J Math Imaging Vis (2018) 60:479–502
https://doi.org/10.1007/s10851-017-0768-7

Hierarchical Segmentations with Graphs: Quasi-flat Zones, Minimum Spanning Trees, and Saliency Maps

Jean Cousty[1] · Laurent Najman[1] · Yukiko Kenmochi[1] · Silvio Guimarães[1,2]

Received: 30 January 2017 / Accepted: 3 October 2017 / Published online: 20 October 2017

Abstract Hierarchies of partitions are generally represented by dendrograms (direct representation). They can also be represented by saliency maps or minimum spanning trees. In this article, we precisely study the links between these three representations. In particular, we provide a new bijection between saliency maps and hierarchies based on quasi-flat zones as often used in image processing and we characterize saliency maps and minimum spanning trees as solutions to constrained minimization problems where the constraint is quasi-flat zones preservation. In practice, these results make up a toolkit for designing new hierarchical methods where one can choose the most convenient representation. They also invite us to process non-image data with morphological hierarchies. More precisely, we show the practical interest of the proposed framework for: (i) hierarchical watershed image segmentations, (ii) combinations of different hierarchical segmentations, (iii) hierarchicalizations of some non-hierarchical image segmentation methods based on regional dissimilarities, and (iv) hierarchical analysis of geographic data.

✉ Jean Cousty
j.cousty@esiee.fr

Laurent Najman
l.najman@esiee.fr

Yukiko Kenmochi
y.kenmochi@esiee.fr

Silvio Guimarães
s.guimaraes@esiee.fr; sjamil@pucminas.br

1 Université Paris-Est, Laboratoire d'Informatique Gaspard-Monge, A3SI, ESIEE Paris, CNRS, Marne-la-Vallée, France

2 PUC Minas - ICEI - DCC - VIPLAB, Belo Horizonte, Brazil

Keywords Mathematical morphology · Hierarchy of partitions · Hierarchical image segmentation · Watershed · Saliency maps · Minimum spanning trees · Hierarchical classification

1 Introduction

Many image segmentation methods look for a partition of the set of image pixels such that each region of the partition corresponds to an object of interest in the image. Hierarchical segmentation methods, instead of providing a unique partition, produce a sequence of nested partitions[1] at different scales, allowing the description of an object of interest as a grouping of several objects of interest that appear at lower scales.

Since the early work of [39], hierarchical image analysis has been the subject of intense research. For instance, one can refer to hierarchical watersheds, pioneered in [7,28,36], to quasi-flat zone hierarchies, studied notably in [29,33,50], to binary partition trees, introduced in [45], and to the scale-set theory, initiated in [17]. In the few last years, hierarchical segmentation has become a hot topic as attested by the popularity of [3], which presents a hierarchical segmentation machinery that reaches excellent practical results on the Berkeley image segmentation dataset.

This article deals with a theory of hierarchical segmentation as used in image processing. More precisely, we investigate different representations of a hierarchy: by a den-

[1] There also exist hierarchical image segmentation and filtering methods, such as, e.g., [46] and [31], that deal with series of nested partial partitions (i.e., nested partitions of subsets of the image pixels). The study of these methods is beyond the scope of this article. The interested reader can refer to [43] for an algebraic study encompassing these methods.

drogram (direct set representation), by a saliency map (a characteristic function), and by a minimum spanning tree (a reduced domain of definition). Our theoretical contributions are threefold:

1. a new bijection theorem between hierarchies and saliency maps (Theorem 1) relying on the quasi-flat zone hierarchies that is simpler and more general than previous bijection theorems for saliency maps;
2. a new characterization of the saliency map of a given hierarchy as the minimum function for which the quasi-flat zones hierarchy is precisely the given hierarchy (Theorem 2); and
3. a new characterization of the minimum spanning trees of a given edge-weighted graph as the minimum subgraphs (for inclusion) whose quasi-flat zone hierarchies are the same as the one of the given graph (Theorem 4).

The links established in this article between the maps that weight the edges of a graph, the hierarchies on the vertex set of this graph, the saliency maps on the edges of this graph, and the minimum spanning trees for the maps that weight the edges of this graph are summarized in the diagram of Fig. 1.

One possible application of these results is the design of algorithms for computing hierarchies. Indeed, our results allow one to use indifferently any of the three hierarchical representations. This can be useful when a given operation is more efficiently performed with one representation than with the two others. Naturally, one could work directly on the hierarchy (or on its tree-based representation, called a dendrogram) and finally compute a saliency map for visualization purposes. For instance, in [17,23], the authors efficiently handle directly the tree-based representation of the hierarchy. Conversely, due to Theorem 1, one can work on a saliency map or, due to Theorem 4, on the weights of a minimum spanning tree and explicitly compute the hierarchy in the end. In [11,13,26,35], a resulting saliency map is computed before a possible extraction of the associated hierarchy of watersheds. In [18], a basic transformation that consists of modifying one weight on a minimum spanning tree according to some criterion is considered. The corresponding operation on the equivalent dendrogram is more difficult to design. When this basic operation is iterated on every edge of the minimum spanning tree, one transforms a given hierarchy into another one. The technique is generic and was applied in [18,20,21] to the measures presented in [15,38,40], respectively. An in-depth exploration of one of these measures, namely the observation scale of Felzenswalb and Huttenlocher [15], is detailed in [19]. In particular, in [19], an extensive assessment based on the framework of [3] shows that the hierarchical method performs at least as well as its non-hierarchical counterpart while providing at once all the possible scales (see, e.g., Fig. 2). The results of this article constitute the theoretical basis of the methods presented in the aforementioned references [11,13,18,20,21,35]. It also opens the door toward new hierarchical image analysis. As an example, we present, in Sect. 8.3, definitions of interesting combinations of hierarchies featuring distinct aspects of a same image. We also provide an efficient combination algorithm based on saliency maps (quasi-linear-time algorithm with-respect to the size underlying graph, provided that the graph edges are either already sorted or can be sorted in linear time).

Another interest of our work is to enable a precise link between hierarchical classification [16,37,47] and hierarchical image segmentation. In particular, it suggests that hierarchical image segmentation methods can be used for

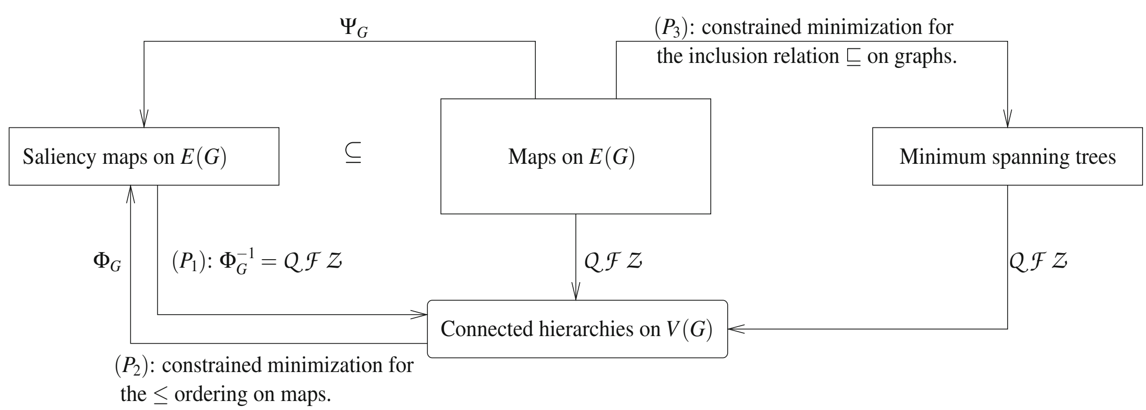

Fig. 1 A diagram that summarizes the results of this article. The solutions to problems (P_1), (P_2), and (P_3) are given by Theorems 1, 2, and 4, respectively. The constraint involved in (P_2) and (P_3) is to leave the induced quasi-flat zones hierarchy unchanged. In the diagram, the symbols G, $V(G)$, and $E(G)$ are used to denote a connected graph, its vertex set, and its edge set, respectively. The symbol $\mathcal{QFZ}$ stands for quasi-flat zones [Eq. (3)], and the symbols Φ_G and Ψ_G stand for the saliency map of a hierarchy [Eq. (5)] and of a map, respectively (Sect. 5)

classification (the converse being carried out for a long time). Indeed, our work is deeply related to hierarchical classification, more precisely, to ultrametric distances, subdominant ultrametrics and single linkage clustering. In classification, representations of hierarchies, on which no connectivity hypothesis is made, are studied since the 1960s. The framework presented in this article deals with connected hierarchies and a graph needs to be specified for defining the connectivity of the regions of the partitions in the hierarchies. The connectivity of regions is the main difference between what has been done in classification and in segmentation. Rather than restricting the work done for classification, the framework studied in this article generalizes it. Indeed the usual notions of classification are recovered from the definitions of this article when a complete graph (every two points are linked by an edge) is considered. For instance, for a complete graph, a saliency map becomes an ultrametric distance, which is known to be equivalent to a hierarchy. However, Theorem 1 shows that when the graph is not complete, we do not need a value for every pair of elements in order to characterize a hierarchy (as done with an ultrametric distance), but one value for each edge of the graph is enough (with a saliency map). Furthermore, when a complete graph is considered, the hierarchy of quasi-flat zones becomes the one of single linkage clustering (see, e.g., [50]). Hence, Theorem 4 allows to recover and to generalize a well-known relation between the minimum spanning trees of the complete graph and single linkage clustering. In order to emphasize the links drawn in this paper between hierarchical segmentation and classification, we present in Sect. 8.4 an original hierarchical analysis of geographic data. We indeed investigate the Knuth Miles dataset [32] (a dataset of 128 US cities with population and position information) with a hierarchical segmentation scheme coming from image analysis, namely hierarchical watershed (see, e.g., [7,11,28,36]).

This article is organized as follows: Sects. 2 and 3 recall basic notions for handling connected hierarchies and quasi-flat zones, respectively; Sect. 4 introduces the notion of a saliency map and provides the correspondence between saliency maps and hierarchies (Theorem 1); Sects. 5 and 6 characterize saliency maps and minimum spanning trees as solutions to constrained minimization problems, where the constraint is quasi-flat zones preservation; Sect. 7 presents a linear-time algorithm for computing the saliency map of a hierarchy and a quasi-linear-time algorithm for the ultrametric opening (i.e., the transformation denoted by Ψ_G in Fig. 1); Finally, Sect. 8 illustrates the versatility of the proposed framework with applications to image, mesh and geographic data processing.

This article extends an article ([12]) published in a conference. In particular, it contains the proof of all properties presented in [12] and illustrations of the proposed framework to image and geographic data analysis.

2 Connected Hierarchies of Partitions

In this section, we provide basic definitions for handling partitions, hierarchies and connectivity based on graphs. We invite the interested reader to refer to [43] for in-depth studies of hierarchies in the general algebraic framework of the lattice of partial partitions.

A *partition* of a finite set V is a set $\mathbf{P}$ of non-empty disjoint subsets of V whose union is V (i.e., $\forall X, Y \in \mathbf{P}$, $X \cap Y = \emptyset$ if $X \neq Y$ and $\cup\{X \in \mathbf{P}\} = V$). Any element of a partition $\mathbf{P}$ of V is called a *region of* $\mathbf{P}$. If x is an element of V, there is a unique region of $\mathbf{P}$ that contains x; this unique region is denoted by $[\mathbf{P}]_x$. Given two partitions $\mathbf{P}$ and $\mathbf{P}'$ of a set V, we say that $\mathbf{P}'$ is a *refinement* of $\mathbf{P}$ if any region of $\mathbf{P}'$ is included in a region of $\mathbf{P}$. A *hierarchy (on V)* is a sequence $\mathcal{H} = (\mathbf{P}_0, \ldots, \mathbf{P}_\ell)$ of partitions of V such that $[\mathbf{P}]_{i-1}$ is a refinement of $[\mathbf{P}]_i$, for any $i \in \{1, \ldots, \ell\}$. If $\mathcal{H} = (\mathbf{P}_0, \ldots, \mathbf{P}_\ell)$ is a hierarchy, the integer ℓ is called the *depth of* $\mathcal{H}$. A hierarchy $\mathcal{H} = (\mathbf{P}_0, \ldots, \mathbf{P}_\ell)$ is called complete if $\mathbf{P}_\ell = \{V\}$ and if $\mathbf{P}_0$ contains every singleton of V (*i.e.*, $\mathbf{P}_0 = \{\{x\} \mid x \in V\}$). The hierarchies considered in this article are complete.

Figure 3 graphically represents a hierarchy $\mathcal{H} = (\mathbf{P}_0, \mathbf{P}_1, \mathbf{P}_2, \mathbf{P}_3)$ on a rectangular subset V of $\mathbb{Z}^2$ made of 9 dots. For instance, it can be seen that $\mathbf{P}_1$ is a refinement of $\mathbf{P}_2$ since any region of $\mathbf{P}_1$ is included in a region of $\mathbf{P}_2$. It can also be seen that the hierarchy is complete since $\mathbf{P}_0$ is made of singletons and $\mathbf{P}_3$ is made of a single region that contains all elements.

In this article, we consider connected regions, the connectivity being given by a graph. Therefore, we remind basic graph definitions before introducing connected partitions and hierarchies.

A *(undirected) graph* is a pair $G = (V, E)$, where V is a finite set and E is composed of unordered pairs of distinct elements in V, *i.e.*, E is a subset of $\{\{x, y\} \subseteq V \mid x \neq y\}$. Each element of V is called a *vertex or a point (of G)*, and each element of E is called an *edge* (of G). A *subgraph of* G is a graph $X = (V', E')$ such that V' is a subset of V, and E' is a subset of E. If X is a subgraph of G, we write $X \sqsubseteq G$. The vertex and edge sets of a graph X are denoted by $V(X)$ and $E(X)$, respectively.

Let G be a graph and let $(x_0, \ldots, x_\ell)$ be a sequence of vertices of G. The sequence $(x_0, \ldots, x_\ell)$ is a *path (in G) from* x_0 *to* x_ℓ if, for any i in $\{1, \ldots, \ell\}$, $\{x_{i-1}, x_i\}$ is an edge of G. The graph G *is connected* if, for any two vertices x and y of G, there exists a path from x to y. Let X be a subset of $V(G)$. The *graph induced by X (in G)* is the graph whose vertex set is X and whose edge set contains any edge of G which is made of two elements in X. If the graph induced by X is connected, we also say, for simplicity, that X *is connected (for G)*. The subset X of $V(G)$ is a *connected component of* G if it is connected for G and maximal for this property, *i.e.*, for any subset Y of $V(G)$, if Y is a connected superset of X, then

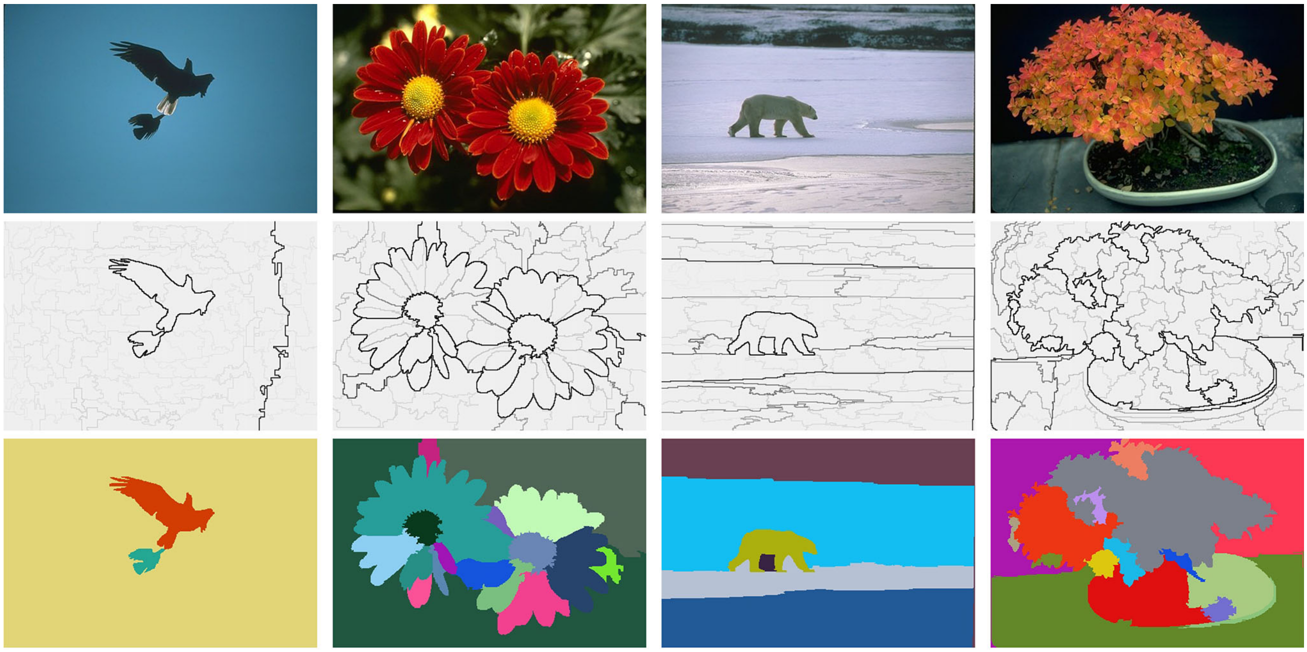

Fig. 2 Top row: some images from the Berkeley database [3]. Middle row: saliency maps according to [18] developed due to the framework of this article. Bottom row: segmentations extracted from the hierarchies with **a** 3, **b** 18, **c** 6 and **d** 16 regions

