NONLOCAL OPERATORS WITH APPLICATIONS TO IMAGE PROCESSING

GUY GILBOA * AND STANLEY OSHER †

Abstract. We propose the use of nonlocal operators to define new types of flows and functionals for image processing and elsewhere. A main advantage over classical PDE-based algorithms is the ability to handle better textures and repetitive structures. This topic can be viewed as an extension of spectral graph theory and the diffusion geometry framework to functional analysis and PDE-like evolutions. Some possible application and numerical examples are given, as is a general framework for approximating Hamilton-Jacobi equations on arbitrary grids in high demensions, e.g., for control theory.

Key words. Nonlocal operators, regularization, total variation, variational methods, spectral graph theory, Hamilton-Jacobi equations.

AMS subject classifications. 35A15, 68U10, 70H20, 65D25, 35S05, 68R10

1. Introduction.

1.1. Motivation. In this paper our goal is to formalize a systematic and coherent framework for nonlocal image and signal processing. By this we mean that any point can interact directly with any other point in the image domain (at least in principle). In practice, for complexity reasons, the number of interactions is limited only to the "most relevant" regions (in some sense which is derived from the application). Our formulation is continuous.

We attempt to extend some known PDE's and variational techniques to this nonlocal framework. The major difference is that classical derivatives are local operators. However, following ideas from graph theory, and specifically the gradient and divergence operators of Zhou and Schölkopf [58, 59], we observe that many PDE-based processes, minimizations and computation methods can be generalized to be nonlocal. A main advantage for image processing is the ability to process both structures (geometrical parts) and textures within the same framework.

We also believe this framework may be useful beyond the scope of image processing, for purposes such as physical modelling of processes with nonlocal behavior. We outline a method for approximating Hamilton-Jacobi equations in high dimensions in Section 3, below.

1.2. Short Background. PDE's have been used very successfully for many image processing tasks, such as denoising, deconvolution, segmentation, inpainting, optical-flow and more. For details regarding the theory and the applications see [1, 43, 42, 17, 50] and the references therein.

Techniques using spectral graph theory [19, 38] were used for image segmentation [51, 46, 56, 29] and in a more general form for various machine-learning applications in the diffusion geometry framework [20, 40]. These techniques are based on manipulation of the eigenvalues of the graph-Laplacian. Total variation type regularizations on graphs were first proposed in [15] and later by [59] and [7]. A related framework

*Department of Mathematics, UCLA, Los Angeles, California 90095, tel. (310)-8254952 fax. (310)-2066673 gilboa@math.ucla.edu.

†Department of Mathematics, UCLA, Los Angeles, California 90095, tel. (310)-8251758 fax. (310)-2066673 sjo@math.ucla.edu. Both authors are supported by grants from the NSF under contracts ITR ACI-0321917, DMS-0312222, and the NIH under contract P20 MH65166. G.G. is also supported by NSF DMS-0714087.

in the context of PDE's is the Beltrami flow on Riemannian manifolds [52, 31] where the metric is image-driven and textures can be handled [49]. This framework however is still local and is based on PDE's in a classical sense.

For image denoising, nonlocal methods were developed based on gray-level pixel affinities in the form of the Yoroslavsky filter [57] and the bilateral filters [54]. Deeper understanding of these filters and their relation to PDE's were given by Barash and Elad [5, 6, 26]. Nonlocal denoising based on patch-distances was proposed by Buades et al in [9]. They have also given in [10] the asymptotic relation of neighborhood filters to Perona-Malik type PDE's [47]. The use of patch distances in [9] followed ideas by Efros and Leung [25] for texture synthesis and completion. We will give a variational interpretation of this process in this paper. In [53] the filter of [9], referred to as nonlocal means, was understood as a special case within the diffusion geometry framework. Other patch distances based on filters were proposed. In [36] a fast algorithm was designed for computing the fully nonlocal version. The study of [30] presented a statistical analysis of the problem and suggested to use an adaptive window approach which minimizes a local risk measure.

The DUDE algorithm [39] denoises data sequences generated by a discrete source and received over a discrete memoryless channel. DUDE assigns image values using similarity of neighborhoods based on image statistics. This resembles the construction of conditional probabilities in Awate and Whitaker [4]. The DUDE approach is limited to discrete-valued signals as opposed to [4] and our approach, which addresses continuous- valued signals, such as those associated with grayscale images. The DUDE algorithm is not very effective in case of additive noise.

Awate and Whitaker's algorithm [4] can be expressed in our framework (without our PDE/regularization steps). They use the entropy as a measure of self similarity and obtain a convolution with weights requiring a convolution. They update their weights as they proceed in time using their gradient descent approach. Their method somewhat resembles the approach in [32], Section 3, where g, the function within the regularizer, involves entropy.

A first variational understanding as a nonconvex minimization was given in [32]. In [28] we proposed an alternative convex quadratic functional, showed the relation to spectral graph theory, and were able to achieve superior filtering properties, compared with [9], using an iterative "nonlocal diffusion" process. We also presented a simple nonlocal supervised segmentation algorithm which follows [51, 35, 29] and analyzed analytically the step-edge case. In [27] a more general convex framework was proposed and a method to compute the energy minimizations using graph-cut techniques was shown. This paper follows and significantly generalizes our previous studies [28, 27].

2. The proposed mathematical framework.

2.1. Basic operators. In the following we use a variant of the gradient and divergence definitions on graphs given in the context of machine learning [58, 59]. In our case, the weights are not normalized pointwise and the definitions are continuous. Recently Bougleux et al. [7] have proposed a regularization framework on graphs which also uses similar operators. In their study, a family of $p-$Laplace operators was defined for discrete data and a variational framework was proposed for image and mesh denoising.

Let $\Omega \subset \mathbb{R}^n$, $x \in \Omega$, $u(x)$ a real function $u : \Omega \to \mathbb{R}$. We extend the notion of

derivatives to a nonlocal framework by the following definition:

$$\partial_y u(x) := \frac{u(y) - u(x)}{\tilde{d}(x,y)}, \quad y, x \in \Omega,$$

where $0 < \tilde{d}(x,y) \leq \infty$ is a positive measure defined between points x and y. To keep with standard notations related to graphs we define the weights as

$$w(x,y) = \tilde{d}^{-2}(x,y).$$

Thus $0 \leq w(x,y) < \infty$. In this paper we assume the weights are symmetric, that is $w(x,y) = w(y,x)$. The nonlocal derivative can be written as

$$\partial_y u(x) := (u(y) - u(x))\sqrt{w(x,y)}. \tag{2.1}$$

The nonlocal gradient $\nabla_w u(x) : \Omega \to \Omega \times \Omega$ is defined as the vector of all partial derivatives:

$$(\nabla_w u)(x,y) := (u(y) - u(x))\sqrt{w(x,y)}, \quad x, y \in \Omega. \tag{2.2}$$

We denote vectors as $\vec{v} = v(x,y) \in \Omega \times \Omega$. The standard L^2 inner product is used for functions

$$\langle u_1, u_2 \rangle := \int_\Omega u_1(x) u_2(x) dx.$$

For vectors we define a dot product

$$(\vec{v_1} \cdot \vec{v_2})(x) := \int_\Omega v_1(x,y) v_2(x,y) dy,$$

and an inner product

$$\langle \vec{v}_1, \vec{v}_2 \rangle := \langle \vec{v}_1 \cdot \vec{v}_2, 1 \rangle = \int_{\Omega \times \Omega} v_1(x,y) v_2(x,y) dx dy.$$

The magnitude of a vector is

$$|\vec{v}|(x) := \sqrt{\vec{v_1} \cdot \vec{v_1}} = \sqrt{\int_\Omega v(x,y)^2 dy}.$$

With the above inner products the nonlocal divergence $\mathrm{div}_w \vec{v}(x) : \Omega \times \Omega \to \Omega$ is defined as the adjoint of the nonlocal gradient:

$$(\mathrm{div}_w \vec{v})(x) := \int_\Omega (v(x,y) - v(y,x))\sqrt{w(x,y)} dy. \tag{2.3}$$

The Laplacian can now be defined by:

$$\Delta_w u(x) := \frac{1}{2} \mathrm{div}_w (\nabla_w u(x)) = \int_\Omega (u(y) - u(x)) w(x,y) dy. \tag{2.4}$$

Note that in order to get the standard Laplacian definition which relates to the graph Laplacian we need a factor of $1/2$.

