Watershed Cuts: Thinnings, Shortest Path Forests, and Topological Watersheds

Jean Cousty, Gilles Bertrand, Laurent Najman, Michel Couprie

Watershed cuts: thinnings, shortest-path forests and topological watersheds

Jean Cousty[1,2], Gilles Bertrand[1], Laurent Najman[1] and Michel Couprie[1]

***Abstract*— We recently introduced the watershed cuts, a notion of watershed in edge-weighted graphs. In this paper, our main contribution is a thinning paradigm from which we derive three algorithmic watershed cut strategies: the first one is well suited to parallel implementations, the second one leads to a flexible linear-time sequential implementation whereas the third one links the watershed cuts and the popular flooding algorithms. We state that watershed cuts preserve a notion of contrast, called connection value, on which are (implicitly) based several morphological region merging methods. We also establish the links and differences between watershed cuts, minimum spanning forests, shortest-path forests and topological watersheds. Finally, we present illsutrations of the proposed framework to the segmentation of artwork surfaces and diffusion tensor images.**

***Index Terms*— Watershed, thinning, minimum spanning forest, shortest-path forest, connection value, image segmentation**

INTRODUCTION

SINCE the early work of Zahn [1], several efficient tools for image segmentation have been expressed in the framework of edge-weighted graphs. In general, they extract a *cut* from a pixel adjacency graph (*i.e.*, a graph whose vertex set is the set of image pixels and whose edge set is given by an adjacency relations on these pixels). Informally, a cut is a set of edges which, when removed from the graph, separates it into different connected components: it is an inter-pixel separation which partition the image. Given a set of seed-vertices, which "mark" regions of interest in the image, the goal of these operators is to find a cut for which each induced connected component contains exactly one seed and which best matches a criterion based on the image contents. In order to define such a criterion, each edge of the graph is weighted by a measure of similarity (or dissimilarity) between the two pixels linked by this edge. In this context, the principle of *min-cut segmentation* [2] (and its variant [3]) is to find a cut for which the (weighted) sum of edge weights is minimal. *Shortest-path forest* approaches such as [4], [5] are also expressed in edge-weighted graphs. They look for a cut such that each vertex is connected to the closest seed for a particular distance in the graph. In [6], the author considers another approach where the weight of an edge is interpreted as the probability that a random walker chooses this edge, when standing at one of its extremity. Then, the proposed segmentation operator finds a cut for which each vertex is connected to the seed that this random walker starting at this vertex will first reach.

[1] Université Paris-Est, Laboratoire d'Informatique Gaspard-Monge, Equipe A3SI, ESIEE Paris, France
[2] INRIA Sophia Antipolis - ASCLEPIOS Team, France
Email addresses: {j.cousty, g.bertrand, l.najman, m.couprie}@esiee.fr
This work was partially supported by ANR grant SURF-NT05-2 45825.

The watershed transform introduced by Beucher and Lantuéjoul [7] for image segmentation is used as a fundamental step in many powerful segmentation procedures. Many approaches [7]–[15] have been proposed to define and/or compute the watershed of a vertex-weighted graph corresponding to a grayscale image. The digital image is seen as a topographic surface: the gray level becomes the elevation, the basins and valleys of the topographic surface correspond to dark areas, whereas the mountains and crest lines correspond to light areas. Intuitively, the watershed is a subset of the domain, located on the ridges of the topographic surface, that delineates its catchment basins.

An important motivation of our work is to provide a notion of watershed in the unifying framework of edge-weighted graphs that can help to precisely determine the relation between watersheds and the popular methods presented in the first paragraph. This paper is the second of a series of two articles dedicated to such a notion of watersheds in graphs whose edges (rather than vertices) are weighted. In this framework, a watershed is a cut. Before going further, let us emphasize that any practical comparison between watersheds in edge-weighted graphs and in vertex-weighted graphs should be made with care. Indeed, in general, the choice of one of these frameworks depends on the application. In particular, the framework of vertex-weighted graphs is adapted when the segmented regions must be separated by pixels. In this case, note that the watershed separation is not necessarily one pixel width and can be arbitrary thick (see a study of this problem in [15], [16]). On the contrary, when an inter-pixel separation is desired, the framework of edge-weighted graphs is appropriate.

A watershed of a topographic surface may be thought of as a separating line-set from which a drop of water can flow down towards several minima. Following this intuitive drop of water principle, we introduce in [16] the watershed cuts, a notion of watershed in edge-weighted graphs. We establish [16] the consistency of watershed cuts: they can be equivalently characterized by their catchment basins (through a steepest descent property) or by their dividing lines (through the drop of water principle). In [17], Meyer shows a link between minimum spanning forests and a flooding algorithm often used to compute watersheds. As proved in [16], there is indeed an equivalence between watershed cuts and cuts induced by minimum spanning forests relative to the minima. Section I of this paper sums up the results of [16] that are necessary in the sequel.

In Section II, we introduce a new thinning paradigm to characterize and compute the watershed cuts. Intuitively, a thinning is obtained from an edge-weighted graph by iteratively lowering the values of the edges that satisfy a certain property. We propose three different properties for selecting the edges which are to be lowered. They lead to three different thinning strategies. The effect of these transforms is to extend the minima of the original map in a way such that the minima of the transformed map

constitute a minimum spanning forest relative to the minima of the original map. Thus, we can prove (Th. 17) that these thinnings allow for a characterization of watershed cuts. The first of these three schemes (Section II-B) uses a purely local strategy to detect the edges which are to be lowered. It is therefore well suited to parallel implementations. The second one (Section II-C) leads to a sequential algorithm (Algorithm $\mathscr{M}$-kernel) which runs in linear-time (with respect to the number of edges of the graph) whatever the range of the weight function. We stress that Algorithm $\mathscr{M}$-kernel, and the one introduced in [16], are the first watershed algorithms that satisfy such a property. Indeed, as far as we know, the watershed algorithms available in the literature (*e.g.*, [4], [8], [9], [13], [14], [18]) all require either a sorting step, a hierarchical queue or a data structure to maintain a collection of disjoint sets under the operation of union and none of these operations can be performed in linear-time whatever the range of the weight function. Moreover, in practice, the algorithm proposed in this paper is more flexible than the one proposed in [16]. Indeed, the proposed algorithm allows the user to choose (with respect to the application requirements) between several strategies for setting the watershed position in the case where multiple acceptable solutions exist (*e.g.*, when the watershed must be positioned across a plateau of constant altitude). Finally, our third thinning strategy (Section II-D) establishes the link between watershed cuts and the popular flooding algorithms.

Due to noise and texture, the weight maps derived from real-world images often have a huge number of regional minima. Thus, their watersheds define too many catchment basins. A common issue to reduce this so-called over-segmentation is to use the result of the watershed as a starting point for a region merging procedure (see, e.g., [19]). In order to identify the pairs of neighboring regions to be merged, many methods are based on the values of the points or edges that belong to the initial separation between regions. In particular, in mathematical morphology, several methods [20]–[22] are implicitly based on the assumption that the initial separation satisfies a fundamental constraint: the values of the points or edges in the separation must convey a notion of contrast, called *connection value*, between the minima of the original image. The connection value [23]–[25] between two minima A and B is the minimal value Υ such that there exists a path from A to B the maximal value of which is Υ. From a topographical point of view, this value can be intuitively interpreted as the minimal altitude that a global flooding of the relief must reach in order to merge the lakes that flood A and B. Surprisingly, in vertex-weighted graphs, several watershed algorithms do not produce a separation that verifies this property. In this case, the watershed is not on the most "significant contours" [25] and cannot be used to correctly compute morphological hierarchies such as those proposed in [20]–[22]. In Section III, we prove (Th. 20) that the values of the edges in any watershed cut (and more generally in any cut induced by a minimum spanning forest) are sufficient to recover the connection values between the minima of the original map.

In fact, the connection value itself is used for defining several important segmentation methods such as the fuzzy connectedness segmentation [5], [26], [27], the image foresting transform [4] or the topological watershed [23]. Indeed, the two first methods fall in the category of shortest-path forests if a shortest path between two points x and y is defined as a path which "realizes" the connection value between x and y. In the sequel, such a shortest-path forest is called an Υ-shortest-path forest. In Section IV, we prove (Th. 21) that any minimum spanning forest is a Υ-shortest-path forest and that the converse is, in general, not true. Then, we show (Th. 25) that any watershed cut is a topological cut (*i.e.*, a separation induced by a topological watershed defined in an edge-weighted graph) but that the converse is, in general, not true. We emphasize that this study helps, in practice, to choose among these segmentation techniques the one which will best solve a particular problem.

The interest of the proposed framework to segment grayscale images is demonstrated in [16]. In Section V, we illustrate its versatility to segment different kinds of geometric objects. We present two recent applications where watershed cuts are used to segment the surface of artwork 3D objects and to segment the corpus callosum in brain diffusion tensors images.

This article is self-contained and proofs of the properties are given in the IEEE digital library.

I. Watershed cuts and minimum spanning forests

The intuitive idea underlying the notion of a watershed comes from the field of topography: a drop of water falling on a topographic surface follows a descending path and eventually reaches a minimum. The watershed may be thought of as the separating lines of the domain of attraction of drops of water. In [16], we follow explicitly this drop of water principle to define the notion of a watershed in an edge-weighted graph. This approach leads to a consistent definition of watersheds (with respect to characterizations of both catchment basins and dividing lines) as assessed by Th. 6 in [16]. In this section, after a presentation of basic notations, we recall the definition of a watershed cut and a property which establishes its optimality.

A. Edge-weighted graphs

Following the notations of [28], we present basic definitions to handle edge-weighted graphs.

We define a *graph* as a pair $X = (V(X), E(X))$ where $V(X)$ is a finite set and $E(X)$ is composed of unordered pairs of $V(X)$, *i.e.*, $E(X)$ is a subset of $\{\{x, y\} \subseteq V(X) \mid x \neq y\}$. Each element of $V(X)$ is called a *vertex or a point (of X)*, and each element of $E(X)$ is called an *edge (of X)*. If $V(X) \neq \emptyset$, we say that X is *non-empty*.

Let X be a graph. If $u = \{x, y\}$ is an edge of X, we say that x and y are *adjacent (for X)*. Let $\pi = \langle x_0, \ldots, x_\ell \rangle$ be an ordered sequence of vertices of X, π is *a path from x_0 to x_ℓ in X (or in $V(X)$)* if for any $i \in [1, \ell]$, x_i is adjacent to x_{i-1}. In this case, we say that *x_0 and x_ℓ are linked for X*. If $\ell = 0$, then *π is a trivial path in X*. We say that *X is connected* if any two vertices of X are linked for X.

Let X and Y be two graphs. If $V(Y) \subseteq V(X)$ and $E(Y) \subseteq E(X)$, we say that *Y is a subgraph of X* and we write $Y \subseteq X$. We say that Y is a *connected component of X*, or simply a *component of X*, if Y is a connected subgraph of X which is maximal for this property, *i.e.*, for any connected graph Z, $Y \subseteq Z \subseteq X$ implies $Z = Y$.

Important remark. *Throughout this paper G denotes a connected graph. In order to simplify the notations, this graph will be denoted by $G = (V, E)$ instead of $G = (V(G), E(G))$. We will also assume that $E \neq \emptyset$.*

Let $X \subseteq G$. An edge $\{x, y\}$ of G is *adjacent to X* if $\{x, y\} \cap V(X) \neq \emptyset$ and if $\{x, y\}$ does not belong to $E(X)$; in this case

and if y does not belong to $V(X)$, we say that y *is adjacent to* X. If π is a path from x to y and y is a vertex of X, then π is a *path from* x *to* X *(in* G*).*

If S is a subset of E, we denote by $\overline{S}$ *the complementary set of* S *in* E*, i.e.,* $\overline{S} = E \setminus S$.

Let $S \subseteq E$*, the graph induced by* S is the graph whose edge set is S and whose vertex set is made of all points which belong to an edge in S*, i.e.,* $(\{x \in V \mid \exists u \in S, x \in u\}, S)$. In the following, when no confusion may occur, the graph induced by S is also denoted by S.

We denote by $\mathcal{F}$ the set of all maps from E to $\mathbb{R}$ and we say that any map in $\mathcal{F}$ *weights* the edges of G.

Let $F \in \mathcal{F}$. If u is an edge of G, $F(u)$ is the *altitude or weight* of u. Let $X \subseteq G$ and $k \in \mathbb{R}$. A subgraph X of G is a *minimum of* F *(at altitude* k*)* if:

- X is connected; and
- k is the altitude of any edge of X; and
- the altitude of any edge adjacent to X is strictly greater than k.

We denote by $M(F)$ the graph whose vertex set and edge set are, respectively, the union of the vertex sets and edge sets of all minima of F. Figs. 1b and c illustrate the definition of minima.

Important remark. *In the sequel of this paper,* F *denotes an element of* $\mathcal{F}$ *and therefore the pair* (G, F) *is called an* edge-weighted graph.

Before presenting the watershed cuts in the next section, let us briefly introduce basic ways to define an edge-weighted graph for segmenting a digital image. In Section V, we also show how to define edge-weighted graphs to segment triangulated surfaces and diffusion tensor images. In applications to grayscale image segmentation, V is the set of picture elements (pixels) and E is any of the usual adjacency relations, *e.g.*, the 4-adjacency in 2D [29]. Then, a grayscale image I attributes a value to each element of V. For watershed segmentation, we suppose that the salient contours of I are located on the highest edges of G. Thus, depending on the application, there are several possibilities to set up the map F from the image I.

A common issue is to segment a grayscale image into its "homogeneous" zones. To this end, one can weight each edge $\{x, y\} \in E$ with a simple dissimilarity function defined by $F(\{x, y\}) = |I(x) - I(y)|$ (see *e.g.*, Figs. 1a and b). This measure of dissimilarity is strictly local in the sense that the weight of an edge depends on the intensity of the two pixels linked by this edge. In some practical situations (*e.g.*, in presence of noise), it is convenient to use a more robust measure based on a larger neighborhood. For instance, one can weight each edge $\{x, y\}$ in E by $F(\{x, y\}) = \max\{I(z) \mid z \in N_u\} - \min\{I(z) \mid z \in N_u\}$, where N_u is the neighborhood of $u = \{x, y\}$ made of all vertices adjacent to either x or y (*i.e.*, $N_u = \{z \in V \mid \{x, z\} \in E$ or $\{y, z\} \in E\}$). This second strategy is illustrated in Fig. 1c. Finally, if we want to segment the dark regions of a grayscale image that are separated by brighter zones, another way to weight each edge $u \in E$, linking two pixels x and y, consists of taking the minimum (or maximum) value of the intensities at points x and y: $F(\{x, y\}) = \min\{I(x), I(y)\}$.

B. Watershed cuts

We first recall the notions of extension [16], [23] and graph cut which play an important role for defining a watershed in an edge-weighted graph. Intuitively, the regions of a watershed (also called catchment basins) are associated with the regional minima of the map. Each catchment basin contains a unique regional minimum, and conversely, each regional minimum is included in a unique catchment basin: the regions of the watershed "extend" the minima.

