Incremental Algorithm for Hierarchical Minimum Spanning Forests and Saliency of Watershed Cuts⋆

Jean Cousty and Laurent Najman

Université Paris-Est, Laboratoire d'Informatique Gaspard-Monge, A3SI, ESIEE
{j.cousty,l.najman}@esiee.fr

Abstract. We study hierarchical segmentations that are optimal in the sense of minimal spanning forests of the original image. We introduce a region-merging operation called *uprooting*, and we prove that optimal hierarchical segmentations are equivalent to the ones given by uprooting a watershed-cut based segmentation. Based on those theoretical results, we propose an efficient algorithm to compute such hierarchies, as well as the first saliency map algorithm *compatible* with the morphological filtering framework.

1   Introduction

We study some optimality properties of hierarchical segmentations ([19–21, 5, 23, 16, 8, 24, 2]) in the framework of edge-weighted graphs, where the cost of an edge is given by a dissimilarity between two points of an image. Since the pioneering work of [17, 15] stating an equivalence between hierarchies and minimum spanning trees (MST), numerous hierarchical schemes rely on the construction of such a tree. Its first appearance for classification in pattern recognition dates from the seminal work of Zahn [31]. Its use for image segmentation was introduced by Morris *et al.* [22] in 1986 and popularized in 2004 by Felzenswalb and Huttenlocher [13]. Meyer was the first to explicitly use it in a morphological context [19]. In mathematical morphology, hierarchies of watershed regions have been proposed notably in [5, 23, 20] and reviewed in [21].

   In this paper, we formalize, in the framework of edge-weighted graph, a fundamental hierarchical scheme proposed in morphology. This formalism allows us to prove strong properties linking hierarchical segmentations and combinatorial optimality with respect to the original image (Th. 4 and 9). We derive from those properties the first saliency algorithm proved to be correct, establishing in particular its compatibility with morphological filtering (Th. 11). In contrast, all previously proposed algorithms (*e.g.* [5, 23, 18]) are heuristic by nature, and to date, no property on their result has been proved. With those previous algorithms, counter-intuitive results are often obtained [26, 11].

   After reminding basic notions in Section 2, hierarchies of minimum spanning forests are presented in Section 3. Then, we formalize, in Section 4, a fundamental

---

operation called uprooting that merges a marked region with one of its neighbors linked by the cheapest cost. When applied sequentially on the weighted graph of neighboring regions, the uprootings build a MST of this neighboring graph. Intuitively, one can see that, if one starts from a minimum spanning forest (MSF) rooted in the minima of the image (or, equivalently, from a watershed cut [9]), then one builds a hierarchy of MSFs of the original image itself, the last uprooting step yielding an MST of this original image. More surprisingly, Th. 4 establishes that the two processes are equivalent. Hence, any MST of the original image can be built from an uprooting sequence on a watershed cut. Thus, watershed cuts are the only family of watersheds that allows us to build hierarchical segmentations that are optimal with respect to the original image, in the sense that they preserve the MST of the original image. Based on those results, we give a detailed description of a fast uprooting algorithm that allows MSF hierarchies to be computed. The time-complexity of this algorithm is analyzed. Then, in Section 5, a quasi linear-time algorithm is proposed to compute saliency maps from MSF hierarchies. The correctness of this algorithm is established by Theorem 9. Finally, in Section 6, the compatibility between the MSF hierarchies and a morphological filtering is established by an equivalence result (Theorem 11).

2   Basic Notions

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

Let X be a graph, the vertex set and edge set of X are denoted by $V(X)$ and $E(X)$ respectively. The graph X is *nontrivial* whenever $E(X) \neq \emptyset$, and it is nonempty whenever $V(X) \neq \emptyset$.

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 $Y \sqsubseteq X$. Let $\mathcal{S}$ be a set of graphs. The *supremum (resp. infimum) of $\mathcal{S}$*, denoted by $\sqcup \mathcal{S}$ (resp. $\sqcap \mathcal{S}$), is the graph whose vertex set and edge set are the unions (resp. intersections) of the vertex sets and of the edge sets of the graphs in $\mathcal{S}$: $\sqcup \mathcal{S} = (\cup \{V(X) \mid X \in \mathcal{S}\}, \cup \{E(X) \mid X \in \mathcal{S}\})$ (resp. $\sqcap \mathcal{S} = (\cap \{V(X) \mid X \in \mathcal{S}\}, \cap \{E(X) \mid X \in \mathcal{S}\})$). We also write $X \sqcup Y$ and $X \sqcap Y$ for respectively $\sqcup \{X, Y\}$ and $\sqcap \{X, Y\}$.

Let X be a graph. The graph X *is connected* if it cannot be partitioned into two nonempty graphs, *i.e.*, for any two nonempty graphs Y and Z, if $Y \sqcup Z = X$, then the graph $Y \sqcap Z$ is necessarily nonempty. Let Y be a graph. We say that Y is a *connected component of X*, or simply a *component of X*, if Y is a subgraph of X that is connected and that is maximal for this property, *i.e.*, for any connected graph Z, if $Y \sqsubseteq Z \sqsubseteq X$, then we have necessarily $Z = Y$. The set of all connected components of X is denoted by $\mathcal{CC}(X)$.

