Nonlocal PDEs-Based Morphology on Weighted Graphs for Image and Data Processing

Vinh-Thong Ta, Abderrahim Elmoataz, and Olivier Lézoray

Abstract—Mathematical morphology (MM) offers a wide range of operators to address various image processing problems. These operators can be defined in terms of algebraic (discrete) sets or as partial differential equations (PDEs). In this paper, we introduce a nonlocal PDEs-based morphological framework defined on weighted graphs. We present and analyze a set of operators that leads to a family of discretized morphological PDEs on weighted graphs. Our formulation introduces nonlocal patch-based configurations for image processing and extends PDEs-based approach to the processing of arbitrary data such as nonuniform high dimensional data. Finally, we show the potentialities of our methodology in order to process, segment and classify images and arbitrary data.

Index Terms—Adaptive operators, data clustering, mathematical morphology, nonlocal patch-based processing, partial differential equations (PDEs), weighted graphs.

I. INTRODUCTION

MATHEMATICAL morphology (MM) is a popular nonlinear approach for image processing that has found numerous applications including shape and texture analysis, biomedical image processing, document recognition or multiresolution techniques. First developed by [1] and [2], MM relies on a fundamental structure: the complete lattice [3]. With the acceptance of complete lattice theory, it is possible to define morphological operators for any type of image once a proper ordering is established. Within this model, morphological operators are represented as mappings between complete lattices in combination with matching patterns called structuring elements. In particular, the two fundamental operators in flat MM are dilation and erosion. For a scalar function

$f^0 : \Omega \subset \mathbb{R}^2 \to \mathbb{R}$, $x \in \Omega$, and a structuring element $B \subset \Omega$, dilation and erosion of f^0 w.r.t. B are

$$\delta_a \colon \sup \left\{ f^0(x-z) \colon z \in B \right\} \text{ and } \varepsilon_a \colon \inf \left\{ f^0(x+z) \colon z \in B \right\}.$$

These two operators form the basis of many other morphological processes such as opening/closing or reconstruction filters [4], [5]. As a consequence, the implementation of such algebraic morphological operators is usually performed within a discrete setting.

For convex structuring elements, an alternative to the algebraic formulation relies on nonlinear partial differential equations (PDEs) [6]–[8]. This framework has several advantages. First, it offers excellent results for nondigitally scalable structuring elements whose shapes cannot be correctly represented on a discrete grid. Second, it allows subpixel accuracy and third, it can be adaptive by introducing, for instance, a local speed evolution term [9].

Let $B = \left\{ z : \|z\|_p \leq 1 \right\}$ be a structuring set induced by the $\mathcal{L}_p$ norm. The flat dilation and erosion of f^0 w.r.t. tB ($t > 0$) can be obtained by the following PDEs:

$$\partial_t f(x,t) = \|\nabla f(x,t)\|_p \quad \text{and}$$
$$\partial_t f(x,t) = -\|\nabla f(x,t)\|_p \tag{1}$$

where f is a modified version of f^0, ∇ is the gradient operator, and the initial condition is $f(.,0) = f^0(.)$.

These versions of dilation and erosion are also the basis of other continuous morphological operators such as leveling or continuous watershed with eikonal equation (see for instance [7], [10], [11] and references therein). To be applicable to images, (1) are discretized and specific numerical schemes are used: for instance, see [12]–[14], and, more recently, a variant of the flux corrected transport technique [15], [16] used for tensor images and matrix morphological processing [17].

Whatever the chosen formulation, if MM is well defined for binary and gray-scale images, the extension to multivariate unorganized data sets is not straightforward [18]. Algebraic formulation requires an ordering on vectors and several orderings have been reported in literature [19] but none of them has yet been widely accepted. Multivariate high dimensional data processing with PDEs-based formulations can be difficult due to the space discretization of irregular domain.

Adaptive algebraic and PDEs-based morphology have recently received a lot of attention. Such operators can be adaptive with spatially-variant structuring elements or intensity level-adaptive. For instance, one can quote the following works

[20]–[22] for image processing or [23]–[25] that extend the property of translation invariance of morphological operators in a spatially-varying morphology framework. The interested reader can also refer to [26] and references therein for a recent overview on adaptive morphology.

In this paper, we propose another approach to design adaptive morphological operators for images. Indeed, nonlocal patch-based schemes have recently received a lot of attention. First introduced for texture synthesis [27], these schemes have then been used for image processing [28]–[31]. These latter works have shown the efficiency of nonlocal patch-based configurations as compared to the local ones. In this work, we propose a set of morphological operators that enables nonlocal patch-based configurations for image processing.

1) Previous Works and Contributions: In [30], [32], we have introduced discrete version of continuous regularization models. These works are based upon new definitions of non-local operators over graphs that enable images, meshes or data processing. Inspired by these works, we have proposed in a conference paper [33] the basis of a new formulation of MM that considers a discrete version of PDEs-based approaches over weighted graphs. An extension of this work has been proposed that formally introduced our morphological differential operators for nonlocal patch-based image and data processing [34]. Our formulations have also been used to extend PDEs-based leveling [35]. Finally, with the same ideas, we have recently adapted the eikonal equation for data clustering and image segmentation [36].

In this work, we present a self-contained paper and a framework that unifies all these previous works. This framework is composed of a collection of adaptive nonlocal morphological operators and equations defined over weighted graphs. We present the relationships between these PDEs-based morphological operators on weighted graphs, continuous and algebraic morphological systems, eikonal equation and other adaptive approaches. Finally, our formulation has the advantage of naturally enabling local and nonlocal patch-based configurations. This point introduces a new family of nonlocal patch-based operators for image processing and segmentation by morphological tools.

The paper is structured as follows. Section II defines and recalls operators on graphs. Section III presents our MM framework: we define basic dilation and erosion as well as leveling and the adaptation of the eikonal equation. Section IV provides basic notions on weighted graphs and key points for their construction. This section also illustrates the adaptivity of our operators with graph weights and topology. Section V shows experiments and illustrates local and nonlocal patch-based image processing and segmentation. Potentialities of our methodology in order to process and cluster any data are also presented. Last section concludes.

II. OPERATORS ON WEIGHTED GRAPHS

In this section, operators on weighted graphs that constitute the basis of our morphological framework are detailed.

A. Notations

We consider the general situation where any discrete domain can be viewed as a weighted graph. Let $G = (V, E, w)$ be a weighted graph composed of a finite set of vertices V and a finite set of weighted edges $E \subset V \times V$. An edge $uv \in E$ connects two adjacent (neighbor) vertices u and v of V. The set of neighbors of a vertex u is denoted by $\mathcal{N}(u) = \{v \in V \backslash \{u\} : uv \in E\}$. The weight $w : E \to [0, 1]$ of an edge can be defined by w_{uv}, if $uv \in E$ and $w_{uv} = 0$ otherwise. Graphs are assumed to be simple, connected and undirected [37], implying that $w_{uv} = w_{vu}, \forall uv \in E$.

B. Weighted Difference On Graphs

All the basic operators considered in this paper are defined from the difference operator. There exists several definitions of this operator on graphs (see for instance [38]). Here, we consider a definition that allows the expression of weighted gradient on graphs. The weighted difference [32] of a function $f : V \to \mathbb{R}$ is

$$(d_w f)(uv) = \sqrt{w_{uv}} \big(f(v) - f(u) \big), \qquad \forall uv \in E. \quad (2)$$

Based upon this operator, we define the external and internal weighted differences by

$$(d_w^+ f)(uv) = \sqrt{w_{uv}} \big(\max \big(f(v), f(u) \big) - f(u) \big) \quad \text{and}$$
$$(d_w^- f)(uv) = \sqrt{w_{uv}} \big(f(u) - \min \big(f(u), f(v) \big) \big) \quad (3)$$

with the following property $(d_w^- f)(uv) = (d_w^+ f)(vu)$. These two operators recover classical difference operators and extend them to weighted graphs that enables more adaption in the difference computation.

C. Gradients on Graphs and Norms

The weighted gradient of a function $f : V \to \mathbb{R}$ at a vertex u is the vector of all edge differences

$$(\nabla_w f)(u) = \big((d_w f)(uv) \big)_{uv \in E}^T. \quad (4)$$

We also introduce the internal and external weighted gradients based upon (3) such that for a vertex u [33], [34]

$$(\nabla_w^+ f)(u) = \big((d_w^+ f)(uv) \big)_{uv \in E}^T \quad \text{and}$$
$$(\nabla_w^- f)(u) = \big((d_w^- f)(uv) \big)_{uv \in E}^T. \quad (5)$$

The external gradient is an operator defined as the difference between an extensive operator and a function (a typical one being the max). Similarly, the internal gradient uses an anti-extensive (the min) operator [39].

In the sequel, we use the $\mathcal{L}_p$ and $\mathcal{L}_\infty$ norms of (5). For a vertex u, we have

$$\|(\nabla_w^\pm f)(u)\|_p = \left[\sum_{v \sim u} w_{uv}^{p/2} |\mathcal{D}_v^\pm f(u)|^p \right]^{1/p} \quad \text{and} \quad (6)$$

$$\|(\nabla_w^\pm f)(u)\|_\infty = \max_{v \sim u} \big(\sqrt{w_{uv}} |\mathcal{D}_v^\pm f(u)| \big). \quad (7)$$
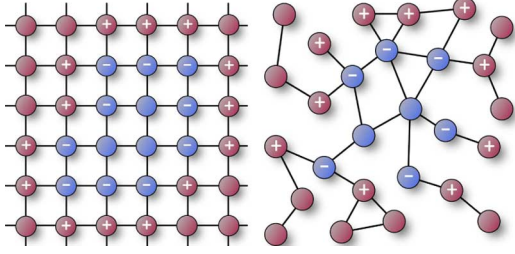
Fig. 1.   Graph boundary sets. Blue vertices: $\mathcal{A}$ set. "Plus" and "minus" vertices: $\partial^+\mathcal{A}$ and $\partial^-\mathcal{A}$ sets, respectively.