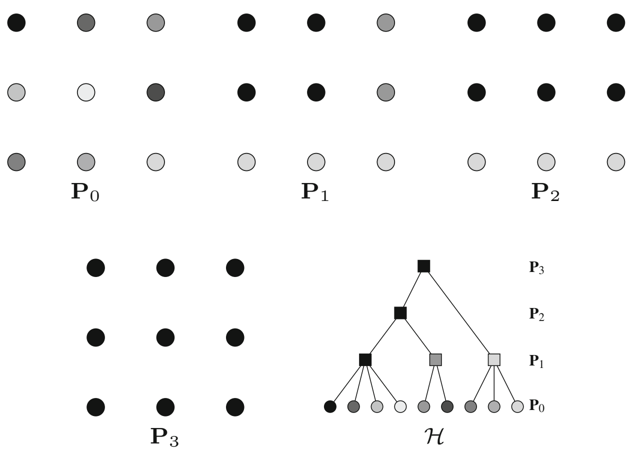

Fig. 3 Illustration of a hierarchy $\mathcal{H} = (\mathbf{P}_0, \mathbf{P}_1, \mathbf{P}_2, \mathbf{P}_3)$. For every partition, each region is represented by a gray level: two dots with the same gray level belong to the same region. The last subfigure represents the hierarchy $\mathcal{H}$ as a tree, often called a dendrogram, where the inclusion relation between the regions of the successive partitions is represented by line segments

we have $Y = X$. In the following, we denote by $\mathbf{C}(G)$ the set of all connected components of G. It is well known that this set $\mathbf{C}(G)$ of all connected components of G is a partition of $V(G)$. This partition is called the *(connected components) partition induced by* G. Thus, the set $[\mathbf{C}(G)]_x$ is the unique connected component of G that contains x.

Given a graph $G = (V, E)$, a *partition of* V *is connected (for* G*)* if every of its regions is connected and a *hierarchy on* V *is connected (for* G*)* if every of its partitions is connected.

For instance, the partitions presented in Fig. 3 are connected for the graph given in Fig. 4a. Therefore, the hierarchy $\mathcal{H}$ made of these partitions, which is depicted as a dendrogram in Fig. 3 (bottom-right subfigure), is also connected for the graph of Fig. 4a.

For image analysis applications, the graph G can be obtained as a pixel or a region adjacency graph: the vertex set of G is either the domain of the image to be processed or the set of regions of an initial partition of the image domain. In the latter case, the regions are often called the "image superpixels" (see, e.g., [1]). In both cases, two typical settings for the edge set of G can be considered: (1) the edges of G are obtained from an adjacency relation between the image pixels, such as the well-known 4- or 8-adjacency relations; and (2) the edges of G are obtained by considering, for each vertex x of G, the nearest neighbors of x for a distance in a features space onto which the vertices of G are mapped. A common feature space (see, e.g., [15]) is the one where each pixel of a color image is mapped to a vector in dimension 5 made of the two spatial coordinates and the three spectral values describing the color of the pixel.

3 Quasi-flat Zones

As established in the next sections, a connected hierarchy can be equivalently treated by means of an edge-weighted graph. We first recall in this section that the level sets of any edge-

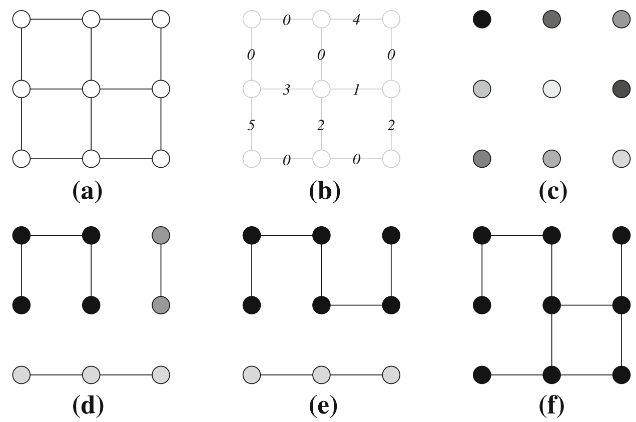

Fig. 4 Illustration of quasi-flat zones hierarchy. **a** A graph G; **b** a map w (numbers in black) that weights the edges of G (in gray); (**c**–**f**) the λ-level graph of G, with $\lambda = 0, 1, 2, 3$. The associated connected component partitions that make up the hierarchy of quasi-flat zones of G for w are depicted in Fig. 3

weighted graph induce a hierarchy of quasi-flat zones. This hierarchy is widely used in image processing [29,33,50] and is sometimes also referred to as the alpha-tree [51].

Let G be a graph, if w is a map from the edge set of G to the set $\mathbb{R}^+$ of positive real numbers, then the pair (G, w) is called an *(edge-)weighted graph*. If (G, w) is an edge-weighted graph, for any edge u of G, the value $w(u)$ is called the *weight of u (for w)*.

Important notation. In the sequel of this paper, we consider a weighted graph (G, w). To shorten the notations, the vertex and edge sets of G are denoted by V and E, respectively, instead of $V(G)$ and $E(G)$. Furthermore, we assume that the vertex set of G is connected. Without loss of generality, we also assume that the range of w is the set $\mathbb{E}$ of all integers from 0 to $|E| - 1$ (otherwise, one could always consider an increasing one-to-one correspondence from the set $\{w(u) \mid u \in E\}$ into the subset $\{0, \ldots, |\{w(u) \mid u \in E\}| - 1\}$ of $\mathbb{E}$). We also denote by $\mathbb{E}^\bullet$ the set $\mathbb{E} \cup \{|E|\}$.

Let X be a subgraph of G and let λ be an integer in $\mathbb{E}^\bullet$. The λ*-level set of X (for w)* is the set $w_\lambda(X)$ of all edges of X whose weight is less than λ:

$$w_\lambda(X) = \{u \in E(X) \mid w(u) < \lambda\}. \tag{1}$$

The λ*-level graph of X (for w)* is the subgraph $w_\lambda^V(X)$ of X whose edge set is the λ-level set of X and whose vertex set is the one of X:

$$w_\lambda^V(X) = (V(X), w_\lambda(X)). \tag{2}$$

The connected component partition $\mathbf{C}(w_\lambda^V(X))$ induced by the λ-level graph of X is called the λ*-level partition of X (for w)*.

For instance, let us consider the graph G depicted in Fig. 4a and the map w shown in Fig. 4b. The 0-, 1-, 2- and 3-level sets of G contain the edges depicted in Fig. 4c–f, respectively. The graphs depicted in these figures are the associated 0-, 1-, 2-, and 3-level graphs of G, and the associated 0-, 1-, 2-, and 3-level partitions are shown in Fig. 3.

Let X be a subgraph of G. If λ_1 and λ_2 are two elements in $\mathbb{E}^\bullet$ such that $\lambda_1 \leq \lambda_2$, it can be seen that any edge of the λ_1-level graph of X is also an edge of the λ_2-level graph of X. Thus, if two points are connected for the λ_1-level graph of X, then they are also connected for the λ_2-level graph of X. Therefore, any connected component of the λ_1-level graph of X is included in a connected component of the λ_2-level graph of X. In other words, the λ_1-level partition of X is a refinement of the λ_2-level partition of X. Hence, the sequence

$$\mathcal{QFZ}(X, w) = (\mathbf{C}(w_\lambda^V(X)) \mid \lambda \in \mathbb{E}^\bullet) \tag{3}$$

of all λ-level partitions of X is a hierarchy. This hierarchy $\mathcal{QFZ}(X, w)$ is called the *quasi-flat zones hierarchy of X (for w)*. It can be seen that this hierarchy is complete whenever X is connected.

For instance, the quasi-flat zone hierarchy of the graph G (Fig. 4a) for the map w (Fig. 4b) is the hierarchy of Fig. 3.

For image analysis applications, we often consider that the weight of an edge $u = \{x, y\}$ represents the dissimilarity of x and y. For instance, in the case where the vertices of G are the pixels of a grayscale image, the weight $w(u)$ can be the absolute difference of intensity between x and y. The setting of the graph (G, w) depends on the application context.

4 Correspondence Between Hierarchies and Saliency Maps

In the previous section, we have seen that any edge-weighted graph induces a connected hierarchy of partitions (called the quasi-flat zone hierarchy). In this section, we tackle the inverse problem:

(P_1) given a connected hierarchy $\mathcal{H}$, find a map w from E to $\mathbb{E}$ such that the quasi-flat zone hierarchy for w is precisely $\mathcal{H}$.

We will see that the saliency maps (defined by Eq. (5), below) provide a solution to this problem. The first notion of a saliency map was introduced in [36] for visualizing some hierarchies of watersheds. Then, it was notably used in [3,4] under the name of ultrametric contour maps. Connections with topological watersheds [6] were studied in [34], and morphological properties were investigated in [23] in the lattice of Jordan nets in the Euclidean 2D plane $\mathbb{R}^2$.

We start this section by defining the saliency map of $\mathcal{H}$. Then, we show that this notion provides a one-to-one correspondence (also known as a bijection) between saliency maps and hierarchies whose inverse correspondence is given by the hierarchy of quasi-flat zones. Finally, we deduce that the saliency map of $\mathcal{H}$ is a solution to problem (P_1).

Until now, we handled the regions of a partition. Let us now study their "dual" that represents "borders" between regions and that are called graph cuts or simply cuts. The notion of a cut will then be used to define the saliency maps.

Let $\mathbf{P}$ be a partition of V, the *cut of* $\mathbf{P}$ *(for G)*, denoted by $\phi_G(\mathbf{P})$, is the set of edges of G made of two vertices in different regions of $\mathbf{P}$:

$$\phi_G(\mathbf{P}) = \left\{ \{x, y\} \in E \mid [\mathbf{P}]_x \neq [\mathbf{P}]_y \right\}. \tag{4}$$

Let $\mathcal{H} = (\mathbf{P}_0, \ldots, \mathbf{P}_\ell)$ be a hierarchy on V. The *saliency map of* $\mathcal{H}$ is the map $\Phi_G(\mathcal{H})$ from E to $\{0, \ldots, \ell\}$ such that the weight of any edge u for $\Phi_G(\mathcal{H})$ is the maximum value λ for which u belongs to the cut of $\mathbf{P}_\lambda$:

$$\Phi_G(\mathcal{H})(u) = \max\{\lambda \in \{0, \ldots, \ell\} \mid u \in \phi_G(\mathbf{P}_\lambda)\}. \tag{5}$$

Dually, the weight of the edge $u = \{x, y\}$ for $\Phi_G(\mathcal{H})$ is directly related to the lowest index of a partition in the hierarchy $\mathcal{H}$ for which x and y belong to the same regions:

$$\Phi_G(\mathcal{H})(u) = \min\left\{\lambda \in \{0, \ldots, \ell\} \mid [\mathbf{P}_\lambda]_x = [\mathbf{P}_\lambda]_y\right\} - 1. \tag{6}$$

Observe from Eqs. (5 and (6) that the value of any edge for the saliency map $\Phi_G(\mathcal{H})$ of a hierarchy $\mathcal{H}$ is always nonnegative. Indeed, since the considered hierarchy $\mathcal{H}$ is complete, the partition $\mathbf{P}_0$ contains every singleton of V. Thus, we have $[\mathbf{P}_0]_x \neq [\mathbf{P}_0]_y$, for any two distinct vertices x and y in V. Thus, the lowest level λ such that $[\mathbf{P}_\lambda]_x = [\mathbf{P}_\lambda]_y$ is at least 1. Hence, the value of $\Phi_G(\mathcal{H})(u)$ is at least 0.

For instance, if we consider the graph G represented by the gray dots and line segments in Fig. 5a, the saliency map of the hierarchy $\mathcal{H}$ shown in Fig. 3 is the map shown with black numbers in Fig. 5a. When the 4-adjacency relation is used, a saliency map can be displayed as an image (Figs. 5e, f, 2(middle row)), which is useful for visualizing the associated hierarchy at a glance. Indeed, as assessed by the next theorem, a saliency map is equivalent to a hierarchy.

As illustrated in Fig. 5e, f, a visualization of a saliency map when the graph is given by the 4-adjacency relation can be obtained due to cubical complexes (also known as Khalimsky grids). Cubical complexes have been promoted in particular by V. Kovalevsky [24] in order to provide a sound topological basis for image analysis. In 2D, a cubical complex is a set of squares, unit line segments (represented by rectangles in Fig. 5e), and unit points (represented by dots Fig. 5e). Each vertex of the graph can be identified to a square of the complex. Then, each edge linking two vertices x and y can be identified to the segment corresponding to the common side of the two squares identified with x and y. The squares are given a null value, whereas the sides are given the value of the associated edges in the saliency map. Finally, for each point of the complex (i.e., the corners of the squares), the maximal value of a side containing it is kept. Thus, every element of the complex has a value. Hence, since the elements of the complex are aligned to a square matrix, the saliency map can be visualized as an image (see Fig. 5f).

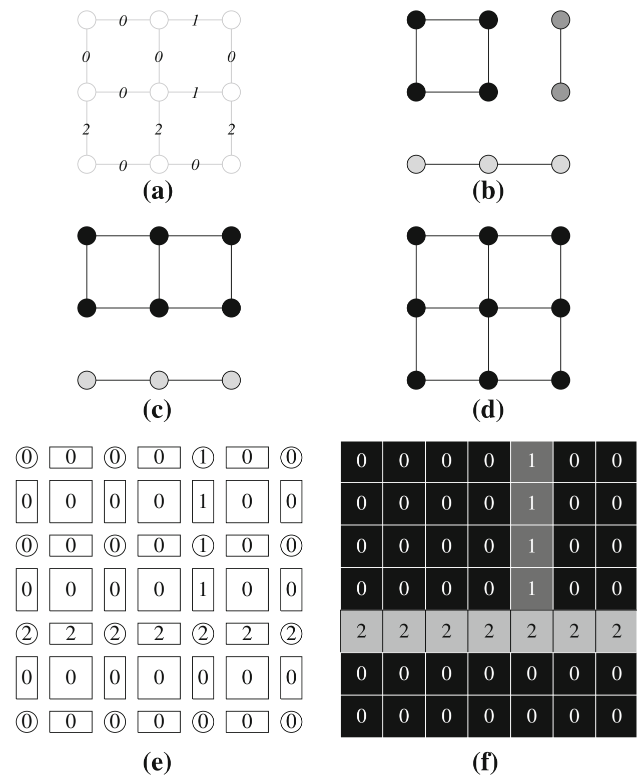

Fig. 5 Illustration of a saliency map. The map (depicted by black numbers) is the saliency map $s = \Phi_G(\mathcal{H})$ of the hierarchy $\mathcal{H}$ shown in Fig. 3 when we consider the graph G depicted in gray. **b**–**d**—the 1-, 2-, and 3-level graphs of G for s. The vertices are colored according to the associated 1-, 2-, and 3-level partitions of G: in each subfigure, two vertices belonging to a same connected components have the same gray level. **e** and **f** show possible image representations of a saliency map when one considers the 4-adjacency graph

We say that a map w from E to $\mathbb{E}$ is a *saliency map* if there exists a hierarchy $\mathcal{H}$ such that w is the saliency map of $\mathcal{H}$ (i.e., $w = \Phi_G(\mathcal{H})$).

If φ is a map from a set S_1 to a set S_2 and if φ^{-1} is a map from S_2 to S_1 such that the composition of φ^{-1} with φ is the identity, then we say that φ^{-1} is the inverse of φ.

The next theorem, whose proof is given in Appendix B, identifies the inverse of the map Φ_G and asserts that there is a bijection between the saliency maps and the connected hierarchies on V.

