Scale-Space Representation Based on Levelings Through Hierarchies of Level Sets

Wonder A.L. Alves[1,2], Alexandre Morimitsu[1], and Ronaldo F. Hashimoto[1(✉)]

[1] Department of Computer Science, Institute of Mathematics and Statistics, University of São Paulo, São Paulo, Brazil
{wonder,alem,ronaldo}@ime.usp.br
[2] Department of Informatics, Nove de Julho University, São Paulo, Brazil

Abstract. This paper presents new theoretical contributions on scale-space representations based on levelings through hierarchies of level sets, i.e., component trees and tree of shapes. Firstly, we prove that reconstructions of pruned trees (component trees and tree of shapes) are levelings. After that, we present a new and fast algorithm for computing the reconstruction based on marker images from component trees. Finally, we show how to build morphological scale-spaces based on levelings through the reconstructions of successive pruning operations (whether based on increasing attributes or marker images).

Keywords: Scale-space · Levelings · Component trees · Tree of shapes

1 Introduction

As we know, an operator in Mathematical Morphology (MM) can be seen as a mapping between complete lattices. In particular, mappings on the set of all gray level images $\mathcal{F}(\mathcal{D})$ defined on domain $\mathcal{D} \subset \mathbb{Z}^2$ and codomain $\mathbb{K} = \{0, 1, ..., K\}$ are of special interest in MM and they are called *image operators*. Furthermore, when ψ enlarges the partition of the space created by the flat zones, it is called *connected operator* [1]. They represent a wide class of operators in which F. Meyer [2, 3, 4] extensively studied their specializations. One of these specializations, known as *levelings*, are powerful simplifying filters that preserve order, do not create new structures (regional extrema and contours) and their values are enclosed by values of a neighborhood of pixels (see Def. 1).

Definition 1 (F. Meyer [3, 2]). *An operator $\psi : \mathcal{F}(\mathcal{D}) \to \mathcal{F}(\mathcal{D})$ is said to be leveling, if and only if, for any $f \in \mathcal{F}(\mathcal{D})$ the following relation is valid for all pairs of adjacent pixels, i.e., $\forall(p, q) \in \mathcal{A}$,*

$$[\psi(f)](p) > [\psi(f)](q) \Rightarrow f(p) \geq [\psi(f)](p) \text{ and } [\psi(f)](q) \geq f(q).$$

where $\mathcal{A}$ is a adjacency relation[1] on $\mathcal{D}$.

[1] An adjacency relation $\mathcal{A}$ on $\mathcal{D}$ is a binary relation on pixels of $\mathcal{D}$. Thus, $(p, q) \in \mathcal{A}$ if and only if p is an adjacent of q. 4 or 8-adjacency are common examples of adjacencies relation on $\mathcal{D}$.

J.A. Benediktsson et al. (Eds.): ISMM 2015, LNCS 9082, pp. 265–276, 2015.
DOI: 10.1007/978-3-319-18720-4_23

From the definition of a class of operators, it is possible to build a binary relation $\mathcal{R}$ on $\mathcal{F}(\mathcal{D})$ as follows: $(f, g) \in \mathcal{R}$ if and only if there exists ψ in this class such that $g = \psi(f)$. Thus, the definition of levelings can be seen as a binary relation $\mathcal{R}_{leveling}$ on $\mathcal{F}(\mathcal{D})$. So, we say that g is leveling of f if and only if $(f, g) \in \mathcal{R}_{leveling}$. In [2], F. Meyer, shows that $\mathcal{R}_{leveling}$ is reflexive and transitive and if we ignore the constant images then $\mathcal{R}_{leveling}$ is anti-symmetric, i.e., $\mathcal{R}_{leveling}$ is a order relation. With the help of this order relation, the levelings can be nested to create a scale-space decomposition of an image $f \in \mathcal{F}(\mathcal{D})$ in the form of a series of levelings $(g_0 = f, g_1, ..., g_n)$, where g_k is leveling of g_{k-1} and as a consequence of transitivity, g_k is also a leveling of each image g_j, for $j < k$ [4, 5]. Thus, a morphological scale-space is generated with the following features: simplification, causality and fidelity [4, 5]. For example, this scale-space can be created with the help from a traditional algorithm Λ to construct levelings that takes as arguments two images: an input image $f \in \mathcal{F}(\mathcal{D})$ and a marker image $h \in \mathcal{F}(\mathcal{D})$. It modifies h in such a way that it becomes a leveling of f. Thus, we will say that $g = \Lambda(f, h)$ is a leveling of f, obtained from the marker h [3, 4]. With this algorithm it is possible to construct a morphological scale-space based on levelings from any family of markers $(h_1, h_2, ..., h_n)$ with the following chaining: $g_1 = \Lambda(f, h_1), g_2 = \Lambda(g_1, h_2), ..., g_n = \Lambda(g_{n-1}, h_n)$ [4].

In this work, we follow a different approach for construction of scale-spaces. Our approach consists of representing an image through a tree based on hierarchies of level sets (i.e., component tree and tree of shapes) and from this tree is proved that reconstruction of pruned trees are levelings. Despite these facts are knowns and/or mentioned by several authors [6, 7, 8, 9, 10, 11] this is the first study that presents a formal proof on the perspective of trees which reconstruction of pruned trees are levelings. In addition, we present a new fast algorithm for computing the reconstruction by dilation (or erosion) which is faster than the algorithm by Luc Vincent [12]. Finally, we show how to construct a morphological scale-space based on levelings through the reconstructions of successive pruning operations (whether based on increasing attributes or marker images).

The remainder of this paper is structured as follows. Section 2 briefly recalls some definitions and properties of image representation by tree structures. In Section 3, we provide the first original result of this work where establishes theoretical links between reconstruction pruned trees with levelings. In Section 4, we associate reconstructions of pruned trees with several morphological operators based on marker images. In Section 5, we present constructions of morphological scale-space either based on increasing attributes or marker images. In Section 6, we show an application of scale-space based on levelings to construct residual operators. Finally, Section 7 concludes this work.