With the following notations. $v \sim u$ will denote that vertices u and v are adjacent. For two vertices u and v

$$\mathcal{D}_v^+ f(u) = \max\big(0, f(v) - f(u)\big) \quad \text{and}$$
$$\mathcal{D}_v^- f(u) = \min\big(0, f(v) - f(u)\big).$$

The general definitions presented in this section are defined over weighted graphs. They can therefore be used to process any discrete data that can be represented by a graph. In the sequel, we will show how these operators leads to a family of PDEs-based morphological processes on weighted graphs.

D. Relations With Algebraic Morphological Operators

For the particular case of an unweighted graph (i.e., $w_{uv} = 1$, $\forall uv \in E$) and with $p = \infty$, (7) corresponds to algebraic morphological operators where the structuring element is provided by the graph neighborhood [40]. Hence, the $\mathcal{L}_\infty$ norm of $\nabla_w^\pm$ are the definitions of algebraic external and internal gradients

$$\|(\nabla_w^+ f)(u)\|_\infty = \max_{v \sim u}\big(f(u), f(v)\big) - f(u)$$
$$= \delta_a(f)(u) - f(u)$$
$$\|(\nabla_w^- f)(u)\|_\infty = f(u) - \min_{v \sim u}\big(f(u), f(v)\big)$$
$$= f(u) - \varepsilon_a(f)(u).$$

With these latter relations, we immediately have the algebraic morphological gradient and Laplace operators

$$\|(\nabla_w^+ f)(u)\|_\infty + \|(\nabla_w^- f)(u)\|_\infty = \delta_a(f)(u) - \varepsilon_a(f)(u)$$

and

$$\|(\nabla_w^+ f)(u)\|_\infty - \|(\nabla_w^- f)(u)\|_\infty = \delta_a(f)(u) + \varepsilon_a(f)(u) - 2f(u).$$

E. Graph Boundary Sets and Relations With Weighted Gradients

Let $\mathcal{A} \subset V$ such that $\forall u \in \mathcal{A}$, there exists a vertex $v \in \mathcal{A}$ with $uv \in E$. We denote by $\partial^+\mathcal{A}$ and $\partial^-\mathcal{A}$, the external and internal boundary sets of $\mathcal{A}$:

$$\partial^+\mathcal{A} = \{u \in V \setminus \mathcal{A} : \exists v \in \mathcal{A} \text{ with } uv \in E\} \quad \text{and}$$
$$\partial^-\mathcal{A} = \{u \in \mathcal{A} : \exists v \in V \setminus \mathcal{A} \text{ with } uv \in E\}. \quad (8)$$

The boundary of V cannot be defined directly from the previous definitions, we assume it is known. Fig. 1 illustrates these notions on two different graphs.

The following properties show relations between graph boundary sets and the weighted gradient norms (6) of the level sets of a function f. The decomposition of f into its level sets is denoted $f^l = \chi(f - l)$ with $\chi : V \to \{0, 1\}$ is the indicator function.

1) Property 1: For any level set f^l, there exists a set $\mathcal{A}^l \subset V$ such that the gradient norms (6) are, $\forall u \in V$

$$\|(\nabla_w^+ f^l)(u)\|_p = \left[\sum_{v \sim u, v \in \mathcal{A}^l} w_{uv}^{p/2} \right]^{1/p} \chi_{(\partial^+\mathcal{A}^l)}(u) \quad \text{and}$$

$$\|(\nabla_w^- f^l)(u)\|_p = \left[\sum_{v \sim u, v \notin \mathcal{A}^l} w_{uv}^{p/2} \right]^{1/p} \chi_{(\partial^-\mathcal{A}^l)}(u). \quad (9)$$

Moreover, $\mathcal{A}^l$ verifies $f^l = \chi_{(\mathcal{A}^l)}$ [34].

Proof: We prove the first equation in (9). One has

$$\|(\nabla_w^+ f^l)(u)\|_p \overset{f^l = \chi_{(\mathcal{A}^l)}}{=} \|(\nabla_w^+ \chi_{(\mathcal{A}^l)})(u)\|_p$$
$$\overset{(6)}{=} \left[\sum_{v \sim u} w_{uv}^{p/2} \max\big(0, \chi_{(\mathcal{A}^l)}(v) - \chi_{(\mathcal{A}^l)}(u)\big)^p \right]^{1/p}.$$

Studying the case where $u \in \mathcal{A}^l$, $u \notin \mathcal{A}^l$ and similarly with its neighbors, we can deduce the first relation in (9): $\|(\nabla_w^+ f)(u)\|_p \neq 0$ only when $u \notin \mathcal{A}^l$ and its neighbors $v \in \mathcal{A}^l$. This vertices' configuration corresponds to the definition of the external set of vertices $\partial^+\mathcal{A}^l$. Second equation in (9) can be obtained with the same scheme. ■

The next property shows the relation between the norms of gradient (4) and gradients (5)

2) Property 2: For any level set f^l, the $\mathcal{L}_p$ norm of the gradient $(\nabla_w f^l)(u)$, at a vertex u, can be decomposed as [34]

$$\|(\nabla_w f^l)(u)\|_p^p = \|(\nabla_w^+ f^l)(u)\|_p^p + \|(\nabla_w^- f^l)(u)\|_p^p.$$

Proof: Using the identities $|a - b| = \max(0, a - b)$ if $a - b > 0$ and $|a - b| = |\min(0, a - b)|$ otherwise, we have

$$\|(\nabla_w f^l)(u)\|_p^p = \sum_{v \sim u} w_{uv}^{p/2}$$
$$\times \big(|\mathcal{D}_v^+ f^l(u)|^p + |\mathcal{D}_v^- f^l(u)|^p\big)$$
$$= \|(\nabla_w^+ f^l)(u)\|_p^p + \|(\nabla_w^- f^l)(u)\|_p^p.$$

■

From this property

$$\|(\nabla_w f^l)(u)\|_p = \begin{cases} \|(\nabla_w^+ f^l)(u)\|_p & \text{if } u \in \partial^+\mathcal{A}^l \\ \|(\nabla_w^- f^l)(u)\|_p & \text{if } u \in \partial^-\mathcal{A}^l. \end{cases}$$

Properties (1) and (2), only consider the $\mathcal{L}_p$ norm with $0 < p < +\infty$. For the $\mathcal{L}_\infty$ norm, same results can be obtained.

III. NONLOCAL PDES-BASED MORPHOLOGICAL FRAMEWORK ON WEIGHTED GRAPHS

In this section, we present our nonlocal MM framework. Our approach uses the previous definitions of weighted gradients that lead to a class of discrete equations that mimic PDEs-based definitions of dilation, erosion, leveling and eikonal equation on weighted graphs.

A. Dilation and Erosion on Weighted Graphs

We define the discrete version of PDEs-based dilation and erosion definitions (1). For a weighted graph $G = (V, E, w)$ and an initial function f^0 defined on V, $\forall u \in V$

$$\text{dilation } \delta : \partial_t f(u, t) = \|(\nabla_w^+ f)(u, t)\|_p \quad \text{and} \quad (10)$$

$$\text{erosion } \varepsilon : \partial_t f(u, t) = -\|(\nabla_w^- f)(u, t)\|_p \quad (11)$$

with the initial condition $f(., 0) = f^0(.)$.

To establish these equations we use the decomposition of f into its level sets f^l. Intuitively with (8), a dilation (resp. erosion) over $\mathcal{A} \subset V$ can be interpreted as a growth (resp. contraction) process that adds (resp. removes) vertices from $\partial^+ \mathcal{A}$ (resp. $\partial^- \mathcal{A}$) to $\mathcal{A}$. As for the continuous case, a simple variational definition of dilation (resp. erosion) applied to f^l can be interpreted as maximization (resp. minimization) of a surface gain proportionally to $\|\nabla_w f^l\|_p$ (resp. $-\|\nabla_w f^l\|_p$).

With the properties defined in Section II-E, dilation of f^l over $\mathcal{A}^l$ corresponds to only consider the set $\partial^+ \mathcal{A}^l$. Hence, dilation can be expressed by $\partial_t f^l(u, t) = \|(\nabla_w^+ f^l)(u, t)\|_p$, $\forall u \in V$, where $\|\nabla_w f^l\|_p$ is reduced to $\|\nabla_w^+ f^l\|_p$. Similarly, erosion can also be defined as $\partial_t f^l(u, t) = -\|(\nabla_w^- f^l)(u, t)\|_p$. Finally, by extending these two processes for all the levels of f, we can naturally consider the two families of dilation (10) and erosion (11) parametrized by p and w over G.

The discrete expression of the internal and external gradients constitute direct numerical schemes. With the usual notation $f^{(n)}(u) \approx f(u, n\Delta t)$, the general iterative scheme for δ and ε can be defined, $\forall u \in V$, at time $n + 1$ as:

$$f^{(n+1)}(u) = f^{(n)}(u) \pm \Delta t \|(\nabla_w^\pm f^{(n)})(u)\|_p. \quad (12)$$

With the $\mathcal{L}_p$ and $\mathcal{L}_\infty$ norms, (12) becomes:

$$f^{(n+1)}(u) = f^{(n)}(u) \\ \pm \Delta t \left[\sum_{v \sim u} w_{uv}^{p/2} |\mathcal{D}_v^\pm f^{(n)}(u)|^p \right]^{1/p} \quad (13)$$

$$f^{(n+1)}(u) = f^{(n)}(u) \\ \pm \Delta t \max_{v \sim u} \left(\sqrt{w_{uv}} |\mathcal{D}_v^\pm f^{(n)}(u)| \right). \quad (14)$$

At each step of these algorithms, the new value at vertex u only depends upon its value at step n and the existing values in its neighborhood.

B. Adaption of Leveling on Weighted Graphs

Nonlinear PDEs that model a general class of morphological filters, the levelings, have been introduced by [10]. Particular cases of such filters are reconstruction openings and closings [41].

