Submission Template for IET Research Journal Papers

Morphological PDEs on Graphs for Saliency Detection

Imane Bouraoui[1,2*], François Lozes[2], Adberrahim Elmoataz[2]

[1]Dept. of Telecommunications, Electronics and Computer Science Faculty, USTHB, P.O. Box 32 El-Alia Bab-Ezzouar 16111, Algiers, Algeria
[2]Image Team, University of Caen Normandy and the ENSICAEN in the GREYC Laboratory, 6 Boulevard Maréchal Juin, 14050 Caen Cedex, France
[*]imbouraoui@gmail.com

Abstract: Visual saliency is a computational process that seeks to identify the most attention drawing regions from a visual point. In this paper, we propose a new algorithm to estimate the saliency based on Partial Difference Equations (PdEs) method. A local or non-local graph is first constructed from the geometry of images. Then, the transcription of PDE on graph is done and resolved by using the mean curvature flow that can be used to perform regularization and the Eikonal equation for segmentation. Finally, an extended region adjacency graph (RAG) is built, which is extended with a *k*-Nearest Neighbor graph (*k*-NNG), in the mean RGB color space of each region in order to estimate saliency. Our algorithm allows to unify a local or non-local graph processing for saliency computing. Furthermore, it works on discrete data of arbitrary topology. For evaluation, we test our method on two different datasets and 3D point clouds. Extensive experimental results show the applicability and effectiveness of our algorithm.

1. Introduction

Visual saliency is an important part of the human attention mechanism, it allows humans to extract relevant and important information from raw input percept. Saliency estimation has become a valuable tool in image processing. However, saliency detection is still a difficult task because it requires a semantic understanding of the image. A large number of algorithms and methodologies have been developed in this task.

The literature is been prolific in the field of visual attention on 2D images, which has been investigated from past five decades. [1, 2] were the first to provide theoretical foundations of visual attention mechanisms. [3] proposed saliency maps by thresholding the color, intensity and orientation maps. However, [4] devised a saliency detection model based on a concept defined as spectral residual. The model in [5] achieved the saliency maps by inverse Fourier transform on a constant amplitude spectrum and the original phase spectrum of images. [6] used kernel density estimation based nonparametric model to represent each region. [7] exploited object/background priors and cues at different levels for a better saliency detection performance by using a fusion of saliency and generic objectness. [8] incorporated the global information of the image into saliency models with different forms, the global uniqueness of color feature and some visual organization rules are combined with the local center-surround difference to generate the saliency map. [9] proposed a saliency detection model based on human visual sensitivity and the amplitude spectrum of quaternion Fourier transform. [10] exploited two contrast measures for rating global uniqueness and spatial distribution of colors in the saliency filter to generate saliency maps. In [11], saliency cues were calculated on three image layers with different scales of segmented regions and then hierarchical inference is exploited to fuse them into a single saliency map. [12] exploited the Bayesian saliency model, convex hull analysis on interest points and Laplacian sparse subspace clustering on super-pixels are used as low-level and mid-level cues, respectively. [13] proposed a saliency tree algorithm as a novel saliency model. However, global contrast and spatially weighted regional contrast have been used in [14], and a bottom-up visual saliency detection algorithm based on background and foreground seed selection has been proposed in [15]. In [31, 32] an improved wavelet-based salient-patch detector has been used to extract the visual informative patches. Therefore, the model developed in [33], is based on multiscale deep features and computed by convolutional neural networks.

In recent years, [16] propose a global saliency which is obtained through Low-Rank Representation and local saliency which is obtained through a sparse coding scheme. [17] introduce an end-to-end deep contrast network for salient object detection, their deep network consists of two complementary components, a pixel-level fully convolutional stream and a segment-level spatial pooling stream. [18] aggregate various saliency maps based on continuous version of conditional random field with different learning and inference. Therefore, [19] present a discriminative regional feature integration approach, which learns a random forest regressor to discriminatively integrate the saliency features for saliency computation. [20] propose a novel framework to refine a saliency map derived from recent state-of-the-art salient object detection methods. However, [21] propose an effective saliency optimization scheme by taking account of the foreground appearance and background prior.

While saliency in images has been extensively studied in recent years, there is very little work on saliency of point clouds, such as presented in [22, 23].

The main goal of this paper is to propose a new definition of the saliency on graph. This new definition is based on the spectral analysis. An algorithm is given to estimate the saliency on images and point clouds based on the Partial difference Equations (PdEs) method. A local or non-local graph is first constructed. Then, the transcription of PDE on graph is done and resolved by using the framework of PdEs [24], such as: the mean curvature flows that can be used to perform regularization and filtering, the Eikonal equation for segmentation. Finally, an extended region adjacency graph (RAG) is built, which is extended with a K-NN graph, in the mean RGB color space of each region in order to estimate saliency.

The rest of this paper is organized as follows. In Section 2, we provide basic definitions and notations on graphs that will be used in the subsequent sections. In Section 3, we present our proposed algorithm for saliency estimation. Then, we provide experimental results using our proposed approach to estimate the saliency on images and 3D point clouds, in section 4. Finally, the conclusion of this work is presented in section 5.

2. Morphological PDEs on graph

In this section, we recall the definition of PDE method, which enables us to describe many PDEs models and algorithms designed for image processing and points cloud.

Let $G = (V, E, w)$ be a weighted graph, where $V = \{v_1, \ldots, v_N\}$ is a finite set of N vertices and $E \subset V \times V$ of a finite set of weighted edges. Let $(v_i, v_j) \in E$ connects two adjacent vertices v_i and v_j. The weight $w(v_i, v_j)$ of an edge is defined by the function $w: V \times V \to \mathbb{R}^+$ if $(v_i, v_j) \in E$ and $w(v_i, v_j) = 0$, otherwise.

Let $f: V \to \mathbb{R}$ be a function of the Hilbert space $H(V)$ or real-valued functions defined on the vertices of a graph. The difference operator of f , is defined by:

$$\partial_{v_j} f(v_j) = \sqrt{w(v_i, v_j)}\left(f(v_i) - f(v_j)\right) \tag{1}$$

From this definition, the external and internal morphological directional partial derivative operators are respectively defined as [24]:

$$\partial_{v_j}^{\pm} f(v_i) = (\partial_{v_j} f(v_i))^{\pm} \tag{2}$$

where $(X)^+ = \max(X, 0)$ and $(X)^- = -\min(X, 0)$.

