Local and Nonlocal Discrete Regularization on Weighted Graphs for Image and Mesh Processing

Sébastien Bougleux (sebastien.bougleux@greyc.ensicaen.fr)
ENSICAEN - GREYC CNRS UMR 6072 - Équipe Image
6 BD du Maréchal Juin, 14050 Caen Cedex France

Abderrahim Elmoataz (abder@greyc.ensicaen.fr)
Université de Caen - GREYC CNRS UMR 6072 - Équipe Image
BD du Maréchal Juin, 14050 Caen Cedex France

Mahmoud Melkemi (mahmoud.melkemi@uha.fr)
Univertsité de Haute-Alsace - LMIA - Équipe MAGE
4 rue des Frères Lumière, 68093 Mulhouse Cedex France

Abstract. We propose a discrete regularization framework on weighted graphs of arbitrary topology, which unifies local and nonlocal processing of images, meshes, and more generally discrete data. 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 the discrete p-Dirichlet form, and an approximation one. The proposed model is parametrized by the degree p of regularity, by the graph structure and by the weight function. The minimization solution leads to a family of simple linear and nonlinear processing methods. In particular, this family includes the exact expression or the discrete version of several neighborhood filters, such as the bilateral and the nonlocal means filter. In the context of images, local and nonlocal regularizations, based on the total variation models, are the continuous analogue of the proposed model. Indirectly and naturally, it provides a discrete extension of these regularization methods for any discrete data or functions.

Keywords: discrete variational problems on graphs; discrete diffusion processes; smoothing; denoising; simplification

1. Introduction

Smoothing, denoising, restoration and simplification are fundamental problems of image processing, computer vision and computer graphics. The aim is to approximate a given image or a given model/mesh, eventually corrupted by noise, by filtered versions which are more regular and simpler in some sense. The principal difficulty of this task is to preserve the geometrical structures existing in the initial data, such as discontinuities (object boundaries, sharp edges), rapid transitions (fine structures), and redundancies (textures).

Many methods have been proposed to handle this problem, depending on the domain of application. Among them, variational models, energy minimization and partial differential equations (PDEs) have shown their efficiency in numerous situations. In the context of image processing, regularization methods based on the total variation (TV) and its variants, as well as non-linear/anisotropic diffusions, are among the most important ones, see for example (Alvarez et al., 1993; Weickert, 1998; Paragios et al., 2005; Chan and Shen, 2005; Aubert and Kornprobst, 2006) and references therein. Another important class of methods are statistical and averaging filters, such as median, mean, mode and bilateral filters (Lee, 1983; Smith and Brady, 1997; Tomasi and Manduchi, 1998; Griffin, 2000). These filters can be interpreted as weighted neighborhood filters, and most of them are related to PDEs and energy minimization (Barash, 2002; Sochen et al., 2001; Buades et al., 2005; Mrázek et al., 2006). While all of these methods use weight functions that take into account local image features, a significant advance is the introduction of the nonlocal means (NLM) filter which uses nonlocal features based on patches (Buades et al., 2005). This latter nonlocal neighborhood filter outperforms the capabilities of the previous methods, particularly in the preservation of fine structures and textures. Then, several other filters using similar ideas have been proposed (Kervrann et al., 2007; Brox and Cremers, 2007). A variational understanding of the NLM filter was first developed as a non-convex energy functional (Kinderman et al., 2005), and more recently as a convex quadratic energy functional (Gilboa and Osher, 2007a; Gilboa and Osher, 2007b).

In the context of mesh processing, smoothing and denoising tasks are usually performed according to geometric flows. The most commonly used technique is the Laplacian smoothing which is fast and simple, but which produces over-smoothing and shrinking effects (Taubin, 1995). Inspired by the efficiency of image denoising methods mentioned above, the most recent methods include the mean and the angle median filters for averaging face normals (Yagou et al., 2002), the bilateral filter (Fleishman et al., 2003; Jones et al., 2003), and the NLM filter (Yoshizawa et al., 2006). Also several anisotropic diffusion flows for simplicial meshes and implicit surfaces have been proposed to preserve and enhance sharp edges, such as: weighted Laplacian smoothing (Desbrun et al., 2000), anisotropic geometric diffusion using diffusion tensor (Clarenz et al., 2000), mean curvature flow (Hildebrandt and Polthier, 2004), discrete Laplace-Beltrami flow (Bajaj and

Xu, 2003; Xu, 2004), and discrete Willmore flow (Bobenko and Schröder, 2005). While these flows are conceived to filter the position of the vertices of a mesh, a different approach is introduced by (Tasdizen et al., 2003). This approach filters the normal map of an implicit surface, and manipulates the surface in order to fit with the processed map.

In both image and mesh processing, the data is discrete by nature. In most of the methods based on energy minimization, PDEs and diffusion flows, data are assumed to be defined on a continuous domain. Then a numerical solution is adapted to the discrete domain upon which the data is naturally defined. An alternative is to formalize the smoothing/denoising problem directly in discrete settings. This is the case for neighborhood filters, which are mainly based on discrete weighted Laplacians. See (Chung, 1997; Cvetković et al., 1980) for a description of these operators in the general context of graph theory. In particular, it is shown that Laplacian filtering is equivalent to Markov matrix filtering, and by consequence it is also related to spectral graph filtering. Similar work for image denoising has been proposed by (Coifman et al., 2006; Szlam et al., 2006). Another interesting work is the digitization of the TV and the ROF model of images onto unweighted graphs (Osher and Shen, 2000; Chan et al., 2001). This discrete formulation has received much less attention than its continuous analogue. An extension of this model, using a normalized p-Dirichlet form on weighted graphs, is proposed by (Zhou and Schölkopf, 2005) in the context of semi-supervised learning. Other methods, developed in the context of image filtering, that can be considered as discrete regularizations on unweighted graphs (Chambolle, 2005; Darbon and Sigelle, 2004). These regularizations yield to Markov random fields where only binary variables are involved in the minimization.

In the same digital context, we propose in this paper a general variational formulation of the smoothing/denoising problem for data defined on weighted graphs (Bougleux et al., 2007a). It is also a direct extension of the digital ROF model, but based on another p-Dirichlet form. There exist several advantages of the proposed approach. In particular, it leads to a family of discrete and semi-discrete diffusion processes based on the combinatorial p-Laplacian. For $p = 2$, this family includes many neighborhood filters used in image processing. Moreover, local and nonlocal regularizations are formalized within the same framework, and which correspond to the transcription of the nonlocal continuous regularizations proposed recently for $p = 2$ and $p = 1$. Thus, data which have a natural graph structure (images, polygonal curves and surfaces, networks, etc), can be represented by more complex graph structures, which take into account local or nonlocal interactions. In the context of image processing, we also show that we can use simplified versions represented by region adjacency graphs (RAG).

The rest of this paper is organized as follows. In the next section, we define difference operators on weighted graphs that are used to construct the regularization framework and the associated family of discrete diffusion processes presented in Section 3. In Section 4, the obtained processes are analyzed and related to existing ones. Finally we give some experimentations for different values of p and weight functions in the context of image and mesh processing (smoothing, denoising). In particular we show that for $p \to 0$, the diffusion processes behave like simplification and clustering methods.

2. Operators on Weighted Graphs

In this section, we recall some basic definitions on graphs, and we define difference operators which can be considered as discrete versions of continuous differential operators. Analogue definitions and properties have also been used in the context of functional analysis on graphs (Bensoussan and Menaldi, 2005; Friedman and Tillich, 2004), semi-supervised learning (Zhou and Schölkopf, 2005) and image processing (Bougleux and Elmoataz, 2005).

2.1. Graphs and Spaces of Functions on Graphs

A *weighted graph* $G = (V, E, w)$ consists of a finite set V of N *vertices* and a finite set $E \subset V \times V$ of *weighted edges*. The weight of each edge $(u, v) \in E$, noted w_{uv}, is non-negative. In many cases, it is given by the *weight function* $w : V \times V \to \mathbb{R}^+$, which verifies :

$$w(u,v) = \begin{cases} w_{uv} & \text{if } (u,v) \in E, \\ 0 & \text{otherwise.} \end{cases}$$

The weight represents the similarity between two vertices of the graph (u and v are similar if $w_{uv} = 1$). In this paper, the considered graphs are connected, undirected ($(u, v) \in E \Leftrightarrow (v, u) \in E$ with $w(u, v) = w(v, u)$), with no self-loops or multiple edges.

The *degree* of a vertex, noted $\delta_w : V \to \mathbb{R}^+$, measures the sum of the weights in the neighborhood of that vertex :

$$\delta_w(u) = \sum_{v \sim u} w_{uv}, \quad \forall u \in V,$$

where the notation $v \sim u$ denotes the vertices of V connected to the vertex u by an edge of E.

Functions on graphs. The graphs considered here are topological. The data to be processed are represented by real-valued functions $f : V \to \mathbb{R}$, which assign a real value $f(u)$ to each vertex $u \in V$ (the case of vector-valued functions is considered in Section 5.1). These functions form a finite N-dimensional space. They can be represented by vectors of $\mathbb{R}^N$, and interpreted as the intensity of a discrete signal defined on the vertices of the graph. When such functions come from the discretization of continuous functions defined in a continuous domain, the geometry of that domain is usually encoded into the weight function.

By analogy with continuous functional spaces, the discrete integral of a function $f : V \to \mathbb{R}$, on the graph G, is defined by $\int_G f = \sum_{u \in V} f(u)m(u)$, where $m : V \to \mathbb{R}^+$ is a measure on the neighborhood of the vertex u. In the sequel, and without lost of generality, we set $m(u) = 1$ for all $u \in V$.

Let $\mathcal{H}(V)$ denotes the Hilbert space of the real-valued functions $f : V \to \mathbb{R}$. It is endowed with the usual inner product:

$$\langle f, h \rangle_{\mathcal{H}(V)} = \sum_{u \in V} f(u)h(u), \quad f, h : V \to \mathbb{R}, \tag{1}$$

and with the induced $\mathcal{L}^2$ norm: $\|f\|_2 = \langle f, f \rangle_{\mathcal{H}(V)}^{1/2}$.

Also, there exist functions defined on the edges of the graph, such as the weight function. Let $\mathcal{H}(E)$ be the space of real-valued functions $F : E \to \mathbb{R}$ defined on the edges of G. It is endowed with the inner product:

$$\langle F, H \rangle_{\mathcal{H}(E)} = \sum_{u \in V} \sum_{v \sim u} F(u, v)H(u, v), \quad F, H : E \to \mathbb{R}, \tag{2}$$

One can remark that the functions do not need to be symmetric, and their inner product can be rewritten as:

$$\langle F, H \rangle_{\mathcal{H}(E)} = \sum_{(u,v) \in E} F(u, v)H(u, v), \quad F, H : E \to \mathbb{R}. \tag{3}$$

The induced $\mathcal{L}^2$ norm is defined by: $\|F\|_2 = \langle F, F \rangle_{\mathcal{H}(E)}^{1/2}$.