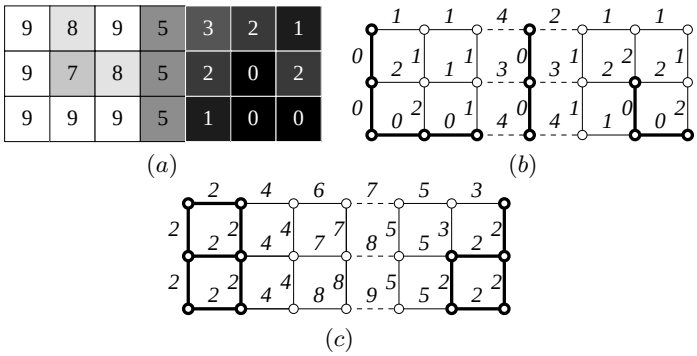


Fig. 1. Illustration of two dissimilarity measures [see text] to weight the edges of a 4-connected graph from a digital image. In (b) and (c) the bold subgraphs represent the minima and the dashed edges the watershed cuts.

Definition 1 (extension, cut): Let X and Y be two non-empty subgraphs of G.
We say that Y is an *extension of* X *(in* G*)* if $X \subseteq Y$ and if any component of Y contains exactly one component of X.
Let $S \subseteq E$. We say that S is a *(graph) cut for* X if $\overline{S}$ is an extension of X and if S is minimal for this property, *i.e.*, if $T \subseteq S$ and $\overline{T}$ is an extension of X, then we have $T = S$.

On a topographic surface, a drop of water flows down towards a regional minimum. Therefore, before reminding the definition of watershed cuts, we need the notion of a descending path.

Let $\pi = \langle x_0, \ldots, x_\ell \rangle$ be a path in G. The path π is *descending (for* F*)* if, for any $i \in [1, \ell - 1]$, $F(\{x_{i-1}, x_i\}) \geq F(\{x_i, x_{i+1}\})$.

Definition 2 (drop of water principle): Let $S \subseteq E$. We say that S *satisfies the drop of water principle (for* F*)* if $\overline{S}$ is an extension of $M(F)$ and if for any $u = \{x_0, y_0\} \in S$, there exist $\pi_1 = \langle x_0, \ldots, x_n \rangle$ and $\pi_2 = \langle y_0, \ldots, y_m \rangle$ which are two descending paths in $\overline{S}$ such that:
- x_n and y_m are vertices of two distinct minima of F; and
- $F(u) \geq F(\{x_0, x_1\})$ (resp. $F(u) \geq F(\{y_0, y_1\})$), whenever π_1 (resp. π_2) is not trivial.

If S satisfies the drop of water principle, we say that S is a *watershed cut, or simply a watershed, of* F.

We illustrate the previous definition on the function F depicted in Fig. 2. The function F contains three minima (in bold Fig. 2a). We denote by S the set of dashed edges depicted in Fig. 2b. It may be seen that $\overline{S}$ (in bold Fig. 2b) is an extension of $M(F)$. Let us consider the edge $u = \{j, k\} \in S$. There exists $\pi_1 = \langle j, f, e, a \rangle$ (resp. $\pi_2 = \langle k \rangle$) a descending path in $\overline{S}$ from j (resp. k) to the minimum whose vertex set contains a (resp. k); on the one hand, the altitude of $\{j, f\}$, the first edge of π_1 is equal to 4 which is a value lower than 5 the altitude of u; on the other hand $\langle k \rangle$ is a trivial path. Similarly to u, it can be verified that the two properties which must be satisfied by the edges in a watershed hold true for any edge in S. Thus, S is a watershed of F. Notice also that a watershed of F is necessarily a cut for $M(F)$.

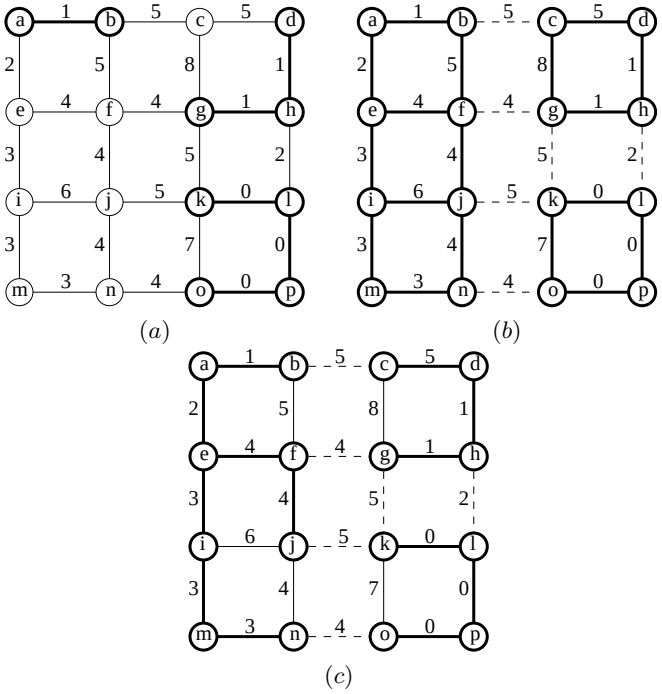


Fig. 2. A graph G and a map F. Edges and vertices in bold depict: (a), $M(F)$, the minima of F; (b), an extension $\overline{S}$ of $M(F)$; (c), a MSF relative to $M(F)$. In (b) (resp. (c)) the set S of dashed edges is a watershed cut of F (resp. a MSF cut for $M(F)$).

C. Minimum spanning forests: watershed optimality

In [16], we establish the optimality of watersheds. To this end, the notion of minimum spanning forests relative to subgraphs of G is introduced. Each of these forests induces a cut. In this subsection, we recall the definition of these forests and the equivalence between the watershed cuts and the cuts induced by minimum spanning forests relative to the minima (see [16] for more details). This result will be used to prove the main claim of this article.

Generally, in graph theory, a forest is defined as a graph that does not contain any cycle. In this paper, the notion of forest is not sufficient since we want to deal with extensions of subgraphs that can contain cycles (*e.g.*, the minima of a map). Therefore, we present hereafter the notion of a relative forest. It generalizes the usual notion of a forest in the sense that any forest is a relative forest, but, in general, a relative forest is not a forest. Intuitively, a forest relative to a subgraph X of G is an extension Y of X such that any cycle in Y is also a cycle in X. In other words, to construct a forest relative to an arbitrary subgraph X of G, one can add edges to X, provided that the added edges do not introduce new cycles and that the obtained graph remains an extension of X. Formally, the notion of cycle is not necessary to define a forest.

Definition 3 (forest): Let X and Y be two non-empty subgraphs of G. We say that Y is a *forest relative to* X if:

i) Y is an extension of X; and
ii) for any extension $Z \subseteq Y$ of X, we have $Z = Y$ whenever $V(Z) = V(Y)$.

We say that Y is a *spanning forest relative to* X *(for* G*)* if Y is a forest relative to X and if $V(Y) = V$.

Let X be a subgraph of G, *the weight of* X *(for* F*)*, denoted by $F(X)$, is the sum of the weights of the edges in $E(X)$: $F(X) = \sum_{u \in E(X)} F(u)$.

Definition 4 (minimum spanning forest): Let X and Y be two subgraphs of G. We say that Y *is a minimum spanning forest (MSF) relative to* X *(for* F*, in* G*)* if Y is a spanning forest relative to X and if the weight of Y is less than or equal to the weight of any other spanning forest relative to X. In this case, we also say that Y is a *relative MSF*.

For instance, the graph Y (bold edges and vertices) in Fig. 2c is a MSF relative to X (Fig. 2a).

Let X be a subgraph of G and let Y be a spanning forest relative to X. There exists a cut S for Y which is composed by the edges of G whose extremities are in two distinct components of Y. Since Y is an extension of X, it can be seen that this cut S is also a cut for X. We say that this cut is the *cut induced by* Y. Furthermore, if Y is a MSF relative to X, we say that that S is an *MSF cut for* X.

We recall the theorem proved in [16] which establishes the optimality of watershed cuts. It states the equivalence between the cuts which satisfy the drop of water principle and those induced by the MSFs relative to the minima of a map.

Theorem 5 (optimality, Th. 9 in [16]): Let $S \subseteq E$. The set S is an MSF cut for $M(F)$ if and only if S is a watershed cut of F.

As an illustration, it can be verified on Fig. 2b,c that the set of dashed edges is both a watershed cut of the map and an MSF cut for its minima.

II. Optimal thinnings

In this section, we introduce a new paradigm to compute MSFs relative to the minima, hence to compute watershed cuts. To this end, we first present a generic thinning paradigm from which we derive three algorithmic schemes. The first of this three schemes is well suited to parallel implementations. The second one leads to a linear-time (with respect to the number of edges of the graph) sequential watershed algorithm. Finally, the third one allows us to highlight the links between the watershed cuts and the immersion paradigm which is frequently used for computing watersheds in vertex-weighted graphs.

A. Thinnings

Intuitively, a thinning of F is a map obtained from F by iteratively lowering down the values of the edges of G which satisfy a given property.

Important remark. *From now on, we will denote by* $F^{\ominus}$ *the map from* V *to* $\mathbb{R}$ *such that for any* $x \in V$, $F^{\ominus}(x)$ *is the minimal altitude of an edge which contains* x, i.e., $F^{\ominus}(x) = \min\{F(u) \mid u \in E, x \in u\}$; $F^{\ominus}(x)$ *is the* altitude *of* x.

The map $F^{\ominus}$ associated to the map F depicted in Fig. 2a is shown in Fig. 3a.

A lowering is a transformation that replaces the weight of an edge u by the weight of the lowest edge adjacent to u while leaving unchanged the weight of any other edge. The weight of u in the transformed map is equal to the minimal altitude of the vertices that belong to u.

Let $u \in E$. The *lowering of* F *at* u is the map F' in $\mathcal{F}$ such that:

- $F'(u) = \min_{x \in u}\{F^{\ominus}(x)\}$; and
- $F'(v) = F(v)$ for any edge $v \in E \setminus \{u\}$.

For instance in Fig. 3, the map depicted in (b) (resp. (c) and (d)) is the lowering of the one shown in (a) at the edge $\{j, n\}$ (resp. $\{c, d\}$ and $\{a, e\}$).

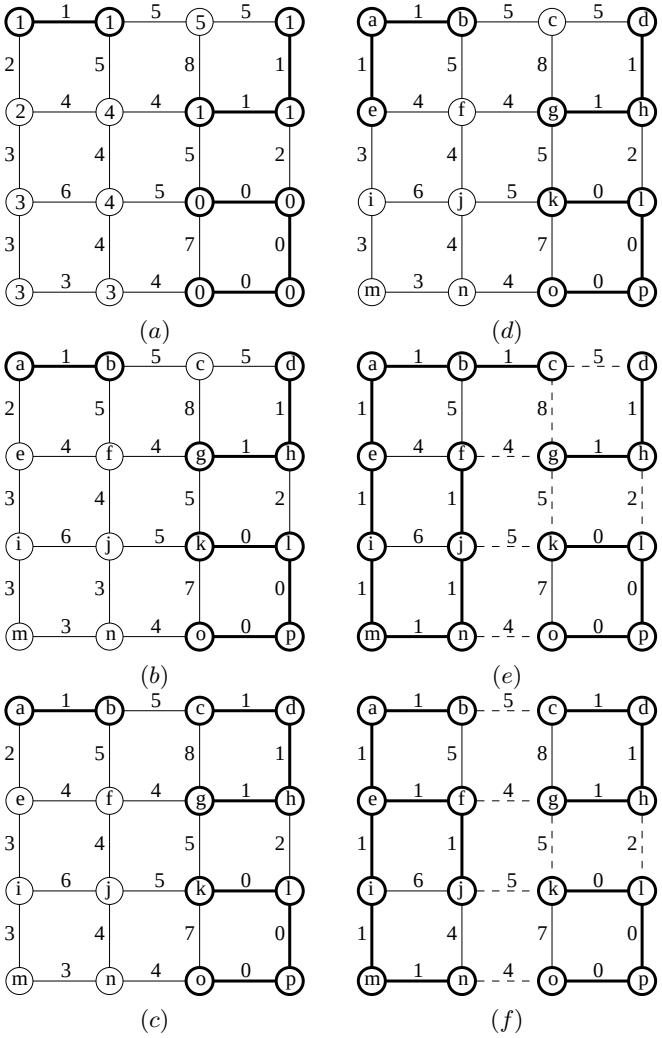


Fig. 3. A graph G and some associated maps. The edges and vertices in bold are the minima of the depicted maps. (a), The values of a map $F \in \mathcal{F}$ are associated to the edges of G; the values of the map $F^{\ominus}$ are associated to the vertices of G. (b, c, d), Three $\mathscr{B}$-thinnings of F; (c, d), two $\mathscr{M}$-thinnings of F; and (d), an $\mathscr{I}$-thinning of F. (e, f), Two $\mathscr{B}$-kernels of F; the two $\mathscr{B}$-cuts associated to the $\mathscr{B}$-kernels are depicted by the dashed edges.

Intuitively, an *edge-property* is a criterion which attributes, to each edge of an edge-weighted graph, either the label $TRUE$ or the label $FALSE$. We will study later on several examples of such edge-properties which will serve us to define several thinning strategies.

Definition 6 (edge-property): An *edge-property* (for G) is a map $\mathscr{P}$ from $E \times \mathcal{F}$ in the set $\{TRUE, FALSE\}$.
Let $\mathscr{P}$ be an edge-property, H be a map in $\mathcal{F}$ and u be an edge in E. If $\mathscr{P}(u, H) = TRUE$, we say that u *satisfies* $\mathscr{P}$ *for* H.

Given an edge-property $\mathscr{P}$, we introduce a transformation, called $\mathscr{P}$-thinning, that acts on maps by iteratively lowering an initial map at the edges which satisfy the edge-property $\mathscr{P}$.

Definition 7 (thinning): Let $\mathscr{P}$ be an edge-property and H be a map in $\mathcal{F}$. The map H is a *$\mathscr{P}$-thinning* of F if:

- $H = F$; or if
- there exists a map J in $\mathcal{F}$ which is a $\mathscr{P}$-thinning of F such that H is the lowering of J at an edge which satisfies $\mathscr{P}$ for J.

If H is a $\mathscr{P}$-thinning of F and if, for any edge u in E, $\mathscr{P}(u, H) = FALSE$, then we say that H is a *$\mathscr{P}$-kernel* of F.

In other words, a map H is a $\mathscr{P}$-thinning of F if there exists a (possibly trivial) sequence of maps $\langle F_0, \ldots, F_\ell \rangle$ such that $F_0 = F$, $F_\ell = H$ and, for any $i \in [1, \ell]$, F_i is the lowering of F_{i-1} at an edge which satisfies $\mathscr{P}$ for F_{i-1}. Furthermore, we say that H is a $\mathscr{P}$-kernel of F if H is a $\mathscr{P}$-thinning of F such that there is no edge of G which satisfies $\mathscr{P}$ for H.