Discrete upwind non-local weighted gradients are defined as:

$$(\nabla_w^{\pm} f)(v_i) = \left(\left(\partial_{v_j}^{\pm} f\right)(v_i)\right)_{v_j \in V}^T \tag{3}$$

The Laplacian norms $\mathcal{L}_\infty$ and $\mathcal{L}_p$, with $p = \{1,2\}$, of these gradients are, respectively, defined by:

$$\|(\nabla_w^{\pm} f)(v_i)\|_\infty = \max_{v_j \sim v_i}\left(w(v_i, v_j)\left|\left(f(v_j) - f(v_i)\right)^{\pm}\right|\right) \tag{4}$$

$$\|\nabla_w^{\pm} f(v_i)\|_p = \left[\sum_{v_j \sim v_i} w^p(v_i, v_j)\left[\left(f(v_j) - f(v_i)\right)^{\pm}\right]^p\right]^{\frac{1}{p}} \tag{5}$$

with the following equality:

$$\|(\nabla_w f)(v_i)\|_p^p = \|(\nabla_w^+ f)(v_i)\|_p^p + \|(\nabla_w^- f)(v_i)\|_p^p \tag{6}$$

where $v_j \sim v_i$ denotes two adjacent vertices.

This latter recovers the usual expression of algebraic morphological external and internal gradients, which correspond to dilatation and erosion, respectively.

The adapted well-known Eikonal equation on continuous domain is defined as [27]:

$$\frac{\partial f}{\partial t}(v, t) = F(v)\|\nabla f(v, t)\|_p, \qquad F(v) \in \mathbb{R} \tag{7}$$

to the discrete following equation on graph [25]:

$$\frac{\partial f}{\partial t}(v_i, t) = F^+(v_i)\|\nabla_w^+ f(v_i)\|_p - F^-(v_i)\|\nabla_w^- f(v_i)\|_p \tag{8}$$

This equation corresponds to a dilatation when $F > 0$ and an erosion when $F < 0$.

3. Proposed algorithm for saliency detection

In this section, we present our proposed algorithm for saliency detection, as shown in (fig. 1). First, we construct a local or non-local graph. Then, we use the framework of PdE method, that defined discrete difference operators on graph [34, 35, 36]; we use the definition of the mean curvature flows on graphs and the morphological scheme, described in [25], that it can be used to perform regularization. Then, we discuss the connection of the Eikonal equation [27] to weighted graphs. After, we describe steps to compute the saliency descriptor on graphs based on the spectral analysis. Finally, we build an extended region adjacency graph (RAG), which is extended with a k-NN graph, for which each vertex represents a region of the image and the associated function is the mean color of each region.

3.1. Graph construction

Considering an image or point clouds composed by a set of vertices, such as $S = \{x_1, x_2, \ldots, x_N\} \subset \mathbb{R}^3$, each raw point x_i associates a vertex of a graph G to define a set of vertices V. The construction of such a graph consists in modeling the neighborhood relationships between the data through the definition of a set of edges E and using a pairwise distance measure $\mu: V \times V \to \mathbb{R}^+$ [26]. The weight function w defines a similarity between two vertices based on the Euclidean distance between the coordinates of the two associated points. This similarity is computed by a similarity measure $s: E \to \mathbb{R}^+$, which satisfies :

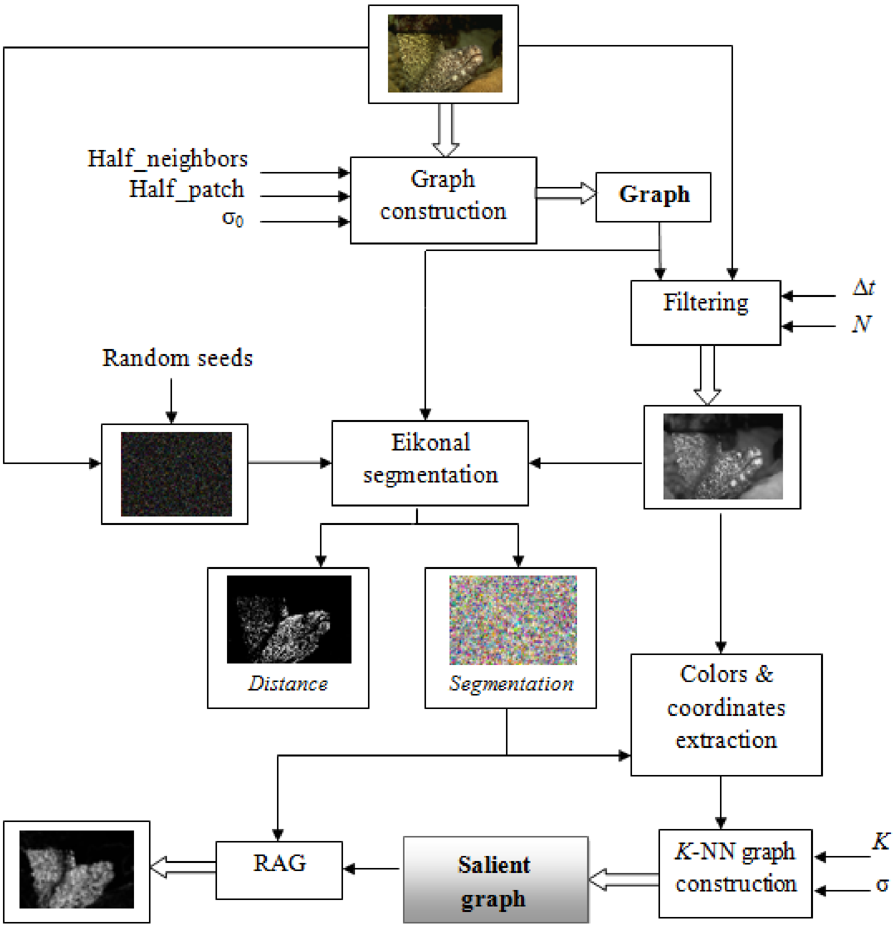

Fig. 1. *Proposed saliency detection scheme.*

$$w(v_i, v_j) = \begin{cases} s(v_i, v_j), & if\ (v_i, v_j) \in E \\ 0, & otherwise \end{cases} \quad (9)$$