2 Theoretical Background

For any $\lambda \in \mathbb{K} = \{0, 1, ..., K\}$, we define $\mathcal{X}_{\downarrow}^{\lambda}(f) = \{p \in \mathcal{D} : f(p) < \lambda\}$ and $\mathcal{X}_{\lambda}^{\uparrow}(f) = \{p \in \mathcal{D} : f(p) \geq \lambda\}$ as the *lower* and *upper level sets* at value λ from an image $f \in \mathcal{F}(\mathcal{D})$, respectively. These level sets are nested, i.e., $\mathcal{X}_{\downarrow}^{1}(f) \subseteq$

$\mathcal{X}_{\downarrow}^{2}(f) \subseteq ... \subseteq \mathcal{X}_{\downarrow}^{K}(f)$ and $\mathcal{X}_{K}^{\uparrow}(f) \subseteq \mathcal{X}_{K-1}^{\uparrow}(f) \subseteq ... \subseteq \mathcal{X}_{0}^{\uparrow}(f)$. Thus, the image f can be reconstructed using either the family of lower or upper sets, i.e., $\forall x \in \mathcal{D}$, $f(x) = \inf\{\lambda - 1 : x \in \mathcal{X}_{\downarrow}^{\lambda}(f)\} = \sup\{\lambda : x \in \mathcal{X}_{\lambda}^{\uparrow}(f)\}$. From these sets, we define two other sets $\mathcal{L}(f)$ and $\mathcal{U}(f)$ composed by the connected components (CCs) of the lower and upper level sets of f, i.e., $\mathcal{L}(f) = \{C \in \mathcal{CC}_4(\mathcal{X}_{\downarrow}^{\lambda}(f)) : \lambda \in \mathbb{K}\}$ and $\mathcal{U}(f) = \{C \in \mathcal{CC}_8(\mathcal{X}_{\lambda}^{\uparrow}(f)) : \lambda \in \mathbb{K}\}$, where $\mathcal{CC}_4(X)$ and $\mathcal{CC}_8(X)$ are sets of 4 and 8 connected CCs of X, respectively. Then, the ordered pairs consisting of the CCs of the lower and upper level sets and the usual inclusion set relation, i.e., $(\mathcal{L}(f), \subseteq)$ and $(\mathcal{U}(f), \subseteq)$, induce two dual trees [13]. They can be represented by a non-redundant data structures known as min-tree and max-tree. Combining this pair of dual trees, min-tree and max-tree, into a single tree, we have the tree of shapes [13]. Then, let $\mathcal{P}(\mathcal{D})$ denote the *powerset* of $\mathcal{D}$ and let $sat : \mathcal{P}(\mathcal{D}) \to \mathcal{P}(\mathcal{D})$ be the operator of saturation [13] (or filling holes), i.e.,

Definition 2. *We call* holes *of $A \in \mathcal{L}(f) \cup \mathcal{U}(f)$ the CCs of $\mathcal{D} \setminus A$. Thus, we call* internal holes *of A, denoted by $Int(A)$, the CCs of $\mathcal{D} \setminus A$ that are subsets of $sat(A)$. Likewise, we call* exterior hole *of A, denoted by $Ext(A)$, the set $\mathcal{D} \setminus sat(A)$. Thus, $sat(A) = \cup\{H \in Int(A)\} \cup A$, where the unions are disjoint.*

Note that, the complement of a CC A is in $Ext(A) \cup Int(A)$ and if $H \in Int(A)$ then $sat(H) \subseteq sat(A)$. Moreover, if $A \in \mathcal{L}(f)$ with internal holes, then elements of $Int(A)$ are CCs of $\mathcal{U}(f)$ (or vice versa, $A \in \mathcal{U}(f) \Rightarrow Int(A) \subseteq \mathcal{L}(f)$). Now, let $\mathcal{SAT}_{\mathcal{L}}(f) = \{sat(C) : C \in \mathcal{L}(f)\}$ and $\mathcal{SAT}_{\mathcal{U}}(f) = \{sat(C) : C \in \mathcal{U}(f)\}$ be the family of CCs of the lower and upper level sets, respectively, with holes filled and consider $\mathcal{SAT}(f) = \mathcal{SAT}_{\mathcal{L}}(f) \cup \mathcal{SAT}_{\mathcal{U}}(f)$. The elements of $\mathcal{SAT}(f)$, called *shapes*, are nested by an inclusion relation and thus the pair $(\mathcal{SAT}(f), \subseteq)$, induces the tree of shapes [13]. The tree of shapes, such as component trees, is a complete representation of an image which can be represented by a compact and non-redundant data structure [14] so that a pixel $p \in \mathcal{D}$ which is associated with the smallest shape or CC of the tree containing it, by the parenthood relationship, is also associated to all the ancestors shapes. Then, we denote by $\mathcal{SC}(\mathcal{T}, p)$ the smallest shape or CC containing p in a tree $\mathcal{T}$.

Extended Trees: In this work, we also consider the extended versions of these trees, i.e., the trees containing all the possible components of an image, defined as follows: Let $Ext(\mathcal{L}(f)) = \{(C, \mu) \in \mathcal{L}(f) \times \mathbb{K} : C \in \mathcal{CC}_4(\mathcal{X}_{\downarrow}^{\mu}(f))\}$ and $Ext(\mathcal{U}(f)) = \{(C, \mu) \in \mathcal{U}(f) \times \mathbb{K} : C \in \mathcal{CC}_8(\mathcal{X}_{\mu}^{\uparrow}(f))\}$ the set of all possible CCs of lower and upper level sets, respectively. Consider $\sqsubseteq$ a relation on $Ext(\mathcal{L}(f))$ (resp. $Ext(\mathcal{U}(f))$), i.e., $\forall (A, i), (B, j) \in Ext(\mathcal{L}(f)), (A, i) \sqsubseteq (B, j) \Leftrightarrow A \subseteq B$ and $i \leq j$ (resp. $\forall (A, i), (B, j) \in Ext(\mathcal{U}(f)), (A, i) \sqsubseteq (B, j) \Leftrightarrow A \subseteq B$ and $i \geq j$). Although, we can similarly build $Ext(\mathcal{SAT}(f))$, in this paper, we only use extended versions of max-tree and min-tree. Therefore, $(Ext(\mathcal{L}(f)), \sqsubseteq)$, $(Ext(\mathcal{U}(f)), \sqsubseteq)$ and $(Ext(\mathcal{SAT}(f)), \sqsubseteq)$ are the version extended of trees $(\mathcal{L}(f), \subseteq)$, $(\mathcal{U}(f), \subseteq)$ and $(\mathcal{SAT}(f), \subseteq)$, respectively. Fig. 1 is an example of these trees for a given image f. Note that, the smallest shape or CC of a tree $\mathcal{T}_f$ containing a $p \in \mathcal{D}$ in $Ext(\mathcal{T}_f)$ is the node $(\mathcal{SC}(\mathcal{T}_f, p), f(p)) \in Ext(\mathcal{T}_f)$.