In the next subsections, we introduce three edge-properties that lead to three thinning transformations from which three different algorithmic strategies for watershed cuts are derived.

B. $\mathscr{B}$-thinnings: a local strategy for watershed cuts

We introduce a classification of edges based exclusively on local properties, *i.e.*, properties which depend only on the adjacent edges. In particular, we present the notion of a border edge. Then, we study the thinning transformation which uses the property of "being a border edge" to detect the edges at which a map should be lowered. Roughly speaking, the effect of this transform is to extend the minima of the original map so that the minima of the transformed map constitute an MSF relative to the minima of the original map. Hence, consequently to Th. 5, this transform can be used to extract watershed cuts. Since the notion of a border edge is local, the associated thinning strategy is well suited to parallel implementations.

Definition 8 (local edge classification): Let $u = \{x, y\} \in E$.

- We say that u is *locally separating (for F)* if $F(u) > \max(F^{\ominus}(x), F^{\ominus}(y))$.
- We say that u is *border (for F)* if $F(u) = \max(F^{\ominus}(x), F^{\ominus}(y))$ and $F(u) > \min(F^{\ominus}(x), F^{\ominus}(y))$[1].
- We say that u is *inner (for F)* if $F^{\ominus}(x) = F^{\ominus}(y) = F(u)$.

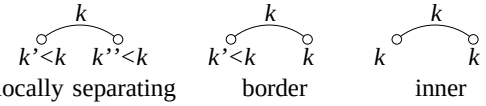


Fig. 4. Illustration of the different local configurations for edges.

Fig. 4 illustrates the above definitions. In Fig. 3, $\{j, n\}$, $\{c, d\}$ and $\{a, e\}$ are examples of border edges for the map shown in (a); $\{i, m\}$ and $\{k, l\}$ are inner edges for (a) and both $\{h, l\}$ and $\{g, k\}$ are locally-separating for (a). Note that any edge of G corresponds exactly to one of the types presented in Def. 8. Therefore, Def. 8 constitutes a classification of the edges of G. Furthermore, this classification is local since, the class of any edge $u = \{x, y\}$ depends only of the values $F(u)$, $F^{\ominus}(x)$ and $F^{\ominus}(y)$.

Definition 9 ($\mathscr{B}$-cut): We denote by $\mathscr{B}$ the edge-property such that, for any edge $u \in E$ and for any map $H \in \mathcal{F}$, $\mathscr{B}(u, H) = TRUE$ if and only if u is a border edge for H.
Let H be a $\mathscr{B}$-kernel of F. The set of all edges in E which are adjacent to two distinct minima of H is called a *$\mathscr{B}$-cut* for F.

In Fig. 3, the maps depicted in (b, c, d) are the lowering of the map (a) at respectively $\{j, n\}$, $\{c, d\}$ and $\{a, e\}$. These three edges are border edges for (a). Thus the maps (b), (c) and (d) are three $\mathscr{B}$-thinnings of F. The map shown in (e) is a $\mathscr{B}$-kernel of the maps (a), (b) and (d) but not a $\mathscr{B}$-kernel of (c). The map (f)

[1]Notice that a notion similar to the one of border edge has been proposed in the context of image segmentation under the name of *min-contractible edge* [30].

is another $\mathscr{B}$-kernel of (a). The $\mathscr{B}$-cuts associated to (e) and (f) are represented by dashed edges in the figure.

We now present an important result of this section which mainly states that the $\mathscr{B}$-kernels can by used to compute MSFs relative to the minima of a map.

Property 10: Let $H \in \mathcal{F}$. If H is a $\mathscr{B}$-thinning of F, then any MSF relative to $M(H)$ (for H) is an MSF relative to $M(F)$ (for F). Furthermore, if H is a $\mathscr{B}$-kernel of F, then $M(H)$ is itself an MSF relative to $M(F)$ (for F).

In other words, the $\mathscr{B}$-thinning transformation preserves some of the MSFs relative to the minima of the original map. More remarkably, the minima of a $\mathscr{B}$-kernel of F constitute precisely an MSF (for F) relative to the minima of F. Hence, the $\mathscr{B}$-kernels can be used to extract MSFs relative to the minima. We remind that an MSF relative to the minima of a map defines a cut composed of all edges which are adjacent to two distinct components of the MSF. Thus, a $\mathscr{B}$-kernel of a map defines a $\mathscr{B}$-cut for this map. Hence, by Prop. 10 and Th. 5, we can easily prove the following corollary which states that a $\mathscr{B}$-kernel of F defines a watershed of F.

Corollary 11: Any $\mathscr{B}$-cut of F is a watershed cut of F.

Thanks to classical algorithms for minima computation [31], an MSF relative to $M(F)$ can be obtained from any $\mathscr{B}$-kernel of F. In fact, using the local classification of Def. 8, the minima of a $\mathscr{B}$-kernel can be extracted in a simpler way. The following property directly follows from the definitions of a $\mathscr{B}$-kernel and of a minimum.

Property 12: Let H be a $\mathscr{B}$-kernel of F. An edge $u \in E$ is in a minimum of H if and only if u is inner for H.

Let H denote a $\mathscr{B}$-kernel of F. On the one hand, the map H and its minima can be derived from F exclusively by *local operations* (see Defs. 8, 9 and Prop. 12). On the other hand, an MSF relative to $M(F)$ is a *globally optimal structure*. The minima of H constitute, by Prop. 10, an MSF relative to $M(F)$. Thus, the local and order-independent operations presented in this section produce a globally optimal structure.

This kind of local, order-independent operations for obtaining optimal structures can be efficiently exploited by dedicated hardware. For instance, raster scanning strategies for extracting a $\mathscr{B}$-kernel and its minima (hence an MSF relative to the minima) can be straightforwardly derived. It has been shown that such strategies can be fast on adapted hardware [32].

As mentioned above, the property $\mathscr{B}$, which selects border edges, can be tested locally: to check whether $\mathscr{B}(u, H)$ (with $u \in E$ and $H \in \mathcal{F}$) equals $TRUE$, one only needs to consider the values of the edges adjacent to u. Thus, if a set of *independent* (*i.e.*, mutually non-adjacent) border edges is lowered in parallel, then the resulting map is a $\mathscr{B}$-thinning. This property offers several possibilities of parallel watershed algorithms. In particular, efficient algorithms for array processors can be derived.

C. $\mathscr{M}$-thinnings: an efficient sequential strategy for watershed cuts

On a sequential computer, a naive algorithm to obtain a $\mathscr{B}$-kernel of F could be the following: $i)$ for all $u = \{x, y\}$ in E, taken in an arbitrary order, check the values of $F(u)$, $F^{\ominus}(x)$ and $F^{\ominus}(y)$ and whenever $\mathscr{B}(u, F) = TRUE$ (*i.e.*, u is a border edge for F), lower the value of u down to the minimum of $F^{\ominus}(x)$ and $F^{\ominus}(y)$; $ii)$ repeat step $i)$ until no border edge remains. Consider the graph G whose vertex set is $V = \{0, \ldots, n\}$ and whose edge set E is made of all the pairs $u_i = \{i, i+1\}$ such that $i \in [0, n-1]$. Let $F(u_i) = n - i$, for all $i \in [0, n-1]$. On this graph, if the edges are processed in the order of their indices, step $i)$ will be repeated exactly $|E|$ times. The cost of step i) (check all edges of G) is $O(|E|)$. Thus, the worst case time complexity of this naive algorithm is at least $O(|E|^2)$.

In order to reduce this complexity, we introduce a second thinning transformation, called $\mathscr{M}$-thinning, in which any edge is lowered at most once. This process is a particular case of $\mathscr{B}$-thinning which also produces, when iterated until stability, a $\mathscr{B}$-kernel of the original map. Thanks to this second thinning strategy, we derive in Section II-F a linear-time algorithm to compute $\mathscr{B}$-kernels and, thus, watersheds.

It may be seen that an edge which is in a minimum at a given step of a $\mathscr{B}$-thinning sequence never becomes a border edge. Thus, lowering first the edges adjacent to the minima seems to be a promising strategy. In order to study and understand this strategy, we may classify any inner, border or locally-separating edge with respect to the adjacent minima. We thus obtain the 8 cases illustrated in Fig. 5. Any edge is classified in exactly one of these classes depending on the values of its adjacent edges and on the regional minima. In this section we study a thinning which iteratively lowers down the values of the border edges adjacent to minima (see Fig. 5F).

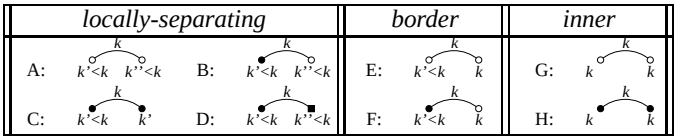


Fig. 5. Edge-classification in a weighted graph. In the figure, any black vertex belongs to a minimum and two vertices represented by different shapes (*i.e.*, square and circle) belong to distinct minima.

Definition 13 ($\mathscr{M}$-cut): We say that an edge u in E is *minimum-border (for F), written M-border*, if u is border for F and if exactly one of the vertices in u is a vertex of $M(F)$.
We denote by $\mathscr{M}$ the edge-property such that, for any edge $u \in E$ and for any map $H \in \mathcal{F}$, $\mathscr{M}(u, H) = TRUE$ if and only if u is an M-border edge for H.
Let H be an $\mathscr{M}$-kernel of F. The set of all edges in E which are adjacent to two distinct minima of H is called an *$\mathscr{M}$-cut* of F.

In Fig. 3, the edges $\{c, d\}$ and $\{a, e\}$ are M-border edges for the map (a) whereas $\{j, n\}$ is not. Thus, the maps (c) and (d) are $\mathscr{M}$-thinnings of (a) whereas (b) is not. Observe that when a map is lowered at an M-border edge, one vertex and one edge are added to a minimum. In Fig. 3, it can be also verified that the maps (e) and (f) are $\mathscr{M}$-kernels of (a) and that the associated $\mathscr{M}$-cuts are watershed cuts of (a). In Section II-E, we indeed prove the equivalence between $\mathscr{M}$-cuts, $\mathscr{B}$-cuts and watershed cuts. In Section II-F an efficient linear-time ($O(|E|)$) algorithm to compute the $\mathscr{M}$-cuts is derived. Thus, thanks to the $\mathscr{M}$-thinnings, we obtain a linear-time sequential algorithm to compute the watershed cuts of a map.

D. $\mathscr{I}$-thinnings: an immersion strategy for watershed cuts

Among the different schemes to compute a watershed in a vertex-weighted graph, the immersion strategies [8], [9] are the most frequently used in applications. They correspond to the intuitive idea of simulating the flooding of a topographic surface

from its minima. The watershed lines are made of dams build at the points where water coming from different minima would meet. Surprisingly, in general, the links between immersion algorithms and watersheds are not straightforward. Indeed, as shown in [25], in vertex-weighted graphs, these algorithms sometimes produce segmentations which are far from corresponding to the topographical intuition of a watershed. Among the immersion strategies, the procedure proposed by F. Meyer in [9] is probably the simplest to describe and understand. In an edge-weighted graph, it could be presented as follows: i) mark the minima with distinct labels; ii) mark the lowest edge containing exactly one labelled vertex with this label; and iii) repeat step ii) until idempotence. At the end of the procedure, the set of edges that link two vertices marked with distinct labels constitute the "watershed by flooding". An important contribution of this subsection and the following one is to prove that, in edge-weighted graphs, this procedure produces a watershed cut. In order to prove this result, we introduce the $\mathscr{I}$-thinnings that can be associated with the above procedure.

Let X be a subgraph of G, we say that an edge u is outgoing from X if one of the vertices in u belongs to the vertex set of X and if the other vertex in u does not.

Definition 14 ($\mathscr{I}$-cut): If u is an edge with minimal altitude among all the edges outgoing from $M(F)$, then we say that u is an *immersion edge for* F.

We denote by $\mathscr{I}$ the edge-property such that, for any edge $u \in E$ and for any map $H \in \mathcal{F}$, $\mathscr{I}(u, H) = TRUE$ if and only if u is an immersion edge for H.

Let H be an $\mathscr{I}$-kernel for F. The set of all edges in E which are adjacent to two distinct minima of H is called an *$\mathscr{I}$-cut* for F.

In order to stress the link between immersion and $\mathscr{I}$-thinnings, let us consider the following straightforward adaptation of the procedure presented in the introduction of the subsection.

(i) Mark the minima with distinct labels.
(ii) Mark the lowest edge u containing exactly one labelled vertex with this label and lower the map F at u (*i.e.,*, $F := F'$ where F' is the lowering of F at u).
(iii) Repeat step (ii) until idempotence.

After each iteration of step (ii), the map F is an $\mathscr{I}$-thinning of the input map. The set of labelled edges correspond to the minima of F and each minimum of F is marked with the label of the corresponding minimum in the input map. Thus, at the end of this algorithm the output map F is an $\mathscr{I}$-kernel of the input map and the set of all edges that link two vertices marked with distinct labels is an $\mathscr{I}$-cut of the input map.

Property 15: Any immersion edge for F is an M-border edge for F.

In Fig. 3, $\{a, e\}$ is an immersion edge for (a) whereas $\{c, d\}$ is not. Thus, the map (d) is an $\mathscr{I}$-thinning of (a) whereas the map (c) is not. On the one hand, as stated by Prop. 15, any immersion edge is an M-border edge. On the other hand, as shown by the previous example, there exist M-border edges which are not immersion edges. Thus, the $\mathscr{M}$-thinning transform generalizes the immersion algorithms in edge-weighted graphs. In the next subsection, we prove that any $\mathscr{I}$-cut is a watershed. For instance, in Fig. 3, the maps (e) and (f) are two $\mathscr{I}$-kernels of (a) and it can be verified that the associated $\mathscr{I}$-cuts are watershed cuts of (a).

Prop. 15 also establishes a link with the minimum spanning tree algorithm due to Prim [33]. To understand this link, we have to consider the construction (described in Sec. III.B of [16]) which was proposed to show the equivalence between computing an MSF relative to the minima and computing a minimum spanning tree. Roughly speaking, from an edge-weighted graph (G, F), we start by contracting each minimum of F into a single vertex. Then we add an extra-vertex linked to each contracted minimum by an edge of minimal weight. We thus obtain a new edge-weighted graph (G', F'). As stated by Meyer in [17], it may be seen that the edges considered by Prim's algorithm applied on (G', F') are the same as those considered in a sequence of $\mathscr{I}$-thinnings. Therefore, Prop. 15 gives us a clue to precisely determine the relation between MSFs relative to the minima and the thinning transforms introduced above. Precisely determining this relation is the topic of the next subsection.

E. Equivalence between $\mathscr{I}$-cuts, $\mathscr{M}$-cuts, $\mathscr{B}$-cuts and watersheds

We clarify the links that exist between the thinnings introduced above, the MSF relative to the minima and the watersheds. In particular, we show (Th. 17) that the $\mathscr{B}$-kernels, the $\mathscr{M}$-kernels and the $\mathscr{I}$-kernels lead to equivalent characterizations of watershed cuts.