The weighting of the edges is done from the following similarity function [24]:

$$\begin{cases} s_0(v_i, v_j) = 1 \\ s_1(v_i, v_j) = \exp\left(-\mu(f^0(v_i), f^0(v_j))\right)/\sigma^2 \end{cases} \quad (10)$$

where the variance parameter $\sigma > 0$ usually depends on the variation of the function μ, and $f: V \to \mathbb{R}$ is used to describe the data at a node v_i, which can be considered as a feature vector F_{v_i} or a patch feature vector noted by:

$$F_{v_i}^{\tau} = \bigcup_{v_j \epsilon \mathcal{W}^{\tau}(v_i)} F_{v_j} \quad (11)$$

where $\mathcal{W}^{\tau}(v_i)$ is a square window of size $(2\tau + 1) \times (2\tau + 1)$ centred at vertex pixel v_i and $F_{v_i}^{\tau}$ allows to incorporate nonlocal features for $\tau \geq 1$.

In the case of point clouds, the patch vector is presented as the set of values inside each oriented patch of the vertex v_i and defined as [26]:

$$\vec{\mathcal{P}}(v_i) = \left(\frac{\sum_{v_j \in V_k(v_i)} w\left(\vec{c}_k, \vec{p}_{v_j}\right) f^0(v_j)}{\sum_{v_j \in V_k(v_i)} w\left(\vec{c}_k, \vec{p}_{v_j}\right)} \right)^T_{k \in \{1,\dots,N^2\}} \quad (12)$$

where $V_k(v_i) = \left\{ v \middle| \vec{p}_{v_j} \in C_k(v_i) \right\}$ is the set of vertices v that was assigned to the k-th patch cell of v_i, $\vec{p}_{v_j}$ are the coordinates' vector of projected points, $C_k(v_i)$ denote the k-th cell of the constructed patch around v_i and $\vec{c}_k$ are the coordinates' vector of the k-th patch cell center. The weighting $w\left(\vec{c}_k, \vec{p}_{v_j}\right) = \exp\left(-\left\|\vec{c}_k - \vec{p}_{v_j}\right\|_2^2/\sigma^2\right)$ enables to take into account the point distribution with the patch cells in order to compute their feature vectors.

3.2. Regularization using mean curvature flows on graphs

The mean curvature flows filtering alternates between the non-local dilation or non-local erosion of the image according to the sign of the curvature. Equation (8) can be expressed by [25]:

$$\begin{cases} \frac{\partial f}{\partial t}(v_i,t) = \left(\kappa_w(v_i,f)\right)^+ \|\nabla_w^+ f(v_i)\|_p \\ \qquad\qquad -\left(\kappa_w(v_i,f)\right)^- \|\nabla_w^- f(v_i)\|_p \\ \\ \qquad\qquad f(v_i,0) = f^0(v_i) \end{cases} \tag{13}$$

where κ_w is the mean curvature of the function f at $v_i \in V$, defined as:

$$\kappa_w(v_i,f) = \frac{\sum_{v_i \in V} w(v_i,v_j) sign\left(f(v_i) - f(v_j)\right)}{\delta_w(v_i)} \tag{14}$$

with $sign(r) = \begin{cases} +1 & if\ r \geq 0 \\ -1 & otherwise \end{cases}$ and $f^0 \in H(V)$ represents an observation of a clean function $h \in H(V)$ corrupted by additive noise $n \in H(V)$.

The time variable can be discretized using explicit Euler method as [25]:

$$\frac{\partial f}{\partial t}(v_i) = \frac{f^{n+1}(v_i) - f^n(v_i)}{\Delta t} \tag{15}$$

The mean curvature flows algorithm used to regularize f^0 corresponds to:

$$\begin{aligned} f^{n+1}(v_i) = {} & [1 - \Delta t|\kappa_w(v_i,f^n)|] f^n(v_i) \\ & + \Delta t\left(\kappa_w(v_i,f^n)\right)^+ . NLD(f^n)(v_i) \\ & + \Delta t\left(\kappa_w(v_i,f^n)\right)^- . NLE(f^n)(v_i) \end{aligned} \tag{16}$$

with:

$$NLD(f)(v_i) = f(v_i) + \|(\nabla_w^+ f)(v_i)\|_\infty \tag{17}$$

$$NLE(f)(v_i) = f(v_i) + \|(\nabla_w^- f)(v_i)\|_\infty \tag{18}$$

This filter alternates between the non-local dilation (*NLD*) or the non-local erosion (*NLE*) of the image according to the sign of the curvature.

3.3. Segmentation with the Eikonal equation on Graphs

The segmentation formulation is based on front propagation using the Eikonal equation [27] to compute general distances on graphs. In the case where F is non-negative on the whole domain, a translation of the equation (7), is expressed as:

$$\begin{cases} \|\nabla_w^- f(v_i)\|_p = \mathrm{P}(v_i), & \forall\, v_i \in V \\ f(v_i) = 0, & \forall\, v_i \in V_0 \end{cases} \tag{19}$$

for which $V_0 \subset V$ corresponds to the initial set of seed vertices and $\mathrm{P}(v_i)$ is a potential function. This equation corresponds to a generalized form of distance computation on a Cartesian grid, by setting $w(v_i,v_j) = 1$ and $\mathrm{P}(v_i) = 1$.

For the case $p = \{1,2\}$, the local solution at a particular vertex can be easily obtained with the iterative algorithm described in [27].