Theorem 1 *The map Φ_G is a one-to-one correspondence between the connected hierarchies on V of depth $|E|$ and the saliency maps (of range $\mathbb{E}$). The inverse Φ_G^{-1} of Φ_G associates with any saliency map w its quasi-flat zone hierarchy: $\Phi_G^{-1}(w) = \mathcal{QFZ}(G, w)$.*

Hence, as a consequence of this theorem, we have:

$$\mathcal{QFZ}(G, \Phi_G(\mathcal{H})) = \mathcal{H}, \tag{7}$$

which means that $\mathcal{H}$ is precisely the hierarchy of quasi-flat zones of G for its saliency map $\Phi_G(\mathcal{H})$. In other words, the saliency map of $\mathcal{H}$ is a solution to problem (P_1). For instance, if we consider the hierarchy $\mathcal{H}$ shown in Fig. 3, it can be observed that the quasi-flat zone hierarchy for $\Phi_G(\mathcal{H})$ (see Fig. 5) is indeed $\mathcal{H}$. From Theorem 1, we also deduce that, for any saliency map w, the relation

$$\Phi_G(\mathcal{QFZ}(G, w)) = w \tag{8}$$

holds true. In other words, a given saliency map w is precisely the saliency map of its quasi-flat zone hierarchy.

From this last relation, we can deduce that there are some maps that weight the edges of G and that are not saliency maps. Indeed, in general, a map w is not equal to the saliency map of its quasi-flat zone hierarchy, which means that Eq. (8) does not hold true for such a map. For instance, the map w in Fig. 4 is not equal to the saliency map of its quasi-flat zone hierarchy which is depicted in Fig. 5. Thus, the map w is not a saliency map. The next section studies a characterization of the maps that are saliency maps.

5 Characterization of Saliency Maps

Following the conclusion of the previous section, given a hierarchy $\mathcal{H}$, there might well exist distinct maps such that the quasi-flat zone hierarchies for these distinct maps are equal to $\mathcal{H}$. Hence, in order to select among such maps, the following problem can be considered:

(P_2) given a hierarchy $\mathcal{H}$, find a minimal map w such that the quasi-flat zone hierarchy for w is precisely $\mathcal{H}$.

The next theorem, whose proof is given in Appendix B, establishes that the saliency map of $\mathcal{H}$ is the unique solution to problem (P_2). Hence, the saliency maps are equivalently characterized by Eq. (5) (or by its dual version Eq. (6)) and as the solutions to (P_2).

Before stating Theorem 2, let us recall that, given two maps w and w' from E to $\mathbb{E}$, the map w' is less than or equal to w (written $w' \leq w$) if we have $w'(u) \leq w(u)$ for any $u \in E$.

Theorem 2 *Let $\mathcal{H}$ be a hierarchy and let w be a map from E to $\mathbb{E}$. The map w is the saliency map of $\mathcal{H}$ if and only if the two following statements hold true:*

1. *the quasi-flat zone hierarchy for w is $\mathcal{H}$; and*
2. *the map w is minimal for statement 1,* i.e., *for any map w' such that $w' \leq w$, if the quasi-flat zone hierarchy for w' is $\mathcal{H}$, then we have $w = w'$.*

Roughly speaking, we can say from Theorem 2 that the saliency map of a hierarchy $\mathcal{H}$ is the minimal characteristic map of $\mathcal{H}$. More formally, we can also deduce (see Lemma 5 in Appendix B) that $w \geq \Phi_G(\mathcal{H})$ whenever the quasi-flat zones of w is $\mathcal{H}$.

Given an edge-weighted graph (G, w), it is sometimes interesting to consider the saliency map of its quasi-flat zone hierarchy. This saliency map is simply called the *saliency map of w* (in G) and is denoted by $\Psi_G(w)$:

$$\Psi_G(w) = \Phi_G(\mathcal{QFZ}(G, w)). \tag{9}$$

Hence Ψ_G is an operator acting on the maps weighting the edges of G. As established by the following property whose proof is provided in Appendix C, this operator is a morphological opening.

Property 3 *1. The operator Ψ_G is idempotent: $\Psi_G(\Psi_G(w)) = \Psi_G(w)$;*
2. the operator Ψ_G is anti-extensive: $\Psi_G(w) \leq w$; and
3. the operator Ψ_G is increasing: for any map w' that weights the edges of G, if $w \geq w'$, then we have $\Psi_G(w) \geq \Psi_G(w')$.

Similar operators, settled in different frameworks, are studied under several names: ultrametric watershed [34], class opening [22], ultrametric opening [25] or subdominant ultrametric [37] when the complete graph is considered. When the considered graph G is complete, it is known in classification (see, e.g., [37]) that this operator is linked to the minimum spanning tree of (G, w). The next section proposes a generalization of this link.

6 Minimum Spanning Trees

Two distinct maps that weight the edges of the same graph (see, e.g., the maps of Figs. 4b, 5a) can induce the same hierarchy of quasi-flat zones. Therefore, in this case, one can guess that some of the edge weights do not convey any useful information with respect to the associated quasi-flat zones hierarchy. More generally, in order to represent a hierarchy by a simple edge-weighted graph (i.e., easier to handle in certain cases) with a low level of redundancy, it is interesting to consider the following problem:

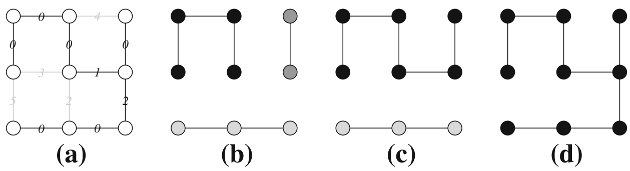

Fig. 6 Illustration of a minimum spanning tree and of its quasi-flat zone hierarchy. **a** A minimum spanning tree X (black edges and black circled vertices) of the weighted graph of Fig. 4b; (**b**–**d**) the 1-, 2-, and 3-level graphs of X. The vertices are colored according to the associated 1-, 2-, and 3-level partitions of X: in each subfigure, two vertices belonging to the same connected components have the same color

(P_3) given an edge-weighted graph (G, w), find a minimal subgraph $X \sqsubseteq G$ such that the quasi-flat zone hierarchies of G and of X are the same.

The main result of this section, namely Theorem 4, provides the set of all solutions to problem (P_3): the minimum spanning trees of (G, w). The proof of Theorem 4 is given in Appendix D. The minimum spanning tree problem is one of the most typical and well-known problems of combinatorial optimization (see [8]) and Theorem 4 provides, as far as we know, a new characterization of minimum spanning trees based on the quasi-flat zone hierarchies as used in image processing.

Let X be a subgraph of G. The weight of X with respect to w is the sum of the weights of all the edges in $E(X)$. The subgraph X is a *minimum spanning tree (MST) of (G,w)* if:

1. X is connected; and
2. $V(X) = V$; and
3. the weight of X is less than or equal to the weight of any subgraph Y of G satisfying (1) and (2) (*i.e.*, Y is a connected subgraph of G whose vertex set is V).

For instance, a MST of the graph shown in Fig. 4b is presented in Fig. 6a.

Theorem 4 *A subgraph X of G is a MST of (G, w) if and only if the two following statements hold true:*

1. *the quasi-flat zone hierarchies of X and of G are the same; and*
2. *the graph X is minimal for statement 1,* i.e.*, for any subgraph Y of X, if the quasi-flat zone hierarchy of Y for w is the one of G for w, then we have $Y = X$.*

Theorem 4 (statement 1) indicates that the quasi-flat zone hierarchy of a graph and of its MSTs is identical. Note that statement 1 appeared in [13], but Theorem 4 completes the result of [13]. Indeed, Theorem 4 indicates that there is no proper subgraph of a MST that induces the same quasi-flat zone hierarchy as the initial weighted graph. Thus, a MST of the initial graph is a solution to problem (P_3), providing a minimal graph representation of the quasi-flat zone hierarchy of (G, w). More remarkably, the converse is also true: a minimal representation of the quasi-flat zones hierarchy of an edge-weighted graph in the sense of (P_3) is necessarily a MST of the original graph. To the best of our knowledge, this result has not been stated before.

Furthermore, the correspondence between saliency maps and hierarchies (Theorem 1) allows us to extend Theorem 4 to the case where a hierarchy $\mathcal{H}$ is given instead of a weight map w. Hence, minimum spanning trees allow for characterizing spatially and functionally minimal representations of any connected hierarchy. The interested reader can refer to Appendix E for a precise definition of these notions, a proof of this statement, and some interesting complementary properties related to hierarchies and minimum spanning trees.

For instance, the level sets, level graphs and level partitions of the MST X (Fig. 6a) of the weighted graph (G, w) (Fig. 4) are depicted in Fig. 6b–d. It can be observed that the level partitions of X are indeed the same as those of G. Thus, the quasi-flat zone hierarchies of X and G are the same.

7 Saliency Map Algorithms

In this section, we study algorithms for computing the saliency map of a hierarchy and for computing the saliency map of a map (i.e., for computing the result of the opening Ψ_G). We start by considering a naive approach before providing efficient (linear-time) algorithms.

Using Eq. (5) straightforwardly, to obtain the saliency map $\Phi_G(\mathcal{H})$ of a hierarchy $\mathcal{H} = (\mathbf{P}_0, \ldots, \mathbf{P}_\ell)$, one can proceed in two steps:

(i) for every level λ of the hierarchy, compute the cut $\phi_G(\mathbf{P}_\lambda)$ of the partition $\mathbf{P}_\lambda$ at level λ; and
(ii) for every edge u of the graph G, set the value of $\Phi_G(\mathcal{H})(u)$ to the maximum level λ such that u belongs to $\phi_G(\mathbf{P}_\lambda)$.

In order to perform step i), a naive approach consists in deciding for each level λ and for each edge u of G whether u belongs to the cut $\phi_G(\mathbf{P}_\lambda)$ or not. For performing step ii), one can check for every edge u of the graph and for every level λ if u belongs to $\phi_G(\mathbf{P}_\lambda)$ and set the value $\Phi_G(\mathcal{H})(u)$ to the maximum value such that this property holds true. Thus, since the hierarchy contains $\ell + 1$ levels, the time complexity of this naive saliency map algorithm is then at least $O(\ell \times |E|)$. Note that we can have a hierarchy of depth $\ell = |V|$ where any two levels are distinct. The time complexity of the naive algorithm is then $O(|V| \times |E|)$. In the next paragraphs, we present a linear-time ($O(|V| + |E|)$) algorithm for computing the saliency map of a hierarchy and a quasi-linear-time algorithm to compute the saliency map of a map. To this end, we consider the dual characterization Eq. (6) of a saliency map.

Given a hierarchy $\mathcal{H}$, Eq. (6) states that the weight of an edge linking x and y for the saliency map of $\mathcal{H}$ is associated with the lowest index of a partition for which x and y belongs to the same region. When a hierarchy $\mathcal{H}$ is stored as a tree data structure, such as , e.g., the dendrogram of Fig. 3, this index can be obtained by finding the index of the least common ancestor of $\{x\}$ and $\{y\}$ in the tree. The problem of finding the least common ancestor of two nodes of a tree is notably studied in [5]. In particular, after preprocessing the tree, finding the least common ancestor of any two nodes can be done in constant time. Thus, an efficient algorithm for computing a saliency map consists of a preprocessing of the tree-based representation of the hierarchy followed by the computation of the saliency map value of each edge. Algorithm 1, given below, provides a precise description of this process. The functions *LCAPreprocess* and *LCA* called in Algorithm 1 correspond to the preprocessing of the tree and to the least common ancestor computation as described in [5]. The preprocessing step runs in linear time with respect to the number of nodes of the considered tree. The tree-based representation of a hierarchy on V is made of at most $2|V|-1$ nodes since a hierarchy on V contains at most $2|V|-1$ distinct regions: $|V|$ singletons and $|V|-1$ regions built from merging two regions of lower levels (see, e.g., [35]). Thus, the preprocessing step runs in $O(|V|)$ time complexity. The main loop consists of repeating constant time operations for each edge of the graph. Thus, it runs in $O(|E|)$ time complexity. Hence, the overall time complexity is $O(|V|+|E|)$. Compared to the naive approach, the proposed strategy allows us to reduce the time complexity for computing a saliency map from quadratic $O(|V|\times|E|)$ to linear $O(|V|+|E|)$.

Algorithm 1: Saliency map.

Data: A connected graph $G=(V,E)$, the tree-based representation T of a hierarchy $\mathcal{H}$ on V, and an array $level$ that maps to every node of T its height (which is also the level at which the corresponding region first appears in the hierarchy).

Result: The saliency map $S=\Phi_G(\mathcal{H})$ of the hierarchy $\mathcal{H}$.

```
LCAPreprocess(T);
foreach edge {x, y} in E do
    S[{x, y}] := level[LCA(T, {x}, {y})] -1;
```

Following the definition of the opening Ψ_G given in Eq. (9, in order to compute the saliency map $\Psi_G(w)$ of a given map w, one can proceed in two steps:

(i) build the quasi-flat zone hierarchy $\mathcal{H}=\mathcal{QFZ}(G,w)$ of G for w; and
(ii) compute the saliency map $\Psi_G(w)=\Phi_G(\mathcal{H})$.

Step i) can be performed with the quasi-linear-time algorithm shown in [35] provided that the graph edges are either already sorted or can be sorted in linear time, and step ii) can be performed in linear time as proposed in the previous paragraph. Thus, the overall time complexity of this algorithm is quasi-linear with respect to the size $|E|+|V|$ of the graph G, provided that the graph edges are either already sorted or can be sorted in linear time.

As far as we know, the algorithm presented in this section is the simplest algorithm for computing a saliency map. It is also the most efficient from both memory and execution-time points of view. An implementation in C of this algorithm is available at http://perso.esiee.fr/~dpt-it/sm.

Note that the algorithm sketched in [34], based on [9], for computing the saliency map of a given map w has the same complexity as the algorithm proposed above. However, the algorithm of [34] is more complicated since it requires to compute the topological watershed of the map. This involves a component tree (a data structure which is more complicated than the quasi-flat zone hierarchy in the sense of [13]), a structure for computing least common ancestors (which is also needed by the above algorithm), and a hierarchical queue [9] (which is not needed by the above algorithm).

8 Illustrations

In this section, we show with several practical examples how one can take advantage of having several representations of a same hierarchy. The two first illustrations present algorithms (and their results) to build interesting hierarchies of image segmentations. These algorithms rely on the links between hierarchies, MST, and saliency maps shown in this article. The third illustration considers saliency maps in order to design operations on hierarchies. Then, the last illustration shows that hierarchical image segmentation methods can be used for the hierarchical classification of non-image data. More precisely, in Sect. 8.1, the framework of hierarchical minimum spanning forests and watersheds is recalled and illustrated on images and three-dimensional meshes. Then, an algorithm to compute these hierarchies is sketched. In Sect. 8.2, we briefly present (more details are provided in [19]) how the proposed framework can be used to hierarchicalize some well-known image segmentation methods which are originally not hierarchical. In Sect. 8.3, saliency maps are used to efficiently combine hierarchies that feature different aspects of a same image. Finally, in Sect. 8.4, we show that hierarchical watersheds can be used to perform a hierarchical classification of non-image data. In particular, the Knuth Miles dataset (i.e., a set of 128 American cities with demographic and position information) is analyzed.

8.1 Hierarchical Minimum Spanning Forests and Watersheds

Minimum spanning forests can be used for marker-based segmentation [10]. Given an edge-weighted graph over the set of points to be studied (e.g., the pixels of an image) and a subset of points that mark the objects of interest, the problem is to find a spanning forest of minimum total weight such that each connected component is rooted in (i.e., contains exactly) one marker. The segmentation is then obtained as the connected components partition of the minimum spanning forest. The resulting segmentation is therefore optimal in the sense of minimum spanning forests. If the markers are ranked by importance, it is possible to obtain a series of nested MSF such that the k-th MSF is rooted in the k-most important markers according to the ranking. Thus, one can obtain a series of nested partitions, hence a hierarchy of partitions as defined in this article, where every partition is optimal. These hierarchies are studied in [11,13,35,41].

A usual choice in morphology is to consider the regional minima of the weight map as markers. Indeed, in this case, minimum spanning forest partitions are watershed segmentations defined by the drop of water principle [10]. The minima are often ranked according to regional attributes such as extinction values [53]. Extinction values can be computed from the component tree [46] of the weight map or directly from its quasi-flat zone hierarchy. Typical attributes are related to the area of the regions, their depth (also called dynamics) or their volume. The resulting hierarchies of partitions are called hierarchical watersheds [11,28,36]. Figure 7 displays hierarchical watersheds of three images. For each image, two hierarchies are computed: For the first one, the minima are ranked with an area attribute, however, for the second one, they are ranked by a dynamics attribute. Figure 8 shows the application of the same method for the segmentation of the surface of a 3D object represented as a mesh. The vertices of the considered graph are the triangles of the mesh and two vertices are linked by an edge if the corresponding triangles share a common side. The edges are weighted with a curvature function.