Given a reference image $f^0 : \Omega \subset \mathbb{R}^2 \to \mathbb{R}$ and a marker image $m^0 : \Omega \subset \mathbb{R}^2 \to \mathbb{R}$, leveling of f^0 by an infinitesimal disk can be obtained, $\forall x \in \Omega$, by [10]

$$\begin{cases} \partial_t f(x, t) = \text{sign}(f^0(x) - f(x, t)) \|\nabla f(x, t)\|_2 \\ f(., 0) = m^0(.) \end{cases} \quad (15)$$

where $\text{sign} : \mathbb{R} \to \{-1, 0, 1\}$ is the sign function defined as $\text{sign}(a)$ equals 1 if $a > 0$, -1 if $a < 0$ and 0 if $a = 0$.

At convergence, the levelings f of f^0 w.r.t. m^0 is obtained. The leveling corresponds to a sign varying process that controls the behavior of f and can be viewed as a conditional dilation/erosion process.

Let $f^0 : V \to \mathbb{R}$ be a reference function and $m^0 : V \to \mathbb{R}$ be a marker from which a reconstruction can be produced. Using δ and ε, the discrete version of (15) for a weighted graph $G = (V, E, w)$, $\forall u \in V$, is

$$\begin{cases} \partial_t f(u, t) = \max\left(0, \text{sign}(f^0(u) - f(u, t))\right) \|(\nabla_w^+ f)(u, t)\|_p \\ + \min\left(0, \text{sign}(f^0(u) - f(u, t))\right) \|(\nabla_w^- f)(u, t)\|_p \\ f(., 0) = m^0(.) \end{cases} \quad (16)$$

Equation (16) describes a family of leveling parametrized by p and w. When $\text{sign}(f^0 - f) > 0$ (resp. $\text{sign}(f^0 - f) < 0$) then (16) acts as a dilation (resp. an erosion) process. Equation (16) is defined on graphs: the same scheme can be used on any discrete data and our formulation naturally enables nonlocal patch-based configurations for image restoration [35].

C. Adaption of Eikonal Equation on Weighted Graphs

Solutions of the nonlinear eikonal equation have found numerous applications. For instance, in computer vision: median axis or skeleton extraction [42], shape from shading [13] or image segmentation can be mentioned. This latter application uses the stationary version of the level sets formulation [43] that corresponds to solving the eikonal equation. For instance, minimal path extraction for 2D and 3D images uses this method [44]. In computer graphics, the eikonal equation is used to compute geodesics on surfaces with (un-)structured meshes on (non)Cartesian domains [45]–[48].

The isotropic eikonal equation can be expressed as

$$\begin{cases} \|\nabla f(x)\|_2 - P(x) = 0, & \forall x \in \Omega \subset \mathbb{R}^m \\ f(x) = \phi(x), & \forall x \in \Gamma \subset \Omega \end{cases} \quad (17)$$

where ϕ is a positive function defined on Ω and f is the traveling time or distance from source Γ. P is a given potential function (in image segmentation, P can depend upon a gradient computed from initial images). Usual methods used to solve (17) are based upon discretization of the gradient and leads to a nonlinear system. This system is usually solved by an iterative method [13], Fast Sweeping [49] or Fast Marching [43] which is based upon an optimal algorithm [50]. The Fast Marching is actually the most used algorithm. Another approach to solve (17) is to consider the following time dependent system:

$$\begin{cases} \frac{\partial f(x, t)}{\partial t} = P(x) - \|\nabla f(x, t)\|_2, & \forall x \in \Omega \setminus \Gamma \\ f(x, t) = \phi(x), & \forall x \in \Gamma \subset \Omega \\ f(., 0) = \phi_0(.), & \forall x \in \Omega \subset \mathbb{R}^m. \end{cases} \quad (18)$$

At convergence, solution of (18) is the solution of (17).

The eikonal equation describes the evolution of a curve (or a surface) with the level set point of view [12], [43], [51]. With level sets, the curve evolution can be expressed by

$$\partial_t \psi(x,t) = \frac{-1}{P(x)\|\nabla\psi(x,t)\|_2} \quad (19)$$

where ψ corresponds to the level set function. Then, solution $f(x)$ can be interpreted as the minimal time where the zero level set of $\psi(x,t)$ crosses the point x. This evolution can also be interpreted as morphological processing [52]. Indeed, the front propagation driven by P in the normal direction can be computed by dilating or eroding the curve's boundary at time t by an unit disk of size tB. Finally, when P equals 1 or -1 then (1) are recovered.

Equation (18) can be viewed as an erosion process regarding the minus sign and constant P. Then, with (11), the discretization of (18) on a weighted graph $G = (V, E, w)$ is

$$\begin{cases} \partial_t f(u,t) = P(u) - \|(\nabla_w^- f)(u,t)\|_p, & \forall u \in V \setminus V_0 \\ f(u,t) = \phi(u), & \forall u \in V_0 \\ f(u,0) = \phi_0(u), & \forall u \in V \end{cases}$$

(20)

where $V_0 \subseteq V$ is the initial set of seed vertices and $f : V \to \mathbb{R}$. This system is parametrized by p and w [36].

D. Related Schemes

With a particular graph, p and weight function, our framework is related to well-known schemes and can be seen as an extension of the related approaches. This section presents these relations.

1) Algebraic Dilation and Erosion on Graphs: With $\Delta t = 1$ and an unweighted graph, without loss of generality, the scheme (14) based upon $\mathcal{L}_\infty$ norm is the algebraic dilation and erosion on graphs [40], [53].

Dilation δ can be rewritten for $u \in V$ as

$$f^{(n+1)}(u) = f^{(n)}(u) + \max_{v \sim u}\left(0, f^{(n)}(v) - f^{(n)}(u)\right).$$

$f^{(n+1)}(u) = \max_{v \sim u}\left(f^{(n)}(v)\right)$ if $f^{(n)}(v) - f^{(n)}(u) > 0$ and $f^{(n+1)}(u) = f^{(n)}(u)$ otherwise. In both cases, flat algebraic dilation over graph is obtained

$$f^{(n+1)}(u) = \max_{v \sim u}\left(f^{(n)}(v), f^{(n)}(u)\right).$$

With the same scheme, flat algebraic erosion over graph is

$$f^{(n+1)}(u) = \min_{v \sim u}\left(f^{(n)}(v), f^{(n)}(u)\right).$$

For both cases, the structuring element is provided by the vertex neighborhood.

2) Osher-Sethian Scheme in m Dimensional Grid: In previous works, we have presented the relation between our numerical scheme and the Osher-Sethian scheme for a 2-D grid. In the sequel, we present the extension to m dimensional grid and any grid spacing size.

Let $G = (V, E, w)$ be a weighted graph that represents a m dimensional grid. Let u be a vertex associated to a m dimensional vector of spatial coordinates: $u = (i_1 h_1, \ldots, i_m h_m)^T$ where $i_j \in \mathbb{N}$ and h_j is the grid spacing size with $j = 1, \ldots, m$.

	p	w	G
Algebraic morphology on graphs [40], [53]	∞	g_1	any
Osher-Sethian numerical scheme [12], [43]	any	g_1	grid graphs
Our framework	any	any	any

The neighborhood of u can be defined as $\mathcal{N}(u) = \{v : v = u \pm h_j e_j\}_{j=1,\ldots,m}$ where $e_j = (q_k)_{k=1,\ldots,m}^T$ is the vector such that $q_k = 1$ if $j = k$ and $q_k = 0$ otherwise.

Without loss of generality, we consider the case of dilation. Replacing in (13), vertices by its spatial coordinates, with $w_{uv_j} = 1/h_j, \forall v_j \in \mathcal{N}(u)$ and with the notations

$$D_j^+ f(u) = \frac{f(u+h_j e_j) - f(u)}{h_j}$$

$$D_j^- f(u) = \frac{f(u) - f(u-h_j e_j)}{h_j}$$

we have for a vertex u and with $p = 2$

$$f^{(n+1)}(u) = f^{(n)}(u)$$
$$+ \Delta t \sqrt{\sum_{j=1}^m \min\left(0, D_j^- f^{(n)}(u)\right)^2 + \max\left(0, D_j^+ f^{(n)}(u)\right)^2}$$

(21)

since $\min(0, a - b)^2 = \max(0, b - a)^2$. This scheme corresponds to the Osher-Sethian numerical scheme for a grid of m dimension and any grid spacing [12], [43].

3) Shortest Path Algorithm on Graphs: Let G be an unweighted graph representing a m dimensional data set. When $p = \infty$ and $\Delta t = 1$, a Dijkstra like algorithm can be obtained with (20). By neglecting P, (20) can be rewritten, as

$$f^{(n+1)}(u) = f^{(n)}(u)$$
$$- \max_{v \sim u}\left(|\min\left(0, f^{(n)}(v) - f^{(n)}(u)\right)|\right)$$
$$= f^{(n)}(u)$$
$$+ \min_{v \sim u}\left(f^{(n)}(v), f^{(n)}(u)\right), \forall u \in V.$$

This algorithm corresponds to a Dijkstra like shortest path algorithm for any graph. At each step, the distance $f(u)$ at vertex u corresponds to the minimal distance in its neighborhood.

Finally, Table I summarizes the approaches that are related to our framework (g_1 corresponds to the unweighted case).

E. Vector Valued Case

All formulations presented in the latter sections consider scalar function. In the case of a vector valued function $f : V \to \mathbb{R}^m$ with $f(u) = \left(f_i(u)\right)_{i=1,\ldots,m}^T$ and $u \in V$, all the previous processes are performed on each component f_i independently, leading to m processes. Component wise processing can have drawbacks. To overcome this limitation, processes acting on vector-valued function need to take into account the inner correlation between vector-valued functions. In our methodology, the edges weight function acts as a coupling term between these components that enables to overcome this limitation.

IV. ADAPTIVE FRAMEWORK

The last section shows that our morphological framework constitutes a family of operators parametrized by p and w. The graph topology can also be adapted depending upon the function to process. With these points, our framework constitutes a set of adaptive operators. It can be considered as one methodology of adaptive morphology among the approaches proposed in literature such as morphological amoebas [54], [55], PDEs-based viscous morphology [56] or continuous anisotropic morphological operators [9].

In this section, we recall key points on graph construction and weights. For image processing, we also show how nonlocal patch-based information can be incorporated within the graph weights.

