Adaptation of Eikonal Equation over Weighted Graphs

Vinh Thong Ta, Abderrahim Elmoataz, Olivier Lezoray

Adaptation of Eikonal Equation over Weighted Graph

Vinh-Thong Ta, Abderrahim Elmoataz, and Olivier Lézoray

Université de Caen Basse-Normandie, GREYC CNRS UMR 6072, Image Team
{vinhthong.ta,abderrahim.elmoataz-billah,olivier.lezoray}@unicaen.fr
WWW home page: http://www.info.unicaen.fr/~vta

Abstract. In this paper, an adaptation of the eikonal equation is proposed by considering the latter on weighted graphs of arbitrary structure. This novel approach is based on a family of discrete morphological local and nonlocal gradients expressed by partial difference equations (PdEs). Our formulation of the eikonal equation on weighted graphs generalizes local and nonlocal configurations in the context of image processing and extends this equation for the processing of any unorganized high dimensional discrete data that can be represented by a graph. Our approach leads to a unified formulation for image segmentation and high dimensional irregular data processing.

1 Introduction

Solutions of the nonlinear eikonal equation have found numerous applications. One can quote for instance, geometric optics, image analysis or computer vision including shape from shading [1, 2], median axis or skeleton extraction [3], topographic segmentation (watershed) [4] or geodesic distance computation on discrete and parametric surfaces [5–9]. The latter works consider both structured and unstructured meshes on cartesian or non-cartesian domains.

The eikonal equation is a special case of the following general continuous Hamilton-Jabobi equation:

$$\begin{cases} H(x,f,\nabla f)=0 & x\in\Omega\subset\mathbb{R}^n \\ f(x)=\phi(x) & x\in\Gamma\subset\Omega \end{cases}\ , \tag{1}$$

where ϕ in the boundary condition is a positive speed function defined on Ω and $f(x)$ is the traveling time or distance from source Γ. Then, the eikonal equation can be expressed by using the following Hamiltonian:

$$H(x,f,\nabla f)=\|\nabla f(x)\|-P(x)\ , \tag{2}$$

where $P(x)$ is a given potential function. Solution of (1) represents the shortest distance from x to the zero distance curve given by Γ (where $\phi(x)$=0).

Solutions of (2) are usually based on a discretization of the Hamiltonian where the approximation of the derivatives is performed by the Godunov [10] or the Lax-Friedrich [11] schemes. Then, many numerical methods have been proposed and investigated to solve the nonlinear system described by (2). For instance, one can quote the following schemes. (i) An iterative scheme [1] relying on fixed point methods that solves a quadratic equation was proposed. (ii) The fast sweeping methods [12] that use Gauss-Seidel type of iterations to update the distance function field. The key point of fast sweeping is to update the points in a certain order. (iii) Tsitsiklis [13] was the first to develop a Dijkstra like method and proposed an optimal algorithm for solving the eikonal equation. Based on this idea, [14, 11] produced the fast marching methods.

Another approach to solve (2) is to consider a time dependent version of the equation and to evolve it to the steady state. Then, (2) can be rewritten as

$$\begin{cases} \partial f(x,t)/\partial t = -\|\nabla f(x)\| + P(x) & x \in \Omega \subset \mathbb{R}^n \\ f(x,t) = \phi(x) & x \in \Gamma \subset \mathbb{R}^n \\ f(x,0) = \phi_0(x) & x \in \Omega \end{cases} . \tag{3}$$

This paper only considers the discrete analogue of the time dependent formulation of the eikonal equation but, in future works, the stationary case (time independent) will be also considered.

Contributions. In this work, we propose an adaptation of (3) over weighted graphs of the arbitrary structure. The goal here is to provide a simple and common formulation that solves the eikonal equation for any discrete data that can be represented by a weighted graph such as images or high dimensional data defined on irregular domains. This alternative formulation for solving the eikonal equation is based on partial difference equations (PdEs) and discrete gradients over weighted graphs.

Our formulation has several advantages. Any discrete domain that can be described by a graph can be considered without any spatial discretization. In the context of image processing, local and nonlocal configurations are directly enabled within a same formulation. Finally, the aim of this paper is not to solve a particular application with the eikonal equation but to show the potentialities of our proposition to address image segmentation, data clustering or distance computation.

Paper Organization. The paper is organized as follows. Section 2 recalls basics, definitions and operators on weighted graphs. Section 3 introduces our formulation for solving the eikonal equation. Section 4 shows the potentialities of our proposition for the segmentation of images and unorganized data processing. Finally, last Section concludes.

2 Discrete Derivatives on Weighted Graphs

This Section recalls basics, definitions, operators and processes on weighted graphs.

