-dimensional vectors horizontally to form the final matrix. To handle the image boundaries we assume symmetric boundary conditions and employ the discrete gradient as defined in [26]. Note however, that our framework is general enough to accommodate for different discretizations of the gradient operator with alternative assumptions on the image boundaries.

Since the non-local gradient operator is linear, we can further define its adjoint operator, which provides a reverse linear mapping from the space  $\mathcal{X}$  to  $\mathbb{R}^N$ . To do so, we first need to equip the space  $\mathcal{X}$ , which is the target space of  $\mathbf{D}_w$ , with the inner product  $\langle \cdot, \cdot \rangle_{\mathcal{X}}$  and the norm  $\|\cdot\|_{\mathcal{X}}$ . To define them, let  $\mathbf{X}, \mathbf{P} \in \mathcal{X}$  with  $\mathbf{X}_n, \mathbf{P}_n \in \mathbb{R}^{d \times N} \forall n = 1, 2, \dots, N$ . Then, we have:

$$\langle \mathbf{X}, \mathbf{P} \rangle_{\mathcal{X}} = \sum_{n=1}^N \text{trace}(\mathbf{P}_n^T \mathbf{X}_n) \quad (27)$$

and

$$\|\mathbf{X}\|_{\mathcal{X}} = \langle \mathbf{X}, \mathbf{X} \rangle_{\mathcal{X}}^{1/2} = \left( \sum_{n=1}^N \|\mathbf{X}_n\|_F^2 \right)^{\frac{1}{2}}, \quad (28)$$

where  $\text{trace}(\cdot)$  is the trace operator of a matrix and  $\|\cdot\|_F$  is the Frobenius matrix norm. For the Euclidean space  $\mathbb{R}^N$  we use the standard inner product and norm which we denote as  $\langle \cdot, \cdot \rangle_2$  and  $\|\cdot\|_2$ , respectively.

The definition of the adjoint operator  $\mathbf{D}_w^* : \mathcal{X} \mapsto \mathbb{R}^N$  is provided through the following relation of the inner products

$$\langle \mathbf{D}_w \mathbf{u}^c, \mathbf{P} \rangle_{\mathcal{X}} = \langle \mathbf{u}^c, \mathbf{D}_w^* \mathbf{P} \rangle_2. \quad (29)$$

<sup>1</sup>In practice, as we describe in Section III-B we use a sparse version of the weighting function  $w_{n,k}$  and therefore the number of ‘‘active’’ neighbors for each pixel of the image will be equal to  $K$  which is significantly smaller than  $N$ . Consequently, the target space of the non-local gradient will be  $\mathcal{X} \triangleq \mathbb{R}^{N \times d \times K}$  and  $\mathbf{D}_w \mathbf{u}_n^c \in \mathbb{R}^{d \times K}$ .

After carrying out some linear algebra calculations we can express  $\mathbf{D}_w^*$  in a more suitable form that facilitates its numerical computation. This form is provided in the following proposition.

**Proposition 1.** *The adjoint operator  $\mathbf{D}_w^*$  of the discrete non-local gradient evaluated on the  $n$ -th matrix entry,  $\mathbf{P}_n \in \mathbb{R}^{d \times N}$ , of the multidimensional matrix  $\mathbf{P} \in \mathcal{X}$  is given by:*

$$\mathbf{D}_w^* \mathbf{P}_n = -\text{div} \left( \sum_{k=1}^N \sqrt{w_{k,n}} \mathbf{P}_k(:, n) \right), \quad (30)$$

where  $\text{div}$  is the discrete divergence operator, defined in accordance to the discretization scheme of the gradient operator, and  $\mathbf{P}_k(:, n) \in \mathbb{R}^d$  refers to the  $n$ -th column of the  $k$ -th matrix entry of  $\mathbf{P}$ .

Having defined the non-local gradient and its adjoint operator, we can further introduce the discrete non-local Jacobian which applies on vector-valued images. The non-local Jacobian,  $\mathbf{J}_w : \mathbb{R}^{NC} \mapsto \mathbb{R}^{N \times d \times (NC)}$ , is a matrix-valued operator which, similarly to the standard Jacobian operator, when evaluated on the  $n$ -th pixel of a vector-valued image  $\mathbf{u}$  corresponds to a matrix of the form:

$$\mathbf{J}_w \mathbf{u}_n = [\mathbf{D}_w \mathbf{u}_n^1 \dots \mathbf{D}_w \mathbf{u}_n^C] \in \mathbb{R}^{d \times (NC)}. \quad (31)$$

By employing the adjoint of the non-local gradient, we can easily obtain the adjoint of the non-local Jacobian as

$$\mathbf{J}_w^* \mathbf{Y}_n = [\mathbf{D}_w^* \mathbf{Y}_n^1 \dots \mathbf{D}_w^* \mathbf{Y}_n^C]^T \in \mathbb{R}^C, \quad (32)$$

where  $\mathbf{Y}_n = [\mathbf{Y}_n^1 \dots \mathbf{Y}_n^C] \in \mathbb{R}^{d \times (NC)}$ .

### B. Non-local Weights Computation

In the definitions of the non-local operators that we provided above, a weight  $w_{n,k}$  is assigned to every pair of pixels  $(n, k)$ . In practice, mainly due to computational considerations, we use a sparse version of the discrete weighting function which is computed as follows: (1) For each pixel  $n$  in the image domain we extract a patch  $\mathcal{P}(\mathbf{u}_n)$  of size  $\tau \times \tau$  centered around this pixel. (2) We compute the distance of this patch from all the patches whose centers lie inside a specified search window of size  $r \times r$ . To do so, we use  $(d_\alpha)_{n,k}$  which is the discrete version of (9) and it is defined as

$$(d_\alpha)_{n,k} = \sum_{j=-\tau/2}^{\tau/2} (g_\alpha)_j |\mathbf{u}_{n+j} - \mathbf{u}_{k+j}|^2, \quad (33)$$

with  $g_\alpha$  denoting a discrete weighting function of size  $\tau \times \tau$  that defines the image neighborhood. Note that  $k$  is restricted to be at most  $r/2$  pixels far from pixel  $n$ . (3) Out of all the computed distances we keep the  $K$  smallest ones while the rest are set to infinity. Then, we compute the corresponding weights as  $w_{n,k} = e^{-(d_\alpha)_{n,k}/\beta^2}$ . We note that our strategy for computing the non-local weights is similar to the one used in NLTV [10].

An efficient computation of the patch distance  $(d_\alpha)_{n,k}$  was proposed in [27] and uses a similar idea with the integral image [28]. However, this method is only applicable when a uniform kernel  $g_\alpha$  is used. Here, we consider an alternative

fast implementation which is applicable to the more general case of a symmetric kernel,  $(g_\alpha)_j = (g_\alpha)_{-j}$ . This is based on the observation that given the symmetric nature of  $g_\alpha$  we can re-write (33) as

$$(d_\alpha)_{n,k} = \sum_{j=-\tau/2}^{\tau/2} (g_\alpha)_j |\mathbf{u}_{n-j} - \mathbf{u}_{n+l-j}|^2, \quad (34)$$

where  $-r/2 \leq l \leq r/2$  is the relative distance of the pixel  $k$  from the pixel  $n$ . Clearly, (34) corresponds to a discrete convolution of  $g_\alpha$  with  $\mathbf{z}_n^l = |\mathbf{u}_n - \mathbf{u}_{n+l}|^2$ . Therefore, it is now possible to compute with a single convolution the patch distances of all the pixel pairs  $(n, k)$  that have a relative distance of  $l$  pixels. We note that a similar observation was made in [29] in order to speed-up the computation time of the NLM method.

### C. Discrete NL-STV

Let us indicate that a pixel  $k$  is linked to a pixel  $n$  with a non-zero weight  $w_{n,k}$  by using the notation  $k \in \mathcal{N}_n$  where  $\mathcal{N}_n = \{k : w_{n,k} > 0\}$ . Next, we define a ‘‘sparse’’ version of the discrete non-local structure tensor as

$$\mathbf{S}_w \mathbf{u}_n = \sum_{k \in \mathcal{N}_n} w_{n,k} \mathbf{J} \mathbf{u}_k (\mathbf{J} \mathbf{u}_k)^\top \quad (35)$$

where  $\mathbf{J}$  is the discrete Jacobian operator. Based on the discrete NL structure tensor we can now formally define the discrete NLSTV regularizers as

$$\text{NLSTV}_p(\mathbf{u}) = \sum_{n=1}^N \left( \sum_{i=1}^d \varepsilon_{n,i}^{p/2} \right)^{1/p}, \quad (36)$$

where  $\varepsilon_{n,i}$  denotes the  $i$ -th eigenvalue of the discrete non-local structure tensor  $\mathbf{S}_w$  applied on  $\mathbf{u}$  and evaluated at the pixel location  $n$ .

The current form of the proposed regularizers is difficult to work with, mainly because it involves the eigenvalues of a non-linear operator. Consequently, it is not clear how one could proceed with the minimization of such penalties in an inverse problem setting. To deal with this difficulty we derive an alternative definition of the proposed functionals that will prove very useful. This new definition arises by using the following result, whose proof is straight-forward and thus is omitted.