2.2. Some properties. Most of the properties involving a double integral can be shown by expanding an integral of the form $\int_{\Omega\times\Omega} f(x,y)dxdy$ to $\frac{1}{2}\int_{\Omega\times\Omega}(f(x,y)+f(y,x))dxdy$, changing the order of integration and using the fact that $w(x,y) = w(y,x)$. We give an example showing the adjoint relation

$$\langle \nabla_w u, \vec{v}\rangle = \langle u, -\mathrm{div}_w \vec{v}\rangle, \tag{2.5}$$

$$\begin{aligned}
\langle \nabla_w u, \vec{v}\rangle &= \int_{\Omega\times\Omega}(u(y)-u(x))\sqrt{w(x,y)}v(x,y)dxdy \\
&= \tfrac{1}{2}\int_{\Omega\times\Omega}\left[(u(y)-u(x))\sqrt{w(x,y)}v(x,y)+(u(x)-u(y))\sqrt{w(y,x)}v(y,x)\right]dxdy \\
&= \tfrac{1}{2}\int_{\Omega\times\Omega}\left[u(y)(v(x,y)-v(y,x))-u(x)(v(x,y)-v(y,x))\right]\sqrt{w(x,y)}dxdy \\
&= \tfrac{1}{2}\int_{\Omega\times\Omega}\left[u(x)(v(y,x)-v(x,y))-u(x)(v(x,y)-v(y,x))\right]\sqrt{w(x,y)}dxdy \\
&= \int_\Omega u(x)\left(-\int_\Omega (v(x,y)-v(y,x))\sqrt{w(x,y)}dy\right)dx.
\end{aligned}$$

"Divergence theorem":

$$\int_\Omega \mathrm{div}_w \vec{v}dx = 0. \tag{2.6}$$

The Laplacian is self adjoint

$$\langle \Delta_w u, u\rangle = \langle u, \Delta_w u\rangle \tag{2.7}$$

and negative semidefinite

$$\langle \Delta_w u, u\rangle = -\langle \nabla_w u, \nabla_w u\rangle \le 0. \tag{2.8}$$

We can also formulate a nonlocal (mean) curvature:

$$\begin{aligned}
\kappa_w &:= \mathrm{div}_w\left(\frac{\nabla_w u}{|\nabla_w u|}\right) \\
&= \int_\Omega (u(y)-u(x))w(x,y)\left(\frac{1}{|\nabla_w u|(x)}+\frac{1}{|\nabla_w u|(y)}\right)dy,
\end{aligned} \tag{2.9}$$

where

$$|\nabla_w u|(q) := \sqrt{\int_\Omega (u(z)-u(q))^2 w(q,z)dz}.$$

2.3. The Regularizing Functionals. Below we propose two types of regularizing nonlocal functionals. The first type is based on the nonlocal gradient. It is set within the mathematical framework described above. The second type is based on differences, it appears to be easier to implement, where the minimization can be accomplished using graph cut techniques, as will be discussed in Section 5. We still investigate the relations between these functionals and when each of them is preferred.

The gradient-based functional is

$$\begin{aligned}
J(u) &= \int_\Omega \phi(|\nabla_w u|^2)dx, \\
&= \int_\Omega \phi(\int_\Omega (u(y)-u(x))^2 w(x,y)dy)dx,
\end{aligned} \tag{2.10}$$

where $\phi(s)$ is a positive function, convex in $\sqrt{s}$ with $\phi(0)=0$.

The difference-based functional is

$$J_a(u) = \int_{\Omega\times\Omega} \phi((u(y)-u(x))^2 w(x,y))dydx. \tag{2.11}$$

The variation with respect to u (Euler-Lagrange) of (2.10) is

$$\partial_u J(u) = -2\int_{\Omega} (u(y)-u(x))w(x,y)(\phi'(|\nabla_w u|^2(x)) + \phi'(|\nabla_w u|^2(y)))dy, \tag{2.12}$$

where $\phi'(s)$ is the derivative of ϕ with respect to s. This can be written more concisely as

$$\partial_u J(u) = -2\mathrm{div}_w\left(\nabla_w u \phi'(|\nabla_w u|^2(x))\right).$$

The variation with respect to u of (2.11) is

$$\partial_u J_a(u) = -4\int_{\Omega} (u(y)-u(x))w(x,y)\phi'((u(y)-u(x))^2 w(x,y))dy. \tag{2.13}$$

Note that for the quadratic case $\phi(s) = s$ the functionals (2.10) and (2.11) coincide (and naturally so do Eqs. (2.12) and (2.13)).

2.3.1. Relation to isotropic and anisotropic local functionals. The functionals which can be written in the form of Eq. (2.10) correspond in the local case to *isotropic* functionals (which have no preferred directionality). The second category, Eq. (2.11), can be related to *anisotropic* functionals in the local case. We suggest later two different methods for efficiently computing each category.

As an example, for total-variation, $\phi(s) = \sqrt{s}$, Eq. (2.10) becomes:

$$J_{NL-TV}(u) = \int_{\Omega} |\nabla_w u|dx = \int_{\Omega} \sqrt{\int_{\Omega} (u(y)-u(x))^2 w(x,y)dy}dx \tag{2.14}$$

whereas Eq. (2.11) becomes

$$J_{NL-TVa}(u) = \int_{\Omega\times\Omega} |u(x)-u(y)|\sqrt{w(x,y)}dydx \tag{2.15}$$

The above functionals correspond in the local two dimensional case to the isotropic TV

$$J_{TV}(u) = \int_{\Omega} |\nabla u|dx = \int_{\Omega} \sqrt{u_{x_1}^2 + u_{x_2}^2}dx$$

and to the anisotropic TV

$$J_{TVa}(u) = \int_{\Omega} (|u_{x_1}| + |u_{x_2}|)dx.$$

Following the discussion in Section 3, another analogue to anisotropic TV is:

$$\int_{\Omega} \sum_{i=1}^{2} \left| \int_{\Omega} (u(y)-u(x))w(x,y)(y_i - x_i)dy \right| dx.$$

3. Computing Hamilton-Jacobi Equations on Arbitrary Grids in High Dimensions. In this section we show how this general framework can be used for computational purposes. Our ultimate goal here is to solve partial differential equations approximately in high dimensions on irregular grids. We assume to be operating on a set of isolated data points $\Omega_d \subset \mathbb{R}^n$, with n large and Ω_d sparse. Note that unlike the nonlocal models of the next section, here the construction of the weights is based on different considerations and is not image or signal driven. The calculus however is similar. A detailed study on the computational aspects with examples will appear elsewhere.

Using the nonlocal gradient defined in (2.2) enables us to obtain partial derivatives as follows. We wish to compute partial derivatives of u, i.e.

$$\left(\frac{\partial u}{\partial x_i}\right)_w, \ i = 1, \ldots, n$$

in a consistent way.

Let us first define an approximation of the unit vector in the x_i direction as follows:

$$(\hat{x}_i)_w := \nabla_w(x_i). \tag{3.1}$$

The corresponding partial derivative estimation is therefore:

$$\begin{aligned}\left(\frac{\partial u}{\partial x_i}\right)_w &:= \nabla_w(u) \cdot (\hat{x}_i)_w = \nabla_w(u) \cdot \nabla_w(x_i) \\ &= \textstyle\int_\Omega (u(y) - u(x)) w(x,y)(y_i - x_i) dy,\end{aligned} \tag{3.2}$$

we remind that dy is shorthand for $dy_1 \ldots dy_n$. Note that this can be generalized to any order, e.g. second order derivatives can be estimated by

$$\left(\frac{\partial^2 u}{\partial x_i^2}\right)_w := \nabla_w\big(\nabla_w(u) \cdot (\hat{x}_i)_w\big) \cdot (\hat{x}_i)_w.$$

We construct w such that the unit vectors are orthonormal:

$$(\hat{x}_i)_w \cdot (\hat{x}_j)_w = \delta_{ij},$$

that is:

$$\begin{aligned}\int (y_j - x_j) w(x,y)(y_i - x_i) dy = \delta_{ij} &= 1 \text{ if } i = j \\ &= 0 \text{ if } i \neq j.\end{aligned} \tag{3.3}$$

A similar framework for approximating partial derivatives for the purpose of strain localization was found by Chen, Zhang and Belytschko in [18]. Our construction of monotone schemes for Hamilton-Jacobi equations in high dimensions is new.

A simple and important class is

$$w(x,y) = w(|x - y|) = w(r) \tag{3.4}$$

normalized so that

$$\int_\Omega r^2 w(r) dx = n \tag{3.5}$$

e.g., if $n = 2$, we need

$$\int r^3 w(r) dr = \frac{1}{\pi}. \tag{3.6}$$

We can take $w(r) = c/r^2,\ 0 \le r \le R,\ c > 0 = 0\ R < r$ with $R = \sqrt{\frac{2}{\pi c}}$.

Another possibility is

$$w(\Omega) = Ce^{-r^2/\sigma},\ C,\ \sigma > 0.$$

For (3.5) we need $\sigma = \sqrt{\frac{2}{\pi C}}$, e.g., for $n = 2$.