2.1 Definitions and Weighted Graphs Construction

Notations and Definitions. 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 two finites sets: vertices V and weighted edges $E \subseteq V \times V$. An edge $(u,v) \in E$ connects two adjacent (neighbor) vertices u and v . The neighborhood of a vertex u is noted $N(u)=\{v \in V \setminus \{u\} : (u,v) \in E\}$. The weight ω_{uv} of an edge (u,v) can be defined with a function $w: V \times V \rightarrow \mathbb{R}^+$ such that $w(u,v)=\omega_{uv}$ if $(u,v) \in E$ and $w(u,v)=0$ otherwise. Graphs are assumed to be simple, connected and undirected implying that function w is symmetric. Let $f: V \rightarrow \mathbb{R}$ be a discrete real-valued function that assigns a real value $f(u)$ to each vertex $u \in V$. We denote by $\mathcal{H}(V)$ the Hilbert space of such functions defined on V.

Weighted Graphs Construction. Any discrete domain can be represented by a weighted graph where functions of $\mathcal{H}(V)$ represents the data to process. In the general case, an unorganized set of points $V \subset \mathbb{R}^n$ can be seen as a function $f^0: V \subset \mathbb{R}^n \rightarrow \mathbb{R}^n$. Then, constructing a graph from this data consists in defining the set of edges E by modeling the neighborhood. It is based on a similarity relationship between data with a pairwise distance measure $\mu: V \times V \rightarrow \mathbb{R}^+$. There exists several methods to transform a set of vertices V into a neighborhood (similarity) graph (see [15] for a survey on proximity and neighborhood graphs). In this paper, we focus on two particular graphs: the τ-neighborhood graphs and a modified version of k-nearest neighbors graphs. The k nearest neighbors graph, noted k-NNG is a weighted graph where each vertex $u \in V$ is connected to its k nearest neighbors which have the smallest distance measure towards u according to function μ. Since this graph is directed, a modified version of this graph is used to make it undirected. The τ-neighborhood graph, noted G_τ is a weighted graph where the τ-neighborhood N_τ for a given vertex $u \in V$ is defined as $N_\tau(u)=\{v \in V \setminus \{u\} : \mu(u,v) \leq \tau\}$ with $\tau>0$ a threshold parameter.

2D images can be viewed as functions $f^0: V \subset \mathbb{Z}^2 \rightarrow \mathbb{R}^n$. In this case, the associated distance μ for construct the neighborhood graph is usually the city block or the Chebychev distances computed with the spatial coordinates of each vertex representing an image pixel. With these distances and the τ-neighborhood graphs, one recovers the two usual graphs used in image processing, the 4-adjacency grid graph (denoted G_0 with the city block distance) and the 8-adjacency grid graph (denoted G_1 with the Chebychev distance) with $\tau \leq 1$. Another useful graph structure in image processing is the region adjacency graph (RAG) where vertices correspond to image regions, and the set of edges is obtained by considering an adjacency distance. With the τ-neighborhood ($\tau=1$), the RAG is the Delaunay graph of an image partition.

Weights Computation. Similarities between data can be incorporated within edges' weights according to a measure of similarity $g: E \rightarrow \mathbb{R}^+$ with $w(u,v)=g(u,v)$ for $(u,v) \in E$. Then, the distance computation between data is performed by comparing their features that generally depend on a given initial function $f^0 \in \mathcal{H}(V)$. To this aim, each vertex $u \in V$ is assigned with a feature vector $F(f^0,u) \in \mathbb{R}^m$.

With F, the following weight functions can be considered. For a given edge $(u,v)\in E$ and a distance measure $\rho:V\times V\to\mathbb{R}^+$ associated to F, we can have

$$g_0(u,v) = 1 \text{ (constant weight case)} \quad ,$$
$$g_1(u,v) = (\rho(F(f^0,u),F(f^0,v)) + \epsilon)^{-1} \text{ with } \epsilon>0, \epsilon\to 0 \quad ,$$
$$g_2(u,v) = \exp(-\rho(F(f^0,u),F(f^0,v))^2/\sigma^2) \text{ with } \sigma>0 \quad ,$$

where σ controls the similarity and ρ is usually the euclidean distance.

Several choices for the expression of F can be considered depending on the features to preserve. The simplest one is $F(f^0,.)=f^0$. In the context of image processing, an important feature vector F is provided by images patches i.e. $F(f^0,u)=F_\tau(f^0,u)=\{f^0(v) : v\in N_\tau(u)\cup\{u\}\}$. In the case of a grayscale image $F_\tau(f^0,.)$ is a vector of size $(2\tau+1)^2$ corresponding to the values of f^0 in a square window of size $(2\tau+1)\times(2\tau+1)$ centered at vertex u (a pixel). Color images can be handled using features of dimension $3\times(2\tau+1)^2$. Then, the resultant weight function directly incorporates local or nonlocal features [16]. This feature vector has been proposed in the context of texture synthesis [17], and further used in the context of image processing [18, 19].