2.2. Difference Operator, Edge Derivative and Adjoint

All the basic operators considered in this paper are defined from the difference operator or the directional derivative. There exist several definitions of these operators on graphs (Requardt, 1997; Bensoussan and Menaldi, 2005; Friedman and Tillich, 2004). Here, we propose a definition of the difference operator that allows to retrieve the expression of the combinatorial p-Laplace operator (Bougleux et al., 2007a), and the expression of the normalized p-Laplace operator (Zhou and Schölkopf, 2005).

The *weighted difference operator* of a function $f \in \mathcal{H}(V)$, noted $d_w : \mathcal{H}(V) \to \mathcal{H}(E)$, is defined on an edge $(u, v) \in E$ by:

$$d_w(f)(u, v) = \gamma_w(v, u)f(v) - \gamma_w(u, v)f(u), \ \forall (u, v) \in E, \tag{4}$$

where $\gamma_w : V \times V \to \mathbb{R}^+$ is in the form $\gamma_w(u, v) = \sqrt{\psi(u, v)w(u, v)}$. In the sequel, the value of γ_w at an edge (u, v) is noted γ_{uv}. The difference operator is *linear* and *antisymmetric*. By analogy with continuous functional analysis, this implies the definition of the edge derivative.

The *edge directional derivative* of a function $f \in \mathcal{H}(V)$ at a vertex u, along an edge $e = (u, v)$, is defined by:

$$\left.\frac{\partial f}{\partial e}\right|_u = \partial_v f(u) = d_w(f)(u, v). \tag{5}$$

If $\gamma_{uv} = \gamma_{vu}$, then this definition is consistent with the continuous definition of the derivative of a function, e.g., if $f(u) = f(v)$ then $\partial_v f(u) = 0$. Moreover, note that $\partial_u f(v) = -\partial_v f(u)$, and $\partial_u f(u) = 0$.

The *adjoint operator* of the difference operator d_w, denoted by $d_w^* : \mathcal{H}(E) \to \mathcal{H}(V)$, is defined by:

$$\langle d_w f, H\rangle_{\mathcal{H}(E)} = \langle f, d_w^* H\rangle_{\mathcal{H}(V)}, \qquad f \in \mathcal{H}(V),\ H \in \mathcal{H}(E). \tag{6}$$

Using the definitions of the inner products in $\mathcal{H}(V)$ and $\mathcal{H}(E)$, and the definition of the difference operator, we obtain the expression of d_w^* at a vertex of the graph.

PROPOSITION 1. *The adjoint operator d_w^* of a function $H \in \mathcal{H}(E)$ can be computed at vertex $u \in V$ by:*

$$d_w^*(H)(u) = \sum_{v \sim u} \gamma_{uv}(H(v, u) - H(u, v)). \tag{7}$$

Proof: see Appendix A.

The adjoint operator is *linear*. It measures the flow of a function in $\mathcal{H}(E)$ at each vertex of the graph. By analogy with continuous differential operators, the *divergence* of a function $F \in \mathcal{H}(E)$ is defined by $div_w F = -d_w^* F$. Then, it is easy to show the following null divergence property.

PROPERTY 1. *If γ_w is symmetric, then $\sum_{u \in V} div_w(F)(u) = 0$, $\forall F \in \mathcal{H}(E)$.*

2.3. Gradient Operator

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

$$\nabla_w f(u) = (\partial_v f(u) : v \sim u) = (\partial_{v_1} f(u), \ldots, \partial_{v_k} f(u)), \qquad v_i \sim u. \tag{8}$$

One can remark that this definition does not depend on the graph structure, and thus the gradient has the same general expression for regular, irregular, geometric and topological graphs.

By analogy with the continuous definition of the gradient, the graph-gradient is a first order operator defined for each vertex in a local space given by the neighborhood of this vertex. Moreover, many discrete gradient operators can be formulated from the above definition by choosing the adequate expressions of the function γ_w and the similarity function w involved in the edge derivative. In particular, in the context of image processing, we can retrieve the classical discrete gradient used in the numerical discretization of the solution of PDE's on grid graphs of 4-adjacency.

The *local variation* of f, at a vertex u, is defined by the following gradient $\mathcal{L}^2$-norm:

$$|\nabla_w f(u)| = \sqrt{\sum_{v\sim u}(\partial_v f(u))^2}. \tag{9}$$

It can be viewed as a measure of the regularity of a function around a vertex. From the definition of the function γ_w, it is also written as:

$$|\nabla_w f(u)| = \sqrt{\sum_{v\in V}(\partial_v f(u))^2} = \sqrt{\sum_{v\in V} w(u,v)\psi(u,v)(f(v)-f(u))^2}.$$

This is due to the fact that $w(u,v) = 0$ if $v \not\sim u$. This formulation is nonlocal, since all the vertices of V are included in the summation. It takes all its meaning in the context of discrete set of data (see Section 5).

Other measurements of the regularity can be performed by different gradient norms, such as the $\mathcal{L}^p$-norm:

$$|\nabla_w f(u)|_p = \left(\sum_{v\sim u} |\partial_v f(u)|^p\right)^{\frac{1}{p}}, \quad p \in (0,+\infty). \tag{10}$$

These gradient norms are used in Section 3 to construct several regularization functionals.

2.4. The p-Laplace Operator

The p-Laplace operator describes a family of second order operators. This family includes the Laplace operator for $p = 2$, and the curvature operator for $p = 1$. Based on the weighted difference operator and its adjoint defined in Section 2.2, we use the classical definition of the p-Laplace operator in order to obtain its local expression at a vertex of the graph.

General Case. Given a value of $p \in (0,+\infty)$, the *weighted p-Laplace operator* $\Delta_w^p : \mathcal{H}(V) \to \mathcal{H}(V)$ is defined by:

$$\Delta_w^p f = d_w^*(|\nabla_w f|^{p-2} d_w f). \tag{11}$$

It is a *nonlinear* operator, excepted in the case of $p = 2$ (since d_w and d_w^* are linear).

PROPOSITION 2. *The weighted p-Laplace operator Δ_w^p of a function $f \in \mathcal{H}(V)$ can be computed at vertex $u \in V$ by:*

$$\Delta_w^p f(u) = \sum_{v\sim u} \gamma_{uv}\left(|\nabla_w f(u)|^{p-2} + |\nabla_w f(v)|^{p-2}\right)\left(\gamma_{uv} f(u) - \gamma_{vu} f(v)\right). \tag{12}$$

Equivalently, it also corresponds to the following expressions:

$$\Delta_w^p f(u) = \sum_{\substack{v\sim u\\ e=(u,v)}} \gamma_{uv} \frac{\partial}{\partial e}\left(\frac{|\bigtriangledown_w f|^{p-2}}{\gamma}\frac{\partial f}{\partial e}\right)\bigg|_u \quad (13)$$

$$\Delta_w^p f(u) = -\sum_{v\sim u} \gamma_{uv}\left(|\bigtriangledown_w f(u)|^{p-2} + |\bigtriangledown_w f(v)|^{p-2}\right)\partial_v f(u). \quad (14)$$

Proof: see Appendix A.

Eq. (13) and Eq. (14) show the relation between second order and first order derivatives.

Remark that the gradient of the function f can be null (locally flat functions). When $p < 2$, in order to avoid a division by zero in the expression of the p-Laplace operator, the gradient has to be regularized as:

$$|\bigtriangledown_w f|_\epsilon = \sqrt{|\bigtriangledown_w f|^2 + \epsilon^2}, \quad (15)$$

where $\epsilon \to 0$ is a positive constant.

Case of p=2, the Laplace Operator. When $p = 2$, Eq. (11) reduces to $\Delta_w^2 f = d_w^*(d_w f) = \Delta_w f$, which is the expression of the *weighted Laplace operator* of the function f on the graph. Since both the difference and its adjoint are linear, it is also a *linear* operator. At a vertex $u \in V$, it can be computed by:

$$\Delta_w f(u) \overset{(12)}{=} 2\sum_{v\sim u} \gamma_{uv}(\gamma_{uv} f(u) - \gamma_{vu} f(v)) \quad (16)$$

$$\overset{(13)}{=} \sum_{\substack{v\sim u\\ e=(u,v)}} \gamma_{uv} \frac{\partial}{\partial e}\left(\frac{1}{\gamma}\frac{\partial f}{\partial e}\right)\bigg|_u \overset{(14)}{=} -\sum_{v\sim u} \gamma_{uv}\partial_v f(u). \quad (17)$$

When γ_w is symmetric, Eq. (17) reduces to $\sum_{v\sim u} \partial_v^2 f(u) = -\sum_{v\sim u} \gamma_w \partial_v f(u)$. Also, Eq. (17) is the discrete analogue of the Laplace-Beltrami operator on manifolds, defined in local coordinates as:

$$\Delta_{\mathcal{M}} f = div_{\mathcal{M}}(\bigtriangledown_{\mathcal{M}}) = \frac{1}{\sqrt{|g|}}\partial_i\left(\sqrt{|g|}g^{ij}\partial_j f\right),$$

where g is a metric tensor on the manifold $\mathcal{M}$, and g^{ij} the components of its inverse. In particular, the Laplace-Beltrami operator is widely used to process meshes and images, see (Xu, 2004; Kimmel et al., 2000). There exists several discrete expressions of the Laplace or the Laplace-Beltrami operator, depending on the context. Many of them can be expressed using Eq. (16). Indeed, general expressions have been formulated in the context of spectral graph theory (Chung, 1997), which studies the eigenvalues and the eigenvectors of Laplacian matrices.

Table I recall the definitions and the local expressions of the two well-known graph Laplacians that can be derived from the weighted Laplace operator (16) by choosing specific forms of the function γ_w. The matrix W is the weight matrix such that $W(u,v) = w(u,v)$ for all $u, v \in V$, and D is the diagonal degree matrix defined by $D(u,v) = 0$ if $u \neq v$ and $D(u,u) = \delta_w(u)$ otherwise.

Table I. Expressions of the Laplace operator related to Eq. (16).

combinatorial Laplacian	*normalized Laplacian*
$L = D - W$	$L_n = D^{-1/2} L D^{-1/2} = I - D^{-1/2} W D^{-1/2}$
$\delta_w(u)f(u) - \sum_{v\sim u} w_{uv} f(v)$	$f(u) - \frac{1}{\sqrt{\delta_w(u)}} \sum_{v\sim u} \frac{w_{uv}}{\sqrt{\delta_w(v)}} f(v)$
$\gamma_1 = \gamma_w = \sqrt{w/2}$	$\gamma_2(u,v) = \gamma_w(u,v) = \sqrt{\frac{w_{uv}}{2\delta_w(u)}}$

These two Laplace operators are used in many applications based on diffusion processes on graphs, which is discussed in Section 4.1.