**Proposition 2.** *The discrete non-local structure tensor operator applied on  $\mathbf{u}$  and evaluated at a pixel location  $n$  can be expressed in terms of the non-local Jacobian as:*

$$\mathbf{S}_w \mathbf{u}_n = \mathbf{J}_w \mathbf{u}_n (\mathbf{J}_w \mathbf{u}_n)^\top. \quad (37)$$

Consequently, the eigenvalues of the NL structure tensor,  $\varepsilon_{n,i}$  for  $i = 1 \dots, d$  and  $n = 1, \dots, N$ , can be computed from the singular values of the NL Jacobian,  $\sigma_{n,i}$ , as

$$\varepsilon_{n,i} = \sigma_{n,i}^2. \quad (38)$$

Proposition 2 directly implies that the discrete NLSTV functionals can be equivalently expressed in terms of the

singular values of the non-local Jacobian. Indeed, from (36) and (38) we have

$$\begin{aligned} \text{NLSTV}_p(\mathbf{u}) &= \sum_{n=1}^N \left( \sum_{i=1}^d \sigma_{n,i}^p \right)^{1/p} \\ &= \sum_{n=1}^N \|\mathbf{J}_w \mathbf{u}_n\|_{\mathcal{S}_p}, \end{aligned} \quad (39)$$

where the r.h.s of (39) arises by using the definition of the Schatten norm of order  $p$  [30]. The Schatten matrix norms are directly related to the  $\ell_p$  vector norms. In particular, for a generic complex matrix  $\mathbf{X} \in \mathbb{C}^{n_1 \times n_2}$  the  $\mathcal{S}_p$  norm of  $\mathbf{X}$  can be expressed as the  $\ell_p$  norm of the vector  $\boldsymbol{\sigma}(\mathbf{X})$ , whose entries are the singular values of  $\mathbf{X}$ , i.e.,  $\|\mathbf{X}\|_{\mathcal{S}_p} = \|\boldsymbol{\sigma}(\mathbf{X})\|_p$ . Also note that the class of Schatten norms includes as its members the nuclear norm ( $p = 1$ ), the Frobenius norm ( $p = 2$ ), and the spectral/operator norm ( $p = \infty$ ).

This alternative formulation of the proposed functionals is much easier to handle in an optimization framework. The reason is that the non-linear NL structure tensor has been substituted by the linear operator  $\mathbf{J}_w$  ( $\mathbf{D}_w$  for the case of scalar images). Moreover, the expression in (39) better highlights the fact that the proposed regularizers are convex w.r.t  $\mathbf{u} \forall p \geq 1$ . Indeed, it is straight-forward for one to show that these regularizers are convex, since they can be expressed as a composition of a norm,  $\sum_n \|\cdot\|_{\mathcal{S}_p}$ , and a linear operator,  $\mathbf{J}_w$ .

Based on (39) and Lemma 4.1 in [9] we further derive the following dual definition

$$\text{NLSTV}_p(\mathbf{u}) = \max_{\boldsymbol{\Omega} \in \mathcal{B}_{\infty,q}} \langle \mathbf{u}, \mathbf{J}_w^* \boldsymbol{\Omega} \rangle_2, \quad (40)$$

where  $\boldsymbol{\Omega} = [\boldsymbol{\Omega}_1 \dots \boldsymbol{\Omega}_N] \in \mathcal{X}$  and  $\mathcal{B}_{\infty,q} = \{\boldsymbol{\Omega} \in \mathcal{X} : \|\boldsymbol{\Omega}_n\|_{\mathcal{S}_q} \leq 1\}$ . This expression is extremely useful in case one needs to re-write the minimization of an objective function that includes NLSTV in a min-max formulation. Such reformulation of the minimization problem is necessary when a primal-dual approach, as those proposed in [31], [32], is employed to obtain the solution.

## IV. NUMERICAL OPTIMIZATION

Based on the discrete version of the forward model in (1) and under the assumption that the noise perturbing the measurements is Gaussian, a solution of the regularized inverse problem is derived as

$$\mathbf{u}^* = \arg \min_{\mathbf{u}} \frac{1}{2} \|\mathbf{v} - \mathbf{A} \mathbf{u}\|_2^2 + \tau \|\mathbf{J}_w \mathbf{u}\|_{1,p} + \iota_{\mathcal{C}}(\mathbf{u}), \quad (41)$$

for any  $p \geq 1$ . In Eq. (41)  $\iota_{\mathcal{C}}$  is the indicator function of a convex set  $\mathcal{C}$ , while  $\|\cdot\|_{1,p}$  is a shorthand notation for the mixed vector-matrix norm  $\sum_n \|\cdot\|_{\mathcal{S}_p}$ . The indicator function  $\iota_{\mathcal{C}}$  takes the value 0 if  $\mathbf{u} \in \mathcal{C}$  and  $\infty$  otherwise, and its role is to enforce the solution to lie in  $\mathcal{C}$ . The reason for including  $\iota_{\mathcal{C}}$  in the overall objective function is that in many cases the need for such a constraint arises naturally. For example in many imaging applications it is common to require that the intensities of the reconstructed images should either be non-negative (non-negativity constraint) or lie in a specific range

(box constraint). If one seeks for the unconstrained solution of (41), this can be obtained by simply setting  $\mathcal{C} = \mathbb{R}^{NC}$ .

The form of the problem in (41) is difficult to work with mainly for two reasons. The first one is that the objective function we want to minimize is non-smooth. This precludes the use of a gradient-based scheme. Given the non-smoothness of the objective function, the second reason is the coupling that exists among the different terms. This coupling makes the minimization task even more challenging. To circumvent these difficulties a common strategy that we will also follow here, is to decouple the different terms of the objective function by introducing a set of auxiliary variables. This will allow us to find the solution by solving a sequence of simpler problems.

In particular, we use the auxiliary variables  $\mathbf{z}_1 = \mathbf{J}_w \mathbf{u} \in \mathcal{X}$  and  $\mathbf{z}_2 = \mathbf{u} \in \mathbb{R}^{NC}$  and we reformulate the problem in (41) in the constrained form

$$\arg \min_{\substack{\mathbf{u}, \mathbf{z}_1, \mathbf{z}_2 \\ \mathbf{K}\mathbf{u} = \mathbf{z}}} \underbrace{\frac{1}{2} \|\mathbf{v} - \mathbf{A}\mathbf{u}\|_2^2}_{f(\mathbf{u})} + \underbrace{\tau \|\mathbf{z}_1\|_{1,p}}_{g_1(\mathbf{z}_1)} + \underbrace{\iota_C(\mathbf{z}_2)}_{g_2(\mathbf{z}_2)}. \quad (42)$$

The *augmented Lagrangian* [33] associated to (42) is given by

$$\begin{aligned} \mathcal{L}_\mu(\mathbf{u}, \mathbf{z}, \boldsymbol{\eta}) &= f(\mathbf{u}) + g(\mathbf{z}) + \langle \boldsymbol{\eta}, \mathbf{K}\mathbf{u} - \mathbf{z} \rangle_{\mathcal{Y}} \\ &\quad + \frac{\mu}{2} \|\mathbf{K}\mathbf{u} - \mathbf{z}\|_{\mathcal{Y}}^2, \end{aligned} \quad (43)$$

where  $\mathcal{Y} \triangleq \mathcal{X} \times \mathbb{R}^N$  is a multilinear space,  $\boldsymbol{\eta} = (\boldsymbol{\eta}_1, \boldsymbol{\eta}_2) \in \mathcal{Y}$  are the Lagrange multipliers,  $\mathbf{z} = (\mathbf{z}_1, \mathbf{z}_2) \in \mathcal{Y}$ ,  $g(\mathbf{z}) = g_1(\mathbf{z}_1) + g_2(\mathbf{z}_2)$ ,  $\mathbf{K} = (\mathbf{J}_w, \mathbf{I})$  is a concatenated version of the non-local Jacobian  $\mathbf{J}_w$  and the identity operator  $\mathbf{I}$ , and  $\mu \geq 0$  is a penalty parameter. Note that  $\mathcal{L}_0$  corresponds to the standard Lagrangian of problem (42) while  $\mathcal{L}_\mu$  corresponds to the Lagrangian of the problem

$$\arg \min_{\substack{\mathbf{u}, \mathbf{z}_1, \mathbf{z}_2 \\ \mathbf{K}\mathbf{u} = \mathbf{z}}} f(\mathbf{u}) + g(\mathbf{z}) + \|\mathbf{K}\mathbf{u} - \mathbf{z}\|_{\mathcal{Y}}^2. \quad (44)$$

In any case, the two problems (42) and (44) are clearly equivalent, since for any feasible solution of (44) the additional quadratic term in (44) evaluates to zero.

To solve the problem in (42) we use the augmented Lagrangian and employ the *alternating-direction method of multipliers* (ADMM) [34]–[36]. This is an iterative method that attacks the constrained optimization by replacing it by a series of unconstrained problems. To this end, one ADMM iteration involves the minimization of  $\mathcal{L}_\mu$  w.r.t  $\mathbf{z}$  while keeping  $\mathbf{u}$  fixed, the minimization of  $\mathcal{L}_\mu$  w.r.t  $\mathbf{u}$  while keeping  $\mathbf{z}$  fixed, and an update of the Lagrange multipliers  $\boldsymbol{\eta}$ . Formally, we can describe the steps involved in every ADMM iteration as:

$$\mathbf{z}^{t+1} = \arg \min_{\mathbf{z} \in \mathcal{Y}} g(\mathbf{z}) + \frac{\mu}{2} \|\mathbf{z} - (\mathbf{K}\mathbf{u}^t + \mathbf{s}^t)\|_{\mathcal{Y}}^2 \quad (45a)$$

$$\mathbf{u}^{t+1} = \arg \min_{\mathbf{u} \in \mathbb{R}^{NC}} f(\mathbf{u}) + \frac{\mu}{2} \|\mathbf{K}\mathbf{u} - (\mathbf{z}^{t+1} - \mathbf{s}^t)\|_{\mathcal{Y}}^2 \quad (45b)$$

$$\mathbf{s}^t = \mathbf{s}^t + \mathbf{K}\mathbf{u}^{t+1} - \mathbf{z}^{t+1}, \quad (45c)$$

where  $\mathbf{s} = \boldsymbol{\eta}/\mu$  corresponds to a scaled version of the original Lagrange multipliers and in each one of the subproblems we have ignored constant terms that are irrelevant to the optimization task.