We compute

$$\begin{aligned}\left(\frac{\partial u}{\partial x_i}\right)_w &= \int (u(y) - u(x)) w(x,y)(y_i - x_i) dy \qquad (3.7)\\ &= u_{x_i}(x) + \frac{1}{2}\sum_{j,k=1}^{n} \int u_{x_j,x_k} w(x,y)(y_j - x_j)(y_k - x_k)(y_i - x_i) dy\\ &\quad + \cdots\\ &= u_{x_i}(x) + \text{error}.\end{aligned}$$

In future work we will estimate the error term and develop a theory for solving Hamilton-Jacobi equations in high dimensions using relatively few data points. Such problems arise in control theory and elsewhere. Radial basis functions were used in [11] to obtain schemes in up to four dimensions. Our present approach seems to be more flexible. We outline it below. See [45], [44] for classical approaches.

In our framework: We are interested in solving

$$\begin{aligned} u_t + H(u_{x_1}, \ldots, u_{x_n}) &= 0 \qquad (3.8)\\ u(x,0) &= \varphi(x).\end{aligned}$$

We are interested in finding the unique viscosity solution [21]. We approximate this by discretization in time, for $x \in \Omega$

$$\begin{aligned}\frac{u^{m+1}(x) - u^m(x)}{\Delta t} &= -\tilde{H}\left(\left(\frac{\partial u^m}{\partial x_1}\right)_w, \left(\frac{\partial u^m}{\partial x_2}\right)_w, \ldots \left(\frac{\partial u^m}{\partial x_n}\right)_w\right) \qquad (3.9)\\ u^0(x) &= \varphi(x)\end{aligned}$$

where $\tilde{H}$ is the numerical Hamilton which is consistent with H (definitions will be given in a future paper)

$$u^m(x) \approx u(x, m\Delta t).$$

A scheme is monotone if $u^{m+1}(x)$ is a nondecreasing function of the values $u^m(x)$. We will take an analogue of the Lax-Friedrichs scheme [45]

$$\begin{aligned}u^{m+1}(x) = u(x) &- \Delta t H\left(\left(\frac{\partial u}{\partial x_1}\right)_w, \ldots, \left(\frac{\partial u}{\partial x_n}\right)_w\right) \qquad (3.10)\\ &+ 2\Delta t \int c(x,y) w(x,y)(u(y) - u(x)) dy\end{aligned}$$

(dropping the superscript m), where $c(x,y)$ is a nonnegative smooth function, chosen so that (3.10) gives us a consistent, monotone approximation to the Hamilton-Jacobi equation, (3.8) (for precise definitions, see [45]).

For this to be monotone, we first require

$$2c(x,y)w(x,y) - \sum_{\nu} |H_\nu| w(x,y) |y_\nu - x_\nu| > 0. \tag{3.11}$$

So we can take on the support of $w(x,y)$:

$$2c(x,y) > \sum_{\nu} |H_\nu| |y_\nu - x_\nu| \tag{3.12}$$

and for consistency $c(x, x+h) \to 0$ as $h \to 0$.

Also, we have a time step restriction:

$$1 + \Delta t \int \left(\sum_{\nu} H_\nu w(x,y)(y_k - x_v) - 2c(x,y)w(x,y) \right) dy \geq 0 \tag{3.13}$$

$$1 \geq \Delta t \int (w(x,y)2c(x,y) - \sum_{\nu} H_\nu w(x,y)(y_\nu - x_\nu)) dy$$

so we can take:

$$1 \geq 4\Delta t w(x,y) c(x,y).$$

Just to illustrate how this becomes rather conventional in a simple case, let $n = 1$ and

$$w(x,y) = \frac{1}{2h^2}(\delta(x - y - h) + \delta(x - y + h)), \delta \text{ the Dirac delta function.} \tag{3.14}$$

Then

$$\left(\frac{\partial u}{\partial x} \right)_w = \frac{u(x+h) - u(x-h)}{2h}$$

(unsurprisingly) and

$$u^{m+1}(x) = u(x) - \Delta t H \left(\frac{u(x+h) - u(x-h)}{2h} \right) \tag{3.15}$$
$$+ \Delta t \frac{(c(x,x+h)(u(x+h) - u(x)) - c(x,x-h)(u(x) - u(x-h))}{h^2}.$$

If we take $c(x,y) = K|y - x|$ for $K > 0$ large enough to satisfy (3.12), we have the conventional Lax-Friedrichs scheme, which is known to converge as $h \to 0$, if (3.13) is satisfied.

4. Basic Models. Our proposed nonlocal models are based on the general functional (2.10). The quadratic case, $\phi(s) = s$ (resulting in a linear steepest descent), was investigated in [28] where applications for denoising and segmentation were shown. Here we will focus on functionals with a TV-type regularizer.

We are interested in the minimizations of the following functionals:

Nonlocal ROF:

$$J_{NL-TV}(u) + \lambda \|f - u\|^2_{L^2}, \tag{4.1}$$

where $J_{NL-TV}(u)$ is defined in (2.14), f is the noisy input image or signal, and the minimization is over u. We are also interested in the inpainting version of this functional, following the local TV-inpainting model of [16]:

$$J_{NL-TV}(u) + \int_\Omega \lambda(x)(f-u)^2 dx, \tag{4.2}$$

with $\lambda(x) = 0$ in the inpainting region and $\lambda(x) = c$ in the rest of the image.

Another very important model following [41, 14] is the extension of $TV - L^1$ to a nonlocal version:

$$J_{NL-TV}(u) + \lambda \|f-u\|_{L^1}. \tag{4.3}$$

We will show later an interesting application of texture regularization using this minimization. It can both detect and remove anomalies or irregularities from images, and specifically textures.

We can further generalize Meyer's G-norm [37] to a nonlocal setting as described below.

4.1. Generalizing Meyer's G-norm. Let us define the nonlocal G space (the dual space of nonlocal TV). In the local case this space was considered by Meyer as the natural space of oscillatory patterns [37]. In our case oscillatory patterns which are regular and repetitive can be included in the nonlocal TV space, if a proper method for calculating the weights is used (as seen in our numerical examples). Thus we anticipate that in this case the nonlocal G space will characterize irregularities and randomness of the signal (and also noise). We have not yet investigated this topic thoroughly.

Let us define the nonlocal G space by

$$G = \{v \in X \ / \ \exists g \in Y \text{ such that } v = \operatorname{div}_w(g)\}. \tag{4.4}$$

The nonlocal G norm (if $v \in G$) is:

$$\|v\|_{NL-G} = \inf\{\|g\|_\infty \ / \ v = \operatorname{div}_w(g)\} \tag{4.5}$$

where $\|g\|_\infty := \sup_x\{|g|(x)\}$.

We can thus choose to minimize the following alternative to NL-ROF (4.1). We shall refer to it as NL TV-G:

$$J_{NL-TV}(u) + \lambda \|f-u\|_{NL-G}. \tag{4.6}$$

In the experimental section, some examples are given, showing the qualitative characteristics of this regularization.

4.2. Computing the weights. In our examples below we have a simplified scheme to compute the weights using only binary values (0 or 1) based on smallest patches distances.

Let us define the patch distance as in [9]:

$$d_a(f(x), f(y)) = \int_\Omega G_a(t)|f(x+t) - f(y+t)|^2 dt,$$

where G_a is a Gaussian of standard deviation a.

For each point we define the following set A of area $|A| = \gamma$ (a parameter) within a search neighborhood $S(x)$ around x (where $A \subset S(x) \subseteq \Omega$, $|A| \ll |S(x)|$):

$$A(x) := \arg\min_{\mathcal{A}} \left\{ \int_{\mathcal{A}} d_a(f(x), f(y))dy, \text{ s.t. } \mathcal{A} \subset S(x), |\mathcal{A}| = \gamma \right\}. \tag{4.7}$$

Then the weights are computed as:

$$w(x,y) = \begin{cases} 1, & \text{if } y \in A(x) \text{ or } x \in A(y) \\ 0, & \text{otherwise.} \end{cases} \tag{4.8}$$

This naturally gives the property of symmetric weights $w(x,y) = w(y,x)$. For the way to discretize the weights see Section 5.1 below. Note that in the following section we keep with the general case of real-valued non-negative weights, and do not assume that the values are binary.

5. Computation.

5.1. Basic Discretization. Let u_i denote the value of a pixel i in the image ($1 \leq i \leq N$), $w_{i,j}$ is the sparsely discrete version of $w(x,y)$. We use the neighbors set notation $j \in \mathcal{N}_i$ defined as $j \in \mathcal{N}_i := \{j : w_{i,j} > 0\}$.