A. Graph Construction

Any discrete domain can be modeled by a weighted graph where each data point is represented by a vertex $u \in V$. This domain can represent unorganized or organized data where functions defined on V correspond to the data to process.

1) Unorganized Data: In this general case, an unorganized set of points $V \subset \mathbb{R}^n$ can be seen as a function $f^0 : V \to \mathbb{R}^m$. Then, defining the set of edges consists in modeling the neighborhood of each vertex based upon similarity relationships between feature vectors of the data set. This similarity depends upon a pairwise distance measure $\mu : V \times V \to \mathbb{R}^+$. A typical choice of μ for unorganized data is the Euclidean distance. Graph construction is application dependent and no general rules can be given. However, there exists several methods to construct a neighborhood graph. The interested reader can refer to [57] for a survey on proximity and neighborhood graphs. In this paper, we focus on two classes of graphs: a modified version of k-nearest neighbors graphs and τ-neighborhood graphs.

The τ-neighborhood graph, noted G_τ is a weighted graph where the τ-neighborhood $\mathcal{N}_\tau$ for a vertex u is defined as $\mathcal{N}_\tau(u) = \{v \in V \setminus \{u\} : \mu(u,v) \leq \tau\}$ where $\tau > 0$ is a threshold parameter.

The k-nearest neighbors graph, noted $k - \mathrm{NNG}_\tau$ is a weighted graph where each vertex u is connected to its k nearest neighbors that have the smallest distance measure towards u according to function μ in $\mathcal{N}_\tau$. Since this graph is directed, a modified version is used to make it undirected i.e., $E = \{uv : u \in \mathcal{N}_\tau(v) \text{ or } v \in \mathcal{N}_\tau(u)\}$ for $u, v \in V$. When $\tau = \infty$, one obtains the $k - \mathrm{NNG}_\infty$ where the k nearest neighbors are computed for all the set $V \setminus \{u\}$. For clarity, the $k - \mathrm{NNG}_\infty$ will be noted k-NNG in the rest of this paper.

2) Organized Data: Typical cases of organized data are signals or images (2D or 3D). Such data can be seen as functions $f^0 : V \subset \mathbb{Z}^n \to \mathbb{R}^m$ with $m = 1, 2, \mathrm{or}/, 3$ corresponding to the previously mentioned cases. Then, the distance μ used to construct the graph corresponds to a distance between spatial coordinates associated to vertices. Several distances can be considered. Among those, for a 2D image, we can quote the city-block distance: $\mu(u,v) = |x_u - x_v| + |y_u - y_v|$ or the Chebyshev distance: $\mu(u,v) = \max(|x_u - x_v|, |y_u - y_v|)$ where vertices u and v is associated with their (x,y) spatial coordinates. With these distances and a τ-neighborhood graph, the usual adjacency graphs used in 2D image processing are obtained where each vertex corresponds to an image pixel. Then, a 4-adjacency and an 8-adjacency grid graphs (denoted G_0 and G_1, respectively) can be obtained with the city-block and the Chebyshev distance, respectively (with $\tau \leq 1$). More generally, $((2s+1)^2 - 1)$-adjacency graphs are obtained with a Chebyshev distance with $\tau \leq s$ and $s \geq 1$. This corresponds to add edges between the central pixel and the other pixels within a square window of size $(2s + 1)^2$. Similar remarks apply for the construction of graphs on 3D images where vertices are associated to voxels.

The Region Adjacency Graphs (RAGs) can also be considered for image processing. Each vertex of this graph corresponds to one region. The set of edges is obtained with an adjacency distance: $\mu(u,v) = 1$ if u and v are adjacent and $\mu(u,v) = \infty$ otherwise, together with a τ-neighborhood graph ($\tau = 1$): this corresponds to the Delaunay graph of an image partition.

Finally, we can mention the cases of polygonal curves or surfaces that have natural graph representations where vertices correspond to mesh vertices and edges are mesh edges.

B. Graph Weights

For an initial function $f^0 : V \to \mathbb{R}^m$, similarity relationship between data can be incorporated within edges weights according to a measure of similarity $g : E \to [0,1]$ with $w(uv) = g(uv), \forall uv \in E$. Computing distances between vertices consists in comparing their features generally depending upon f^0. To this end, each vertex u is associated to a feature vector $\mathcal{F}_\tau^{f^0} : V \to \mathbb{R}^{m \times q}$ where q corresponds to this vector size

$$\mathcal{F}_\tau^{f^0}(u) = \left(f^0(v) : v \in \mathcal{N}_\tau(u) \cup \{u\} \right)^T. \tag{22}$$

Then, the following weight functions can be considered. For an edge uv and a distance measure $\rho : \mathbb{R}^{m \times q} \times \mathbb{R}^{m \times q} \to \mathbb{R}^+$ associated to $\mathcal{F}_\tau^{f^0}$, we can have

$$g_1(uv) = 1 \text{ (unweighted case)},$$

$$g_2(uv) = \exp\left(\frac{-\rho\left(\mathcal{F}_\tau^{f^0}(u), \mathcal{F}_\tau^{f^0}(v)\right)^2}{\sigma^2} \right) \text{ with } \sigma > 0$$

$$g_3(uv) = \frac{1}{\left(1 + \rho\left(\mathcal{F}_\tau^{f^0}(u), \mathcal{F}_\tau^{f^0}(v)\right)\right)}. \tag{23}$$

Usually, ρ is the Euclidean distance function. Several choices can be considered for the expression of $\mathcal{F}_\tau^{f^0}$ depending upon the features to preserve. The simplest one is $\mathcal{F}_0^{f^0} = f^0$.

1) Nonlocal Patch-Based Image Processing: In image processing, an important feature vector $\mathcal{F}_\tau^{f^0}$ is provided by image patches. For a gray-scale image $f^0 : V \to \mathbb{R}$, the vector $\mathcal{F}_\tau^{f^0}$ defined by an image patch corresponds to the values of f^0 in a square window of size $(2\tau + 1)^2$ centered at a vertex (a pixel). Fig. 2 shows examples of image patches. A given vertex (central pixel represented in red) is not only characterized by its gray-scale value but also by the gray-scale values contained in a square window (a patch) centered on it. This feature vector has been proposed for texture synthesis [27], and recently used for image restoration and filtering [28], [29]. In image processing,
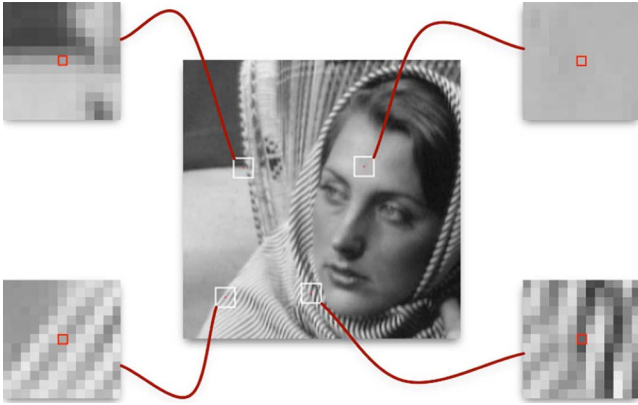
Fig. 2. Four examples of images patches. Central pixels (in red) are associated with vectors of gray-scale values within a patch of size 13×13 ($\mathcal{F}_6^{f^0}$).
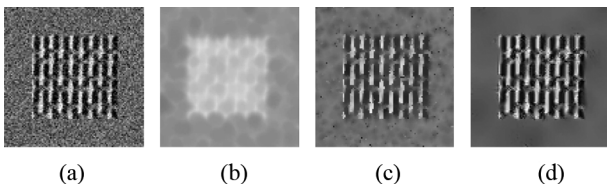

| (a) | (b) | (c) | (d) |

Fig. 3. Adaptivity with weight function. Closing operation. (a) Noisy image. (b) Unweighted. (c) Weighted. (d) Nonlocal patch-based weights.

such configurations are called "nonlocal" and these latter works have shown the efficiency of nonlocal patch-based methods as compared to local ones, in particular by better capturing complex structures such as objects boundaries or fine and repetitive patterns. The notion of "nonlocality" (as defined by [29]) includes two notions: 1) the search window of the most similar neighbors for a given pixel and 2) the feature vector to compare these neighbors. In [29], nonlocal processing consists in comparing, for a given pixel, all the patches contained in an image. In practice, to avoid this high computational cost, one can use a search window of fixed size.

Local and nonlocal methods are naturally included in weighted graphs. Indeed, nonlocal patch-based configurations are simply expressed by the graph topology (the vertex neighborhood) and the edges weights (distances between vertex features). Nonlocal processing of images becomes local processing on similarity graphs.

Our morphological operators on graphs (weighted differences and gradients) naturally enable local and nonlocal configurations (with the weight function and the graph topology) and introduce new morphological tools for image processing.

C. Adaptive Morphological Framework

In this section, we show that our graph-based morphological operators can be considered as adaptive operators. This adaptivity is expressed by the value of p, the weight function w and the graph topology. In the sequel, we show this adaptivity through graph weight and topology.

1) Adaptive Morphology by Graph Weights and Topologies:
Adaptivity with graph weights. We illustrate the adaptivity of our morphological framework with the graph weights, in particular the nonlocal patch-based weights. Fig. 3 illustrates an

adaptive weighted closing ($\varepsilon(\delta)$) on a noisy image [Fig. 3(a)]. In the unweighted case [Fig. 3(b)], a classical closing operation is performed. This processing tends to destroy the main image components. Using a weight (g_2), a better preservation of image structures and the image subregion is obtained [Fig. 3(c)]. With nonlocal patch-based configuration [Fig. 3(d)], the processing acts mainly on similar pixels. In this latter case, a denoising effect is obtained while preserving the complex structure of the subregion. Finally, the weight function can be seen as a local speed function that controls the morphological processing.