Next, we focus on the solution of the individual subproblems. First, based on the definition of  $g(\mathbf{z})$ , we note that the  $\mathbf{z}$ -update of ADMM (45a) can be decoupled in the following two independent problems

$$\begin{aligned} \mathbf{z}_1^{t+1} &= \arg \min_{\mathbf{z}_1 \in \mathcal{X}} \frac{1}{2} \|\mathbf{z}_1 - (\mathbf{J}_w \mathbf{u}^t + \mathbf{s}_1^t)\|_{\mathcal{X}}^2 + \frac{\tau}{\mu} \|\mathbf{z}_1\|_{1,p} \\ \mathbf{z}_2^{t+1} &= \arg \min_{\mathbf{z}_2 \in \mathbb{R}^{NC}} \frac{1}{2} \|\mathbf{z}_2 - (\mathbf{u}^t + \mathbf{s}_2^t)\|_2^2 + \iota_C(\mathbf{z}_2). \end{aligned} \quad (46)$$

The first problem corresponds to the evaluation of the proximal map of the function  $g_1(\mathbf{z}_1)$  scaled by  $\mu$ . Indeed, the proximal map of a function  $f$  with  $\text{dom} f = \mathcal{X}$ , evaluated at  $\mathbf{z}$  is defined as [37]

$$\text{prox}_f(\mathbf{z}) = \arg \min_{\mathbf{u} \in \mathcal{X}} \frac{1}{2} \|\mathbf{u} - \mathbf{z}\|_{\mathcal{X}}^2 + f(\mathbf{u}). \quad (47)$$

Therefore, it holds that

$$\mathbf{z}_1^{t+1} = \text{prox}_{\frac{\tau}{\mu} \|\cdot\|_{1,p}}(\mathbf{J}_w \mathbf{u}^t + \mathbf{s}_1^t). \quad (48)$$

Due to the separability of  $\|\cdot\|_{1,p}$  and of the quadratic term  $\|\cdot\|_{\mathcal{X}}^2$ , the above problem can be further decomposed in  $N$  independent subproblems of the form:

$$\begin{aligned} (\mathbf{z}_1^{t+1})_n &= \text{prox}_{\frac{\tau}{\mu} \|\cdot\|_{S_p}}(\boldsymbol{\Omega}_n) \\ &= \arg \min_{(\mathbf{z}_1)_n} \frac{1}{2} \|(\mathbf{z}_1)_n - \boldsymbol{\Omega}_n\|_F^2 + \frac{\tau}{\mu} \|(\mathbf{z}_1)_n\|_{S_p}, \end{aligned} \quad (49)$$

where  $\boldsymbol{\Omega} = \mathbf{J}_w \mathbf{u}^t + \mathbf{s}_1^t \in \mathcal{X}$  and  $\boldsymbol{\Omega}_n \in \mathbb{R}^{d \times K_C}$  denotes the  $n$ -th matrix entry of  $\boldsymbol{\Omega}$  ( $K$  is the number of neighbors for a pixel  $n$  with non-zero weights  $w_{n,k}$ ).

To compute the solution in (49) we use the following result, where  $\mathbb{U}^n = \{\mathbf{X} \in \mathbb{C}^{n \times n} : \mathbf{X}^{-1} = \mathbf{X}^H\}$  denotes the set of unitary matrices and  $\mathbb{D}^{n_1 \times n_2} = \{\mathbf{X} \in \mathbb{R}_+^{n_1 \times n_2} : \mathbf{X}_{(i,j)} = 0 \forall i \neq j\}$  denotes the set of PSD diagonal matrices.

**Proposition 3** ([38]). *Let  $\mathbf{Y} \in \mathbb{C}^{n_1 \times n_2}$  be a generic complex matrix with an SVD decomposition  $\mathbf{Y} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^H$ , where  $\mathbf{U} \in \mathbb{U}^{n_1}$ ,  $\mathbf{V} \in \mathbb{U}^{n_2}$ , and  $\boldsymbol{\Sigma} \in \mathbb{D}^{n_1 \times n_2}$ . The proximal map*

$$\mathbf{X}^* = \text{prox}_{\tau \|\cdot\|_{S_p}}(\mathbf{Y}) \quad (50)$$

can be obtained as

$$\mathbf{X}^* = \mathbf{U} \text{diag}\left(\text{prox}_{\tau \|\cdot\|_p}(\boldsymbol{\sigma})\right) \mathbf{V}^H, \quad (51)$$

where  $\text{diag}(\cdot)$  is the operator that transforms a vector to a diagonal matrix and  $\boldsymbol{\sigma}$  denotes the vector whose entries are the singular values of  $\mathbf{Y}$ .

In words, Proposition 3 states that the proximal map of an  $S_p$  matrix-norm can be computed by (1) decomposing the matrix  $\mathbf{Y}$  in its singular values and singular vectors, (2) evaluating the proximal map of the  $\ell_p$  norm at the vector  $\boldsymbol{\sigma}$  consisting of the singular values of  $\mathbf{Y}$  and, (3) deriving the final result by a singular value reconstruction that involves the singular vectors of  $\mathbf{Y}$  and as singular values the result of step 2.

**Efficient evaluation of the proximal map:** Clearly the solution of (49) depends on our ability to evaluate the proximal

map of an  $\ell_p$  norm weighted by a constant  $\tau$ . To do so for an arbitrary  $p \geq 1$ , we rely on the iterative proximal algorithm introduced in [39]. Furthermore, for the most interesting cases  $p = 1, 2$  we can derive the solution in closed-form. Specifically, for  $p = 1$  the proximal map corresponds to the soft-thresholding function  $S_\tau^1(\boldsymbol{\sigma}) = \max(\boldsymbol{\sigma} - \tau, 0)$  [40] where the max is computed component-wise. For  $p = 2$  the proximal map corresponds to another shrinkage function of the form  $S_\tau^2(\boldsymbol{\sigma}) = \max((\|\boldsymbol{\sigma}\|_2 - \tau) / \|\boldsymbol{\sigma}\|_2, 0) \cdot \boldsymbol{\sigma}$  [35], [37]. Notably in this case, we can obtain the proximal map of the Frobenius norm ( $S_2$  norm) without having to resort to an SVD decomposition of the input matrix. Specifically, we have

$$\text{prox}_{\tau \|\cdot\|_F}(\mathbf{Y}) = \max\left(\frac{\|\mathbf{Y}\|_F - \tau}{\|\mathbf{Y}\|_F}, 0\right) \cdot \mathbf{Y}. \quad (52)$$

Another consideration regarding the evaluation of the proximal map of (49) is the efficient computation of the singular values and vectors of the matrices  $\boldsymbol{\Omega}_n = \mathbf{U}_n \boldsymbol{\Sigma}_n \mathbf{V}_n^T \in \mathbb{R}^{d \times KC}$  with  $n = 1, 2, \dots, N$ . Since even for images of moderate sizes the value of  $N$  can be of the order  $10^5$ , we need an efficient way to perform these SVDs. Here, we are interested in 2D images ( $d = 2$ ) and we can follow an efficient strategy as described next. First, we observe that the matrix  $\boldsymbol{\Omega}_n \boldsymbol{\Omega}_n^T$  is  $2 \times 2$  symmetric with an eigenvalue decomposition  $\mathbf{U}_n \boldsymbol{\Sigma}_n^2 \mathbf{U}_n^T$ . Therefore, both  $\mathbf{U}_n$  and  $\boldsymbol{\Sigma}_n$  are obtained in closed-form. Now, let  $\boldsymbol{\Sigma}_n^+$  be the pseudo-inverse of  $\boldsymbol{\Sigma}_n$ . Then, based on Proposition 3 we can compute the proximal maps as

$$\text{prox}_{\frac{\tau}{\mu} \|\cdot\|_{S_p}}(\boldsymbol{\Omega}_n) = (\mathbf{U}_n \boldsymbol{\Sigma}_n^* \boldsymbol{\Sigma}_n^+ \mathbf{U}_n^T) \boldsymbol{\Omega}_n \quad (53)$$

where  $\boldsymbol{\Sigma}_n^* = \text{diag}(\text{prox}_{\tau \|\cdot\|_p}(\boldsymbol{\sigma}_n))$  and  $\boldsymbol{\sigma}_n$  is the vector formed by the diagonal elements of  $\boldsymbol{\Sigma}_n$ .

The second minimization problem in (46) corresponds to the projection of  $\mathbf{u}^t + \mathbf{s}_2^t$  on the convex set  $\mathcal{C}$ . For the imaging applications that we study later, we consider the convex set  $\mathcal{C} \triangleq \{\mathbf{u} \in \mathbb{R}^N : 0 \leq \mathbf{u}_n \leq \gamma, \forall n = 1, \dots, N\}$ . In this case the projection is simple to compute and is given by  $\Pi_{\mathcal{C}}(\mathbf{u}) = \min(\max(0, \mathbf{u}), \gamma)$  where the min and max operations are computed component-wise.