Important notation. *Throughout this paper $G = (V, E)$ denotes a nontrivial connected graph.*

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

Let $S \subseteq E$, we denote by $\delta^{\bullet}(S)$ the set of all vertices in V that belong to an edge in S. Remark that $(\delta^{\bullet}(S), S)$ is a graph (see [12] for morphological properties of $\delta^{\bullet}$). This graph $(\delta^{\bullet}(S), S)$ is called the *graph induced by S*.

In the following, the sets of integers and real numbers are denoted by $\mathbb{Z}$ and $\mathbb{R}$ respectively. Let $i, j \in \mathbb{Z}$, we denote by $[i, j]$ the set $\{k \in \mathbb{Z} \mid i \leq k \leq j\}$.

We denote by $\mathcal{F}$ the set of all maps from E into $\mathbb{R}$, 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, the number $F(u)$ is the *altitude or weight* of u. Let $k \in \mathbb{R}$, we denote by $F[k]$ the set of edges of G whose weight (for F) is less than or equal to k: $F[k] = \{u \in E \mid F(u) \leq k\}$. The set of edges $F[k]$ is called the *(lower) cross section of F at level k*.

Let $F \in \mathcal{F}$, and let $X \sqsubseteq G$. We say that X is a *connected component of F*, or simply a *component of F*, if there exists an element $k \in \mathbb{R}$ such that X is a component of the graph induced by $F[k]$. We denote by $\mathcal{CC}(F)$ the set of all components of F. The set $\mathcal{CC}(F)$ is closely related to the component tree [28, 25] of F, a tree widely used for filtering.

Let $F \in \mathcal{F}$, and let $X \sqsubseteq G$. The graph X is a *minimum of F* if X is a component of F that does not strictly contains any component of F, i.e., for any $Y \in \mathcal{CC}(F)$, if $Y \sqsubseteq X$, then we have necessarily $Y = X$. We denote by $\mathcal{M}_F$ the set made of the minima of F, and by M_F its supremum: $M_F = \sqcup \mathcal{M}_F$. This notion of minima, as well as further presented notions, is illustrated in Fig. 1

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

3   Minimum Spanning Forests Hierarchies

This section first presents the minimum spanning forests rooted in subgraphs of G. This notion of a forest, which is useful for (seeded) image segmentation (see *e.g.* [9, 10, 6]), is known to be equivalent to the one of minimum spanning tree studied in combinatorial optimization. Then, hierarchies of minimum spanning forests are introduced. Each such hierarchy induces a hierarchy of partitions on V, which is thus optimal in the sense of rooted minimum spanning forests.

Let $X \sqsubseteq G$. *The weight of X (for F)*, denoted by $F(X)$, is the sum of the weights of the edges in $E(X)$: $F(X) = \sum_{u \in E(X)} F(u)$. Let $V' \subseteq V$, we say that X is *spanning for V'* if $V(X) = V'$.

Definition 1 (rooted MSF). *Let X and Y be two nonempty subgraphs of G.*

We say that Y is rooted in X *if $V(X) \subseteq V(Y)$ and if the vertex set of any component of Y contains the vertex set of exactly one component of X.*

We say that Y is a minimum spanning forest (MSF) rooted in X *(with respect to F) if:*

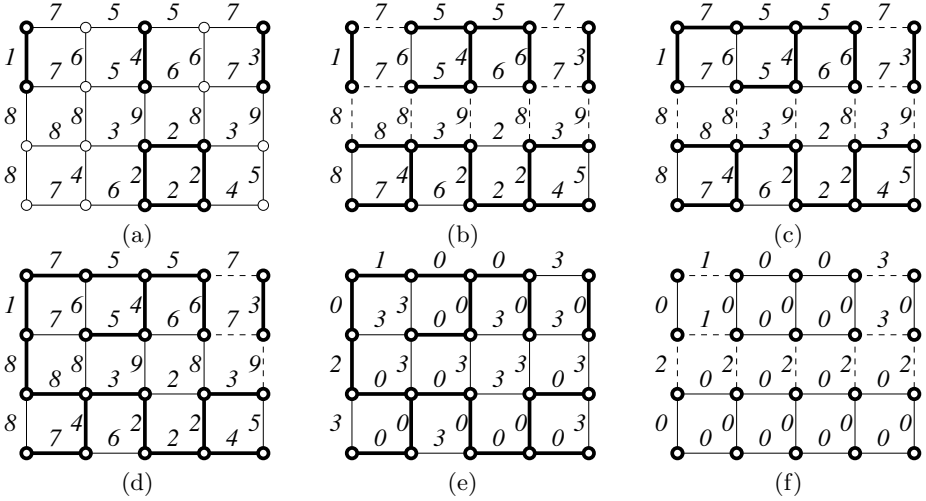1. *Y is spanning for V;*
2. *Y is rooted in X; and*

Fig. 1. (a): A graph G and a map F whose minima are depicted in bold. (b): A graph $X_0 \in MSF(M_F)$ represented in bold and its induced cut $C_0 = C(X_0)$ represented by dashed edges. (c,d): two bold graphs called respectively X_1 and X_2 such that $\mathcal{T} = \langle X_0, X_1, X_2 \rangle$ is both an MSF hierarchy for and an uprooting by $\langle M_1, M_2 \rangle$ (where M_i is the minimum of F whose altitude is i); their induced cuts $C_1 = C(X_1)$ and $C_2 = C(X_2)$ are represented by dashed edges. (e): The indicator of the uprooting $\mathcal{U}$. (f): The saliency map of the MSF cut hierarchy $\langle C_0, C_1, C_2 \rangle$.