The following property states that the minima of $\mathscr{B}$-kernels, the minima of $\mathscr{M}$-kernels and the minima of $\mathscr{I}$-kernels of F are all MSFs relative to $M(F)$. More remarkably, any MSF relative to $M(F)$ can be obtained as the minima of an $\mathscr{M}$-kernel of F, as the minima of an $\mathscr{I}$-kernel of F and also as the minima of a $\mathscr{B}$-kernel of F. Therefore, in this sense of minimum spanning forests, these thinning transformations may be seen as *optimal thinnings*.

Lemma 16: Let $X \subseteq G$. The four following statements are equivalent:

(i) there exists an $\mathscr{I}$-kernel H of F such that $M(H) = X$;
(ii) there exists an $\mathscr{M}$-kernel H of F such that $M(H) = X$;
(iii) there exists a $\mathscr{B}$-kernel H of F such that $M(H) = X$;
(iv) X is an MSF relative to $M(F)$.

Since a relative MSF induces a graph cut for $M(F)$, from the previous lemma, we immediately deduce that the $\mathscr{I}$-cuts, $\mathscr{M}$-cuts and $\mathscr{B}$-cuts are also graph cuts for $M(F)$. Hence, the following theorem which states the equivalence between watershed cuts, $\mathscr{B}$-cuts, $\mathscr{M}$-cuts and $\mathscr{I}$-cuts can be straightforwardly deduced from Lem. 16.

Theorem 17: Let $S \subseteq E$. The four following statements are equivalent:

(i) S is an $\mathscr{I}$-cut for F;
(ii) S is an $\mathscr{M}$-cut for F;
(iii) S is a $\mathscr{B}$-cut for F;
(iv) S is a watershed cut for F.

A major consequence of this theorem is that any algorithm which computes an $\mathscr{I}$-cut, an $\mathscr{M}$-cut or a $\mathscr{B}$-cut also computes a watershed. Conversely any watershed of a map can be obtained as an $\mathscr{I}$-cut, as an $\mathscr{M}$-cut and as a $\mathscr{B}$-cut. In the next section, we propose an algorithm for $\mathscr{M}$-cuts.

F. Linear-time watershed algorithm based on $\mathscr{M}$-kernels

An efficient linear-time algorithm (Algorithm $\mathscr{M}$-kernel) to extract the watershed cuts is proposed. It consists of computing an $\mathscr{M}$-kernel of a map and its minima. Therefore, by Th. 17, the watersheds can be computed by taking the edges which link distinct minima of the $\mathscr{M}$-kernels. The correctness and time-complexity of this algorithm are analyzed. Finally, implementation details

to select "interesting" cuts when several watersheds exist are discussed.

Before presenting Algorithm $\mathscr{M}$-kernel, we recall that $u \in E$ is a border edge for F if the altitude of one of its extremities equals the altitude of u and the altitude of the other one is strictly less than the altitude of u.

Algorithm: $\mathscr{M}$-kernel

Data: (V, E, F): an edge-weighted graph;
Result: F: an M-border kernel of the input map, and (V_M, E_M) its minima.

1 $L \leftarrow \emptyset$;
2 Compute $M(F) = (V_M, E_M)$ and $F^{\ominus}(x)$ for each $x \in V$;
3 **foreach** $u \in E$ *outgoing from* (V_M, E_M) **do** $L \leftarrow L \cup \{u\}$;
4 **while** *there exists* $u \in L$ **do**
5 $\quad L \leftarrow L \setminus \{u\}$;
6 $\quad$ **if** u *is border for* F **then**
7 $\quad\quad x \leftarrow$ the vertex in u such that $F^{\ominus}(x) < F(u)$;
8 $\quad\quad y \leftarrow$ the vertex in u such that $F^{\ominus}(y) = F(u)$;
9 $\quad\quad F(u) \leftarrow F^{\ominus}(x)$; $F^{\ominus}(y) \leftarrow F(u)$;
10 $\quad\quad V_M \leftarrow V_M \cup \{y\}$; $E_M \leftarrow E_M \cup \{u\}$;
11 $\quad\quad$ **foreach** $v = \{y', y\} \in E$ *with* $y' \notin V_M$ **do**
12 $\quad\quad\quad L \leftarrow L \cup \{v\}$; //$v$ is outgoing from $M(F)$

In Algorithm $\mathscr{M}$-kernel, to achieve a linear complexity, the graph (V, E) can be stored as an array of lists which maps to each point the list of all its adjacent vertices. An additional mapping can be used to access in constant time the two vertices which compose a given edge. Nevertheless, for applications to image processing, and when usual adjacency relations are used, these structures do not need to be explicit.
Furthermore, to achieve a linear complexity, the minima of F must be known at each iteration. To this end, in a first step (line 2), the minima of F are computed and represented by two Boolean arrays V_M and E_M, the size of which are respectively $|V|$ and $|E|$. This step can be performed in linear time thanks to classical algorithms [31]. Then, in the main loop (line 4), after each lowering of F (line 9), V_M and E_M are updated (line 10). In order to access, in constant time, the edges which are M-border, the (non-already examined) edges outgoing from the minima are stored in a set L (lines 3 and 12). This set can be, for instance, implemented as a queue. Thus, we obtain the following property.

Property 18: At the end of Algorithm $\mathscr{M}$-kernel, F is an $\mathscr{M}$-kernel of the input function F. Furthermore, Algorithm $\mathscr{M}$-kernel terminates in linear time with respect to $|E|$.

As far as we know, the watershed algorithms available in the literature (*e.g.*, [4], [8], [9], [13], [14], [18]) all require either a sorting step, a hierarchical queue or a data structure to maintain a collection of disjoint sets under the operation of union. On the one hand, the global complexities of a sorting step and of a (monotone) hierarchical queue (*i.e.*, a structure from which the elements can be removed in the order of their altitude) are equivalent [34]: they both run in linear-time only if the range of the weights is sufficiently small. On the other hand, the best complexity for the disjoint set problem is quasi-linear [35]. Therefore, we emphasize that, to the best of our knowledge, the proposed algorithm (together with the one introduced in [16]) is the first watershed algorithm that runs in linear-time whatever the range of the weighting map.

In practice, Algorithm $\mathscr{M}$-kernel runs about 2 times slower than the algorithm proposed in [16] which is as fast as minima computation algorithms. However, Algorithm $\mathscr{M}$-kernel is more flexible. Let us consider a map that contains "non-minima plateaus" (*i.e.*, connected subgraphs with constant altitude). The map F of Fig. 6a illustrates such a situation (see also reference [36] for an in-depth study of such situations). There exist several watersheds of F. More precisely, any set containing a single edge at altitude 3 is a watershed of F. In theory, any of these watersheds can be obtained by Algorithm $\mathscr{M}$-kernel. Nevertheless, in practice, Algorithm $\mathscr{M}$-kernel can be implemented to compute exclusively some particular watersheds. If the set L is implemented as a stack (the last element inserted in L is the first one removed from L), the obtained watershed will be located on the plateaus borders. In this case, the watershed of F computed by Algorithm $\mathscr{M}$-kernel will be either $\{\{b, c\}\}$ or $\{\{f, g\}\}$, depending on the scanning order. On the other hand, if the set L is implemented as a (monotone) priority queue, such as the hierarchical queue proposed in [9], then the obtained watershed will be "centered" (according to the distance induced by G) on the plateaus. In this case, the watershed of F computed by Algorithm $\mathscr{M}$-kernel will be composed by $\{d, e\}$. Figs. 6b,c and d illustrate the differences between the watersheds obtained by these two implementations, on a two-dimensional image. Note that the second implementation of Algorithm $\mathscr{M}$-kernel runs in linear time only if the range of the weights is sufficiently small since it uses a monotone priority queue. Note also that the centering condition neither allows us to uniquely define a watershed (consider *e.g.*, a map with a plateau of even width), nor to compute it order-independently (see [37], [38] for examples of order-independent segmentation methods).

Algorithm $\mathscr{M}$-kernel associates a catchment basin to each minimum. In applications, one does not always need a basin for each minimum. In order to reduce this over-segmentation, some methods in mathematical morphology use the connection value to determine which basins to merge. The next section studies the relation between watersheds and connection value.

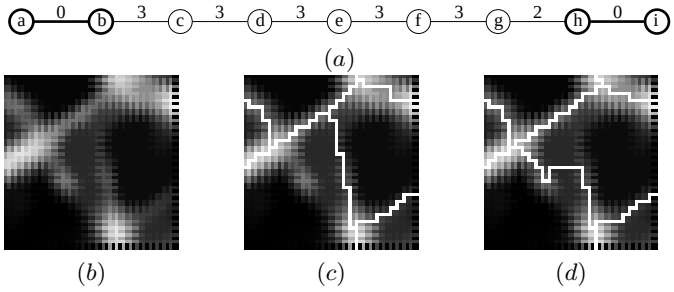


Fig. 6. Illustration of watershed cuts in presence of plateaus. (a) A graph G and a map F which has one plateau at altitude 3. (b) An image representation of an edge-weighted graph (4-adjacency relation) derived from a real-world image (close-up on a microscopic view of a cross-section of a uranium oxyde ceramics). The weight map is obtained by assigning to each edge the minimum of the values, in the original image, of its two extremities and the image representation is obtained by doubling the resolution. (c, d) Two watershed cuts (superimposed in white) obtained by Algorithm $\mathscr{M}$-kernel implemented with respectively a stack and a hierarchical queue.

III. Connection value

From a topographical point of view, the connection value (also called degree of connectivity [39] or fuzzy connectedness [26]

up to an inversion of F [23], [40]) between two minima can be seen as the altitude of the lowest pass between these two minima. It corresponds to the minimal altitude at which one needs to climb in order to reach one minimum from the other. As stated in the introduction, this value is important for morphological region merging methods [20]–[22] which simulate the overflows of catchment basins during a flooding of the topographic surface. We start this section by defining the connection value. Then, we show that any MSF relative to any arbitrary subgraph of G "preserves" the connection values. Thus, knowing the values of the edges in an MSF cut for X, one can recover the connection values between any two components of X. Hence, according to Th. 5, the watershed cuts also "preserve" the connection value.

Definition 19 (connection value): Let $\pi = \langle x_0, \ldots, x_l \rangle$ be a path in G. If π is non-trivial, we set $\Upsilon_F(\pi) = \max\{F(\{x_{i-1}, x_i\}) \mid i \in [1, l]\}$. If π is trivial, we set $\Upsilon_F(\pi) = F^{\ominus}(x_0)$. Let X and Y be two subgraphs of G, we denote by $\Pi(X, Y)$ the set of all paths from X to Y in G. The *connection value between X and Y (in G, for F)* is $\Upsilon_F(X, Y) = \min\{\Upsilon_F(\pi) \mid \pi \in \Pi(X, Y)\}$.

Let X be any subgraph of G. The following theorem asserts that, if the connection value between two components of X is equal to k, then the connection value between the two corresponding components in any MSF relative to X is also k: relative MSFs preserve the connection values. A major consequence of this theorem is that the cuts induced by relative MSFs convey the connection value between the components of the original subgraph.

Theorem 20: Let X be a subgraph of G. If Y is an MSF relative to X, then for any two distinct components A and B of X, we have $\Upsilon_F(A, B) = \Upsilon_F(A', B')$, where A' and B' are the two components of Y such that $A \subseteq A'$ and $B \subseteq B'$.

For example, in Fig. 2a, the connection value between the two minima at altitude 1 is equal to 4. Indeed, $\Upsilon_F(\langle a, e, f, g \rangle) = 4$ whereas the length of any other path from one of these minima to the other is greater than 4. It can be verified that the connection value between the two corresponding components of the MSFs relative to the minima, depicted in Figs. 2c is also 4 (notice in particular that $\Upsilon_F(\langle f, g \rangle) = 4$).

Let $S \subseteq E$ be a watershed cut of F. As a corollary of Th. 20, it may be deduced that the connection value between two distinct catchment basins (*i.e.*, two components of $\overline{S}$) is equal to the connection value between the two corresponding minima of F. Thus, knowing the values of the edges in a watershed of F, one can recover the connection values between the minima of F.

The connection value itself is used to define several important segmentation methods [4], [5], [12]. Hence, Th. 20 invites us to study the links between the watershed and these methods.

IV. Watersheds, shortest-path forests and topological watersheds

In practice, to choose among the numerous segmentation techniques available in the literature the one which will best solve a given problem, it is necessary to understand the differences or links between these techniques [40]–[42]. An interesting feature of the framework settled in this paper is to provide a means to compare, from a mathematical point of view, several methods used for image segmentation. Thanks to relative MSFs and $\mathscr{M}$-kernels, we provide a mathematical comparison between watershed cuts, shortest-path forests (the theoretical basis of the Image Foresting Transform [4] and of the fuzzy connected image segmentation [5], [40]) and topological watersheds [12], [23]. Furthermore, in [43], based on the framework of this paper, a link between min-cuts [2] and watershed cuts is provided.

A. Shortest-path forests

We investigate the links between relative MSFs and shortest-path forests which also constitute an optimization paradigm used for image segmentation. In particular, the image foresting transform [4], the inter-pixel flooding watershed [9], [44], and the relative fuzzy connected image segmentation [5], [26], [27], [40] fall in the scope of shortest-path forests. Intuitively, these methods partition the graph into connected components associated to seed points (also called markers). The component of each seed consists of the points that are "more closely connected" to this seed than to any other. In many cases, in order to define the relation is "more closely connected to", the chosen measure is precisely the connection value, *i.e.*, a path π' is considered as shorter than a path π whenever $\Upsilon_F(\pi') < \Upsilon_F(\pi)$. Then, point x is more closely connected to seed s than to seed s' if the connection value between x and s is less than the connection value between x and s'. Given a set of seed points (or seed graph), the corresponding segmentation can be obtained by an Υ-shortest-path forest, *i.e.*, a shortest-path forest for which Υ defines the length of a path. We show that any MSF relative to a subgraph X is an Υ-shortest-path spanning forest relative to X and that the converse is not true[2]. Furthermore, we prove that both concepts are equivalent whenever X corresponds to the minima of the considered map F. A consequence of this last result is the equivalence between the watersheds of F and the cuts induced by the Υ-shortest-path spanning forests relative to the minima.

Intuitively, a shortest-path forest relative to a subgraph X of G is a forest relative to X which is such that, for each vertex, there exists a path in the forest, which is a shortest path (in G) from this vertex to the subgraph X.

If $x \in V$, to simplify the notation, the graph $(\{x\}, \emptyset)$ is denoted by x. Let X and Y be two subgraphs of G, we say that Y is an *Υ-shortest-path forest relative to X* if Y is a forest relative to X and if, for any $x \in V(Y)$, there exists, from x to X, a path π in Y such that $\Upsilon_F(\pi) = \Upsilon_F(x, X)$. If Y is an Υ-shortest-path forest relative to X and $V(Y) = V$, then Y is an *Υ-shortest-path spanning forest relative to X*, and the cut induced by Y is an *SPF cut for X*.

