

Nonlocal Discrete Regularization on Weighted Graphs: a Framework for Image and Manifold Processing

Abderrahim Elmoataz, Olivier Lezoray, Sébastien Bougleux

Abstract—We introduce a nonlocal discrete regularization framework on weighted graphs of the arbitrary topologies for image and manifold processing. The approach considers the problem as a variational one, which consists in minimizing a weighted sum of two energy terms: a regularization one that uses a discrete weighted p -Dirichlet energy, and an approximation one. This is the discrete analogue of recent continuous Euclidean nonlocal regularization functionals. The proposed formulation leads to a family of simple and fast nonlinear processing methods based on the weighted p -Laplace operator, parameterized by the degree p of regularity, the graph structure and the graph weight function. These discrete processing methods provide a graph-based version of recently proposed semi-local or nonlocal processing methods used in image and mesh processing, such as the bilateral filter, the TV digital filter or the nonlocal means filter. It works with equal ease on regular 2D-3D images, manifolds or any data. We illustrate the abilities of the approach by applying it to various types of images, meshes, manifolds and data represented as graphs.

Index Terms—Nonlocal discrete regularization, Weighted graph, p -Laplacian, Image and manifold processing.

I. INTRODUCTION

IN many computer science applications, it is necessary to process images, meshes, manifolds and more generally data. This refers to the following research fields: image processing, computer graphics and data mining.

In the context of image processing, smoothing and denoising are key filtering processes. Among the existing methods, the variational ones, based on regularization, provide a general framework to design such efficient filter processes. Solutions of variational models can be obtained by minimizing appropriate energy functions. The minimization is usually performed by designing continuous Partial Differential Equations (PDE), whose solutions are discretized in order to fit with the image domain. A complete overview of these methods can be found in [1], [2], [3], [4] and references therein.

In the context of mesh processing, smoothing and denoising are also key processes dedicated to noise removal causing minimal damage to geometric features. Most mesh smoothing

methods are based on the discrete Laplace-Beltrami regularization or on the discrete curvature regularization [5], [6]. Other smoothing methods, based on feature preserving were mostly inspired by anisotropic diffusion in image processing [7], [8], [9]. Geometric flows have been extensively used in mesh processing [10]. In particular, surface flows based on functional minimization (i.e. evolving a surface so as to progressively decrease an energy functional) is a common methodology in geometric processing with applications to spanning surface diffusion [11], denoising of scanned meshes [12], shape optimization and surface design [13], minimal surfaces [14], etc.

In the context of manifold and data processing, dimensionality reduction [15], [16], [17] (extracting low dimensional structure from high dimensional data), clustering [18], [19] (automatic identification of groups of similar objects) and classification [20], [21], [22] (assigning labels to instances) are key processes. In particular, methods based on graph Laplacian have become increasingly popular in machine learning to perform any of the above-mentioned key processes. However, manifolds and discrete data, as images and meshes, can also contain inner noise. Therefore, regularization occurs as a natural candidate for manifold processing [23], [24] since e.g. noise filtering can be useful to avoid overfitting in a learning process. Manifold regularization denotes also a family of learning algorithms [20] based on regularization with both labeled and unlabeled data. Transductive graph learning algorithms and standard methods like Support Vector Machines [22] can then be considered as special cases of manifold regularization. All these methods are based on the assumption that the data lies on a submanifold. However, sampled data lies almost never exactly on the submanifold due to the noise scattered around it. Since most previous methods are sensitive to noise, it is essential to denoise the manifold data to project it onto a submanifold [25].

As we have just mentioned, regularization is a principle the interest of which concerns a wide range of computer science domains. The application of partial and variational methods to images, meshes, manifolds and any discrete data processing has shown its effectiveness allowing high quality regularization processes. However, there are some limitations to the functionals used in regularization such as total variational ones [2]. Indeed, the latter are based on derivatives which consider local features of the data. Since the advent of the nonlocal means filter [1], the use of nonlocal interactions, to capture the complex structure of the data, has received a lot

The authors are with the Université de Caen Basse-Normandie and the ENSICAEN in the GREYC Laboratory, Image Team, 6 Boulevard Maréchal Juin, F-14050 CAEN cedex FRANCE.

This research work was partially supported by the ANR foundation under grant ANR-06-MDCA-008-01/FOGRIMMI and by FEDER and FNADT funds.

Email: abder.elmoataz@greyc.ensicaen.fr, olivier.lezoray@unicaen.fr, sebastien.bougleux@greyc.ensicaen.fr

Phone: +33(0)231472720, Fax: +00(0)231472698

of interest and has shown to be very effective. A variational understanding of the nonlocal means filter was first developed in [26] as a non convex functional and later in [27], [28] as a convex quadratic functional. Moreover, the use of manifolds to describe nonlocal interactions of geometric structures in signals and images has recently been explored [23]. In this latter work, diffusion over these manifolds is shown to be equivalent to nonlocal processing methods. However, most of these variational nonlocal formulations are expressed in a continuous setting and unfortunately the discretization of the underlying differential operators is difficult for high dimensional data.

Moreover, whatever the origin of the data (images or databases), a common representation can be used to model them: graphs. For the particular case of images, pixels have a specific organization expressed by their spatial connectivity. Therefore, a typical graph used to represent images is a grid graph. For the particular case of unorganized data, a graph can also be associated with by modeling neighborhood relationships between the data elements. As a consequence, by using the same adapted underlying graph representation to model data with a given organization (images) or without any (general databases), one can define a framework which can be used to process general databases. We take that point of view of processing any data ranging from images to manifolds.

Inspired by continuous regularization of images and recent works on nonlocal functional for continuous regularization, we propose a general framework extending our previous works [29], [30], [31], [32] based on a nonlocal discrete regularization on weighted graphs of the arbitrary topologies. This framework can also be considered as a discrete analogue of the nonlocal continuous regularization for the case of images [33], [27]. Furthermore, the proposed framework works on any discrete data represented by weighted graphs which enables a nonlocal regularization with equal ease on images and manifolds. Our approach starts directly with a discrete variational problem and works with data living on any general discrete domain. To take into account the nonlocal interactions in images and manifolds, we explicitly introduce discrete nonlocal derivatives and functionals on graphs of the arbitrary topologies, to transcribe the continuous regularization. This framework unifies the regularization of images, meshes, manifolds and data. The latter regularization enables local, semi-local or nonlocal regularization by using appropriate graph topologies and edge weights. Let $G = (V, E, w)$ be a weighted graph consisting in a set of vertices V , a set of edges $E \subset V \times V$, and a similarity weight function w defined on edges. Let $\mathcal{H}(V)$ be a Hilbert space defined on the vertices of G . We formalize the discrete data regularization of a function $f^0 \in \mathcal{H}(V)$ by the following minimization problem:

$$\min_{f \in \mathcal{H}(V)} \left\{ E_w^p(f, f^0, \lambda) = \frac{1}{p} \sum_{v \in V} |\nabla_w f(v)|^p + \frac{\lambda}{2} \|f - f^0\|_2^2 \right\}, \quad (1)$$

where $p \in (0, +\infty)$ is the smoothness degree, λ is the fidelity parameter, and $|\nabla_w f|$ represents the weighted local variation of the function f over the graph. The solution of problem (1) leads to a family of nonlinear processing methods, parameter-

ized by the weight function, the degree of smoothness, and the fidelity parameter. There exists two main advantages of using this framework, which can be considered as a discrete analogue of continuous regularization on weighted graphs.

First, the regularization is expressed directly in a discrete setting. Involved processing methods are computed by simple and efficient iterative algorithms, without solving any PDEs. Second, the topology of graphs can be arbitrary. Since the proposed approach is general, any discrete data set can be transformed into a weighted graph, by using a similarity measure between data features. Thus, we can consider any function defined on these data as a function defined on the graph vertices.

The regularization term in problem (1) is obtained by defining nonlocal functionals on weighted graphs. The idea of using functionals on graphs, in a regularization process, has also been proposed in other contexts, such as semi-supervised data learning [34], [35] and image segmentation [36]. In this paper, we introduce a large family of regularization problems based on the p -Laplacian which is the discrete analogue of the well-known Euclidean p -Laplacian to graphs. This leads to a family of discrete linear and nonlinear filters which includes and extends several well-known models used in image processing for the classical cases of $p = 2$ or $p = 1$. The case of $p < 1$ is also considered in this paper. Moreover, the proposed framework naturally embeds local to nonlocal processing. To summarize, the proposed regularization framework on graphs unifies several existing discrete image filters, generalizes them and extends them to the processing of arbitrary data.

In this paper, we first define functionals on weighted graphs in Section II. Section III presents the discrete regularization problem (1), and the associated filter family. Section IV analyzes the obtained filters and gives relations to existing methods. In Section V, we show some regularization examples, in the context of images, meshes, manifolds and data processing. Last Section concludes.

II. OPERATORS ON WEIGHTED GRAPHS

In this Section, we recall some basic definitions on graphs, and we define nonlocal operators which can be considered as discrete versions of continuous differential operators. Analogous definitions and properties have also been used in the context of semi-supervised learning [34], and differential calculus on graphs [37], [38].

A. Preliminary Notations and Definitions

A *weighted graph* $G = (V, E, w)$ consists of a finite set $V = \{v_1, \dots, v_N\}$ of N vertices and a finite set $E \subset V \times V$ of weighted edges. We assume G to be undirected, with no self-loops and no multiple edges. Let (u, v) be the edge of E that connects vertices u and v of V . Its weight, denoted by $w(u, v)$, represents the similarity between its vertices. Similarities are usually computed by using a positive symmetric function $w : V \times V \rightarrow \mathbb{R}_+$ satisfying $w(u, v) = 0$ if (u, v) is not an edge of E . The notation $u \sim v$ is also used to denote two adjacent vertices. We say that G is connected whenever, for any pair of

vertices (u, v) there is a finite sequence $u = u_0, u_1, \dots, u_n = v$ such that u_{i-1} is a neighbor of u_i for every $i \in \{1, \dots, n\}$.

Let $\mathcal{H}(V)$ be the Hilbert space of real-valued functions defined on the vertices of a graph $G = (V, E, w)$. A function $f : V \rightarrow \mathbb{R}$ of $\mathcal{H}(V)$ assigns a real value $f(v)$ to each vertex $v \in V$. Clearly, the function f can be represented by a column vector $f = [f(v_1), \dots, f(v_N)]^T$. By analogy with functional analysis on continuous spaces, the integral of a function $f \in \mathcal{H}(V)$, over the set of vertices V , is defined as $\int_V f = \sum_V f$. The space $\mathcal{H}(V)$ is endowed with the usual inner product $\langle f, h \rangle_{\mathcal{H}(V)} = \sum_{v \in V} f(v)h(v)$, where $f, h : V \rightarrow \mathbb{R}$.