3. *the weight of Y is less than or equal to the weight of any graph Z satisfying (1) and (2) (*i.e., Z is both spanning for V and rooted in X).

The set of all minimum spanning forests rooted in X is denoted by $MSF(X)$.

The above definition of rooted MSFs, which is illustrated in Fig. 1b, allows the usual notions of graph theory based on trees and forests to be recovered. In particular, if x is a vertex of V, it can be seen that any element in $MSF((\{x\}, \emptyset))$ is a *minimum spanning tree of* (G, F), and that, conversely, any minimum spanning tree of (G, F) belongs to $MSF((\{x\}, \emptyset))$. In the following, by convention, this remarkable set, which is made of all minimum spanning trees of (G, F), will also be denoted by $MSF((\emptyset, \emptyset))$.

A possible definition for watershed, called watershed cuts, follows the drop of water principle. In [9], we have proved the equivalence between MSF rooted in the set of minima and watershed cuts. In practice, watersheds from markers are often computed, and subsets of minima of the original edge-weighted graph constitute robust markers [3]. The next definition, illustrated in Figs. 1b, c, and d, presents a notion of hierarchy of MSFs rooted in such subsets.

Definition 2 (MSF hierarchy). *Let $\mathcal{S} = \langle M_1, \ldots, M_\ell \rangle$ be a sequence of pairwise distinct minima of F and let $\mathcal{T} = \langle X_0, \ldots X_\ell \rangle$ be a sequence of subgraphs of G. We say that $\mathcal{T}$ is an MSF hierarchy for $\mathcal{S}$ if:*

1. *for any $i \in [0, \ell]$, the graph X_i is an MSF rooted in $\sqcup[\mathcal{M}_F \setminus \{M_j \mid j \in [1, i]\}]$; and*
2. *for any $i \in [1, \ell]$, we have $X_{i-1} \sqsubseteq X_i$.*

4   Uprootings and MSF Hierarchies

In this section, we formalize a simple transformation, called uprooting, that allows a forest X rooted in a graph M to be incrementally transformed into a forest Y rooted in a graph M' obtained by removing some components of M. Through an equivalence theorem, we establish an important link between the uprooting transform and the MSF hierarchies. This result allows efficient algorithms for computing MSF hierarchies to be considered.

Let X be a subgraph of G that is spanning for V, and let $x \in V$. We denote by $CC_x(X)$ the component of X whose vertex set contains x. Let $V' \subseteq V$, we set $CC_{V'}(X) = \sqcup\{CC_x(X) \mid x \in V'\}$.

Let $X \subseteq G$, and let $u = \{x, y\} \in E$. The edge u is *outgoing from* X if u is made of a vertex in $V(X)$ and of a vertex in $\overline{V(X)}$. In the following, by abuse of notation, we write $X \sqcup \{u\}$ for the supremum of X and the graph induced by $\{u\}$: $X \sqcup \{u\} = (V(X) \cup u, E(X) \cup \{u\})$.

Let X, Y, and M be three subgraphs of G such that X is spanning for V and such that $X \neq Y$. We say that Y is an *elementary uprooting of X by M* if there exists an edge u of minimum weight among the edges outgoing from $CC_{V(M)}(X)$ such that $Y = X \sqcup \{u\}$. We also say that Y is an *elementary uprooting of X by M* if $Y = X$ and if there is no edge outgoing from $CC_{V(M)}(X)$.

Definition 3. *Let $\mathcal{S} = \langle M_1, \ldots, M_\ell \rangle$ be a sequence of pairwise distinct minima of F. An* uprooting by $\mathcal{S}$ *is a sequence $\langle X_0, \ldots, X_\ell \rangle$ of graphs such that:*

1. *$X_0 \in MSF(M_F)$; and*
2. *X_i is an elementary uprooting of X_{i-1} by M_i, for any $i \in [1, \ell]$.*

The following theorem, one of the main results of this paper, states an equivalence property between MSF hierarchies and uprootings by sequences of minima of the original map that weight the edges of the graph.

Theorem 4. *Let $\mathcal{S} = \langle M_1, \ldots, M_\ell \rangle$ be a sequence of pairwise distinct minima of F, and let $\mathcal{T} = \langle X_0, \ldots X_\ell \rangle$ be a sequence of subgraphs of G. The sequence $\mathcal{T}$ is an MSF hierarchy for $\mathcal{S}$ if and only if the sequence $\mathcal{T}$ is an uprooting by $\mathcal{S}$.*

Any MSF hierarchy can be represented by a unique edge-weighed graph such that a given threshold of this graph provides the associated level of the hierarchy. More formally, let $\mathcal{S} = \langle M_1, \ldots, M_\ell \rangle$ be a sequence of pairwise distinct minima of F, and let $\mathcal{T} = \langle X_0, \ldots, X_\ell \rangle$ be an uprooting by $\mathcal{S}$. The *indicator of $\mathcal{T}$*, denoted by $I_\mathcal{T}$, is the map from E into $[0, \ell + 1]$ defined by:

- $I_\mathcal{T}(u) = \min\{i \in [0, \ell] \mid u \in E(X_i)\}$, for any $u \in E(X_\ell)$; and
- $I_\mathcal{T}(u) = \ell + 1$, for any $u \in E \setminus E(X_\ell)$.