Let G be the graph in Fig. 7 and F be the associated map. Let X, Y, Z be the bold graphs in Figs. 7a,b and c. The graphs Y and Z are Υ-shortest-path spanning forests relative to X.

Theorem 21: Let X and Y be two subgraphs of G. If Y is an MSF relative to X, then Y is an Υ-shortest-path spanning forest relative to X. Thus, any MSF cut for X is a SPF cut for X.

The converse of Th. 21 is, in general, not true. For example, the graph Z (Fig. 7c), is an Υ-shortest-path spanning forest relative to the graph X (Fig. 7a) whereas it is not an MSF relative to this graph. On the same example (Fig. 7c) we can also observe that, contrarily to relative MSFs, Υ-shortest-path spanning forests do not always preserve the connection value (in the sense of Th. 20). In particular, in Figs. 7a and c, the connection value between the two components of X is equal to 8, whereas the connection

[2]This result has been independently presented in two papers [43], [45] published at a same conference.

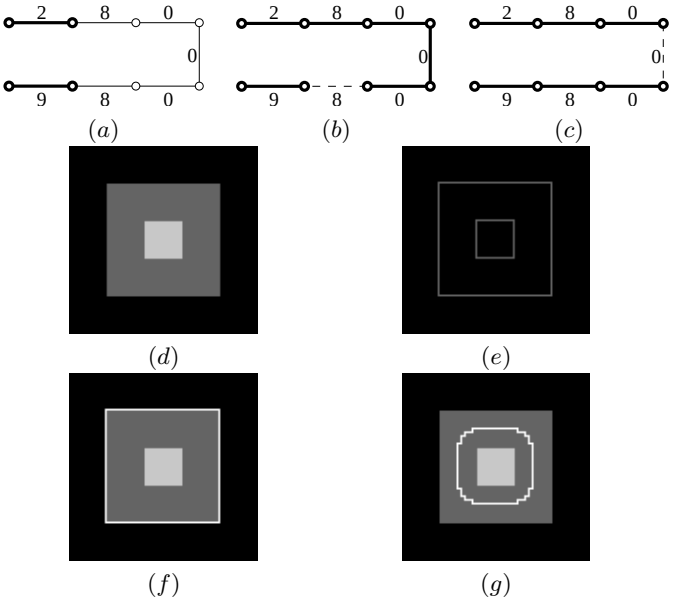


Fig. 7. Links and differences between Υ-shortest-path and minimum spanning forests. First row: a graph G and a map F. The bold subgraphs are: (a), a graph X; (b), an MSF relative to X; (c), an Υ-shortest-path spanning forest relative to X that is not an MSF relative to X. Second row: illustration, on a synthetic image, of the differences between SPF and MSF cuts [see text].

value between the two components of Z is equal to 0. Then, on the contrary of cuts induced by relative MSFs (see for instance Fig. 7b), the cuts induced by Υ-shortest-path spanning forests are not necessarily located on the "crests" of the function. The second row of Fig. 7 illustrates the differences between MSF and SPF cuts on a synthetic 2D image. The image in Fig. 7d is composed of three overlaid squares whose intensities are respectively 0, 100 and 200. From this image an edge-weighted graph (G, F) is derived by considering the 4-adjacency relation and by assigning to each edge $u = \{x, y\}$ the absolute value of the difference of the intensities of x and y. Thus, the weight of any edge that links two pixels belonging to a same zone is equal to 0 whereas the weight of any edge that links two different zones is equal to 100. An image representation of this edge-weighted graph is plotted in Fig. 7e. Let us also consider as a marker a subgraph X of G made of two isolated vertices: the first one is located in the black zone whereas the second one is located at the center of the image. In Fig. 7f and g, two SPF cuts relative to X are superimposed in white to the original image. The first one is furthermore an MSF cut whereas the second one is not.

In fact, if the marker X equals $M(F)$, the equivalence between both concepts can be proved.

Property 22: Let Y be a subgraph of G. A necessary and sufficient condition for Y to be an Υ-shortest-path spanning forest relative to $M(F)$ is that Y is an MSF relative to $M(F)$. Furthermore, a subset of E is an MSF cut for $M(F)$ if and only if it is a SPF cut for $M(F)$.

Whereas the notions of Υ-shortest-path forests and relative MSFs are equivalent when extensions of the minima are considered (Prop. 22), when we consider extensions of arbitrary subgraphs, the relative MSFs satisfy additional properties, such as the preservation of the connection value (Th. 20) or the optimality (in the sense of Def. 4). Relative MSFs is thus a method of choice for marker-based segmentation procedures, an illustration of which is provided in [16].

B. Topological watershed

The topological approach to the watershed [12], [23] is settled in graphs whose vertices are weighted by a function I. It considers a transformation that iteratively lowers the values of I while preserving some topological properties, namely the number of connected components of each lower threshold of I. This transform and its result are called a *W-thinning*; a *topological watershed* being a W-thinning minimal for the $\leq$ relation on maps (see Appendix in the digital library for formal definitions). For instance, the map of Fig. 8e is a topological watershed of the one of Fig. 8d. The *divide* of a topological watershed is the set of all vertices which do not belong to any minimum (see the non-bold vertices in Fig. 8e). A topological watershed and its divide constitute an interesting segmentation which satisfies important properties (see [18], [23], [25]) not guaranteed by most popular watershed algorithms. In particular, in [23], [25], the equivalence between a class of transformations which preserves the connection value and the W-thinnings is proved. Thus, Th. 20 invites us to recover the links between watershed cuts and topological watersheds.

The notion of line graphs presented below (see [15], [46], [47]) provides a way to automatically infer definitions and properties from vertex-weighted graphs to edge-weighted graphs.

Definition 23 (line graph): The *line graph of* $G = (V, E)$ is the graph (E, Γ), such that $\{u, v\}$ belongs to Γ whenever $u \in E$, $v \in E$, and u and v are adjacent, *i.e.*, $|u \cap v| = 1$.

To each graph G whose edges are weighted by a map F, we can associate its line graph G'. The vertices of G' are weighted by F and thus any transformation of F can be performed either in G or in G'. Fig. 8 illustrates such a procedure. Let G be the graph depicted in (a), (b) and (c). The line graph of G is depicted in (d), (e) and (f). The map shown in (b, e) is a topological watershed of the one shown in (a, d) and the map in (c, f) is a $\mathscr{B}$-kernel.

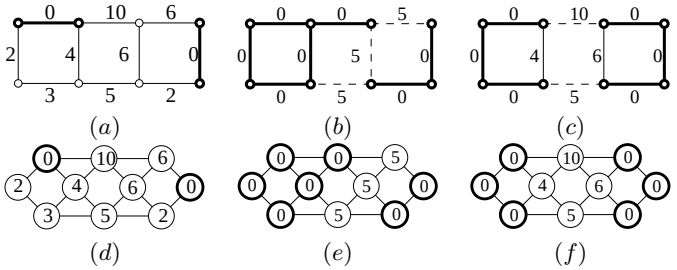
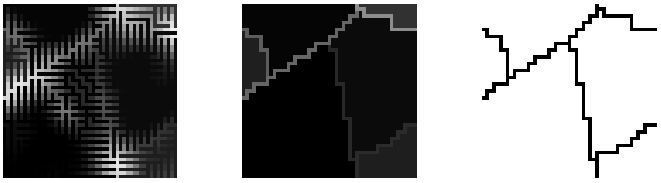


Fig. 8. Illustration of line graphs and topological watersheds. The graph in (d) (resp. (e), (f)) is the line graph of the one in (a) (resp. (b), (c)). The minima of the associated functions are depicted in bold.; (b, e): a topological watershed of (a, d); (c, f): a $\mathscr{B}$-kernel of (a, d) which is also a W-thinning of (a, d).

Definition 24 (topological cut): Let $S \subseteq E$ be a cut for $M(F)$. We say that S is a *topological cut for* F if there exists a W-thinning H of F, in the line graph of G, such that S is the set of all edges of G which are adjacent to two distinct minima of H.

Theorem 25: Let H be a map from E to $\mathbb{R}$. If H is a $\mathscr{B}$-thinning of F in G, then H is a W-thinning of F in the line graph of G. Moreover, any $\mathscr{B}$-cut for F is a topological cut for F.

The previous property is illustrated in Fig. 8 where the map depicted in Fig. 8c is a $\mathscr{B}$-thinning of F (Fig. 8a), thus a W-thinning of F. The converse of Th. 25 is not true. The map H (Fig. 8b) is a topological watershed of F but it is not a $\mathscr{B}$-kernel of F. Indeed, there is no MSF relative to the minima of F associated to the cut produced by the topological watershed H. Observe, in particular, that the produced cut is not located on the highest "crests" of the original map F. Fig. 9a shows an image representation of a $\mathscr{B}$-kernel H obtained from the map F represented in Fig. 6a and from which we derived the cut shown Fig. 6b. Fig. 9b is a topological watershed of H which, by Th. 25, is also a topological watershed of F. Fig. 9c represents the watershed cut associated with these two maps.



(a) (b) (c)

Fig. 9. Image representation of: (a), a $\mathscr{B}$-kernel of Fig. 6b; (b), a topological watershed of (a); and (c), the watershed cut associated to both (a) and (b).

An important consequence of Th. 25 is that $\mathscr{B}$-cuts (hence, by Th. 16, watershed cuts) directly inherit all the properties of W-thinnings proved for vertex-weighted graphs [18], [23], [25].

In recent papers [15], [47], [48], we have studied and proposed solutions to some of the problems encountered by region merging methods which consider frontiers made of vertices as initial segmentations. In particular, we have introduced an adjacency relation on $\mathbb{Z}^n$ which is adapted for region merging. An important property (Prop. 54 in [47]) is that the induced grids, called the perfect fusion grids, are line graphs. If we consider a map which assigns a weight to the vertices of such a grid, then the set of definitions and properties given in this paper are still valid. Thus, the perfect fusion grids constitute an interesting alternative for defining a watershed which is based on vertices and which satisfies the drop of water principle.

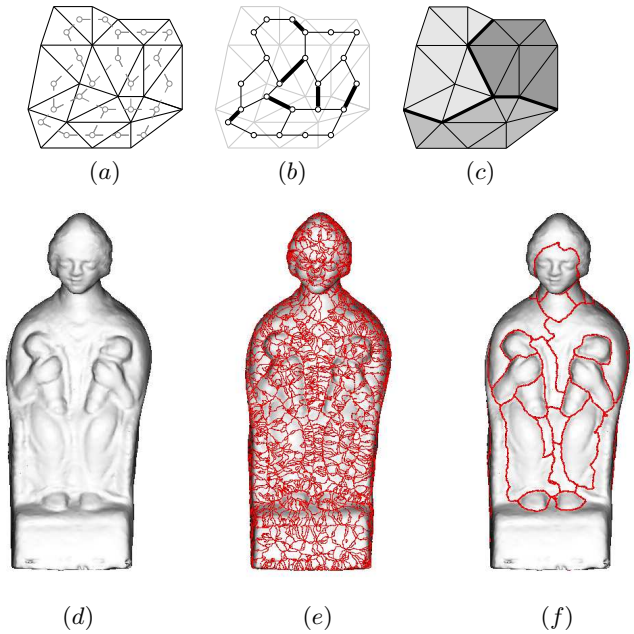
V. Illustration to segmentation

In Section I-A, we present different ways to derive edge-weighted graphs from grayscale images. In [16], we showed how to use these graphs to automatically segment an image into a predefined number of regions by coupling watershed cuts with connected filters [49]. We also illustrated the use of relative MSF as a marker-based procedure for grayscale image segmentation.

In this section, we illustrate the versatility of the proposed framework to perform segmentation on different kinds of geometric objects. Firstly, we show how to segment triangulated surfaces by watershed cuts, and secondly we apply the watershed cuts to the segmentation of diffusion tensors images, which are medical images associating a tensor to each voxel.

A. Surface segmentation

3D shape acquisition and digitizing have received more and more attention for a decade, leading to an increasing amount of 3D surface-models (or meshes) such as the one in Fig. 10d. In a recent work [50], a new search engine has been proposed for indexing and retrieving objects of interests in a database of meshes (EROS 3D) provided by the French Museum Center for Research. One key idea of this search engine is to use region descriptors rather than global shape descriptors. In order to produce such descriptors, it is then essential to obtain meaningful mesh segmentations.



(a) (b) (c)

(d) (e) (f)

Fig. 10. Surface segmentation by watershed. (a): A mesh in black and its associated graph in gray. (b): A cut on this graph (in bold); and (c), the corresponding segmentation of the mesh. (d): Rendering of the mesh of a sculpture. (e): A watershed (in red) of a map F which behaves like the inverse of the mean curvature and, in (f), a watershed of a filtered version of F. The mesh shown in (d) is provided by the French Museum Center for Research.

Informally, a *mesh* M in the 3D Euclidean space is a set of *triangles*, *sides* of triangles and *points* such that each side is included in exactly two triangles (see Fig. 10a). In order to perform a watershed cut on such a mesh, we build a graph $G = (V, E)$ whose vertex set V is the set of all triangles in M and whose edge set E is composed by the pairs $\{x, y\}$ such that x and y are two triangles of M that share a common side (see Fig. 10a).

To obtain a segmentation of the mesh M thanks to a watershed cut, we need to weight the edges of G (or equivalently the sides of M) by a map whose values are high around the boundaries of the regions that we want to separate. We have found that the interesting contours on the EROS 3D meshes are mostly located on concave zones. Therefore, we weight the edges of G by a map F which behaves like the inverse of the mean curvature of the surface (see [50] for more details). Then, we can compute a watershed cut (in bold in Fig. 10b) which leads to a natural and accurate mesh segmentation in the sense that the "borders" of the regions are made of sides of triangles (in bold in Fig. 10c) of high curvature.

The direct application of this method on the mesh shown Fig. 10d leads to a strong over-segmentation (Fig. 10e) due to the huge number of local minima. By using the methodology introduced in mathematical morphology and our notions, we can extract all the minima which have a dynamics [24] greater than a predefined threshold (here 50) and suppress all other minima by

a geodesic reconstruction [31]. A watershed cut of the map F' (obtained from F with such a filtering step) is depicted in Fig. 10f.

