

Multi-image segmentation: A collaborative approach based on binary partition trees

Jimmy Francky Randrianasoa, Camille Kurtz, Eric Desjardin, Nicolas Passat

▶ **To cite this version:**

Jimmy Francky Randrianasoa, Camille Kurtz, Eric Desjardin, Nicolas Passat. Multi-image segmentation: A collaborative approach based on binary partition trees. International Symposium on Mathematical Morphology (ISMM), 2015, Reykjavik, Iceland. pp.253-264, 10.1007/978-3-319-18720-4_22 . hal-01695074

HAL Id: hal-01695074
https://hal.univ-reims.fr/hal-01695074

Submitted on 15 Feb 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Multi-Image Segmentation: A Collaborative Approach Based on Binary Partition Trees*

Jimmy Francky Randrianasoa[1], Camille Kurtz[2], Éric Desjardin[1], and Nicolas Passat[1]

[1] Université de Reims Champagne-Ardenne, CReSTIC, France
[2] Université Paris-Descartes, LIPADE, France
jimmy.randrianasoa@univ-reims.fr

Abstract. Image segmentation is generally performed in a "one image, one algorithm" paradigm. However, it is sometimes required to consider several images of a same scene, or to carry out several (or several occurrences of a same) algorithm(s) to fully capture relevant information. To solve the induced segmentation fusion issues, various strategies have been already investigated for allowing a consensus between several segmentation outputs. This article proposes a contribution to segmentation fusion, with a specific focus on the "*n* images" part of the paradigm. Its main originality is to act on the segmentation research space, *i.e.*, to work at an earlier stage than standard segmentation fusion approaches. To this end, an algorithmic framework is developed to build a binary partition tree in a collaborative fashion, from several images, thus allowing to obtain a unified hierarchical segmentation space. This framework is, in particular, designed to embed consensus policies inherited from the machine learning domain. Application examples proposed in remote sensing emphasise the potential usefulness of our approach for satellite image processing.

Keywords: segmentation fusion, morphological hierarchies, multi-image, collaborative strategies, binary partition tree, remote sensing.

1 Introduction

In image processing / analysis, segmentation is a crucial task. The concept of segmentation is also quite generic, in terms of semantics (from low-level definition of homogeneous areas to high-level extraction of specific objects), in terms of definition (object versus background, or total partition of the image support), and in terms of algorithmics.

The principal invariant of segmentation is the "one image, one algorithm" paradigm. Indeed, for a given application, a specific algorithm is generally chosen (or designed) with respect to its adequacy with the considered image processing / analysis problem; parametrized with respect to the physical / semantic properties of the targeted images; and then applied (once) on a given image, or on each image of a given dataset.

Since segmentation is an ill-posed problem, it is plain that the result of a segmentation algorithm applied on an image cannot be completely satisfactory. Based on this assertion, it is sometimes relevant to relax the "one image, one algorithm" paradigm,

* This research was partially funded by the French *Agence Nationale de la Recherche* (Grant Agreements ANR-10-BLAN-0205 and ANR-12-MONU-0001).

either by operating one algorithm on several images of a same scene – in order to enrich / improve the information provided as input – or by applying several algorithms, or the same algorithm with several parameter sets – in order to enrich / improve the information provided as output.

In the literature, these "n images, one algorithm" and "one image, n algorithms" paradigms are dealt with under the common terminology of *segmentation fusion*. Most of the time, the strategy consists of computing n segmentation maps, and to develop a consensus strategy to gather and unify these n results into a single, assuming that an accurate global result will emerge from a plurality of less accurate ones.

This fusion strategy has also been proposed – and intensively studied – in the research field of machine learning, especially in clustering [1]. Intrinsically, the problems are the same, with the only – but important – difference that image segmentation, by contrast with general data clustering, implies the handling of the spatial organisation of the processed data.

In this article, we propose a new approach for segmentation fusion, that focuses on the "n images, one algorithm" paradigm. The relevance and novelty of this approach, that lies in the framework of morphological hierarchies and connected operators [2], derives from the early stage where the fusion occurs. By contrast with most segmentation fusion approaches, we do not intend to fuse several segmentation maps obtained from several images, but we directly develop a consensus when defining the research space, *i.e.*, during the construction of a hierarchy that models the n images.

This strategy, that acts at the research space level, induces two important side effects. First, it allows us to rely on hierarchical approaches for segmentation, and thus to propose a rich and versatile segmentation framework, that can be easily instantiated according to the application field. Second, by operating the fusion on the internal data-structures involved in the algorithmic construction of the hierarchy, instead of spatial regions of the segmentation maps, we can directly benefit from all the strategies devoted to "non-spatial" data fusion, previously proposed by the machine learning community.