3.4. Saliency on Graph

The saliency is real value estimated at each node of the graph. Let define $V_0 \subset V$ and $N = |V_0|$. Let W be a square matrix of $N \times N$ weights, such as:

$$\begin{cases} W(v_i,v_j) = w(v_i,v_j), & if\ (v_i,v_j) \in E \\ or\ W(v_i,v_i) = 1 & otherwhise \\ or\ W(v_i,v_j) = 0 & otherwhise \end{cases}$$

Let D be a diagonal matrix of $N \times N$ degrees, with $D(v_i,v_i) = \delta_w(v_i)$ and $D(v_i,v_j) = 0$ for $v_i \neq v_j$. The degree of a vertex v_i is defined as:

$$\delta_w(v_i) = \sum_{v_j \sim v_i} w(v_i,v_j) \tag{20}$$

As shown in [28], the spectral decomposition of the matrix $P = D^{-1}W$ gives a set of eigen vectors $\{\phi_1,\phi_2,\dots,\phi_N\}$ associated with their eigen values $|\lambda_1| = 1 \geq |\lambda_2| \geq \cdots \geq |\lambda_N| \geq 0$, solution of $P\phi_i = \lambda_i\phi_i$.

The saliency is defined as $\Gamma_N(v_i) = \sum_{i=1}^N \lambda_i$, which is equals to the trace of the matrix P , thus $\Gamma_N(v_i) = \sum_{i=1}^N P(i,i) = \sum_{i=1}^N (W(i,i)/D(i,i)) = \sum_{i=1}^N \left(1/\delta_W(v_j)\right)$, because $W(v_i,v_i) = 1$.

We slightly modified the saliency formula $\Gamma_N(v_i)$ to define the saliency on graph at a node v_i, such as:

$$\Gamma(v_i) = \sum_{v_j \sim v_i} \frac{1}{1 + \delta_N(v_j)} \tag{21}$$

with $\delta_N(v_i) = \frac{\delta_W(v_i)}{|v_j \sim v_i|}$ the normalized degrees at a node v_i.

Finally, the normalized saliency is determined with: $\delta(v_i) = \frac{\Gamma(v_i)}{|v_j \sim v_i|}$.

3.5. Images reconstruction

To transpose the data into "segmented" representation in the original database to obtain the saliency, we associate region-based graphs that can be adjacency graphs (RAGs) or *k*-nearest neighbour graphs (*k*-NNGs).

Any discrete domain can be modelled by a weighted graph where each data point is represented by a vertex $v_i \in V$. This domain can represent unorganized or organized data where functions defined on V correspond to the data to process. In the case of unorganized data; *k*-NNG is used where each vertex v_i is connected with its *k*-nearest neighbours according to a pairwise distance μ. In the case of structured data; Region adjacency graphs (RAG) can be built for any structured data represented by a graph, where a region R_i is defined as a set of connected vertices such that $\bigcup R_i = V$ and $\bigcap R_i = \emptyset$. Two regions R_i and R_j are adjacent if $\exists v_i \in R_i\ and\ v_j \in R_j \,|\, v_j \sim v_i$ [24, 27].

4. Experimental results

In this section, we present our experimental steps and its application on 2D images and 3D point clouds. Then, we test our method on the 5000 benchmark test images of MSRA dataset used in [29] and 300 images of SOD dataset [30], these images database include original images and their corresponding ground-truth saliency maps. Finally, we compare our results with results of other methods in the state of the art.

The quantitative evaluation for a saliency detection algorithm is to see how much the saliency map from algorithm overlaps with the ground-truth saliency map, and then for a ground-truth saliency map GT and the detected saliency map S, we have:

$$precision = \frac{\sum_x S_x GT_x}{S_x} \tag{22}$$

$$recall = \frac{\sum_x S_x GT_x}{GT_x} \tag{23}$$

Then, we calculate *F-measure*, which is the harmonic mean of precision and recall, to evaluate the overall detection performance as follows:

$$F\_measure = \frac{(1+\beta^2) \times precision \times recall}{\beta^2 \times precision + recall} \tag{24}$$

where the coefficient β^2 is set to 0.3 as used in literature.

The Mean Absolute Error (MAE) is also used to measure how much the foreground is highlighted and the background is suppressed. The MAE is defined as:

$$MAE = \frac{1}{w.h} \sum\nolimits_x^{w.h} |S_x - GT_x| \tag{25}$$

4.1. Experiments

Our experimental steps are presented in (fig. 1) and summarized as follows:

1. The creation of local or non-local graph consists of the creation of vertices from raw data to process, these vertices are connected with edges and weights associated to each edge are deduced. In local graph, only local close neighbours are considered during the creation of edges. In a non-local graph, edges are created between vertices that are spatially far apart. Weights on each edge are deduced from values associated to vertices and patches can be used to compute a better similarity value accounting local neighbourhood similarities. In the case of point clouds, the patch is constructed from two-dimensional grid describing the close neighbourhood around each vertex, this grid is defined on the tangent plane of the vertex. Then, the patch is oriented accordingly in order to be filled in with a weighted average of the graph signal values in the local neighbourhood. [26]
2. The filtering consists to provide illustrations of mean curvature flows, which enables to group similar vertices around high curvature regions. The time discretization iterative algorithm, equation (16), has been used to perform regularization. Filtering results of local and non-local graphs are illustrated in (fig. 2 and 3), respectively.

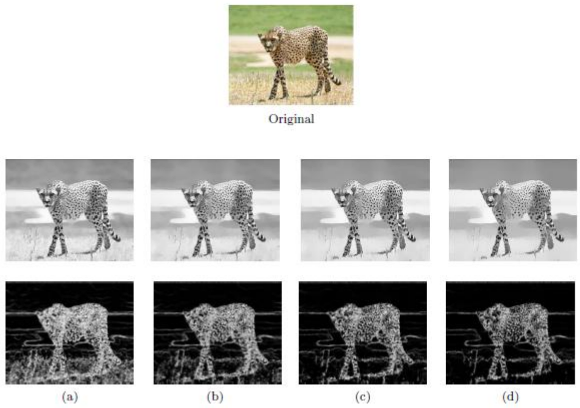

Fig. 2. *Illustration of local filtering (first line) and their corresponding saliency (second line) after* ***(a)*** *20,* ***(b)*** *50,* ***(c)*** *100 and* ***(d)*** *200 iterations.*