2.2 Graph Based Discrete Gradients

Let $G=(V,E,w)$ be a weighted graph. The discrete weighted gradient of a function $f\in\mathcal{H}(V)$ at a vertex $u\in V$ is defined by

$$(\nabla_w f)(u) = (\partial_v f(u))_{(u,v)\in E}$$

where $\partial_v f(u)=\sqrt{w(u,v)}(f(v)-f(u))$ corresponds to the discrete (partial) derivative of f with respect to the edge (u,v). These definitions have been used by [20] for image and mesh regularization. Based on the latter works, two discrete formulations of weighted morphological gradients on graphs have been proposed by [21]: namely, the weighted external ∇_w^+ and the internal ∇_w^- gradient operators. For $u\in V$

$$(\nabla_w^+ f)(u) = (\partial_v^+ f(u))_{(u,v)\in E} \quad \text{and} \quad (\nabla_w^- f)(u) = (\partial_v^- f(u))_{(u,v)\in E} \quad , \tag{4}$$

where the external $\partial_v^+ f(u)$ and the internal $\partial_v^- f(u)$ discrete partial derivatives are

$$\partial_v^+ f(u) = \max(0, \partial_v f(u)) \quad \text{and} \quad \partial_v^- f(u) = -\min(0, \partial_v f(u)) \quad ,$$

with $\partial_v^- f(u)=\partial_u^+ f(v)$. When the weight is constant ($w=g_0$) these definitions recover the classical directional derivative operators.

The $\mathcal{L}_p$-norm (with $0<p<+\infty$) and the $\mathcal{L}_\infty$-norm of gradients (4) are

$$\|(\nabla_w^\pm f)(u)\|_p = \Big[\sum_{v\sim u} w(u,v)^{p/2}|(f(v)-f(u))^\pm|^p\Big]^{1/p} \text{ and} \tag{5}$$

$$\|(\nabla_w^\pm f)(u)\|_\infty = \max_{v\sim u}\big(w(u,v)^{1/2}|(f(v)-f(u))^\pm|\big) \quad . \tag{6}$$

Notation $v{\sim}u$ means that vertex v is adjacent to u. $\nabla_w^{\pm}$ refers to both external and internal gradient (with respect to the sign) and $(a)^{+}{=}\max(0,a)$ and $(a)^{-}{=}\min(0,a)$. These gradients have the following property: $\|(\nabla_w f)(u)\|_p^p = \|(\nabla_w^+ f)(u)\|_p^p + \|(\nabla_w^- f)(u)\|_p^p$ with $0{<}p{<}+\infty$. Moreover, with a constant weight function and $p{=}\infty$, (6) recovers the usual expression of algebraic morphological external and internal gradients.

Associated Dilation and Erosion Processes. Continuous morphology (see [22] and references therein) defines flat dilation $\delta : \mathbb{R}^n \rightarrow \mathbb{R}^n$ and erosion $\varepsilon : \mathbb{R}^n \rightarrow \mathbb{R}^n$ of a given function $f{:}\mathbb{R}^n{\rightarrow}\mathbb{R}$ by a structuring set $B{=}\{x{\in}\mathbb{R}^n : \|x\|_p{\leq}1\}$ with the following general partial differential equations (PDEs):

$$\frac{\partial \delta(f)}{\partial t} = \partial_t f = +\|\nabla f\|_p \quad \text{and} \quad \frac{\partial \varepsilon(f)}{\partial t} = \partial_t f = -\|\nabla f\|_p \ . \tag{7}$$

Ta et al. [21] has proposed a discrete version over graphs of these equations by using morphological gradients (4). Given a graph $G{=}(V,E,w)$, a function $f{\in}\mathcal{H}(V)$ and for all $u{\in}V$, discrete analogue of (7) are

$$\frac{\partial \delta(f(u))}{\partial t}{=}\partial_t f(u){=}{+}\|(\nabla_w^+ f)(u)\|_p \text{ and } \frac{\partial \varepsilon(f(u))}{\partial t}{=}\partial_t f(u){=}{-}\|(\nabla_w^- f)(u)\|_p \ . \tag{8}$$

Equations (8) correspond to dilation and erosion over weighted graphs. They constitute a morphological framework [21] based on PdEs that extends algebraic and continuous morphological operators for images and high dimensional data processing.