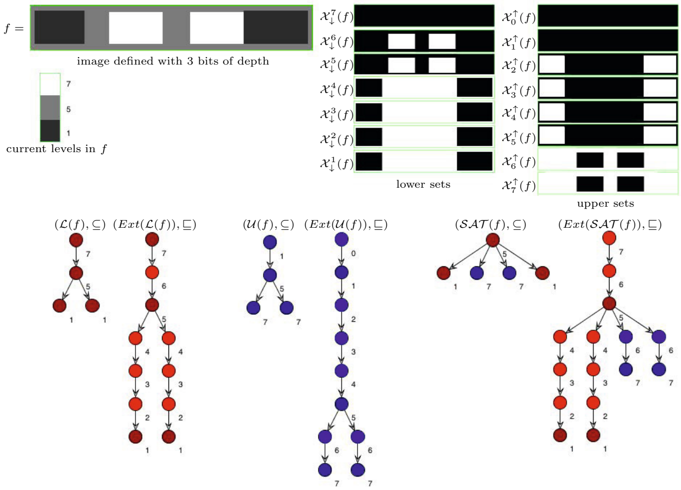

Fig. 1. An example of the construction of the extended trees

2.1 Image Reconstruction and Pruning Operation

As we have seen, an image can be reconstructed from its level sets. Now, we will show how to reconstruct an image f given a tree $\mathcal{T}_f$ (min-tree, max-tree or tree of shapes). This leads us to define the functions $level_{\mathcal{L}} : \mathcal{L}(f) \to \mathbb{K}$, $level_{\mathcal{U}} : \mathcal{U}(f) \to \mathbb{K}$ and $level_{\mathcal{SAT}} : \mathcal{SAT}(f) \to \mathbb{K}$ as follows $level_{\mathcal{L}}(C) = \min\{\lambda - 1 : C \in \mathcal{CC}_4(\mathcal{X}_{\downarrow}^{\lambda}(f)), \lambda \in \mathbb{K}\}$, $level_{\mathcal{U}}(C) = \max\{\lambda : C \in \mathcal{CC}_8(\mathcal{X}_{\lambda}^{\uparrow}(f)), \lambda \in \mathbb{K}\}$ and $level_{\mathcal{SAT}}(C) = f(y)$ such that $y \in \arg\max\{|\mathcal{SC}(\mathcal{T}_f, x)| : x \in C\}$. For the sake of simpler notation, from now on, the subscript $\mathcal{T}_f$ will be dropped from the level function when it is clear from context. Obviously, the function *level* for $Ext(\mathcal{L}(f))$ and $Ext(\mathcal{U}(f))$ is simply $level_{Ext(\mathcal{L})}(C) = \min\{\lambda : (C, \lambda) \in Ext(\mathcal{L}(f))\}$ and $level_{Ext(\mathcal{U})}(C) = \max\{\lambda : (C, \lambda) \in Ext(\mathcal{U}(f))\}$, respectively. Using this function, it is possible to prove that an image $f \in \mathcal{F}(\mathcal{D})$ can be reconstructed from a tree $\mathcal{T}_f$ as follows: $\forall x \in \mathcal{D}, f(x) = level(\mathcal{SC}(\mathcal{T}_f, x))$. In such a case, we write: $f = Rec(\mathcal{T}_f)$. In particular, if f is obtained by $Rec((\mathcal{L}(f), \subseteq))$ (resp. $Rec((\mathcal{U}(f), \subseteq))$, and $Rec((\mathcal{SAT}(f), \subseteq))$) then we call this operation *lower* (resp. *upper* and *shape*) *reconstruction*.

Now, the following definition (Def. 3) gives the conditions for a pruning operation of a tree $(\mathcal{T}, \preceq)$.

Definition 3. *We say that $(\mathcal{T}', \preceq)$ is obtained by a pruning operation of a tree $(\mathcal{T}, \preceq)$ if and only if, $\mathcal{T}' \subseteq \mathcal{T}$, for any $X \in \mathcal{T}'$, $\nexists Y \in (\mathcal{T} \setminus \mathcal{T}')$ such that $X \preceq Y$. In such a case, we write $\mathcal{T}' = \mathcal{P}_{runing}(\mathcal{T})$.*

Following this definition, if $\mathcal{T}_f$ is a tree of an image $f \in \mathcal{F}(\mathcal{D})$, then we say $\mathcal{T}_g$ is the pruned version of $\mathcal{T}_f$ if and only if $\mathcal{T}_g = \mathcal{P}_{runing}(\mathcal{T}_f)$. Also, one can easily see that $\mathcal{T}_g \subseteq \mathcal{T}_f$ and $\mathcal{T}_g$ is still a tree. In addition, since the nodes of $\mathcal{T}_f$ and $\mathcal{T}_g$ are nested by the order relation, it can be proved that, $p \in \mathcal{D}$, $\mathcal{SC}(\mathcal{T}_f, p) \subseteq \mathcal{SC}(\mathcal{T}_g, p)$.

3 Links Between Reconstruction of Pruned Trees and Levelings

Once the pruning operation and the reconstruction of pruned trees are established, we can relate the reconstruction of pruned trees with levelings. We begin observing that, if $\mathcal{T}_g$ is obtained by a pruning operation of a max-tree (resp. min-tree) $\mathcal{T}_f$, then, $\forall p \in \mathcal{D}$, $level(\mathcal{SC}(\mathcal{T}_f, p)) \geq level(\mathcal{SC}(\mathcal{T}_g, p))$ (resp. $level(\mathcal{SC}(\mathcal{T}_f, p)) \leq level(\mathcal{SC}(\mathcal{T}_g, p)))$, thanks to the well-defined ordering of the level sets. Thus, $Rec(\mathcal{T}_f) \geq Rec(\mathcal{T}_g)$ (resp. $Rec(\mathcal{T}_f) \leq Rec(\mathcal{T}_g)$). This property shows that upper (resp. lower) reconstructions are anti-extensive (resp. extensive). Now, we state a simple property, given by Prop. 1, thanks to the well-defined ordering of the level sets.

Proposition 1. *Let* $(\mathcal{U}(f), \subseteq)$ *be the max-tree* (*resp. min-tree* $(\mathcal{L}(f), \subseteq)$) *of an image* f. *Let* $(p, q) \in \mathcal{A}$. *Then,* $f(p) > f(q)$ (*resp.* $f(p) < f(q)$) *if and only if* $\mathcal{SC}(\mathcal{U}(f), p) \subset \mathcal{SC}(\mathcal{U}(f), q)$ (*resp.* $\mathcal{SC}(\mathcal{L}(f), p) \subset \mathcal{SC}(\mathcal{L}(f), q)$).

This fact shows that, if $\mathcal{T}_g$ is obtained by a pruning operation of a max-tree (resp. min-tree) $\mathcal{T}_f$, then $g = Rec(\mathcal{T}_g)$ is a leveling of $f = Rec(\mathcal{T}_f)$, since, for any $(p, q) \in \mathcal{A}$, the following condition holds: $g(p) > g(q) \Rightarrow f(p) \geq g(p) > g(q) = f(q)$ (resp. $g(p) > g(q) \Rightarrow f(p) = g(p) > g(q) \geq f(q)$). Furthermore, if we consider the extended version of the max-tree (resp. min-tree) then there is an equivalence between upper (resp. lower) reconstruction and anti-extensive (resp. extensive) levelings.