B. Segmentation of diffusion tensor images

In the medical context, *Diffusion Tensor Images (DTIs)* [51] provide a unique insight into oriented structures within tissues. A *DTI* T maps the set of voxels $V \subseteq \mathbb{Z}^3$ (*i.e.,* V is a cuboid of $\mathbb{Z}^3$) into the set of 3×3 tensors (*i.e.,* 3×3 symmetric positive definite matrices). The value $T(x)$ of a DTI T at a voxel $x \in V$ describes the diffusion of water molecules at x. For instance, the first eigenvector of $T(x)$ (*i.e.,* the one whose associated eigenvalue is maximal) provides the principal direction of water molecules diffusion at point x and its associated eigenvalue gives the magnitude of the diffusion along this direction. Since water molecules highly diffuse along fiber tracts and since the white matter of the brain is mainly composed of fiber tracts, DTIs are particularly adapted to the study of brain architecture. Fig. 11a shows a representation of a cross-section of a brain DTI where the tensors are represented by ellipsoids. Indeed, the datum of a tensor is equivalent to the one of an ellipsoid. In the brain, the corpus callosum is an important structure made of fiber tracts connecting homologous areas of each hemisphere. In order to track the fibers that pass through the corpus callosum, it is necessary to segment it first. The next paragraph briefly reviews how to reach this goal thanks to watershed cuts (see [52] for more details).

We consider the graph $G = (V, E)$ induced by the 6-adjacency and defined by $\{x, y\} \in E$ iff $x \in V, y \in V$ and $\Sigma_{i \in \{1,2,3\}} |x_i - y_i| = 1$, where $x = (x_1, x_2, x_3)$ and $y = (y_1, y_2, y_3)$. In order to weight any edge $\{x, y\}$ of G by a dissimilarity measure between the tensors $T(x)$ and $T(y)$, we choose the Log-Euclidean distance which is known to satisfy an interesting property of invariance by similarity [53]. Then, we associate to each edge $\{x, y\} \in E$ the value $F(\{x, y\}) = \| \log(T(x)) - \log(T(y)) \|$, where $\log$ denotes the matrix logarithm and $\|.\|$ the Euclidean (sometimes also called Frobenius) norm on matrices. To segment the corpus callosum in this graph, we extract (thanks to a statistical atlas, see [52]), markers for both the corpus callosum and its background and we compute an MSF-cut for these markers. An illustration of this procedure is shown in Fig. 11.

CONCLUSION

Fig. 12 summarizes the main results presented in [16] and in this paper. In the framework of edge-weighted graphs, we introduced the *watershed cuts*. Through seven equivalence relations and two original efficient algorithms, we established strong links between three important paradigms: topographical, thinning and optimality paradigms. As far as we know, this constitutes the only discrete framework in which all these properties hold true.

On the topographical side, we proved in [16] that the *watershed cuts* can be equivalently defined by their "catchment basins" (through a steepest descent property formalized in the definition of a *basin cut*, see Definition 5 in [16]) or by their "dividing lines" (through a formalization of the intuitive "drop of water principle"). From the notion of a *basin cut*, we derived in [16] a first efficient watershed algorithm.

On the thinning side, we introduced a new paradigm to characterize and compute the watershed cuts. A thinning consists of iteratively lowering the values of the edges that satisfy a certain property. We proposed three different properties for selecting the edges which are to be lowered. The corresponding three transforms extend the minima of the original map in a way such that the set of edges (called $\mathscr{B}$*-cuts*, $\mathscr{M}$*-cuts* and $\mathscr{I}$*-cuts*) linking two minima of the transformed map constitute precisely a watershed cut of the original map. Conversely, any watershed cut is necessarily a $\mathscr{B}$-cut, an $\mathscr{M}$-cut and an $\mathscr{I}$-cut. The first of these thinnings uses a purely local strategy to detect the edges which are to be lowered and, therefore, it is well suited to parallel implementations. The second one leads to a flexible sequential linear-time (with respect to the number of edges) watershed algorithm. Finally, the third one establishes the link between the watershed cuts and the popular immersion scheme which fall in the topographical category.

On the optimization side, we showed the equivalence between the watershed cuts and the separations (called *MSF-cuts* and *SPF-cuts*) induced by two optimal structures: the minimum spanning forests and the Υ-shortest paths forests relative to the minima.

On the algorithmic side, we would like to emphasize that the two proposed algorithms both run in linear time whatever the range of the input function. To the best of our knowledge, these are the first watershed algorithms satisfying such a property.

Finally, we have shown that any watershed cut allows for recovering the connection value between the minima of the original map and thus that it is a topological cut. In mathematical morphology, this property plays a fundamental role for defining watershed-based hierarchical segmentation methods [20], [22].

Future works will be focused, on the one hand, on the above mentioned hierarchical segmentation schemes (including *geodesic saliency of watershed contours* [22] and *incremental MSFs*) and also on watersheds in weighted simplicial complexes, an image representation adapted to the study of topological properties. On the other hand, we will study a new minimum spanning tree algorithm based on watersheds.

REFERENCES

[1] C. Zahn, "Graph-theoretical methods for detecting and descibing gestalt clusters," *IEEE Transactions on Computers*, vol. C-20, no. 1, pp. 99–112, 1971.

[2] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 11, pp. 1222–1239, 2001.

[3] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 888–905, 2000.

[4] A. X. Falcão, J. Stolfi, and R. de Alencar Lotufo, "The image foresting transform: theory, algorithm and applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, pp. 19–29, 2004.

[5] P. K. Saha and J. K. Udupa, "Relative fuzzy connectedness among multiple objects: Theory, algorithms, and applications in image segmentation," *Comput. Vis. Image Underst.*, vol. 82, pp. 42–56, 2001.

[6] L. Grady, "Random walks for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 11, pp. 1768–1783, 2006.

[7] S. Beucher and C. Lantuéjoul, "Use of watersheds in contour detection," in *Proc. International Workshop on Image Processing Real-Time Edge and Motion Detection/Estimation*, 1979.

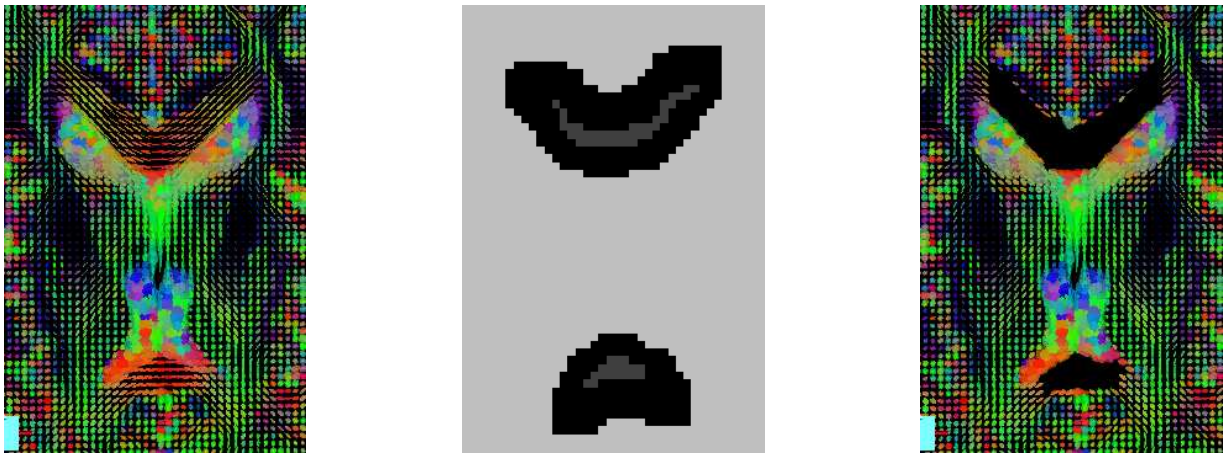
[8] L. Vincent and P. Soille, "Watersheds in digital spaces: An efficient algorithm based on immersion simulations," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 13, no. 6, pp. 583–598, June 1991.

[9] F. Meyer, "Un algorithme optimal de ligne de partage des eaux," in *Proc. 8ème Congrès AFCET*, 1991, pp. 847–859.

[10] ——, "Topographic distance and watershed lines," *Signal Processing*, vol. 38, no. 1, pp. 113–125, 1993.

[11] L. Najman and M. Schmitt, "Watershed of a continuous function," *Signal Processing*, vol. 38, no. 1, pp. 68–86, 1993.

[12] M. Couprie and G. Bertrand, "Topological grayscale watershed transform," in *Procs. SPIE Vision Geometry V*, vol. 3168, 1997, pp. 136–146.



(a) (b) (c)

Fig. 11. Diffusion tensor images segmentation. (a): A close-up on a cross-section of a 3D brain DTI. (b): Image representation of the markers (same cross-section as (a)), obtained from a statistical atlas, for the corpus callosum (dark gray) and for its background (light gray) (c): Segmentation of the corpus callosum by an MSF-cut for the markers. The tensors belonging to the component of the MSF which extends the marker labelled "corpus callosum" are removed from the initial DTI: thus the corresponding voxels appear black.

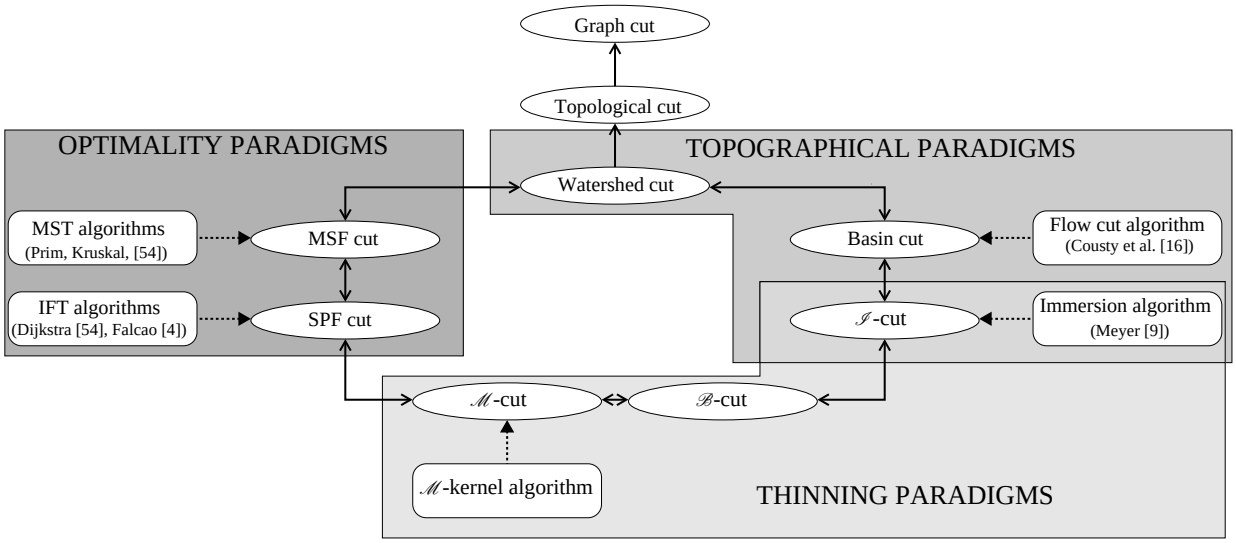


Fig. 12. Summary of the properties on cuts for the regional minima of a map. In the figure, $N \rightarrow N'$ means that the notion N is a particular case of the notion N', hence, $N \leftrightarrow N'$ means that the notions N and N' are equivalent; $A \dashrightarrow N$ means that the notion N can be computed thanks to algorithm A.

[13] A. Meijster and J. Roerdink, "A disjoint set algorithm for the watershed transform," in *Proc. Eusipco : European signal processing conference*, 1998, pp. 1669–1672.

[14] A. Bieniek and A. Moga, "A connected component approach to the watershed segmentation," in *Mathematical Morphology and its Applications to Image and Signal Processing, proc. ISMM'98*, 1998, pp. 215–222.

[15] J. Cousty, M. Couprie, L. Najman, and G. Bertrand, "Weighted fusion graphs: merging properties and watersheds," *Discrete Appl. Math.*, vol. 156, no. 15, pp. 3011–3027, 2008.

[16] J. Cousty, G. Bertrand, L. Najman, and M. Couprie, "Watershed cuts: minimum spanning forests and the drop of water principle," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2009, to appear.

[17] F. Meyer, "Minimum spanning forests for morphological segmentation," in *Proc. 2nd international conference on Mathematical Morphology and its Applications to Image Processing*, September 1994, pp. 77–84.

[18] M. Couprie, L. Najman, and G. Bertrand, "Quasi-linear algorithms for the topological watershed," *J. Math. Imaging Vis.*, vol. 22, no. 2-3, pp. 231–249, 2005.

[19] A. Bleau and L. J. Leon, "Watershed-based segmentation and region merging," *Comput. Vis. Image Underst.*, vol. 77, no. 3, pp. 317–370, 2000.

[20] S. Beucher, "Watershed, hierarchical segmentation and waterfall algorithm," in *Proc. 2nd international conference on Mathematical Morphology and its Applications to Image Processing*, 1994, pp. 69–76.

[21] F. Meyer, "The dynamics of minima and contours," in *Mathematical Morphology and its Application to Image and Signal Processing*, 1996, pp. 329–336.

[22] L. Najman and M. Schmitt, "Geodesic saliency of watershed contours and hierarchical segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 18, no. 12, pp. 1163–1173, December 1996.

[23] G. Bertrand, "On topological watersheds," *J. Math. Imaging Vis.*, vol. 22, no. 2-3, pp. 217–230, May 2005.

[24] ——, "On the dynamics," *Image Vision Comput.*, vol. 25, no. 4, 2007.

[25] L. Najman, M. Couprie, and G. Bertrand, "Watersheds, mosaics and the emergence paradigm," *Discrete Appl. Math.*, vol. 147, no. 2-3, pp. 301–324, 2005.

[26] J. K. Udupa and S. Samarsekara, "Fuzzy connectedness and object definition: Theory, algorithms, and applications in image segmentation," *Graphical Models and Image Processing*, vol. 58, pp. 246–261, 1996.