In order to compute a hierarchical watershed, a key idea of the algorithms in [11,35] is to compute a weight map whose quasi-flat zone hierarchy is the desired hierarchical watershed segmentation. This allows the time complexity to be reduced compared to a direct computation of the hierarchy. Therefore, the theoretical results of this article constitute a necessary basis to build and to justify the algorithms presented in the aforementioned articles. As far as we know, the study of this basis was lacking before the present article.

Let us briefly sketched the main steps of the algorithm presented in [13,35].

1. Given the edge-weighted graph (G, w), the first step consists of computing a binary partition tree by altitude ordering, denoted by BPTAO for short in the following. This structure is simply the hierarchy of partitions of V obtained during Kruskal's minimum spanning tree algorithm (see, e.g., [8]). We initially consider a partition into singletons. Then, when an edge is selected by Kruskal's algorithm, we build the next level of the hierarchy by merging the largest regions containing the vertices of the selected edge. In terms of tree, the newly created region is a new node of the BPTAO, which becomes the parent of the two nodes associated to the merged regions. At the end of the algorithm, the obtained BPTAO is a tree whose non-leaf nodes correspond to the edges of the minimum spanning tree T produced by Kruskal's algorithm. It has been shown that, if needed, the quasi-flat zones can be straightforwardly recovered from this BPTAO. At this step, we take advantage of the link between MSTs and quasi-flat zones established by Theorem 4.
2. From this BPTAO, the minima of the weight map are identified and regional attributes as well as extinction values of the minima can be computed. For instance, computing area attribute requires only to traverse the BPTAO once from the leaves to the root and a second traversal of the tree, from the root to the leaves, allows extinction values to be obtained.
3. Once extinction values of the minima are obtained, they can be extended to all nodes of the tree: the extinction of a non-leaf node being the highest extinction value of its descendants. These values can be computed by traversing the tree once more from the leaves to the root. At steps 2 and 3, we only work on the direct tree-based representation of the initial hierarchy.
4. Then, we set the persistence of each non-leaf node to be the minimum of the extinction of its two children. Thus, we end up with one persistence for each non-leaf node of the BPTAO. Since BPTAO non-leaf nodes correspond to the edges of the minimum spanning tree, we end up with one persistence value for each edge of the minimum spanning tree. In other words, we have produced a new weight map p (by persistence values) for the edges of the minimum spanning tree T.
5. The hierarchical watershed is simply the quasi-flat zones hierarchy of T for the map p. At steps 4 and 5, the new hierarchy of watersheds is built by first considering its saliency map (step 4) before explicitly computing the hierarchy (step 5). Hence, at these steps, we take advantage of the links between saliency maps and hierarchies established by Theorems 1 and 2.

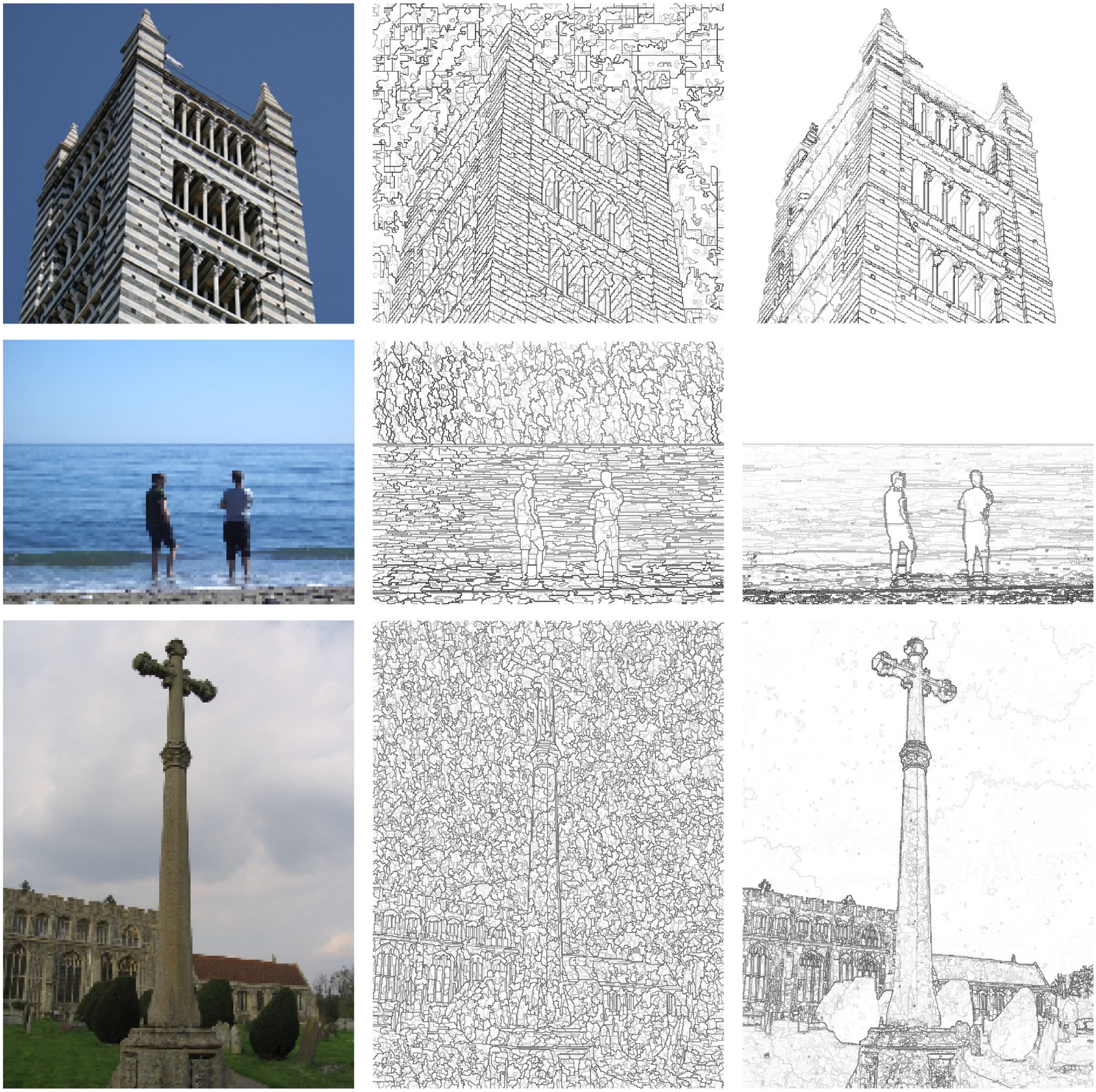

Fig. 7 First column: three color images; second and third columns: hierarchies of watersheds (saliency maps) driven by area attribute and by dynamics attribute, respectively

8.2 Hierarchizing Graph-Based Image Segmentation Algorithms Relying on a Region Dissimilarity: The Case of the Felzenszwalb–Huttenlocher Method

In the applicative companion manuscript [19], a generic algorithm that builds a new kind of hierarchy of image segmentations is proposed. The main idea of this algorithm consists of transforming a first hierarchy into a second one obtained by hierarchically grouping the regions of the first one according to a given dissimilarity measure, called an observation scale, between regions. The hierarchies considered by this method are all connected. They can therefore be handled, as established by the framework of this article, as dendrograms, saliency maps or weighted graphs. Hence, instead of explicitly transforming hierarchies, our algorithm transforms a first weighted graph into a second one. More precisely, it "re-weights" (*i.e.*, produces a new weight function for) a MST associated with the first hierarchy. The new weights are obtained by considering the edges of the MST in increasing order of the weights associated with the first hier-

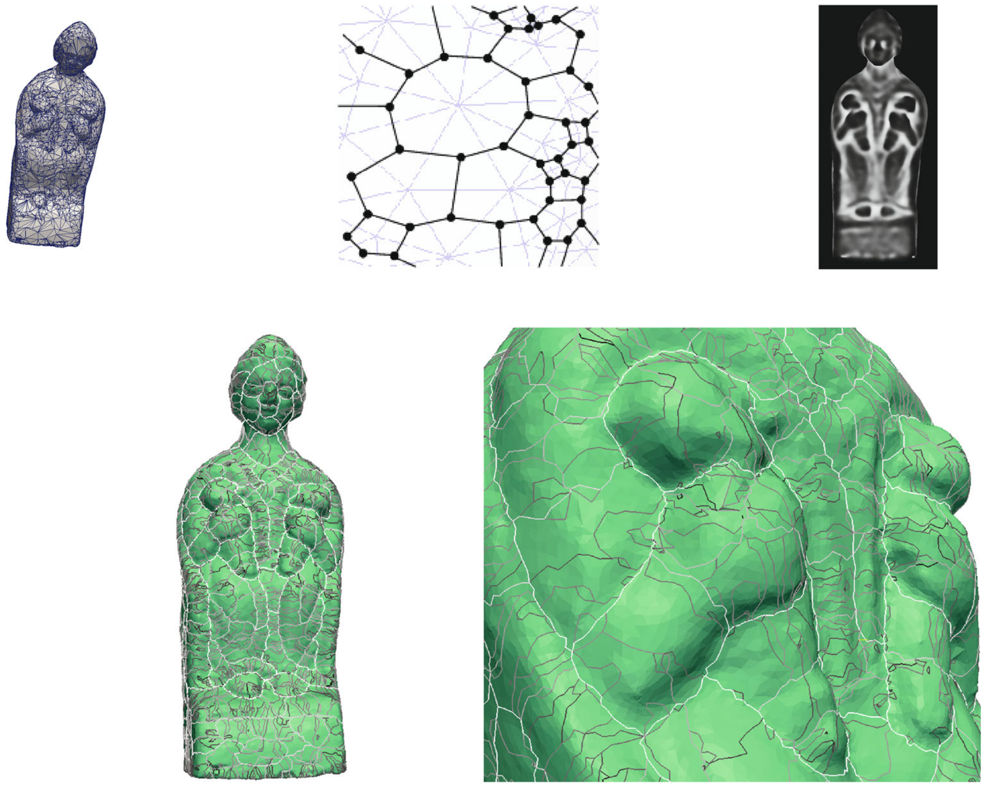

Fig. 8 Illustration of the segmentation of the surface of a 3D object. First row: a triangular mesh, a crop on its associated dual graph, and its pseudo-inverse curvature. Second row: a saliency map representing a hierarchical segmentation of the surface. A framework for the indexing and retrieval of ancient artwork 3D models, using shape descriptors adapted to the surface regions of the segmentations, is detailed in [42]. The mesh is provided by the French Museum Center for Research and Restoration (C2RMF, Le Louvre, Paris)

archy. The new weights are computed based on dissimilarity measures between regions.

Despite the appearance, the segmentation method proposed in [15] is not hierarchical (see counter examples of the hierarchical properties of [15] in [19]). Thus, we use the generic algorithm described above to produce a hierarchical segmentation based on the observation scale measure proposed by [15]. In [19], we show that the hierarchical method compares favorably to its non-hierarchical counterpart. Figure 2 presents some saliency maps obtained with the hierarchical version of the method.

8.3 Combinations of Hierarchies

One difficulty in the design of many segmentation methods relies on combining different kinds of measures that are not necessarily homogeneous (e.g., the Mumford and Shah functional integrates photometric and boundary lengths measures). The same difficulty can occur with hierarchical segmentations, where different methods can capture distinct properties. With the hierarchical method presented in Sect. 8.1, the use of different attributes leads to hierarchies featuring different aspects of the image. For instance, with the area attribute, at the highest levels of the hierarchy, small regions vanish, but low contrasted regions can remain. A high level of the area based hierarchies of Fig. 7 is represented in the first column of Fig. 9. On the other hand, with the dynamics attribute, the highest levels only contain contrasted regions but very small regions may remain. The second column of Fig. 9 presents a high level of each of the dynamics based hierarchy shown in Fig. 7. Attributes combining contrast and area can be designed, but such attributes would probably not be increasing. Attributes that are not increasing are known to be difficult to handle [46,52] and to lead to hierarchies lacking some important stability properties related to morphological filtering (see Theorem 11 in [11] for a link between morphological filtering and hierarchical watersheds). Another approach, which we investigate in this section, consists of combining hierarchies. To this end, we work on saliency maps instead of on the direct representation of the hierarchy. This approach was pioneered in [14] in the framework of graphs and with illustration in image segmentation. It was also investigated in [22] in the framework of Jordan nets in the Euclidean 2D plane $\mathbb{R}^2$, with applications to fusion of ground truths. In this section, we explicit this later approach in the framework of graphs, which allows, in

Fig. 9 First and second columns: one level of the hierarchies depicted in Fig. 7; third column: one level of the hierarchies depicted in Fig. 10. First (resp. second, and third) row: the partitions contain 500 (resp. 75, and 250) regions

particular, for processing images of arbitrary dimension, and we provide an efficient quasi-linear algorithm for combining hierarchies by infimum, by supremum, and by average.

8.3.1 Combination by Infimum and Supremum

In order to investigate the combinations of hierarchies by infimum and supremum, we first equip hierarchies with a lattice structure (see [30,43]).

If a partition $\mathbf{P}$ is a refinement of a partition $\mathbf{P}'$, we say that $\mathbf{P}$ is *finer than* $\mathbf{P}'$ and that $\mathbf{P}'$ is *coarser than* $\mathbf{P}$. The set of all partitions of V, together with the relation "is coarser than," is a lattice. The *infimum (resp. supremum) of two partitions* is the coarsest (resp. finest) partition which is finer (resp. coarser) than the two original partitions. We can extend the order relation "is coarser than" on partitions to the hierarchies of a given depth: a hierarchy is *coarser* than another if, at every level, the partition of the first hierarchy is coarser than the partition of the second hierarchy. With this setting, the *infimum (resp. supremum) of two hierarchies* is given by considering, at every level, the infimum (resp. supremum) of the partitions of the two hierarchies.

Based on the definition, to compute the infimum and the supremum of two hierarchies, one needs to compute the infimum and supremum of two partitions for every level of the hierarchy, which cannot be done efficiently in a direct manner. However, computation becomes efficient when saliency maps are considered. Indeed, the infimum $\mathcal{H}_1 \curlywedge \mathcal{H}_2$ and supremum $\mathcal{H}_1 \curlyvee \mathcal{H}_2$ of two hierarchies $\mathcal{H}_1$ and $\mathcal{H}_2$ are given by the quasi-flat zone hierarchy of the supremum and infimum, respectively, of the saliency maps of $\mathcal{H}_1$ and of $\mathcal{H}_2$:

$$\mathcal{H}_1 \curlywedge \mathcal{H}_2 = \mathcal{QFZ}(G, \Phi_G(\mathcal{H}_1) \vee \Phi_G(\mathcal{H}_2)); \tag{10}$$

and

$$\mathcal{H}_1 \curlyvee \mathcal{H}_2 = \mathcal{QFZ}(G, \Phi_G(\mathcal{H}_1) \wedge \Phi_G(\mathcal{H}_2)), \tag{11}$$

where for every edge u in E we have:

$$[\Phi_G(\mathcal{H}_1) \vee \Phi_G(\mathcal{H}_2)](u) = \min\{\Phi_G(\mathcal{H}_1)(u), \Phi_G(\mathcal{H}_2)(u)\}; \tag{12}$$

and

$$[\Phi_G(\mathcal{H}_1) \wedge \Phi_G(\mathcal{H}_2)](u) = \max\{\Phi_G(\mathcal{H}_1)(u), \Phi_G(\mathcal{H}_2)(u)\}. \tag{13}$$

Hence, to compute the infimum or supremum of $\mathcal{H}_1$ and $\mathcal{H}_2$, we need to compute two saliency maps, the edge-wise maximum or minimum of the two saliency maps and the quasi-flat zone hierarchy of the resulting map. Using the algorithms introduced in Sect. 7, the first and last steps can be done in linear and quasi-linear time with respect to the size of G, whereas the edge-wise maximum and minimum of two functions can also be done straightforwardly in linear time with respect to the number of edges. Hence, the infimum or supremum of two hierarchies can be obtained in quasi-linear time with respect to the size of the graph.