Let ∇_{wd} be the discretization of ∇_w:

$$\nabla_{wd}(u_i) := (u_j - u_i)\sqrt{w_{i,j}}, \quad j \in \mathcal{N}_i \tag{5.1}$$

Let div_{wd} be the discretization of div_w:

$$\text{div}_{wd}(p_{i,j}) := \sum_{j \in \mathcal{N}_i} (p_{i,j} - p_{j,i})\sqrt{w_{i,j}}. \tag{5.2}$$

The discrete inner product for functions is $< u, v >:= \sum_i (u_i v_i)$ and for vectors we have the discretized dot product $(p \cdot q)_i := \sum_j (p_{i,j} q_{i,j})$ and inner product $< p, q >:= \sum_i \sum_j (p_{i,j} q_{i,j})$. The vector magnitude is therefore $|p|_i := \sqrt{\sum_j (p_{i,j})^2}$.

Binary weights:. We use binary weight values of 0 or 1. This way rare features which also have a very large "patch distance" between them and any other patch in the image can be regularized as well. In the more common case where the weights are computed with a Gaussian-like formula, e.g. as in [9],[28],[53],[30], the weights decay fast for distances above a certain threshold (usually related to the noise variance). This results in very weak connections (low weight values) for singular regions, thus such regions are essentially isolated from the rest of the image. This may be a good property in the case of denoising which avoids blurring of singular patches. However it is not adequate for the applications presented here, where the purpose is to remove irregularities. Note also that with binary weights the "manifold", as defined by the values $u(x)$ and the "metric" $w(x,y)$, is not necessarily smooth and can contain discontinuities or edges (which are handled well by the nonlocal TV regularizer).

The weights are descretized as follows: we take a patch around a pixel i, compute the distances $(d_a)_{i,j}$ (a discretization of $d_a(x,y)$) to all the patches in the search window and select the k closest (with the lowest distance value). The number of neighbors k is an integer proportional to the area γ. For each selected neighbor j we assign the value 1 to $w_{i,j}$ and to $w_{j,i}$. A maximum of up to $m = 2k$ neighbors for each pixel is allowed in our implementation. In the examples of Figs. 6.4 - 6.8 we used 5×5 pixel patches, a search window of size 21×21 and $m = 10$.

5.2. Steepest Descent. In this convex framework, one can resort as usual to a steepest descent method for computing the solutions. One initializes u at $t = 0$, e.g. with the input image: $u|_{t=0} = f$, and evolves numerically the flow:

$$u_t = -\partial_u J_d - \partial_u H_d(f, u),$$

where $\partial_u J_d$ is the discretized version of Eq. (2.12) or Eq. (2.13) and $H_d(f, u)$ is the discretized fidelity term functional. As in the local case, here also one should introduce a regularized version of the total variation: $\phi(s) = \sqrt{s + \epsilon^2}$ (where s is the square gradient magnitude). Thus the E-L equations are well defined, also for a zero gradient. When the L^1 norm is used as fidelity, a similar regularization is needed for that term also. The time-step restriction (CFL) is proportional to the regularizing ϵ and thus convergence is slow.

5.3. Graph-Cuts. For the difference-based ("anisotropic") functional, Eq. (2.11), we can generalize known fast algorithms which use graph-cuts techniques [8, 34]. A generalization of the algorithm of Darbon and Sigelle [23, 22] can be seen in our CAM report with Darbon and Chan [27].

5.4. Projections. For the gradient-based ("isotropic") case graph-cuts techniques cannot be generalized in a straightforward manner. They are restricted to pairwise interactions between nodes of the graph. When minimizing expressions involving the nonlocal gradient, however, this restriction is not met and one has to resort to an alternative method.

The projection algorithm of Chambolle [12] generalizes easily in this case. We show below how to compute the nonlocal ROF and a good approximation of nonlocal $TV - L^1$.

5.4.1. Nonlocal ROF. Chambolle's projection algorithm [12] for solving ROF [48] can be extended to solve nonlocal ROF.

A minimizer for the discrete version of Eq. (4.1) can be computed by the following iterations (fixed point method):

$$p_{i,j}^{n+1} = \frac{p_{i,j}^n + \tau(\nabla_{wd}(\mathrm{div}_{wd}(p^n) - 2\lambda f))_{i,j}}{1 + \tau|(\nabla_{wd}(\mathrm{div}_{wd}(p^n) - 2\lambda f))_{i,j}|} \tag{5.3}$$

where $p^0 = 0$, and the operators ∇_{wd} and div_{wd} are defined in (5.1) and (5.2), repectively. The solution is $u = f - \frac{1}{2\lambda}\mathrm{div}_{wd}(p)$.

THEOREM 5.1. *The algorithm converges to the global minimizer as $n \to \infty$ for any $0 < \tau \leq \frac{1}{\|\mathrm{div}_{wd}\|^2_{L^2}}$.*

Proof. The proof follows the same lines as the original proof of [12]. One should replace the numerical gradient, divergence and dot product defined in [12] by the nonlocal discrete definitions given here (Equations (5.2), (5.1) and the definitions which follow). Then everything follows in a straightforward manner: Obviously, the nonlocal TV, Eq. (2.14), is one-homogeneous, that is $J_{NL-TV}(\lambda u) = \lambda J_{NL-TV}(u)$. Thus we have a similar "characteristic function" structure of the Legendre-Fenchel transform ($J^*_{NL-TV} := \sup_u < u, v > -J(u)$). Solving the projection ,we reach a similar Euler-Lagrange equation for the constrained problem, resolve the value of the Lagrange multiplier using the same arguments and reach the above fixed point iterations. The bound on τ (Th. 3.1 in [12]) follows through in the same manner, having transformed the operators to their nonlocal counterparts. The only difference is that in our case $\|\mathrm{div}_{wd}\|^2_{L^2}$ is not resolved (with the definitions of [12] it is shown that $\|\mathrm{div}\|^2_{L^2} \leq 8$). □

A bound on τ. The bound on τ depends on the operator norm $\|\text{div}_{wd}\|^2$ which is a function of the weights $w_{i,j}$. As the weights are image dependent, so is $\|\text{div}_{wd}\|^2$. We propose below a simple bound which is very straightforward and does not depend on the image. We assume that the maximal number of neighbors for each pixel is a fixed parameter (not image dependent) and that the weights are bounded by some value, typically 1.

PROPOSITION 5.2. *Let m be the maximal number of neighbors of a pixel, $m := \max_i\{\sum_j(\text{sign}(w_{i,j}))\}$. If the weights are in the range $0 \le w_{i,j} \le 1\ \forall i,j$, then for $0 < \tau \le \frac{1}{4m}$ the algorithm converges.*

Proof. We need to show that $\|\text{div}_{wd}\|^2 \le 4m$:

$$\begin{aligned}\|\text{div}_{wd}(p)\|^2 &= \sum_i \left(\sum_j (p_{i,j} - p_{j,i})\sqrt{w_{i,j}}\right)^2 \\ &\le 2\sum_i \left(\sum_j (p_{i,j}^2 + p_{j,i}^2)\right)\left(\sum_j w_{i,j}\right) \\ &\le 4\max_i \left(\sum_j w_{i,j}\right)\sum_i\sum_j p_{i,j}^2 \\ &\le 4m\|p\|^2.\end{aligned}$$

□

Remark: . Note that in [12] the discrete local gradient and divergence operators are not symmetric, thus they do not fall precisely to the framework of this paper. Yet, the divergence operator of [12] can be viewed as div_{wd} with *nonsymmetric weights* of unit value where $m = 2$. In this sense the original bound $\|\text{div}\|^2 \le 8$ can be viewed as a special case of the bound presented above.

5.4.2. Nonlocal TV-L1. To solve (4.3) we generalize the algorithm of [3]. We consider the problem:

$$\inf_{u,v}\left\{J_{NL-TV}(u) + \frac{1}{2\alpha}\|f - u - v\|_{L^2}^2 + \lambda\|v\|_{L^1}\right\} \tag{5.4}$$

The parameter α is small so that we almost have $f = u + v$, thus (5.4) is a very good approximation of (4.3). We can solve the discretized version of (5.4) by iterating:

- v being fixed (we have a nonlocal ROF problem), find u using the nonlocal Chambolle's projection algorithm:

$$\inf_u\left(J_{NL-TV}(u) + \frac{1}{2\alpha}\|f - u - v\|_{L^2}^2\right)$$

- u being fixed, find v which satisfies:

$$\inf_v \frac{1}{2\alpha}\|f - u - v\|_{L^2}^2 + \lambda\|v\|_{L^1}.$$

The solution for v is given by soft-thresholding $f - u$ with $\alpha\lambda$ as the threshold [13], denoted by $ST_{\alpha\lambda}(f-u)$, where

$$ST_\beta(q) := \begin{cases} q - \beta, & q > \beta \\ 0, & |q| \le \beta \\ q + \beta, & q < -\beta. \end{cases} \tag{5.5}$$

PROPOSITION 5.3. *The algorithm converges to the global minimizer as $n \to \infty$ for any $0 < \tau \le \frac{1}{\|\text{div}_{wd}\|_{L^2}^2}$.*