3 Eikonal Equation on Weighted Graphs

In this Section, we present our formulation to approximate the eikonal equation (3) over weighted graphs by considering PdEs and the morphological gradients presented in the previous Section.

With morphological processes described by (8), the time dependent eikonal formulation (3) can be viewed as an erosion process regarding the minus sign and a null potential function P. With the corresponding internal gradient (∇_w^-) involved in discrete PdEs based erosion process, (3) can be directly rewritten with weighted graphs. Given a graph $G{=}(V,E,w)$ and a function $f{\in}\mathcal{H}(V)$, we obtain a discrete PdEs based version of the system (3)

$$\begin{cases} \partial f(u,t)/\partial t = -\|\nabla_w^- f(u)\|_p + P(u) & u{\in}V \\ f(u,t){=}\phi(u) & u{\in}V_0 \subset V \\ f(u,0){=}\phi_0(u) & u{\in}V \end{cases} ,$$

where V_0 corresponds to the initial seed vertices.

With $f^n(u) \approx f(u, n\Delta t)$, this iterative numerical scheme is obtained for all $u{\in}V$:

$$f^{n+1}(u){=}f^n(u) - \Delta t\big(\|(\nabla_w^- f^n)(u)\|_p - P(u)\big) \ . \tag{9}$$

The steady state (i.e. given a fixed number n of iteration or when $\|f^{n+1}-f^n\| < \epsilon$) of this process is the solution of the eikonal equation (2).

Injecting the corresponding internal gradient norm in (9), we obtain for the $\mathcal{L}_p$-norm (5) and the $\mathcal{L}_\infty$-norm (6)

$$f^{n+1}(u)=f^n(u) - \Delta t\Big(\Big[\sum_{v\sim u} w(u,v)^{p/2}|\min(0,f(v)-f(u))|^p\Big]^{1/p} - P(u)\Big) \ , \quad (10)$$

$$f^{n+1}(u)=f^n(u) - \Delta t\Big(\max_{v\sim u}\big(w(u,v)^{1/2}|\min(0,f(v)-f(u))|\big) - P(u)\Big) \ . \quad (11)$$

The proposed methodology leads to a simple and common formulation that constitutes an adaptative framework for the eikonal equation. Indeed, our approach only depends on the p value and the weight function w. In Sect. 4, experiments show how the framework can be adapted to address image segmentation or data clustering.

Relations with other schemes. Scheme (9) has the advantage to work on any graph structures. Then, with an adapted graph topology and an appropriated weight function, the proposed formulation is linked to well-known schemes such as Osher-Sethian Hamiltonian discretization scheme or the graph based Dikjstra algorithm.

Osher-Sethian scheme. Let $G_0=(V,E,g_0)$ be an unweighted 4-adjacency grid graph associated with an image. Then, (10) recovers the exact Osher-Sethian upwind first order Hamiltonian discretization scheme [14] when $p=2$ and using G_0:

$$f^{n+1}(u)=f^n(u) - \Delta t\Big(\Big[\sum_{v\sim u} |\min(0,f(v)-f(u))|^2\Big]^{1/2} - P(u)\Big) \ .$$

Replacing vertices $u\in V$ and their neighborhood by their spatial coordinates (x,y), the latter expression can be rewritten as

$$\begin{aligned} f^{n+1}((x,y))=f^n((x,y))-\Delta t\Big[&\Big(\big|\min\big(0,f^n((x,y))-f^n((x-1,y))\big)\big|^2 \\ &+\big|\max\big(0,f^n((x+1,y))-f^n((x,y))\big)\big|^2 \\ &+\big|\min\big(0,f^n((x,y))-f^n((x,y-1))\big)\big|^2 \\ &+\big|\max\big(0,f^n((x,y+1))-f^n((x,y))\big)\big|^2\Big)^{1/2} - P((x,y))\Big] \ , \end{aligned}$$

since $\min(0,a-b)^2=\max(0,b-a)^2$. This equation corresponds to the discretization scheme of the Hamilton-Jacobi equations proposed by [14].

Dikjstra scheme. Let $G=(V,E,g_0)$ be an unweighted graph. Then, (11) corresponds to an iterative version of the Dikjstra shortest path algorithm defined on graphs of arbitrary structure. Indeed, in the case where $p=\infty$, $\Delta t=1$ and with G, (11) becomes, for all $u\in V$

$$f^{n+1}(u) = f^n(u) - \max_{v\sim u}\big(|\min(0,f(v)-f(u))|\big) + P(u) = \min_{v\sim u}(f^n(v)) + P(u) \ ,$$

by considering the neighborhood of u as $N(u) \cup \{u\}$ and with the properties that $\max(0, a-b) = -\min(0, b-a)$ and $\min(0, a-b) = \min(a, b) - b$. This equation corresponds to a shortest path algorithm for a given graph where at each step, the distance $f(u)$ at vertex u corresponds to the minimal distance in its neighborhood.