The notion of indicator of an uprooting is illustrated in Fig. 1e.

Any uprooting has a unique indicator, and any two distinct uprootings have distinct indicators. Hence, the indicator of an uprooting is sufficient to recover all the elements of the uprooting: if $\mathcal{T} = \langle X_0, \ldots, X_\ell \rangle$ is an uprooting by a sequence $\mathcal{S} = \langle M_1, \ldots, M_\ell \rangle$ of minima of F, then X_i is the graph induced by the cross-section of $I_\mathcal{T}$ at level i, (*i.e.*, $X_i = (\delta^\bullet(I_\mathcal{T}[i]), I_\mathcal{T}[i])$).

We are now ready to describe the uprooting algorithm, which by Theorem 4 allows the computation of MSF hierarchies. It inputs a sequence $\mathcal{S} = \langle M_1, \ldots, M_\ell \rangle$ of minima of F and outputs the indicator $I_\mathcal{T}$ of an uprooting $\mathcal{T} = \langle X_0, \ldots, X_\ell \rangle$ by $\mathcal{S}$. From a high level point of view, the algorithm can be sketched, once a minimum spanning forest X relative to M has been computed, by iterations of the following three region merging steps:

  i Find the component C of X that contains M_i the next element in $\mathcal{S}$.
  ii Find a component C' of X linked to C by an edge v of minimum weight
  iii Merge the two components C and C'

Algorithm. Uprooting

Data: (V, E, F): an edge-weighted graph;
$X \leftarrow$ any element in $MSF(M_F)$;
A sequence $\mathcal{S} = \langle M_0, \ldots, M_\ell \rangle$ of pairwise distinct minima of F;
Result: I, the indicator of an uprooting by $\mathcal{S}$;

```
1  i ← 0; ;                                            /* Initialisation */
2  foreach x ∈ E do
3  │  MakeSet(x); L[x] ← MakeList();

4  foreach u = {x, y} ∈ E do
5  │  if u ∈ E(X) then  I(u) ← 0;else I(u) ← ℓ + 1;

6  foreach u = {x, y} ∈ E(X) do
7  │  x' ← Find(x); y' ← Find(y);
8  │  if x' ≠ y' then  Link(x',y');

9  foreach u = {x, y} ∈ E \ E(X) do
10 │  x' ← Find(x); y' ← Find(y);
11 │  if x' ≠ y' then Insert_x'(y', u, F(u)); Insert_y'(x', u, F(u));

12 while i < ℓ do                                  /* Incremental uprooting */
13 │  i ← i + 1;
14 │  Set x to any element of V(M_i); x' ← Find(x);
15 │  repeat
16 │  │  (y, v, w) ← DeleteMin_x'();
17 │  │  y' ← Find(y);
18 │  until x' ≠ y' ;
19 │  I(v) ← i;
20 │  z ← Link(x',y'); L[z] ← Meld(x',y');
```

In order to ease the reading of the algorithm, let us first present the two main data structures that are used.

The structure $\mathcal{C}$ is a collection of disjoint subsets of V. Each set C in $\mathcal{C}$ is represented by a unique element x of C, called the canonical element of C. In the following, x and y denote two distinct elements of V. The collection $\mathcal{C}$ is managed by three operations:

- **MakeSet(x):** add the set $\{x\}$ to the collection $\mathcal{C}$, provided that the element x does not already belong to a set in $\mathcal{C}$.
- **Find(x):** returns the canonical element of the set in $\mathcal{C}$ that contains x.
- **Link(x, y):** let C_x and C_y denote two sets in $\mathcal{C}$ whose canonical elements are x and y respectively. Both sets are removed from $\mathcal{C}$, their union $C = C_x \cup C_y$ is added to $\mathcal{C}$ and a canonical element for C is selected and returned.

The structure $\mathcal{L}$ is a collection of $n = |V|$ lists: $\mathcal{L}[x_1], \ldots, \mathcal{L}[x_n]$, with $V = \{x_1, \ldots, x_n\}$. Each element of these lists is a triplet (x, u, w) such that x is a vertex of G, u is an edge of G, and w is an element in $\mathbb{R}$, called the weight of the triplet. In the following, x and y denote two distinct elements of V, u denotes an edge in E, and w an element in $\mathbb{R}$. The collection $\mathcal{L}$ is managed by four operations:

- **MakeList():** returns an empty list.
- **Insert$_x$(y, u, w):** adds the triplet (y, u, w) to the list $\mathcal{L}[x]$;
- **DeleteMin$_x$()** returns and removes from $\mathcal{L}[x]$ a triplet of minimum weight.
- **Meld(x, y):** returns the list of the triplets of $\mathcal{L}[x]$ and of $\mathcal{L}[y]$ and suppresses the lists $\mathcal{L}[x]$ and $\mathcal{L}[y]$.