Case of p=1, the curvature Operator. When $p = 1$, Eq. (11) reduces to $\Delta_w^1 f = d_w^*(|\bigtriangledown_w f|^{-1} d_w f) = \kappa_w f$, which represents the *weighted curvature operator* of the function f. It is *nonlinear* and it can be computed locally by:

$$\kappa_w f(u) \overset{(12)}{=} \sum_{v\sim u} \gamma_{uv} \left(\frac{1}{|\bigtriangledown_w f(v)|} + \frac{1}{|\bigtriangledown_w f(u)|} \right) (\gamma_{uv} f(u) - \gamma_{vu} f(v)). \quad (18)$$

When the graph is unweighted ($w = 1$) and $\gamma = \gamma_1$, this last expression corresponds to the curvature operator proposed by (Osher and Shen, 2000; Chan et al., 2001) in the context of image processing (see next section). Then, the p-Laplace operator, defined by Eq. (12) with γ_1 and γ_2, can be seen as a direct extension of the combinatorial and normalized Laplace and curvature operators.

3. Proposed Framework

In this section, we present the variational model that we propose to regularize functions defined on the vertices of graphs, as well as the discrete diffusion processes associated with it.

3.1. The Discrete Variational Model Based on Regularization

Let $G = (V, E, w)$ be a weighted graph, and let $f^0 : V \to \mathbb{R}$ be a given function of $\mathcal{H}(V)$. In real applications, f^0 represents measurements which are perturbed by noise (acquisition, transmission, processing). We consider in this paper the case of additive noise $\mu \in \mathcal{H}(V)$, such that $f^0 = h + \mu$ and $h \in \mathcal{H}(V)$ is the noise free version of f^0. To recover the unknown function h, f^0 is regularized by seeking for a function $f \in \mathcal{H}(V)$ which is not only regular enough on G, but also close enough to the initial function f^0. This optimization problem can be formalized by the minimization of a weighted sum of two energy terms:

$$\min_{f:V\to\mathbb{R}} \left\{ R_G^p(f) + \tfrac{\lambda}{2}\|f - f^0\|_2^2 \right\}. \tag{19}$$

The first energy functional R_G^p measures the regularity of the function f over the graph, while the second measures its closeness to the initial function. The parameter $\lambda \geq 0$ is a fidelity parameter which specifies the trade-off between the two competing functionals.

The regularity of the desired solution f is measured by its p-Dirichlet energy based on the local variation (9), which is given by:

$$\begin{aligned} R_G^p(f) &= \tfrac{1}{p} \sum_{u\in V} |\nabla_w f(u)|^p, \qquad p \in (0, +\infty), \\ &\overset{(4)}{=} \tfrac{1}{p} \sum_{u\in V} \left(\sum_{v\sim u} (\gamma_{vu} f(v) - \gamma_{uv} f(u))^2 \right)^{\frac{p}{2}}. \end{aligned} \tag{20}$$

It is the weighted discrete analogue of the p-Dirichlet energy of continuous functions defined on a continuous bounded domain Ω of the Euclidean space: $J^p(f) = \frac{1}{p}\int_\Omega |\nabla_{|x} f|^p dx$, $f : \Omega \subset \mathbb{R}^m \to \mathbb{R}$.

When $p = 2$, the regularization functional (20) is the Dirichlet energy and the minimizer (19) corresponds to the Tikhonov regularization. Another important case is the total variation and the ROF model of images (or fitted TV), which are obtained with $p = 1$.

For $p \geq 1$, both functionals in the minimizer (19) are strictly convex. Then if the solution of problem (19) exists, it is unique. As $\lim_{f\to\infty} E_w^p(f) = \infty$, by standard arguments in convex analysis, problem (19) has a unique solution which can be computed by solving:

$$\frac{\partial}{\partial f(u)} \left(R_G^p(f) + \tfrac{\lambda}{2}\|f - f^0\|_2^2 \right) = 0, \quad \forall u \in V.$$

The derivative of the discrete p-Dirichlet functional is computed using the following property.

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

Proof: see Appendix B.

Then, the solution of problem (19) is the solution of the following system of equations:

$$\Delta_w^p f(u) + \lambda(f(u) - f^0(u)) = 0, \quad \forall u \in V. \tag{21}$$

This last equation can be interpreted as discrete Euler-Lagrange equations. Contrary to the continuous case, it does not involve any PDEs and it is independent of the graph structure. By substituting the expression of the p-Laplace operator into Eq. (21), we obtain directly:

$$\left(\lambda + \sum_{v \sim u} \alpha_{uv}(f)\right) f(u) - \sum_{v \sim u} \beta_{uv}(f) f(v) = \lambda f^0(u), \quad \forall u \in V, \tag{22}$$

where the coefficients α and β are used to simplify the notations:

$$\begin{cases} \alpha_{uv}(f) = \left(|\nabla_w f(u)|^{p-2} + |\nabla_w f(v)|^{p-2}\right) \gamma_{uv}^2 \\ \beta_{uv}(f) = \left(|\nabla_w f(u)|^{p-2} + |\nabla_w f(v)|^{p-2}\right) \gamma_{uv}\gamma_{vu}. \end{cases}$$

When $p \neq 2$, (22) is a nonlinear system. When $p = 2$, the system is linear and can be solved efficiently with several numerical methods which converge close to the solution of the minimization problem. In the next sections, we propose to use simple and fast algorithms to find a solution in the general case.

When $p < 1$, R_G^p is non-convex, and the global minimization may not exist. Nevertheless, this does not mean that the diffusion processes associated with this case are not interesting.

3.2. Discrete Diffusion Processes

As in the continuous case, the solution of the minimization problem can be formulated as diffusion processes. The solution of the system of equations (21) can be obtained by considering the infinitesimal marching step descent:

$$\begin{cases} f^{(0)} = f^0 \\ \frac{d}{dt} f^{(t)}(u) = -\Delta_w^p(f)(u) + \lambda(f^0(u) - f(u)), \quad \forall u \in V, \end{cases} \tag{23}$$

where $f^{(t)}$ is the parametrization of the function f by an artificial time. This is a system of ordinary differential equations. Contrary to PDEs methods, no space discretization is necessary. Its solution can be efficiently approximated by local iterative methods. By simply using the Runge-Kutta method of order one, the algorithm that computes the approximated solution is given by:

$$\begin{cases} a.\ \text{Initialization with } f^{(0)}(u) = f^0(u),\ \forall u \in V. \\ b.\ \text{For } t = 0 \text{ to a fixed or iteratively computed stopping time, do:} \\ \quad f^{(t+1)}(u) = f^{(t)}(u) + \tau(-\Delta_w^p f^{(t)}(u) + \lambda(f^0(u) - f^{(t)}(u))),\ \forall u \in V, \end{cases} \tag{24}$$

where $\tau > 0$ is the size of the infinitesimal marching step.

Another method to solve the system of equations (22) is to use the Gauss-Jacobi iterative algorithm given by the following steps:

$$\begin{cases} a.\ \text{Initialization with } f^{(0)}(u) = f^0(u),\ \forall u \in V. \\ b.\ \text{For } t = 0 \text{ to a fixed or iteratively computed stopping time, do:} \\ \begin{cases} \alpha_{uv}(f^{(t)}) = \gamma_{uv}^2 \left(|\nabla_w f^{(t)}(v)|^{p-2} + |\nabla_w f^{(t)}(u)|^{p-2} \right),\ \forall (u,v) \in E \\ \beta_{uv}(f^{(t)}) = \gamma_{uv}\gamma_{vu} \left(|\nabla_w f^{(t)}(v)|^{p-2} + |\nabla_w f^{(t)}(u)|^{p-2} \right),\ \forall (u,v) \in E \\ f^{(t+1)}(u) = \dfrac{\lambda f^0(u) + \sum_{v\sim u} \beta_{uv}(f^{(t)}) f^{(t)}(v)}{\lambda + \sum_{v\sim u} \alpha_{uv}(f^{(t)})},\ \forall u \in V. \end{cases} \end{cases} \tag{25}$$

Let φ be the function given by:

$$\varphi_{uv}(f) = \frac{\beta_{uv}(f)}{\lambda + \sum_{v\sim u} \alpha_{uv}(f)} \text{ if } u \neq v, \text{ and } \varphi_{vv}(f) = \frac{\lambda}{\lambda + \sum_{v\sim u} \alpha_{uv}(f)}$$

Then, the regularization algorithm (25) is rewritten as:

$$\begin{cases} f^{(0)} = f^0 \\ f^{(t+1)}(u) = \varphi_{vv}(f^{(t)}) f^0(u) + \sum_{v\sim u} \varphi_{uv}(f^{(t)}) f^{(t)}(v),\ \forall u \in V. \end{cases} \tag{26}$$

At each iteration, the new value $f^{(t+1)}$, at a vertex u, depends on two quantities, the original value $f^0(u)$, and a weighted average of the existing values in a neighborhood of u. When the function γ_w is symmetric, e.g. $\alpha_{uv} = \beta_{uv}$, the function φ satisfies $\varphi_{uu} + \sum_{v\sim u} \varphi_{uv} = 1$. In this case, the proposed algorithm describes a forced low-pass filter.

The above methods describe families of diffusion processes, parametrized by the graph structure, the weight function, the fidelity parameter λ, and the degree of regularity p. For specific values of these parameters, the algorithm (31) corresponds exactly to well-known diffusion processes used in image processing. It is the one we use in the applications described in Section 5.

4. Analysis and Related Works

In the sequel, we discuss particular cases of the proposed regularization framework, and we show the relation with spectral graph theory and recent nonlocal continuous functionals defined in the context of image processing.

4.1. Link to graph theory and spectral filtering

Let $G = (V, E, w)$ be a weighted graph. Let $f : V \to \mathbb{R}$ be a function in $\mathcal{H}(V)$ represented as a vector. It is easy to see that the classical smoothness functionals associated with the Laplacians L and L_n (see Table I) are particular cases of the proposed regularization functional R_G^2 for specific functions γ_w:

$$\begin{cases} R_G^L(f) = \langle f, Lf \rangle = \frac{1}{2} \sum_{u \in V} \sum_{v \in V} w(u,v)(f(u) - f(v))^2, \\ R_G^{L_n}(f) = \langle f, L_n f \rangle = \frac{1}{2} \sum_{u \in V} \sum_{v \in V} w(u,v) \left(\frac{f(u)}{\sqrt{\delta_w(u)}} - \frac{f(v)}{\sqrt{\delta_w(v)}} \right)^2. \end{cases}$$

Then the proposed p-Dirichlet energy R_G^p can be seen as a direct extension of the above ones. In particular, R_G^1 associated with $\gamma_w = 1$ (unweighted graphs) has been proposed by (Osher and Shen, 2000) in the context of image restoration. Also, R_G^p associated with $\gamma_w(u,v) = \sqrt{w(u,v)/\delta_w(u)}$ has been proposed by (Zhou and Schölkopf, 2005) in the context of semi-supervised classification. In the present paper, we propose to use R_G^p associated with $\gamma_w = \sqrt{w}$ in the context of image and mesh filtering (Bougleux et al., 2007a). Recently, the non-linear flows associated with p-Dirichlet energies on graphs has been replaced by a non-iterative thresholding in a non-local spectral basis (Peyré, 2008).