The problem in (45b) is quadratic and thus the  $\mathbf{u}$ -update is obtained as the solution of the set of linear equations

$$\mathbf{B} \mathbf{u}^{t+1} = \frac{1}{\mu} \mathbf{A}^T \mathbf{v} + \mathbf{J}_w^* (\mathbf{z}_1^{t+1} - \mathbf{s}_1^t) + \mathbf{z}_2^{t+1} - \mathbf{s}_2^t, \quad (54)$$

where  $\mathbf{B} = \left(\frac{1}{\mu} \mathbf{A}^T \mathbf{A} + \mathbf{J}_w^* \mathbf{J}_w + \mathbf{I}\right)$ . In practice, the inversion of  $\mathbf{B}$  is prohibitive due to its large size. Therefore, given that  $\mathbf{B}$  is a symmetric positive definite matrix, we employ instead the conjugate gradient (CG) method [41]. We have experimentally observed that for several inverse imaging problems it suffices to run as few as two CG iterations in every ADMM iteration. This choice does not seem to compromise the convergence of the overall algorithm if we follow a ‘‘warm-start’’ strategy where in each ADMM iteration we initialize CG with the solution of  $\mathbf{u}$  obtained in the previous iteration.

---

**Algorithm 1** : Image Reconstruction Algorithm.

---

**Input:**  $\mathbf{v}$ ,  $\mathbf{A}$ ,  $\tau > 0$ ,  $\mu > 0$ ,  $p \geq 1$ .

Initialization:  $\mathbf{u}^0 = \mathbf{v}$ ,  $\mathbf{s}_1^0 = \mathbf{0}$ ,  $\mathbf{s}_2^0 = \mathbf{0}$ ,  $t = 0$ .

**while** stopping criterion is not satisfied **do**

$$\begin{aligned} \mathbf{z}_1^{t+1} &\leftarrow \text{prox}_{\frac{\tau}{\mu} \|\cdot\|_{1,p}}(\mathbf{J}_w \mathbf{u}^t + \mathbf{s}_1^t); \\ \mathbf{z}_2^{t+1} &\leftarrow \Pi_{\mathcal{C}}(\mathbf{u}^t + \mathbf{s}_2^t); \\ \mathbf{B} &\leftarrow \left(\frac{1}{\mu} \mathbf{A}^T \mathbf{A} + \mathbf{J}_w^* \mathbf{J}_w + \mathbf{I}\right); \\ \boldsymbol{\omega}_1^{t+1} &\leftarrow \mathbf{z}_1^{t+1} - \mathbf{s}_1^t; \\ \boldsymbol{\omega}_2^{t+1} &\leftarrow \mathbf{z}_2^{t+1} - \mathbf{s}_2^t; \\ \mathbf{u}^{t+1} &\leftarrow \mathbf{B}^{-1} \left(\frac{1}{\mu} \mathbf{A}^T \mathbf{v} + \mathbf{J}_w^* \boldsymbol{\omega}_1^{t+1} + \boldsymbol{\omega}_2^{t+1}\right); \\ \mathbf{s}_1^{t+1} &\leftarrow \mathbf{s}_1^t + \mathbf{J}_w \mathbf{u}^{t+1} - \mathbf{z}_1^{t+1}; \\ \mathbf{s}_2^{t+1} &\leftarrow \mathbf{s}_2^t + \mathbf{u}^{t+1} - \mathbf{z}_2^{t+1}; \\ t &\leftarrow t + 1; \end{aligned}$$

**end**

**return**  $\mathbf{u}^t$ ;

---



Fig. 2. Thumbnails of the color versions of the 12 images used in the experiments (the numbering order is from left to right and top to bottom). These images are of size of  $481 \times 321$  pixel and are a subset of the Berkeley BSDS500 image dataset.

Finally, the update of the scaled Lagrange multipliers in (45c) is decoupled and it is computed as

$$\begin{aligned} \mathbf{s}_1^{t+1} &= \mathbf{s}_1^t + \mathbf{J}_w \mathbf{u}^{t+1} - \mathbf{z}_1^{t+1}, \\ \mathbf{s}_2^{t+1} &= \mathbf{s}_2^t + \mathbf{u}^{t+1} - \mathbf{z}_2^{t+1}. \end{aligned} \quad (55)$$

A summary of our overall minimization approach is provided in Algorithm 1.

## V. APPLICATIONS AND EXPERIMENTS

To assess the potentials of the proposed non-local functionals, we compare their reconstruction performance with that of other related methods on several inverse imaging applications. In particular, we consider the problems of image denoising, image deblurring, and image reconstruction from undersampled Fourier measurements. For all the problems under study we report results on both grayscale and color versions of the images shown in Fig. 2, where their intensities have been normalized to lie in the range  $[0, 1]$ . The comparisons are performed among the first-order functionals TV [2],  $\text{STV}_1$  [7], NLTV [10], and our proposed  $\text{NLSTV}_1$ . For the color case,

since TV is applicable only to grayscale images we use instead for our comparisons the vectorial total variation (VTV) that was proposed in [16]. The same applies to NLTV where for the color case we use a vectorial version that has been studied in [24], [25]. Furthermore, for the problem of image denoising we also include the results obtained by the NLM filter [20].

The reconstruction performance of the regularization methods under comparison highly depends on the choice of the regularization parameter  $\tau$ . This also holds true for the NLM filter where the parameter  $\beta$  in (8) needs to be tuned. In order our comparisons to be consistent, we have experimentally fine-tuned these parameters and here we report only the results that led to the best peak signal-to-noise ratio (PSNR). For  $STV_1$  we also had to choose the convolution kernel that is used for the computation of the structure tensor in (5). This was set to be a Gaussian of standard deviation  $\sigma = 0.5$  and support of  $3 \times 3$  pixel. For the non-local techniques we had to further specify the size of the neighborhood that is used for the computation of the non-local weights  $w(\cdot, \cdot)$ . We have considered a uniform neighborhood (patch) of size  $5 \times 5$  and  $7 \times 7$  and a search window of size  $11 \times 11$ . For NLTV and NLM we experimentally observed that the  $5 \times 5$  patch led to better results on average than the  $7 \times 7$  patch, while for  $NLSTV_1$  the  $7 \times 7$  patch turned out to be the best choice. Finally, for NLTV and  $NLSTV_1$  we considered a sparse version of the weight function, where only  $K$  neighbors are linked to each pixel of the image with a non-zero weight. In order to be able to compare directly the performance of  $STV_1$  with our non-local extension  $NLSTV_1$ , we set  $K = 9$ . For NLTV we set  $K = 14$  (the 4 closest neighbors plus the best 10 non-local neighbors) as suggested in [42].

For the minimization of the objective functions that are related to the regularizers we are comparing, we employ an optimization strategy similar to the one described in Algorithm 1. Our rationale for using a common optimization framework for all the studied regularizers is that this way we can make sure that the convergence behavior is comparable in all cases. Consequently, the image reconstruction quality depends on the choice of the regularizer rather than on the efficiency of the employed minimization scheme.

#### A. Image denoising

Image denoising is the simplest form of inverse imaging problem where the system matrix  $\mathbf{A}$  reduces to the identity operator  $\mathbf{I}$ . In our scenario we consider i.i.d Gaussian noise at three different noise levels (low, medium, and high) that correspond to a standard deviation of  $\sigma_n = 0.05, 0.075, 0.1$ , respectively. Regarding the stopping criterion of the minimization algorithms, this is set to either reaching a relative normed difference of  $5 \cdot 10^{-5}$  between two successive image estimates, or a maximum of 150 iterations. In addition, for the non-local methods we need to compute the non-local weights. These weights according to the definitions we provided earlier depend on the underlying image. Here, however, we compute them from a smoothed version of the noisy image by a Gaussian filter. We have experimentally observed that the weights obtained in this way lead to better denoising results than the ones computed directly from the noisy image itself.

In Table I we report the grayscale and color denoising results for all test images and noise levels. The performance of all the methods is measured in terms of the PSNR improvement (ISNR) w.r.t the noisy image. By inspecting the grayscale results we observe that TV is the least performing method for all noise levels. NLM filtering leads to better results than TV by exploiting the non-local self-similarity property. However it performs worse than the semi-local  $STV_1$  regularizer. Additionally, NLTV which is inspired by NLM performs better than all these methods. Finally,  $NLSTV_1$  consistently outperforms all the tested methods and shows an improvement over its semi-local version of about 0.3 dBs on average. Similar observations are drawn from the color denoising results. However, in this case  $STV_1$  outperforms NLTV but not  $NLSTV_1$ .

Besides the quantitative comparisons, to allow for a visual assessment of the reconstruction performance, we provide in Fig. 3 representative grayscale and color image denoising results. From these results we observe that our non-local functional achieves a satisfactory denoising performance without introducing staircase artifacts, which are present in TV and NLTV reconstructions, or excessively smoothing important image structures, such as in the case of NLM. Moreover, due to its non-local nature  $NLSTV_1$  proves more efficient in removing the noise than its semi-local counterpart,  $STV_1$ .