Similarly, let $\mathcal{H}(E)$ be the space of real-valued functions defined on the edges of G . It is endowed with the inner product $\langle F, H \rangle_{\mathcal{H}(E)} = \sum_{(u,v) \in E} F(u, v)H(u, v) = \sum_{u \in V} \sum_{v \sim u} F(u, v)H(u, v)$, where $F, H : E \rightarrow \mathbb{R}$ are two functions of $\mathcal{H}(E)$.

B. Difference Operator and its Adjoint

Let $G = (V, E, w)$ be a weighted graph, and let $f : V \rightarrow \mathbb{R}$ be a function of $\mathcal{H}(V)$. The *difference operator* of f , noted $d : \mathcal{H}(V) \rightarrow \mathcal{H}(E)$, is defined on an edge $(u, v) \in E$ by:

$$(df)(u, v) = \sqrt{w(u, v)}(f(v) - f(u)). \quad (2)$$

The *directional derivative* (or *edge derivative*) of f , at a vertex $v \in V$, along an edge $e = (u, v)$, is defined as:

$$\left. \frac{\partial f}{\partial e} \right|_u = \partial_v f(u) = (df)(u, v). \quad (3)$$

This definition is consistent with the continuous definition of the derivative of a function: $\partial_v f(u) = -\partial_u f(v)$, $\partial_v f(v) = 0$, and if $f(v) = f(u)$ then $\partial_v f(u) = 0$.

The *adjoint* of the difference operator, noted $d^* : \mathcal{H}(E) \rightarrow \mathcal{H}(V)$, is a linear operator defined by:

$$\langle df, H \rangle_{\mathcal{H}(E)} = \langle f, d^* H \rangle_{\mathcal{H}(V)}, \quad (4)$$

for all $f \in \mathcal{H}(V)$ and all $H \in \mathcal{H}(E)$.

Proposition 1: The adjoint operator d^* , of a function $H \in \mathcal{H}(E)$, can be expressed at a vertex $u \in V$ by the following expression:

$$(d^* H)(u) = \sum_{v \sim u} \sqrt{w(u, v)}(H(v, u) - H(u, v)). \quad (5)$$

Proof: By using the definition of the inner product in $\mathcal{H}(E)$, the left term of (4) is rewritten as:

$$\begin{aligned} \langle df, H \rangle_{\mathcal{H}(E)} &= \sum_{(u,v) \in E} (df)(u, v)H(u, v) \\ &\stackrel{(2)}{=} \sum_{(u,v) \in E} \sqrt{w(u, v)}(f(v) - f(u))H(u, v) \\ &= \sum_{(u,v) \in E} \sqrt{w(u, v)}f(v)H(u, v) - \sum_{(u,v) \in E} \sqrt{w(u, v)}f(u)H(u, v) \\ &= \sum_{u \in V} \sum_{v \sim u} \sqrt{w(v, u)}f(u)H(v, u) - \sum_{u \in V} \sum_{v \sim u} \sqrt{w(u, v)}f(u)H(u, v) \\ &= \sum_{u \in V} f(u) \sum_{v \sim u} \sqrt{w(u, v)}(H(v, u) - H(u, v)) \\ &\stackrel{(4)}{=} \langle f, d^* H \rangle = \sum_{u \in V} f(u)(d^* H)(u). \end{aligned}$$

The proof is completed by comparing the last two equations. \blacksquare

The *divergence operator*, defined by $-d^*$, measures the net outflow of a function of $\mathcal{H}(E)$, at each vertex of the graph.

Proposition 2: Each function $H \in \mathcal{H}(E)$ has a null divergence over the entire set of vertices: $\sum_{u \in V} (d^* H)(v) = 0$.

Proof: From Proposition 1, $\sum_{u \in V} (d^* H)(v)$ is the sum of terms $\sqrt{w(v, u)}(H(v, u) - H(u, v)) + \sqrt{w(u, v)}(H(u, v) - H(v, u))$. Since w is symmetric, then this sum is null. \blacksquare

Other general definitions of the difference operator have been proposed in the literature. In particular, the difference operator recently proposed by Zhou and Schölkopf [34], and defined by:

$$(df)(u, v) = \sqrt{w(u, v)} \left(\frac{f(v)}{\sqrt{g_w(v)}} - \frac{f(u)}{\sqrt{g_w(u)}} \right)$$

with $g_w(u) = \sum_{v \sim u} w(u, v)$, is not null when the function f is locally constant (also pointed out by Hein *et al.* [39]). Moreover, its adjoint does not satisfy Proposition 2.

C. Gradient Operator

The *weighted gradient operator* of a function $f \in \mathcal{H}(V)$, at a vertex $u \in V$, is the vector operator defined by:

$$\begin{aligned} \nabla_w f(u) &= [\partial_v f(u) : v \sim u]^T \\ &= [\partial_{v_1} f(u), \dots, \partial_{v_k} f(u)]^T, \quad \forall (u, v_i) \in E. \end{aligned}$$

The \mathcal{L}_2 -norm of this vector represents the *local variation* of the function f at a vertex of the graph. It is defined by:

$$\begin{aligned} |\nabla_w f(u)| &= \sqrt{\sum_{v \sim u} (\partial_v f(u))^2} \\ &\stackrel{(3)}{=} \sqrt{\sum_{v \sim u} w(u, v)(f(v) - f(u))^2}. \end{aligned} \quad (6)$$

The local variation is a seminorm which measures the regularity of a function around a vertex of the graph.

D. A Family of Weighted p -Laplace Operators

Let $p \in (0, +\infty)$ be a real number. The *weighted p -Laplace operator* of a function $f \in \mathcal{H}(V)$, noted $\Delta_w^p : \mathcal{H}(V) \rightarrow \mathcal{H}(V)$, is defined by:

$$\Delta_w^p f(u) = \frac{1}{2} d^* (|\nabla_w f(u)|^{p-2} df(u, v)). \quad (7)$$

Proposition 3: The p -Laplace operator of $f \in \mathcal{H}(V)$, at a vertex $u \in V$, can be computed by:

$$\Delta_w^p f(u) = \frac{1}{2} \sum_{v \sim u} \gamma_w^f(u, v)(f(u) - f(v)), \quad (8)$$

with

$$\gamma_w^f(u, v) = w(u, v)(|\nabla_w f(v)|^{p-2} + |\nabla_w f(u)|^{p-2}). \quad (9)$$

Proof: From (7), (5) and (2), we have:

$$\begin{aligned} \Delta_w^p f(u) &= \frac{1}{2} \sum_{v \sim u} \sqrt{w(u, v)} \left(\frac{(df)(v, u)}{|\nabla_w f(v)|^{2-p}} - \frac{(df)(u, v)}{|\nabla_w f(u)|^{2-p}} \right) \\ &= \frac{1}{2} \sum_{v \sim u} w(u, v) \left(\frac{f(u) - f(v)}{|\nabla_w f(v)|^{2-p}} - \frac{f(v) - f(u)}{|\nabla_w f(u)|^{2-p}} \right). \end{aligned}$$

This last term is exactly (8). \blacksquare

Remark 1: In (7), $\nabla_w f$ is chosen to be evaluated at u . If one chooses to evaluate $\nabla_w f$ at v , the same expression of (7), but with a minus sign in (9), is obtained. To have $\gamma_w^f(u, v)$ as a positive expression, we prefer to have $\nabla_w f$ to be evaluated at u .

The p -Laplace operator is nonlinear, with the exception of $p = 2$. In this latter case, it corresponds to the *combinatorial graph Laplacian*, which is one of the classical second order operators defined in the context of spectral graph theory [40]:

$$\Delta_w f(u) = \sum_{v \sim u} w(u, v)(f(u) - f(v)). \quad (10)$$

Another particular case of the p -Laplace operator is obtained with $p = 1$. In this case, it is the *weighted curvature* of the function f on the graph:

$$\kappa_w f(u) = \frac{1}{2} \sum_{v \sim u} \left(\frac{w(u, v)}{|\nabla_w f(v)|} + \frac{w(u, v)}{|\nabla_w f(u)|} \right) (f(u) - f(v)). \quad (11)$$

If the graph is unweighted ($w(u, v) = 1, \forall (u, v) \in E$), κ_w corresponds to the curvature operator proposed in [41], [42] in the context of image restoration. More generally, κ_w is the discrete analogue of the mean curvature of the level curve of a function defined on a continuous domain of \mathbb{R}^n . This is explained by the following property, which is a rewriting of Proposition 3.

Corollary 1: The p -Laplace operator of a function $f \in \mathcal{H}(V)$ can be expressed, at a vertex $u \in V$, by:

$$\Delta_w^p f(u) = \frac{1}{2} \sum_{\substack{v \sim u \\ e=(u,v)}} \frac{\partial}{\partial e} \left(|\nabla_w f|^{p-2} \frac{\partial f}{\partial e} \right) \Big|_u. \quad (12)$$

Proof: We show that (12) is equivalent to (8). From (3), the right term of (12) is equal to:

$$\frac{1}{2} \sum_{\substack{v \sim u \\ e=(u,v)}} \sqrt{w(u, v)} \left(|\nabla_w f(v)|^{p-2} \frac{\partial f}{\partial e} \Big|_v - |\nabla_w f(u)|^{p-2} \frac{\partial f}{\partial e} \Big|_u \right).$$

By using (3) again, we obtain exactly the first equation of the proof of Proposition 3, which completes the proof. \blacksquare

Then, the p -Laplace operator (8) can be seen as an extension of Δ_w and κ_w . To avoid zero denominator in (11) when $p \leq 1$, the local variation (6) is replaced by its regularized version: $|\nabla_w f(v)|_\epsilon = \sqrt{|\nabla_w f(v)|^2 + \epsilon^2}$, where $\epsilon \rightarrow 0$ is a small fixed constant [42].

III. REGULARIZATION ON WEIGHTED GRAPHS

Let $f^0 : V \rightarrow \mathbb{R}$ be a given function defined on the vertices of a weighted graph $G = (V, E, w)$. In a given context, the function f^0 represents an observation of a clean function $g : V \rightarrow \mathbb{R}$ corrupted by a given noise n such that $f^0 = g + n$. Such noise is assumed to have zero mean and variance σ^2 , which usually corresponds to observation errors.

To recover the uncorrupted function g , a commonly used method is to seek for a function $f : V \rightarrow \mathbb{R}$ which is regular enough on G , and also close enough to f^0 . This inverse problem can be formalized by the minimization of an energy functional, which typically involves a regularization term plus

an approximation term (also called fitting term). In this paper, we propose to consider the following variational problem:

$$g \approx \min_{f: V \rightarrow \mathbb{R}} \left\{ E_w^p(f, f^0, \lambda) = R_w^p(f) + \frac{\lambda}{2} \|f - f^0\|_2^2 \right\}, \quad (13)$$

where the regularization functional $R_w^p : \mathcal{H}(V) \rightarrow \mathbb{R}$ is the discrete p -Dirichlet form of the function $f \in \mathcal{H}(V)$:

$$R_w^p(f) = \frac{1}{p} \sum_{u \in V} |\nabla_w f(u)|^p \stackrel{(6)}{=} \frac{1}{p} \sum_{u \in V} \left(\sum_{v \sim u} w(u, v)(f(v) - f(u))^2 \right)^{\frac{p}{2}}. \quad (14)$$

The trade-off between the two competing terms in the functional E_w^p , is specified by the fidelity parameter $\lambda \geq 0$. By varying the value of λ , the variational problem (13) allows one to describe the function f^0 at different scales, each scale corresponding to a value of λ .