Relation between discrete diffusion and spectral filtering. We consider the discrete diffusion process (26), for $\lambda = 0$, $p \in (0, +\infty)$ and $\gamma_w = \sqrt{w}$. Under these conditions, an iteration of this process is given by:

$$f^{(t+1)}(u) = \sum_{v \sim u} \varphi_{uv}(f^{(t)}) f^{(t)}(v), \ \forall u \in V, \tag{27}$$

where the function φ reduces to:

$$\varphi_{uv}(f) = \frac{w_{uv}(|\bigtriangledown_w f(u)|^{p-2} + |\bigtriangledown_w f(v)|^{p-2})}{\sum_{v \sim u} w_{uv}\,(|\bigtriangledown_w f(u)|^{p-2} + |\bigtriangledown_w f(v)|^{p-2})}, \quad \forall (u,v) \in E.$$

As we have $\varphi_{vu} \geq 0$ and $\sum_{v \sim u} \varphi_{uv} = 1$, φ_{uv} can be interpreted as the probability of a random walker to jump from u to v in a single step. Let P be the Markov matrix defined by: $P(u,v) = \varphi_{uv}$ if the edge $(u,v) \in E$, and $P(u,v) = 0$ otherwise. Then the expression (27) can be rewritten as:

$$f^{(t+1)} = P f^{(t)} = P^t f^{(0)}. \tag{28}$$

An element $P^t(u,v)$ describes the probability of transition in t steps. The matrix P^t encodes local similarities between vertices of the graph and diffuses this local information for t steps to larger and larger neighborhoods of each vertex.

The spectral decomposition of the matrix P is given by $P\phi_i = a_i \phi_i$, with $1 \geq a_1 \geq \ldots \geq a_i \geq \ldots \geq a_N \geq 0$ the eigenvalues of P, and ϕ_i its eigenvectors. The eigenvectors associated with the k first eigenvalues contain the principal information. The top non-constant eigenvector ϕ_1 is usually used for finding clusters and computing cuts (Shi and Malik, 2000). Thus, an equivalent way to look at the power of P in the diffusion process (28) is to decompose each value of f on the first eigenvectors of P. Moreover, the eigenvectors of the matrix P can be seen as an extension of the Fourier transform basis functions with a_i^{-1} representing frequencies. It defines a basis of any function f in $\mathcal{H}(V)$, and the function f can be decomposed on the k first eigenvectors of P as:

$$f \approx \sum_{i=1}^{i=k} \langle f, \phi_i \rangle \phi_i.$$

This can be interpreted as a filtering process in the spectral domain. Such a process is used to study the geometry of data set and to analyze functions defined on it, see (Coifman et al., 2005; Szlam et al., 2006) and references therein.

4.2. Link to continuous nonlocal regularization functionals

The proposed p-Dirichlet energy is by nature both local and nonlocal, depending on the topology of the graph and the choice of the weight function. Its nonlocal version is given by:

$$R_G^p(f) = \tfrac{1}{p} \sum_{u \in V} |\nabla_w f(u)|^p = \tfrac{1}{p} \sum_{u \in V} \left(\sum_{v \in V} (\gamma_{vu} f(v) - \gamma_{uv} f(u))^2 \right)^{\frac{p}{2}}$$

This is the discrete analogue of the following continuous nonlocal regularizer of a function $f : \Omega \to \mathbb{R}$ defined on a bounded domain Ω of the Euclidean space:

$$J_{NL}^p(f) = \tfrac{1}{p} \int_\Omega \left(\int_\Omega (\gamma_{yx} f(y) - \gamma_{xy} f(x))^2 dy \right)^{\frac{p}{2}} dx.$$

In particular, for $\gamma_w = \sqrt{w}$, $p = 2$ and $p = 1$, this latter corresponds respectively to:

$$\begin{cases} J_{NL}^2(f) = \displaystyle\int_{\Omega \times \Omega} w(x,y)(f(y) - f(x))^2 dy dx, \\ J_{NL}^1(f) = \displaystyle\int_\Omega \left(\int_\Omega w(x,y)(f(y) - f(x))^2 dy \right)^{\frac{1}{2}} dx. \end{cases} \tag{29}$$

These two regularizers have been proposed recently in the context of image processing (Gilboa and Osher, 2007a; Gilboa and Osher, 2007b). The first one is a continuous variational interpretation of a family of neighborhood filters, such as the NLM filter (Buades et al., 2005). The proposed regularization framework is also a variational interpretation of these filters, but established in discrete settings. The second regularizer is the nonlocal TV functional.

Also in (Gilboa and Osher, 2007b), a nonlocal anisotropic TV functional based on differences is proposed:

$$J_{NLa}(f) = \tfrac{1}{2} \int_{\Omega \times \Omega} \sqrt{w(x,y)} |f(y) - f(x)| dy dx.$$

In the same spirit, we can formulate a general discrete regularizer using the $\mathcal{L}^p$-norm (10) of the weighted gradient as:

$$
\begin{aligned}
\widetilde{R^p_G}(f) =& \frac{1}{2p} \sum_{u \in V} |\nabla_w f(u)|_p^p, \quad p \in (0, +\infty), \\
\overset{(10),(4)}{=}& \frac{1}{2p} \sum_{u \in V} \sum_{v \in V} |\gamma_{vu} f(v) - \gamma_{uv} f(u)|^p.
\end{aligned} \tag{30}
$$

We can remark that (29) and (30) are the same if $p = 2$. In the particular case of $\gamma_w = \sqrt{w}$, (30) becomes:

$$
\widetilde{R^p_G}(f) = \frac{1}{2p} \sum_{u \in V} \sum_{v \in V} w(u,v)^{\frac{p}{2}} |f(v) - f(u)|^p.
$$

For $p = 1$, it is the discrete analogue of the nonlocal anisotropic TV functional J_{NLa}. One can remark that the discrete energy $\widetilde{R^p_G}$ is formalized by using the gradient operator, while the continuous one have been constructed using differences (Gilboa and Osher, 2007b).

In order to solve the continuous variational model associated to the nonlocal functionals, the image domain is discretized and becomes equivalent to a graph. So, both approaches (discrete and continuous) are equivalent. Nevertheless, our approach can be used to process any function defined on a graph structure, and extends the notion of regularity with the parameter p.

5. Applications

The family of regularization processes proposed in Section 3 can be used to regularize any function defined on the vertices of a graph, or on any discrete data set. Through examples, we show how it can be used to perform image and polygonal mesh smoothing, denoising and simplification. To do this, we use the discrete diffusion process (25) with the function $\gamma_w = \sqrt{w}$:

$$
\begin{cases}
f^{(0)} =& f^0 \\
f^{(t+1)}(u) =& \dfrac{\lambda f^0(u) + \sum_{v \sim u} w_{uv} (|\nabla_w f^{(t)}(v)|^{p-2} + |\nabla_w f^{(t)}(u)|^{p-2}) f^{(t)}(v)}{\lambda + \sum_{v \sim u} w_{uv} (|\nabla_w f^{(t)}(v)|^{p-2} + |\nabla_w f^{(t)}(u)|^{p-2})}.
\end{cases} \tag{31}
$$

The regularization parameters (p and λ), as well as the structure of the graph and the choice of the weight function depend on the application. The aim of this section is not to present the best results, but some applications of the regularization, in the context of image and mesh processing. In particular, the regularization using $p \to 0$ behaves like a simplification or a clustering process, in both local and nonlocal schemes.

5.1. Case of vector-valued functions

In the case of a vector-valued function $f : V \to \mathbb{R}^m$, with $f(u) = (f_1(u), \ldots, f_m(u))$, the regularization is performed on each component f_i independently. This comes to have m regularization processes. Then, the local variation $|\bigtriangledown_w f_i|$ is different for each component. Applying the regularization in a component-wise manner is interesting to develop a computational efficient solution. However, in many applications, component-wise processing can have serious drawbacks contrary to vector processing solutions.

To overcome this limitation, a regularization process acting on vector-valued functions needs to be driven by equivalent attributes, taking the coupling between vector components into account. Therefore, component-wise regularization does not have to use different local geometries of the function on the graph, but a vector one. In the case of $p = 2$, the Laplace operator (16) is the same for the m components, and the regularization can be performed independently on each component. But in the case of $p \neq 2$, the p-Laplace operator (12) is different for each component, and the m regularization processes can be totally independent if the weight function w does not incorporate any inter-component information. In order to take into account the inner correlation aspect of vector-valued functions, the local variation (9) is replaced by the multi-dimensional norm:

$$|\bigtriangledown_w f(u)|_{mD} = \sqrt{\sum_{k=1}^{m} |\bigtriangledown_w f_i(u)|^2}.$$

Then, the proposed regularization applies to each component of the vector-valued function with a weighting of edges and a vector gradient norm acting both as coupling between components to avoid drawbacks of applying the regularization in a component-wise manner.

5.2. Application to image Processing

To process an image of pixels $f^0 : V \subset \mathbb{Z}^2 \to \mathcal{X} \subset \mathbb{R}^m$ defined on a discrete space V, several graph structures can be used. The ones based on geometric neighborhoods are particularly well-adapted to represent the geometry of the space, as well as the geometry of the function defined on that space. The most commonly used graph is the k-neighborhood graph $G_k = (V, E, w)$, where the k-neighborhood of a vertex $u = (i, j) \in V$ is the set of vertices located at a non-null distance lower than k:

$$\mathcal{N}_k(u) = \{v = (i', j') \in V \setminus \{u\} \ : \ \mu(u, v) \leq k, \ k > 0\},$$

where $\mu : V \times V \to \mathbb{R}^+$ measures the proximity between two vertices. Then an edge (u, v) is in G_k iff $v \in \mathcal{N}_k(u)$ (and reciprocally). By using the Chebyshev distance $\mu((i, j), (i', j')) = \max\{|i - i'|, |j - j'|\}$, the shape of the neighborhood corresponds to the standard square widow of size $2k + 1$. In particular, G_1 is the 8-adjacency graph of pixels, the 4-adjacency graph of pixels is noted G_0, and the complete graph G_∞. We can also consider other distances to construct the neighborhood, such as the Euclidean distance. The similarity between two connected vertices is described by the weight function w. In the sequel, we use the two following ones, which allow to retrieve and to extend several filtering processes:

$$\begin{aligned} w_1(u,v) &= \exp\left(-\frac{\|u-v\|^2_{\mathcal{L}^2(\mathbb{R}^2)}}{\sigma_P^2}\right)\exp\left(-\frac{\|f^0(u)-f^0(v)\|^2_{\mathcal{H}(V)}}{\sigma_{\mathcal{X}}^2}\right) \\ w_2^{k'}(u,v) &= \exp\left(-\frac{\rho_a(F_{k'}^{f^0}(u), F_{k'}^{f^0}(v))}{h^2}\right) \end{aligned}$$

where $F_{k'}^{f^0}(u) \in \mathcal{X}^{(2k'+1)(2k'+1)}$ is the local feature corresponding to the values of f^0 in the neighborhood $\mathcal{N}_{k'}(v) \cup \{u\}$, with $0 \le k' < k$ fixed:

$$F_{k'}^{f^0}(u) = \{f^0(v) \ : \ v \in \mathcal{N}_{k'}(u) \cup \{u\}\}.$$

The function ρ_a measures the distance between the values of f^0 in the neighborhood $\mathcal{N}_{k'}$:

$$\rho_a(F_{r'}^{f^0}(u), F_{r'}^{f^0}(v)) = \sum_{i=-r'}^{r'}\sum_{j=-r'}^{r'} g_a((i,j))\|f^0(u+(i,j)) - f^0(v+(i,j))\|_2^2,$$

where g_a is a Gaussian kernel of standard deviation a. This latter can be replaced by the Chebyshev distance between the position of pixels.