Proof. The proof is similar to the one of [3] (which is in the spirit of [2]). Using similar arguments one can show that having solved the nonlocal ROF problem the iterative process presented here converges to the global minimizer. For the convergence of the NL-ROF part we use Theorem 5.1 above. □

5.4.3. Nonlocal TV-G. One can repeat the same arguments for minimizing nonlocal $TV - G$, Eq. (4.6), and modify the iterative projection algorithm of [2] to be nonlocal, in a similar manner as the algorithms presented above. We briefly summarize the idea: Let X be the Euclidean space and let G_μ be defined as

$$G_\mu = \{v \in NL - G / \|v\|_{NL-G} \leq \mu\}.$$

We consider the following problem:

$$\inf_{(u,v)\in X\times G_\mu} \left\{ J_{NL-TV}(u) + \frac{1}{2\alpha}\|f-u-v\|_{L^2}^2 \right\}. \tag{5.6}$$

To find the minimizer of the discretized functional one can apply the following simple iterations of NL-ROF minimizations. Initialize $u = f, v = 0$. Iterate until convergence:

- v being fixed (one needs to solve a nonlocal ROF problem), find u using the nonlocal projection algorithm (see Section 5.4.1),

$$\inf_u \left(J_{NL-TV}(u) + \frac{1}{2\alpha}\|f-u-v\|_{L^2}^2 \right)$$

- u being fixed, find v using the nonlocal projection algorithm,

$$\inf_{\tilde{u}} \left(J_{NL-TV}(\tilde{u}) + \frac{1}{2\mu}\|f-u-\tilde{u}\|_{L^2}^2 \right),$$

  where $v = f - u - \tilde{u}$.

In the second step, where u is fixed, we need to solve $\inf_{v\in G_\mu} \left\{ \|f-u-v\|_{L^2}^2 \right\}$. See the analysis of [2] or Section 3 in [3] showing that the minimization amounts to an ROF problem. The arguments for our nonlocal version are similar. Note that in practice one does not need to fully converge for each NL-ROF solution and can use only a few iterations.

6. Experiments.

6.1. Nonlocal TV Inpainting. Here we show the distinct difference between local and nonlocal TV-inpainting (see [16] for the local method). We minimize the functional (4.2). As usual for this problem we assume having the inpainting regions $\Omega_{inp} \in \Omega$ in advance. We would like to fill-in the missing information in a sensible manner according to the data in the rest of the image Ω/Ω_{inp}. In the toy example of Fig. 6.1 a textured region has to be inpainted. The local TV chooses a locally smooth solution (non-oscillatory) which does not fit this data. The nonlocal smoothness, as defined by our functional (using patch-based distances) fills-in the information correctly. The notion of *smoothness* is generalized to *regularity*. Thus patches with similar partial data are selected to fill-in the missing information. In Fig. 6.2 two examples of filling in parts of the Barbara image are shown. One is of regular texture (knee part) and one is non-texture (face). It is shown that the missing information is being replaced well by the algorithm, such that it is hard to distinguish visually

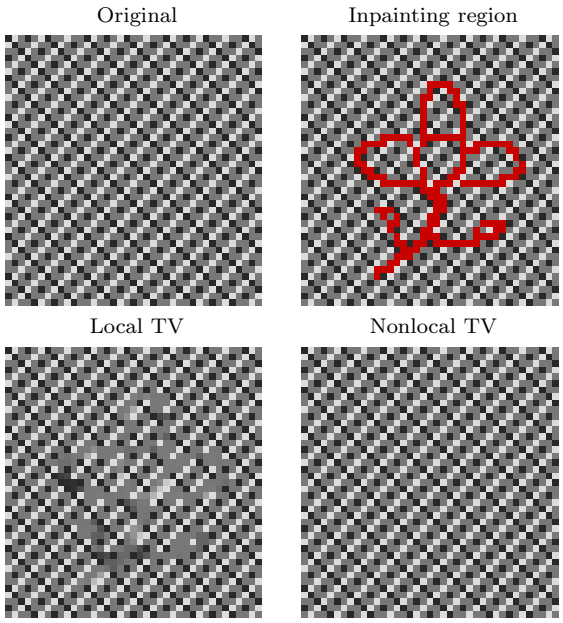

FIG. 6.1. *Nonlocal vs. local TV inpainting. Top: original texture (left), the texture with the inpainting region (in red). Bottom: results of local TV-inpainting [16] (left) and the nonlocal method using Eq.* (4.2). *The nonlocal method recovers the texture pattern correctly.*

between the original and the inpainted image. The errors of the inpainting results, with respect to the original image, are presented on the right side (second and fourth rows of the knes and face, respectively).

In the above cases we performed a single inpainting iteration and the missing regions are required to be smaller than the patches. For filling-in larger regions, an iterative process is necessary, where the boundaries have to be filled in first and one can then recompute the weights deeper into the inpainting region and regularize again. This process can be viewed as a variational understanding of the process suggested by Efros and Leung [25]. See also a deterministic approach suggested in [24].

The inpainting regularization can work also with the quadratic regularizer $J(u) = \int_\Omega |\nabla_w u|^2 dx$. In the inpainting problem however, as oppose to denoising, the weights for pixels to be inpainted are computed based on partial patches, where the central point of the inpainting region is unknown. Thus there are cases where two types of completions are possible. The quadratic regularizer will have a weighted averaging solution whereas the TV-type regularizer will have a weighted median solution, which is usually sharper and more attractive visually (see illustration in Fig. 6.3).

6.2. Nonlocal $TV - L^1$ regularization. This model presents a new way of viewing signal and image variational regularization. It replaces the local notion of smoothness by the global notion of regularity. Thus features which appear frequently are preserved.

The local $TV - L^1$ model is known to remove outliers, such as impulsive noise [41]. It is also known to keep intact large structures without reducing contrast (as oppose to the $TV - L^2$ case) while eliminating the smaller scales [14].

The nonlocal concept of "large" scales replaces the physical size of objects (pixels with a constant color) with the frequency of their appearance. Thus smaller scales

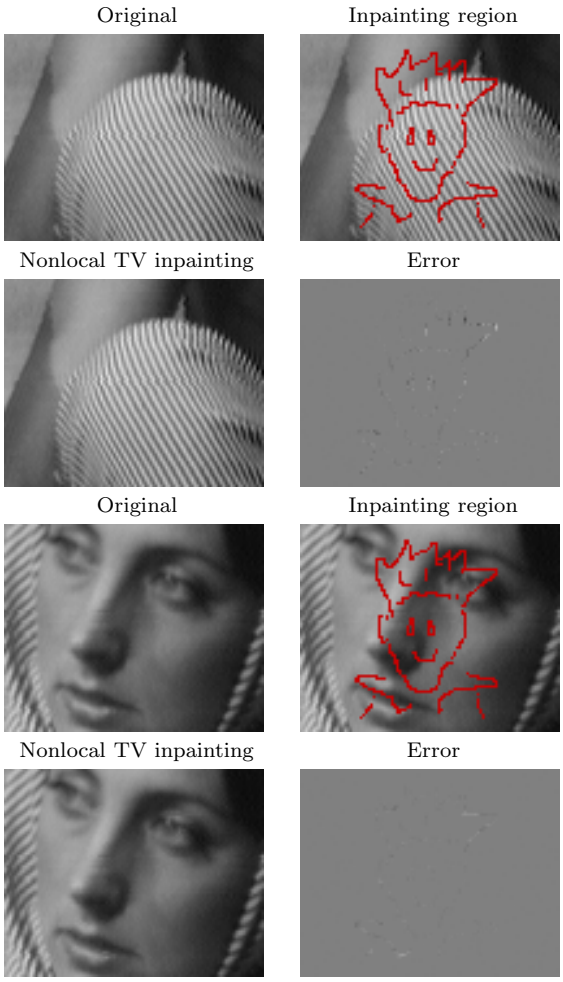

FIG. 6.2. *Nonlocal TV inpainting, Barbara image. Top: original, knee part (left), inpainting region (in red). Second row: results of nonlocal TV-inpainting, Eq.* (4.2) *(left) and errors from the original image. On the third and fourth rows an example of inpainting the face is shown (non-textural part). The algorithm (with the same parameters) recovers well both types of regions.*

should be interpreted as rare features. We obtain a variational regularization procedure which detects and removes irregularities. This can be very useful for regularizing textures, as seen in the examples below.

See Section for details regarding the calculation of the weights.

In Fig. 6.4 we give a toy example of small but very repetitive features versus large but rare ones. We also add some white Gaussian noise (of standard deviation $\sigma = 10$). We can observe that this regularization keeps the textures (small physical scale) and removes the larger objects (replacing them with texture). The texture itself is also regularized in the sense that the noise is removed. The residual part $f - u$ can be viewed as an anomaly detector.