The degree of regularity, which has to be preserved, is controlled by the value of $p > 0$. When $p = 2$, the minimizer (13) corresponds to the well-known Tikhonov regularization on weighted graphs. When $p = 1$, it is the fitted total variation, previously proposed on unweighted graphs in [41], [42]. The more general problem treated in this paper can be seen as a natural extension of these two latter cases. In the sequel, we discuss the existence of its solution, then we show that the solution is based on the p -Laplace operator (8), and we give an iterative algorithm to approximate it.

A. Existence of the Solution and Regularization Equations

When $p \geq 1$, the energy E_w^p is a convex functional of functions of $\mathcal{H}(V)$. As we have $E_w^p(f, f^0, \lambda) \rightarrow +\infty$ when $f \rightarrow +\infty$, by standard arguments in convex analysis, there exists a global minimum for the minimizer (13). Then, any local minimum of E_w^p is a global minimum of E_w^p . Let $\mathcal{C}(V)$ be the set of constant functions $f : V \rightarrow \mathbb{R}$, such that $f(u) = c$ for all $u \in V, c \in \mathbb{R}$. So, E_w^p is also a strictly convex functional of functions in $\mathcal{H}(V) \setminus \mathcal{C}(V)$. In this case, the minimization of E_w^p has a unique solution. This is also the case for functions in $\mathcal{C}(V)$. Indeed, R_w^p is translation invariant, e.g. $R_w^p(f + c) = R_w^p(f)$ for all $c \in \mathbb{R}$. Thus, there exists only one global minimum.

To get the solution of the minimizer (13), we consider the following system of equations:

$$\frac{\partial E_w^p(f, f^0, \lambda)}{\partial f(u)} = 0, \quad \forall u \in V, \quad (15)$$

which is rewritten as:

$$\frac{\partial R_w^p(f)}{\partial f(u)} + \lambda(f(u) - f^0(u)) = 0, \quad \forall u \in V. \quad (16)$$

If this system has a solution, then it is the solution of (13).

When $p < 1$, R_w^p is non-convex, and the global minimum of (13) is not insured. However, this case is also considered in this paper, in order to analyze the behavior of the associated diffusion processes beyond the usual bound $p = 1$.

The solution of the system (16) is computed by using the following property.

Proposition 4: Let f be a function in $\mathcal{H}(V)$. Then,

$$\frac{\partial R_w^p(f)}{\partial f(u)} = 2\Delta_w^p f(u), \quad \forall u \in V. \quad (17)$$

Proof: Let u_1 be a vertex of V . From (14), the u_1 -th term of the derivative of $R_w^p(f)$ is given by:

$$\frac{\partial R_w^p(f)}{\partial f(u_1)} = \frac{1}{p} \frac{\partial}{\partial f(u_1)} \left(\sum_{u \in V} \left(\sum_{v \sim u} w(u, v) (f(v) - f(u))^2 \right)^{\frac{p}{2}} \right).$$

It only depends on the edges incident to u_1 . Let v_1, \dots, v_k be the vertices of V connected to u_1 by an edge of E . Then, by using the chain rule, we have:

$$\begin{aligned} \frac{\partial R_w^p(f)}{\partial f(u_1)} = & - \sum_{v \sim u_1} w_{u_1 v} (f(v) - f(u_1)) |\nabla_w f(u_1)|^{p-2} \\ & + w_{v_1 u_1} (f(u_1) - f(v_1)) |\nabla_w f(v_1)|^{p-2} \\ & + \dots + w_{v_k u_1} (f(u_1) - f(v_k)) |\nabla_w f(v_k)|^{p-2}, \end{aligned}$$

which is equal to:

$$\sum_{v \sim u_1} w_{u_1 v} (f(u_1) - f(v)) (|\nabla_w f(u_1)|^{p-2} + |\nabla_w f(v)|^{p-2}).$$

From (8), this latter expression is exactly $2\Delta_w^p f(u_1)$. ■

By using Proposition 4, the system (16) is rewritten as:

$$2\Delta_w^p f(u) + \lambda(f(u) - f^0(u)) = 0, \quad \forall u \in V, \quad (18)$$

which is equivalent to the following system of algebraic equations (see Proposition 3):

$$\left(\lambda + \sum_{v \sim u} \gamma_w^f(u, v) \right) f(u) - \sum_{v \sim u} \gamma_w^f(u, v) f(v) = \lambda f^0(u). \quad (19)$$

This can be considered as the discrete Euler-Lagrange equation associated to the problem (13). Contrary to the continuous case, it does not involve any PDEs. To get its solution, several methods can be used, depending on the value of the regularity parameter p . In the linear case ($p = 2$), iterative methods such as the Gauss-Jacobi method, or the infinitesimal steepest descent method, converge close to the solution of (13). In the following section, we give a general method to approximate the solution for $p > 0$.

B. A Family of Discrete Diffusion Processes

We propose to use the linearized Gauss-Jacobi iterative method to solve the system (19). Let t be an iteration step, and let $f^{(t)}$ be the solution of (19) at the step t . Then, the method is given by the following algorithm:

$$\begin{cases} f^{(0)} = f^0 \\ f^{(t+1)}(u) = \frac{\lambda f^0(u) + \sum_{v \sim u} \gamma_w^{f^{(t)}}(u, v) f^{(t)}(v)}{\lambda + \sum_{v \sim u} \gamma_w^{f^{(t)}}(u, v)}, \quad \forall u \in V. \end{cases} \quad (20)$$

It describes a family of discrete diffusion processes, which is parameterized by the structure of the graph (topology and weight function), the parameter p , and the parameter λ . Also, the stopping time can be given *a priori*, or can be determined by a stopping criterion. To get the convergence of the process, a classical stopping criterion is $\|f^{(t+1)} - f^{(t)}\|_2 < \tau$, where $\tau \rightarrow 0$ is a small fixed constant.

These processes can be interpreted as forced lowpass filters on graphs. Let φ_w^f be the filter coefficients defined by:

$$\begin{cases} \varphi_w^f(u, v) = \frac{\gamma_w^f(u, v)}{\lambda + \sum_{v' \sim u} \gamma_w^f(u, v')}, & \text{if } u \neq v \\ \varphi_w^f(u, u) = \frac{\lambda}{\lambda + \sum_{v' \sim u} \gamma_w^f(u, v')}, & \text{otherwise.} \end{cases} \quad (21)$$

Then, the regularization algorithm (20) is rewritten as:

$$\begin{cases} f^{(0)} = f^0 \\ f^{(t+1)}(u) = \varphi_w^{f^{(t)}}(u, u) f^0(u) + \sum_{v \sim u} \varphi_w^{f^{(t)}}(u, v) f^{(t)}(v). \end{cases} \quad (22)$$

One can remark that φ_w^f satisfies the following properties:

$$\begin{cases} \varphi_w^f(u, u) \geq 0, \quad \forall u \in V \\ \sum_{v \sim u} \varphi_w^f(u, v) \geq 0, \quad \forall u \in V \\ \varphi_w^f(u, u) + \sum_{v \sim u} \varphi_w^f(u, v) = 1, \quad \forall u \in V. \end{cases} \quad (23)$$

At each iteration of the algorithm (22), the new value $f^{(t+1)}(u)$ depends on two quantities: the original value $f^0(u)$, and a weighted average of the filtered values of $f^{(t)}$ in a neighborhood of u .

Proposition 5: For all $p \in [1, +\infty)$, if the algorithm (20), or equivalently (22), converges, then it converges to the solution of the minimizer (13). Moreover, when $\lambda = 0$, it converges.

Proof: From (23) we can observe that $f^{(t+1)}(u)$ is a convex combination of $f^0(u)$ and $f^{(t)}(v)$, for all $t \geq 0$. Thus we have:

$$\begin{aligned} f^{(t+1)}(u) & \geq \min \left\{ f^0(u), \min_{v \sim u} f^{(t)}(v) \right\}, \\ f^{(t+1)}(u) & \leq \max \left\{ f^0(u), \max_{v \sim u} f^{(t)}(v) \right\}. \end{aligned}$$

Then, by recursion, an iteration step $t \geq 0$ of (22) satisfies the minimum-maximum principle:

$$\min_{v \in V} f^0(v) \leq f^{(t)}(u) \leq \max_{v \in V} f^0(v), \quad \forall u \in V.$$

Let $\mathcal{M}(V)$ be the set of functions $f : V \rightarrow \mathbb{R}$ such that $\|f\|_\infty \leq \|f^0\|_\infty$, where $\|f\|_\infty = \max_{v \in V} f(v)$ is the infinity norm. From (22), an iteration can be written as $f^{(t+1)} = \phi(f^{(t)})$, with $\phi : \mathcal{M}(V) \rightarrow \mathcal{M}(V)$. Then the minimum-maximum principle stability implies that $\phi(\mathcal{M}(V)) \subseteq \mathcal{M}(V)$. By considering that $\lambda \neq 0$, then each component $\phi(u) : \mathcal{M}(V) \rightarrow \mathbb{R}$ is continuous with respect to $\|\cdot\|_\infty$, for all $u \in V$. This implies that ϕ is a continuous mapping. As $\mathcal{M}(V)$ is non-empty, compact and convex, the Schauder fixed point theorem [43] shows that there exists $f \in \mathcal{M}(V)$ which satisfies $f = \phi(f)$. Then, if the algorithm (22) converges, it is to the limit function f which satisfies (19). Since for $p \geq 1$, (19) is the solution of the minimizer (13), (22) also converges to the solution of (13).

When $\lambda = 0$, we can remark that an iteration of the diffusion process (22) can be written in matrix form as $f^{(t+1)} = Q(f^{(t)})f^{(t)}$, where Q is a stochastic matrix (nonnegative, symmetric, unit row sum). Then, as demonstrated in [44],

this discrete diffusion process satisfies the minimum-maximum principle, and converges to a constant as $t \rightarrow +\infty$. ■

Remark 2: From Proposition 5, when $p < 1$ (non-convex case), if the algorithm (22) (or equivalently (20)) converges to some function $f \in \mathcal{H}(V)$, the latter is not insured to be the global minimum of the minimizer (13).

IV. REGULARIZATION OF FUNCTIONS ON DISCRETE DATA

The minimization problem (13), and the discrete diffusion processes (20), can be used to regularize any function defined on a finite set $V = \{v_1, \dots, v_N\}$ of discrete data $v_i \in \mathbb{R}^m$. This is achieved by constructing a weighted graph $G = (V, E, w)$, and by considering the function to be regularized as a function $f^0 : V \rightarrow \mathbb{R}$, defined on the vertices of G .