Adaptivity with graph topologies. We can adapt morphological processing by changing the graph topology. Indeed, the vertex neighborhood acts as a structuring element. Fig. 4 shows an example of adaptive dilation on a noisy image [Fig. 4(a)]. Results presented are obtained with five iterations and correspond to an algebraic operation. First result [Fig. 4(b)] shows a standard dilation with a graph G_2 that corresponds to a square structuring element of size 5×5 (with Chebyshev distance). In this case, the dilation acts uniformly over all the image and image structures. Second result [Fig. 4(c)] shows an adaptive dilation where the structuring element is locally computed. The final graph is a $25 - \mathrm{NNG}_5$ (each pixel is connected with the 25 nearest neighbors selected in a search window of size 11×11). With locally adapted structuring elements, one can obtain local morphological processing that better preserves image structures.

2) Related Adaptive Morphological Methods: Our morphological framework is related to adaptive methods proposed in literature such as morphological amoebas [54], [55] or PDEs-based viscous morphology [56].

Morphological amoebas. Let $f^0 : V \to \mathbb{R}$ be a scalar initial image. Let $G_0 = (V, E, g_1)$ be an unweighted 4-adjacency grid graph associated to f^0. Let s be a path that connects two vertices u and v such as $u = u_1$ and $v = u_m$ with $u_i u_{i+1} \in E$ and $i = 1, \ldots, m - 1$. Let G_τ^1 be a τ-neighborhood graph (constructed from G_0) used for encoding the structuring element. We can define the distance μ as $\mu(u, v) = \mu_a^q(u, v) = \min_s L_q(s)$ with $q = 1, 2,$ or 3 where

$$L_1(s) = \sum_{i=0}^{m-1} \left(1 + \sigma \| f^0(u_i) - f^0(u_{i+1}) \|_2 \right)$$

with $\sigma > 0$: a parameter that penalizes large variations of f^0. L_1 corresponds to the amoeba distance defined in [54]. Combining vertex (x, y) spatial coordinates and f^0 gives

$$L_2(s) = \sum_{i=0}^{m-1} \sqrt{D_{x,y}^2(u_i, u_{i+1})} + \sigma | f^0(u_i) - f^0(u_{i+1}) |$$

$$L_3(s) = \sum_{i=0}^{m-1} \sqrt{D_{x,y}^2(u_i, u_{i+1}) + \sigma^2 \| f^0(u_i) - f^0(u_{i+1}) \|_2^2}$$

where $D_{x,y}^2(u_i, u_{i+1}) = (x_{u_i} - x_{u_{i+1}})^2 + (y_{u_i} - y_{u_{i+1}})^2$. L_2 and L_3 correspond to two alternative amoeba distances proposed by [55].

G_τ^1 is then obtained with the following vertex neighborhood. For $u \in V$, $\mathcal{N}_\tau(u) = \{ v \in V : \mu_a^q(u, v) \leq \tau \}$. This latter graph corresponds to the graph structure used by morphological amoebas.
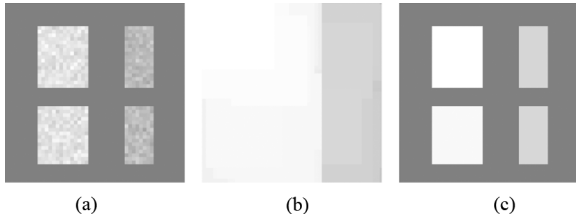
Fig. 4. Adaptivity with graph topology. (a) Noisy image. (b) Dilation with 5×5 square structuring element (G_2). (c) Dilation with a $25 - \text{NNG}_5$.

TABLE II
ADAPTIVE APPROACHES RELATED TO OUR FRAMEWORK

	w	μ
Morphological amoebas [54], [55]	g_1	μ_a^q
PDEs-based viscous morphology [56]	λ	μ_0
Our framework	any	any

Continuous viscous morphology. Our morphological framework is also related with PDEs-based viscous morphology proposed by[56]. Let λ be a function that depends upon an initial gray-scale image f^0 and can be any function of the luminance of f^0 [56]. Since λ only depends upon a given vertex u, then $\sqrt{w_{uv}} = \lambda(f^0, f, u, t) \ \forall uv \in E$. For $0 < p < +\infty$, (10) and (11) become

$$\partial_t f(u, t) = \pm \left[\lambda(f^0, f, u, t)^p \sum_{v \sim u} |\mathcal{D}_v^\pm f(u, t)|^p \right]^{1/p}$$
$$= \pm \lambda(f^0, f, u, t) \|(\nabla_{g_1}^\pm f)(u, t)\|_p.$$

This latter expression corresponds to the PDEs-based viscous morphology defined in [56] where the λ function is $\lambda(f^0, f, u, t) = (f_{\max}^0 - f(u, t))$ or $\lambda(f^0, f, u, t) = (f(u, t) - f_{\min}^0)$. g_1 is the constant weight defined in (23).

Finally, Table II summarizes the special cases of adaptive morphology that are related with our approach where notation μ_0 corresponds to the city-block distance.

V. EXPERIMENTS

In this section, we present various experiments in image and data processing. It is important to note that the following experiments only show the potentialities and the behavior of our methodology. The objective is not to solve a particular problem or application.

A. Image Processing and Segmentation

In this section, experiments for image processing and segmentation with different graph topologies are provided in order to show the behavior of our proposals. We also show the benefits of nonlocal patch-based configurations. We also illustrate the ability of our framework to consider any image graph such as region-based graph.

1) Pixel-Based Graphs: As mentioned in the previous sections, nonlocal patch-based configurations in graph context consists in computing a similarity graph where the vertex neighborhood and the graph weights take into account the nonlocal

patch-based information contained in images. All graphs considered in the following experiments are based upon pixels.

Image processing. The objective of the following experiments is to illustrate the ability of our morphological tools to filter or reconstruct images without destroying main components.

Figs. 5 and 6 present the behavior of our nonlocal patch-based morphological operators for different images and show differences between local and nonlocal patch-based processing. These experiments show dilation δ, closing $\varepsilon(\delta)$ (Fig. 5) and leveling operations (Fig. 6). In both figures, initial images (Fig. 5(a) and first column of Fig. 6) are considered as a function $f^0 : V \to \mathbb{R}$ mapping vertices to gray-scale values.

For local processing, the graphs are weighted 4-adjacency grid graphs G_0 (g_2 for Fig. 5 and g_1 for Fig. 6) with weights computed with the gray-scale value associated to each vertex ($\mathcal{F}_0^{f^0} = f^0$). For nonlocal patch-based processing, the graphs use extended neighborhood with the neighbors selected in a search window and weights are computed with patches. For the woman image (Fig. 5) a weighted 25-adjacency grid graph (G_2, g_2) is considered.

For textured images in Figs. 5 and 6, graphs are based upon 4-adjacency grid graphs (G_0) coupled with $k - \text{NNG}_\tau$ graphs. Resulting graphs are $G_0 \cup k - \text{NNG}_\tau$ where the k nearest neighbors are selected in a search window of size $(2\tau + 1)^2$. For dilation and closing cases (Fig. 5), $k = 10$, $\tau = 10$. For leveling case (Fig. 6), $k = 4$, $\tau = 3$. All weights (in both figures) are computed with patches of size 5×5 that define feature vectors $\mathcal{F}_2^{f^0}$.

In Fig. 5, a local processing (first row) denoises images but tends to destroy small structures. Nonlocal patch-based processing (second row) clearly have superior behavior and allows better denoising while preserving fine and repetitive structures. All results are obtained with same iteration numbers. In Fig. 6, the second column presents the initial marker (m^0) obtained by Gaussian filtering. This experiment shows again the superior behavior of nonlocal patch-based configuration (last column) while local leveling (third column) fail to reconstruct the initial image. Both processing use the same initial marker and all results are obtained at convergence.

These two experiments clearly show the superiority of nonlocal patch-based configurations as compared to local ones. Moreover, k-NN based graphs have the advantage of reducing vertex degrees in nonlocal patch-based configurations leading to reduce the computation time.

Image segmentation. With the same ideas, we can apply nonlocal patch-based configurations for image segmentation with our formulation of the eikonal equation (20).

Image segmentation can be viewed as a semisupervised graph clustering formulated as follows. Let $G = (V, E, w)$ be a weighted graph representing the data to cluster. A semisupervised vertices clustering consists in grouping the set V into c given classes. V is composed of an initial labeled set V_0 and an initial unlabeled one $(V \setminus V_0)$. The objective is to estimate the label of the unlabeled data from labeled ones. To address this problem, methods based upon regularization on graphs have been proposed so far (see [58] and references therein). Here,
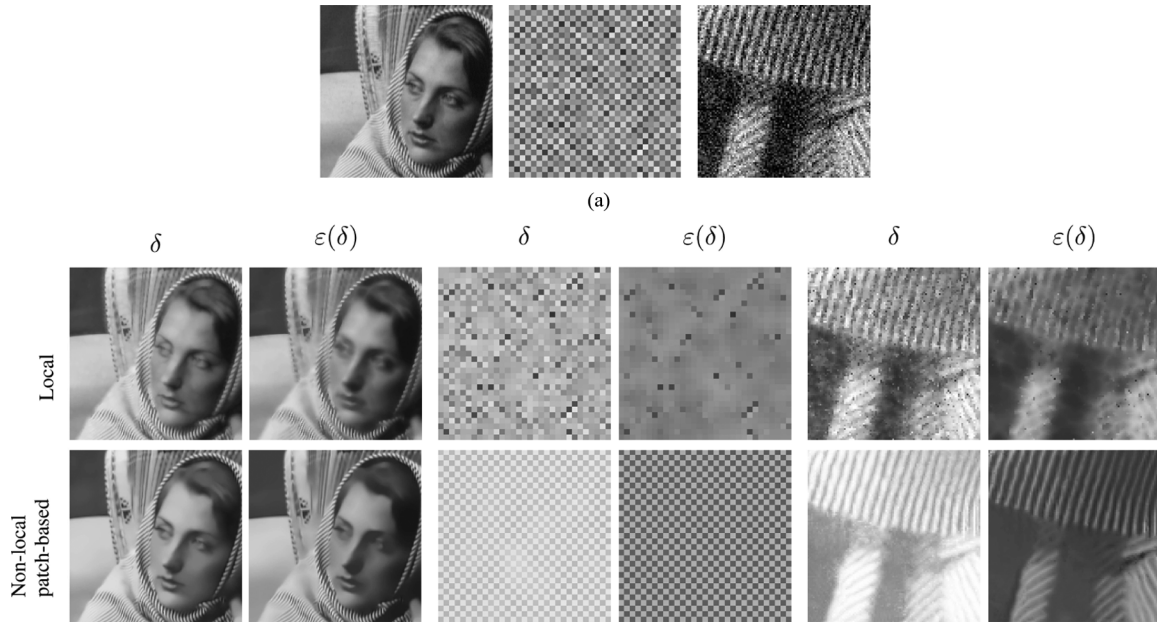
Fig. 5. Local and nonlocal patch-based morphological image processing: dilation δ and closing $\varepsilon(\delta)$. (a) Initial images. Graph structures at first row: G_0, g_2 with $\mathcal{F}_0^{f^0}$. At second row: G_2, g_2 with $\mathcal{F}_2^{f^0}$ for woman image; $G_0 \cup 10 - \mathrm{NNG}_{10}, g_2$ with $\mathcal{F}_3^{f^0}$ for the two textured images.
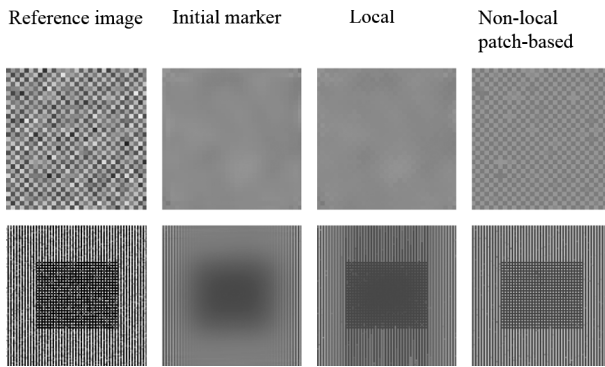