It can be seen that the saliency map of the infimum of $\mathcal{H}_1$ and $\mathcal{H}_2$ is simply the supremum of $\Phi_G(\mathcal{H}_1)$ and $\Phi_G(\mathcal{H}_2)$. On the other hand, the saliency map of the supremum of $\mathcal{H}_1$ and $\mathcal{H}_2$ is not the infimum of $\Phi_G(\mathcal{H}_1)$ and $\Phi_G(\mathcal{H}_2)$, but it is the saliency map of $\Phi_G(\mathcal{H}_1) \wedge \Phi_G(\mathcal{H}_2)$, namely $\Psi_G(\Phi_G(\mathcal{H}_1) \wedge \Phi_G(\mathcal{H}_2))$.

In practice, the combination of two hierarchies by infimum does not lead to interesting results. For instance, the combination of the area and dynamics hierarchies shown in Fig. 7 leads to a hierarchy featuring the drawbacks of both initial hierarchies: At high levels of the resulting hierarchies, some small regions as well as some uncontrasted ones can be found. In order to obtain a hierarchy whose high level contains only large and contrasted regions, combination by supremum can be considered. However, in the next section, we see that, following a similar approach, hierarchies can be combined by averaging saliency maps. On the tested images, the best results (visually) are obtained by this last technique.

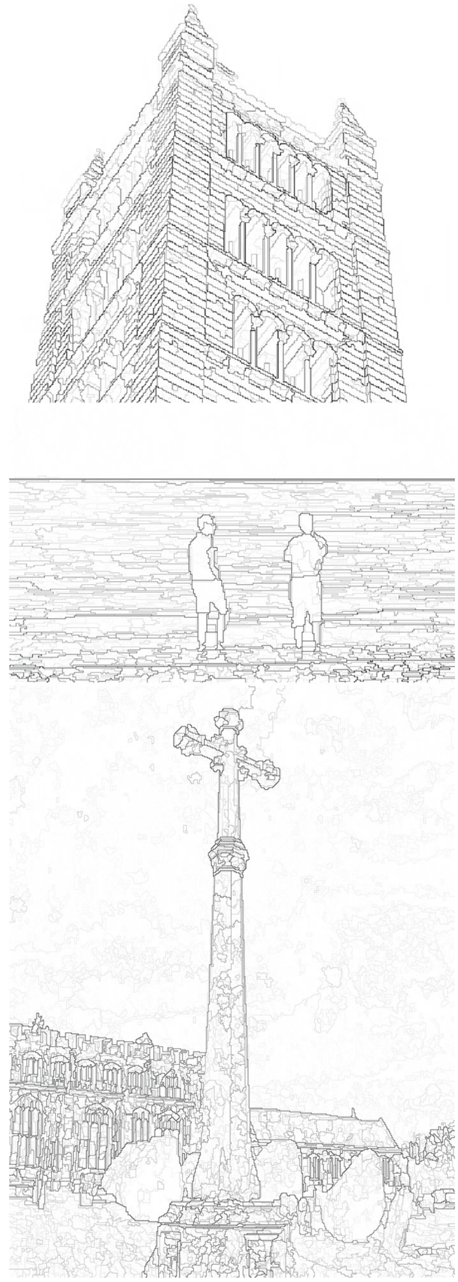

Fig. 10 Hierarchies of partitions (depicted as saliency maps) obtained from the images of Fig. 7 (first column). Each hierarchy is the combination by average of the hierarchical watersheds by area attribute (second column of Fig. 7) and by dynamics attribute (third column of Fig. 7) obtained from the images of Fig. 7 (first column)

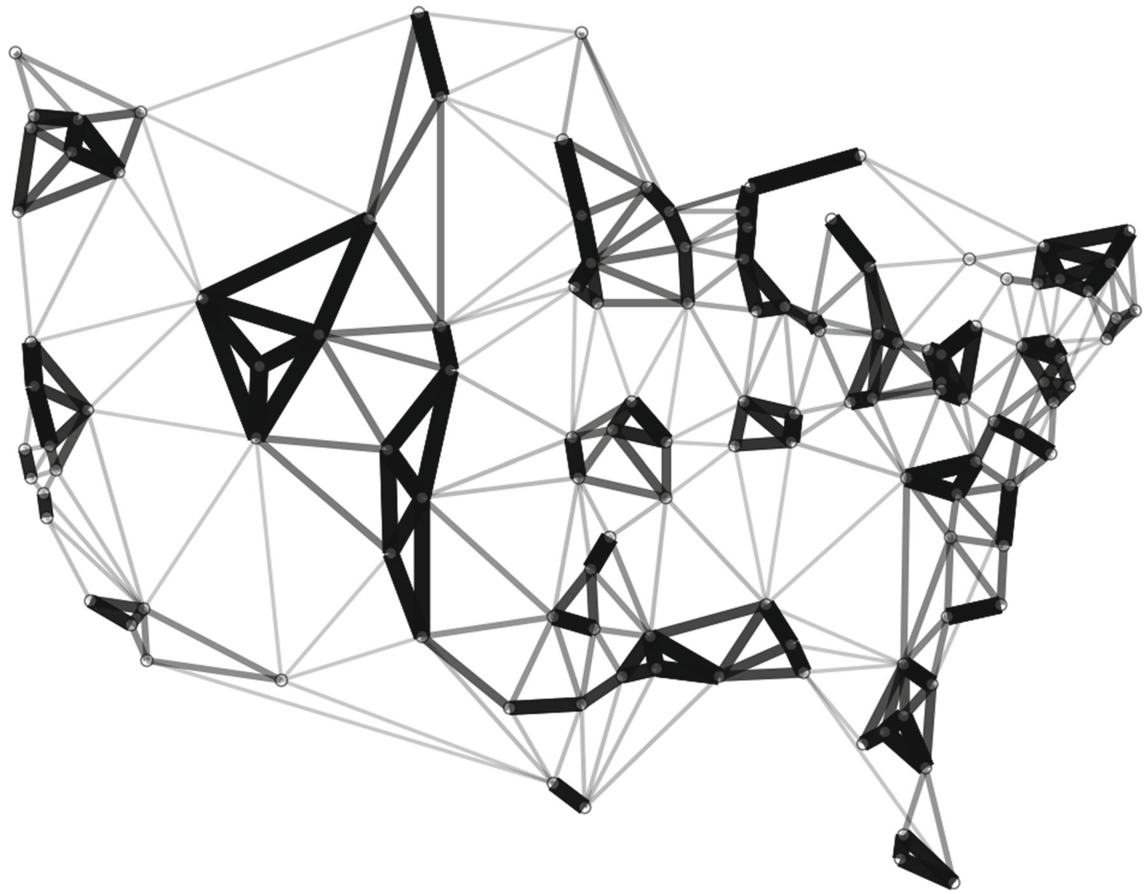

Fig. 11 Saliency map of a hierarchical watershed (driven by population attribute) on the Knuth Miles dataset (i.e., 128 representative US cities with positions and populations). Each vertex is a city and two neighboring cities are linked by an edge if they share an edge in the Voronoi diagram of the cities. The width and gray level of an edge is the inverse of its weight in the associated saliency map

8.3.2 Combination by Average

We define the *combination by average of two hierarchies* $\mathcal{H}_1$ *and* $\mathcal{H}_2$, denoted by $\mathcal{AVG}(\mathcal{H}_1, \mathcal{H}_2)$, as the quasi-flat zone hierarchy of the average of the saliency maps of $\mathcal{H}_1$ and of $\mathcal{H}_2$:

$$\mathcal{AVG}(\mathcal{H}_1, \mathcal{H}_2) = \mathcal{QFZ}(G, avg(\Phi_G(\mathcal{H}_1), \Phi_G(\mathcal{H}_2))), \tag{14}$$

where for every edge u in E we have:

$$[avg(\Phi_G(\mathcal{H}_1), \Phi_G(\mathcal{H}_2))](u) = \frac{1}{2}(\Phi_G(\mathcal{H}_1)(u) + \Phi_G(\mathcal{H}_2)(u)). \tag{15}$$

Figure 10 presents, for each image of Fig. 7, the saliency maps of the combination by average of the hierarchies obtained with the area and depth attributes (second and third columns of Fig. 7). One level of each of these hierarchies is represented in the third row of Fig. 9.

In fact, any combination of the saliency maps of two (or more) hierarchies can be used before a possible extraction of a quasi-flat zone hierarchy. More precisely, in Eq. (14), one can replace the function avg by any function from $\mathcal{F} \times \mathcal{F}$ into $\mathcal{F}$, where $\mathcal{F}$ denotes the set of all maps weighting the edges of G. Exploring and determining precisely the combinations that lead to the best practical results is beyond the scope of this article and is left for future work.

8.4 Geographic Data Processing

We finish this section by an illustration where the proposed framework is used for geographic data analysis. The goal is to illustrate on a small example that the catchment areas of cities could be studied with a hierarchical method coming from the field of image analysis, namely hierarchical watersheds.

We consider the Knuth Miles dataset [32] that contains the position and population of 128 US cities. From this, we build a graph where each vertex is a city and where

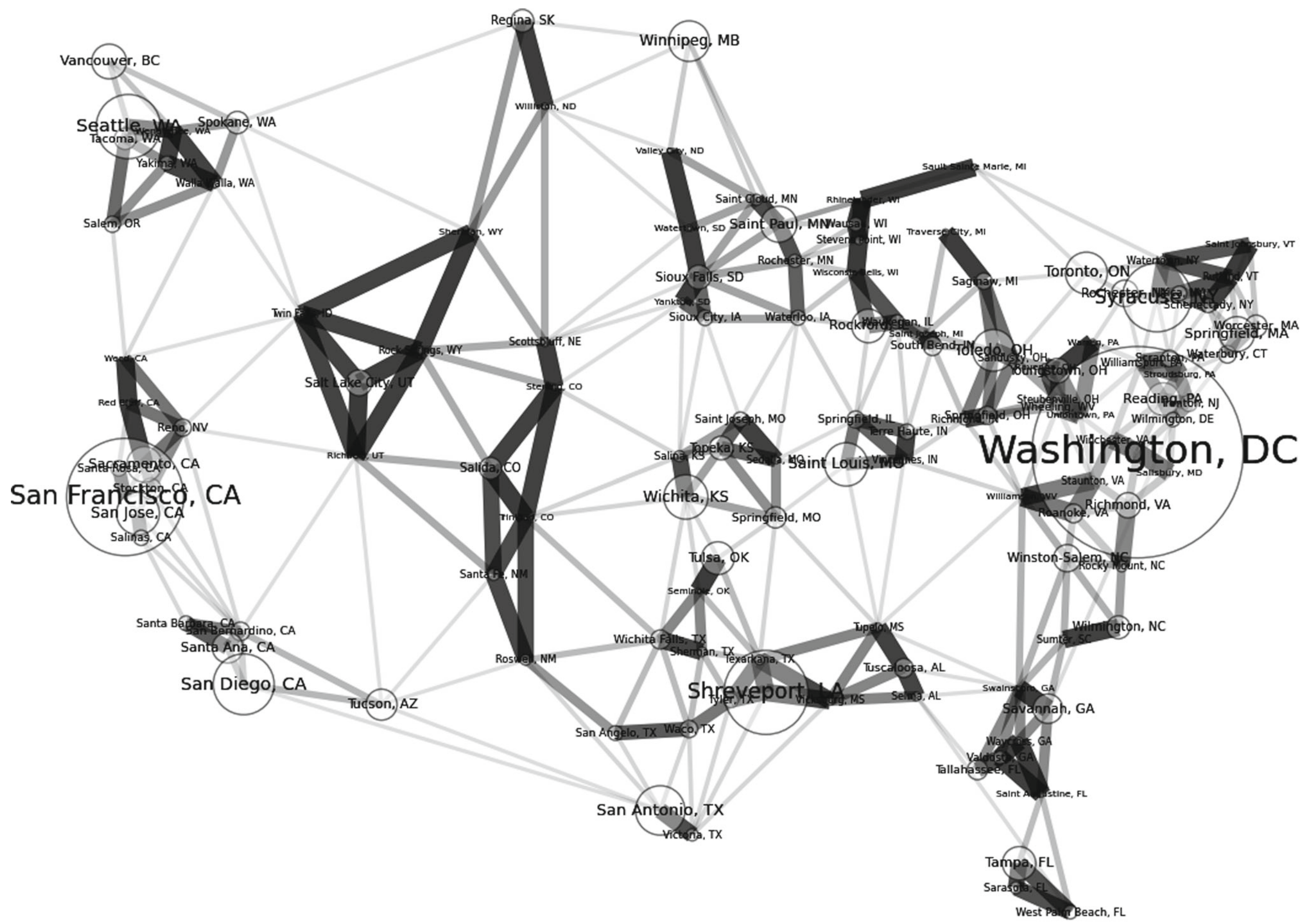

Fig. 12 Same as Fig. 11 but the size of the vertices and of the labels are given by the extinction value (for the population attribute) of the cities

two neighboring cities are connected by a weighted edge. The weight of an edge is the Euclidean distance between two inter-connected cities. The edges are obtained from the Voronoi diagram of the cities. Two cities are said to be neighbors if the corresponding regions of the Voronoi diagram are adjacent. Then, a morphological hierarchical analysis is performed as described in Sect. 8.1 with the area attribute. However, in this experiment, the area of a vertex is given by the population of the corresponding city and the area of a region is then the sum of the populations of the cities that belong to this region. The morphological analysis provides:

1. a hierarchy of optimal partitions of the cities such that at a given level of the hierarchy there are only regions with more than a certain number of inhabitants (see the saliency map in Fig. 11 and a projection of the saliency map on a geographic map in Fig. 13); and
2. a ranking (see Fig. 14) of the cities by extinction values. In our case, the extinction value of a city can be thought of as the number of inhabitants of its catchment area, meaning that, following our model, if the extinction value of a city is n, then at most n inhabitants can be attracted by this city. Thus, if you consider the level of the hierarchy corresponding to n inhabitants, each region contains more than n inhabitants and contains exactly one city with an extinction value greater than n. The extinction values of the cities are graphically presented on a geographic map in Fig. 12.

As far as we know, apart from image segmentation applications, representations of hierarchical clusterings by saliency maps are not usual. In the field of information visualization, a related approach consists of spatializing the data by projecting hierarchical clusters on an artificial topographical map that represents some relations between the data (see, e.g., [48,49]). However, in data analysis, hierarchical clusterings are most often represented by dendrograms. Such dendrograms (see, e.g., Fig. 15) become difficult to read when the numbers of clusters and of levels exceed a few dozens. Concerning the hierarchical clustering of the 128 cities of Knuth Miles dataset, the dendrograms would be unreadable. Someone used to read dendrograms may take some time to get used to saliency maps because

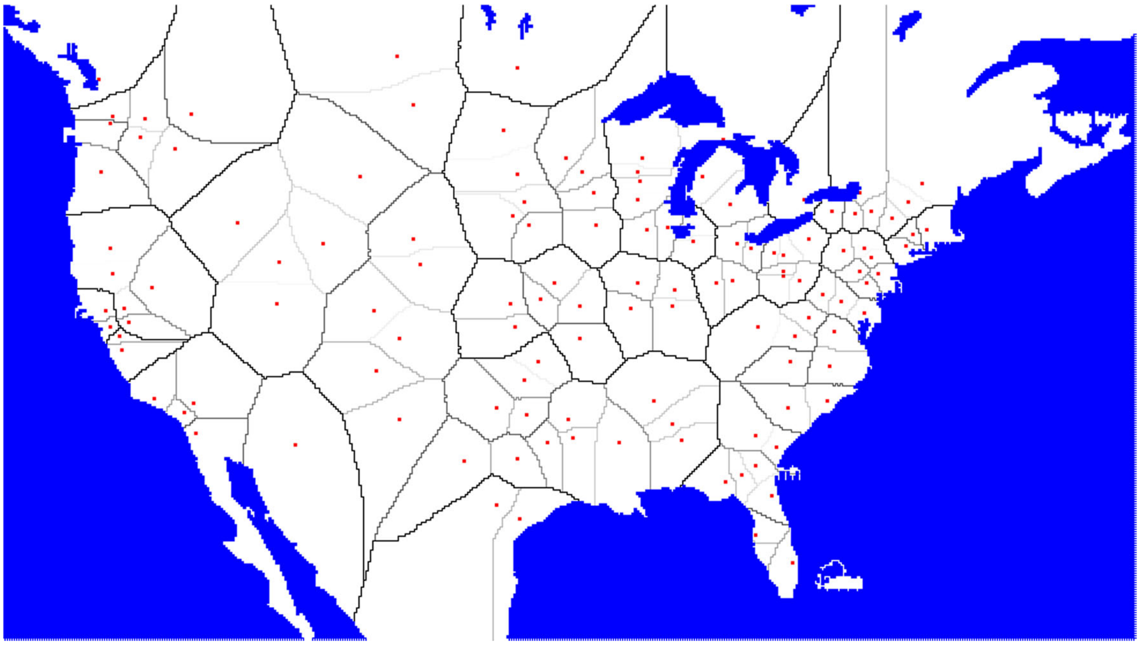

Fig. 13 Saliency map of a hierarchical watershed (driven by population attribute) on the Knuth Miles dataset (i.e., 128 representative US cities with positions and populations). The saliency weights are projected on the edges of the Voronoi diagram of the cities