A. Construction of Graphs

There exist several popular methods that transform the set V , with a given pairwise distance measure $\mu : V \times V \rightarrow \mathbb{R}_+$, into a neighborhood graph (or similarity graph). Constructing such a graph consists in modeling neighborhood relationships between data. Among the existing graphs, the simplest of them is the δ -neighborhood graph, where two data $v_i, v_j \in V$ are connected by an edge of E if $\mu(v_i, v_j) \leq \delta$, with $\delta > 0$. We can also quote the minimum spanning tree, the k -nearest neighbors graph, the Delaunay triangulation, or the relative neighborhood graph, as other possible graph topologies (see [45] for a survey on neighborhood graphs used in pattern recognition problems). The choice of the graph topology enables us to perform several regularization processes that can integrate local, semi-local or nonlocal interactions between the data. These interactions are said to be fully nonlocal if the constructed graph is the complete graph $G_c = (V, E, w)$, with $E = V \times V \setminus \{(v_i, v_i) : v_i \in V\}$.

The weights are computed according to a measure of similarity $g : V \times V \rightarrow \mathbb{R}$, which satisfies:

$$w(u, v) = \begin{cases} g(u, v) & \text{if } (u, v) \in E \\ 0 & \text{otherwise} \end{cases} \quad (24)$$

This measure can simply be defined as the inverse of the distance measure: $g(u, v) = \mu(u, v)^{-1}$. Distances between data are estimated by comparing their features which generally depend on the initial function f^0 and on the set V . To this aim, every data $u \in V$ is assigned with a feature vector denoted by $F(f^0, u) \in \mathbb{R}^q$. Several choices can be considered for the expression of F , depending on the nature of the features to be preserved during the regularization. In the simplest case, one can consider $F(f^0, u) = f^0(u)$. It also can be computed from the values of f^0 , taken in the neighborhood of u , and from the data in that neighborhood. Therefore, the weight function w , associated to a given graph, can naturally incorporate local, semi-local or nonlocal features according to the topology of the graph. For instance, one can consider the three following general weight functions:

- 1) $g_1(u, v) = \frac{1}{\varepsilon + \mu(F(f^0, u), F(f^0, v))}$,
- 2) $g_2(u, v) = \exp\left(-\frac{\mu(F(f^0, u), F(f^0, v))^2}{\sigma^2}\right)$,
- 3) $g_3(u, v) = g_2(u, v) \exp\left(-\frac{\mu_V(u, v)^2}{2\sigma_V^2}\right)$,

where σ_V and σ are two parameters depending respectively on the variations of the function μ_V and the function μ over the graph. These parameters are usually given *a priori*. Also, they can be computed automatically from the topology of the graph and the initial function f^0 . In particular, σ can be computed locally at each vertex of the graph (see [31] and references therein).

The graph topology, associated with one of the above weight functions, describes a general family of regularizers. By changing the graph topology and the edge weights, we naturally obtain an expression of local, semi-local or nonlocal processing methods which is embedded in the graph structure. Indeed, once edges are added between vertices, they are considered as direct neighbors and the processing is local over the graph. From these considerations, we show that with an adapted graph topology and a given appropriate weight function, the proposed family is linked to several filters defined in the context of image and mesh processing.

B. Related Works in Image Processing

First, we analyze the case of $\lambda = 0$, e.g. without approximation term. In this case, when $p = 2$, the proposed diffusion (20) is linear and becomes:

$$\begin{cases} f^{(0)} = f^0 \\ f^{(t+1)}(u) = \frac{\sum_{v \sim u} w(u, v) f^{(t)}(v)}{\sum_{v \sim u} w(u, v)}, \quad \forall u \in V. \end{cases} \quad (25)$$

As described in Section III-A, when this latter diffusion is iterated until convergence, the function f is the solution of the heat equation $\Delta_w f(u) = 0$, for all $u \in V$. Moreover, from Proposition 5, f is also the solution of the minimization of the Dirichlet energy $R_w^2(f) = \sum_{u \in V} |\nabla_w f(u)|^2$. By considering a fixed number of iterations, the diffusion (25) performs Laplacian smoothing, which is a general neighborhood filter used in image processing. Also, the rewriting of (25) in matrix form leads to Markov matrix filtering and spectral graph filtering [40], [46], [24], [30]. Many specific filters are related to the diffusion (25). In particular, one iteration of the diffusion process (25), associated with the weight function g_3 and the scalar feature $F(f^0, \cdot) = f^0$, is equivalent to the bilateral filter introduced in the context of image denoising [47], [48]. For this filter, the distance functions μ and μ_V are Euclidean distances. Another important case is obtained by using g_3 and the feature vector $F(f^0, v) = [f^0(u) : u \in B_{v,s}]^T$, where $B_{v,s}$ is a bounding box of size $(2s+1) \times (2s+1)$ centered at v . In this case, one iteration of (25) corresponds to the nonlocal means (NLMeans) filter [1] (see also [23] and [24]). For this latter filter, the function μ is the Euclidean distance between feature vectors, weighted by a Gaussian kernel. Bilateral and NLMeans filters have also been redefined to denoise meshes [49], [50], [51]. These two filters, and all the filters described by (25), are generalized by the proposed method: by the regularity parameter p , and by the scale parameter λ .

When $\lambda \neq 0$ and w is constant over the set of edges (unweighted graph), the diffusion (20) corresponds exactly to the digitized PDE filters proposed in the context of image

restoration [41], [42]. If $p = 1$, it is the fitted TV regularization; and if $p = 2$, it is the Tikhonov regularization.

Table I resumes how our framework includes and extends some above-mentioned discrete models used in image processing. Table I reads in the following way. The first three columns present four different configurations given by the parameters λ and w , and the graph topology. Once these latter parameters are fixed, one can then consider different values of p (next columns in the row of each given configuration).

λ	w	Graph	$p = 1$	$p = 2$	$p \in]0, 1[$
0	g_3 ,	semi-local	Our	Bilateral[47]	Our
0	g_2	nonlocal	Our	NLMMeans [1]	Our
$\neq 0$	constant	local	TV Digital[42]	L_2 Digital	Our
$\neq 0$	any	nonlocal	Our	Our	Our

TABLE I

WORKS RELATED TO OUR FRAMEWORK IN IMAGE PROCESSING.

C. Related Works in Local and Nonlocal Continuous Regularization

The regularization of a given function $f^0 : V \rightarrow \mathbb{R}$, on a complete graph G_c , takes into account fully nonlocal interactions between the data of V , through the local variations $|\nabla_w f(u)|$ that are involved in the discrete p -Dirichlet form:

$$\begin{aligned} R_w^p(f) &\stackrel{(14)}{=} \frac{1}{p} \sum_{u \in V} |\nabla_w f(u)|^p \\ &= \frac{1}{p} \sum_{u \in V} \left(\sum_{v \in V} w(u, v) (f(v) - f(u))^2 \right)^{\frac{p}{2}}. \end{aligned} \quad (26)$$

One can observe that this last functional is exactly the same as the functional (14), where $\sum_{v \sim u}$ can be replaced by $\sum_{v \in V}$ ($w(u, v) = 0$ if $(u, v) \notin E$). This shows that local, semi-local and nonlocal regularizations on graphs have the same formulation. One can remark that the corresponding continuous analogue of the above discrete regularization functionals, which considers only local interactions, is:

$$J_w^p(f) = \frac{1}{p} \int_{\Omega} |\nabla f|^p dx,$$

where $f : \Omega \rightarrow \mathbb{R}$ is a continuous function defined in a bounded domain $\Omega \subset \mathbb{R}^m$. Similarly, the discrete diffusion process (26) can also be rewritten in a continuous setting which uses nonlocal interactions:

$$J_w^p(f) = \frac{1}{p} \int_{\Omega} \left(\int_{\Omega} w(x, y) (f(y) - f(x))^2 dy \right)^{\frac{p}{2}} dx.$$

In particular, for $p = 2$ and $p = 1$, it corresponds to the nonlocal functionals introduced in [27] and [28] in the context of image processing:

$$\begin{aligned} J_w^2(f) &= \frac{1}{2} \int_{\Omega \times \Omega} w(x, y) (f(y) - f(x))^2 dx dy \\ J_w^1(f) &= \int_{\Omega} \left(\int_{\Omega} w(x, y) (f(y) - f(x))^2 dy \right)^{\frac{1}{2}} dx \end{aligned}$$

Moreover, in [52], we show that, with the same gradient operator, replacing R_w^p by a p -norm leads to the following expression:

$$R_w^p = \frac{1}{2p} \sum_{u \in V} |\nabla_w f(u)|^p = \sum_{v \sim u} w(u, v)^{\frac{p}{2}} |f(v) - f(u)|^p,$$

which can be considered as the discrete analogue of the recently proposed continuous nonlocal anisotropic functionals [27], [28].

V. APPLICATIONS

The family of filters proposed in Section III can be used to regularize any function defined on the vertices of a graph, or on any discrete data set.

For any of these different applications, one wants to regularize a function $f^0 : V \subset \mathbb{R}^n \rightarrow \mathbb{R}^m$, where $f^0(u) = [f_1^0(u), \dots, f_m^0(u)]^T$ and $f_i^0 : V \rightarrow \mathbb{R}$ is the i -th component of $f^0(u)$. When the regularization operates on vector-valued functions, one regularization process per vector component is considered. Each component is processed independently and this comes to have m independent iterative regularization schemes:

$$\begin{cases} f_i^{(0)} = f_i^0 \\ f_i^{(t+1)}(v) = \frac{f_i^0(v) + \sum_{u \sim v} \gamma_i^{(t)}(u, v) f_i^{(t)}(u)}{\lambda + \sum_{u \sim v} \gamma_i^{(t)}(u, v)} \end{cases} \quad (27)$$

The p -Laplace operator is different for each component and one has

$$\gamma_{f_i^{(t)}}^{f_i^{(t)}}(u, v) = w(u, v) (|\nabla_w f_i^{(t)}(v)|^{p-2} + |\nabla_w f_i^{(t)}(u)|^{p-2}). \quad (28)$$

for the i^{th} component. Applying the regularization in a component-wise manner is interesting to develop a computationally efficient solution. However, component-wise processing of vector-valued data can have serious drawbacks contrary to vector processing solutions. To overcome this limitation, a regularization process acting on vector-valued data needs to be driven by equivalent geometric attributes, taking the coupling between vector components into account [53], [54]. Therefore, component-wise regularization does not have to use different local geometries but a vector one. In the case of $p = 2$ the p -Laplace operator is the same for all the components, but in the case of $p \neq 2$ it is different. As we have just mentioned, this is not interesting. Indeed, in this case, the m regularization processes can be totally independent if $w(u, v)$ does not incorporate any inter-component information resulting in no coupling between the m independent regularizations. To overcome this limitation and in order to take into account the inner correlation aspect of vectorial data, the p -Laplace operator is considered as being the same for the m component regularizations (component coupling). This is achieved by using the same vectorial local variation defined by:

$$|\nabla_w f^{(t)}(v)| = \sqrt{\sum_{i=1}^m |\nabla_w f_i^{(t)}(v)|^2}, \quad (29)$$

where $|\nabla_w f^{(t)}(v)|$ replaces $|\nabla_w f_i^{(t)}(v)|$, for all i in (28). The inner correlation is also considered by using vectorial weight functions, such as the ones defined in Section IV-A.