4 Experiments

The proposed formulation of the eikonal equation and can be used to process any function defined on vertices of a graph or on any arbitrary discrete domain. This Section illustrates the potentialities of our formulation through examples of weighted distance computation, image segmentation and unorganized high dimensional data processing. Different graph structures and weight functions are also used to show the flexibility of our approach. In the sequel, all experiments are obtained with a constant potential function $P=1$. Clearly, a different potential function can be adapted for a particular application. The objective of the following experiments is not to solve a particular application. They only illustrate the potential and the behavior of our eikonal equation formulation.

Adaptative Front Propagation and Weighted Distances. Figure 1 shows the adaptivity of our formulation in order to compute weighted distances. Indeed, this example shows results for different p values, graph topologies, weight functions and features F. The initial seed is located at the top left corner of the original grayscale image $f^0 : \mathcal{H}(V) \rightarrow \mathbb{R}$. First, second and third rows of Fig. 1 show results for $p=2$, 1 and ∞ respectively, where (10) and (11) are used. All the results correspond to color distance maps (red for small and blue for large distances) where iso-levels sets are superimposed in white. First and second columns of Fig. 1 show results obtained with unweighted ($w=g_0$) graphs. First column uses a 4-adjacency grid graph (G_0) and corresponds to the classical case. Second column uses a 25-adjacency grid graph (G_2) and shows the effect of a larger neighborhood. Third and fourth columns show results obtained with weighted graphs. Third column considers graph G_0 weighted by function g_2 with $F=f^0$. By using non constant weights, image information is automatically integrated in the distance computation that modifies the front evolution speed particularly into the textured sub-image. Fourth column shows the nonlocal case where graph G_2 is constructed and weighted with function g_2 associated with patches of size 11×11. In that case, repetitive information are clearly captured by the weights that stops the front propagation around the textured sub-image. Finally, segmentation of the textured sub-image can be simply obtained by thresholding the computed distances.

Image Segmentation with Region Based Graphs. The goal of the following two examples is not to show a perfect segmentation but to show how we can take advantage of graph topologies in image segmentation. The basic idea is to consider that image pixels are not the only relevant components in image

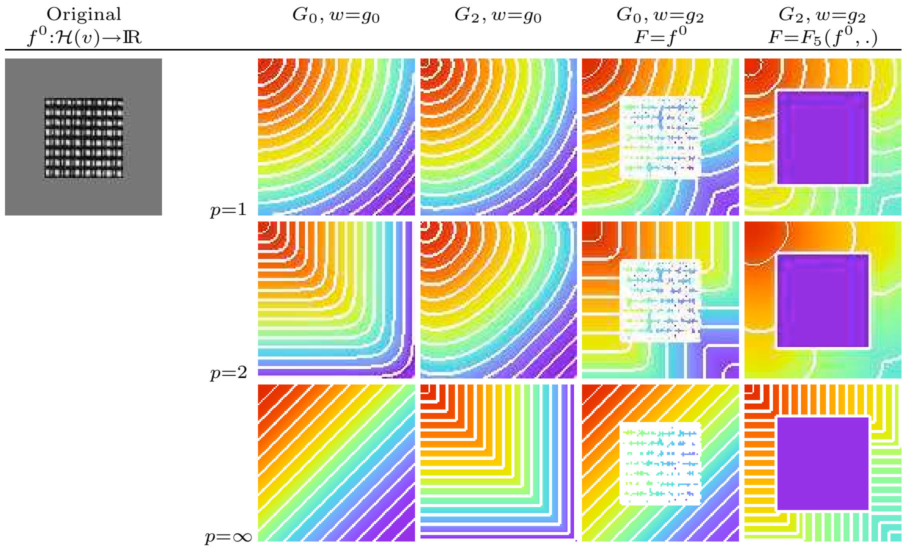
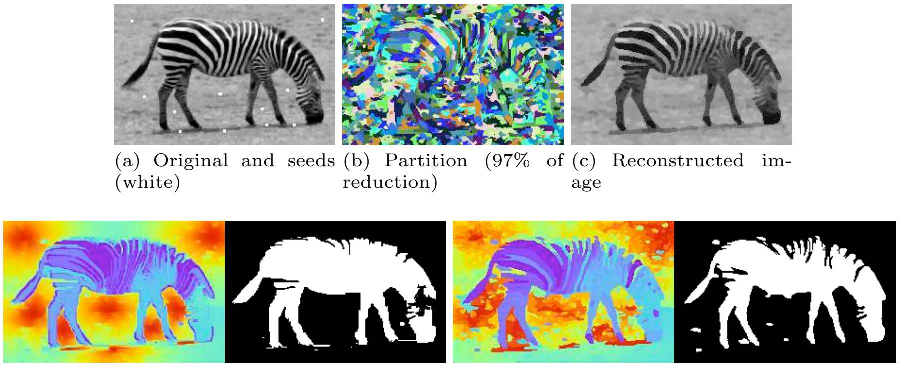