The discrete diffusion associated with the weight function w_1 and the graph G_∞ is semilocal, and corresponds to the bilateral filter (Tomasi and Manduchi, 1998) if $p = 2$, $\lambda = 0$ and one iteration. The discrete diffusion associated with the weight function $w_2^{k'}$ and the graph G_∞ is nonlocal, according to the similarity measure. For $p = 2$, $\lambda = 0$ and one iteration, this latter diffusion corresponds to the NLM filter (Buades et al., 2005). For several iterations, these two cases can be seen as iterated bilateral and NLM filters, without updating the weights at each iteration[1]. Iterated versions of these filters, with the weights being updated at each iteration, exist in the literature. See for example (Paris et al., 2007) for a recent survey of the bilateral filter and its variants, and (Brox and Cremers, 2007) for the iterated NLM filter. More generally, for $p = 2$ and any weight function, the discrete diffusion performs weighted (combinatorial) Laplacian smoothing.

Another particular case of the discrete diffusion process (31) is the TV digital filter (Osher and Shen, 2000; Chan et al., 2001), obtained with $p = 1$ and $w(u,v) = 1$ for all $(u,v) \in E$. Due to the constant weight function, the size

[1] The weights are updated at each iteration if they depend on the filtered function f.

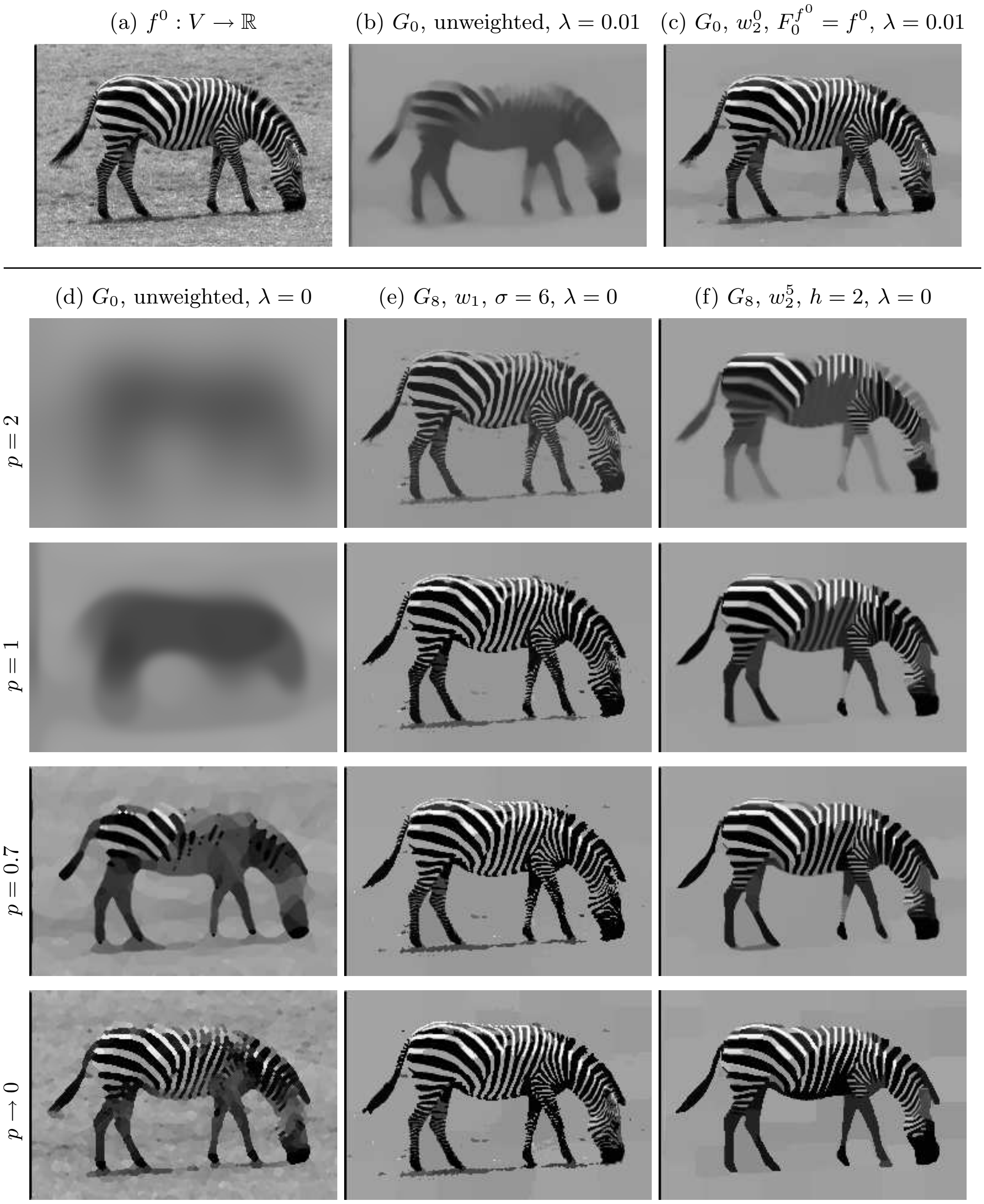

Figure 1. (a) The initial image f^0 is regularized until convergence of the discrete diffusion process (31). (b) Discrete TV regularization. (c) Discrete weighted-TV regularization. (d), (e) and (f): Behavior of the regularization with $\lambda = 0$ and 800 iterations of (31). On G_0, it corresponds to the unweighted Laplacian smoothing for $p = 2$ and to the digital TV filter for $p = 1$. On G_8 with w_1, it is the iterative bilateral filter (without updating the weights) for $p = 2$. On G_8 with w_2^5, it is the iterative NLM filter (without updating the weights). The other cases are the ones proposed in this paper.

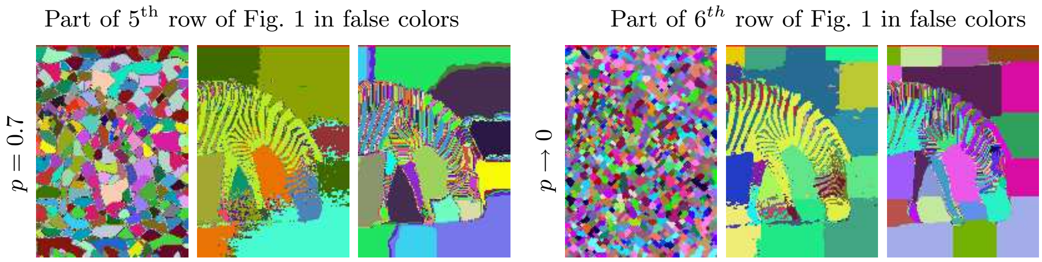

Figure 2. Results presented in Fig. 1 for $p < 1$ and rendered here in false colors (each color corresponds to a gray value). We can observe the relation between the size of the neighborhood and the leveling of the image.

of the neighborhood of the graph cannot be greater than one to preserve the discontinuities. By using a weighted graph, we obtain the weighted-TV digital filter, which can be local, semilocal or nonlocal, depending on the weight function. The difference between the weighted and unweighted cases is illustrated in Fig. 1(b) and 1(c) on the graph G_0, and until convergence of the diffusion. We can observe that for the same value of λ, using a weight function helps to preserve the image discontinuities.

In order to compare the results with the particular cases described above, we give examples of the proposed regularization, for several values of p, and the weight functions w_1 and w_2.

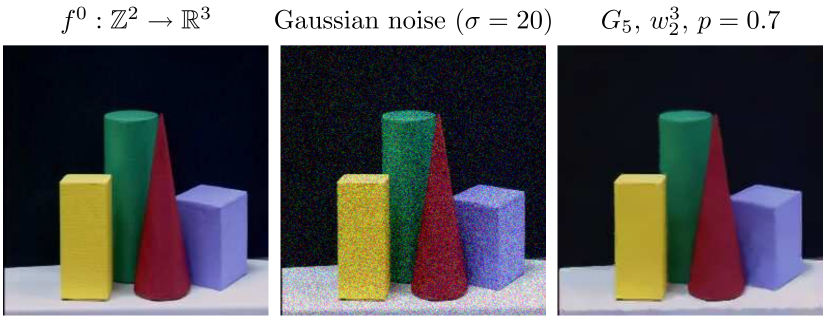

Figure 3. Denoising of a color image by nonlocal regularization with 4 iterations and $\lambda = 0.01$.

Image smoothing/denoising. The behavior of the proposed regularization is illustrated in Fig. 1(d), 1(e) and 1(f) on an intensity image, for several values of p, several graph structures and $\lambda = 0$ (without the approximation term). The number of iterations is the same for all the cases (800). We can do two principal observations. The size of neighborhood of the graph helps to preserve sharp edges and image redundancies, as well as the use of nonlocal weights. Also, when $p < 1$ and particularly when $p \rightarrow 0$, the regularization behaves like a simplification procedure. This last observation is depicted in the first row of Fig. 2, where we can see the effect of the structure of the graph. The local case (with G_0), which is computed efficiently, could be used in simplification and segmentation processes. The denoising of a color image is illustrated in Fig. 3 using a nonlocal representation with $p = 0.7$. In our experiments, we found that using $p < 1$ in the regularization process, with a local or a nonlocal representations, helps to preserve sharp edges.

Image simplification. Another way to simplify an image is to work on a more abstract representation than adjacency or neighborhood graphs. One possible representation is obtained by constructing a fine partition (or over-segmentation) of the image and by considering neighborhood relations between the regions. It is generally the first step of segmentation schemes and it provides a reduction of the number of elements to be analyzed by other processing methods. To compute the fine partition, many methods have been proposed, such as the ones based on morphological operators (Meyer, 2001) or graph cut techniques and random walks (Meila and Shi, 2000). Here, we present a method that uses a graph-based version of the generalized Voronoi diagram presented by (Arbeláez and Cohen, 2004). The initial image to be simplify is represented by a graph $G_k = (V, E)$, $k = 0$ or 1, and a function $f : V \to \mathbb{R}^m$, as described previously.