Theorem 1. *Anti-extensive (resp. extensive) levelings and upper (resp. lower) reconstructions are equivalent.*

Proof. Let $f \in \mathcal{F}(\mathcal{D})$ be an image and $\mathcal{T}_f = (Ext(\mathcal{L}(f)), \sqsubseteq)$ the extended version of the max-tree of f. Thus, we have: $g \in \mathcal{F}(\mathcal{D})$ is an upper reconstruction of f

$$\begin{aligned}
&\iff g = Rec(\mathcal{T}_g) \text{ such that } \mathcal{T}_g = \mathcal{P}runing(\mathcal{T}_f).\\
&\iff g \leq f \text{ and } \forall (p,q) \in \mathcal{A},\\
&\qquad \begin{cases} \text{either } \mathcal{SC}(\mathcal{T}_f, p) \sqsubseteq \mathcal{SC}(\mathcal{T}_f, q) \sqsubseteq \mathcal{SC}(\mathcal{T}_g, p) \sqsubseteq \mathcal{SC}(\mathcal{T}_g, q) \\ \text{or} \quad \mathcal{SC}(\mathcal{T}_f, p) \sqsubseteq \mathcal{SC}(\mathcal{T}_g, p) \sqsubseteq \mathcal{SC}(\mathcal{T}_f, q) = \mathcal{SC}(\mathcal{T}_g, q) \end{cases}\\
&\iff g \leq f \text{ and } \forall (p,q) \in \mathcal{A},\ \mathcal{SC}(\mathcal{T}_g, p) \sqsubset \mathcal{SC}(\mathcal{T}_g, q)\\
&\qquad\qquad \Rightarrow \begin{cases} \mathcal{SC}(\mathcal{T}_f, p) \sqsubseteq \mathcal{SC}(\mathcal{T}_g, p) \\ \text{and} \\ \mathcal{SC}(\mathcal{T}_g, q) = \mathcal{SC}(\mathcal{T}_f, q) \end{cases}\\
&\iff g \leq f \text{ and } \forall (p,q) \in \mathcal{A},\ level(\mathcal{SC}(\mathcal{T}_g, p)) > level(\mathcal{SC}(\mathcal{T}_g, q))\\
&\qquad\qquad \Rightarrow \begin{cases} level(\mathcal{SC}(\mathcal{T}_f, p)) \geq level(\mathcal{SC}(\mathcal{T}_g, p)) \\ \text{and} \\ level(\mathcal{SC}(\mathcal{T}_g, q) = level(\mathcal{SC}(\mathcal{T}_f, q)) \end{cases}\\
&\iff g \leq f \text{ and } \forall (p,q) \in \mathcal{A},\ g(p) > g(q) \Rightarrow f(p) \geq g(p) \text{ and } g(q) = f(q)\\
&\iff g \text{ is anti-extensive leveling of } f.
\end{aligned}$$

The proof for extensive levelings and lower reconstruction follows similarly.

Now, to establish links between shape reconstructions and levelings, it is necessary to know relations between neighboring pixels in the nodes of the tree. In this sense, Propositions 2, 3, 4 and 5 help us understand how the neighboring pixels are related in the tree. Thus, the Prop. 2 is a corollary of Theo. 2.16 given in [15], the Prop. 3 is a directly consequence of Prop. 1 and the Prop. 5 is a direct consequence of Prop. 4.

Proposition 2. *Let* $(\mathcal{SAT}(f), \subseteq)$ *be the tree of shapes of an image* f. *If* $(p, q) \in \mathcal{A}$ *such that* $f(p) \neq f(q)$ *then* $\mathcal{SC}(\mathcal{SAT}(f), p)$ *and* $\mathcal{SC}(\mathcal{SAT}(f), q)$ *are comparable or disjoint.*

Proposition 3. *Let* $(\mathcal{SAT}(f), \subseteq)$ *be the tree of shapes of an image* f. *If* $(p, q) \in \mathcal{A}$ *such that* $\mathcal{SC}(\mathcal{SAT}(f), p) \subset \mathcal{SC}(\mathcal{SAT}(f), q)$ *and both* $\mathcal{SC}(\mathcal{SAT}(f), p)$ *and* $\mathcal{SC}(\mathcal{SAT}(f), q)$ *belong to* $\mathcal{SAT}_{\mathcal{U}}(f)$ (*resp.* $\mathcal{SAT}_{\mathcal{L}}(f)$), *then* $f(p) > f(q)$ (*resp.* $f(p) < f(q)$).

Proposition 4. *Let* $A \in \mathcal{L}(f) \cup \mathcal{U}(f)$ *such that* $sat(A) \in \mathcal{SAT}(f)$. *If* $B \in Int(A)$ *and* $(p, q) \in \mathcal{A}$ *such that* $p \in B$ *and* $q \notin B$, *then* $q \in sat(A)$.

Proof. Suppose, by contradiction, $q \notin sat(A)$. Then q belongs to the complement of $sat(A)$, i.e., $q \in (\mathcal{D} \setminus sat(A)) = Ext(A) \subseteq (\mathcal{D} \setminus A)$. As $Int(A)$ contains the CCs of $(\mathcal{D} \setminus A)$ included in $sat(A)$ and $B \in Int(A)$, we have that both B and $Ext(A)$ are CCs of $(\mathcal{D} \setminus A)$. With that fact in mind, and, since $p \in B$, $q \in Ext(A)$, and $(p, q) \in \mathcal{A}$, we have that $Ext(A) = B$. But, this is a contradiction, since $q \notin B$. Therefore, $q \in sat(A)$.

Corollary 1. *Let* $A, B \in \mathcal{SAT}(f)$ *such that* $B \subset A$, $A \in \mathcal{SAT}_{\mathcal{U}}(f)$ *and* $B \in \mathcal{SAT}_{\mathcal{L}}(f)$ (*resp.* $A \in \mathcal{SAT}_{\mathcal{L}}(f)$ *and* $B \in \mathcal{SAT}_{\mathcal{U}}(f)$). *If* $(p, q) \in \mathcal{A}$ *such that* $p \in B$ *and* $q \notin B$, *then* $q \in A$.