#### B. Image deblurring

In the image deblurring setting we consider a circulant system matrix  $\mathbf{A}$  that models the point spread function (PSF) of the imaging device. We test all the methods for two blurring kernels, a Gaussian of support  $9 \times 9$  pixel with a standard deviation  $\sigma_d = 6$  and a motion kernel of support  $19 \times 19$  pixel<sup>2</sup>. As an additional degradation we consider three different levels of Gaussian noise which correspond to a blurred SNR (BSNR) of the input image of 20, 25, and 30 dBs, respectively. The BSNR is defined as  $BSNR = \text{var}(\mathbf{A}\mathbf{u}) / \sigma_n^2$ , where  $\text{var}(\mathbf{A}\mathbf{u})$  is the variance of the blurred input and  $\sigma_n$  is the standard deviation of the noise.

For this problem a typical strategy for computing the non-local weights is to minimize the objective function w.r.t to the underlying image and the weighting function [42]. This way the weights are updated in every iteration of the algorithm. In this case, however, we need to solve a non-convex problem and, thus, we lack any guarantees about reaching the global minimum. Here, we follow a different approach which leads to a convex optimization problem. Specifically, we estimate the non-local weights from a smooth version of the image that has been previously deblurred by a Wiener filter. Then, we use these fixed weights to minimize the objective function. Finally, the stopping criteria of the minimization algorithm remain the same with the ones used in the denoising problem.

In Table II we report the grayscale and color deblurring results for all test images, blurring kernels, and noise levels. Once again the performance of the methods under comparison is measured in terms of ISNR w.r.t to the blurred and noisy image. The conclusions that we can draw from these results

<sup>2</sup>This psf was obtained from <http://www.wisdom.weizmann.ac.il/~levina/papers/LevinEtalCVPR09Data.rar>

TABLE I  
ISNR COMPARISONS ON GRAYSCALE/COLOR IMAGE DENOISING

Method $\sigma$ (std.)	TV/VTV			STV <sub>1</sub>			NLM <sub>(5×5)</sub>			NLTV <sub>(5×5)</sub>			NLSTV <sub>1</sub> <sub>(7×7)</sub>		
	0.05	0.075	0.1	0.05	0.075	0.1	0.05	0.075	0.1	0.05	0.075	0.1	0.05	0.075	0.1
Img.1	5.82/6.50	7.28/7.98	8.42/8.98	6.25/6.73	7.70/8.20	8.82/9.33	5.79/6.02	7.35/7.64	8.58/8.95	6.30/6.77	7.86/8.21	9.08/9.30	<b>6.59/7.06</b>	<b>8.08/8.60</b>	<b>9.23/9.81</b>
Img.2	9.21/9.95	10.85/11.64	12.06/11.72	9.87/10.40	11.42/12.00	12.57/13.18	10.01/10.29	11.61/12.03	12.74/13.28	10.04/10.48	11.62/12.03	12.72/13.04	<b>10.19/10.60</b>	<b>11.76/12.25</b>	<b>12.89/13.44</b>
Img.3	5.22/5.86	6.41/7.08	7.34/7.96	5.90/6.34	7.07/7.54	7.99/8.46	5.08/5.09	6.66/6.75	7.89/8.06	5.93/6.05	7.18/7.12	8.13/7.93	<b>6.17/6.59</b>	<b>7.30/7.81</b>	<b>8.17/8.77</b>
Img.4	8.10/8.89	9.51/10.35	10.57/10.80	8.57/9.15	9.98/10.61	11.04/11.69	8.20/8.61	9.83/10.30	11.06/11.56	8.58/8.85	10.11/10.22	11.23/11.22	<b>8.84/9.42</b>	<b>10.23/10.94</b>	<b>11.28/12.10</b>
Img.5	6.01/6.65	7.13/7.93	8.01/8.72	7.00/7.41	8.10/8.64	8.92/9.55	6.79/6.84	8.18/8.42	9.20/9.57	7.06/7.09	8.28/8.24	9.18/8.97	<b>7.41/7.67</b>	<b>8.53/8.98</b>	<b>9.34/9.94</b>
Img.6	5.40/6.09	6.75/7.48	7.83/8.51	5.85/6.34	7.21/7.73	8.28/8.81	5.69/5.89	7.14/7.44	8.21/8.61	6.01/6.55	7.47/7.97	8.60/9.02	<b>6.30/6.80</b>	<b>7.71/8.28</b>	<b>8.79/9.42</b>
Img.7	4.12/4.76	5.43/6.13	6.51/7.24	4.39/4.85	5.72/6.22	6.80/7.32	3.99/4.23	5.44/5.72	6.56/6.91	4.42/4.82	5.88/6.26	7.03/7.35	<b>4.67/5.09</b>	<b>6.05/6.57</b>	<b>7.15/7.75</b>
Img.8	4.81/5.47	6.28/6.96	7.47/8.11	5.24/5.71	6.69/7.17	7.85/8.33	4.73/4.95	6.38/6.67	7.68/8.03	5.17/5.57	6.71/6.99	7.92/8.12	<b>5.44/5.82</b>	<b>6.88/7.31</b>	<b>8.02/8.53</b>
Img.9	3.32/3.92	4.65/5.31	5.73/6.42	3.66/4.12	5.02/5.50	6.10/6.60	3.10/3.35	4.69/5.03	5.98/6.36	3.56/3.91	5.02/5.37	6.18/6.49	<b>3.80/4.14</b>	<b>5.18/5.59</b>	<b>6.26/6.75</b>
Img.10	5.74/6.46	7.22/7.96	8.45/9.10	6.16/6.69	7.61/8.15	8.80/9.33	5.69/5.96	7.19/7.56	8.45/8.87	6.05/6.60	7.61/8.08	8.87/9.25	<b>6.45/6.93</b>	<b>7.89/8.40</b>	<b>9.05/9.61</b>
Img.11	3.83/4.51	5.07/5.80	6.11/6.86	4.19/4.70	5.45/5.98	6.48/7.02	3.83/4.10	5.15/5.50	6.27/6.68	4.16/4.67	5.46/5.95	6.53/6.95	<b>4.47/4.99</b>	<b>5.73/6.33</b>	<b>6.73/7.41</b>
Img.12	5.06/5.72	6.15/6.90	7.02/7.81	5.62/6.12	6.72/7.28	7.58/8.17	5.82/5.98	7.05/7.34	7.96/8.37	6.00/6.47	7.22/7.67	8.14/8.54	<b>6.13/6.57</b>	<b>7.25/7.80</b>	<b>8.09/8.75</b>
Avg.	5.55/6.23	6.89/7.63	7.96/8.52	6.06/6.55	7.39/7.92	8.44/8.98	5.73/5.94	7.22/7.53	8.38/8.77	6.11/6.49	7.53/7.84	8.63/8.85	<b>6.37/6.81</b>	<b>7.72/8.24</b>	<b>8.75/9.36</b>

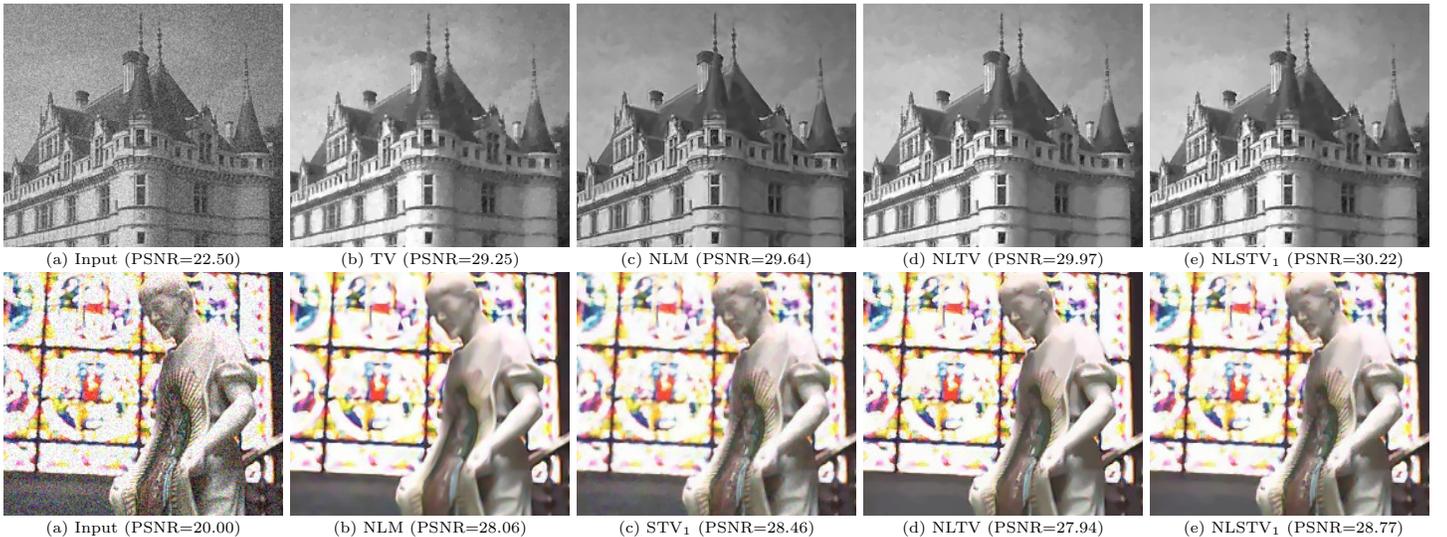


Fig. 3. Image denoising examples. Close-ups of noisy inputs and their corresponding denoised versions. Top row: grayscale denoising of input image with noise level  $\sigma_n = 0.075$ . Bottom row: color denoising of input image with noise level  $\sigma_n = 0.1$ .

are on par with those in the image denoising case. Indeed, TV is the least performing regularizer, NLTV and STV<sub>1</sub> perform comparably, while NLSTV<sub>1</sub> consistently outperforms all the other regularizers both on the grayscale and color images. Representative deblurring examples are shown in Fig. 4.