A *path* $c(u, v)$ is a sequence of vertices $(v_1, \ldots, v_m)$ such that $u = v_1$, $v = v_m$, and $(v_i, v_{i+1}) \in E$ for all $1 \leq i < m$. Let $C_G(u, v)$ be the set of paths connecting u and v. We define the pseudo-metric $\mu : V \times V \to \mathbb{R}^+$ to be:

$$\mu(u, v) = \min_{c \in C_G(u,v)} \left(\sum_{i=1}^{m-1} \|d_w(f)(v_i, v_{i+1})\| \right),$$

where d_w is the difference operator (4) defined in Section 2.2. Given a finite set of source vertices $S = \{s_1, \ldots, s_k\} \subset V$, the *energy* induced by μ is given by the minimal individual energy:

$$\mu_S(u) = \inf_{s_i \in S} \mu(s_i, u), \qquad \forall u \in V.$$

Based on the pseudo-metric μ, the *influence zone* of a source vertex s_i is defined to be the set of vertices of V that are closer to s_i than to any other source vertex of S:

$$Z_\mu(s_i, S, V) = \{u \in V \ : \ \mu(s_i, u) \leq \mu(s_j, u), \forall s_j \in S\}.$$

The *energy partition* of the graph G, with respect to the set of sources S and the pseudo-metric μ, corresponds to the set of influence zones:

$$E_\mu(S, G) = \{Z_\mu(s_i, S, V), \forall s_i \in S\}.$$

With these definitions, the image pre-segmentation consists in finding a set of source vertices and a pseudo-metric. We use the set of extrema of the intensity of the function f as a set of source vertices. To obtain exactly an energy partition which considers the total variation of f along a path, we use $d_w(f)(u, v) = f(v)-$

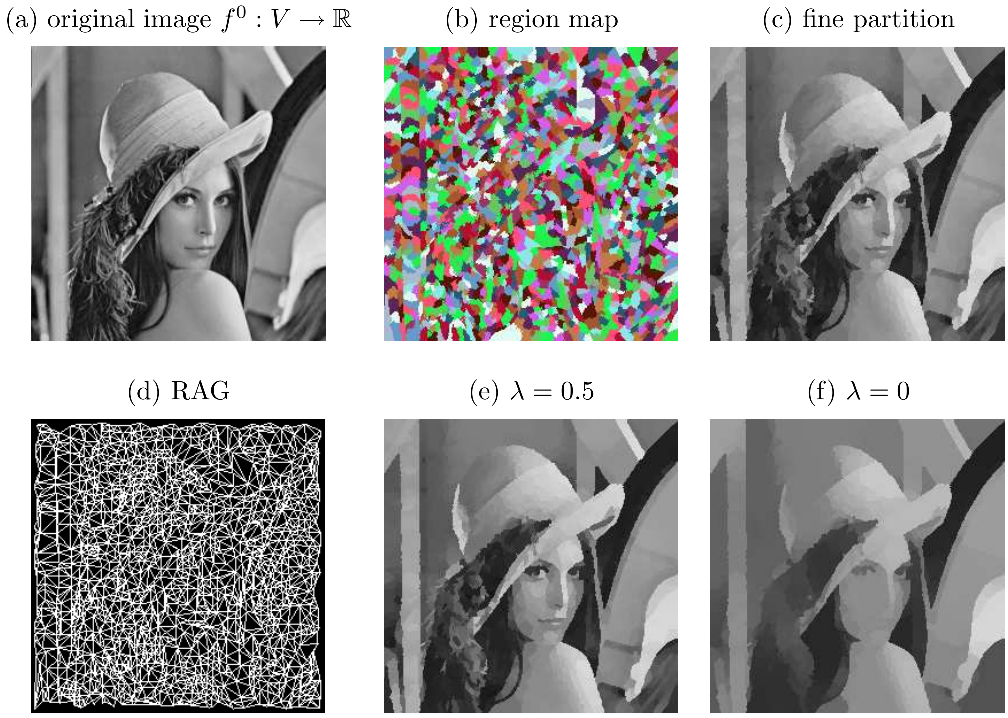

Figure 4. Illustration of image simplification. First row: construction of the fine partition by energy partition. The information in the fine partition is 8 percent of the one in the original image. Second row: regularization of the fine partition on the RAG with $p = 2$, w_2^0 with $F_0^f = f$, and 30 iterations.

$f(u)$ in the pseudo-metric. Then, the energy partition of the graph represents an approximation of the image, by assigning a model to each influence zone of the partition. The model is determined by the distribution of the graph values on the influence zone. Among the different models, the simplest are the constant ones, as mean or median value of the influence zone. The resultant graph $G' = (V', E')$ is a connectivity graph where $V' = S$ and E' is the set of edges connecting two vertices $s_i, s_j \in S$ if there exists a vertex of $Z_\mu(s_i, S, V)$ connected to a vertex of $Z_\mu(s_j, S, V)$. This last graph is known as the region adjacency graph (RAG) of the partition. Therefore, image simplification can be performed on the RAG (or a more complex neighborhood graph) and so the graph regularization can be computed much faster relatively to classical images. The acceleration factor depends on the fineness of the partition and on the considered graph.

A result of the simplification process is illustrated in Fig. 4 on an image of intensity. Fig. 4(b) represents the partition in random colors, and Fig. 4(c) the reconstructed image from the computed influence zones by assigning the mean intensity to modelize each zone. In our experiments, we found that the reduction of the information is at least of 90 percent. As illustrated in Fig. 4(e) and (f), the simplification scheme can be carried on by regularizing the value of the zones on the RAG associated to the fine partition. This simplification scheme has shown its effectiveness in the context of image segmentation, with other methods to construct the fine partition (Lezoray et al., 2007).

5.3. Application to polygonal mesh processing

By nature, polygonal curves and surfaces have a graph structure. 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 $f^0 : V \subset \mathbb{R}^n \to \mathbb{R}^m$ defined on the graph $G = (V, E, w)$.

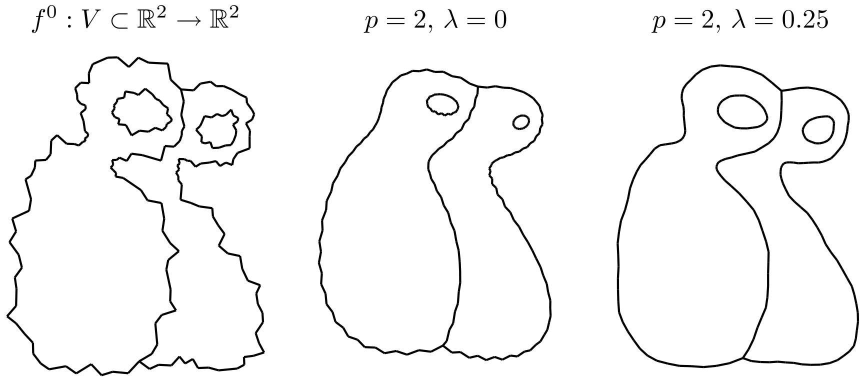

Figure 5. Polygonal curve denoising by diffusion of the position of the vertices. The polygons edges are unweighted. In the case of $\lambda = 0$ shrinkage effects are introduced (30 iterations). The case of $\lambda > 0$ (100 iterations) helps to avoid these undesirable effects.

Mesh denoising. The discrete regularization can be used to smooth and denoise polygonal meshes. As in image processing, denoising a polygonal curve or surface consist in removing spurious details while preserving geometric features. There exists two common frameworks in the literature to do this. The first one considers the position of the vertices as the function to be processed. The second one consists in regularizing the normals direction at the mesh vertices.

We illustrate the first scheme in order to show the importance of using the fitting term in regularization processes, which is not commonly used in mesh processing. This is illustrated Fig. 5 on a polygonal curve. One can observe that the regularization performed using the fitting term ($\lambda > 0$) helps to avoid the shrinkage effects obtained without using the fitting term ($\lambda = 0$), and which corresponds to the Laplacian smoothing (Taubin, 1995). The regularization of polygonal surfaces is illustrated in Fig. 6. Here again, the fitting term helps to avoid shrinkage effects. On can remark that we use the discrete diffusion (31), which is not the classical algorithm to perform Laplacian smoothing in mesh processing. Most of the methods are based on the algorithm (24) with $\lambda = 0$ (Taubin, 1995; Desbrun et al., 2000; Xu, 2004). The proposed p-Laplacian diffusion, which extends the Laplacian diffusion, is described next for $p < 1$.

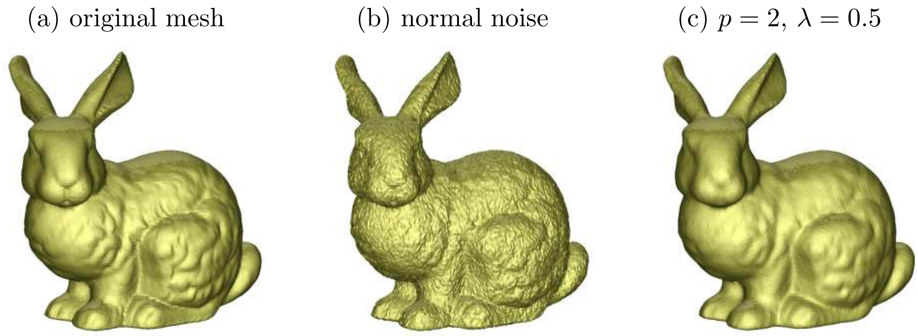

Figure 6. Mesh denoising. (a) Original Stanford Bunny with $|V| = 35949$. (b) Noisy Bunny with normal noise. (c) Regularization of vertex coordinates on the graph (8 iterations of the diffusion process and $w(u, v) = 1/\|u - v\|_2^2$).

Curve and surface simplification. As in the case of images, when $p < 1$ the regularization process can be seen as a clustering method. This is illustrated in Fig. 7 on a polygonal curve, and in Fig. 8 on a polygonal surface. One can observe that when $p \to 0$, the vertices aggregates. Also, the global shape of the curve is preserved, as well as the discontinuities, without important shrinkage effects. This provides a new way of simplifying meshes.

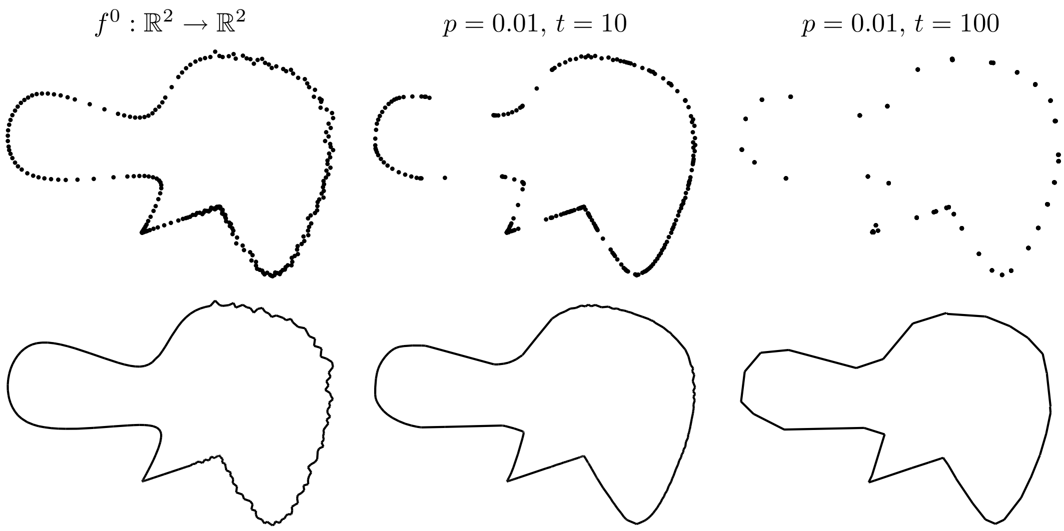

Figure 7. Polygonal curve simplification by regularization of the position of the vertices (with $p < 1$). The graph is the polygon itself, and $w = 1$. First row: the vertices. Second row: the associated processed polygons.