Proposition 5. *Let* $(p, q) \in \mathcal{A}$ *and let* $\mathcal{SC}(\mathcal{SAT}(f), p)$ *and* $\mathcal{SC}(\mathcal{SAT}(f), q)$ *be elements of* $\mathcal{SAT}_{\mathcal{U}}(f)$ (*resp.* $\mathcal{SAT}_{\mathcal{L}}(f)$). *If* $X \in \mathcal{SAT}(f)$ *such that* $\mathcal{SC}(\mathcal{SAT}(f), p) \subset X \subset \mathcal{SC}(\mathcal{SAT}(f), q)$, *then* $X \in \mathcal{SAT}_{\mathcal{U}}(f)$ (*resp.* $X \in \mathcal{SAT}_{\mathcal{L}}(f)$).

Proof. Suppose, by contradiction, $X \notin \mathcal{SAT}_{\mathcal{U}}(f)$. Thus, $X \in \mathcal{SAT}_{\mathcal{L}}(f)$ since $X \in \mathcal{SAT}(f)$. Then, thanks to Corol. 1, it follows that $q \in X$, since $\mathcal{SC}(\mathcal{SAT}(f), p) \in \mathcal{SAT}_{\mathcal{U}}(f)$ and $\mathcal{SC}(\mathcal{SAT}(f), p) \subset X$. So we have a contradiction, since $X \subset \mathcal{SC}(\mathcal{SAT}(f), q)$ and $\mathcal{SC}(\mathcal{SAT}(f), q)$ is the smallest shape containing q. Therefore, $X \in \mathcal{SAT}_{\mathcal{U}}(f)$.

Theorem 2. *Shape reconstructions are levelings.*

Proof. Let $f \in \mathcal{F}(\mathcal{D})$ be an image and $\mathcal{T}_f$ be the tree of shapes of f. Then, $g \in \mathcal{F}(\mathcal{D})$ is a shape reconstruction of f if and only if $g = Rec(\mathcal{T}_g)$ such that $\mathcal{T}_g = \mathcal{P}runing(\mathcal{T}_f)$. To prove that g is leveling of f, we just need to check if, $\forall (p, q) \in \mathcal{A}$, the definition of leveling holds, that is, $g(p) > g(q) \Rightarrow f(p) \geq g(p)$ and $g(q) \geq f(q)$.

Let us consider two cases, where $f(p) = f(q)$ and $f(p) \neq f(q)$. In the first case, g meets the definition of leveling by vacuity. In the second case, we have $\mathcal{SC}(\mathcal{T}_f, p) \neq \mathcal{SC}(\mathcal{T}_f, q)$ and the pruning in $\mathcal{T}_f$, which generates $\mathcal{T}_g$, can: (1) preserve both nodes; or (2) eliminate both nodes; or (3) eliminate one of the nodes. See in Fig. 2 the illustrations of pruning settings.

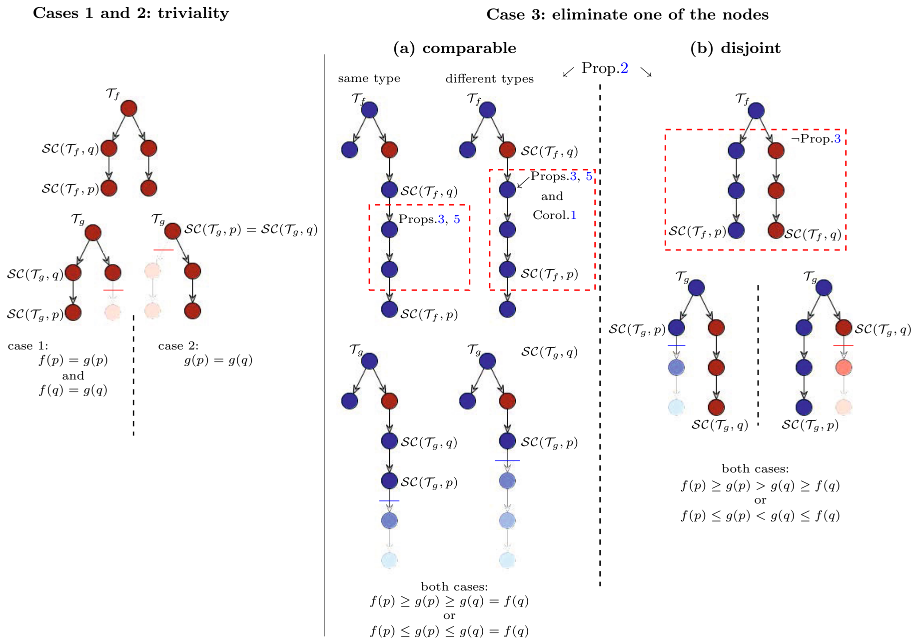

Fig. 2. Illustrations of pruning settings

1. If both nodes are preserved, then $f(p) = g(p)$ and $f(q) = g(q)$ that meets the definition of leveling (see Fig. 2 - Case 1);
2. If both nodes are eliminated and comparable, then $g(p) = g(q)$, and thus the definition of leveling is valid by vacuity (see Fig. 2 - Case 2). In case they are eliminated and not comparable, see case $3(b)$;
3. If only one of the two nodes is eliminated, then, thanks to Prop. 2, $\mathcal{SC}(\mathcal{T}_f, p)$ and $\mathcal{SC}(\mathcal{T}_f, q)$ are either comparable or disjoint (see Fig. 2 - Case 3).
   (a) If $\mathcal{SC}(\mathcal{T}_f, p)$ and $\mathcal{SC}(\mathcal{T}_f, q)$ are comparable (see Fig. 2 - Case 3(a)), then suppose without loss of generality that $\mathcal{SC}(\mathcal{T}_f, p) \subset \mathcal{SC}(\mathcal{T}_f, q)$. Thus, $\mathcal{SC}(\mathcal{T}_f, p) \neq \mathcal{SC}(\mathcal{T}_g, p)$ and $\mathcal{SC}(\mathcal{T}_f, q) = \mathcal{SC}(\mathcal{T}_g, q)$ and consequently $g(q) = f(q)$.
      – If $\mathcal{SC}(\mathcal{T}_f, p)$ and $\mathcal{SC}(\mathcal{T}_f, q)$ belong to $\mathcal{SAT}_{\mathcal{U}}(f)$ (resp. $\mathcal{SAT}_{\mathcal{L}}(f)$) then thanks to Prop. 5, follows that $\mathcal{SC}(\mathcal{T}_g, p) \in \mathcal{SAT}_{\mathcal{U}}(f)$. Thus, thanks to Prop. 3, follows that $f(p) \geq g(p) \geq g(q) = f(q)$ (resp. $f(p) \leq g(p) \leq g(q) = f(q)$), that meets the definition of leveling (see left tree of Fig. 2 - Case 3(a));
      – If $\mathcal{SC}(\mathcal{T}_f, p)$ and $\mathcal{SC}(\mathcal{T}_f, q)$ are of different types, then thanks to Corollary 1, it follows that $\forall (r, s) \in \mathcal{A}$ such that $r \in \mathcal{SC}(\mathcal{T}_f, p)$ and $s \notin \mathcal{SC}(\mathcal{T}_f, p)$ follows that $s \in \mathcal{SC}(\mathcal{T}_f, q)$. Thus, $\mathcal{SC}(\mathcal{T}_f, p) \subset \mathcal{SC}(\mathcal{T}_f, s) \subseteq \mathcal{SC}(\mathcal{T}_f, q)$ and consequently either $f(p) < f(s) \leq f(q) \Rightarrow f(p) \leq g(p) \leq g(q) = f(q)$ or $f(p) > f(s) \geq f(q) \Rightarrow f(p) \geq g(p) \geq$

$g(q) = f(q)$, that meets the definition of leveling (see right tree of Fig. 2 - Case 3(a)).