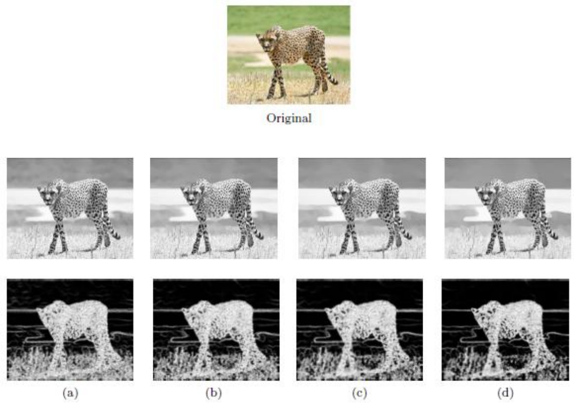

Fig. 3. *Illustration of non-local filtering (first line) and their corresponding saliency (second line) after* ***(a)*** *20,* ***(b)*** *50,* ***(c)*** *100 and* ***(d)*** *200 iterations and* (5×5) *patch size.*

3. Segmentation consists of partitioning the filtered data to multiple regions using the Eikonal equation (19) and grouping pixels in a region map while preserving boundaries, as illustrated in (fig. 4), in order to reduce image complexity.

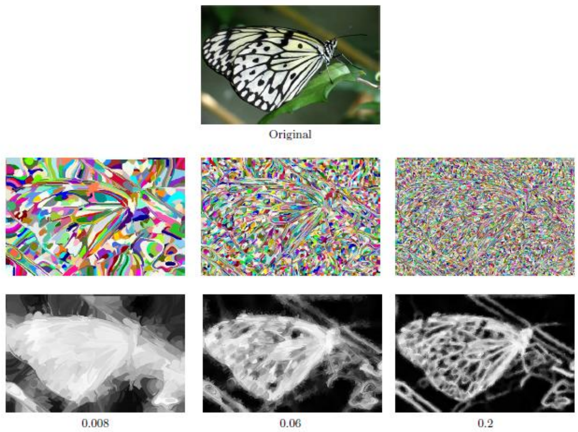

Fig. 4. *Illustration of color segmentation (first line) and their corresponding saliency (second line) according to random seeds percentage*

4. This latter gives a reduced version of the image and allows us to reconstruct a new k-NN graph. In the other hand, such partition can be easily transformed in a Region Adjacency Graph (RAG), which can be used as a simplified version of the initial image that preserves texture information and strong boundaries.

4.2. Results and discussion

In this sub-section, we present the results of saliency detection on 2D images and 3D point clouds.

First, we start by presenting the results of saliency on 2D images. On all the experiments, the following parameters have been used: the k nearest neighbour equal to 3 for graph creation with $\sigma_0 = 15$, $\Delta t = 0.5$, *random seeds* = 20% for the region. k-NN = 8 and $\sigma = 10$ in each region for graph reconstruction.

Fig. 2 and 3, show an example of local and nonlocal with (5×5 *patchs*) filtering of a 2D image after 20,50, 100 and 200 iterations, respectively. We can see that nonlocal structure better preserves image details and gives good precision after 100 iterations.

Fig. 4 illustrates saliency with the variation of random seeds percentage in non-local graph and after 100 iterations. The random seed controls the precision of the segmentation (respectively from low to high for a coarse to fine segmentation). The more segmentation is the coarse segmentation, the more saliency computing will be fast. A good tradeoff between precision and speedness is 20%, as shown in table 1.

Table 1 Evaluation of Random seeds according to the number of mean curvature flow iterations using nonlocal graph of MSRA dataset images

Seeds	F-measure			MAE		
	20	50	100	20	50	100
5%	0.416	0.572	0.664	0.263	0.243	0.229
10%	0.612	0.750	0.801	0.240	0.221	0.206
20%	0.730	0.793	0.857	0.229	0.210	0.165

Fig. 5 presents precision-recall curves of local and non-local graphs on MSRA and SOD datasets, after 100 iterations with 20% random seeds by using the same parameters described above. We can notice that the use of non-local graph on the MSRA database gives the best performance.

Moreover, as shown in fig. 6, we can remark the capability of processing different scenes for our approach using non-local graph, it can also highlights the internal object with their contour by detecting the outlines of the saliency and their interior of small, large and transparent objects. However, it cannot suppress the background well in the case of complex textures and multiple objects.

Then, we illustrate results of local and non local saliency detection in 3D point clouds models are obtained from (http://www.cloudcasterlite.com) using the same parameters of 2D images, as shown in Fig. 7.

The saliency is obtained from the coordinates of the nodes to build the patch of 100 cells and on the colors of the nodes during patch construction. It can be noted that the planar regions appear non-salient while the fluctuating regions are considered salient, thus the regions having significant color variations appear salient. Knowing that warm colors (red and yellow) present a high saliency then cold colors (blue and green) present a low saliency. We can also notice that the saliency obtained from the local graph is less important than that obtained from the non-local graph.

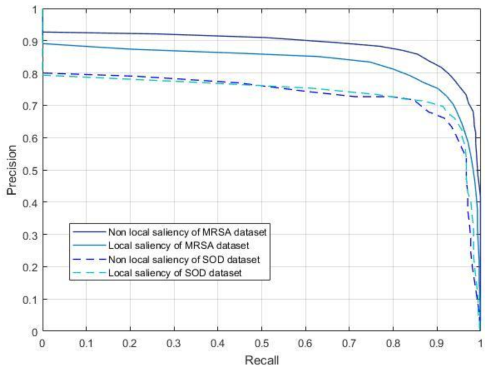

Fig. 5. *Precision-recall curves of proposed saliency model on MSRA and SOD dataset.*