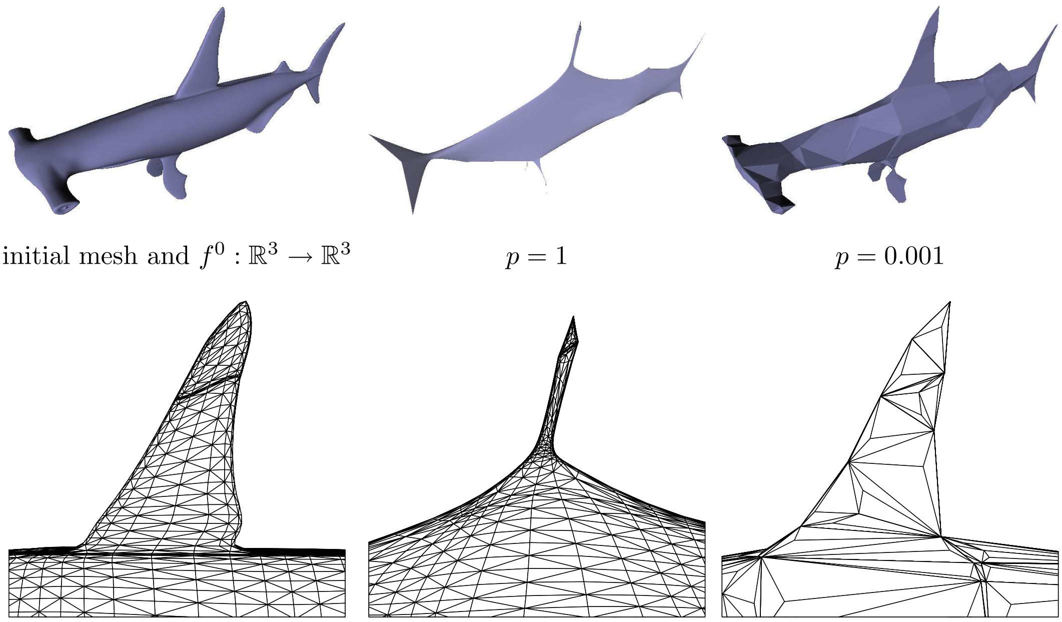

Figure 8. Polygonal surface simplification by discrete diffusion until convergence, with $\lambda = 0.001$ and $w = 1$.

Skeleton simplification. The regularization framework is also interesting to help the estimation of geometric and topological features such as normals, curvatures, or shape skeletons. Fig.9 illustrates the simplification of the skeleton of a polygonal curve by smoothing the position of its vertices. Here the skeleton is given by a subgraph of the (Euclidean) Voronoi diagram computed from the set of the polygon vertices, see for example (Bougleux et al., 2007b). The graph is the polygon itself and $w = 1$. After the diffusion step, the skeleton is extracted a second time. We can observe that the branches of the skeleton, related to the smoother parts of the initial curve, are not affected by the diffusion, while the other parts are simplified.

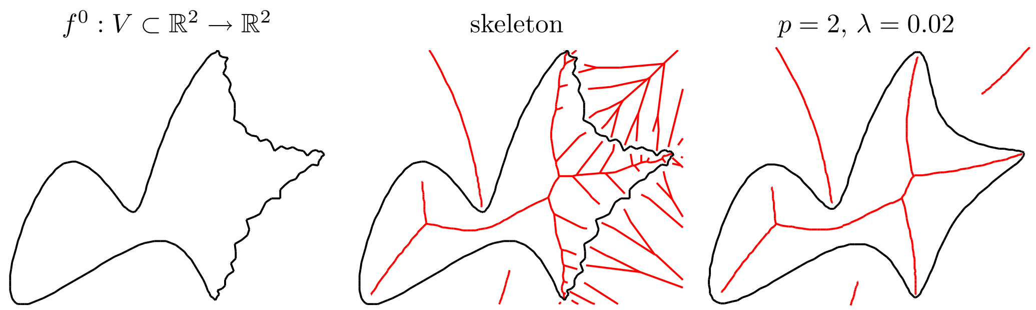

Figure 9. Skeleton simplification by smoothing the position of the vertices. The edges of the polygon are unweighted. From left to right: the original polygonal curve, its skeleton, and the simplified skeleton obtained with 10 iterations of the discrete diffusion process.

6. Conclusion

We propose a general discrete framework for regularizing real-valued or vector-valued functions on weighted graphs of arbitrary topology. The regularization, based on a discrete p-Dirichlet energy, leads to a family of nonlinear iterative processes which includes several filters used in image and mesh processing.

The choice of the graph topology and the choice of the weight function allow to regularize any discrete data set or any function defined on a discrete data set. Indeed, the data can be structured by neighborhood graphs weighted by functions depending on data features. This can be applied in the context of image smoothing, denoising or simplification. We also show that local and nonlocal regularization functionals have the same expression when defined on graphs. The main ongoing work is to use the proposed framework in the context of hierarchical mesh segmentation and point cloud clustering.

Appendix

A. Proofs of Section 2

Proof of Proposition 1: From the expressions of the inner product in $\mathcal{H}(E)$ (Eq. (2)) and the difference operator, the left side of Eq. (6) is written as:

$$\begin{aligned}\langle H, d_w f\rangle_{\mathcal{H}(E)} &= \sum_{xy\in E} H(xy)\left(\gamma_{yx} f(y) - \gamma_{xy} f(x)\right)\\ &= \sum_{xy\in E} \gamma_{yx} H(xy) f(y) - \sum_{xy\in E} \gamma_{xy} H(xy) f(x).\end{aligned}$$

By replacing $\sum_{xy\in E}$ by $\sum_{x\in V}\sum_{y\sim x}$, and x and y by u and v, we have:

$$\begin{aligned}\langle H, d_w f\rangle_{\mathcal{H}(E)} &= \sum_{u\in V}\sum_{v\sim u} \gamma_{uv} H(v,u) f(u) - \sum_{u\in V}\sum_{v\sim u} \gamma_{uv} H(u,v) f(u)\\ &= \sum_{u\in V} f(u) \sum_{v\sim u} \gamma_{uv}\left(H(v,u) - H(u,v)\right) \qquad (32)\\ &\overset{(6)}{=} \langle d_w^* H, f\rangle_{\mathcal{H}(V)} \overset{(1)}{=} \sum_{u\in V} d_w^*(H)(u) f(u). \qquad (33)\end{aligned}$$

Then, the result is obtained from Eq. (32) and Eq. (33) by taking $f(u) = 1$, for all $u \in V$. □

Proof of Proposition 2: From the definition of the p-Laplace operator (Eq. (11)), and the expression of the difference operator and its adjoint (Proposition 1), we have:

$$\begin{aligned}\Delta_w^p f(u) &= \sum_{v\sim u} \gamma_{uv}\left(|\bigtriangledown_w f(v)|^{p-2} d_w(f)(v,u) - |\bigtriangledown_w f(u)|^{p-2} d_w(f)(u,v)\right)\\ &= \sum_{v\sim u} \gamma_{uv}\left(|\bigtriangledown_w f(v)|^{p-2} + |\bigtriangledown_w f(u)|^{p-2}\right) d_w(f)(v,u). \ \square\end{aligned}$$

Now, we show that Eq. (13) is equal to Eq. (12), using the definition of the edge derivative (Eq. (5)):

$$\begin{aligned}(13) &= \sum_{\substack{v\sim u\\ e=(u,v)}} \gamma_{uv}\left(\frac{\gamma_{vu}|\bigtriangledown_w f(v)|^{p-2}}{\gamma_{vu}}\frac{\partial f}{\partial e}\bigg|_v - \frac{\gamma_{uv}|\bigtriangledown_w f(u)|^{p-2}}{\gamma_{uv}}\frac{\partial f}{\partial e}\bigg|_u\right)\\ &= \sum_{\substack{v\sim u\\ e=(u,v)}} \gamma_{uv}\left(|\bigtriangledown_w f(v)|^{p-2}(\gamma_{uv}f(u) - \gamma_{vu}f(v))\right.\\ &\qquad\qquad \left. -|\bigtriangledown_w f(u)|^{p-2}(\gamma_{vu}f(v) - \gamma_{uv}f(u))\right)\\ &= (12). \ \square\end{aligned}$$

B. Proof of Section 3

Proof of Property 2: The partial derivative of $R_w^{p\text{-}TV}(f)$, at a vertex $u_1 \in V$ is given by:

$$\frac{\partial}{\partial f}\left(\sum_{u\in V}|\bigtriangledown_w f(u)|^p\right)\bigg|_{u_1} \overset{(9)}{=} \frac{\partial}{\partial f}\left(\sum_{u\in V}\left(\sum_{v\sim u}(\gamma_{vu}f(v) - \gamma_{uv}f(u))^2\right)^{\frac{p}{2}}\right)\bigg|_{u_1}. \tag{34}$$

The derivative depends only on the edges incident to u_1. Let $v_1, \ldots, v_k$ be the vertices of V connected to u_1 by an edge of E. Then we have:

$$\begin{aligned}(34) = -p\sum_{v\sim u_1}\gamma_{u_1v}(\gamma_{vu_1}f(v) - \gamma_{u_1v}f(u_1))\left(\sum_{v\sim u_1}(\gamma_{vu_1}f(v) - \gamma_{u_1v}f(u_1))^2\right)^{\frac{p-2}{2}}\\ + p\gamma_{u_1v_1}(\gamma_{u_1v_1}f(u_1) - \gamma_{v_1u_1}f(v_1))\left(\sum_{v\sim v_1}(\gamma_{vv_1}f(v) - \gamma_{v_1v}f(v_1))^2\right)^{\frac{p-2}{2}}\\ + \ldots + p\gamma_{u_1v_k}(\gamma_{u_1v_k}f(u_1) - \gamma_{v_ku_1}f(v_k))\left(\sum_{v\sim v_k}(\gamma_{vv_k}f(v) - \gamma_{v_kv}f(v_k))^2\right)^{\frac{p-2}{2}}\\ \overset{(9)}{=} p\sum_{v\sim u_1}\gamma_{u_1v}(\gamma_{u_1v}f(u_1) - \gamma_{vu_1}f(v))|\bigtriangledown_w f(u_1)|^{p-2}\\ + p\sum_{v\sim u_1}\gamma_{u_1v}(\gamma_{u_1v}f(u_1) - \gamma_{vu_1}f(v))|\bigtriangledown_w f(v)|^{p-2} \overset{(12)}{=} p\Delta_w^p f(u_1). \ \square\end{aligned}$$

References

Alvarez, L., F. Guichard, P.-L. Lions, and J.-M. Morel: 1993, 'Axioms and fundamental equations of image processing'. *Archive for Rational Mechanics and Analysis* **123**(3), 199–257.