(b) If $\mathcal{SC}(\mathcal{T}_f, p)$ and $\mathcal{SC}(\mathcal{T}_f, q)$ are disjoint (see Fig. 2 - Case 3(b)), then they are of different types (see Prop. 3). Moreover, $\mathcal{SC}(\mathcal{T}_f, p)$ and $\mathcal{SC}(\mathcal{T}_f, q)$ are not in the same branch of $\mathcal{T}_f$. In this case, certainly there exists a node $\mathcal{SC}(\mathcal{T}_f, r)$ which is common ancestor of both nodes $\mathcal{SC}(\mathcal{T}_f, p)$ and $\mathcal{SC}(\mathcal{T}_f, q)$. Thus, $\mathcal{SC}(\mathcal{T}_f, r)$ is the same type of $\mathcal{SC}(\mathcal{T}_f, p)$ or $\mathcal{SC}(\mathcal{T}_f, q)$. Then, either $f(p) > f(r) > f(q)$ or $f(p) < f(r) < f(q)$. Without loss of generality, assume $f(p) > f(r) > f(q)$. Thus, if $\mathcal{SC}(\mathcal{T}_f, p)$ is removed and $\mathcal{SC}(\mathcal{T}_f, q)$ is preserved, then we have that $\mathcal{SC}(\mathcal{T}_f, p) \subset \mathcal{SC}(\mathcal{T}_g, p) \subseteq \mathcal{SC}(\mathcal{T}_f, r)$ and $\mathcal{SC}(\mathcal{T}_g, q) = \mathcal{SC}(\mathcal{T}_f, q)$ and consequently $f(p) > g(p) \geq f(r)$ and $g(q) = f(q)$. Therefore, $f(p) > g(p) \geq f(r) > g(q) = f(q) \Rightarrow f(p) > g(p) > g(q) = f(q)$ which in turn meets the leveling definition. But, if $\mathcal{SC}(\mathcal{T}_f, p)$ is preserved and $\mathcal{SC}(\mathcal{T}_f, q)$ is removed, then we have $\mathcal{SC}(\mathcal{T}_g, p) = \mathcal{SC}(\mathcal{T}_f, p)$ and $\mathcal{SC}(\mathcal{T}_f, q) \subset \mathcal{SC}(\mathcal{T}_g, q) \subseteq \mathcal{SC}(\mathcal{T}_f, r)$ and consequently $f(p) = g(p)$ and $f(r) \geq g(q) > f(q)$. Therefore, $f(p) = g(p) > g(q) > f(q)$ which in turn meets the leveling definition.

4 Morphological Reconstruction Based on a Marker Image

In this section, present a new and fast algorithm for morphological reconstructions based on a marker image by reconstruction of pruned trees. In fact, pruning strategy based on marker image is not well explored in the literature. The strategy is to use the marker image to determine the place of pruning in extended versions of max-trees and min-trees.

The reconstruction operator is relatively simple in binary case, which consists in extracting the CCs of an binary image $X \subseteq \mathcal{D}$ which are marked by another binary image $M \subseteq X$. The binary images M and X are respectively called marker and mask. Then, the reconstruction $\rho_B(X, M)$ of mask X from marker M is the union of all the CCs of X which contain at least one pixel of M, i.e., $\rho_B(X, M) = \{C \in \mathcal{CC}(X) : C \cap M \neq \emptyset\}$ [12].

To extend the reconstruction operator to grayscale images, we recall that any increasing operator defined for binary images can be extended to grayscale images through threshold decomposition [16]. Thus, given a mask image $f \in \mathcal{F}(\mathcal{D})$ and a marker image $g \in \mathcal{F}(\mathcal{D})$ such that $g \leq f$, we have:

$$\forall p \in \mathcal{D}, [\rho(f,g)](p) = \sup\{\mu \in \mathbb{K} : p \in \rho_B(\mathcal{X}^{\uparrow}_{\mu}(f), \mathcal{X}^{\uparrow}_{\mu}(g))\}$$
$$\iff [\rho(f,g)](p) = \sup\{\mu \in \mathbb{K} : p \in \{C \in \mathcal{CC}(\mathcal{X}^{\uparrow}_{\mu}(f)) : C \cap \mathcal{X}^{\uparrow}_{\mu}(g) \neq \emptyset\}\}$$
$$\iff [\rho(f,g)](p) = \sup\{\mu \in \mathbb{K} : \mathcal{SC}(\mathcal{U}(f), p) \cap \mathcal{X}^{\uparrow}_{\mu}(g) \neq \emptyset\}.$$

Note that, to construct $\rho(f,g)$ for $p \in \mathcal{D}$, we need to find the smallest set $\mathcal{X}^{\uparrow}_{\mu}(g)$ such that $\mathcal{X}^{\uparrow}_{\mu}(g) \cap \mathcal{SC}(\mathcal{U}(f), p) \neq \emptyset$. Fortunately, this can be expressed as a pruning operation in the extended version of $(\mathcal{U}(f), \subseteq)$ as follows: remove (resp. preserve) all nodes $(C, \mu) \in Ext(\mathcal{U}(f))$, if and only if, there exists a pixel $p \in C$ such that $\mu > g(p)$ (resp. $g(p) \geq \mu$). Therefore, $\rho(f,g) = Rec((\mathcal{T}_{\mathcal{U}}, \sqsubseteq))$ such that $\mathcal{T}_{\mathcal{U}} = \{(C, \mu) \in Ext(\mathcal{U}(f)) : \bigvee_{p \in C} g(p) \geq \mu\}$.

Following these ideas, we present the Algorithm 1 to computes the reconstruction by dilation. This algorithm makes use of a priority queue to process the pixels of the marker image in an orderly manner and so we do not reprocess the nodes that already were visited.