Fig. 6.5 depicts an experiment where the search neighborhood $S(x)$ is changed. $S(x)$ controls the size of the region around each pixel for which similar patches are examined (see Section 4.2 for details). It should reflect the expected *auto-similarity*

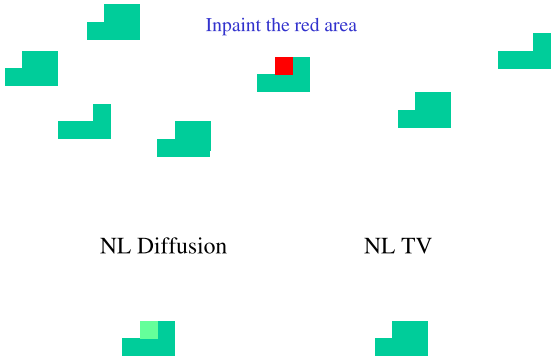

FIG. 6.3. *TV versus quadratic regularization for inpainting. When two filling-in options are possible, the TV regularizer takes a median solution, whereas the quadratic regularizer takes an averaging solution.*

scale of the image, or how far we should look in order to find repetitive structures. In this example the patterns are about 30 pixels apart. When $S(x)$ is too small (9×9 pixels), the regularization does not take into account the large-scale regularity of the image. Thus the regularization quality is degraded: corners are eroded and outliers and scratches are not removed well.

In Figs. 6.6 and 6.7 we try to detect and remove texture irregularities. We compare the nonlocal $TV - L^1$ with the local version and a simple 3×3 median filter. We can see that the local $TV - L^1$ behaves qualitatively in a similar manner to a median filter (removes small physical features) whereas the nonlocal version keeps well regular feature, even at the smallest scales. Outliers with a large scale can be removed while retaining the fine coherence of the textural nature. For the nonlocal and local $TV - L^1$ regularization we retained the same residual L^1 norm: $\|f - u\|_{L^1}$. This fair comparison is harder to obtain in the case of the median filter, but the residual norm is of a similar value.

In Fig. 6.8 we regularized part of a zebra image with different values of λ to show qualitative the behavior of the nonlocal $TV - L^1$ "scale-space". Smaller value of λ means stronger regularization.

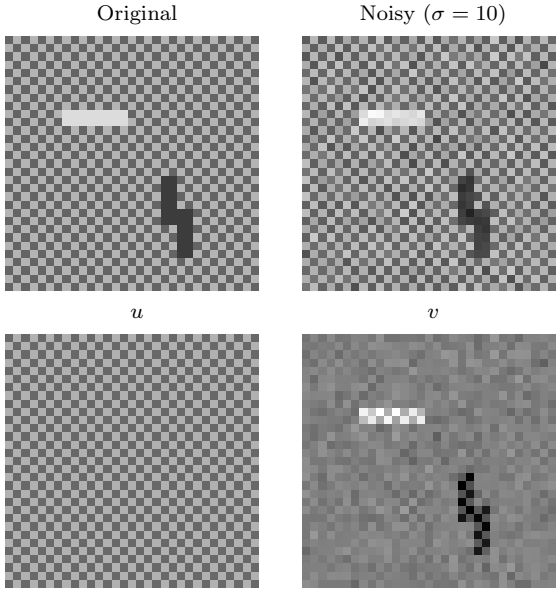

FIG. 6.4. *Removal of Anomalies by nonlocal $TV - L^1$. Top: original image (left), image with additive white Gaussian noise. Bottom: result of nonlocal $TV - L^1$ regularization u (left), $v \approx f - u$. This type of regularization retains repetitive patterns and removes rare and irregular ones (the light and dark larger symbols in this case). Note also that a standard removal of the noise is achieved.*

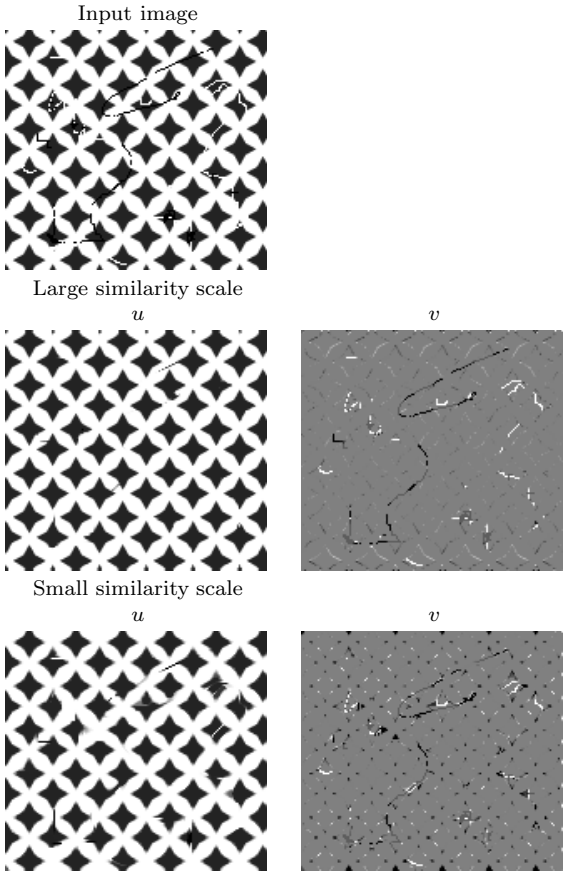

FIG. 6.5. *Auto similarity scale in nonlocal* $TV - L^1$. *The search neighborhood* $S(x)$ *for computing the weights (Section 4.2) controls the similarity scale of the regularization. Top: input image. Middle: nonlocal* $TV - L^1$ *with a large search neighborhood (*$S(x)$ *is a window of size* 61×61*). Bottom: nonlocal* $TV - L^1$ *with a small search neighborhood (*$S(x)$ *is a window of size* 9×9*).*

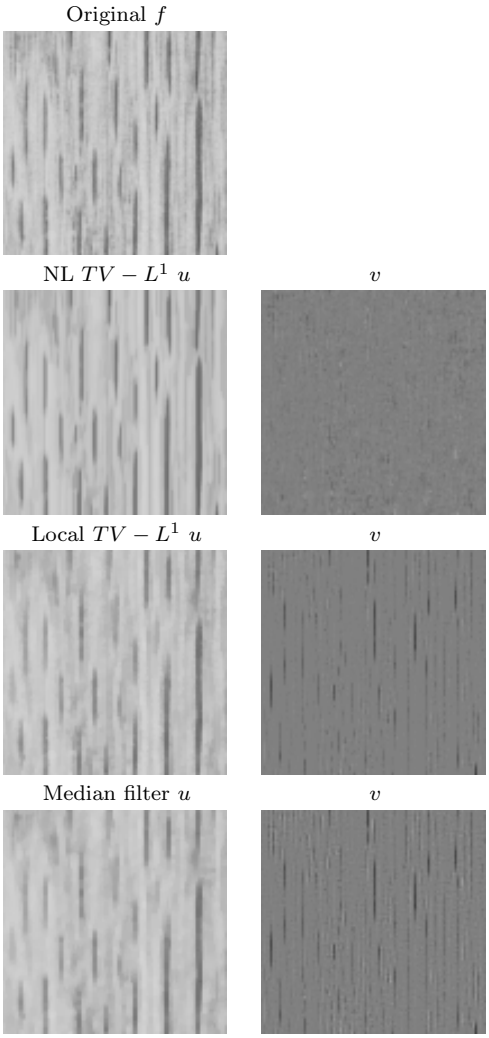

FIG. 6.6. *Detecting and removing irregularities from textures by different methods. Example 1. u (left) - regularized texture. v (right) - texture irregularities. The same L^1 norm of the residual $\|v\|_{L^1}$ is used for the nonlocal $TV - L^1$ and the local $TV - L^1$.*

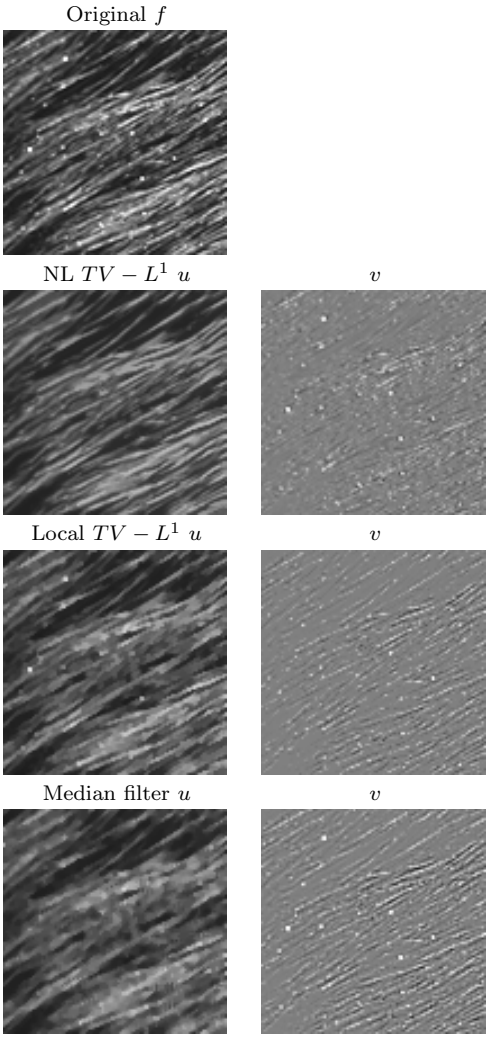

FIG. 6.7. *Detecting and removing irregularities from textures by different methods. Example 2.* $\|v\|_{L^1}$ *is the same for the nonlocal* $TV - L^1$ *and the local* $TV - L^1$.