Fig. 14 Ranking (from top to bottom and left to right) of the Knuth Miles dataset cities according to catchment basins size (i.e., extinction value of the cities by population attribute)

City	Pop.	B. S.	City	Pop.	B. S.	City	Pop.	B. S.
Washington, DC	638	15280	South Bend, IN	109	118	Watertown, NY	27	27
San Francisco, CA	678	4692	San Bernardino, CA	118	118	Selma, AL	26	26
Shreveport, LA	205	2424	Springfield, OH	72	113	Steubenville, OH	26	26
Syracuse, NY	170	1620	Waterbury, CT	103	103	Twin Falls, ID	26	26
Seattle, WA	493	1416	Waco, TX	101	101	Walla Walla, WA	25	25
San Diego, CA	875	1271	Reno, NV	100	100	Vicksburg, MS	25	25
San Antonio, TX	786	836	Springfield, IL	100	100	Scottsbluff, NE	14	25
Wichita, KS	279	664	Scranton, PA	88	93	Sumter, SC	24	24
Saint Louis, MO	453	634	Saginaw, MI	77	92	Tupelo, MS	23	23
San Jose, CA	629	629	Trenton, NJ	92	92	Stevens Point, WI	22	22
Toronto, ON	599	599	Salem, OR	89	89	Staunton, VA	21	21
Toledo, OH	354	594	Santa Rosa, CA	83	83	Winchester, VA	20	20
Winnipeg, MB	564	564	Sioux City, IA	82	82	Sedalia, MO	20	20
Saint Paul, MN	270	444	Terre Haute, IN	61	81	Vincennes, IN	20	20
Sacramento, CA	275	424	Salinas, CA	80	80	Waycross, GA	19	19
Springfield, MA	152	416	Saint Joseph, MO	76	76	Rock Springs, WY	19	19
Vancouver, BC	414	414	Waterloo, IA	75	75	Rutland, VT	18	18
Rockford, IL	139	387	Utica, NY	75	75	Wenatchee, WA	17	17
Tampa, FL	271	382	Santa Barbara, CA	74	74	Salisbury, MD	16	16
Tulsa, OK	360	368	San Angelo, TX	73	73	Watertown, SD	15	15
Reading, PA	78	366	Wilmington, DE	70	70	Sheridan, WY	15	15
Tucson, AZ	330	330	Tyler, TX	70	70	Traverse City, MI	15	15
Santa Ana, CA	204	322	Wheeling, WV	43	69	Sault Sainte Marie, MI	14	14
Savannah, GA	141	296	Schenectady, NY	67	67	Uniontown, PA	14	14
Winston-Salem, NC	131	252	Waukegan, IL	67	67	Williston, ND	13	13
Rochester, NY	241	241	Yakima, WA	49	66	Yankton, SD	12	12
Salt Lake City, UT	163	228	Wausau, WI	32	63	Warren, PA	12	12
Richmond, VA	219	219	West Palm Beach, FL	63	63	Saint Augustine, FL	11	11
Youngstown, OH	115	209	Rochester, MN	57	57	Sterling, CO	11	11
Sioux Falls, SD	81	197	Valdosta, GA	37	56	Ravenna, OH	11	11
Topeka, KS	115	191	Victoria, TX	50	50	Red Bluff, CA	9	11
Wilmington, NC	139	180	Sarasota, FL	48	48	Trinidad, CO	9	9
Regina, SK	162	175	Santa Fe, NM	48	48	Saint Joseph, MI	9	9
Spokane, WA	171	171	Saint Cloud, MN	42	42	Seminole, OK	8	8
Salida, CO	44	165	Salina, KS	41	41	Saint Johnsbury, VT	7	7
Worcester, MA	161	161	Richmond, IN	41	41	Rhinelander, WI	7	7
Tacoma, WA	158	158	Rocky Mount, NC	41	41	Swainsboro, GA	7	7
Springfield, MO	133	153	Roswell, NM	39	39	Valley City, ND	7	7
Stockton, CA	149	149	Williamsport, PA	33	33	Williamson, WV	5	5
Tallahassee, FL	81	137	Sandusky, OH	31	31	Stroudsburg, PA	5	5
Wichita Falls, TX	94	124	Texarkana, TX	31	31	Richfield, UT	5	5
Tuscaloosa, AL	75	124	Sherman, TX	30	30	Wisconsin Dells, WI	2	2
Roanoke, VA	100	121				Weed, CA	2	2

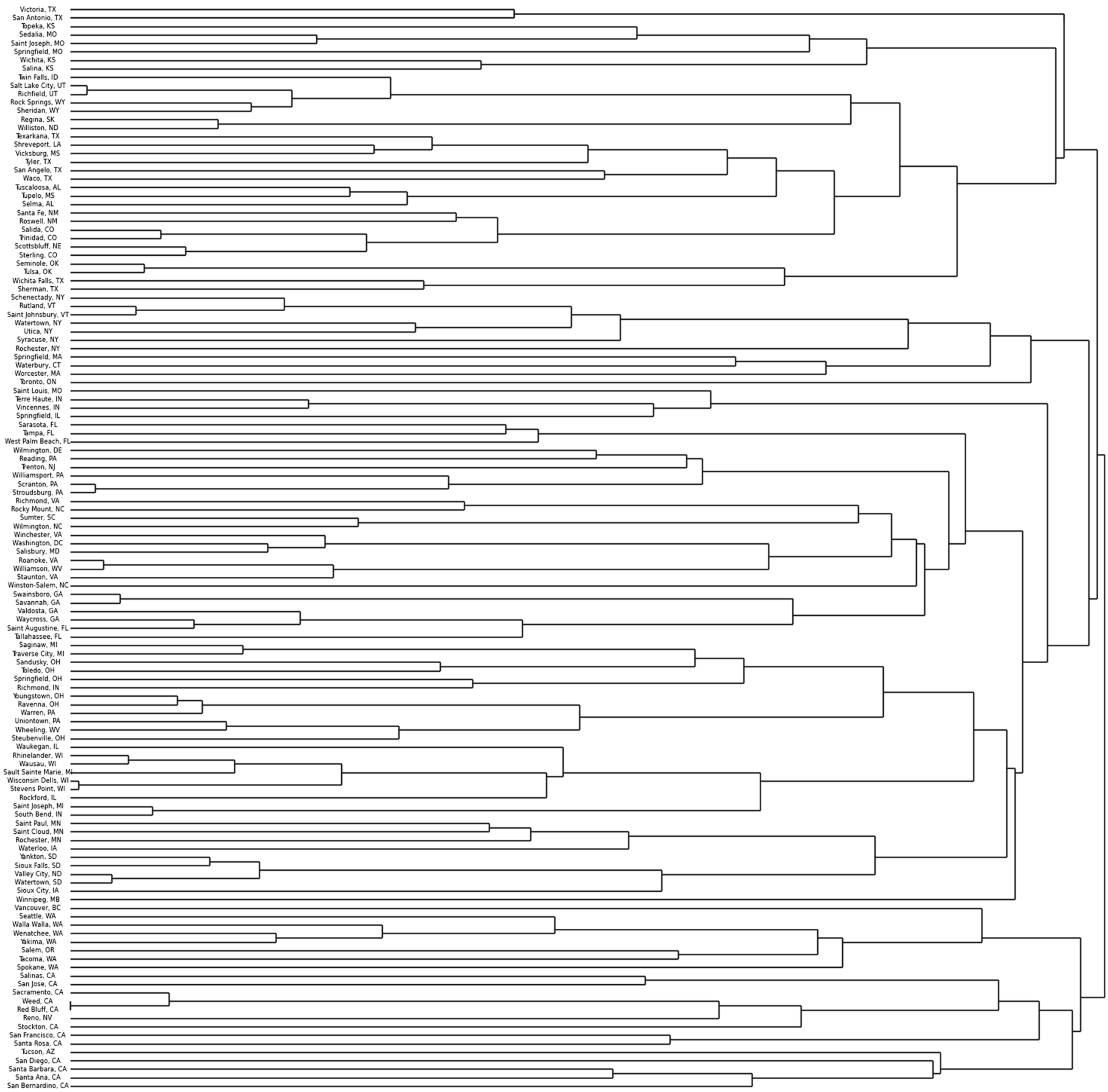

Fig. 15 Dendrogram representing the hierarchy obtained by morphological analysis of the Knuth Miles dataset

the information is shown in a dual way. Indeed, roughly speaking, one may say that dendrograms display hierarchies by classes, whereas saliency maps depict their borders. From our experience, after a few minutes and some simple explanations, saliency maps have been found to be pretty readable. Therefore, saliency maps could constitute an interesting tool for information visualization. Assessing precisely how they can be used on larger databases for which the points are not paired to 2D positions is beyond scope of this paper but is an interesting perspective for future work.

9 Conclusions

In this article, we study three representations for a hierarchy of partitions: direct representation (i.e., dendrogram), saliency map, and minimum spanning trees. We show a new bijection between hierarchies and saliency maps, and we characterize the saliency map of a hierarchy and the minimum spanning trees of a graph as minimal elements preserving quasi-flat zones. In practice, these results allow us to indifferently handle a hierarchy by a dendrogram (the direct tree structure given by the hierarchy), by a saliency map, or by an edge-weighted tree. These representations make up a toolkit for the design of hierarchical (segmentation) methods where one can choose the most convenient representation or the one that leads to the most efficient implementation for a given particular operation. We show that the proposed tools are at the basis of very efficient hierarchical watershed algorithms and are powerful to design new hierarchical segmentation methods arising from the combination of several hierarchies. Furthermore, the results of this paper were used in [18] to provide a framework for hierarchizing a certain class of non-hierarchical methods. We study in particular a hierarchicalization of [15]. In [19], we provide more details on this method as well as a precise practical evaluation of the gain of the hierarchical method with respect to its non-hierarchical counterpart. On the tested cases (Grabcut [44], Weizmann [2], and Berkeley [27] datasets), the hierarchical method is always as good as and is sometimes better than the non-hierarchical one. Furthermore, the hierarchical method provides all the scales in one run, which is about 2.5 faster than obtaining 50 segmentations, with 50 distinct parameter values, with the non-hierarchical method.

Another important aspect of the present work is to underline and to precise the close link that exists between classification and hierarchical image segmentation. Whereas classification methods were used as image segmentation tools for a long time, our results incite us to use hierarchical methods initially designed for image segmentation for processing non-image data. We showed preliminary results of the use of hierarchical watersheds and saliency maps for analyzing and visualizing a dataset of cities. With the emergence of the so-called "big-data," exploring the analysis of large databases with morphological tools seems a promising direction for future research.

Acknowledgements The research leading to these results has received funding from the French Agence Nationale de la Recherche (Contract ANR-2010-BLAN-0205-03), the French Committee for the Evaluation of Academic and Scientific Cooperation with Brazil, and the Brazilian Federal Agency of Support and Evaluation of Postgraduate Education (Program CAPES/PVE: Grant 064965/2014-01, and Program CAPES/COFECUB: Grant 592/08).

Appendix A: Proof of Theorem 1

Proof In order to establish Theorem 1, we will prove that the two following statements hold true:

(1) for any connected hierarchy $\mathcal{H} = (\mathbf{P}_0, \dots, \mathbf{P}_\ell)$, we have: $\Phi_G^{-1}(\Phi_G(\mathcal{H})) = \mathcal{QFZ}(G, \Phi_G(\mathcal{H})) = \mathcal{H}$; and
(2) for any saliency map w, we have $\Phi_G(\Phi_G^{-1}(w)) = \Phi_G(\mathcal{QFZ}(G, w)) = w$.

(1) Let $\mathcal{QFZ}(G, \Phi_G(\mathcal{H})) = (\mathbf{P}'_0, \dots, \mathbf{P}'_\ell)$. Since $\mathcal{H}$ and $\mathcal{QFZ}(G, \Phi_G(\mathcal{H}))$ are complete hierarchies, we have $\mathbf{P}_0 = \mathbf{P}'_0$. Thus, in order to complete the proof of (1), we will establish that $\mathbf{P}_\lambda = \mathbf{P}'_\lambda$, for any $\lambda \in \{1, \dots, \ell\}$. Let $\lambda \in \{1, \dots, \ell\}$ and let x and y be two points in V. The following statements are equivalent:

i $[\mathbf{P}'_i]_x = [\mathbf{P}'_i]_y$;
ii x and y belong to the same connected component of $\Phi_G(\mathcal{H})^V_\lambda(G)$ [by Eq. (3)];
iii there exists a path $\pi = (x = x_0, \dots, x_k = y)$ from x to y in the graph $\Phi_G(\mathcal{H})^V_\lambda(G)$;
iv there exists a path $\pi = (x = x_0, \dots, x_k = y)$ from x to y in the graph $(V, \{u \in E \mid \Phi_G(\mathcal{H})(u) < \lambda\}$ [by Eqs. (2) and (1)];
v there exists a path $\pi = (x = x_0, \dots, x_k = y)$ in G from x to y such that $\Phi_G(\mathcal{H})(\{x_{i-1}, x_i\}) < \lambda$, for any $i \in \{1, \dots, k\}$;
vi there exists a path $\pi = (x = x_0, \dots, x_k = y)$ in G from x to y such that $\max\left\{j \in \{0, \dots, \ell\} \mid [\mathbf{P}_j]_{x_{i-1}} \neq [\mathbf{P}_j]_{x_i}\right\} < \lambda$, for any $i \in \{1, \dots, k\}$ [by Eqs. (5) and (4)];
vii there exists a path $\pi = (x = x_0, \dots, x_k = y)$ in G from x to y such that $[\mathbf{P}_\lambda]_{x_{i-1}} = [\mathbf{P}_\lambda]_{x_i}$, for any $i \in \{1, \dots, k\}$;
viii $[\mathbf{P}_\lambda]_x = [\mathbf{P}_\lambda]_y$ (since $[\mathbf{P}]_\lambda$ is a connected partition for G).

Thus, since statements i. and viii. are equivalent, we deduce that $\mathbf{P}_\lambda = \mathbf{P}'_\lambda$, which completes the proof of statement (1).

(2) Let w be a saliency map. By the definition of a saliency map, there exists a hierarchy $\mathcal{H}$ such that $w = \Phi_G(\mathcal{H})$. By statement (1), we have $\mathcal{H} = \Phi_G^{-1}(\Phi_G(\mathcal{H}))$. Thus, we deduce that $w = \Phi_G(\Phi_G^{-1}(\Phi_G(\mathcal{H})))$. Then, since $w = \Phi_G(\mathcal{H})$, we have $w = \Phi_G(\Phi_G^{-1}(w))$. ◻

Appendix B: Proof of Theorem 2

In order to prove Theorem 2, we first established the following lemma.