### C. Image reconstruction from sparse Fourier samples

In this section we examine the problem of image reconstruction from a limited number of Fourier measurements. In this case the system matrix is expressed as  $\mathbf{A} = \mathbf{M}\mathbf{F}$ , where  $\mathbf{F}$  represents the Fourier transform while  $\mathbf{M}$  is a masking operator that retains only a subset of the Fourier coefficients and discards the rest. For our comparisons we consider a mask consisting of 32 radial lines. This corresponds to retaining about 7% of the Fourier coefficients. The Fourier measurements are further corrupted by complex Gaussian noise at three different levels. These correspond to a SNR of the fully sampled image in the Fourier domain of 10, 20, and 30 dBs. The adopted forward model is closely related to the one encountered in magnetic resonance imaging (MRI). The main difference is that in our case the underlying image is real-valued rather than complex-valued. Similarly to the deblurring problem, we use fixed weights for the non-local regularizers.

These weights are computed from the back-projected image  $\mathbf{u}_b = \mathbf{F}^H \mathbf{M}^T \mathbf{v}$ . Finally, regarding the stopping criterion of the minimization algorithm, the number of maximum iterations is set to 200.

In Table III we provide the ISNR scores of all the methods under comparison on both grayscale and color images. Similarly to the previous two inverse problems, we observe that on average our NLSTV<sub>1</sub> regularizer leads to the best reconstruction performance. On the contrary, NLTV shows a different behavior and does not perform as well as it did in the denoising and deblurring tasks. In fact, the results indicate that it is the worst performing method. This might be due to the computation of the non-local weights from the back-projected images whose quality are rather poor compared to the ground-truth data. If this is the case, then NLSTV<sub>1</sub> is less sensitive in the choice of the non-local weights since it seems not to be affected as much as NLTV.

## VI. CONCLUSIONS

In this work we combined ideas from local and non-local regularization strategies and proposed a novel family of non-local functionals to regularize inverse imaging problems. Our

TABLE II  
ISNR COMPARISONS ON GRAYSCALE/COLOR DEBLURRING

Gaussian PSF												
Method	TV/VTV			STV <sub>1</sub>			NLTV <sub>(5×5)</sub>			NLSTV <sub>1(7×7)</sub>		
	20 dB	25 dB	30 dB	20 dB	25 dB	30 dB	20 dB	25 dB	30 dB	20 dB	25 dB	30 dB
BSNR												
Img.1	4.19/4.81	5.28/5.84	6.49/6.97	4.55/4.99	5.67/6.04	6.88/7.20	4.63/4.98	5.70/6.00	6.88/7.11	<b>4.75/5.34</b>	<b>5.91/6.43</b>	<b>7.13/7.52</b>
Img.2	5.00/5.78	5.67/6.31	6.69/7.21	5.29/5.81	6.01/6.39	7.04/7.32	5.52/6.15	6.21/6.67	7.16/7.54	<b>5.74/6.41</b>	<b>6.47/6.96</b>	<b>7.46/7.81</b>
Img.3	4.41/5.19	5.58/6.33	6.89/7.56	5.12/5.54	6.27/6.68	7.52/7.92	5.09/5.08	6.19/6.16	7.45/7.44	<b>5.19/5.64</b>	<b>6.35/6.81</b>	<b>7.65/8.12</b>
Img.4	4.65/4.81	5.75/5.82	6.83/6.86	5.02/5.00	6.07/6.00	7.06/6.97	4.90/4.81	6.12/6.00	<b>7.28/7.17</b>	<b>5.17/5.26</b>	<b>6.26/6.32</b>	<b>7.28/7.31</b>
Img.5	4.07/4.65	5.19/5.60	6.54/6.84	4.78/5.09	5.95/6.09	7.35/7.40	4.78/4.79	5.96/5.80	7.44/7.15	<b>4.94/5.14</b>	<b>6.16/6.19</b>	<b>7.65/7.58</b>
Img.6	3.14/3.60	4.00/4.37	5.07/5.37	3.41/3.72	4.35/4.56	5.43/5.61	3.41/3.61	4.38/4.46	5.52/5.60	<b>3.56/3.93</b>	<b>4.53/4.81</b>	<b>5.66/5.89</b>
Img.7	2.58/2.92	3.08/3.40	3.81/4.13	2.74/2.99	3.28/3.50	4.02/4.24	2.75/2.89	3.31/3.44	4.10/4.26	<b>2.81/3.09</b>	<b>3.38/3.67</b>	<b>4.18/4.44</b>
Img.8	3.62/4.10	4.39/4.84	5.33/5.74	3.93/4.21	4.68/4.95	5.59/5.85	3.92/4.05	4.67/4.80	5.59/5.76	<b>4.02/4.33</b>	<b>4.81/5.14</b>	<b>5.75/6.05</b>
Img.9	3.72/4.01	4.70/4.92	5.73/5.89	3.96/4.10	4.98/5.07	5.98/6.03	<b>4.20/4.18</b>	5.13/5.11	6.12/6.09	4.18/4.26	<b>5.22/5.25</b>	<b>6.23/6.22</b>
Img.10	3.16/3.64	3.49/3.90	4.17/4.56	3.42/3.77	3.73/4.01	4.42/4.67	3.50/3.78	3.77/3.93	4.40/4.53	<b>3.52/3.90</b>	<b>3.82/4.14</b>	<b>4.51/4.79</b>
Img.11	3.00/3.33	3.90/4.19	4.92/5.14	3.27/3.46	4.20/4.35	5.16/5.28	3.18/3.26	4.10/4.18	5.14/5.25	<b>3.34/3.51</b>	<b>4.28/4.44</b>	<b>5.29/5.44</b>
Img.12	3.08/3.62	4.11/4.60	5.32/5.74	3.48/3.88	4.60/4.94	5.82/6.10	3.63/3.96	4.79/5.07	6.12/6.38	<b>3.68/4.11</b>	<b>4.84/5.21</b>	<b>6.13/6.42</b>
Avg.	3.72/4.21	4.60/5.01	5.65/6.00	4.08/4.38	4.98/5.21	6.02/6.22	4.13/4.29	5.03/5.13	6.10/6.19	<b>4.24/4.58</b>	<b>5.17/5.45</b>	<b>6.24/6.47</b>

Motion PSF												
Method	TV/VTV			STV <sub>1</sub>			NLTV <sub>(5×5)</sub>			NLSTV <sub>1(7×7)</sub>		
	20 dB	25 dB	30 dB	20 dB	25 dB	30 dB	20 dB	25 dB	30 dB	20 dB	25 dB	30 dB
BSNR												
Img.1	6.11/6.74	7.93/8.43	10.18/10.55	6.54/6.95	8.36/8.66	10.61/10.81	6.59/7.04	8.40/8.74	10.59/10.89	<b>6.93/7.45</b>	<b>8.76/9.18</b>	<b>10.98/11.31</b>
Img.2	6.62/7.34	8.02/8.60	9.94/10.42	7.12/7.57	8.62/8.94	10.66/10.86	7.19/7.73	8.76/9.14	10.78/11.04	<b>7.56/8.02</b>	<b>9.04/9.37</b>	<b>11.00/11.21</b>
Img.3	6.12/6.93	8.24/9.05	10.77/11.55	6.83/7.36	9.03/9.56	11.60/12.12	6.83/6.98	9.03/9.16	11.58/11.79	<b>7.03/7.70</b>	<b>9.30/9.97</b>	<b>11.91/12.53</b>
Img.4	7.89/7.79	10.41/10.10	13.19/12.71	8.40/8.05	10.93/10.41	13.69/13.02	8.13/8.03	10.84/10.39	13.60/12.98	<b>8.76/8.57</b>	<b>11.29/10.92</b>	<b>14.00/13.50</b>
Img.5	6.10/6.38	8.34/8.29	10.99/10.68	6.93/6.90	9.30/8.93	12.06/11.45	7.05/6.84	9.46/8.87	12.14/11.35	<b>7.36/7.28</b>	<b>9.83/9.39</b>	<b>12.60/11.90</b>
Img.6	5.14/5.47	6.85/7.03	9.10/9.11	5.56/5.69	7.33/7.30	9.60/9.43	5.61/5.78	7.44/7.54	9.70/9.78	<b>5.88/6.10</b>	<b>7.73/7.83</b>	<b>10.03/10.03</b>
Img.7	4.78/5.33	6.58/7.10	8.93/9.39	5.09/5.44	6.89/7.21	9.22/9.50	5.17/5.46	7.00/7.31	9.26/9.58	<b>5.35/5.81</b>	<b>7.20/7.65</b>	<b>9.51/9.91</b>
Img.8	5.68/6.32	7.37/7.97	9.47/10.02	6.06/6.50	7.79/8.20	9.92/10.29	5.89/6.16	7.59/7.89	9.71/10.03	<b>6.19/6.61</b>	<b>7.95/8.33</b>	<b>10.10/10.43</b>
Img.9	6.15/6.42	8.35/8.55	10.95/11.06	6.50/6.58	8.71/8.74	11.31/11.29	6.53/6.69	8.68/8.78	11.19/11.20	<b>6.79/6.84</b>	<b>8.98/8.98</b>	<b>11.53/11.45</b>
Img.10	4.81/5.38	6.03/6.53	7.92/8.35	5.18/5.55	6.47/6.75	8.40/8.61	5.13/5.32	6.32/6.51	8.18/8.39	<b>5.32/5.68</b>	<b>6.62/6.91</b>	<b>8.56/8.79</b>
Img.11	5.67/6.01	7.99/8.20	10.71/10.82	6.03/6.16	8.36/8.39	11.07/11.03	5.82/6.04	8.17/8.32	10.89/10.97	<b>6.22/6.40</b>	<b>8.58/8.71</b>	<b>11.30/11.37</b>
Img.12	5.47/6.02	7.72/8.19	10.51/10.83	6.08/6.40	8.40/8.63	11.17/11.31	6.38/6.96	8.89/9.35	11.64/12.01	<b>6.59/7.00</b>	<b>9.03/9.37</b>	<b>11.84/12.08</b>
Avg.	5.88/6.34	7.82/8.17	10.22/10.46	6.36/6.60	8.35/8.48	10.78/10.81	6.36/6.59	8.38/8.50	10.77/10.83	<b>6.67/6.96</b>	<b>8.69/8.88</b>	<b>11.11/11.21</b>