Arbeláez, P. A. and L. D. Cohen: 2004, 'Energy Partitions and Image Segmentation'. *Journal of Mathematical Imaging and Vision* **20**(1-2), 43–57.

Aubert, G. and P. Kornprobst: 2006, *Mathematical Problems in Image Processing, Partial Differential Equations and the Calculus of Variations*, No. 147 in Applied Mathematical Sciences. Springer, 2nd edition.

Bajaj, C. L. and G. Xu: 2003, 'Anisotropic diffusion of surfaces and functions on surfaces'. *ACM Trans. on Graph.* **22**(1), 4–32.

Barash, D.: 2002, 'A Fundamental Relationship between Bilateral Filtering, Adaptive Smoothing, and the Nonlinear Diffusion Equation'. *IEEE Trans. Pattern Analysis and Machine Intelligence* **24**(6), 844–847.

Bensoussan, A. and J.-L. Menaldi: 2005, 'Difference Equations on Weighted Graphs'. *Journal of Convex Analysis* **12**(1), 13–44.

Bobenko, A. I. and P. Schröder: 2005, 'Discrete Willmore Flow'. In: M. Desbrun and H. Pottmann (eds.): *Eurographics Symposium on Geometry Processing*. pp. 101–110.

Bougleux, S. and A. Elmoataz: 2005, 'Image Smoothing and Segmentation by Graph Regularization.'. In: *Proc. Int. Symp. on Visual Computing (ISVC)*, Vol. 3656 of *LNCS*. pp. 745–752, Springer.

Bougleux, S., A. Elmoataz, and M. Melkemi: 2007a, 'Discrete Regularization on Weighted Graphs for Image and Mesh Filtering'. In: F. Sgallari, A. Murli, and N. Paragios (eds.): *Proc. of the 1st Int. Conf. on Scale Space and Variational Methods in Computer Vision (SSVM)*, Vol. 4485 of *LNCS*. pp. 128–139, Springer.

Bougleux, S., M. Melkemi, and A. Elmoataz: 2007b, 'Local Beta-Crusts for Simple Curves Reconstruction'. In: C. M. Gold (ed.): *4th International Symposium on Voronoi Diagrams in Science and Engineering (ISVD'07)*. pp. 48–57, IEEE Computer Society.

Brox, T. and D. Cremers: 2007, 'Iterated Nonlocal Means for Texture Restoration'. In: F. Sgallari, A. Murli, and N. Paragios (eds.): *Proc. of the 1st Int. Conf. on Scale Space and Variational Methods in Computer Vision (SSVM)*, Vol. 4485 of *LNCS*. pp. 12–24, Springer.

Buades, A., B. Coll, and J.-M. Morel: 2005, 'A review of image denoising algorithms, with a new one'. *Multiscale Modeling and Simulation* **4**(2), 490–530.

Chambolle, A.: 2005, 'Total Variation Minimization and a Class of Binary MRF Models'. In: *Proc. of the 5th Int. Work. EMMCVPR*, Vol. 3757 of *LNCS*. pp. 136–152, Springer.

Chan, T., S. Osher, and J. Shen: 2001, 'The Digital TV Filter and Nonlinear Denoising'. *IEEE Trans. Image Processing* **10**(2), 231–241.

Chan, T. and J. Shen: 2005, *Image Processing and Analysis - variational, PDE, wavelets, and stochastic methods*. SIAM.

Chung, F.: 1997, 'Spectral Graph Theory'. *CBMS Regional Conference Series in Mathematics* **92**, 1–212.

Clarenz, U., U. Diewald, and M. Rumpf: 2000, 'Anisotropic geometric diffusion in surface processing'. In: *VIS'00: Proc. of the conf. on Visualization*. pp. 397–405, IEEE Computer Society Press.

Coifman, R., S. Lafon, A. Lee, M. Maggioni, B. Nadler, F. Warner, and S. Zucker: 2005, 'Geometric diffusions as a tool for harmonic analysis and structure definition of data'. *Proc. of the National Academy of Sciences* **102**(21).

Coifman, R., S. Lafon, M. Maggioni, Y. Keller, A. Szlam, F. Warner, and S. Zucker: 2006, 'Geometries of sensor outputs, inference, and information processing'. In: *Proc. of the SPIE: Intelligent Integrated Microsystems*, Vol. 6232.

Cvetković, D. M., M. Doob, and H. Sachs: 1980, *Spectra of Graphs, Theory and Application*, Pure and Applied Mathematics. Academic Press.

Darbon, J. and M. Sigelle: 2004, 'Exact Optimization of Discrete Constrained Total Variation Minimization Problems'. In: R. Klette and J. D. Zunic (eds.): *Proc. of the 10th Int. Workshop on Combinatorial Image Analysis*, Vol. 3322 of *LNCS*. pp. 548–557.

Desbrun, M., M. Meyer, P. Schröder, and A. Barr: 2000, 'Anisotropic Feature-Preserving Denoising of Height Fields and Bivariate Data'. *Graphics Interface* pp. 145–152.

Fleishman, S., I. Drori, and D. Cohen-Or: 2003, 'Bilateral mesh denoising'. *ACM Trans. on Graphics* **22**(3), 950–953.

Friedman, J. and J.-P. Tillich: 2004, 'Wave equations for graphs and the edge-based laplacian'. *Pacific Journal of Mathematics* **216**(2), 229–266.

Gilboa, G. and S. Osher: 2007a, 'Nonlocal linear image regularization and supervised segmentation'. *SIAM Multiscale Modeling and Simulation* **6**(2), 595–630.

Gilboa, G. and S. Osher: 2007b, 'Nonlocal Operators with Applications to Image Processing'. Technical Report 07-23, UCLA, Los Angeles, USA.

Griffin, L. D.: 2000, 'Mean, Median and Mode Filtering of Images'. *Proceedings: Mathematical, Physical and Engineering Sciences* **456**(2004), 2995–3004.

Hildebrandt, K. and K. Polthier: 2004, 'Anisotropic Filtering of Non-Linear Surface Features'. *Eurographics 2004: Comput. Graph. Forum* **23**(3), 391–400.

Jones, T. R., F. Durand, and M. Desbrun: 2003, 'Non-iterative, feature-preserving mesh smoothing'. *ACM Trans. Graph.* **22**(3), 943–949.

Kervrann, C., J. Boulanger, and P. Coupé: 2007, 'Bayesian non-local means filter, image redundancy and adaptive dictionaries for noise removal'. In: F. Sgallari, A. Murli, and N. Paragios (eds.): *Proc. of the 1st Int. Conf. on Scale Space and Variational Methods in Computer Vision (SSVM)*, Vol. 4485 of *LNCS*. pp. 520–532, Springer.

Kimmel, R., R. Malladi, and N. Sochen: 2000, 'Images as embedding maps and minimal surfaces: Movies, color, texture, and volumetric medical images'. *International Journal of Computer Vision* **39**(2), 111–129.

Kinderman, S., S. Osher, and S. Jones: 2005, 'Deblurring and denoising of images by nonlocal functionals'. *SIAM Multiscale Modeling and Simulation* **4**(4), 1091–1115.

Lee, J. S.: 1983, 'Digital Image Smoothing and the Sigma Filter'. *Computer Vision, Graphics, and Image Processing* **24**(2), 255–269.

Lezoray, O., A. Elmoataz, and S. Bougleux: 2007, 'Graph regularization for color image processing'. *Computer Vision and Image Understanding* **107**(1-2), 38–55.

Meila, M. and J. Shi: 2000, 'Learning segmentation by random walks'. *Advances in Neural Information Processing Systems* **13**, 873–879.

Meyer, F.: 2001, 'An overview of morphological segmentation'. *International Journal of Pattern Recognition and Artificial Intelligence* **15**(7), 1089–1118.

Mrázek, P., J. Weickert, and A. Bruhn: 2006, 'On robust estimation and smoothing with spatial and tonal kernels'. In: *Geometric Properties from Incomplete Data*, Vol. 31 of *Computational Imaging and Vision*. pp. 335–352, Springer.

Osher, S. and J. Shen: 2000, 'Digitized PDE method for data restoration'. In: E. G. A. Anastassiou (ed.): *In Analytical-Computational methods in Applied Mathematics*. Chapman&Hall/CRC, pp. 751–771.

Paragios, N., Y. Chen, and O. Faugeras (eds.): 2005, *Handbook of Mathematical Models in Computer Vision*. Springer.

Paris, S., P. Kornprobst, J. Tumblin, and F. Durand: 2007, 'A gentle introduction to bilateral filtering and its applications'. In: *SIGGRAPH '07: ACM SIGGRAPH 2007 courses*. ACM.

Peyré, G.: 2008, 'Image processing with non-local spectral bases'. to appear in SIAM Multiscale Modeling and Simulation.

Requardt, M.: 1997, 'A New Approach to Functional Analysis on Graphs, the Connes-Spectral Triple and its Distance Function'.

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

Smith, S. and J. Brady: 1997, 'SUSAN - a new approach to low level image processing'. *International Journal of Computer Vision (IJCV)* **23**, 45–78.

Sochen, N., R. Kimmel, and A. M. Bruckstein: 2001, 'Diffusions and Confusions in Signal and Image Processing'. *Journal of Mathematical Imaging and Vision* **14**(3), 195–209.

Szlam, A. D., M. Maggioni, and R. R. Coifman: 2006, 'A General Framework for Adaptive Regularization Based on Diffusion Processes On Graphs'. Technical Report YALE/DCS/TR1365, YALE.

Tasdizen, T., R. Whitaker, P. Burchard, and S. Osher: 2003, 'Geometric surface processing via normal maps'. *ACM Trans. on Graphics* **22**(4), 1012–1033.

Taubin, G.: 1995, 'A signal processing approach to fair surface design'. In: *SIGGRAPH'95: Proc. of the 22nd Annual Conference on Computer Graphics and Interactive Techniques*. pp. 351–358, ACM Press.

Tomasi, C. and R. Manduchi: 1998, 'Bilateral Filtering for Gray and Color Images'. In: *ICCV'98: Proc. of the 6th Int. Conf. on Computer Vision*. pp. 839–846, IEEE Computer Society.

Weickert, J.: 1998, *Anisotropic Diffusion in Image Processing*, ECMI series. Teubner-Verlag.

Xu, G.: 2004, 'Discrete Laplace-Beltrami operators and their convergence'. *Computer Aided Geometric Design* **21**, 767–784.

Yagou, H., Y. Ohtake, and A. Belyaev: 2002, 'Mesh smoothing via mean and median filtering applied to face normals'. In: *Proc. of the Geometric Modeling and Processing (GMP'02) - Theory and Applications*. pp. 195–204, IEEE Computer Society.

Yoshizawa, S., A. Belyaev, , and H.-P. Seidel: 2006, 'Smoothing by example: mesh denoising by averaging with similarity-based weights'. In: *Proc. International Conference on Shape Modeling and Applications*. pp. 38–44.

Zhou, D. and B. Schölkopf: 2005, 'Regularization on Discrete Spaces'. In: *Proc. of the 27th DAGM Symp.*, Vol. 3663 of *LNCS*. pp. 361–368, Springer.