In the sequel, we provide examples of obtained results with the proposed framework for the simplification of images (scalar or vectorial), polygonal curves, meshes, manifolds, and databases.

Scalar image smoothing: Let f^0 be a scalar image of N pixels, with $f^0 : V \subset \mathbb{Z}^2 \rightarrow \mathbb{R}$ which defines a mapping from the vertices to gray levels. We illustrate the abilities of the proposed framework for smoothing scalar images. All the results are shown at convergence. To this aim, Figure 1 shows sample results obtained with different parameters of the proposed regularization: different values of p , different values of λ and different weight functions. The first line of Figure 1 presents the initial image (first column) and the computed gradient on the initial image with constant weights ($w = 1$, second column), with computed weights in local (third column, $\sigma = 20$ for $w = g_2$) or nonlocal configurations (fourth column, $\sigma = 4$ for $w = g_2$). For the nonlocal configuration, we consider a 19×19 neighborhood window and a 11×11 patch as a feature vector. In the table part of Figure 1, the first two grouped columns present local regularizations with constant weights ($w = 1$) and with different values of λ (provided above each column). For these cases, the considered graph is a 4-adjacency grid graph and the results obtained for $p = 1$ or $p = 2$ correspond exactly to the digital TV and L_2 filters of Chan and Osher [41], [42]. In the table part of Figure 1, the grouped third and fourth columns present results obtained with the proposed framework with the same parameters but with computed weights ($w = g_2$). For these cases, the considered graph is a 8-adjacency grid graph. The first thing to consider is that our approach is a direct extension of [41], [42] to weighted graphs that enables the production of better visual results by using computed edge weights. The second thing to notice is that a low value of p enables a better preservation of edges and fine structures. Moreover, it tends leveling the image in flat zones. Changing the value of parameter λ enables us to obtain solutions either close or far from the original image. All these noticed points become more evident once one uses a nonlocal configuration which enhances once again the preservation of edges and the production of flat zones. Nonlocal results are provided in last column of the table part of Figure 1. This shows how our proposed frameworks generalizes the NLmeans filter [1].

Vectorial image smoothing and denoising: Let f^0 be a color image of N pixels, $f^0 = [f_1^0, \dots, f_N^0]^T$ with $f^0 : V \subset \mathbb{Z}^2 \rightarrow \mathbb{R}^3$ which defines a mapping from the vertices to a vector of color channels. Figure 2 presents several results of image denoising on an original image (Figure 2(a)) corrupted by Gaussian noise ($\sigma = 15$, PSNR between original and corrupted images is 85.04db, Figure 2(b)). Results are shown at convergence. Figures 2(c), (f), (i) present results of a local processing, for $p = 2$, $p = 1$ and $p = 0.7$, on a 8-adjacency grid graph with edges weighted

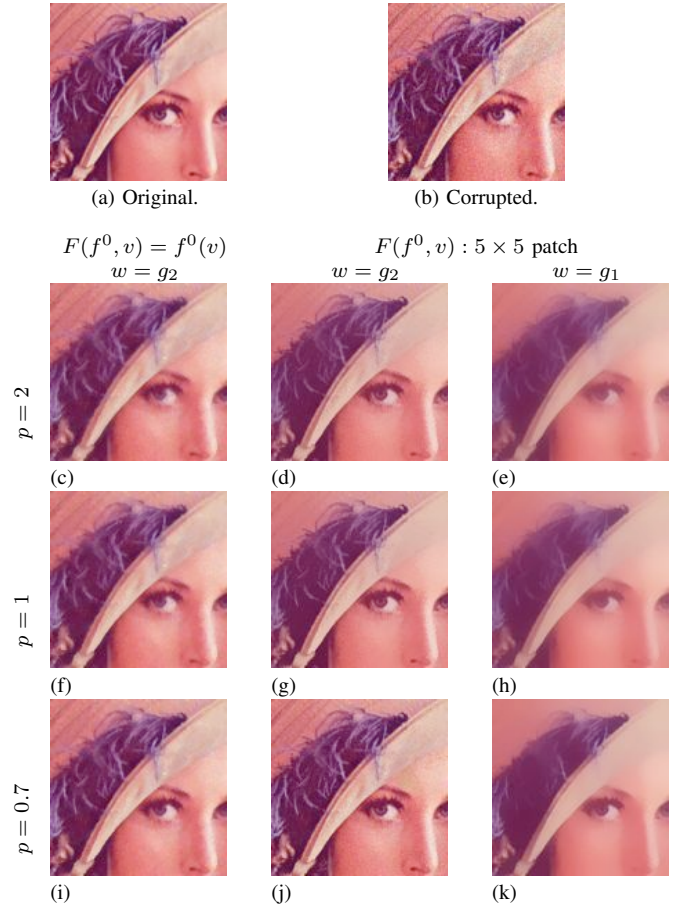


Fig. 2. Image denoising illustration on an original color image (Figure 2(a)) corrupted by Gaussian noise (Figure 2(b)) with $\sigma = 15$, PSNR between original and corrupted images is 85.04db. Results are provided for different values of p for local (Figures 2(c), (f), (i)) with $w = g_2$ and a 3×3 neighborhood) and nonlocal processing (Figures 2(d), (g), (j)) for $w = g_2$, Figures 2(e), (h), (k) for $w = g_1$, both with a 11×11 neighborhood and 5×5 patch).

by g_2 and $F(f^0, v) = f^0(v)$. This extends the vectorial digital TV filter [42] to weighted graphs as mentioned above for scalar images. Figures 2(d), (g), (j) present results of a nonlocal processing, for $p = 2$, $p = 1$ and $p = 0.7$, on a 120-adjacency grid graph (a search window of size 11×11) with edges weighted by g_2 and $F(f^0, v)$ defined on a patch of size 5×5 . Figures 2(d), (g), (j) present results with the same nonlocal parameters except that $w = g_1$. Parameters λ and σ (for $w = g_2$) are automatically estimated (see in [31] for further details). A visual analysis of the results shows that values of $p \leq 1$ enable a better preservation of edges while increasing the graph connectivity and using patches for similarities enable to better preserve some details. Figures 2(i)-(k) show the importance of the similarity measure for computing the edges weights. Indeed, with $w = g_1$, an over-smoothing effect is obtained without preserving details. This confirms that this is not a nonlocal view but the use of an appropriate similarity measure which enables the preservation of fine structures.

Image simplification: On the contrary to classical image

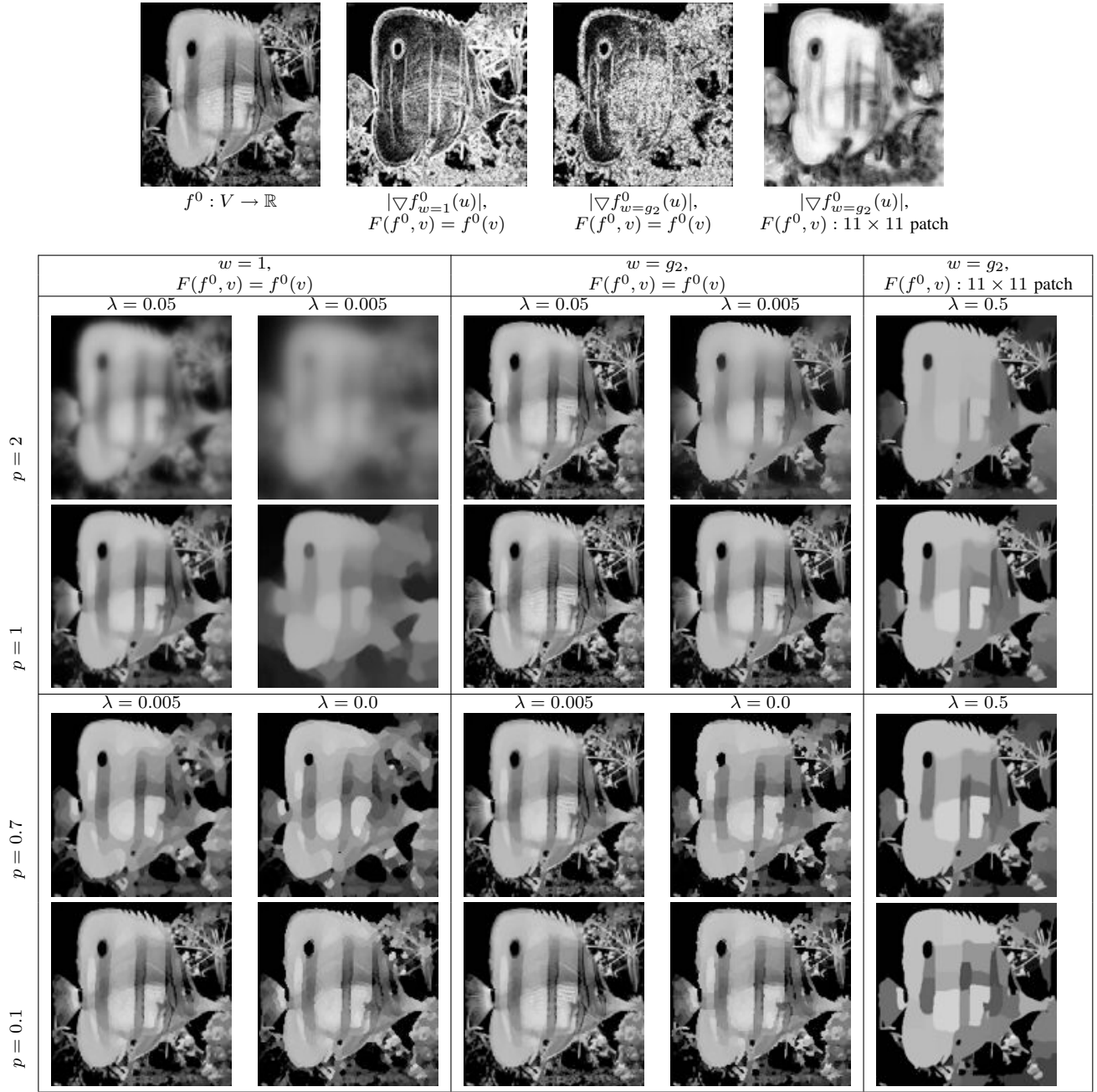


Fig. 1. Scalar image simplification on a gray level image with different parameters values of p , λ and different weight functions in local or nonlocal configurations. First row presents the original image and the gradient in different configurations. Next rows present simplification results where each row corresponds to a value of p (see text for more details).