[27] J. K. Udupa, P. K. Saha, and R. A. Lotufo, "Relative fuzzy connectedness and object definition: Theory, algorithms, and applications in image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 11,

pp. 1485–1500, 2002.
[28] R. Diestel, *Graph Theory*, ser. Graduate Texts in Mathematics. Springer, 1997.
[29] T. Kong and A. Rosenfeld, "Digital topology: Introduction and survey," *Comput. Vision Graph. Image Process.*, vol. 48, no. 3, pp. 357–393, 1989.
[30] R. Englert and W. Kropatsch, "Image structure from monotonic dual graph," in *AGTIVE'99*, vol. LNCS 1779, 2000, pp. 297–308.
[31] P. Soille, *Morphological Image Analysis*. Springer-Verlag, 1999.
[32] A. N. Moga, T. Viero, M. Gabbouj, M. Nölle, G. Schreiber, and H. Burkhardt, "Parallel watershed algorithm based on sequential scanning," in *Proc. IEEE Workshop on Nonlinear Signal and Image Processing*, 1995.
[33] R. Prim, "Shortest connection networks and some generalizations," *Bell Syst. Tech. J.*, vol. 36, pp. 1389–1401, 1957.
[34] M. Thorup, "On RAM priority queues," in *Proc. 7th ACM-SIAM Symposium on Discrete Algorithms*, 1996, pp. 59–67.
[35] R. Tarjan, "Efficiency of a good but not linear set union algorithm," *Journal of the ACM*, vol. 22, pp. 215–225, 1975.
[36] R. Audigier and R. Lotufo, "Uniquely-determined thinning of the tie-zone watershed based on label frequency," *J. Math. Imaging Vis.*, vol. 27, no. 2, pp. 157–173, February 2007.
[37] J. Serra, "A lattice approach to image segmentation," *J. Math. Imaging Vis.*, vol. 24, no. 1, pp. 83–130, February 2006.
[38] P. Soille, "Constrained connectivity for hierarchical image partitioning and simplification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 7, pp. 1132–1145, 2008.
[39] A. Rosenfeld, "On connectivity properties of grayscale pictures," *Pattern Recogn.*, vol. 16, pp. 47–50, 1983.
[40] R. Audigier and R. Lotufo, "Duality between the watershed by image foresting transform and the fuzzy connectedness segmentation approaches," in *Proc. SIBGRAPI*, 2006, pp. 53–60.
[41] K. C. Ciesielski and J. K. Udupa, "A general theory of image segmentation: level set segmentation in the fuzzy connectedness framework," in *Proc. SPIE, Medical Imaging*, vol. 6512, 2007.
[42] A. Sinop and L. Grady, "A seeded image segmentation framework unifying graph cuts and random walker which yields a new algorithm," in *Computer Vision, proc. ICCV'07, IEEE*, 2007, pp. 1–8.
[43] C. Allène, J.-Y. Audibert, M. Couprie, J. Cousty, and R. Keriven, "Some links between min-cuts, optimal spanning forests and watersheds," in *Mathematical Morphology and its Applications to Signal and Image Processing, Proc. 8th International Symposium on Mathematical Morphology*, 2007, pp. 253–264.
[44] S. Beucher and F. Meyer, "The morphological approach to segmentation: the watershed transformation," *E. Dougherty (Ed.), Mathematical Morphology in Image Processing, Marcel Decker*, pp. 443–481, 1993.
[45] R. Audigier and R. Lotufo, "Watershed by image foresting transform, tie-zone, and theoretical relationships with other watershed definition," in *Mathematical Morphology and its Applications to Signal and Image Processing, Proc. 8th International Symposium on Mathematical Morphology*, 2007, pp. 277–288.
[46] C. Berge, *Graphs and hypergraphs*. Elsevier Science Ltd, 1985.
[47] J. Cousty, G. Bertrand, M. Couprie, and L. Najman, "Fusion graphs: merging properties and watersheds," *J. Math. Imaging Vis.*, vol. 30, no. 1, pp. 87–104, 2008. [Online]. Available: http://igm.univ-mlv.fr/LabInfo/rapportsInternes/2005/04.v2.pdf
[48] J. Cousty and G. Bertrand, "Uniqueness of the perfect fusion grid on Z^d," *J. Math. Imaging Vis.*, 2009, to appear. [Online]. Available: http://igm.univ-mlv.fr/LabInfo/rapportsInternes/2008/01.pdf
[49] J. Serra and L. Vincent, "An overview of morphological filtering," *Circuits Systems Signal Process*, vol. 11, no. 1, pp. 48–107, 1992.
[50] M. Alcoverro, S. Philipp-Foliguet, M. Jordan, L. Najman, and J. Cousty, "Region-based artwork indexing and classification," in *Proc. 2-nd 3DTV-conference*. IEEE, 2008, pp. 393–396.
[51] P. J. Basser, J. Mattiello, and D. LeBihan, "MR diffusion tensor spectroscopy and imaging." *Biophys. J.*, vol. 66, no. 1, pp. 259–267, 1994.
[52] J. Cousty, P. Fillard, and X. Pennec, "Watersheds and statistical atlases for brain DTI analysis," in preparation.
[53] V. Arsigny, P. Fillard, X. Pennec, and N. Ayache, "Log-Euclidean metrics for fast and simple calculus on diffusion tensors," *Magnetic Resonance in Medicine*, vol. 56, no. 2, pp. 411–421, August 2006.
[54] T. H. Cormen, C. Leiserson, and R. Rivest, *Introduction to algorithms, second edition*. MIT Press, 2001.

Jean Cousty received his Ingénieur's degree from the École Supérieure d'Ingénieurs en Électrotechnique et Électronique (ESIEE Paris, France) in 2004 and the Ph.D. degree from the Université de Marne-la-Vallée (France) in 2007. After a one-year post-doctoral 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 Institut Gaspard Monge, Université Paris-Est Marne-la-Vallée. His current research interests include medical image analysis and discrete mathematics.

Gilles Bertrand received his Ingénieur's degree from the École Centrale des Arts et Manufactures in 1976. Until 1983 he was with the Thomson-CSF company, where he designed image processing systems for aeronautical applications. He received his Ph.D. from the École Centrale in 1986. He is currently teaching and doing research with the Informatics Department, ESIEE, Paris, and with the Institut Gaspard Monge, Université Paris-Est Marne-la-Vallée. His research interests are image analysis, pattern recognition, mathematical morphology and digital topology.

Laurent Najman received the Habilitation à Diriger les Recherches in 2006 from University 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 Institut Gaspard Monge, Université Paris-Est Marne-la-Vallée. His current research interest is discrete mathematical morphology.

Michel Couprie received his Ingénieur's degree from the Ecole Supérieure d'Ingénieurs en Electrotechnique et Electronique (Paris, France) in 1985, the Ph.D. degree from the Pierre et Marie Curie University (Paris, France) in 1988, and the Habilitation à Diriger des Recherches in 2004 from the University of Marne-la-Vallée (France). Since 1988 he has been working in ESIEE where he is a Professor. He is a member of the Informatics Department, ESIEE, Paris, and of Institut Gaspard Monge, Université Paris-Est Marne-la-Vallée. His current research interests include image analysis and discrete matematics.

PROOFS

This appendix provides the proofs of the properties given in this article.

Proofs of Section II

In order to prove Prop. 10, we use a characterization of the MSFs relative to the minima of a map which involve paths with steepest descent.

Let $\pi = \langle x_0, \ldots, x_\ell \rangle$ be a path in G. The path π is *a path with steepest descent for* F if, for any $i \in [1, \ell]$, $F(\{x_{i-1}, x_i\}) = F^\ominus(x_{i-1})$.

Let $\pi = \langle x_0, \ldots, x_\ell \rangle$ be a path in G. We say that π is a *simple path* if for any two distinct i and j in $[0, \ell]$, $x_i \neq x_j$. We say that π is an *$\mathcal{M}$-path (for F)* if π is a simple path, if x_ℓ is a vertex of $M(F)$ and if none of $x_0, \ldots, x_{\ell-1}$ is a vertex of $M(F)$. Remark that an $\mathcal{M}$-path does not contain any edge of $M(F)$. Furthermore, it may be seen that if Y is a forest relative $M(F)$, there exists a unique $\mathcal{M}$-path from each vertex of Y.

Lemma 26 (Lemme 8 in [16]): Let X be a spanning forest relative to $M(F)$. The graph X is an MSF relative to $M(F)$ if and only if, for any x in V, there exists a path in X from x to $M(F)$ which is a path with steepest descent for F.

Lemma 27: Let $H \in \mathcal{F}$. If H is a $\mathscr{B}$-thinning of F, then any forest relative to $M(H)$ is a forest relative to $M(F)$.

Proof: Let u be a border edge for F and let H be the lowering of F at u. We first prove the property for H. Then, Lem. 27 can be easily established by induction. If u is not an edge of $M(H)$ then $M(H) = M(F)$: the proof is done. Suppose now that u is an edge of $M(H)$. Let $u = \{x, y\}$ with $F^\ominus(x) \geq F^\ominus(y)$. The fact that u is border for F implies $F(u) = F^\ominus(x)$ and $F(u) > F^\ominus(y)$. Thus, u is not an edge of $M(F)$ and x cannot belong to an edge of $M(F)$ (otherwise we would have $F^\ominus(x) < F(u)$). Therefore, x is not a vertex of $M(F)$. The edge u belongs to S, the edge set of one minimum of H. Since $H(u) = F^\ominus(y)$ and $F(u) > F^\ominus(y)$ (by definition of a lowering at a border edge), there is an edge $v \neq u$ which contains y such that $F(v) = H(v) = F^\ominus(y) = H(u)$. Necessarily v belongs to S. Hence, $S \setminus u \neq \emptyset$ and it may be seen that $S \setminus u$ is exactly the edge set of a minimum of F. Thus, y is a vertex of $M(F)$ and $M(H)$ is an extension of $M(F)$. Furthermore, since x is not a vertex of $M(F)$, any cycle in $M(H)$ is also a cycle in $M(F)$. Thus, from the very definition of a forest, $M(H)$ is a forest relative to $M(F)$ and any forest relative to $M(H)$ is also a forest relative to $M(F)$. ■

The next lemma follows straightforwardly from the definition of a path with steepest descent.

Lemma 28: If $\langle x_0, \ldots, x_\ell \rangle$ and $\langle x_\ell, \ldots, x_m \rangle$ are two paths with steepest descent for F, then $\pi = \langle x_0, \ldots, x_m \rangle$ is a path with steepest descent for F.

Lemma 29: Let H be a $\mathscr{B}$-thinning of F.
(i) For any vertex x of a minimum of H, there exists a path in $M(H)$ from x to $M(F)$ which is a path with steepest descent for F.
(ii) Any $\mathcal{M}$-path (for H), with steepest descent for H is a path with steepest descent for F.

Proof: Let us first suppose that H is the lowering of F at a border edge u for F.
(i) Let x and y be the two vertices in u. If none of x and y is a vertex of $M(F)$, then $M(F) = M(H)$ and the proof is trivial. Suppose that y is a vertex of $M(F)$. Since u is a border edge, $F^\ominus(x) = F(u)$. Thus, $\langle x, y \rangle$ is a path in $M(H)$ with steepest descent for F. Let z be any vertex of $M(H)$, $z \neq x$. Necessarily z is also a vertex of $M(F)$. Hence, $\langle z \rangle$ is a path in $M(H)$ from z to $M(F)$ with steepest descent for F.
(ii) The property is verified for any trivial path. Let us consider the case of non-trivial paths. Let $x_0 \in V \setminus V(M(H))$ and let $\pi = \langle x_0, \ldots, x_\ell \rangle$ be an $\mathcal{M}$-path (for H) with steepest descent for H. Since x_ℓ is a vertex of $M(H)$ and $\{x_{\ell-1}, x_\ell\}$ is not an edge of $M(H)$, by the very definition of a minimum, $H(\{x_{\ell-1}, x_\ell\}) > H(x_\ell)$. Hence, from the definition of a lowering, we deduce that $u \neq \{x_{\ell-1}, x_\ell\}$.
Suppose that there exists $i \in [1, \ell - 1]$ such that $u = \{x_{i-1}, x_i\}$. As π is a path with steepest descent for H, $F^\ominus(x_i) = H(\{x_i, x_{i+1}\})$. By the very definition of a lowering, $\{x_{i-1}, x_i\}$ is the only edge of G whose altitude is different for F and H. Thus, $F(\{x_i, x_{i+1}\}) = H(\{x_i, x_{i+1}\}) = F^\ominus(x_i)$. By definition of $F^\ominus$, $F^\ominus(x_i) \leq F(\{x_i, x_{i+1}\})$, hence $F^\ominus(x_i) \leq F^\ominus(x_i)$. Since H is a lowering of F, $F^\ominus(x_i) \leq F^\ominus(x_i)$. Hence, $F^\ominus(x_i) = F^\ominus(x_i) = F(\{x_i, x_{i+1}\})$. Therefore, since $F(\{x_{i-1}, x_i\}) > H(\{x_{i-1}, x_i\})$, necessarily $F(\{x_{i-1}, x_i\}) > F^\ominus(x_i)$ and since u is border for F, $F^\ominus(x_{i-1}) = F(\{x_{i-1}, x_i\})$. Furthermore, for any $v \in E$, $v \neq u$, $F(v) = H(v)$. Thus, in this case, π is a path with steepest descent for F.
Suppose now that for any $i \in [1, \ell]$, $u \neq \{x_{i-1}, x_i\}$. By definition of a lowering $F(u) > H(u)$, hence, for any $i \in [0, \ell]$, $F^\ominus(x_i) = F^\ominus(x_i)$. Thus π is a path with steepest descent for F.
By induction on (i) and (ii) and thanks to Lem. 28, it may be seen that Lem. 29 holds true for any $\mathscr{B}$-thinning of F. ■

Lemma 30: There is no border edge for F if and only if V is the vertex set of $M(F)$.

Proof: (i) Suppose that V is not the vertex set of $M(F)$. Then, there exists $x_0 \in V$ which is not a vertex of $M(F)$. Since (V, E) is finite, there exists an $\mathcal{M}$-path $\pi = \langle x_0, \ldots, x_\ell \rangle$ with steepest descent for F. Thus, $F^\ominus(x_{\ell-1}) = F(\{x_{\ell-1}, x_\ell\})$. Since π is an $\mathcal{M}$-path, $F^\ominus(x_\ell) < F(\{x_{\ell-1}, x_\ell\})$. Hence, $\{x_{\ell-1}, x_\ell\}$ is a border edge for F.
(ii) Suppose that there exists $u = \{x, y\}$ which is a border edge for F. Without loss of generality, assume that $F^\ominus(x) = F(u)$ and $F^\ominus(y) < F(u)$. There is no minimum of F whose vertex set contains x since $F^\ominus(x) = F(u)$ and since there is an edge that contains y whose altitude is strictly less than the one of u. Thus, V is not the vertex set of $M(F)$. ■

Proof: [of Prop. 10] (i) Let X be an MSF relative to $M(H)$ for H. By Lem. 27, X is a (spanning) forest relative to $M(F)$. We will prove that for any point x_0 in V, there exists in X a path from x_0 to $M(F)$ which is a path with steepest descent for F. Thus, by Lem. 26, this will establish the first part of Prop. 10. From Lem. 26, it may be seen that there exists in X an $\mathcal{M}$-path (for H), denoted by $\pi = \langle x_0, \ldots, x_\ell \rangle$, which is a path with steepest descent for H. By Lem. 29.ii, π is a path with steepest descent for F. Since x_ℓ is a vertex of $M(H)$, by Lem. 29.i, there exists in $M(H)$ a path $\pi' = \langle x_\ell, \ldots, x_m \rangle$ from x_ℓ to $M(F)$ which is a path with steepest descent for F. Since X is an extension of $M(H)$, $M(H) \subseteq X$. Hence, π' is a path in X. Moreover, π is by construction a path in X. Therefore, $\pi'' = \langle x_0, \ldots, x_m \rangle$, is a path in X. Since both π and π' are paths in X with steepest descent for F, by Lem. 28, π'' is also a path in X with steepest descent for F, which, by construction, is a path from x_0 to $M(F)$.
(ii) Suppose that H is a $\mathscr{B}$-kernel of F. From

Lem. 30, $V(M(H)) = V$. Then, any MSF relative to $M(H)$ is equal to $M(H)$. Hence, from (i), we prove (ii). ■

Proof: [of Prop. 15] Let $u = \{x, y_0\} \in E$, with x being a vertex of $M(F)$, be an immersion edge for F. By definition of a minimum, we have $F(u) > F^{\ominus}(x)$. Let $\pi = \langle y_0, \ldots, y_\ell \rangle$ be any $\mathcal{M}$-path with steepest descent for F. It may be seen that $F(\{y_{\ell-1}, y_\ell\}) \leq F^{\ominus}(y_0)$. Since π is an $\mathcal{M}$-path, $\{y_{\ell-1}, y_\ell\}$ is outgoing from $M(F)$. By hypothesis, $F(u) \leq F(\{y_{\ell-1}, y_\ell\})$. Thus, $F(u) \leq F^{\ominus}(y_0)$ and since $y_0 \in u$, necessarily $F(u) = F^{\ominus}(y_0)$. Hence, u is a border edge for F. ■

The following lemma is used to prove Lem. 16. The proof is similar to the one of Lem. 30 and, thus, omitted.