Fig. 4. Image deblurring examples. Close-ups of blurred and noisy inputs and their corresponding deblurred versions. Top row: grayscale deblurring of input image degraded by motion blur and noise level of BSNR=20 dBs. Bottom row: color deblurring of input image degraded by Gaussian blur and noise level of BSNR=25 dBs.

non-local regularizers differ from the existing ones in the sense that they employ a non-local version of the structure tensor as the regularization operator. Therefore, they depend on the standard image gradient rather than the graph gradient. This way we are able to exploit both the local structural regularity and the non-local self-similarity properties of natural images. Further, we proposed an efficient minimization algorithm that

is based on a variable splitting strategy. Finally, we assessed the reconstruction performance of our regularizers on several inverse imaging problems. Our reconstruction results were shown to compare favorably to the ones obtained by other competing local and non-local regularization methods.

TABLE III  
ISNR COMPARISONS ON GRAYSCALE/COLOR SPARSE FOURIER RECONSTRUCTION

Method	TV/VTV				STV <sub>1</sub>			NLTV (5×5)				NLSTV <sub>1</sub> (7×7)				
	10 dB	20 dB	30 dB	30 dB	10 dB	20 dB	30 dB	10 dB	20 dB	30 dB	10 dB	20 dB	30 dB	10 dB	20 dB	30 dB
Img.1	2.39/2.79	2.98/3.35	3.78/3.98		2.53/2.86	3.29/3.56	4.16/4.31	2.32/2.77	2.88/3.52	3.55/4.24	<b>2.60/2.93</b>	<b>3.48/3.87</b>	<b>4.39/4.66</b>			
Img.2	3.58/4.47	3.85/4.67	5.09/5.81		3.86/4.56	4.09/4.74	5.43/5.99	3.48/4.34	3.36/4.36	4.27/5.28	<b>4.02/4.73</b>	<b>4.37/5.11</b>	<b>5.75/6.35</b>			
Img.3	1.82/2.19	2.58/2.91	3.07/3.32		<b>2.14/2.39</b>	2.92/3.15	3.42/3.61	1.68/2.18	2.43/2.99	2.83/3.44	2.14/2.34	<b>3.09/3.44</b>	<b>3.61/3.98</b>			
Img.4	2.58/3.26	4.01/4.41	4.76/4.91		2.88/3.42	4.29/4.62	5.03/5.18	2.24/3.20	3.19/4.21	3.66/4.67	<b>3.06/3.84</b>	<b>4.53/5.20</b>	<b>5.19/5.81</b>			
Img.5	2.67/3.37	3.69/4.45	4.23/5.07		3.08/3.70	4.21/4.89	4.88/5.64	2.60/3.35	3.70/4.68	4.29/5.42	<b>3.10/3.64</b>	<b>4.39/5.01</b>	<b>5.12/5.79</b>			
Img.6	1.68/2.14	2.07/2.46	2.43/2.78		1.82/2.22	2.27/2.62	2.68/3.02	1.61/2.07	1.92/2.39	2.24/2.78	<b>1.87/2.30</b>	<b>2.32/2.76</b>	<b>2.75/3.21</b>			
Img.7	1.48/1.82	1.95/2.21	2.34/2.46		<b>1.64/1.89</b>	2.16/2.34	2.59/2.66	1.33/1.65	1.86/2.09	2.29/2.44	1.63/1.94	<b>2.20/2.40</b>	<b>2.63/2.72</b>			
Img.8	2.10/2.40	2.69/2.86	3.00/3.08		2.30/2.52	2.92/3.03	3.26/3.30	2.02/2.22	2.51/2.74	2.78/3.00	<b>2.35/2.62</b>	<b>2.99/3.13</b>	<b>3.32/3.40</b>			
Img.9	1.56/1.72	1.91/1.92	2.04/2.02		1.69/1.78	2.06/2.03	2.22/2.16	1.46/1.60	1.80/1.85	1.96/1.99	<b>1.75/1.84</b>	<b>2.10/2.01</b>	<b>2.26/2.10</b>			
Img.10	2.67/3.16	2.87/3.41	3.67/4.04		<b>2.87/3.25</b>	3.10/3.51	3.92/4.21	2.66/2.98	2.84/3.30	3.59/4.04	2.83/3.28	<b>3.14/3.57</b>	<b>3.99/4.27</b>			
Img.11	1.05/1.34	1.53/1.69	1.80/1.85		<b>1.21/1.43</b>	<b>1.74/1.83</b>	<b>2.02/2.05</b>	0.90/1.19	1.37/1.60	1.64/1.82	1.16/1.47	1.73/1.87	2.01/2.09			
Img.12	1.02/1.48	1.45/1.88	1.76/2.14		1.23/1.61	1.76/2.11	2.10/2.44	1.04/1.56	1.55/2.13	1.85/2.45	<b>1.28/1.68</b>	<b>1.87/2.28</b>	<b>2.25/2.64</b>			
Avg.	2.05/2.51	2.63/3.02	3.16/3.46		2.27/2.64	2.90/3.20	3.48/3.71	1.94/2.43	2.45/2.99	2.91/3.46	<b>2.32/2.72</b>	<b>3.02/3.39</b>	<b>3.61/3.92</b>			

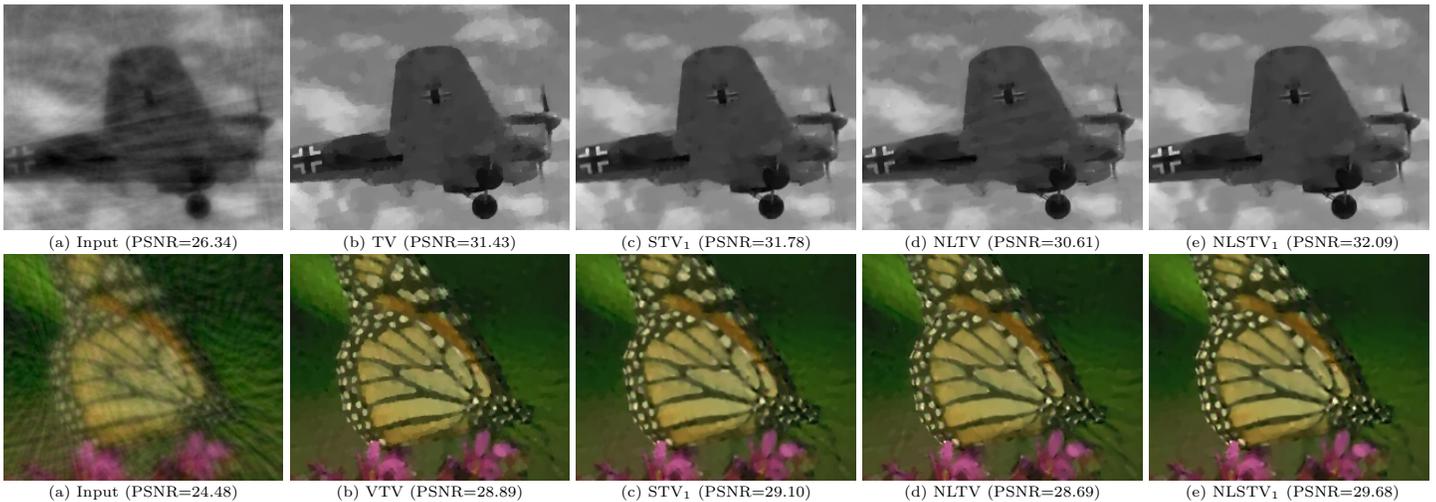


Fig. 5. Examples of image reconstruction from sparse Fourier measurements. Close-ups of compressed and noisy inputs and their corresponding reconstructed versions. Top row: grayscale input (back-projected) image sampled with a radial mask of 32 lines and at noise level of SNR=30 dBs. Bottom row: color input (back-projected) image sampled with a radial mask of 32 lines and at a noise level of SNR=20 dBs.