When the algorithm terminates the map I is the indicator of an uprooting by the input sequence $\mathcal{S}$. Moreover, using Tarjan's union find [29] and Fredman and Tarjan's Fibonnacci heap [14] algorithms to manage the collections $\mathcal{C}$ and $\mathcal{L}$, the overall complexity of the algorithm is $O(|V| \times \alpha(|V|, |V|) + |E| \times \alpha(|E|, |V|) + |E| \log |E|)$, where α is a function which grows very slowly: for all practical purposes $\alpha(m, n)$ is never greater than four. In other words, the complexity of the algorithm is quasi $O(|V| + |E| \log |E|)$ in the sense of Tarjan's union find.

5   MSF Cut Hierarchy, Saliency and Connection Value

Until now, we have dealt with regions (components of forests). Let us now study their "dual", called cuts, that represent borders between regions.

Let X be a subgraph of G that is spanning for V. The *(graph) cut induced by X*, denoted by $C(X)$, is the set of all edges of G made of vertices of two distinct components of X, *i.e.* $C(X) = \{\{x, y\} \in V \mid CC_x(X) \neq CC_y(X)\}$.

Definition 5 (MSF cut hierarchy). *Let X be a subgraph of G, and let $C \subseteq E$. The set C is called an* MSF cut *for X (with respect to F) if there exists an MSF Y rooted in X, such that C is the cut induced by Y.*

Let $\mathcal{S} = \langle M_1, \ldots, M_\ell \rangle$ be a sequence of pairwise distinct minima of F and let $\mathcal{T} = \langle C_0, \ldots C_\ell \rangle$ be a sequence of subsets of E. We say that $\mathcal{T}$ is an MSF cut *hierarchy for $\mathcal{S}$ if:*

1. for any $i \in [0, \ell]$, the set C_i is an MSF cut for $\sqcup[\mathcal{M}_F \setminus \{M_j \mid j \in [1, i]\}]$; and
2. for any $i \in [1, \ell]$, we have $C_i \subseteq C_{i-1}$.

The following result asserts that there is indeed an equivalence relation between MSF hierarchies and MSF cut hierarchies.

Property 6. *Let* $\mathcal{S} = \langle M_1, \ldots, M_\ell \rangle$ *be a sequence of pairwise distinct minima of* F *and let* $\mathcal{T} = \langle C_0, \ldots C_\ell \rangle$ *be a sequence of subsets of* E. *The sequence* $\mathcal{T}$ *is an MSF cut hierarchy if and only if there exists an MSF hierarchy* $\langle X_0, \ldots, X_\ell \rangle$ *such that* $C(X_i) = C_i$, *for any* $i \in [0, \ell]$.

The hierarchies presented above are hierarchical segmentations as defined in [24]. Therefore, the MSF cuts belonging to a hierarchy can be "stacked" to form a map that equivalently represents this hierarchy. Intuitively, such a map, called a saliency map, weights the cuts with their "level of disappearance" in the hierarchy. Hence, they are convenient for visualizing hierarchies.

Definition 7 (saliency map). *Let* $\mathcal{S} = \langle M_1, \ldots, M_\ell \rangle$ *be a sequence of minima of* F *and let* $\mathcal{T} = \langle C_0, \ldots C_\ell \rangle$ *be an MSF cuts hierarchy for* $\mathcal{S}$. *The* saliency map *of* $\mathcal{T}$, *denoted by* $S_\mathcal{T}$, *is the map from* E *into* $[0, \ell + 1]$ *defined by:*

1. $S_\mathcal{T}(u) = \min\{i \in [0, \ell] \mid u \in \overline{C_i}\}$, *for any* $u \in E \setminus C_\ell$; *and*
2. $S_\mathcal{T}(u) = \ell + 1$, *for any* $u \in C_\ell$.

An illustration of the notion of saliency map is given in Fig. 1f.

Let H be any map and let $k \in \mathbb{R}$, the set $\overline{H[k]}$ of the edges of G whose weight (for H) is greater than k is called the *upper cross section* of H *at level* k.

Let $\mathcal{T} = \langle C_0, \ldots, C_\ell \rangle$ be an MSF cut hierarchy for a sequence $\mathcal{S}$ of minima of F, and let H be a map from E into $[0, \ell + 1]$. It can be seen that $H = S_\mathcal{T}$ if and only if, for any $i \in [0, \ell]$, the set C_i is the upper cross-section of H at level i. Thus, the saliency map $S_\mathcal{T}$ is an equivalent representation of the hierarchy $\mathcal{T}$.

The altitudes of the passes between minima of the image play a fundamental role in morphological filtering. In our framework, pass altitudes are called connection values. In the following, we show their importance for computing a saliency map from the indicator of an uprooting.

Let $H \in \mathcal{F}$ and let $X \subseteq G$. The altitude of X for H, denoted by $\check{H}(X)$, is the maximum value of an edge of X.

Definition 8 (connection value). *Let* $H \in \mathcal{F}$, *and let* x *and* y *be two points of* G. *The* connection value *between* x *and* y *for* H (*in* G) *is the value* $\Upsilon_H(x, y) = \min\{\check{H}(X) \mid X \in \mathcal{CC}(F) \text{ and } \{x, y\} \subseteq V(X)\}$.