simplification which considers grid graphs, one can simplify an image by first considering a fine partition of this image (or over-segmentation), where the pixel values of each region of the partition are replaced by the mean or the median pixel value of this region. The partition can be associated with a Region Adjacency Graph (RAG), where vertices represent regions and where edges link adjacent regions. Let $G = (V, E, w)$ be a RAG. Let $f^0 : V \subset \mathbb{Z}^2 \rightarrow \mathbb{R}^m$ be a mapping from the vertices of G to the mean or median value of their regions. Then, the simplification is achieved by regularizing the function f^0 on G . Moreover, we can build an irregular pyramid of partitions by alternating simplification

(only one iteration) and graph decimation since one iteration of the simplification brings neighbor vertices to similar models. We present the results of a local simplification performed with different values of p and λ after $t = 5$ iterations of sequential graph filtering and graph decimation. Edges of the RAG are weighted by g_1 . The graph is decimated along the iterations by merging regions the difference of which is lower than a given arbitrary threshold of 2 in our experiments. First row of Figure 3 presents the original image, its fine partition (obtained by the concept of homogeneous zones [55]) and the associated color median image ($m = 3$). When a color median image is presented in Figure 3, we

provide the PSNR between the original and the color median image. When a partition is presented in Figure 3, we provide the number of vertices of the RAG. Last rows of Figure 3 present results (color median and partition images) where each row corresponds to a value of p for two different values of λ . When $\lambda = 0$, a small value of p enables a better preservation of the initial image content with a lower number of vertices. When $\lambda \neq 0$, a small value of p also enables a good preservation of the initial image content but with a higher number of vertices. It is worth noting that merging similar regions to decimate the RAG during the simplification enables to accelerate the simplification process. Therefore, the use of the RAG instead of the classical grid-graph enables a faster processing [29], [56] since there are less vertices as compared to the number of pixels. Similar works based on RAGs have also been proposed [57], [58].

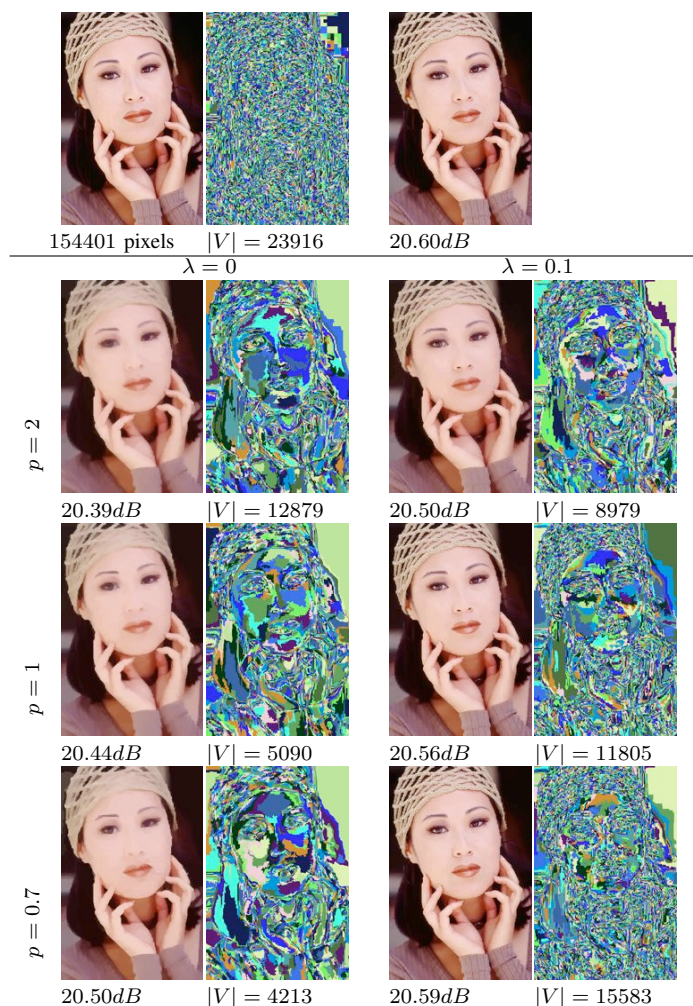


Fig. 3. Image simplification illustration. From an original image, one computes a pre-segmentation and the associated color median image (images in first row). Regularization and decimation are applied on the RAG of the pre-segmentation and simplified region maps and color median images are obtained for different values of λ and p (next rows).

Polygonal curve and surface simplification: By nature, polygonal curves and surface meshes have a graph structure. First, we consider the case of a polygonal curve represented

by a graph: $f^0 : V \subset \mathbb{R}^2 \rightarrow \mathbb{R}^2$ and edges are weighted by a constant $w = 1$. Figure 4 presents results of filter (27) on a polygonal curve. The behavior is studied and illustrated in Figure 4 for different values of parameters p and λ for a local regularization. First, as attended, acting on parameter λ enables to reach different levels of simplification. One important thing to note here is that the presented graphs all have the same number of vertices: simplification enables to groups similar vertices around high curvature regions. The second thing to point out is that acting on p enables to perform a denoising of the original graph but also to smooth (for $p \geq 1$) or to preserve sharp angles (for $p < 1$).

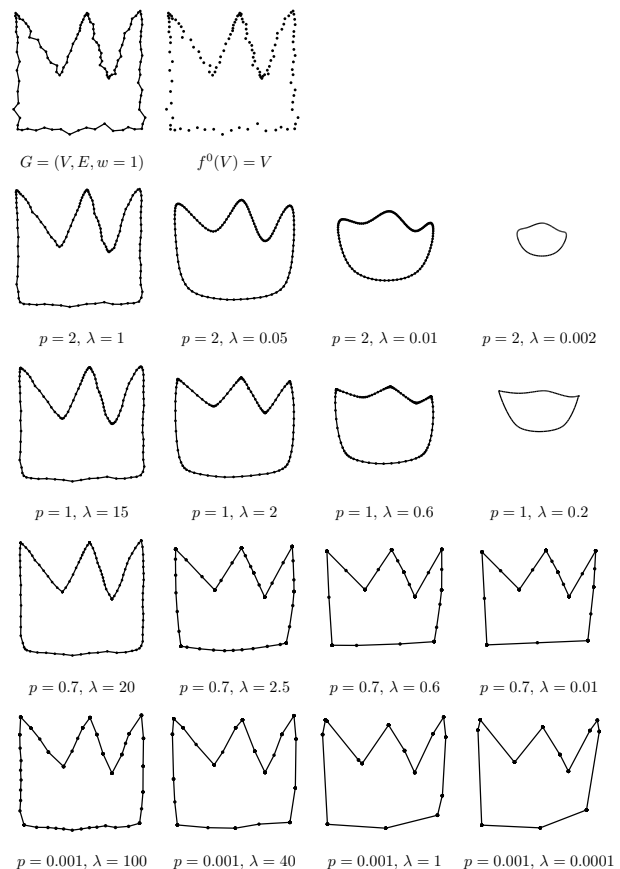


Fig. 4. Behavior of the regularization of a polygonal curve G (first row, first column), by using the discrete diffusion process (until convergence). First row presents the graph constructed over an initial set of points. Next rows present simplification results where each row corresponds to a value of p . For each p , the four regularizations are obtained with decreasing values of the scale parameter $\lambda > 0$. The original mesh ($|V| = 24930$) is from Shape Repository, Aim@Shape Project (<http://shapes.aim-at-shape.net>).

Second, we consider the case of surface meshes. Let V be the set of mesh vertices, and let E be the set of mesh edges. If the input mesh is noisy, we can regularize vertex coordinates or any other function $f^0 : V \subset \mathbb{R}^3 \rightarrow \mathbb{R}^3$ defined on the graph $G = (V, E, w)$. Results of filter (27) are given in Figure 5 for triangular meshes with a local regularization for edges weighted by function g_1 . We can observe the same key points as with polygonal curves. Moreover, the shrinkage effect obtained with $p = 1$ or $p = 2$ when $\lambda \rightarrow 0$ is highly

reduced with low values of p which shows the benefits of our approach (see the cropped and zoomed areas in three last columns of the first row in Figure 5). As for image simplification on a RAG, one could also merge vertices along the iterations as they become similar.

Manifold smoothing: Graph-based methods have recently emerged as a powerful tool for analyzing high dimensional data that has been sampled from a low dimensional submanifold [17], [20]. These methods begin by constructing a graph in which the nodes represent input patterns and the edges represent neighborhood relations. The resulting graph (assumed connected) can be viewed as a discretized approximation of the submanifold sampled by the input patterns. Let $G = (V, E, w)$ be a graph defined over the manifold. Typical graphs are the complete graph and the k -nearest neighbor graph. Typical manifolds being image libraries, we consider the USPS handwritten digit database for illustration. Each digit is a 16×16 image which is considered as a vector of 256 dimensions. Let $f^0 : V \rightarrow \mathbb{R}^{16 \times 16}$ be a mapping from the vertices of G to the elements of the manifold. We consider the complete graph weighted by function g_2 and constructed over the manifold. To each vertex is associated a feature vector representing each digit. The filter (27) is considered (256 parallel coupled regularization processes are performed) with $p = 2$ and different amounts of data attachment. Since the complete graph is considered, the regularization is naturally nonlocal. Figure 6 presents regularization results on 200 digits from the USPS database. Without the data term, the manifold reduces to its main digit which is an artificial one since the manifold is smoothed. By increasing the data term, the regularized manifold remains closer to the original one. Such manifold regularization can be useful for classification purposes on a noiseless submanifold [25] extracted from a noisy manifold.

Data smoothing: Given any data, the latter can be associated with a graph by considering a similarity measure. Therefore, any database can be regularized with our method as long as it is associated with a graph structure. To show the efficiency of such processing, we consider two well-known databases: Iris and Ionosphere [59]. Iris database contains 3 classes of samples in 4 dimensions with 50 samples in each class. Ionosphere database contains 2 classes of samples in 34 dimensions with for each class 225 and 126 samples. Figure 7(a) presents a pairwise feature projection of the Iris database. Due to the high dimensionality of Ionosphere, only few relevant feature pairs projections are shown. With such real-world databases, some noise is present and data smoothing is therefore of interest. Let $G = (V, E, w)$ be a graph defined over the data. Let $f^0 : V \rightarrow \mathbb{R}^m$ be a mapping from the vertices of G to the elements of the database ($m = 4$ for Iris and $m = 34$ for Ionosphere). We consider a complete graph and regularization is performed with $p = 2$ and $\lambda = 0.01$ after $t = 10$ iterations. Figure 7(b) presents the regularization result on original Iris data depicted in Figure 7(a). Figure 7(d) presents the regularization result on original Ionosphere data depicted in Figure 7(c). One can see the benefits of the regularization: input points which belong to the same class tend to be closer

than in the original database. Therefore, this is an efficient method to map input points into a regularized space where clusters are more easily separable: the submanifold where the data lies has been recovered. This effect is illustrated by the results obtained with a standard k -means classification on the original and the regularized versions: the processing enables to increase the recognition rate.