Lemma 5 *For any map z from E to $\mathbb{E}$, the following inequality holds true:*

$$\Phi_G\left(\mathcal{QFZ}\left(G, z\right)\right) \leq z.$$

Proof Let $\mathcal{H} = \mathcal{QFZ}(G, z) = (\mathbf{P}_0, \dots, \mathbf{P}_\ell)$. For any $\lambda \in \{0, \dots \ell\}$, the partition $\mathbf{P}_\lambda$ is the connected component partition of the λ-level graph $z^V_\lambda(G)$ of G for z. By Eq. (2), we have $z^V_\lambda(G) = (V, z_\lambda(G))$, for any $\lambda \in \{0, \dots \ell\}$. Let $u = \{x, y\}$ be any edge in E. In order, to establish Lemma 5, it is sufficient to prove that $z(u) \geq \Phi_G(\mathcal{H})(u)$. For any $\lambda \in \{z(u) + 1, \dots, \ell\}$, the edge u belongs to $z_\lambda(G)$. Thus, for any $\lambda \in \{z(u)+1, \dots, \ell\}$, we have $[\mathbf{P}_\lambda]_x = [\mathbf{P}_\lambda]_y$. By Eq. (6), we deduce that $\min\{\lambda \in \{0, \dots, \ell\} \mid [\mathbf{P}_\lambda]_x =$

$[\mathbf{P}_\lambda]_y\} = \Phi_G(\mathcal{H})(u) + 1$. Thus, we have $z(u) \geq \Phi_G(\mathcal{H})(u)$. □

Proof of Theorem 2

1. Let us first prove the forward implication of Theorem 2. To this end, let $\mathcal{H} = (\mathbf{P}_0, \ldots, \mathbf{P}_\ell)$ and let us assume that w is the saliency map of $\mathcal{H}$ (i.e., $w = \Phi_G(\mathcal{H})$). Thus, we have:
   $\mathcal{QFZ}(G, w) = \mathcal{QFZ}(G, \Phi_G(\mathcal{H}))$.
   Hence, by Theorem 1 (see, in particular, Eq. (7) which follows straightforwardly from Theorem 1), we deduce that $\mathcal{QFZ}(G, w) = \mathcal{H}$, which establishes statement 1. Let z be any map from E to $\mathbb{E}$ such that $\mathcal{QFZ}(G, z) = \mathcal{H}$ and such that $z \leq w$. By Lemma 5, we have $\Phi_G(\mathcal{QFZ}(G, z)) \leq z$. Thus, since $\mathcal{QFZ}(G, z) = \mathcal{H}$, we deduce that $\Phi_G(\mathcal{H}) \leq z$. Hence, we have $w \leq z$. Therefore, we conclude that $w = z$, which establishes statement 2.
2. Let us now prove the backward implication of Theorem 2. To this end, let us suppose that the map w is such that: (1) the quasi-flat zone hierarchies for w is $\mathcal{H}$ (i.e., $\mathcal{QFZ}(G, w) = \mathcal{H}$); and (2) the map w is minimal for statement (1), *i.e.*, for any map w' such that $w' \leq w$, if the quasi-flat zone hierarchy for w' is $\mathcal{H}$, then we have $w = w'$. By Lemma 5, we deduce that $\Phi_G(\mathcal{QFZ}(G, w)) \leq w$. Thus, we have $\Phi_G(\mathcal{H}) \leq w$. By Theorem 2 (see, in particular, Eq. (7)), we have $\mathcal{QFZ}(G, \Phi_G(\mathcal{H})) = \mathcal{H}$. Thus, by definition of w (see in particular statement (2)), we deduce that $\Phi_G(\mathcal{H}) = w$. □

Appendix C: Proof of Property 3

Proof 1. By Eq. (9), we have:

$$\Psi_G(\Psi_G(w)) = \Phi_G(\mathcal{QFZ}(G, \Phi_G(\mathcal{QFZ}(G, w)))).$$

Hence, by Eq. (7), we deduce that:

$$\Psi_G(\Psi_G(w)) = \Phi_G(\mathcal{QFZ}(G, w)).$$

Thus, by Eq. (9), we conclude that:

$$\Psi_G(\Psi_G(w)) = \Psi_G(w).$$

2. Lemma 5.
3. Let w' be a map from E to $\mathbb{E}$ such that $w' \leq w$. Let $u = \{x, y\}$ be any edge in E, we are going to prove that $[\Psi_G(w')](u) \leq [\Psi_G(w)](u)$. By Eq. (9), we have $\Psi_G(w') = \Phi_G(\mathcal{QFZ}(G, w'))$ and $\Psi_G(w) = \Phi_G(\mathcal{QFZ}(G, w))$. Let $\mathcal{QFZ}(G, w') = (\mathbf{P}'_0, \ldots, \mathbf{P}'_{|E|})$ and $\mathcal{QFZ}(G, w) = (\mathbf{P}_0, \ldots, \mathbf{P}_{|E|})$. Let $k = [\Psi_G(w')](u)$. From Eq. (6), we deduce that $[\mathbf{P}_{k+1}]_x = [\mathbf{P}_{k+1}]_y$. By Eq. (3), we have $\mathbf{P}_{k+1} = \mathbf{C}(w^V_{k+1}(G))$. Hence, there exists a path $(x_0, \ldots, x_\ell)$ such that $x_0 = x$, $x_\ell = y$, and $w(\{x_{i-1}, x_i\}) < k + 1$ for any $i \in \{1, \ldots, \ell\}$. Since $w' \leq w$, we also have $w'(\{x_{i-1}, x_i\}) < k + 1$ for any $i \in \{1, \ldots, \ell\}$. Thus, we have $[\mathbf{P}'_{k+1}]_x = [\mathbf{P}'_{k+1}]_y$. Hence, by Eq. (6), we have $[\Phi_G(\mathcal{QFZ}(w'))](u) \leq k$. Thus, we have, $[\Psi_G(w')](u) \leq [\Psi_G(w)](u)$. □

Appendix D: Proof of Theorem 4

In order to establish the equivalence Theorem 4, we first prove the backward implication (Property 7) and then the forward implication (Property 8)

Before establishing Properties 7 and 8, let us state the following propositions which can be derived from classical properties of trees.

Let S be a subset of V and let $\{x, y\}$ be an edge of G. We say that $\{x, y\}$ is *outgoing from* S if we have $x \in S$ and $x \in V \setminus S$ (or $y \in S$ and $x \in V \setminus S$).

Lemma 6 *Let X be a connected subgraph of G. If, for any subset S of V, there is an edge u of X outgoing from S such that the weight of u is less than or equal to the weight of any edge of G outgoing from S, then, there exists a subgraph of X that is an MST of (G, w)*

Let X be a graph and let $\pi = (x_0, \ldots, x_k)$ be a path in X. We say that π is a *simple path* if for any two distinct i and j in $\{0, \ldots, k\}$, we have $x_i \neq x_j$. Let x and y be two vertices of X, there exists a path from x to y in X if and only if there is a simple path in X from x to y.

Property 7 *Let X be a MST of (G, w). Then, the two following statements hold true:*

1. *the quasi-flat zone hierarchies of X and of G are the same; and*
2. *the graph X is minimal for Theorem 4.1,* i.e., *for any subgraph Y of X, if the quasi-flat zones hierarchy of Y for w is the one of G for w, then we have $Y = X$.*

Proof Let $\mathcal{H} = (\mathbf{P}_0, \ldots, \mathbf{P}_\ell) =$ and $\mathcal{H}' = (\mathbf{P}'_0, \ldots, \mathbf{P}'_\ell)$ be the quasi-flat zone hierarchy of G and X, respectively. It can be seen that $\mathbf{P}_0 = \mathbf{P}'_0$ since $\mathcal{H}$ and $\mathcal{H}'$ are complete hierarchies. Let $\lambda \in \{1, \ldots, \ell\}$ and let x and y be two points of V. In order to complete the proof of Theorem 4.1, we are going to establish that:

(i) if $[\mathbf{P}'_\lambda]_x = [\mathbf{P}'_\lambda]_y$, then $[\mathbf{P}_\lambda]_x = [\mathbf{P}_\lambda]_y$;
(ii) if $[\mathbf{P}_\lambda]_x = [\mathbf{P}_\lambda]_y$, then $[\mathbf{P}'_\lambda]_x = [\mathbf{P}'_\lambda]_y$

In order to establish i), we assume that $\left[\mathbf{P}'_\lambda\right]_x = \left[\mathbf{P}'_\lambda\right]_y$ and we will prove that $[\mathbf{P}_\lambda]_x = [\mathbf{P}_\lambda]_y$. Since $\left[\mathbf{P}'_\lambda\right]_x = \left[\mathbf{P}'_\lambda\right]_y$, by definition of the quasi-flat zone hierarchy of X, there exists a path $\pi = (x_0, \ldots, x_k)$ in X such that $x_0 = x$, $x_k = y$, and $w(\{x_{i-1}, x_i\}) < \lambda$, for any $i \in \{1, \ldots, k\}$. Since X is a subgraph of G, the path π is also a path in G. Thus, the vertices x and y belong to the same connected component of the λ-level graph of G. Hence, we have $\left[\mathbf{P}'_\lambda\right]_x = \left[\mathbf{P}'_\lambda\right]_x$.

We now establish ii) by contradiction. Therefore, we assume that $[\mathbf{P}'_\lambda]_x \neq [\mathbf{P}'_\lambda]_y$ and we will prove that $[\mathbf{P}_\lambda]_x \neq [\mathbf{P}_\lambda]_y$. Since X is a spanning tree, there exists a simple path $\pi = (x_0, \ldots, x_k)$ such that $x_0 = x$ and $x_k = y$. As $[\mathbf{P}'_\lambda]_x \neq [\mathbf{P}'_\lambda]_y$, there exists an index $i \in \{1, \ldots, k\}$ such that $w(\{x_{i-1}, x_i\}) \geq \lambda$. Let i be the lowest index in $\{1, \ldots, k\}$ such that $w(\{x_{i-1}, x_i\}) \geq \lambda$. Let $X' = (V, E(X) \setminus \{\{x_{i-1}, x_i\}\})$ and let C be the connected component of X' that contains the vertex x. Observe that any edge u of G which is outgoing from C is such that $w(u) \geq w(\{x_{i-1}, x_i\})$ (otherwise the graph $(V, E(X') \cup \{w\})$ would be connected and of weight less than the weight of X, which is a contradiction with the fact that X is a MST of (G, w)). Observe also that the vertex y belongs to $V \setminus C$ (otherwise X' would be connected and of weight strictly less that the weight of X, which is a contradiction with the fact that X is a MST of (G, w)). Therefore, any path in G from x to y has an edge outgoing from C. Thus, any path in G from x to y has an edge of weight greater than or equal to λ. Hence, the vertices x and y belong to two distinct connected components of the λ-level graph of G and therefore, we have $[\mathbf{P}_\lambda]_x \neq [\mathbf{P}_\lambda]_y$.

Let us now prove the second proposition of Property 7. Let Y be a subgraph of X such that $Y \neq X$ and such that the quasi-flat zone hierarchy of Y for w is the one of G for w. Thus, we have $\mathbf{C}(w^V_\ell(Y)) = \mathbf{C}(w^V_\ell(G))$. By definition of (G, w), we have $\mathbf{C}(w^V_\ell(G)) = \{V\}$ where $\ell = |E|$. Therefore, we also have $\mathbf{C}(w^V_\ell(Y)) = \{V\}$. Hence, we deduce that $V(Y) = V$ and that Y is connected. Thus, we have $Y = X$, since X is a MST of (G, w). □

Property 8 *Let X be a subgraph of G such that*

(1) the quasi-flat zone hierarchies of X and of G are the same; and
(2) the graph X is minimal for (1), i.e., *for any subgraph Y of X, if the quasi-flat zone hierarchy of Y for w is the one of G for w, then we have $Y = X$.*

Then, the graph X is a MST of (G, w).

Proof (by contradiction) Let us assume that X is not a MST of (G, w). We distinguish three cases.

i. We first assume that X is not connected. Then, the $|E|$-level graph of X is not connected. Thus, the $|E|$-level partition of X is not trivial, which is a contradiction with the fact that quasi-flat zone hierarchies of X and of G are the same since G is connected and $\mathbb{E}$ is the range of w.
ii. We now assume that X is connected and that there exists a MST Y of (G, w) which is a proper subgraph of X. Then, by Property 7, the quasi-flat zone hierarchies of Y and of G are the same, which is a contradiction with (2).
iii. We finally assume that X is connected and that there is no subgraph of X which is a MST for w. By the contraposition of Lemma 6, we deduce that there is a subset S of V and an edge $v = \{x, y\}$ in $E \setminus E(X)$ outgoing from S and of weight less than the weight of any edge of X outgoing from S. Let $\lambda = w(v) + 1$. It can be seen that x and y belong to the same region of the λ-level partition of G. In order to complete the proof, we will show that x and y do not belong to the same λ-level partition of X, which constitutes a contradiction with statement *(1)*. To this end, we are going to show that there is no simple path (hence, from the observation above Property 7, no path) in the λ-level graph of X from x to y. Since any path in the λ-level graph of X is a path in X, it is sufficient to prove that any simple path $\pi = (x_0, \ldots, x_k)$ in X such that $x_0 = x$ and $x_k = y$ is not a path in the λ-level graph of X. Without loss of generality, let us assume that x belongs to S and that y belongs to $V \setminus S$. Thus, there is an index $i \in \{1, \ldots, k\}$ such that x_{i-1} belongs to S and x_i belongs to $V \setminus S$. Since π is a path in X, the edge $u = \{x_{i-1}, x_i\}$ belongs to $E(X)$. Therefore, u is an edge of X outgoing from S. Hence, by definition of v, the weight of u is greater than the weight v. Thus, the path π is not a path in the λ-level graph of X. □

Appendix E: Minimum Spanning Tree and Minimal Representation of a Hierarchy

In Sects. 5 and 6, we consider the problems of finding a minimal subgraph X and a minimal function w' such that the quasi-flat zone hierarchy of G for w, the quasi-flat zone hierarchy of G for w' and the quasi-flat zone hierarchy of X for w are the same. In this appendix section, we are interested in a similar problem given a connected hierarchy $\mathcal{H}$ instead of a weight map w. More precisely, we investigate the following problem:

(P_4) given a graph G and a connected hierarchy $\mathcal{H}$, find a minimal pair (X, w) such that the quasi-flat zone hierarchy of X for w is precisely $\mathcal{H}$.

Hence, the solutions to this problem can be considered as spatially and functionally minimal representations of the given hierarchy $\mathcal{H}$.

Before stating the main result of this section, let us deduce some interesting properties from Theorem 4. These properties are useful to prove the main result of this section, namely Theorem 12.

We recall that a tree is a graph that is connected and that cannot be "reduced" by edge removal while remaining connected. More formally, a connected graph X is a tree if, for any connected subgraph Y of X such that $V(Y) = V(X)$, we have $X = Y$.

Property 9 *Let $\mathcal{H}$ be a hierarchy such that $\mathcal{H} = \mathcal{QFZ}(G, w)$. If G is a tree, then we have $w = \Phi_G(\mathcal{H})$.*

Proof Let $u = \{x, y\}$ be any edge of G and let $\lambda = \Phi_G(\mathcal{H})(u)$. In order to establish Property 9, we will prove that $w(u) = \lambda$. Let $\mathcal{H} = (\mathbf{P}_0, \ldots, \mathbf{P}_\ell)$. By Lemma 5, we have $\Phi_G(\mathcal{H}) \leq w$. Hence, we have $\underline{\lambda \leq w(u)}$. Since G is a tree, the edge u appears in any path from x to y. By Eq. (6), we deduce that $\left[\mathbf{P}_{\lambda+1}\right]_x = \left[\mathbf{P}_{\lambda+1}\right]_y$. Thus, there is a path π from x to y in the graph $w^V_{\lambda+1}(G)$ and the edge u appears in π. Hence, from Eq. (2), we deduce that $u \in w_{\lambda+1}(G)$ and, from Eq. (1), we can affirm that $\underline{w(u) < \lambda + 1}$. Since the range of w is a subset of the integers, we deduce from the two underlined relations that $w(u) = \lambda$. □

Property 10 *If G is a tree, then we have $w = \Psi_G(w)$.*

Proof Let $\mathcal{H} = \mathcal{QFZ}(G, w)$. By Eq. (9), we have $\Psi_G(w) = \Phi_G(\mathcal{H})$. Hence, since G is a tree, by Property 9, we deduce that $\Psi_G(w) = w$. □

Property 11 *If X is a MST of (G, w), then for any edge u of X, we have $\Psi_G(w)(u) = w(u)$.*

Proof Let $\mathcal{H} = \mathcal{QFZ}(G, w)$. By Theorem 4, we also have $\mathcal{QFZ}(X, w) = \mathcal{H}$. Thus, by Eq. (6), for any $u \in E(X)$, we have $\Phi_G(\mathcal{H})(u) = \Phi_X(\mathcal{H})(u)$. Since X is a tree, by Property 9, we have $\Phi_X(\mathcal{H})(u) = w(u)$. Thus, for any $u \in E(X)$, we have $\Phi_G(\mathcal{H})(u) = w(u)$. Hence, by Eq. (9), for any $u \in E(X)$, we have $\Psi_G(w)(u) = w(u)$. □

Let $\mathcal{H}$ be a hierarchy, let X be a subgraph of G and let f be a map from $E(X)$ to $\mathbb{E}$. We say that (X, f) is a *representation of $\mathcal{H}$* if $\mathcal{H}$ is the quasi-flat zone hierarchy of X for f. A representation (X, f) of $\mathcal{H}$ is said to be *spatially minimal* whenever, for any representation (Y, f) of $\mathcal{H}$ such that $Y \sqsubseteq X$, we have $Y = X$; the representation (X, f) of $\mathcal{H}$ is said to be *functionally minimal* if for any representation (X, g) of $\mathcal{H}$ such $g \leq f$, we have $g = f$.

Due to Theorems 2 and 4, we are able to prove the following characterization of the spatially and functionally minimal representations of a hierarchy.