Algorithm 1. Compute the reconstruction by dilation

Image reconstruction(*Max-tree* $\mathcal{T}_f$, *Image marker* g) **begin**
    Initialize priority queue $\mathcal{Q}$
    foreach $C \in Ext(\mathcal{T}_f)$ **do** remove$[C]$ = true **foreach** $p \in \mathcal{D}$ *such that* $g(p) \leq f(p)$ **do** add $(\mathcal{Q}, p, g(p))$
    while $\mathcal{Q}$ *is not empty* **do**
        p = removeMaxPriority($\mathcal{Q}$);
        $(C, \mu) = \mathcal{SC}(Ext(\mathcal{T}_f), p)$
        if C *was not processed* **then**
            while remove$[C]$ *is true* AND $\mu > g(p)$ **do**
                C = parent (C)
            remove[C] = false
            while parent(C) *is not null* AND remove[parent(C)] *is true* **do**
                remove[parent(C)] = false
                C = parent(C)
    /* reconstruction of pruned tree, i.e., $Rec(\mathcal{P}runing(\mathcal{T}_f))$ such that $\mathcal{P}runing(\mathcal{T}_f) = \{(C, \mu) \in Ext(\mathcal{T}_f) : \text{remove}[C] = \text{false}\}$ */
    Initialize queue $\mathcal{Q}_{fifo}$
    enqueue($\mathcal{Q}_{fifo}$, root($Ext(\mathcal{T}_f)$))
    while $\mathcal{Q}_{fifo}$ *is not empty* **do**
        (C, μ) = dequeue($\mathcal{Q}_{fifo}$)
        if remove$[C]$ *is false* **then** $\forall p \in C, f(p) = \mu$ **foreach** $S \in$ children(C) **do** enqueue($\mathcal{Q}_{fifo}, S$)
    /* The image $f = \rho(f, g)$ is the result of reconstruction by dilation of mask image f using the marker image g. */
    return f

The reconstruction using upper level sets can also be defined through the geodesic dilation of f with respect to g iterated until stability. In this respect, a traditional fast algorithm for computing reconstruction by dilation has been proposed by Luc Vincent [12]. Thus, we show in Fig. 3 a graphic for a comparison simples.

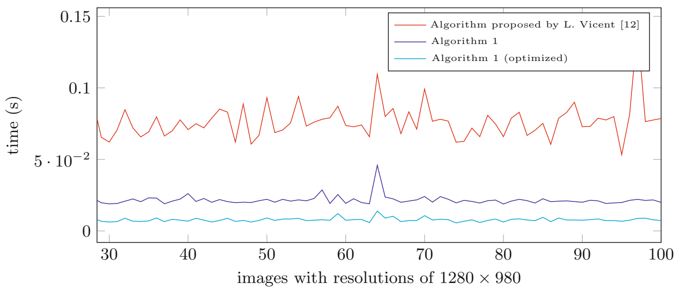

Fig. 3. Comparison of computation time. In this comparison each image of ICDAR dataset [17] is tested with 10 different marker images and the average value is plotted.

Of course, by duality, one can obtain the reconstruction by erosion ρ^* using similarly lower level sets. These two morphological reconstructions techniques are at basis of numerous operators, such as: opening and closing by reconstruction, top-hat by reconstruction, h-basins, h-domes and others [3, 12].

5 Scale-Space Representation Through a Sequence of Reconstructions of Successive Prunings

It is already known that some operators can be obtained by reconstructions of pruned trees, as for example: attribute opening (resp. closing), grain filters, and others. From the previous section, we showed that some operators based on marker images also can be obtained by pruning operation such as opening by reconstruction, top-hat by reconstruction, h-basins and others. Taking advantage of this property, we will show in this section a way to build scale-space representation by a sequence of successive prunings.

Let $\mathcal{T}_f$ be the tree (max-tree, min-tree or tree of shapes) that represents an image f. Since levelings can be nested to create a space-scale decomposition of an image, by Theo. 1 and 2, we have that $Rec(\mathcal{P}runing(\mathcal{P}runing(\mathcal{T}_f)))$ is a leveling of $Rec(\mathcal{P}runing(\mathcal{T}_f))$ and $Rec(\mathcal{P}runing(\mathcal{T}_f))$ is a leveling of $Rec(\mathcal{T}_f) = f$. Then, by transitivity, we also have that $Rec(\mathcal{P}runing(\mathcal{P}runing(\mathcal{T}_f)))$ is a leveling of $Rec(\mathcal{T}_f)$. This shows that the tree generates a family of levelings that further simplifies the image f, thus constituting a morphological space-scale and this leads us to Prop. 6.

Proposition 6. *Let $\mathcal{T}_f$ be the tree (max-tree, min-tree or tree of shapes) that represents an image f. Then, the sequence of reconstructions of successive prunings ($g_0 = Rec(\mathcal{T}_f)$, $g_1 = Rec(\mathcal{P}runing(\mathcal{T}_f))$, $g_2 = Rec(\mathcal{P}runing(\mathcal{P}runing(\mathcal{T}_f)))$, ..., $g_n = Rec(\mathcal{P}runing(...(\mathcal{P}runing(\mathcal{P}runing(\mathcal{T}_f)))))$) is a space-scale of levelings such that g_k is a leveling of g_l for all $0 \leq l \leq k \leq n$.*

Thus, we can build through successive pruning: (1) scale-space based on attributes from increasing criteria on attributes and so generate scale-space of opening, closing and grain filter by attribute (or extinction values) and others; (2) scale-space based on marker images from a family of markers and so generate scale-space of reconstruction by opening and closing, top-hat by reconstruction, h-basins, h-domes and others. In addition, following F. Meyer [2], from Eq. 1 is possible define the self-dual reconstruction combining the reconstruction by dilation and erosion, and so generate scale-space of self-dual reconstruction. In fact, different families of markers may be used to generate a morphological scale-space based on levelings as shown by F. Meyer in [4].

$$\forall x \in \mathcal{D}, [\nu(f,g)](x) = \begin{cases} [\rho(f, g \wedge f)](x) & \text{, if } g(x) < f(x), \\ [\rho^*(f, g \vee f)](x) & \text{, if } g(x) > f(x), \\ f(x) & \text{, otherwise.} \end{cases} \tag{1}$$

Based on this idea, Fig. 4 presents some images of a scale-space generated with marker images produced by alternate sequential filtering.