Fig. 1. Front propagation and weighted distances with different p values, graph configurations G, weights w and features F. Figures represent color distance maps with iso-level sets obtained by thresholding the distances. The seed is located at the top left corner (see text for more details).

and more abstract elements such as image regions can be used. Hence, we suggest to work directly with a reduced version of images: image partitions. Image partitions can be obtained by image pre-processing methods such as watershed. Figures 2(b) and 3(b) show such partitions computed from Figs. 2(a) and 3(a). Figures 2(c) and 3(c) are reconstructed images from partitions with the mean color value for each region.

Figure 2 presents an example of image segmentation based on RAG and also shows that this graph structure can accelerate segmentation processes. This example compares segmentation obtained by a 4-adjacency grid graph G_0 weighted by function g_2 with pixel grayscale values (Fig. 2(d)) and segmentation result with a RAG constructed from partition (Fig. 2(b)) and weighted by function g_2 with mean values (Fig. 2(e)). Color distance maps are obtained with the initial seeds (white points) in Fig. 2(a). Segmentations are performed by thresholding the obtained distances. Results show similar behaviors both on distance maps and segmentations while drastically speeding-up the segmentation process in the RAG case. Indeed, the number of vertices in the RAG represents approximatively 3% as compared to the number of vertices in the pixel based graph. The direct consequence is a decreasing of the computational complexity thanks to the reduced amount of data to consider. On a standard computer the computing time can be decreased by a 10 factor.

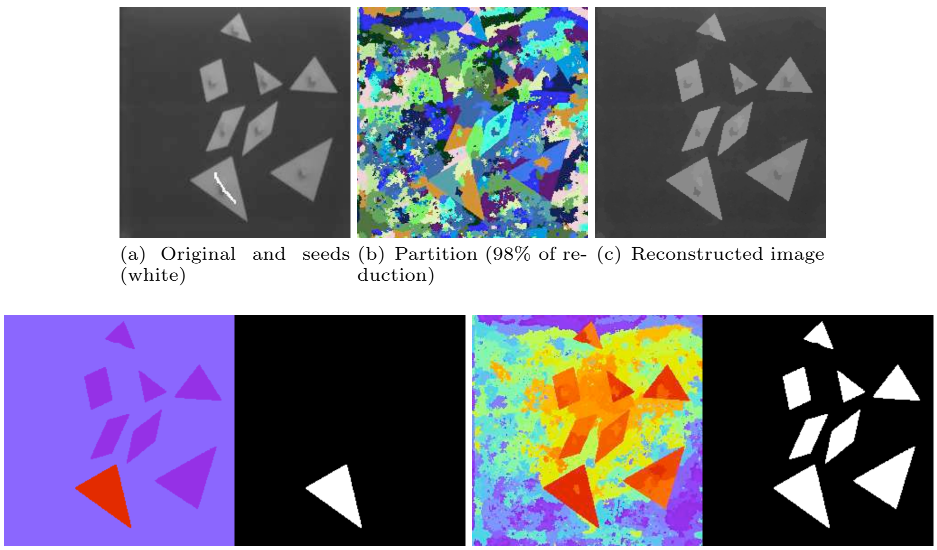
Figure 3 shows another benefit of using a RAG structure: nonlocal (non spatially connected) object segmentation. This experiment compares segmentation results with RAG (Fig. 3(d)) and nonlocal RAG (Fig. 3(e)). Both graphs are computed from partition 3(b) weighted by function g_2 with mean color values.



(a) Original and seeds (white) (b) Partition (97% of reduction) (c) Reconstructed image

(d) Grid graph G_0, $w{=}g_2$ (e) RAG, $w{=}g_2$

Fig. 2. Acceleration of image segmentation process. (a) original image (150×235) with 35 250 pixels. (b) partition with 999 regions (97% of reduction in terms of image components). (c) reconstructed image with mean color value. (d) and (e): at left, distance color maps (red for small and blue for large distances) and at right, final segmentations. Images (d) are obtained with a pixel based graph computed from (a). Images (e) are obtained with a RAG constructed with (b) and (c) (see text for more details).