The connection value, for any map H, between any two points can be computed in constant time (see [7]) from the (min) component tree $(\mathcal{CC}(H), \sqsubseteq)$ of H, provided a linear time preprocessing (see [4]). In other words, once the component tree is built (see [25] for a quasi linear time algorithm) and preprocessed (in linear time), computing the connection value between any arbitrary pair of points can be done in constant time.

Hierarchical segmentations are in bijection with ultrametric watersheds [24]. The following theorem provides an analog result for MSF cut hierarchies.

Theorem 9. *Let $\mathcal{S} = \langle M_1, \ldots, M_\ell \rangle$ be a sequence of pairwise distinct minima of F, and let H be a map from E into $[0, \ell+1]$. The map H is the saliency map of an MSF cut hierarchy for $\mathcal{S}$ if and only if there exists an uprooting $\mathcal{T}$ by $\mathcal{S}$ such that, for any $u = \{x, y\}$, $F(u) = \Upsilon_{I_{\mathcal{T}}}(x, y)$.*

Hence, in order to know the value of an edge in a saliency map it is sufficient to know the connection value between the two points linked by this edge for the indicator of an uprooting by $\mathcal{S}$. Thus, knowing the indicator of an uprooting by $\mathcal{S}$ and its component tree, a saliency map can be computed in linear time, *i.e.*, one constant time operation per edge of G.

Figs. 2b,c, and d illustrates the use of this algorithm on the image of Fig. 2a. The underlying graph is the one induced by the 4-adjacency relation whose edges are weighted by a simple color gradient (maximum, over the RGB channels,
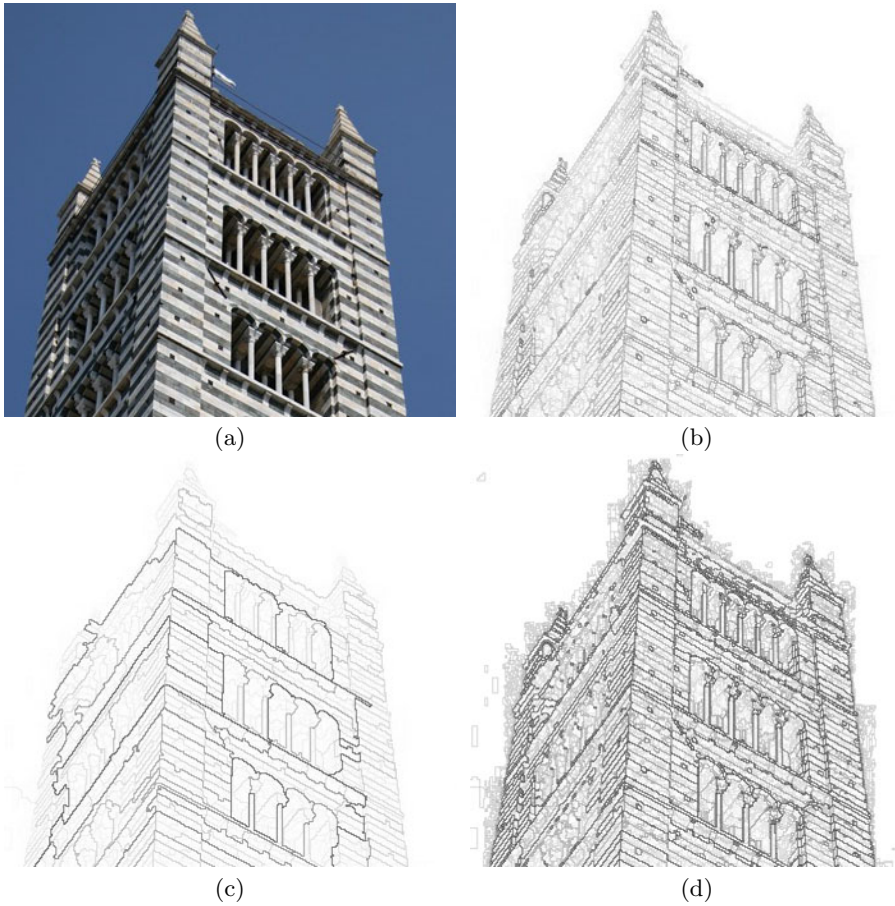

(a)

(b)

(c)

(d)

Fig. 2. Illustration of saliencies of watershed cuts (original picture (a) from *koakoo*: http://blog.photos-libres.fr/)