Fig. 6. Local and nonlocal patch-based leveling. Graph structures for local and nonlocal processing: G_0, g_1 with $\mathcal{F}_0^{f^0}$ and $G_0 \cup 4 - \mathrm{NNG}_3, g_2$ with $\mathcal{F}_2^{f^0}$.
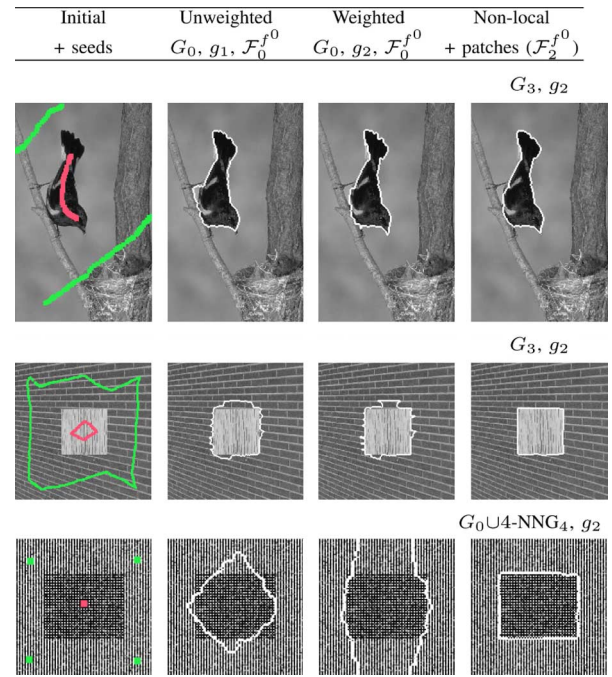

Fig. 7. Segmentation with eikonal equation. First column: initial images and seeds. Next columns: boundaries of segmented regions superimposed on initial images.

we propose to consider the clustering problem with the eikonal equation and to compute c distance functions (f_i) where the set V_0 corresponds to initial seeds. At convergence, the final clustering is obtained for a vertex u as the minimal distance among all the distances i.e., $\arg\min_i\{f_i(u)\}$.

The objective of the following experiments is not to solve a particular image segmentation problem but only to show the behavior of our methodology in order to segment images and the benefits of nonlocal configurations.

Fig. 7 shows semisupervised graph clustering applied to nonlocal patch-based image segmentation. The first column shows the initial images with initial seeds superimposed (red for objects and green for the background). P function involved in (20) corresponds to the initial image gradient amplitude. The three last columns show segmentation results where white boundaries correspond to segmented regions. The second column of Fig. 7 shows results corresponding to unweighted local methods. The third column shows a weighted case. One can note that weights enable to better capture boundaries information. Last column

shows nonlocal patch-based configuration. These results illustrate the benefits of such configurations especially for textured and noisy image where local methods fail to found correctly the desired objects.

2) Region-Based Graphs: Our framework is defined for any graph. We suggest here to use other image structures such as regions instead of pixels. To do this, a partition of initial image is computed (any methods such as watershed or energy partitions
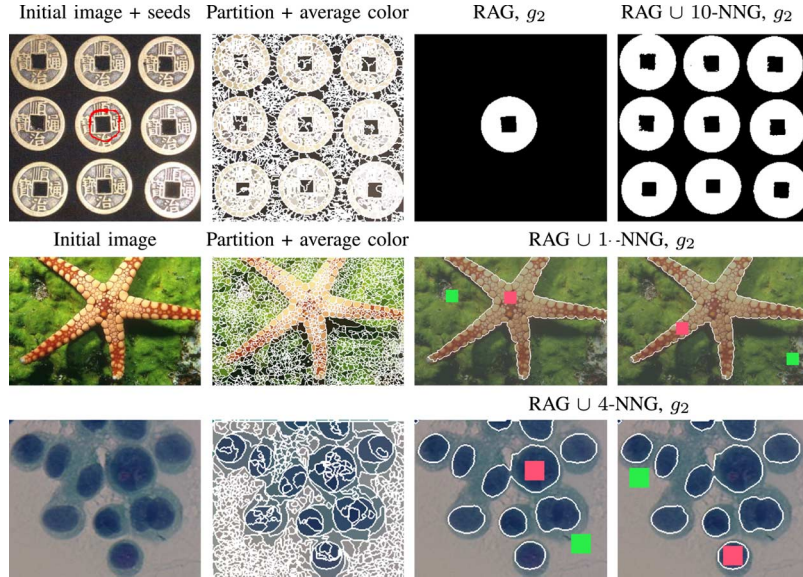
Fig. 8. Segmentation with region-based graphs. Second column: partitions and reconstructed images. First row: local and nonlocally object segmentation with two different graphs. Segmentations are obtained by thresholding distance functions. Last two row: nonlocally object segmentation.
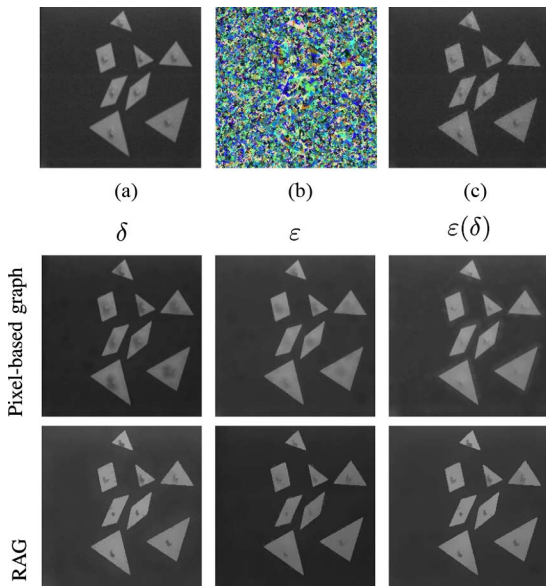

Fig. 9. Processing with region-based graphs. (a) Initial image (65 536 pixels). (b) Partition (11 853 regions i.e., 82% of reduction of vertices). (c) Reconstructed image with average color. First row: weighted (g_2) pixel-based graph computed from (a). Last row: region-based graph (RAG, g_2) computed from (b) and (c).

Fig. 10. Databases processing. (a) Initial databases and associated k-NNGs. Each vertex corresponds to one digit image of size 16×16 $(f^0 : V \to \mathbb{R}^{256})$. Two last rows: morphological processing for each database.

[59] can be used). With this partition, we associate region-based graphs that can be adjacency graph (RAGs) or k-NNGs.

Figs. 8 and 9 show examples of image processing and segmentation with region-based graphs.

Examples of partition and reconstructed image with average color are presented in Fig. 9(b) and (c), and at the second column of Fig. 8. One can note the reduction of graph vertices with region-based graph computed from image partition (82% for Fig. 9 and 98.5% for Fig. 8) as compared to a pixel-based graph computed from initial image.
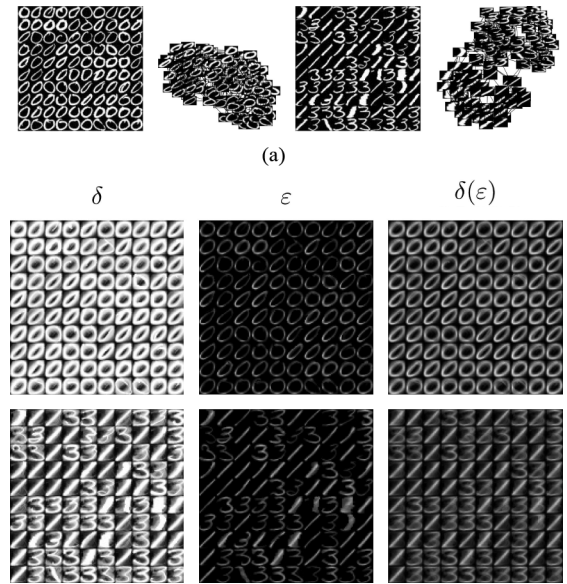
Fig. 9 presents dilation, erosion and closing on pixel and region-based graphs. Results show similar behavior between pixel and region-based graph while, in the latter case, computation time is decreased due to the reduced number of vertices to process. Fig. 8 shows advantages to use such graphs for image segmentation. The second column shows the computed partitions where the region boundaries are superimposed in white and each region is replaced by the average color of corresponding pixels. The first row compares object segmentation with two different region-based graphs: a RAG (third image) and a $\mathrm{RAG} \cup 10 - \mathrm{NNG}$ (last image) where the nearest neighbors are selected among all the regions of the partition. In this