(a) Original and seeds (white) (b) Partition (98% of reduction) (c) Reconstructed image

(d) RAG, $w{=}g_2$ (e) RAG∪5-KNNG,$w{=}g_2$

Fig. 3. Nonlocal region based image segmentation. (a) original image (256×256) with 65 536 pixels. (b) partition with 1 324 regions (98% of reduction as compared to original one). (d) and (e) at left, distance color maps (red for small and blue for large distances) and at right, final segmentations. Graphs used in (d) and (e) are computed from (b) and (c) (see text for more details).

In nonlocal RAG case, each vertex neighborhood is extended by a 5 nearest neighborhood based on mean value feature. The obtained graph is a RAG∪5-NNG graph. Figures 3(d) and 3(e) show color distance maps computed from initial seeds (white stroke) in Fig. 3(a) and final segmentations. For local case

(Fig. 3(d)), object marked by seeds is well segmented with respect to close distances (red color). The other objects are far (blue color) and the final segmentation only extracts the marked one. For nonlocal case (Fig. 3(e)), the distance within the marked object is close to the initial seeds. In addition the distances to other triangles in the scene are also computed as close to seeds (red color). The consequence is that all the objects in the image are extracted by thresholding even if they are not spatially close with a minimal number of initial seeds.

Unorganized High Dimensional Data Processing The following experiments show applications of our formulation of the eikonal equation for the processing of high dimensional data in irregular domains. Figure 4 shows applications of the eikonal equation for data clustering and shortest path problems. The initial data set (Fig. 4(a)) is constituted of 133 images of head pose. Each image is of size 29×29. From this data set, two possible applications can be performed: clustering and head pose transition estimation. The goal here is not to solve machine leaning problems, but to show that these problems can be addressed by our formulation of eikonal equation. In order to process such data, a graph ($|V|$=133) is constructed where each vertex represents an image and is described by a feature of size 29×29 (i.e $\mathbb{R}^{841}$). In the following results, initial seeds (images) are represented with white boundaries. Points that are close and far to seeds are respectively represented with blue and red colors in distance maps (Fig. 4(b) and 4(c)).

Figures 4(b) and 4(d) show the application of the eikonal equation for data clustering. Such an application can be used for data set exploration or semi-supervised learning: given an input seed (query) one wants to obtained the closest points with respect to the initial input. Figure 4(b) shows the distance map obtained from a single initial seed. Figure 4(d) shows clustering results. Initial input has a white boundary. The 10 closest images are located at the top and the 10 farthest are located at the bottom of Fig. 4(d).

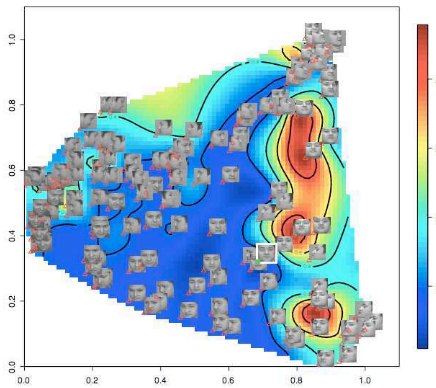
Figures 4(c) and 4(e) shows another example of application of the eikonal equation for data set. Given two initial images, one wants to recover a transition sequence of images that separates them. This problem can be viewed as a shortest path problem solved by the eikonal equation. Figure 4(c) shows the distance map obtained from the initial seeds. Figure 4(e) shows the obtained path from seed at top left to seed at bottom right.

These experiments show satisfying results and the ability of our approach to address machine learning problems even if a simple euclidean distance is used to compare data points. Clearly, results can be improved by using well adapted distances or features estimation.

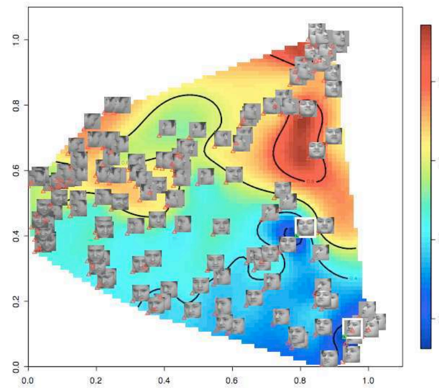
5 Conclusion

In this paper, a discrete version of the eikonal equation over weighted graphs of arbitrary structure is proposed. Solution of the eikonal equation based on PdEs, discrete gradients and weighted graphs is presented. The proposed formulation constitutes a simple, common and adaptative framework that recovers