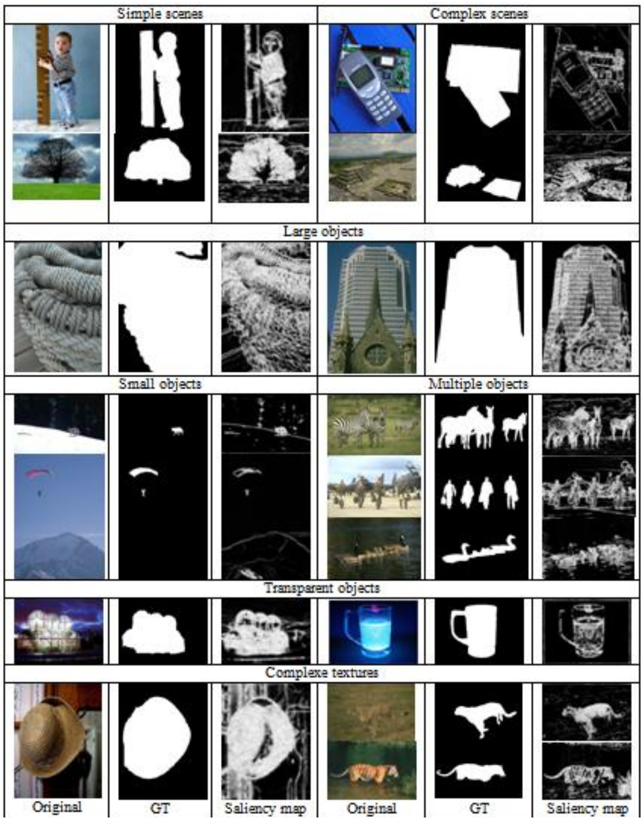

Fig. 6. *Selected results of our approach.*

4.3. Comparison

In this subsection, we compare the performance and computational time of our approach with several state-of-the-art methods including: SVO [7], CA [8], SF [10], HS [11], RC [14], MDF [33], DCL [17], DRFI [19] and FABP [21].

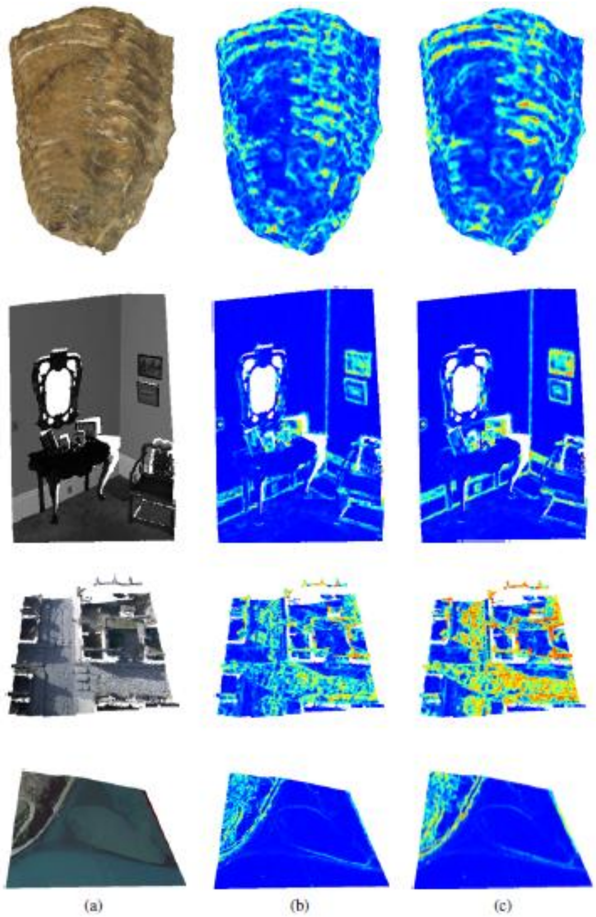

Fig. 7. Saliency detection in 3D point clouds (a) original, saliency map of (b) local graph and (c) non-local graph.

From fig. 8, 9 and 10, the proposed method shows better performance in Precision-Recall curve, F-measure, and MAE value on the both MSRA and SOD datasets than SVO [7], CA [8], SF [10], HS [11], RC [14] and FABP [21]. So, the unsupervised methods, such as SVO [7], CA [8] and RC [14] have a weak background suppression, as shown in fig. 11. Therefore, SF [10] has a better ability to suppress the background but it cannot highlight the whole object when the object consists of different colors. However, parameters on HS [11] and FABP [21] are crucial to performance. Otherwise, supervised learning methods (fig. 12), such as: DCL [17], MDF [33] and DRFI [19] show better performance than ours; DCL [17] only simply fuse the skip layers with different scales for more advanced feature representation building, MDF [33], trained a deep neural network for deriving a saliency map from multiscale features extracted using deep convolutional neural networks, and DRFI [19] fuses the saliency scores across multiple levels, yielding the saliency map but it has limited ability to discover all the salient objects within one image.

Our saliency graph-based algorithm has the benefit to unify local and non-local processing. Furthermore, the presented approach works on discrete data of arbitrary topology, so the same algorithm works on both 2D images and 3D point clouds. Moreover, as opposed to deep learning approaches, our approach is ready to use without the need to select hyper-parameters nor compute databases.

The proposed framework has been implemented in C++. The computation time of our proposed framework is 17.09 s per image (300×400) on an Intel i7-3612QM CPU (2.10 GHz) and 8 GB of memory, as shown in table 2, it takes more times compared with other methods, which is a limitation.

Table 2 Comparison of the average running time

Method	CA	RC	SF	FABP	Ours
Code	Matlab	C++	C++	Matlab	C++
Times (s)	53.1	0.253	0.202	4.2361	17.09

5. Conclusion and future work

In this paper we have applied the framework of PdEs on weighted graphs which is new approach for saliency detection on images and point clouds, by exploiting a new definition of the saliency on graphs, the mean curvature flows for filtering, Eikonal equation for segmentation and RAG and *k*-NN graph reconstruction. Experimental results show the applicability and effectiveness of our saliency model. In the near future, we would like to add another algorithm in order to improve the saliency results by ensuring good background suppression and faster computation time. It is also proposed, a deeper study of saliency on 3D point clouds.