VI. CONCLUSION

We proposed a general discrete framework for regularizing real-valued or vector-valued functions on weighted graphs of arbitrary topology. The regularization, based on the p -Laplace operator, leads to a family of nonlinear iterative filters. This family includes the TV digital filter, the nonlocal means filter and the bilateral filter, widely used in image processing. Also, the family is linked to spectral graph filtering and is the discrete analogue of recent continuous nonlocal regularizations.

The choice of the graph topology and the choice of the weight function enable one to regularize any discrete data set or any function on it. Indeed, the data can be structured by neighborhood graphs weighted by functions depending on data features. This can be applied in the contexts of image smoothing, denoising or simplification. We also show that mesh smoothing and denoising can be performed by the same filtering process. Similarly, manifolds can be processed by the same means to recover a noiseless submanifold from a noisy manifold.

ACKNOWLEDGEMENTS

The authors thank the reviewers for their valuable comments and suggestions to improve the quality of the paper. The authors would also like to thank Vinh Thong Ta for providing the results on manifold and data smoothing.

REFERENCES

- [1] A. Buades, B. Coll, and J.-M. Morel, "A review of image denoising algorithms, with a new one," *Multiscale Modeling and Simulation*, vol. 4, no. 2, pp. 490–530, 2005.
- [2] T. Chan and J. Shen, *Image Processing and Analysis - variational, PDE, wavelets, and stochastic methods*. SIAM, 2005.
- [3] Y.-H. R. Tsai and S. Osher, "Total variation and level set methods in image science," *Acta Numerica*, vol. 14, pp. 509–573, mai 2005.
- [4] L. Alvarez, F. Guichard, P.-L. Lions, and J.-M. Morel, "Axioms and fundamental equations of image processing," *Archive for Rational Mechanics and Analysis*, vol. 123, no. 3, pp. 199–257, 1993.
- [5] G. Taubin, "Geometric signal processing on polygonal meshes," in *Eurographics, State of the Art Report*, 2000.
- [6] Y. Ohtake, A. Belyaev, and I. Bogaeski, "Mesh regularization and adaptive smoothing," *Computer-Aided Design*, vol. 33, pp. 789–800, 2001.
- [7] M. Desbrun, M. Meyer, P. Schröder, and A. Barr, "Anisotropic feature-preserving denoising of height fields and bivariate data," *Graphics Interface*, 2000.
- [8] C. Bajaj and G. Xu, "Anisotropic diffusion of surfaces and functions on surfaces," *ACM Trans. on Graph.*, vol. 22, no. 1, pp. 4–32, 2003.
- [9] K. Hildebrandt and K. Polthier, "Anisotropic filtering of non-linear surface features," *Eurographics 2004: Comput. Graph. Forum*, vol. 23, no. 3, pp. 391–400, 2004.
- [10] I. Eckstein, J.-P. Pons, Y. Tong, C. C. J. Kuo, and M. Desbrun, "Generalized surface flows for mesh processing," in *Eurographics Symposium on Geometry Processing*, 2007.
- [11] G. Taubin, "A signal processing approach to fair surface design," in *SIGGRAPH*, 1995, pp. 351–358.

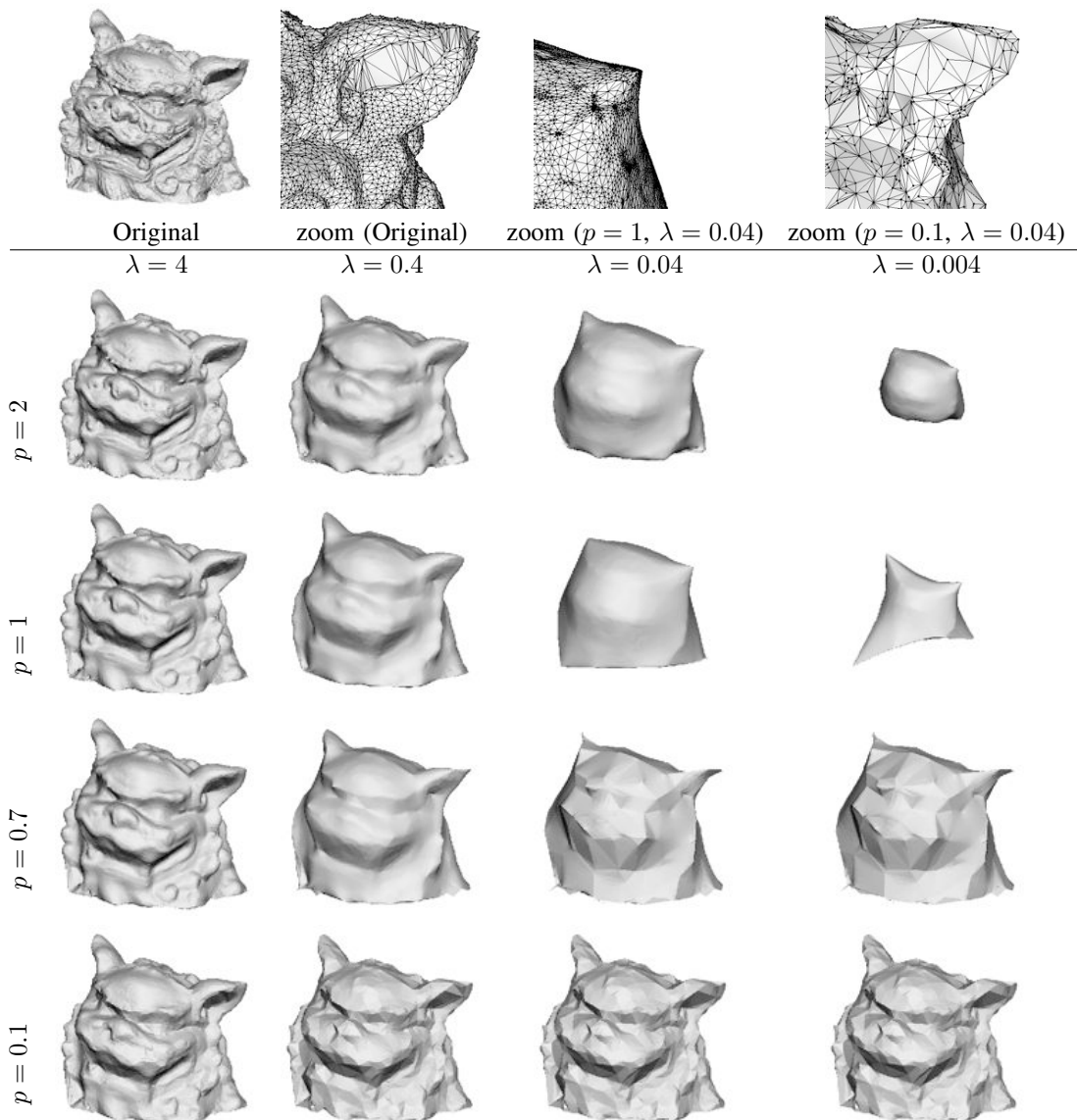


Fig. 5. Behavior of the regularization of a triangular mesh (first row, first column), by using the discrete diffusion process (until convergence). First line presents the original mesh (first column), a zoom on a part of this mesh (the ear of the lion) before (second column) and after regularization for $p = 1$ and $p = 0.1$ with $\lambda = 0.04$. Next rows present simplification results where each row corresponds to a value of p . For each p , four regularizations are obtained with decreasing values of the scale parameter $\lambda > 0$.

- [12] M. Desbrun, M. Meyer, P. Schröder, and A. H. Barr, “Implicit fairing of irregular meshes using diffusion and curvature flow.” *SIGGRAPH*, 1999, pp. 317–324.
- [13] G. Xu, Q. Pan, and C. L. Bajaj, “Discrete surface modelling using partial differential equations.” *Computer Aided Geometric Design*, vol. 23, no. 2, pp. 125–145, 2006.
- [14] U. Pinkall and K. Polthier, “Computing discrete minimal surfaces and their conjugates.” *Experimental Mathematics*, vol. 2, no. 1, 1993.
- [15] M. Belkin and P. Niyogi, “Laplacian eigenmaps for dimensionality reduction and data representation.” *Neural Computation*, vol. 15, no. 6, pp. 1373–1396, 2003.
- [16] S. Lafon, Y. Keller, and R. Coifman, “Data fusion and multicue data matching by diffusion maps.” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 11, pp. 1784–1797, 2006.
- [17] S. Lafon and A. Lee, “Diffusion maps and coarse-graining: A unified framework for dimensionality reduction, graph partitioning and data set parameterization.” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 9, pp. 1393–1403, 2006.
- [18] U. von Luxburg, “A tutorial on spectral clustering.” *Statistics and Computing*, vol. 17, no. 4, pp. 395–416, 2007.
- [19] J. Shi and J. Malik, “Normalized cuts and image segmentation.” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 888–905, 2000.
- [20] M. Belkin, I. Matveeva, and P. Niyogi, “Regularization and semi-supervised learning on large graphs,” in *COLT*, 2004, pp. 624–638.
- [21] D. Zhou and B. Schölkopf, “A regularization framework for learning from graph data,” in *ICML Workshop on Statistical Learning and Its Connections to Other Fields*, 2004, pp. 132–137.
- [22] A. J. Smola and R. Kondor, “Kernels and regularization on graphs,” in *COLT: Proc of the 16th Annual Conf. on Computational Learning Theory*, B. Schölkopf and M. Warmuth, Eds., 2003, pp. 144–158.
- [23] G. Peyré, “Manifold models for signals and images,” ceremade, preprint 17, 2007.
- [24] A. Szlam, M. Maggioni, and R. Coifman, “A general framework for adaptive regularization based on diffusion processes on graphs,” Yale, Tech. Rep., 2006.
- [25] M. Hein and M. Maier, “Manifold denoising,” in *Advances in Neural Information Processing Systems*, vol. 20, 2006, pp. 1–8.
- [26] S. Kindermann, S. Osher, and P. W. Jones, “Deblurring and denoising of images by nonlocal functionals,” UCLA, Tech. Rep., 2005.
- [27] G. Gilboa and S. Osher, “Nonlocal linear image regularization and



(a) Original USPS handwritten digit 0 database.