(a) Original data, 133 images of size 29×29 ($|V|$=133, $f^0 : V \to \mathbb{R}^{841}$)



(b) Color distance map+seed (white boundaries)



(c) Color distance map+seeds (white boundaries)

(d) Local clustering with (b)

(e) Shortest path with (c)

Fig. 4. High dimensional data clustering and shortest path. (b) and (c) color distance maps (blue for small and red for large distance) images superimposed with white boundaries are initial seeds. (d) clustering results where at top the 10 closest and at bottom the 10 farthest with respect to the seed (white boundary). (e) shortest path from the two initial seeds (white boundary).

well-known definitions and unifies local and nonlocal configurations in the context of image processing. This framework can consider any discrete data that can be represented by weighted graphs. Through experiments, we have shown the potentiality and the flexibility of our approach to address image segmentation and unorganized high dimensional data processing. Finally, an ongoing work is to address the stationary (time independent) version of the eikonal equation and to solve this equation by considering fast marching like methods on arbitrary graphs within our framework.

References

1. Rouy, E., Tourin, A.: A viscosity solutions approach to shape-from-shading. SIAM J. Num. Anal. **29** (1992) 867–884
2. Sethian, J.: A fast marching level set methods for monotonically advancing fronts. Proc. Nat. Acad. Sci. **41**(2) (1999) 199–235
3. Siddiqi, K., Bouix, S., Tannenbaum, A., Zucker, S.W.: The hamilton-jacobi skeleton. In: Proc. ICCV. (1999) 828–834
4. Maragos, P., Butt, M.: Curve evolution, differential morphology and distance transforms as applied to multiscale and eikonal problems. Fundamentae Informatica **41** (2000) 91–129
5. Bronstein, A.M., Bronstein, M.M., Kimmel, R.: Weighted distance maps computation on parametric three-dimensional manifolds. J. Comput. Phys. **225**(1) (2007) 771–784
6. Sethian, J.A., Vladimirsky, A.: Ordered upwind methods for static hamiton-jacabi equations: Theory and algorithms. SIAM J. Num. Anal. **41**(1) (2003) 325–363
7. Abgrall, R.: Numerical discretization of the first-order hamilton-jacobi equations on triangular meshes. Comm. Pure and Applied Math. **49** (1996) 1339–1373
8. Chi-Wang Shu and Yong-Tao Zhang: High order WENO schemes for hamilton-jacobi equations on triangular meshes. SIAM J. Scien. Comp. **24** (2003) 1005–1030
9. Mémoli, F., Sapiro, G.: Fast computation of weighted distance functions and geodesics on implicit hyper-surfaces. J. Comput. Phys **173** (2001) 730–764
10. Leveque, R.: Finite Volume Methods for Hyperbolic Problems. Cambridge University Press (2002)
11. Sethian, J.A.: Level Set Methods and Fast Marching Methods. Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Materials Science. second edn. Cambridge University Press (1999)
12. Zhao, H.K.: Fast sweeping method for eikonal equations. Math. Comp. **74** (2005) 603–627
13. Tsitsiklis, J.: Efficient algorithms for globally optimal trajectories. IEEE Trans. Autom. Control **40**(9) 1528–1538
14. Osher, S., Sethian, J.: Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations. J. Comput. Phys. **79** (1988) 12–49
15. Jaromczyk, J., Toussaint, G.: Proc. ieee. Relative Neighborhood Graphs and Their Relatives **80**(9) (1992) 1502–1517
16. Elmoataz, A., Lézoray, O., Bougleux, S., Ta, V.T.: Unifying local and nonlocal processing with partial difference operators on weighted graphs. In: Proc. LNLA. (2008) 11–26
17. Efros, A., Leung, T.: Texture synthesis by non-parametric sampling. In: Proc. ICCV. (1999) 1033–1038
18. Buades, A., Coll, B., Morel, J.: Nonlocal image and movie denoising. IJCV **76**(2) (2008) 123–139
19. Gilboa, G., Osher, S.: Nonlocal operators with applications to image processing. Report 07-23, UCLA (2007)
20. Bougleux, S., Elmoataz, A., Melkemi, M.: Discrete regularization on weighted graphs for image and mesh filtering. In: Proc. SSVM. (2007) 128–139
21. Ta, V.T., Elmoataz, A., Lézoray, O.: Partial difference equations over graphs: Morphological processing of arbitrary discrete data. In: Proc. ECCV. (2008) 668–680
22. Brockett, R., Maragos, P.: Evolution equations for continuous-scale morphological filtering. IEEE Trans. Signal Process. **42**(12) (1994) 3377–3386