(a)

(b)   (c)

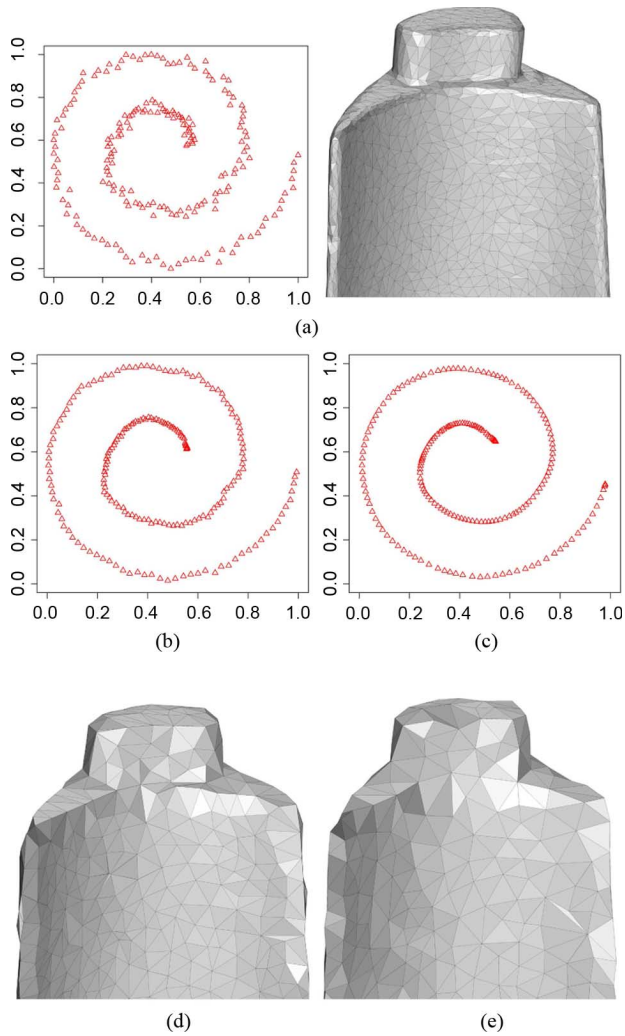(d)   (e)

Fig. 11.   Data sets alternated filtering and evolution with iterations n. (a) Initial data. (b), (c) $n = 2$ and 4. (d) , (e) $n = 10$ and 50.

latter case, by extending the neighborhood, one obtains a nonadjacent object extraction with only one initial seed. The last two rows show two other examples of nonadjacent object segmentation and illustrate the robustness of our method regarding seeds initialization.

Finally, we can summarize the advantages of region-based graphs for image processing and segmentation: 1) Fast computation (as compared to a pixel-based graph) by reducing the number of vertices to process. 2) Reduced initial seed as shown in Fig. 8. 3) Nonadjacent object segmentation. In Fig. 8 only one object is marked and all other ones are extracted even if they are not spatially close.

B. Arbitrary Data Processing

The proposed methodology is formulated on graphs implying that any discrete data can be considered. As a result, our formulation provides a natural extension of PDEs-based method to high dimensional data that can be defined on irregular domains.

1) Database Processing: Fig. 10 presents morphological operations on real world database. Database used here comes from the United States Postal Service (USPS). It consists in
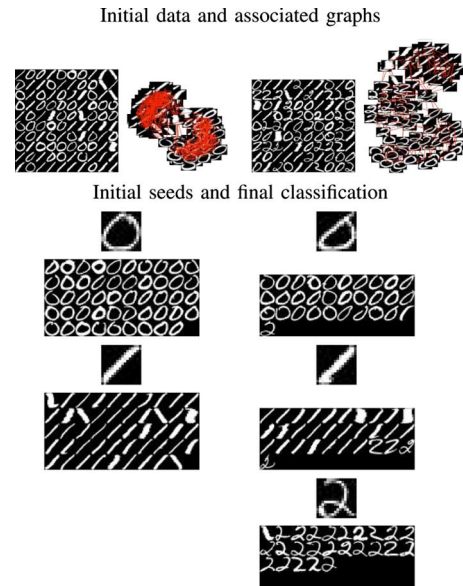

Initial data and associated graphs

Initial seeds and final classification

Fig. 12.   Data clustering with eikonal equation. First row: initial data and associated k-NNGs. Each last column: initial seed (at top) and obtained clustering.

scanned gray-scale images of handwritten digits from "0" to "9." Each image is of size 16×16. In these experiments, we use two randomly subsampled test sets of 100 samples from digits "0" and "1–3." The first row shows the initial data and the associated k-NNG (weighted by g_3). It is important to note that each vertex of the graph corresponds to one digit image sample and is described by a 256-dimensions $(\mathbb{R}^{16 \times 16})$ feature vector where each feature corresponds to a pixel gray-scale value. Data are processed directly without any dimensional reduction step. Second and last rows show dilation, erosion and opening operations, respectively. One can observe the filtering effect of the opening which tends to reduce the initial data to new artificial and uniform samples. Finally, such operations on databases can be used as preprocessing steps for classification or clustering methods and open new insights for MM in domains such as machine learning or data mining.

2) Data Filtering and Simplification: A possible application of our framework is the denoising and the simplification of data sets such as points clouds or meshes. Fig. 11 illustrates a morphological alternated filtering on two data sets that consists in alternating opening and closing operations. Fig. 11(a) shows initial data. In order to filter the noisy spiral, a 10-NNG is constructed where each vertex is associated to the spatial coordinates of the corresponding data point. For the mesh case, we use the natural graph representation of meshes. Both graphs are weighted by function g_2. The two next rows show effects and evolutions w.r.t. iteration n. One can observe the denoising on the noisy spiral and the simplification on the mesh. In the latter case, it is important to note that the number of vertices does not change during the process. Vertices have moved to similar spatial coordinates.

3) Database Clustering: Based upon the idea of image segmentation with the eikonal equation, we apply the same scheme to cluster and classify databases since our framework is formulated on graphs. Fig. 12 shows two examples of classification based on the eikonal equation. First row shows the test sets. First

experiment (a two classes problem) uses two randomly subsampled test sets of 50 samples from digits "0–1." The second one (a three classes problem) uses three randomly subsampled test sets of 30 samples from digits 0"0–1-2." The next columns show the classification results where the initial seeds for each class is located at top. Initial seeds are randomly selected from the test sets. Results presented here, show good classification results (the rates are of 93.1% and 100% for the three and the two classes problems, respectively) even if a simple Euclidean distance on pixels is used in order to compute the graph weights. Weight function used here is g_2. For this particular database, a more relevant distance measure can be used for weighting the graph that takes into consideration the geometric features of the images or that is robust to geometric transformations. Finally, these examples show the potentialities of the eikonal equation to address high dimensional unorganized data classification problem.

VI. CONCLUSION

In this paper, a discrete nonlocal PDEs-based MM operators over weighted graphs is proposed. Our morphological framework adapts and extends PDEs-based dilation, erosion, leveling and the eikonal equation on weighted graphs.

Our methodology enables local and nonlocal patch-based configurations for image processing and segmentation. It also allows the processing and the classification of any nonuniform multivariate high dimensional data by morphological operations that can be useful for machine learning.

Through experiments, we have shown the potentialities and the flexibility of our approach to address image and data processing as well as classification. We have shown the efficiency and the superiority of nonlocal patch-based schemes as compared to local ones for image filtering, reconstruction or segmentation. Advantages of region-based graphs for image processing and segmentation have also been illustrated. Finally, the potentialities of our methodology for database and discrete data filtering, simplification and clustering by morphological tools have been presented.