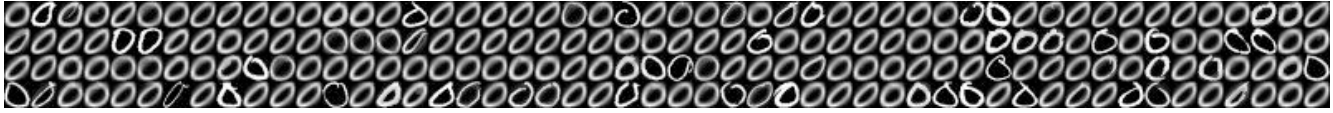
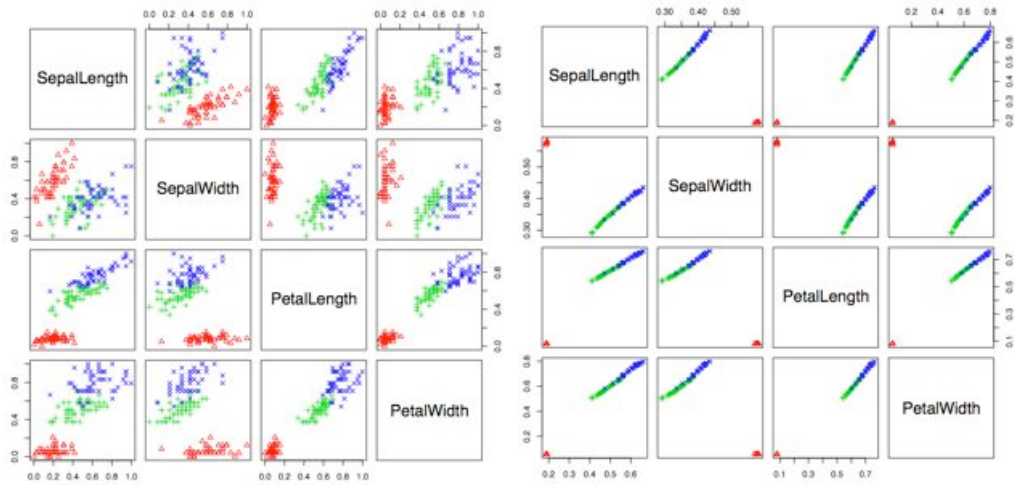
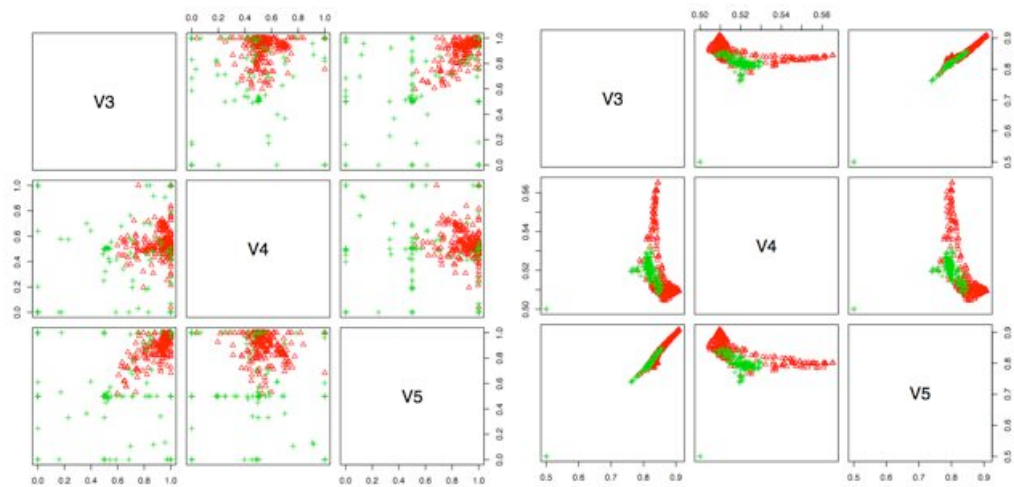
(b) Nonlocal regularization with $\lambda = 0.01$.(c) Nonlocal regularization with $\lambda = 0$.

Fig. 6. Nonlocal manifold regularization ($p = 2$) for different values of λ . The original manifold is the digit 0 from the USPS handwritten digit database. See text for details.



(a) Original Iris database (Recognition rate: 85.3%).

(b) Regularized Iris database with $w = g_2$ (Recognition rate: 95.3%).

(c) Original Ionosphere database (Recognition rate: 71.8%).

(d) Regularized Ionosphere database with $w = g_1$ (Recognition rate: 75.4%).

Fig. 7. Nonlocal Data base regularization ($p = 2$, $\lambda = 0.01$, $t = 10$) with recognition rates obtained with a k -means (averaged over 50 runs).

- supervised segmentation,” *SIAM Multiscale Modeling and Simulation*, vol. 6, no. 2, pp. 595–630, 2007.
- [28] —, “Nonlocal operators with applications to image processing,” UCLA, Tech. Rep. CAM Report 07–23, July 2007.
- [29] O. Lezoray, A. Elmoataz, and S. Boughleux, “Graph regularization for color image processing,” *Computer Vision and Image Understanding (CVIU)*, vol. 107, no. 1-2, pp. 38–55, 2007.
- [30] S. Boughleux, M. Melkemi, and A. Elmoataz, “Discrete regularization on weighted graphs for image and mesh filtering,” in *International Conference on Scale Space and Variational Methods in Computer Vision (SSVM)*, ser. LNCS, Springer, Ed., vol. 4485, 2007, pp. 128–139.
- [31] O. Lezoray, S. Boughleux, and A. Elmoataz, “Parameterless discrete regularization on graphs for color image filtering,” in *ICIAR*, ser. Lecture Notes in Computer Science, Springer, Ed., 2007.
- [32] V. Ta, O. Lezoray, and A. Elmoataz, “Graphs regularization for data sets and images: Filtering and semi-supervised classification,” in *PASCAL Workshop: Learning from and with graphs, 6th IAPR -TC-15 Workshop on Graph based Representations in Pattern Recognition*, 2007.
- [33] G. Gilboa, J. Darbon, S. Osher, and T. F. Chan, “Nonlocal convex functionals for image regularization,” UCLA, CAM Report 06-57, October 2006.
- [34] D. Zhou and B. Schölkopf, “Regularization on discrete spaces,” in *LNCS 3663, Proc. of the 27th DAGM Symp.* Springer-Verlag, august 2005, pp. 361–368.
- [35] —, “Discrete regularization,” in *Semi-supervised learning*, O. Chapelle, B. Schölkopf, and A. Zien, Eds. MIT Press, Cambridge, MA, 2006, pp. 221–232.
- [36] S. Boughleux and A. Elmoataz, “Image smoothing and segmentation by graph regularization,” in *LNCS 3656, Proc. Int. Symp. on Visual Computing*. Springer-Verlag, 2005, pp. 745–752.
- [37] A. Bensoussan and J.-L. Menaldi, “Difference equations on weighted graphs,” *Journal of Convex Analysis*, vol. 12, no. 1, pp. 13–44, 2005.
- [38] J. Friedman and J.-P. Tillich, “Wave equations for graphs and the edge-based laplacian,” *Pacific Journal of Mathematics*, vol. 216, no. 2, pp. 229–266, 2004.
- [39] M. Hein, J.-Y. Audibert, and U. von Luxburg, “Convergece of graph Laplacians on random neighborhood graphs,” *Journal of Machine Learning Research*, vol. 8, pp. 1325–1370, 2007.
- [40] F. R. Chung, “Spectral graph theory,” *CBMS Regional Conference Series in Mathematics*, vol. 92, pp. 1–212, 1997.
- [41] S. Osher and J. Shen, “Digitized PDE method for data restoration,” in *In Analytical-Computational methods in Applied Mathematics*, E. G. A. Anastassiou, Ed. Chapman&Hall/CRC, 2000, pp. 751–771.
- [42] T. Chan, S. Osher, and J. Shen, “The digital TV filter and nonlinear denoising,” *IEEE Transactions on Image Processing*, vol. 10, no. 2, pp. 231–241, 2001.
- [43] S. Karamadian, *Fixed points, algorithms and applications*. Academic Press, 1977.
- [44] J. Weickert, *Anisotropic Diffusion in Image Processing*. Teubner-Verlag, 1998.
- [45] J. O’Rourke and G. Toussaint, “Pattern recognition,” in *Handbook of discrete and computational geometry*, J. Goodman and J. O’Rourke, Eds. Chapman & Hall/CRC, New York, 2004, ch. 51, pp. 1135–1162.
- [46] R. Coifman, S. Lafon, A. Lee, M. Maggioni, B. Nadler, F. Warner, and S. Zucker, “Geometric diffusions as a tool for harmonic analysis and structure definition of data,” *Proc. of the National Academy of Sciences*, vol. 102, no. 21, 2005.
- [47] C. Tomasi and R. Manduchi, “Bilateral filtering for gray and color images,” in *ICCV’98: Proc. of the 6th Int. Conf. on Computer Vision*. IEEE Computer Society, 1998, pp. 839–846.
- [48] D. Barash, “A fundamental relationship between bilateral filtering, adaptive smoothing, and the nonlinear diffusion equation,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 6, pp. 844–847, 2002.
- [49] S. Fleishman, I. Drori, and D. Cohen-Or, “Bilateral mesh denoising,” *ACM Trans. on Graphics*, vol. 22, no. 3, pp. 950–953, 2003.
- [50] T. R. Jones, F. Durand, and M. Desbrun, “Non-iterative, feature-preserving mesh smoothing,” *ACM Trans. Graph.*, vol. 22, no. 3, pp. 943–949, 2003.
- [51] S. Yoshizawa, A. Belyaev, , and H.-P. Seidel, “Smoothing by example: mesh denoising by averaging with similarity-based weights,” in *Proc. International Conference on Shape Modeling and Applications*, 2006, pp. 38–44.
- [52] V. Ta, S. Boughleux, A. Elmoataz, and O. Lezoray, “Nonlocal anisotropic discrete regularization for image, data filtering and clustering,” University of Caen, Tech. Rep., 2007.
- [53] D. Tschumperlé and R. Deriche, “Vector-valued image regularization with PDEs: A common framework for different applications,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 4, pp. 506–517, 2005.
- [54] J. Weickert, “Coherence-enhancing diffusion of colour images,” *Image Vision Comput.*, vol. 17, no. 3-4, pp. 201–212, 1999.
- [55] C. Meurie and O. Lezoray, “A new method of morphological hierarchical segmentation,” in *Visual Information Engineering*, IEEE and IET, Eds., 2007.
- [56] O. Lezoray, C. Meurie, P. Belhomme, and A. Elmoataz, “Multi-scale image segmentation in a hierarchy of partitions,” in *EUSIPCO (European Signal Processing Conference)*, ser. CD Proceedings, 2006.
- [57] E. Sharon, A. Brandt, and R. Basri, “Fast multiscale image segmentation,” in *CVPR*, 2000, pp. 2070–1077.
- [58] L. Guigues, J. P. Cocquerez, and H. L. Men, “Scale-sets image analysis,” *International Journal of Computer Vision*, vol. 68, no. 3, pp. 289–317, 2006.
- [59] S. Hettich, C. Blake, and C. Merz, “UCI repository of machine learning databases,” University of California, Irvine, Dept. of Information and Computer Sciences, Tech. Rep., 1998. [Online]. Available: <http://www.ics.uci.edu/~mllearn/MLRepository.html>