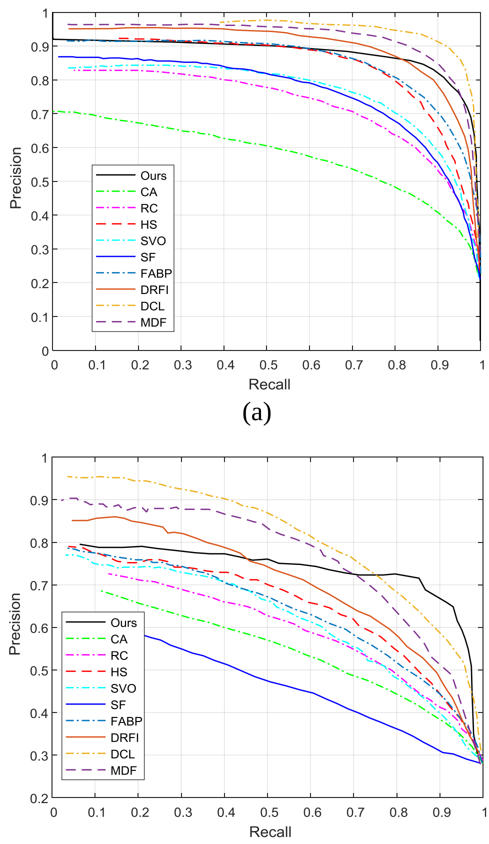

Fig. 8. *Precision-recall curves of different saliency models on* (a) *MSRA and* (b) *SOD dataset.*

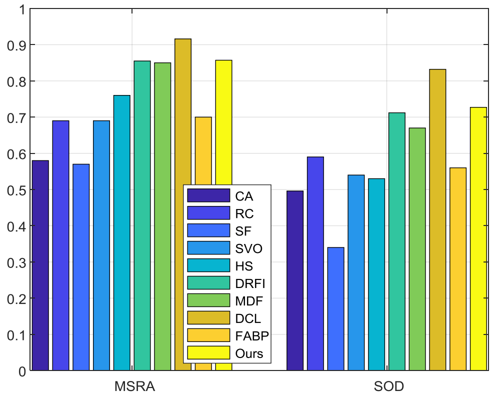

Fig. 9. *F-measure of different saliency models on MSRA and SOD dataset.*

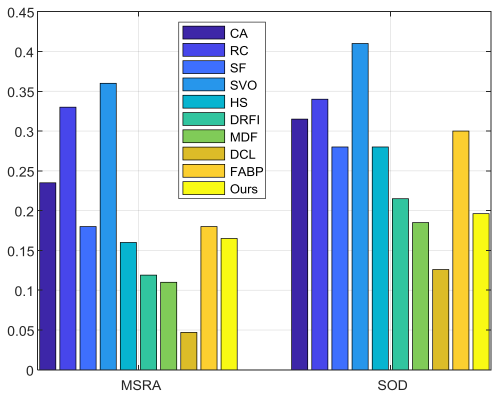

Fig. 10. *MAE of different saliency models on MSRA and SOD dataset.*

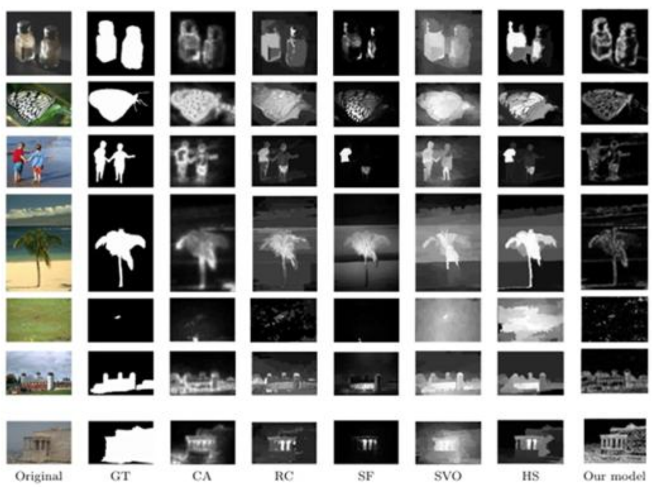

Fig. 11. *Visual comparison of saliency maps generated from state-of-the-art methods, including our graph based approach and the ground truth (GT)*

6. References

[1] Koch, C., Ullman, S.: 'Shifts in selective visual attention: Towards the underlying neural circuitry', Journal of Human Neurobiology, 1985, 4, pp. 219–227.

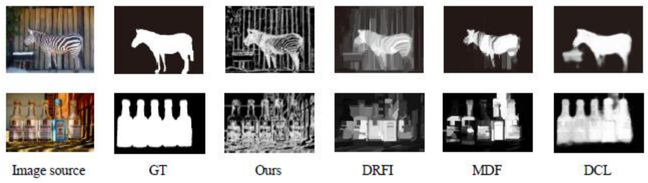

Fig. 12. *Visual comparison of saliency maps generated from supervised learning methods, including our graph based approach and the ground truth (GT)*

[2] Itti, L., Koch, C., Niebur, E.: 'A model of saliency based visual attention for rapid scene analysis', IEEE Transactions on Pattern Analysis and Machine Intelligence, 1998, 20, pp. 1254–1259.

[3] Hu, Y., Xie, X., Chia, L., Rajan, D.: 'Salient region detection using weighted feature maps based on the human visual attention mode'. Springer Lecture Notes in Computer Science, 2004, pp. 993–1000.

[4] Hou, X., Zang, L.: 'Saliency detection: a spectral residual approach', in Proceedings of the 25th IEEE Conference on Computer Vision and Pattern Recognition, Minneapolis, USA, June 2007, pp. 1-8.

[5] Guo, C., Ma, Q., Zang, L.: 'Spatio-temporal Saliency detection using phase spectrum of qu aternion fourier transform', in Proceedings of the 26th IEEE Conference on Computer Vision and Pattern Recognition Anchorage, USA, 23-28 June 2008, pp. 1-8

[6] Liu, Z., Xue, Y., Shen, L., Zhang, Z.: ' Nonparametric saliency detection using kernel density estimation', in Proc. IEEE International Conference on Image Processing, Hong Kong, China, Sep, 2010, pp. 253–256

[7] Chang, K. Y., Liu, T. L., Chen, H. T., Lai, S. H.: 'Fusing generic objectness and visual saliency for salient object detection', in Proc. 13th IEEE International Conference on Computer Vision, Barcelona, Spain, Nov. 2011, pp. 914–921

[8] Goferman, S., Zelnik-Manor, L., Tal, A.: 'Context-aware saliency detection', IEEE Transactions on Pattern Analysis and Machine Intelligence, 2012, Vol. 34 (10), pp. 1915 - 1926.