REFERENCES

[1] G. Matheron, *Random Sets and Integral Geometry*. Hoboken, NJ: Wiley, 1975.
[2] J. Serra, *Image Analysis and Mathematical Morphology*. London, U.K.: London Academic, 1982.
[3] C. Ronse, "Why mathematical morphology needs complete lattices," *Signal Process.*, vol. 21, no. 2, pp. 129–154, 1990.
[4] H. Heijmans, *Morphological Image Operators*. New York: Academic, 1994.
[5] P. Soille, *Morphological Image Analysis, Principles and Applications*, 2nd ed. New York: Springer-Verlag, 2002.
[6] R. van den Boomgaard and A. Smeulders, "The morphological structure of images: The differential equations of morphological scale-space," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 16, no. 11, pp. 1101–1113, Nov. 1994.
[7] L. Alvarez, F. Guichard, P.-L. Lions, and J.-M. Morel, "Axioms and fundamental equations of image processing," *Arch. Ration. Mech. An.*, vol. 123, no. 3, pp. 199–257, 1993.
[8] R. Brockett and P. Maragos, "Evolution equations for continuous-scale morphology," in *Proc. ICASSP*, 1992, vol. 3, pp. 125–128.
[9] M. Breuß, B. Burgeth, and J. Weickert, "Anisotropic continuous-scale morphology," in *Proc. IbPRIA*, 2007, pp. 515–522.
[10] P. Maragos and F. Meyer, "Nonlinear PDEs and numerical algorithms for modeling levelings and reconstruction filters," in *Proc. Scale-Space*, 1999, pp. 363–374.
[11] P. Maragos and M. A. Butt, "Curve evolution, differential morphology and distance transforms applied to multi-scale and eikonal problems," *Fund. Inform.*, vol. 41, pp. 91–129, 2000.
[12] S. Osher and J. Sethian, "Fronts propagating with curvature dependent speed: Algorithms based on Hamilton-Jacobi formulations," *J. Comput. Phys.*, vol. 79, pp. 12–49, 1988.
[13] E. Rouy and A. Tourin, "A viscosity solutions approach to shape-from-shading," *SIAM J. Numer. Anal.*, no. 3, pp. 867–884, 1992.
[14] L. Pizarro, B. Burgeth, M. Breuß, and J. Weickert, "A directional Rouy-Tourin scheme for adaptive matrix-valued morphology," in *Proc. ISMM*, 2009, pp. 102–114.
[15] M. Breuß and J. Weickert, "A shock-capturing algorithm for the differential equations of dilation and erosion," *J. Math. Imag. Vis.*, vol. 25, no. 2, pp. 187–201, 2006.
[16] M. Breuß and J. Weickert, "Highly accurate PDE-based morphology for general structuring elements," in *Proc. SSVM*, 2009, pp. 758–769.
[17] B. Burgeth, M. Breuß, L. Pizarro, and J. Weickert, "PDE-driven adaptive morphology for matrix fields," in *Proc. SSVM*, 2009, pp. 247–258.
[18] J. Goutsias, H. Heijmans, and K. Sivakumar, "Morphological operators for image sequences," *Comput. Vis. Image Und.*, vol. 62, no. 3, pp. 326–346, 1995.
[19] E. Aptoula and S. Lefèvre, "A comparative study on multivariate mathematical morphology," *Pattern Recognit.*, vol. 40, no. 11, pp. 2914–2929, 2007.
[20] F. Y. Shih and S. Cheng, "Adaptive mathematical morphology for edge linking," *Inf. Sci.*, vol. 167, no. 1–4, pp. 9–21, 2004.
[21] R. Verdú-Monedero, J. Angulo, and J. Serra, "Spatially-variant anisotropic morphological filters driven by gradient fields," in *Proc. ISMM*, 2009, pp. 115–125.
[22] O. Tankyevych, H. Talbot, P. Dokládal, and N. Passat, "Direction-adaptive grey-level morphology. Application to 3d vascular brain imaging," in *Proc. ICIP*, 2009, pp. 2261–2264.
[23] M. Charif-Chefchaouni and D. Schonfeld, "Spatially-variant mathematical morphology: Minimal basis representation," in *Proc. ISMM*, 1996, pp. 49–56.
[24] N. Bouaynaya, M. Charif-Chefchaouni, and D. Schonfeld, "Theoretical foundations of spatially-variant mathematical morphology—part I: Binary images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 5, pp. 823–836, May 2008.
[25] N. Bouaynaya and D. Schonfeld, "Theoretical foundations of spatially-variant mathematical morphology—part II: Gray-level images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 5, pp. 837–850, May 2008.
[26] P. Maragos and C. Vachier, "Overview of adaptive morphology: Trends and perspectives," in *Proc. ICIP*, 2009, pp. 2241–2244.
[27] A. A. Efros and T. K. Leung, "Texture synthesis by non-parametric sampling," in *Proc. ICCV*, 1999, vol. 2, pp. 1033–1038.
[28] G. Gilboa and S. Osher, "Non-local linear image regularization and supervised segmentation," *Multiscale Model. Sim.*, vol. 6, no. 2, pp. 595–630, 2007.
[29] A. Buades, B. Coll, and J.-M. Morel, "Non-local image and movie denoising," *Int. J. Comput. Vis.*, vol. 76, no. 2, pp. 123–139, 2008.
[30] A. Elmoataz, O. Lézoray, and S. Bougleux, "Non-local discrete regularization on weighted graphs: A framework for image and manifold processing," *IEEE Trans. Image Process.*, vol. 17, no. 7, pp. 1047–1060, Jul. 2008.
[31] A. Buades, B. Coll, and J.-M. Morel, "Image denoising methods: A review," *Siam Rev.*, vol. 52, pp. 113–147, 2010.
[32] S. Bougleux, A. Elmoataz, and M. Melkemi, "Discrete regularization on weighted graphs for image and mesh filtering," in *Proc. SSVM*, 2007, pp. 128–139.
[33] V.-T. Ta, A. Elmoataz, and O. Lézoray, "Partial difference equations on graphs for mathematical morphology operators overs images and manifolds," in *Proc. ICIP*, 2008, pp. 801–804.
[34] V.-T. Ta, A. Elmoataz, and O. Lézoray, "Partial difference equations over graphs: Morphological processing of arbitrary discrete data," in *Proc. ECCV*, 2008, pp. 668–680.
[35] V.-T. Ta, O. Lézoray, and A. Elmoataz, "Non-local morphological levelings by partial difference equations over weighted graphs," in *Proc. ICPR*, 2008.
[36] V.-T. Ta, A. Elmoataz, and O. Lézoray, "Adaptation of eikonal equation over weighted graphs," in *Proc. SSVM.*, 2009, pp. 187–199.
[37] R. Diestel, "Graph theory," in *Ser. Graduate Texts in Mathematics*, 3rd ed. , Heidelberg: Springer-Verlag, 2005, vol. 173.

[38] A. Bensoussan and J.-L. Menaldi, "Difference equations on weighted graphs," *J. Convex Anal.*, vol. 12, no. 1, pp. 13–44, 2003.

[39] J.-F. Rivest, P. Soille, and S. Beucher, "Morphological gradients," *J. Electron. Imag.*, vol. 2, no. 4, pp. 326–336, 1993.

[40] H. Heijmans, P. Nacken, A. Toet, and L. Vincent, "Graph morphology," *J. Vis. Commun. Image R.*, vol. 3, no. 1, pp. 24–38, 1992.

[41] P. Salembier and J. Serra, "Flat zones filtering, connected operators, and filters by reconstruction," *IEEE Trans. Image Process.*, vol. 4, no. 8, pp. 1153–1160, Aug. 1995.

[42] K. Siddiqi, S. Bouix, A. Tannenbaum, and S. W. Zucker, "The Hamilton-Jacobi skeleton," in *Proc. ICCV*, 1999, pp. 828–834.

[43] J. Sethian, *Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision and Materials Science*. Cambridge, U.K.: Cambridge Univ. Press, 1999.

[44] T. Deschamps and L. Cohen, "Fast extraction of minimal paths in 3d images and application to virtual endoscopy," *Med. Image Anal.*, vol. 5, no. 4, 2001.

[45] A. Bronstein, M. Bronstein, and R. Kimmel, "Weighted distance maps computation on parametric three-dimensional manifolds," *J. Comput. Phys.*, vol. 225, no. 1, 2007.

[46] J. Sethian and A. Vladimirsky, "Ordered upwind methods for static Hamilton-Jacobi equations: Theory and algorithms," *SIAM J. Num. Anal.*, vol. 41, no. 1, pp. 325–363, 2003.

[47] R. Abgrall, "Numerical discretization of the first-order Hamilton-Jacobi equations on triangular meshes," *Commun. Pure Appl. Math.*, vol. 49, pp. 1339–1373, 1996.

[48] C.-W. Shu and Y.-T. Zhang, "High order weno schemes for Hamilton-Jacobi equations on triangular meshes," *SIAM J. Sci. Comput.*, vol. 24, pp. 1005–1030, 2003.

[49] H.-K. Zhao, "Fast sweeping method for eikonal equations," *Math. Comput.*, vol. 74, pp. 603–627, 2005.

[50] J. N. Tsitsiklis, "Efficient algorithms for globally optimal trajectories," *IEEE Trans. Autom. Control*, vol. 40, no. 9, pp. 1528–1538, Sep. 1995.

[51] S. Osher, "A level set formulation for the solution of the Dirichlet problem for Hamilton-Jacobi equations," *SIAM J. Math. Anal.*, vol. 24, no. 5, pp. 1145–1152, 1993.

[52] A. B. Arehart, L. Vincent, and B. B. Kimia, "Mathematical morphology: The Hamilton Jacobi connection," in *Proc. ICCV*, 1993, pp. 215–219.

[53] H. Heijmans and L. Vincent, *Graph Morphology in Image Analysis*, E. Dougherty, Ed. New York: Marcel Dekker, 1992, ch. 6, pp. 171–203.

[54] R. Lerallut, E. Decencière, and F. Meyer, "Image filtering using morphological amoebas," *Image Vis. Comput.*, vol. 25, pp. 395–404, 2007.

[55] M. Welk, M. Breuß, and O. Vogel, "Differential equations for morphological amoebas," in *Proc. ISMM*, 2009, pp. 104–114.

[56] P. Maragos and C. Vachier, "A PDE formulation for viscous morphological operators with extensions to intensity-adaptive operators," in *Proc. ICIP*, 2008, pp. 2200–2202.

[57] J. W. Jaromczyk and G. T. Toussaint, "Relative neighborhood graphs and their relatives," *Proc. IEEE*, vol. 80, no. 9, Sep. 1992.

[58] V.-T. Ta, O. Lézoray, A. Elmoataz, and S. Schüpp, "Graph-based tools for microscopic cellular image segmentation," *Pattern RecognIT.*, vol. 42, no. 6, pp. 1113–1125, 2009.

[59] P. A. Arbeláez and L. Cohen, "Energy partitions and image segmentation," *J. Math. Imag. Vis.*, vol. 20, pp. 43–57, 2004.

Vinh-Thong Ta received the M.Sc. and Doctoral degrees in computer science from the Université de Caen Basse-Normandie, France, in 2004 and 2009, respectively.

From 2009 to 2010, he was an Assistant Professor in computer science with the School of Engineers of Caen (ENSICAEN), France. Since 2010, he has been an Associate Professor with the Computer Science Department, School of Engineers ENSEIRB-MAT-MECA (Institut Polytechnique de Bordeaux). His research mainly concerns image and data processing with discrete variational and mathematical morphology methods for the processing of functions defined on graphs.

Abderrahim Elmoataz is a full-time Professor of computer science in the Computer Science Department, Université de Caen Basse-Normandie, France. His research concerns graph-based and discrete variational methods for image and data processing.

Olivier Lézoray received the M.Sc. and Doctoral degrees in computer science from the Université de Caen Basse-Normandie, France, in 1996 and 2000, respectively.

From 1999 to 2000, he was an Assistant Professor with the Computer Science Department, Université de Caen Basse-Normandie. From 2000 to 2009, he was an Associate Professor Communications, Networks and Services Department, Cherbourg Institute of Technology. Since 2010, he has been a Full Professor at the Cherbourg Institute of Technology. His research concerns color image segmentation and filtering (graph-based variational and morphological methods) and machine learning techniques for image mining (neural networks and support vector machines).