<table>
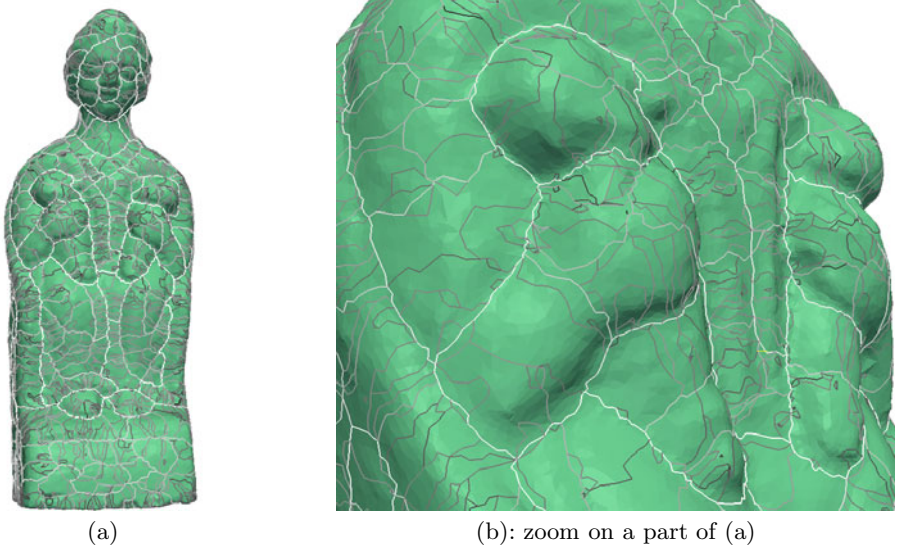<tr><td>(a)</td><td>(b): zoom on a part of (a)</td></tr>
</table>

Fig. 3. Illustration of saliencies of watershed cuts on a mesh provided by the French Museum Center for Research and Restoration (C2RMF, Le Louvre, Paris)

of the absolute differences of pixel values). The minima are ordered thanks to extinction values [30] related to dynamics, volume and color consistency, leading to sequences $\mathcal{S}_1$, $\mathcal{S}_2$, and $\mathcal{S}_3$ of minima. The saliency maps of MSF hierarchies for $\mathcal{S}_1$, $\mathcal{S}_2$, and $\mathcal{S}_3$ are rendered (up to an anamorphosis) in sub-figures b,c, and d, respectively. Figs. 3 illustrates the use of the algorithm to segment a surface (*i.e.*, a mesh) embedded in the 3D Euclidean space. The vertex set of the considered graph is the set of triangles of the mesh, and its edge set is composed by the pair of triangles that share a common side. The edges of this graph are weighted by map which behaves like the inverse of the mean curvature of the surface (see [27]). The minima are ordered thanks to volume extinction values and the saliency maps of the induced MSF hierarchies is rendered up to an anamorphosis.

6 Hierarchy by Geodesic Reconstructions

A desirable compatibility property in the context of morphological filtering is that any threshold of a saliency map is a watershed of the geodesic reconstruction of the original map by the minima corresponding to the threshold level. Theorem 11 below shows that the results of our algorithm do satisfy such a compatibility property. It has to be noted that, in other frameworks (*e.g.*, node-weighted graphs), such a property is in general not guaranteed [26].

Let $x \in V$. We denote by $\mathcal{CC}_x(F)$ the set of all components of F whose vertex sets contain x. Let $V' \subseteq V$, we set $\mathcal{CC}_{V'}(F) = \cup\{\mathcal{CC}_x(F) \mid x \in V'\}$.

Let $V' \subseteq V$. The *geodesic reconstruction of F by V'* is the map F' such that, for any edge u in E, the value $F'(u)$ is the minimum altitude of a component in $\mathcal{CC}_{V'}(F)$ that contains u: $F'(u) = \min\{\check{F}(X) \mid X \in \mathcal{CC}_{V'}(F)\}$.

Watershed from markers are classically defined through geodesic reconstruction (also called swamping or flooding). It is shown in [1] that reconstruction plus MSF cut can be replaced by an MSF cut relative to the initial map.

Property 10 (from Theorem 6.3 in [1]). *Let $V' \subseteq V$, let F' be the geodesic reconstruction of F by V', and let C be a subset of E. If C is an MSF cut for M_H with respect to F, then C is an MSF cut for M_H with respect to F'.*

The converse of Prop. 10 is, in general, not true. In fact, as long as a sequence of nested partitions (*i.e.*, a hierarchy) is involved, the MSF cuts of reconstructed maps and the MSF cuts of the original maps are equivalent.

Theorem 11 (Compatibility). *Let $\mathcal{S} = \langle M_1, \ldots, M_\ell \rangle$ be a sequence of pairwise distinct minima of F. Let G_i denote $\sqcup[\mathcal{M}_F \setminus \{M_j \mid j \in [1, i]\}]$, and let F_i be the geodesic reconstruction of F by $V(G_i)$, for any $i \in [0, \ell]$. Let $\mathcal{T} = \langle C_0, \ldots, C_\ell \rangle$ be a sequence of subsets of E such that $C_i \subseteq C_{i-1}$ for any $i \in [1, \ell]$. Then, the two following propositions are equivalent:*

1. *for any $i \in [0, \ell]$, the set C_i is an MSF cut for G_i with respect to F_i; and*
2. *for any $i \in [0, \ell]$, the set C_i is an MSF cut for G_i with respect to F.*

7   Conclusion and Perspective

In this paper, a classical morphological scheme for building hierarchical segmentation is formalized. This formalism leads us to establish strong properties linking hierarchical segmentations and combinatorial optimality in terms of minimum spanning forests of the original image. Recent work of Couprie *et al.* [6] link some schemes based on minimum spanning forests to global energy minimization. Hence, a promising perspective is the investigation of hierarchical schemes defined through energy minimization such as the one presented in [16].