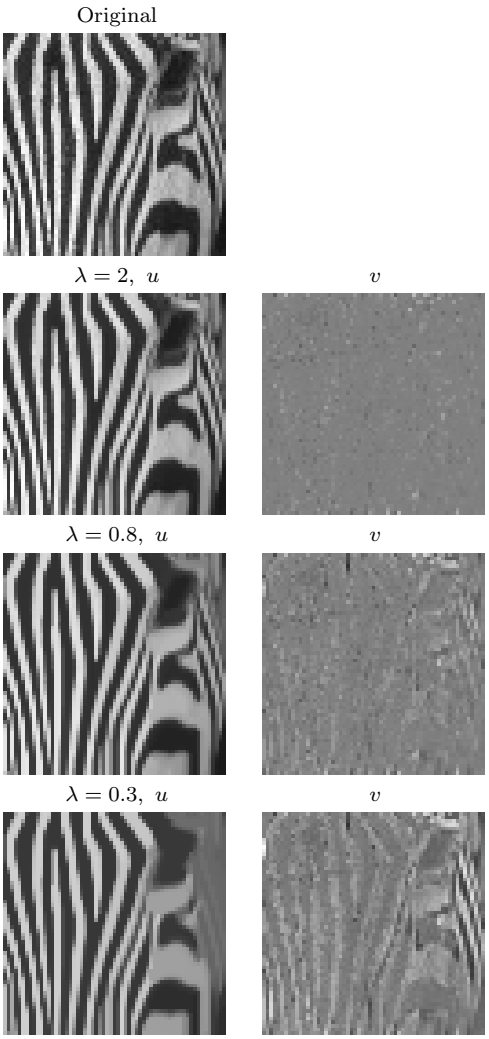

FIG. 6.8. *NL TV* $- L^1$. *Regularization results for different values of* λ.

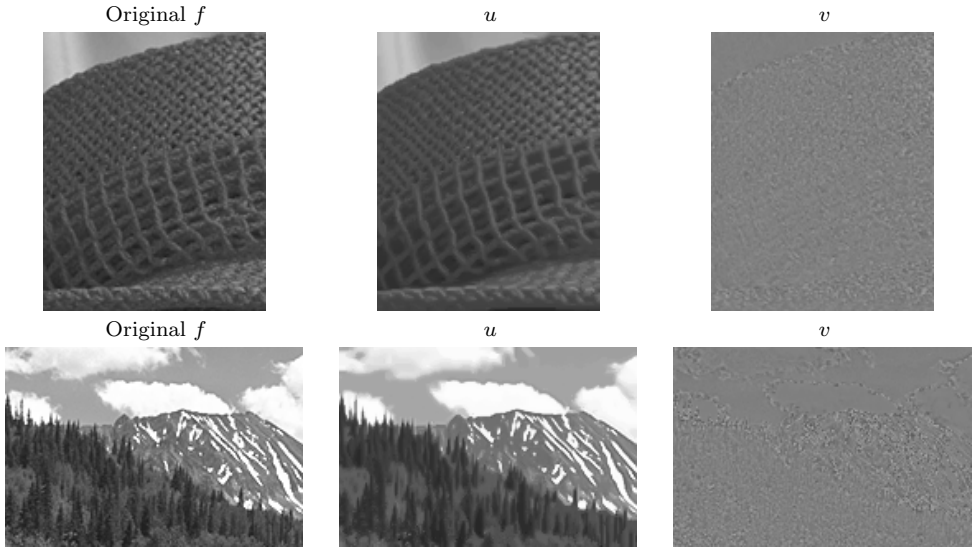

FIG. 6.9. *Examples of regularizing images with nonlocal $TV-G$, Eq.* (4.6). *The filter removes well random parts of the image, preserving edges and regular patterns.*

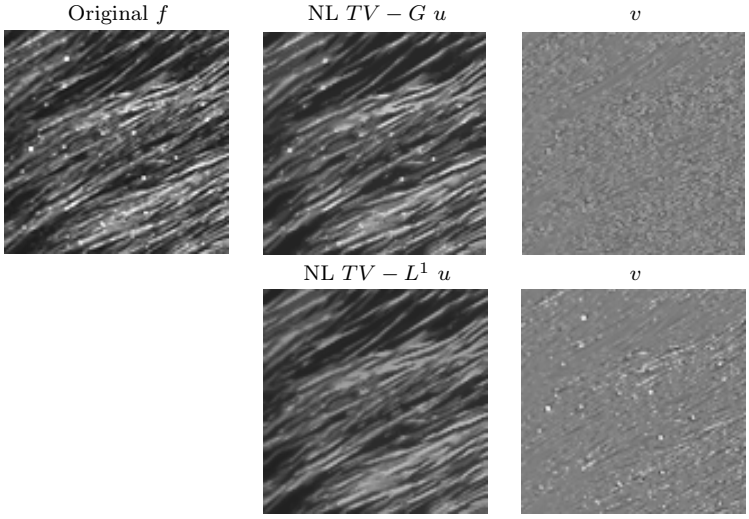

FIG. 6.10. *Nonlocal $TV-G$ regularization, Eq.* (4.6), *compared with nonlocal $TV-L^1$, Eq.* (4.3). *Nonlocal $TV-G$ removes random oscillations but keeps outliers.*

6.3. Nonlocal $TV-G$ regularization. In Fig. 6.9 two examples of regularizing images with nonlocal $TV-G$, Eq. (4.6), are shown. The images are taken from the Kodak collection [33]. The qualitative properties of this regularization are different from the original functional proposed by Meyer [37], see examples e.g. in [55, 2, 3]. A notable difference is that the regular textural part is preserved and only random textures are removed from u. Edges, as in the original model, are well preserved without erosion of contrast. Fig 6.10 compares the nonlocal $TV-G$ minimization with nonlocal $TV-L^1$, showing that the latter is more suited for removing outliers.

7. Conclusion. A very general framework is presented for processing signals and images non-locally. Two categories of functionals are suggested: one which is based on generalized nonlocal gradient and divergence operators. The other is based on differences. In this paper we focus on the first category, present the general framework and generalize several projection algorithm for computing the nonlocal versions of ROF [48], $TV - L^1$ [41] and $TV - G$ [37].

In essence two steps are required for this type of regularization: a first step consists of finding the weights between pixels. We used patch based similarities following [9]. Other affinity measures between regions and pixels can naturally be proposed. The second step is choosing the appropriate regularization and functional minimization. It is shown how nonlocal $TV - L^1$ can be used to detect and remove irregularities from textures. In addition we demonstrate that nonlocal TV-inpainting can fill-in repetitive textures correctly.

Preliminary calculations done elsewhere already indicate that Hamilton-Jacobi equations in at least five space dimensions can be solved effectively using the approach outlined in Section 3.

We currently would like to extend the theoretical foundations and also to investigate additional applications for which this framework can contribute.

REFERENCES

[1] G. Aubert and P. Kornprobst. *Mathematical Problems in Image Processing*, volume 147 of *Applied Mathematical Sciences*. Springer-Verlag, 2002.

[2] J.F. Aujol, G. Aubert, L. Blanc-Féraud, and A. Chambolle. Image decomposition into a bounded variation component and an oscillating component. *JMIV*, 22(1), January 2005.

[3] J.F. Aujol, G. Gilboa, T. Chan, and S. Osher. Structure-texture image decomposition – modeling, algorithms, and parameter selection. *International Journal of Computer Vision*, 67(1):111–136, 2006.

[4] S. Awate and R. Whitaker. Higher-order image statistics for unsupervised, information-theoretic, adaptive image filteringg. In *Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition*, volume 2, pages 44–51, 2005.

[5] D. Barash. A fundamental relationship between bilateral filtering, adaptive smoothing and the nonlinear diffusion equation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(6):844–847, 2002.

[6] D. Barash and D. Comaniciu. A common framework for nonlinear diffusion, adaptive smoothing, bilateral filtering and mean shift. *Image and Vision Computing*, 22(1):73–81, 2004.

[7] 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)*, volume 4485 of *Lecture Notes in Computer Science*, pages 128–139, 2007.