## REFERENCES

- [1] M. Bertero and P. Boccacci, *Introduction to Inverse Problems in Imaging*. IOP Publishing, 1998.
- [2] L. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D*, vol. 60, pp. 259–268, 1992.
- [3] T. Chan, A. Marquina, and P. Mulet, "High-order total variation-based image restoration," *SIAM J. Sci. Comput.*, vol. 22, pp. 503–516, 2000.
- [4] G. Gilboa, N. Sochen, and Y. Zeevi, "Variational denoising of partly textured images by spatially varying constraints," *IEEE Trans. Image Process.*, vol. 15, no. 8, pp. 2281–2289, 2006.
- [5] M. Grasmair, "Locally adaptive total variation regularization," in *Scale Space and Variational methods in Computer Vision*, 2009, pp. 331–342.
- [6] M. Grasmair and F. Lenzen, "Anisotropic total variation filtering," *Applied Mathematics & Optimization*, vol. 62, pp. 323–339, 2010.
- [7] S. Lefkimmatis, A. Roussos, P. Maragos, and M. Unser, "Structure tensor total variation," *SIAM J. Imaging Sci.*, 2015, in press.
- [8] K. Bredies, K. Kunisch, and T. Pock, "Total generalized variation," *SIAM J. Imaging Sci.*, vol. 3, pp. 492–526, 2010.
- [9] S. Lefkimmatis, J. Ward, and M. Unser, "Hessian Schatten-norm regularization for linear inverse problems," *IEEE Trans. Image Process.*, vol. 22, no. 5, pp. 1873–1888, 2013.
- [10] G. Gilboa and S. Osher, "Nonlocal operators with applications to image processing," *Multiscale Model. Simul.*, vol. 7, pp. 1005–1028, 2008.
- [11] D. Zhou and B. Schölkopf, "Regularization on discrete spaces," in *Pattern Recognition*. Springer, 2005, pp. 361–368.
- [12] A. Elmoataz, O. Lezoray, and S. Boughleux, "Nonlocal discrete regularization on weighted graphs: a framework for image and manifold processing," *IEEE Trans. Image Process.*, vol. 17, pp. 1047–1060, 2008.
- [13] M. Figueiredo, J. Bioucas-Dias, and R. Nowak, "Majorization-minimization algorithms for wavelet-based image restoration," *IEEE Trans. Image Process.*, vol. 16, pp. 2980–2991, 2007.
- [14] I. Selesnick and M. Figueiredo, "Signal restoration with overcomplete wavelet transforms: Comparison of analysis and synthesis priors," in *SPIE (Wavelets XIII)*, 2009.
- [15] H. Attouch, G. Buttazzo, and G. Michaille, *Variational Analysis in Sobolev and BV Spaces: Applications to PDEs and Optimization*, ser. MPS-SIAM Series on Optimization. Society for Industrial and Applied Mathematics, 2006.
- [16] P. Blomgren and T. Chan, "Color TV: Total Variation methods for restoration of vector-valued images," *IEEE Trans. Image Process.*, vol. 7, pp. 304–309, 1998.
- [17] B. Goldluecke, E. Strelakovsky, and D. Cremers, "The natural vectorial total variation which arises from geometric measure theory," *SIAM J. Imaging Sci.*, vol. 5, pp. 537–563, 2012.
- [18] W. Förstner and E. Gülch, "A fast operator for detection and precise location of distinct points, corners and centres of circular features," in *Proc. ISPRS Intercommission Conference on Fast Processing of Photogrammetric data*, 1987, pp. 281–305.
- [19] A. Buades, B. Coll, and J.-M. Morel, "A review of image denoising algorithms, with a new one," *Multiscale Model. Simul.*, vol. 4, pp. 490–530, 2005.
- [20] —, "Image denoising methods. A new nonlocal principle," *SIAM review*, vol. 52, pp. 113–147, 2010.
- [21] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *IEEE Int. Conf. Comput. Vision*, 1998, pp. 839–846.
- [22] S. Kindermann, S. Osher, and P. W. Jones, "Deblurring and denoising

- of images by nonlocal functionals,” *Multiscale Model. Simul.*, vol. 4, pp. 1091–1115, 2005.
- [23] G. Gilboa and S. Osher, “Nonlocal linear image regularization and supervised segmentation,” *Multiscale Model. Simul.*, vol. 6, pp. 595–630, 2007.
- [24] Q. Cheng, H. Shen, L. Zhang, and P. Li, “Inpainting for remotely sensed images with a multichannel nonlocal total variation model,” *IEEE Trans. Geoscience and Remote Sensing*, vol. 52, pp. 175–187, 2014.
- [25] G. Chierchia, N. Pustelnik, B. Pesquet-Popescu, and J.-C. Pesquet, *IEEE Trans. Image Processing*, vol. 23, pp. 5531–5544, 2014.
- [26] A. Chambolle, “An algorithm for total variation minimization and applications,” *J. Math. Imag. Vis.*, vol. 20, pp. 89–97, 2004.
- [27] J. Darbon, A. Cunha, T. Chan, S. Osher, and G. Jensen, “Fast nonlocal filtering applied to electron cryomicroscopy,” in *IEEE International Symposium on Biomedical Imaging*, 2008, pp. 1331–1334.
- [28] P. Viola and M. Jones, “Rapid object detection using a boosted cascade of simple features,” in *IEEE Int. Conf. Comput. Vision Patt. Recogn. (CVPR)*, vol. 1, 2001, pp. 511–518.
- [29] L. Condat, “A simple trick to speed up and improve the non-local means,” 2010. [Online]. Available: [http://www.gipsa-lab.grenoble-inp.fr/~laurent.condat/publis/condat\\_resreport\\_NLmeansv3.pdf](http://www.gipsa-lab.grenoble-inp.fr/~laurent.condat/publis/condat_resreport_NLmeansv3.pdf)
- [30] R. Bhatia, *Matrix Analysis*. Springer, 1997.
- [31] A. Chambolle and T. Pock, “A first-order primal-dual algorithm for convex problems with applications to imaging,” *J. Math. Imag. Vis.*, vol. 40, no. 1, pp. 120–145, 2011.
- [32] T. Goldstein, E. Esser, and R. Baraniuk, “Adaptive primal-dual hybrid gradient methods for saddle-point problems,” *arXiv preprint arXiv:1305.0546*, 2013.
- [33] D. Bertsekas, *Constrained optimization and Lagrange multiplier methods*. Athena Scientific, 1996.
- [34] J. Eckstein and D. Bertsekas, “On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators,” *Mathematical Programming*, vol. 55, no. 1, pp. 293–318, 1992.
- [35] E. Esser, “Applications of Lagrangian-based alternating direction methods and connections to split Bregman,” *CAM report*, vol. 9, 2009.
- [36] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, *Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers*. Now Publishers, 2011.
- [37] P. L. Combettes and V. R. Wajs, “Signal recovery by proximal forward-backward splitting,” *Multiscale Model. Simul.*, vol. 4, pp. 1168–1200, 2005.
- [38] S. Lefkimiatis and M. Unser, “Poisson image reconstruction with Hessian Schatten-norm regularization,” *IEEE Trans. Image Process.*, vol. 22, pp. 4314–4327, 2013.
- [39] J. Liu and J. Ye, “Efficient  $\ell_1/\ell_q$  norm regularization,” *Arxiv preprint arXiv:1009.4766*, 2010.
- [40] D. L. Donoho, “Denoising by soft-thresholding,” *IEEE Trans. Inf. Theory*, vol. 41, pp. 613–627, 1995.
- [41] J. R. Shewchuk, “An introduction to the conjugate gradient method without the agonizing pain,” 1994. [Online]. Available: <http://www.cs.cmu.edu/~jrs/jrspapers.html>
- [42] X. Zhang, M. Burger, X. Bresson, and S. Osher, “Bregmanized non-local regularization for deconvolution and sparse reconstruction,” *SIAM Journal on Imaging Sciences*, vol. 3, no. 3, pp. 253–276, 2010.



**Stamatis Lefkimiatis** (S’08-M’11) received his M.Sc degree in computer engineering and informatics (with highest honors) from the University of Patras, Patras, Greece in 2004 and his Ph.D. degree from the National Technical University of Athens (NTUA), Athens, Greece in 2009. From 2010 to 2014 he was a research scientist in the Biomedical Imaging Group at the Swiss Federal Institute of Technology, Lausanne (EPFL). Since May 2014, he is a SNSF (Swiss National Science Foundation) postdoctoral research fellow in the Department of

Mathematics at University of California, Los Angeles (UCLA). His main research interests lie in the areas of image analysis, statistical modeling, convex optimization, and inverse problems in imaging with focus on biomedical and computer vision applications.



**Stanley Osher** is a Professor of Mathematics, Computer Science, Chemical Engineering and Electrical Engineering at UCLA. He is also an Associate Director of the NSF-funded Institute for Pure and Applied Mathematics at UCLA. He received his MS and PhD degrees in Mathematics from the Courant Institute of NYU. Before joining the faculty at UCLA in 1977, he taught at SUNY Stony Brook, becoming professor in 1975. He has received numerous academic honors and co-founded three successful companies, each based largely on his own (joint)

research. Osher has been elected to the US National Academy of Science and the American Academy of Arts and Sciences. He was awarded the SIAM Pioneer Prize at the 2003 ICIAM conference and the Ralph E. Kleinman Prize in 2005. He was awarded honorary doctoral degrees by ENS Cachan, France, in 2006 and by Hong Kong Baptist University in 2009. He is a SIAM and AMS Fellow. He gave a one hour plenary address at the 2010 International Conference of Mathematicians. He also gave the John von Neumann Lecture at the SIAM 2013 annual meeting. He is a Thomson-Reuters highly cited researcher-among the top 1% from 2002-2012 in both Mathematics and Computer Science with an h index of 100. In 2014 he received the Carl Friedrich Gauss Prize from the International Mathematics Union-this is regarded as the highest prize in applied mathematics. His current interests involve information science, which includes optimization, image processing, compressed sensing and machine learning and applications of these techniques to the equations of physics, engineering and elsewhere.