Fig. 4. Some images of a scale-space generated with markers produced by alternate sequential filtering

6 Application Example

In many application in Image Processing and Analysis, the objects of interest which must be detected, measured, segmented, or recognized in an image are, in general case, not in a fixed but in many scales. For such situations, several multi-scale operators have been developed over the last few decades. In this sense, this section briefly illustrates (see Fig. 5) the application of some residual operators defined on an scale-space based on levelings [18, 19, 20]. They are: ultimate attribute opening (UAO) (resp. closing (UAC)) [20] and ultimate grain filters (UGF) [19]. They belong to a larger class of residual operators that we call ultimate levelings and defined from a indexed family of levelings $\{\psi_i : i \in I\}$ such that $i, j \in I$, $i \leq j \Rightarrow \psi_j$ is a leveling of ψ_i. Thus, the an ultimate leveling is defined by $\mathcal{R}_\theta(f) = \mathcal{R}_\theta^+(f) \vee \mathcal{R}_\theta^-(f)$ where $\mathcal{R}_\theta^+(f) = \sup\{r_i^+(f) : r_i^+(f) = [\psi_i(f) - \psi_{i+1}(f) \vee 0]\}$ and $\mathcal{R}_\theta^-(f) = \sup\{r_i^-(f) : r_i^-(f) = [\psi_{i+1}(f) - \psi_i(f) \vee 0]\}$. They can be implemented efficiently through of a max-tree, min-tree or tree of shapes [21, 19].

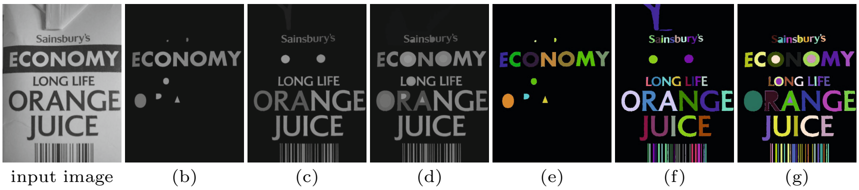

Fig. 5. Example of extraction of contrast and segmentation using UAO (b) and (e), UAC (c) and (f), and UGF (d) and (g)

7 Conclusion

In this work, we have presented scale-space representations of an image based on levelings through hierarchies of level sets (component trees and tree of shapes). For that, we first proved the main result of this paper in Section 3 that reconstructions of pruned trees are levelings. After that, in Section 4 we present a new and fast algorithm for computing the reconstruction based on marker images from component trees. Finally, in Section 5 we show how to build morphological scale-spaces based on levelings through the reconstructions of successive pruning operations (whether based on increasing attributes or marker images).

Acknowledgements. We would like to thank the financial support from CAPES, CNPq, and FAPESP (grant #2011/50761-2).

References

[1] Salembier, P., Serra, J.: Flat zones filtering, connected operators, and filters by reconstruction. IEEE Transactions on Image Processing 4, 1153–1160 (1995)
[2] Meyer, F.: From connected operators to levelings. In: Mathematica Morphology and its Applications to Image and Signal Processing, pp. 191–198 (1998)
[3] Meyer, F.: The levelings. In: Mathematical Morphology and its Applications to Image and Signal Processing, pp. 199–206 (1998)
[4] Meyer, F., Maragos, P.: Nonlinear scale-space representation with morphological levelings. J. Vis. Commun. Image Represent. 11(2), 245–265 (2000)
[5] Meyer, F., Maragos, P.: Morphological scale-space representation with levelings. In: Nielsen, M., Johansen, P., Fogh Olsen, O., Weickert, J. (eds.) Scale-Space 1999. LNCS, vol. 1682, pp. 187–198. Springer, Heidelberg (1999)
[6] Salembier, P., Oliveras, A., Garrido, L.: Anti-extensive connected operators for image and sequence processing. IEEE Trans. on Image Processing 7, 555–570 (1998)
[7] Caselles, V., Monasse, P.: Grain filters. J. M. Imaging Vision 17(3), 249–270 (2002)
[8] Xu, Y., Geraud, T., Najman, L.: Morphological filtering in shape spaces: Applications using tree-based image representations. In: 2012 21st International Conference on Pattern Recognition (ICPR), pp. 485–488 (2012)
[9] Dalla Mura, M., Benediktsson, J.A., Bruzzone, L.: Self-dual attribute profiles for the analysis of remote sensing images. In: Soille, P., Pesaresi, M., Ouzounis, G.K. (eds.) ISMM 2011. LNCS, vol. 6671, pp. 320–330. Springer, Heidelberg (2011)
[10] Salembier, P., Wilkinson, M.: Connected operators. IEEE Signal Processing Magazine 26(6), 136–157 (2009)
[11] Terol-Villalobos, I.R., Vargas-Vázquez, D.: Openings and closings with reconstruction criteria: a study of a class of lower and upper levelings. Journal of Electronic Imaging 14(1), 013006–013006–11 (2005), doi:10.1117/1.1866149
[12] Vincent, L.: Morphological grayscale reconstruction in image analysis: applications and efficient algorithms. IEEE Trans. on Image Processing 2(2), 176–201 (1993)
[13] Caselles, V., Meinhardt, E., Monasse, P.: Constructing the tree of shapes of an image by fusion of the trees of connected components of upper and lower level sets. Positivity 12(1), 55–73 (2008)
[14] Géraud, T., Carlinet, E., Crozet, S., Najman, L.: A quasi-linear algorithm to compute the tree of shapes of nd images. In: Hendriks, C.L.L., Borgefors, G., Strand, R. (eds.) ISMM 2013. LNCS, vol. 7883, pp. 98–110. Springer, Heidelberg (2013)
[15] Caselles, V., Monasse, P.: Geometric Description of Images As Topographic Maps, 1st edn. Springer Publishing Company, Incorporated (2009)
[16] Maragos, P., Ziff, R.: Threshold superposition in morphological image analysis systems. IEEE Trans. on Pattern Analysis and Machine Intel. 12, 498–504 (1990)
[17] Lucas, S.M., Panaretos, A., Sosa, L., Tang, A., Wong, S., Young, R.: Icdar 2003 robust reading competitions. ICDAR 2003 2, 682 (2003)
[18] Alves, W., Morimitsu, A., Castro, J., Hashimoto, R.: Extraction of numerical residues in families of levelings. In: 2013 26th SIBGRAPI - Conference on Graphics, Patterns and Images (SIBGRAPI), pp. 349–356 (2013)
[19] Alves, W.A.L., Hashimoto, R.: Ultimate grain filter. In: 2014 IEEE International Conference on Image Processing (ICIP), Paris, France, pp. 2953–2957 (2014)
[20] Marcotegui, B., Hernández, J., Retornaz, T.: Ultimate opening and gradual transitions. In: Soille, P., Pesaresi, M., Ouzounis, G.K. (eds.) ISMM 2011. LNCS, vol. 6671, pp. 166–177. Springer, Heidelberg (2011)
[21] Fabrizio, J., Marcotegui, B.: Fast implementation of the ultimate opening. In: Proc. 9th International Symposium on Mathematical Morphology, pp. 272–281 (2009)