[8] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(11):1222–1239, 2001.

[9] A. Buades, B. Coll, and J-M. Morel. On image denoising methods. *SIAM Multiscale Modeling and Simulation*, 4(2):490–530, 2005.

[10] A. Buades, B. Coll, and J-M Morel. Neighborhood filters and PDE's. *Numerische Mathematik*, 105(10):1–34, 2006.

[11] T. Cecil, J.L. Qian, and S. Osher. Numerical methods for high dimensional Hamilton-Jacobi equations using radial bases functions. *J. Comput. Phys.*, 196:327–347, 2004.

[12] A. Chambolle. An algorithm for total variation minimization and applications. *JMIV*, 20:89–97, 2004.

[13] A. Chambolle, R.A. De Vore, N. Lee, and B.J. Lucier. Nonlinear wavelet image processing: Variational problems, compression, and noise removal through wavelet shrinkage. *IEEE Transcations on Image Processing*, 7(3):319–335, March 1998.

[14] T. Chan and S. Esedoglu. Aspects of total variation regularized l^1 function approximation, 2004. CAM report 04-07.

[15] T. F. Chan, S. Osher, and J. Shen. The digital TV filter and nonlinear denoising. *IEEE Trans. Image Process.*, 10(2):231–241, 2001.

[16] T.F. Chan and J. Shen. Mathematical models of local non-texture inpaintings. *SIAM J. Appl. Math.*, 62(3):1019–1043, 2001.
[17] T.F. Chan and J. Shen. *Image Processing and Analysis*. SIAM, 2005.
[18] J-S. Chen, X. Zhang, and T. Belytschko. An implicit gradient model by a reproducing kernel strain regularization in strain localization problems. *Computer Methods in Applied Mechanics and Engineering*, 193:2827–2844, 2004.
[19] F. Chung. *Spectral Graph Theory*. Number 92 in CBMS Regional Conference Series in Mathematics. American Mathematical Society, 1997.
[20] R.R. Coifman, S. Lafon, A.B. Lee, M. Maggioni, B. Nadler, F. Warner, and S. Zucker. Geometric diffusion as a tool for harmonic analysis and structure definition of data, part i: Diffusion maps. *Proceedings of the National Academy of Sciences*, 102(21):7426–7431, 2005.
[21] M.G. Crandall, H. Ishii, and P.-L. Lions. User's guide to viscosity solutions of second order partial differential equations. *A.M.S. Bull.*, 27:1–67, 1992.
[22] J. Darbon and M. Sigelle. Image restoration with discrete constrained total variation part ii: Levelable functions, convex priors and non-convex cases. *Accepted to the Journal of Mathematical Imaging and Vision*, 2005.
[23] J. Darbon and M. Sigelle. Image restoration with discrete constrained total variation part i: Fast and exact optimization. *Journal of Mathematical Imaging and Vision*, 2006.
[24] L. Demanet, B. Song, and T. Chan. Image inpainting by correspondence maps: a deterministic approach. In *Proc. VLSM, Nice*, 2003. [see also UCLA CAM Report 03-40].
[25] A.A. Efros and T.K. Leung. Texture synthesis by non-parametric sampling. In *ICCV (2)*, pages 1033–1038, 1999.
[26] M. Elad. On the bilateral filter and ways to improve it. *IEEE Transactions On Image Processing*, 11(10):1141–1151, 2002.
[27] G. Gilboa, J. Darbon, S. Osher, and T.F. Chan. Nonlocal convex functionals for image regularization, 2006. UCLA CAM Report 06-57.
[28] G. Gilboa and S. Osher. Nonlocal linear image regularization and supervised segmentation. *SIAM Multiscale Modeling and Simulation*, 6(2):595–630, 2007.
[29] L. Grady. Random walks for image segmentation. *to appear in IEEE Trans. on Pattern Analysis and Machine Intelligence*, 2006.
[30] C. Kervrann and J. Boulanger. Unsupervised patch-based image regularization and representation. In *Proc. European Conf. Comp. Vision (ECCV'06), Graz, Austria*, 2006.
[31] R. Kimmel, R. Malladi, and N. Sochen. Images as embedding maps and minimal surfaces: Movies, color, texture, and volumetric medical images. *International Journal of Computer Vision*, 39(2):111–129, 2000.
[32] S. Kindermann, S. Osher, and P. Jones. Deblurring and denoising of images by nonlocal functionals. *SIAM Multiscale Modeling and Simulation*, 4(4):1091 – 1115, 2005.
[33] Kodak. Kodak image collection, 2002. see http://www.cipr.rpi.edu/resource/stills/kodak.html.
[34] V. Kolmogorov and R. Zabih. What energy functions can be minimized via graph cuts? *IEEE Trans. Pattern Anal. Mach. Intell.*, 26(2):147–159, 2004.
[35] R.I. Kondor and J.D. Lafferty. Diffusion kernels on graphs and other discrete input spaces. In *ICML*, pages 315–322, 2002.
[36] M. Mahmoudi and G. Sapiro. Fast image and video denoising via nonlocal means of similar neighborhoods. *IEEE Signal Processing Letters*, 12(12):839–842, 2005.
[37] Y. Meyer. Oscillating patterns in image processing and in some nonlinear evolution equations, March 2001. The Fifteenth Dean Jacquelines B. Lewis Memorial Lectures.
[38] B. Mohar. The Laplacian spectrum of graphs. In *Y. Alavi, G. Chartrand, O. R. Oellermann, A. J. Schwenk (Eds.), Graph Theory, Combinatorics, and Applications, Wiley*, volume 2, pages 871–898, 1991.
[39] G. Motta, E. Ordentlich, I. Rami'rez, G. Seroussi, , and M.J. Weinberger. The dude framework for continuous tone image denoising. In *Proc. IEEE Int. Conf. on Image Processing*, volume 3, pages 345–348, 2005.
[40] B. Nadler, S. Lafon, R.R. Coifman, and I.G. Kevrekidis. Diffusion maps, spectral clustering, and the reaction coordinates of dynamical systems. Report, Math. Dept. Yale, Nov. 2004. To appear in Journal of Applied and Computational Harmonic Analysis.
[41] M. Nikolova. A variational approach to remove outliers and impulse noise. *JMIV*, 20(1-2):99–120, 2004.
[42] S. Osher and N. Paragios (Eds.). *Geometric Level Set Methods in Imaging, Vision, and Graphics*. Springer-Verlag, 2003.
[43] S. Osher and R. Fedkiw. *Level Set Methods and Dynamic Implicit Surfaces*. Springer, 2002.
[44] S. Osher and J.A. Sethian. Fronts propagating with curvature dependent speed-algorithms

based on Hamilton-Jacobi formulations. *J. Comput. Phys.*, 79:12–49, 1988.

[45] S. Osher and C.-W. Shu. High order essentially nonoscillatory schemes for Hamilton-Jacobi equations. *SINUM*, 28:907–922, 1991.

[46] P. Perona and W.T. Freeman. A factorization approach to grouping. In *ECCV*, pages 655–670, 1998.

[47] P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. *PAMI*, 12(7):629–639, 1990.

[48] L. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D*, 60:259–268, 1992.

[49] C. Sagiv, N. Sochen, and Y.Y. Zeevi. Integrated active contours for texture segmentation. *IEEE Trans. on Image Processing*, 15(6):1633–1646, 2006.

[50] G. Sapiro. *Geometric Partial Differential Equations and Image Processing*. Cambridge University Press, 2001.

[51] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

[52] N. Sochen, R. Kimmel, and R. Malladi. A general framework for low level vision. *IEEE Transactions on Image Processing*, 7:310–318, 1998.

[53] A.D. Szlam, M. Maggioni, Jr. J.C. Bremer, and R.R. Coifman. Diffusion-driven multiscale analysis on manifolds and graphs: top-down and bottom-up constructions. In *SPIE*, 2005.

[54] C. Tomasi and R. Manduchi. Bilateral filtering for gray and color images. In *ICCV '98*, pages 839–846, 1998.

[55] L. Vese and S. Osher. Modeling textures with total variation minimization and oscillating patterns in image processing. *Journal of Scientific Computing*, 19:553–572, 2003.

[56] Y. Weiss. Segmentation using eigenvectors: A unifying view. In *International Conference on Computer Vision*, pages 975–982, 1999.

[57] L.P. Yaroslavsky. *Digital Picture Processing, an Introduction.* Springer-Verlag, Berlin, 1985.

[58] D. Zhou and B. Scholkopf. A regularization framework for learning from graph data. In *ICML Workshop on Statistical Relational Learning and Its Connections to Other Fields*, 2004.

[59] D. Zhou and B. Scholkopf. Regularization on discrete spaces. In *Pattern Recognition, Proceedings of the 27th DAGM Symposium, Berlin, Germany*, pages 361–368, 2005.