Theorem 12 *Let $\mathcal{H}$ be a hierarchy of depth $|E|$, let X be a subgraph of G, and let f be any map from $E(X)$ to $\mathbb{E}$. The pair (X, f) is a spatially and functionally minimal representation of $\mathcal{H}$ if and only if X is a minimum spanning tree of $(G, \Phi_G(\mathcal{H}))$ and $f(u) = \Phi_G(\mathcal{H})(u)$ for any $u \in E(X)$.*

Proof Let $g = \Phi_G(\mathcal{H})$. In order to establish Theorem 12, we will first prove the forward implication and then the backward one.

1. Let us assume that (X, f) is a spatially and functionally minimal representation of $\mathcal{H}$. Let Y be any MST of (X, f). Then, by Theorem 4, we have $\mathcal{QFZ}(Y, f) = \mathcal{QFZ}(X, f) = \mathcal{H}$. Since (X, f) is a minimal representation of $\mathcal{H}$, we deduce that $Y = X$. Hence, the graph X is a tree. Then, by Property 9, we have $f = \Phi_X(\mathcal{H})$. Thus, since $g = \Phi_G(\mathcal{H})$ and since $X \sqsubseteq G$, we deduce from Eq. (6) that, for any $u \in E(X)$, we have $f(u) = g(u)$. Furthermore, we then have $\mathcal{QFZ}(X, g) = \mathcal{QFZ}(X, f)$. By definition of g, we have $\mathcal{QFZ}(G, g) = \mathcal{H}$. Thus, since (X, f) is spatially minimal and since $\mathcal{QFZ}(X, g) = \mathcal{H}$, we deduce from Theorem 4 that X is a MST of (G, g).
2. Let us now assume that X is a minimum spanning tree of (G, g) and that $f(u) = g(u)$ for any $u \in E(X)$. By Eq. (7), we have $\mathcal{QFZ}(G, g) = \mathcal{H}$. By Theorem 4, we deduce that $\mathcal{QFZ}(X, g) = \mathcal{H}$ and that for any $Y \sqsubseteq X$ such that $\mathcal{QFZ}(Y, g) = \mathcal{H}$, we have $Y = X$. By definition of f, we can then also deduce that $\mathcal{QFZ}(X, f) = \mathcal{H}$ and that for any $Y \sqsubseteq X$ such that $\mathcal{QFZ}(Y, f) = \mathcal{H}$, we have $Y = X$. Thus, the pair (X, f) is a spatially minimal representation of $\mathcal{H}$. Furthermore, since X is a tree, by Property 9, we have $\Phi_X(\mathcal{H}) = f$. Hence, by Theorem 2, we deduce that the representation (X, f) of $\mathcal{H}$ is also functionally minimal. □

References

1. Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., Süsstrunk, S.: Slic superpixels compared to state-of-the-art superpixel methods. IEEE Trans. Pattern Anal. Mach. Intell. **34**(11), 2274–2282 (2012)
2. Alpert, S., Galun, M., Brandt, A., Basri, R.: Image segmentation by probabilistic bottom-up aggregation and cue integration. IEEE Trans. Pattern Anal. Mach. Intell. **34**(2), 315–327 (2012)
3. Arbelaez, P., Maire, M., Fowlkes, C., Malik, J.: Contour detection and hierarchical image segmentation. IEEE Trans. Pattern Anal. Mach. Intell. **33**(5), 898–916 (2011)
4. Arbeláez, P., Cohen, L.: A metric approach to vector-valued image segmentation. Int. J. Comput. Vis. **69**(1), 119–126 (2006)
5. Bender, M., Farach-Colton, M.: The LCA problem revisited. In: Latin American Theoretical INformatics, pp. 88–94 (2000)
6. Bertrand, G.: On topological watersheds. J. Math. Imaging Vis. **22**(2–3), 217–230 (2005)
7. Beucher, S.: Watershed, hierarchical segmentation and waterfall algorithm. In: Serra, J., Soille, P. (eds.) Mathematical Morphology and Its Applications to Image and Signal Processing, pp. 69–76. Kluwer (1994)
8. Cormen, T.H., Leiserson, C.E., Rivest, R.L., Stein, C., et al.: Introduction to Algorithms, 2nd edn. MIT Press, Cambridge (2001)

9. Couprie, M., Najman, L., Bertrand, G.: Quasi-linear algorithms for the topological watershed. J. Math. Imaging Vis. **22**(2–3), 231–249 (2005)
10. Cousty, J., Bertrand, G., Najman, L., Couprie, M.: Watershed cuts: minimum spanning forests and the drop of water principle. IEEE Trans. Pattern Anal. Mach. Intell. **31**(8), 1362–1374 (2009)
11. Cousty, J., Najman, L.: Incremental algorithm for hierarchical minimum spanning forests and saliency of watershed cuts. Math. Morphol. Its Appl. Image Signal Process. Lect. Notes Comput. Sci. **6671**, 272–283 (2011)
12. Cousty, J., Najman, L., Kenmochi, Y., Guimarães, S.: New characterizations of minimum spanning trees and of saliency maps based on quasi-flat zones. In: Benediktsson, J.A., Chanussot, J., Najman, L., Talbot, H. (eds.) Mathematical Morphology and Its Applications to Signal and Image Processing, Lecture Notes in Computer Science, vol. 9082, pp. 205–216. Springer, UK (2015). doi:10.1007/978-3-319-18720-4_18
13. Cousty, J., Najman, L., Perret, B.: Constructive links between some morphological hierarchies on edge-weighted graphs. In: Luengo Hendriks, C.L., Borgefors, G., Strand, R. (eds.) Mathematical Morphology and Its Applications to Image and Signal Processing, pp. 86–97. Springer, Berlin (2013)
14. Cousty, J., Najman, L., Serra, J.: Raising in watershed lattices. In: 15th IEEE International Conference on Image Processing, 2008 (ICIP 2008), pp. 2196–2199. IEEE (2008)
15. Felzenszwalb, P., Huttenlocher, D.: Efficient graph-based image segmentation. Int. J. Comput. Vis. **59**, 167–181 (2004)
16. Gordon, A.D.: A review of hierarchical classification. J. R. Stat. Soc. Ser. A (Gen.) **150**, 119–137 (1987)
17. Guigues, L., Cocquerez, J.P., Men, H.L.: Scale-sets image analysis. Int. J. Comput. Vis. **68**(3), 289–317 (2006)
18. Guimarães, S.J.F., Cousty, J., Kenmochi, Y., Najman, L.: A hierarchical image segmentation algorithm based on an observation scale. In: SSPR/SPR, pp. 116–125. Springer, Berlin (2012)
19. Guimarães, S., Kenmochi, Y., Cousty, J., Patrocínio, Z., Jr., Najman, L.: Hierarchizing Graph-Based Image Segmentation Algorithms Relying on Region Dissimilarity the Case of the Felzenszwalb–Huttenlocher Method (2016). http://hal-upec-upem.archives-ouvertes.fr/hal-01342967v2/document. Research report on HAL
20. Guimarães, S.J.F., do Patrocínio, Z.K.G.J.: A graph-based hierarchical image segmentation method based on a statistical merging predicate. In: Petrosino, A. (ed.) Image Analysis and Processing—ICIAP 2013, Lecture Notes in Computer Science, vol. 8156, pp. 11–20. Springer, Berlin (2013)
21. Guimarães, S.J.F., do Patrocínio, Z.K.G.J., Kenmochi, Y., Cousty, J., Najman, L.: Hierarchical image segmentation relying on a likelihood ratio test. In: Murino, V., Puppo, E. (eds.) Image Analysis and Processing—ICIAP 2015, Lecture Notes in Computer Science, vol. 9280, pp. 25–35. Springer. Berlin (2015)
22. Kiran, B.R., Serra, J.: Fusion of ground truths and hierarchies of segmentations. Pattern Recognit. Lett. **47**, 63–71 (2014). Advances in Mathematical Morphology
23. Kiran, B.R., Serra, J., et al.: Global-local optimizations on hierarchies of segmentations. Pattern Recognit. **47**(1), 12–24 (2014)
24. Kovalevsky, V.A.: Finite topology as applied to image analysis. Comput. Vis. Graph. Image Process. **46**(2), 141–161 (1989)
25. Leclerc, B.: Description combinatoire des ultramétriques. Mathématiques et Sciences humaines **73**, 5–37 (1981)
26. Malmberg, F., Hendriks, C.L.L.: An efficient algorithm for exact evaluation of stochastic watersheds. Pattern Recognit. Lett. **47**, 80–84 (2014). Advances in Mathematical Morphology
27. Martin, D.R., Fowlkes, C.C., Malik, J.: Learning to detect natural image boundaries using local brightness, color, and texture cues. IEEE Trans. Pattern Anal. Mach. Intell. **26**(5), 530–549 (2004)
28. Meyer, F.: The dynamics of minima and contours. In: Maragos, P., Schafer, R., Butt, M. (eds.) Mathematical Morphology and Its Applications to Image and Signal Processing, pp. 329–336. Kluwer Academic Publisher (1996)
29. Meyer, F., Maragos, P.: Morphological scale-space representation with levelings. In: Nielsen, M., Johansen, P., Olsen, O.F., Weickert, J. (eds.) Scale-Space Theories in Computer Vision, Lecture Notes in Computer Science, vol. 1682, pp. 187–198. Springer, Berlin (1999)
30. Meyer, F., Najman, L.: Segmentation, Minimum Spanning Tree and Hierarchies. Mathematical Morphology: From Theory to Applications, pp. 229–261. Wiley (2013)
31. Monasse, P., Guichard, F.: Fast computation of a contrast-invariant image representation. IEEE Trans. Image Process. **9**(5), 860–872 (2000)
32. NetworkX Examples: Knuth Miles. http://networkx.github.io/documentation/latest/examples/drawing/knuth_miles.html. Accessed 21 Sept 2015
33. Nagao, M., Matsuyama, T., Ikeda, Y.: Region extraction and shape analysis in aerial photographs. Comput. Graph. Image Process. **10**(3), 195–223 (1979)
34. Najman, L.: On the equivalence between hierarchical segmentations and ultrametric watersheds. J. Math. Imaging Vis. **40**(3), 231–247 (2011)
35. Najman, L., Cousty, J., Perret, B.: Playing with Kruskal: algorithms for morphological trees in edge-weighted graphs. In: Luengo Hendriks, C.L., Borgefors, G., Strand, R. (eds.) Mathematical Morphology and Its Applications to Image and Signal Processing, pp. 135–146. Springer, Berlin (2013)
36. Najman, L., Schmitt, M.: Geodesic saliency of watershed contours and hierarchical segmentation. IEEE Trans. Pattern Anal. Mach. Intell. **18**(12), 1163–1173 (1996)
37. Nakache, J.P., Confais, J.: Approche pragmatique de la classification: arbres hiérarchiques, partitionnements. Editions Technip (2004)
38. Nock, R., Nielsen, F.: Statistical region merging. IEEE Trans. Pattern Anal. Mach. Intell. **26**(11), 1452–1458 (2004)
39. Pavlidis, T.: Structural Pattern Recognition, vol. 2. Springer, New York (1977)
40. Peng, B., Zhang, L., Zhang, D.: Automatic image segmentation by dynamic region merging. IEEE Trans. Image Process. **20**(12), 3592–3605 (2011)
41. Perret, B., Cousty, J., Guimarães, S., Maya, D.: Evaluation of Hierarchical Watersheds (2016). http://hal.archives-ouvertes.fr/hal-01430865/ (**submitted**)
42. Philipp-Foliguet, S., Jordan, M., Najman, L., Cousty, J.: Artwork 3D model database indexing and classification. Pattern Recognit. **44**(3), 588–597 (2011)
43. Ronse, C.: Ordering partial partitions for image segmentation and filtering: merging, creating and inflating blocks. J. Math. Imaging Vis. **49**(1), 202–233 (2014)
44. Rother, C., Kolmogorov, V., Blake, A.: GrabCut: interactive foreground extraction using iterated graph cuts. ACM Trans. Graph. **23**(3), 309–314 (2004)
45. Salembier, P., Garrido, L.: Binary partition tree as an efficient representation for image processing, segmentation, and information retrieval. IEEE Trans. Image Process. **9**(4), 561–576 (2000)
46. Salembier, P., Oliveras, A., Garrido, L.: Antiextensive connected operators for image and sequence processing. IEEE Trans. Image Process. **7**(4), 555–570 (1998)
47. Silla Jr., C.N., Freitas, A.A.: A survey of hierarchical classification across different application domains. Data Min. Knowl. Discov. **22**(1–2), 31–72 (2011)
48. Skupin, A.: The world of geography: visualizing a knowledge domain with cartographic means. Proc. Natl. Acad. Sci. **101**(suppl 1), 5274–5278 (2004)

49. Skupin, A., Fabrikant, S.I.: Spatialization methods: a cartographic research agenda for non-geographic information visualization. Cartogr. Geogr. Inf. Sci. **30**(2), 99–119 (2003)
50. Soille, P.: Constrained connectivity for hierarchical image partitioning and simplification. IEEE Trans. Pattern Anal. Mach. Intell. **30**(7), 1132–1145 (2008)
51. Soille, P., Najman, L.: On morphological hierarchical representations for image processing and spatial data clustering. In: Köthe, U., Montanvert, A., Soille, P. (eds.) Applications of Discrete Geometry and Mathematical Morphology: First International Workshop, WADGMM 2010, Istanbul, Turkey, August 22, 2010, Revised Selected Papers, pp. 43–67. Springer, Berlin (2012)
52. Urbach, E., Roerdink, J., Wilkinson, M.: Connected shape-size pattern spectra for rotation and scale-invariant classification of gray-scale images. IEEE Trans. Pattern Anal. Mach. Intell. **29**(2), 272–285 (2007)
53. Vachier, C., Meyer, F.: Extinction value: a new measurement of persistence. In: IEEE Workshop on Nonlinear Signal and Image Processing, pp. 254–257 (1995)

Jean Cousty received his Ingénieur's degree from the Ecole Supérieure d'Ingénieurs en Electrotechnique et Electronique (ESIEE Paris, France) in 2004 and the Ph.D. degree from the Université de Marne-la-Vallée (France) in 2007. After a one-year postdoctoral period in the ASCLEPIOS research team at INRIA (Sophia-Antipolis, France), he is now teaching and doing research with the Informatics Department, ESIEE Paris, and with the Laboratoire d'Informatique Gaspard-Monge, Université Paris-Est. His current research interests include medical image analysis and discrete mathematics.

Laurent Najman received the Habilitation à Diriger les Recherches in 2006 from the University of Marne-la-Vallée, a Ph.D. of applied mathematics from Paris-Dauphine University in 1994 with the highest honour (Félicitations du Jury) and an Ingénieur degree from the Ecole des Mines de Paris in 1991. After earning his engineering degree, he worked in the Central Research Laboratories of Thomson-CSF for three years, working on some problems of infrared image segmentation using mathematical morphology. He then joined a start-up company named Animation Science in 1995, as director of research and development. The technology of particle systems for computer graphics and scientific visualisation, developed by the company under his technical leadership received several awards, including the European Information Technology Prize 1997 awarded by the European Commission (Esprit programme) and by the European Council for Applied Science and Engineering and the Hottest Products of the Year 1996 awarded by the Computer Graphics World journal. In 1998, he joined OCÉ Print Logic Technologies, as senior scientist. He worked there on various problem of image analysis dedicated to scanning and printing. In 2002, he joined the Informatics Department of ESIEE Paris, where he is professor and a member of the Laboratoire d'Informatique Gaspard-Monge, Université Paris-Est Marne-la-Vallée. His current research interests are discrete mathematical morphology, discrete topology and optimization.

Yukiko Kenmochi received the D.Eng. degree in information and computer sciences from Chiba University, Japan, in 1998. She was a Research Assistant at Japan Advanced Institute of Science and Technology between 1998 and 2003, during which she got a JSPS oversea research fellowship and spent two years at Laboratoire A2SI, ESIEE, France. After working as a Lecturer at Okayama University, Japan, she joined Laboratoire d'Informatique Gaspard-Monge, Université Paris-Est in 2004, where she is a CNRS Research Associate and a member of the research group A3SI. Her research interests include discrete geometry for computer imagery.

Silvio Guimarães received his Ph.D. degree from the Universidade Federal de Minas Gerais and Université de Marne-la-Valle (France) in 2003. He is now teaching and doing research with the Computer Science Department, at PUC Minas. He is the head of the Audio-Visual Processing Laboratory (VIPLAB) at PUC Minas. His current research interests include mathematical morphology and multimedia retrieval.