The source code of the algorithms presented in this paper is available at: `http://www.esiee.fr/~info/sm`

References

1. Allène, C., Audibert, J.-Y., Couprie, M., Keriven, R.: Some links between extremum spanning forests, watersheds and min-cuts. IVC 28(10), 1460–1471 (2010)
2. Arbeláez, P.A., Cohen, L.D.: A metric approach to vector-valued image segmentation. IJCV 69(1), 119–126 (2006)
3. Audigier, R., Lotufo, R.: Seed-relative segmentation robustness of watershed and fuzzy connectedness approaches. In: IEEE SIBGRAPI 2007, pp. 61–70 (2007)
4. Bender, M.A., Farach-Colton, M.: The LCA problem revisited. In: Latin American Theoretical Informatics, pp. 88–94 (2000)
5. Beucher, S.: Watershed, hierarchical segmentation and waterfall algorithm. In: ISMM 1994, pp. 69–76 (1994)
6. Couprie, C., Grady, L., Najman, L., Talbot, H.: Power Watersheds: A Unifying Graph Based Optimization Framework. PAMI (to appear, 2011)

7. Couprie, M., Najman, L., Bertrand, G.: Quasi-linear algorithms for the topological watershed. JMIV 22(2-3), 231–249 (2005)
8. Cousty, J., Najman, L., Serra, J.: Raising in watershed lattices. In: 15th IEEE ICIP 2008, pp. 2196–2199 (2008)
9. Cousty, J., Bertrand, G., Najman, L., Couprie, M.: Watershed Cuts: Minimum Spanning Forests and the Drop of Water Principle. PAMI 31(8), 1362–1374 (2009)
10. Cousty, J., Bertrand, G., Najman, L., Couprie, M.: Watershed cuts: thinnings, shortest-path forests and topological watersheds. PAMI 32(5), 925–939 (2010)
11. Cousty, J., Najman, L., Bertrand, G., Couprie, M.: Weighted fusion graphs: merging properties and watersheds. DAM 156(15), 3011–3027 (2008)
12. Cousty, J., Najman, L., Serra, J.: Some morphological operators in graph spaces. In: Wilkinson, M.H.F., Roerdink, J.B.T.M. (eds.) ISMM 2009. LNCS, vol. 5720, pp. 149–160. Springer, Heidelberg (2009)
13. Felzenszwalb, P., Huttenlocher, D.: Efficient graph-based image segmentation. International Journal of Computer Vision 59, 167–181 (2004)
14. Fredman, M.L., Tarjan, R.E.: Fibonacci heaps and their uses in improved network optimization algorithms. J. ACM 34, 596–615 (1987)
15. Gower, J.C., Ross, G.J.S.: Minimum spanning tree and single linkage cluster analysis. Appl. Stats. 18, 54–64 (1969)
16. Guigues, L., Cocquerez, J.P., Men, H.L.: Scale-sets image analysis. IJCV 68(3), 289–317 (2006)
17. Jardine, N., Sibson, R.: Mathematical taxonomy. Wiley, Chichester (1971)
18. Marcotegui, B., Beucher, S.: Fast implementation of waterfall based on graphs. In: ISMM 2005, pp. 177–186 (2005)
19. Meyer, F.: Minimum spanning forests for morphological segmentation. In: ISMM 1994, pp. 77–84 (1994)
20. Meyer, F.: The dynamics of minima and contours. In: ISMM, pp. 329–336 (1996)
21. Meyer, F., Najman, L.: Segmentation, minimum spanning tree and hierarchies. In: Mathematical Morphology, ch. 9, pp. 229–261. ISTE-Wiley (2010)
22. Morris, O.J., Lee, M.d.J., Constantinides, A.G.: Graph theory for image analysis: an approach based on the shortest spanning tree. IEE Proc. on Communications, Radar and Signal 133(2), 146–152 (1986)
23. Najman, L., Schmitt, M.: Geodesic saliency of watershed contours and hierarchical segmentation. PAMI 18(12), 1163–1173 (1996)
24. Najman, L.: On the equivalence between hierarchical segmentations and ultrametric watersheds. JMIV 40(3), 231–247 (2011)
25. Najman, L., Couprie, M.: Building the component tree in quasi-linear time. IEEE TIP 15(11), 3531–3539 (2006)
26. Najman, L., Couprie, M., Bertrand, G.: Watersheds, mosaics and the emergence paradigm. DAM 147(2-3), 301–324 (2005)
27. Philipp-Foliguet, S., Jordan, M., Najman, L., Cousty, J.: Artwork 3D Model Database Indexing and Classification. Patt. Recogn. 44(3), 588–597 (2011)
28. Salembier, P., Oliveras, A., Garrido, L.: Anti-extensive connected operators for image and sequence processing. IEEE TIP 7(4), 555–570 (1998)
29. Tarjan, R.E.: Efficiency of a good but not linear set union algorithm. J. ACM 22, 215–225 (1975)
30. Vachier, C., Meyer, F.: Extinction value: a new measurement of persistence. In: IEEE Workshop on Nonlinear Signal and Image Processing, pp. 254–257 (1995)
31. Zahn, C.T.: Graph-theoretical methods for detecting and descibing gestalt clusters. IEEE Transactions on Computers C-20(1), 99–112 (1971)