[9] Fang, Y., Lin, W., Lee, B.S., Lau, C.T., Chen, Z., Lin, C.W. 'Bottom-up saliency detection model based on human visual sensitivity and amplitude spectrum'. IEEE Transactions on Multimedia, 2012. 14, 187–198

[10] Perazzi1, F., Krähenbül, P. , Pritch, Y. , Hornung, A., "Saliency filters: Contrast based filtering for salient region detection," in Proc. 25th IEEE Conference on Computer Vision and Pattern Recognition, Rhode, Island, Jun. 2012, pp. 733–740.

[11] Yan, Q., Xu, L., Shi, J., Jia, J.: 'Hierarchical saliency detection', in Proc. 26th IEEE Conference on Computer

Vision and Pattern Recognition, Portland, USA, Jun. 2013, pp. 1155–1162.

[12] Xie, Y., Lu, H., Yang, M.H.: 'bayesian saliency via low and mid level cues'. IEEE Transactions on Image Processing, 2013, 22, 1689–1698

[13] Liu, Z., Zou, W., Le Meur, O.: 'Saliency tree : A novel saliency detection framework'. IEEE Transactions on Image Processing, 2014. 23, 1937–1952.

[14] Cheng, M. M., Mitra, N. J., Huang, X., Torr, P. H. S., and Hu, S. M.: 'Global contrast based salient region detection', IEEE Transactions on Pattern Analysis and Machine Intelligence, 2015, vol. 37 (3), pp. 569–582

[15] Wang, J., Lu, H., Li, X., Tong, N., Liu, W.: 'Saliency detection via background and foreground seed selection', Neurocomputing, 2015, vol. 152, pp. 359-368.

[16] Fan, Q., Qi, C.: 'Saliency detection based on global and local short-term sparse representation', Neurocomputing, 2016, vol. 175, Part A, pp. 81-89.

[17] Li, G., Yu, Y.: 'Deep contrast learning for salient object detection', in IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, USA, June 2016. DOI:10.1109/CVPR.2016.58

[18] Fu, K., Gu, Y.-H., Yang, J.: 'Saliency detection by fully learning a continuous conditional random field', IEEE Trans. Multimedia, 2017, 19 (7), pp. 1531–1544.

[19] Wang, J., Jiang, H., Yuan, Z., Cheng, M.-M., Hu, X., Zheng, N.: 'Salient object detection: A discriminative regional feature integration approach', Int. Journal of Computer Vision, 2017, vol. 123(2), pp. 251– 268.

[20] Eun, H., Kim, Y., Jung, C., Kim, C.: 'Saliency refinement: Towards a uniformly highlighted salient object', Signal Processing: Image Communication, 2018, 62, pp.16–32

[21] Li, L., Zhou, F., Zheng, Y., Bai, X.: 'Saliency detection based on foreground appearance and background-prior, Neurocomputing, 2018, doi: 10.1016/j.neucom.2018.03.049

[22] Tasse, F.P., Kosinka, J., Dodgson, N.A.: 'Cluster-based point set saliency'. In IEEE International Conference on Computer Vision, Santiago, Chile, 2015, pp. 637–639

[23] Guo, Y., Wang, F., Xin, J.: 'Point-wise saliency detection on 3D point clouds via covariance Descriptors'. The Visual Computer journal, Springer-Verlag, June 2017, DOI 10.1007/s00371-017-1416-3

[24] Elmoataz, A., Toutain, M., Tenbrinck, D.: 'On the p-laplacian and 1- laplacian on graphs with applications in image and data processing'. SIAM Journal on Imaging Sciences, 2015, vol.8, 2412-2451

[25] Chakik, A.E., Elmoataz, A., Desquesnes, X.,: 'Mean curvature flow on graphs for image and manifold restoration and enhancement', Signal Processing, 2014, vol. 105, 449-463

[26] Lozes, F., Elmoataz, A., Lezoray, O.: 'Partial difference operators on weighted graphs for image processing on surfaces and point clouds', Image Processing, IEEE Transactions, 2014, vol. 23, pp. 3896-3909

[27] Desquesnes, X., Elmoataz, A., Lezoray, O.: 'Eikonal equation adaptation on weighted graphs: fast geometric diffusion process for local and non-local image and data processing. Journal of Mathematical Imaging and Vision, 2013, vol. 46, pp.238-257.

[28] Coifman, R.R., Lafon, S.: 'Diffusion maps'. Applied and computational harmonic analysis, 2006, 21(1), pp. 5-30.

[29] Liu, T., Yuan, Z., Sun, J., Wang, J., Zheng, N., Tang, X., Shum, H.: 'Learning to detect a salient object', IEEE Transactions on Pattern Analysis and Machine Intelligence, 2011, vol. 33(2), pp. 353-367.

[30] Movahedi, V., Elder, J. H.: 'Design and perceptual validation of performance measures for salient object segmentation', in IEEE Conference on Computer Vision and Pattern Recognition -Workshops, San Francisco, USA, June 2010, pp. 49-56.

[31] Jian, M., Lam, K..M., Dong, J.: 'Facial-feature detection and localization based on a hierarchical scheme'. Information Sciences, 2014, vol. 262, pp. 1–14.

[32] Jian, M., Lam, K..M., Dong, J., Shen, L.: 'Visual-Patch-Attention-Aware Saliency Detection'. IEEE Transactions on Cybernetics, 2015, vol. 45 (8), pp. 1575 - 1586.

[33] Li, G., Yu., Y.: 'Visual saliency based on multiscale deep features'. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, USA, June 2015. DOI: 10.1109/CVPR.2015.7299184

[34] Elmoataz, A., Lezoray, O., Bougleux, S.: 'Nonlocal discrete regularization on weighted graphs: A framework for image and manifold processing'. IEEE Transaction on Image Processing, 2008, vol. 17, pp. 1047-1060

[35] Ta, V.T., Elmoataz, A., Lezoray, O.: ' Nonlocal PDEs-based morphology on weighted graphs for image and data processing'. IEEE Transaction on Image Processing, 2011, vol. 26, pp. 1504-1516

[36] Elmoataz, A., Lozes, F., Toutain, M.: ' Nonlocal PDEs on graphs: From tug-of-war games to unified interpolation on images and point clouds'. Journal of Mathematical Imaging and Vision, 2017, vol. 57, pp. 381-401