Lemma 31: There is no M-border edge (resp. immersion edge) for F if and only if V is the vertex set of $M(F)$.

Thanks to the construction presented in Sec. III.B of [16], the following lemmas can be derived from basic results on minimum spanning trees (see, in particular Th. 23.1, p. 563, in [54] in order to prove Lem. 33).

Let $X \subseteq G$, $u \in E(X)$. We write $X \backslash u$ for $(V(X), E(X) \backslash \{u\})$. Let $v \in E \backslash E(X)$. We write $X \cup v$ for the graph $(V(X) \cup v, E(X) \cup \{v\})$.

Lemma 32: Let X be a subgraph of G and Y be a spanning forest relative to X. If $u = \{x, y\} \in E(Y) \setminus E(X)$, then there exists a unique component of $Y \setminus u$ which does not contain a component of X. Furthermore, either x or y is a vertex of this component.

Lemma 33: Let X be a subgraph of G, let Y be an MSF relative to X, and let $Z \subseteq Y$ be a forest relative to X such that $Z \neq Y$. Let u be an edge of minimal altitude among all the edges of Y outgoing from Z. Then, the altitude of any edge of G outgoing from Z is greater than or equal to $F(u)$.

Proof: [of Lem. 16] $(i) \implies (ii)$: Let H be an $\mathscr{I}$-kernel of F and let $X = M(H)$. By Prop. 15, H is a $\mathscr{B}$-thinning of F. By Lem. 31, V is the vertex set of $M(H)$ and, again from Lem. 31, we deduce that H is an $\mathscr{M}$-kernel of F.
$(ii) \implies (iii)$: Let H be an $\mathscr{M}$-kernel of F and let $X = M(H)$. Trivially H is a $\mathscr{B}$-thinning of F. By Lem. 31, V is the vertex set of $M(H)$. Thus, by Lem. 30, H is a $\mathscr{B}$-kernel of F.
$(iii) \implies (iv)$: Prop. 10.
$(iv) \implies (i)$: Let X be an MSF relative to $M(F)$ and let us consider a sequence of graphs $X_0, \ldots, X_k$ such that:
- $X_0 = M(F)$;
- for any $i \in [1, k]$, $X_i = X_{i-1} \cup u_i$ where u_i is an edge of minimal altitude (for F) among all the edges of X outgoing from X_{i-1}; and
- V is the vertex set of X_k.
It may be seen that such a sequence always exists. Consider also the associated sequence of maps $F_0, \ldots F_k$ such that $F_0 = F$ and for any $i \in [1, k]$, F_i is the lowering of F_{i-1} at u_i.
We will proceed by induction to establish, for any $i \in [1, k]$, the following proposition:
$(\mathcal{P}_i)$: F_i is an $\mathscr{I}$-thinning of F such that $X_i = M(F_i)$.
Let $i \in [1, k]$ and suppose that $(\mathcal{P}_{i-1})$ holds true. By Prop. 15 and Lem. 27, $(\mathcal{P}_{i-1})$ implies that X_{i-1} is a forest relative to $M(F)$. Therefore, it follows from Lem. 33, that the altitude (for F) of any edge of G outgoing from X_{i-1} is greater than or equal to $F(u_i)$. By construction of F_{i-1}, we have $F(v) = F_{i-1}(v)$ for any edge v outgoing from X_{i-1}. Thus, u_i is an edge with minimal altitude (for F_{i-1}) among all the edges outgoing from X_{i-1}. Furthermore, thanks to $(\mathcal{P}_{i-1})$, $X_{i-1} = M(F_{i-1})$. Hence, u_i is an immersion edge for F_{i-1}, and it follows straightforwardly that F_i is an $\mathscr{I}$-thinning of F_{i-1}. Moreover, by $(\mathcal{P}_{i-1})$, F_i is an $\mathscr{I}$-thinning of F. Consequently to the definition of a lowering at an immersion edge, $M(F_i) = M(F_{i-1}) \cup u_i$. Hence, $M(F_i) = X_{i-1} \cup u_i = X_i$, which completes the proof of $(\mathcal{P}_i)$.
Since $(\mathcal{P}_0)$ is trivially verified, by induction, $\mathcal{P}_k$ is established. Therefore, by Prop. 15 and Lem. 27, $M(F_k) = X_k$ is a forest relative to $M(F)$. Since $V(X_k) = V$, since $X_k \subseteq X$ (by construction) and since X is a forest relative to $M(F)$, by the definition of a spanning forest, we have necessarily $X_k = X$. By Lem. 31, F_k is a $\mathscr{I}$-kernel of F. Hence, by $(\mathcal{P}_k)$, there exists an $\mathscr{I}$-thinning $H = F_k$ of F such that $X = X_k = M(H)$. ■

Proofs of Section III

Proof: [of Th. 20] Suppose that Y is an MSF relative to X. Suppose also that there exist A and B, two components of X such that $\Upsilon_F(A, B) \neq \Upsilon_F(A', B')$, where A' and B' are the two components of Y such that $A \subseteq A'$ and $B \subseteq B'$. Since $\Pi(A, B) \subseteq \Pi(A', B')$, $\Upsilon_F(A, B) > \Upsilon_F(A', B')$. Let $\pi = \langle x_k, \ldots, x_\ell \rangle$ be a path from A' to B' such that $\Upsilon_F(\pi) = \Upsilon_F(A', B')$ and such that x_k (resp. x_ℓ) is the only vertex of A' (resp. B') in π. Notice that $\{x_k, x_{k+1}\}$ and $\{x_{\ell-1}, x_\ell\}$ are not edges of Y. Let $\pi_A = \langle x_0, \ldots, x_k \rangle$ (resp. $\pi_B = \langle x_\ell, \ldots, x_m \rangle$) be a simple path in A' (resp. B'), such that x_0 (resp. x_m) is the only point of π_A (resp. π_B) which is a point of A (resp. B). Since $\pi' = \langle x_0, \ldots, x_m \rangle$ is a path from A to B, $Upsilon_F(\pi') \geq \Upsilon_F(A, B)$. Remark that $\Upsilon_F(\pi) < \Upsilon_F(A, B)$ since $\Upsilon_F(\pi) = \Upsilon_F(A', B')$ and $\Upsilon_F(A', B') < \Upsilon_F(A, B)$. Thus, we have either $\Upsilon_F(\pi_A) \geq \Upsilon_F(A, B)$ or $\Upsilon_F(\pi_B) \geq \Upsilon_F(A, B)$. Without loss of generality, assume that $\Upsilon_F(\pi_A) \geq \Upsilon_F(A, B)$. Let u be any edge of π_A such that $\Upsilon_F(u) = \Upsilon_F(\pi_A)$. Since $\Upsilon_F(\pi) < \Upsilon_F(\pi_A)$, $\Upsilon_F(u) > \Upsilon_F(\{x_k, x_{k+1}\})$. Since π_A is a simple path in A', since x_0 is the only point of π_A which is in A, and since $\{x_k, x_{k+1}\}$ is not in Y, it may be seen that $(Y \setminus u) \cup \{x_k, x_{k+1}\}$ is a spanning forest relative to X. Since $\Upsilon_F(u) > \Upsilon_F(\{x_k, x_{k+1}\})$, $(Y \setminus u) \cup \{x_k, x_{k+1}\}$ has a cost strictly less than Y. Thus, Y is not an MSF relative to X, a contradiction. ■

Proofs of Section IV

Proof: [of Th. 21] Suppose that Y is an MSF relative to X which is not an Υ-shortest-path spanning forest relative to X. There exists $x_0 \in V(Y)$ such that for any path π in Y from x_0 to X, we have $\Upsilon_F(\pi) > \Upsilon_F(x_0, X)$. Let $\pi = \langle x_0, \ldots, x_\ell \rangle$ be any such path and suppose, without loss of generality, that π is a simple path. Let $i \in [0, \ell-1]$ be such that $F(\{x_i, x_{i+1}\}) = \Upsilon_F(\pi)$ and let $u = \{x_i, x_{i+1}\}$. We denote by C the connected component of $Y \setminus u$ such that $x_0 \in V(C)$. Since π is a simple path, from Lem. 32, we deduce that C is the unique connected component of $Y \setminus u$ which does not contain a connected component of X. Let $\pi' = \{y_0 = x_0, \ldots, y_m\}$ be a path in G from x_0 to X such that $\Upsilon_F(\pi') = \Upsilon_F(x_0, X)$. Let $j \in [0, m-1]$ be such that $y_j \in V(C)$ whereas $y_{j+1} \notin V(C)$. Let $v = \{y_j, y_{j+1}\}$. Thus, $(Y \backslash u) \cup v$ is a spanning forest relative to X. Necessarily, $F(v) \leq \Upsilon_F(\pi')$. Hence, since $\Upsilon_F(\pi') = \Upsilon_F(x_0, X)$ and $\Upsilon_F(\pi) > \Upsilon_F(x_0, X)$, $F(v) < \Upsilon_F(\pi)$ and $F(v) < F(\{x_i, x_{i+1}\})$. Thus, from the two previous observations, we deduce that Y is not an MSF relative to X, a contradiction. ■

Since G is a finite graph, for any $x \in V$ there exists a path π with steepest descent for F from x to $M(F)$. Then, it may be seen that $\Upsilon_F(\pi) = F(x) = \Upsilon_F(x, M(F))$.

Proof: [of Prop. 22] (i) Suppose that X is a spanning forest relative to $M(F)$ which is not an MSF relative to $M(F)$. From Lem. 26, there exists a vertex $x \in V$ such that none of the paths in X from x to $M(F)$ is with steepest descent for F. Let P be the set of all points that can be reached from x by a path in X with steepest descent for F. Let y_0 be the vertex of P with minimal altitude. By hypothesis, y_0 is not a vertex of $M(F)$. Let $\pi = \langle y_0, \ldots, y_\ell \rangle$ be the unique $\mathcal{M}$-path, in X, from y_0 to $M(F)$. Let $i \in [0, \ell - 1]$ be the lowest index such that $y_i \in P$ and $y_{i+1} \in V \setminus P$. If $F(\{y_i, y_{i+1}\}) = F^{\ominus}(y_i)$, then there exists $j \in [0, i-1]$ such that $F^{\ominus}(y_j) < F(\{y_j, y_{j+1}\})$ (otherwise y_{i+1} would belong to P) and thus, $F(\{y_j, y_{j+1}\}) > F^{\ominus}(y_0)$ (since $F^{\ominus}(y_j) \geq F(y_0)$ by definition of y_0, i and j). If $F(\{y_i, y_{i+1}\}) > F^{\ominus}(y_i)$, then $F^{\ominus}(y_0) < F(\{y_i, y_{i+1}\})$ since $F^{\ominus}(y_0) \leq F^{\ominus}(y_i)$. In both cases, $\Upsilon_F(\pi) > F^{\ominus}(y_0)$. From the remark stated above this proof, we have $\Upsilon_F(\pi) > \Upsilon_F(y_0, M(F))$, hence, X is not an Υ-shortest-path forest relative to $M(F)$.
(ii) a direct consequence of Th. 21. ■

Before proving Th. 25, let us introduce a minimal set of definitions to handle the framework of vertex-weighted graphs in which topological watersheds are defined.

Let $P \subseteq V$. The *subgraph of G induced by P*, denoted by G_P, is the graph whose vertex set is P and whose edge set is made of all edges of G linking two points in P, *i.e.*, $G_P = (P, \{\{x, y\} \in E \mid x \in P, y \in P\})$. Let I be a map from V to $\mathbb{Z}$, and let $k \in \mathbb{Z}$. We denote by $\overline{I}[k]$ the subgraph of G induced by the set of all points $x \in V$ such that $I(x) < k$; $\overline{I}[k]$ is called a *(level k) lower-section* of I.

Definition 34: Let I be a map from V to $\mathbb{Z}$. Let x in V and $k = I(x)$. If x is adjacent to exactly one component of $\overline{I}[k]$, we say that x is W-destructible for I.
Let J be a map from V to $\mathbb{Z}$. We say that J is a *W-thinning of I (in G)* if $J = I$ or if J may be derived from I by iteratively lowering the values of W-destructible points by one.
We say that J is a *topological watershed of I* if J is a W-thinning of I and if there is no W-destructible point for J.

Let us consider the map I depicted in Fig. 8d (main text). The points at altitude 2 are both W-destructible whereas the point at altitude 5 is not. The maps J and K depicted, respectively, in (e) and (f) are W-thinnings of F. The reader can verify that there exists a sequence of maps to obtain J (resp. K) from I by iteratively lowering by one the values of W-destructible points. Notice that J is a topological watershed of I, since there is no W-destructible point for J and that K is not a topological watershed of I since the points at altitude 10, 6 and 4 are W-destructible.

Important remark. To be consistent with the definition of a topological watershed, in the following, we suppose that F maps E to $\mathbb{Z}$. Nevertheless, the topological watershed can be extended to maps from E to $\mathbb{R}$ and Th. 25 can be generalized to any map $F \in \mathcal{F}$.

Proof: [of Th. 25] Let $u = \{x, y\} \in E$ be a border edge for F such that $F(u) = F^{\ominus}(x) = k$. We will prove that the lowering of F at u is a W-thinning of F, hence, by induction, this will establish Th. 25. From the definition of a border edge, $F^{\ominus}(y) < k$. Thus, there exists a set of edges $S \subseteq E$, such that $S = \{v_i = \{y, y_i\} \in E \mid y_i \neq x \text{ and } F(v_i) < k\}$. Since any element in S contains y, all the edges in S are in the same component of $\overline{F}[k]$. Since $F^{\ominus}(x) = k$, none of the edges $v_j = \{x, z_j\} \in E$ with $z_j \neq y$, is in $\overline{F}[k]$. Thus, u is adjacent to exactly one component of $\overline{F}[k]$. Hence, u is W-destructible for F and the map obtained by lowering the value of u by one is a W-thinning of F. By iterating the same arguments, it may be seen that u can be lowered down to $F^{\ominus}(y)$. In other words, the lowering of F at u is a W-thinning of F. ■