Our approach relies on Binary Partition Trees (BPTs) [3] as hierarchical model. This choice is motivated both by the possibility to tune a BPT construction, by contrast with other tree structures more deterministically deriving from the images (namely, component-trees or trees of shapes). It is also justified by the frequent and successful application of BPTs in remote sensing, where the use of multiple images of a same scene is frequent, such as the use of fusion strategies for clustering issues.

This article is organized as follows. In Section 2, a brief overview of related works is proposed, describing previous segmentation fusion strategies, and applications of BPTs and hierarchies for analysing multiple remote sensing images. In Section 3, the general data-structure / algorithmic framework for building a BPT from several images is described, and various consensus strategies for instantiating this framework are discussed. In Section 4, experiments are performed on remote sensing images, in order to illustrate the relevance and usefulness of BPT segmentation from several images, by considering two application cases, namely mono-date multi-imaging on urban areas, and multi-date imaging on agriculture areas. In Section 5, a discussion concludes this article by emphasising perspectives and potential further works.

2 Related Works

2.1 Segmentation Fusion

Segmentation fusion consists of establishing a consensus between several segmentation maps. The taxonomy of segmentation fusion is directly mapped on that of segmentation.

On the one hand, segmentation can be viewed as a process that aims to extract one structure (the object) versus the remainder of the image (the background). Typical examples of such segmentation strategies are deformable models, graph-cuts, *etc.* In this context, segmentation fusion can be interpreted as a geometrical problem of shape or contour averaging / interpolation (see, *e.g.*, [4, 5]). This subfamily of segmentation fusion methods, mainly used in medical imaging, is out of the scope of our study.

On the other hand, segmentation can be viewed as a process that aims to define a partition of the whole image, in order to extract meaningful homogeneous areas. Examples of such segmentation strategies are watersheds, split-and-merge and, more generally, connected operators. Clustering methods also enter in this category, with the difference that they generally do not take into account the spatial organisation of the image data.

Following a machine learning vision, some methods handled segmentation fusion via clustering ensemble [1]; a comparative study can be found in [6]. It was also proposed to interpret the information gathered at each pixel in the various segmentation maps as feature vectors then involved in optimization procedures. In particular, a two-stage K-MEANS approach was considered in [7], or probabilistic frameworks in [8, 9].

Other approaches explicitly took into account the spatial organisation of the segmentation maps. In pioneering works [10], images of a same scene obtained from various modalities were merged and optimized based on edge information. Later, statistical analysis of the co-occurrence probability of neighbour pixels was considered to improve the accuracy of a partition from several versions with slightly disturbed borders [11].

More recently, connected operators were also involved. Stochastic watersheds [12] were introduced as a solution for improving the robustness of marker-based watersheds, by fusing the results obtained from different initial seeds. This approach further inspired a stochastic minimum spanning forest approach [13] for segmentation / classification of hyperspectral remote sensing images. Random walkers [14] were also considered for segmenting a graph generated from the information derived from the degree of accordance between different segmentation maps.

2.2 Morphological Hierarchies, Multi-Images and Remote Sensing

Morphological hierarchies associated to connected operators [2] have been successfully involved in image segmentation in the "one image, one algorithm" paradigm. In the associated tree-structures, the nodes model homogeneous regions in the image whereas the edges represent their inclusion relations.

Classical trees, as component-trees [15] or trees of shapes [16], allow us to perform hierarchical segmentation by fusion of flat zones. These structures provide as output partial partitions of an image with tunable levels of details. However, they strongly rely on the image intensity, which is not compliant with the specificities of satellite images.

A first solution to deal with this issue relies on the constrained connectivity [17]. The connectivity relation generates a partition of the image domain; fine to coarse partition hierarchies are then produced by varying a threshold value associated with each connectivity constraint. In a different manner, the BPT [3] reflects a (chosen) similarity measure between neighbouring regions, and models the hierarchy between these regions. The BPTs were used to segment various types of satellite images [18, 19]. However, objects in satellite images appear often too much heterogeneous to be correctly segmented from a single image. It appears then relevant to consider n images of a same scene to enrich the data space and improve the capability of segmentation hierarchies.

In this context, efforts were conducted to extend morphological hierarchies for handling n images. In [20] an extension of the BPT model was proposed to deal with multiresolution satellite images by considering one hierarchy per resolution image. In a similar vein, an approach based on multiple morphological hierarchies was developed in [21] to segment a multispectral image. The originality of this approach was to build independently one hierarchy per radiometric band and combine them to select meaningful segments, optimizing a mixed spectral / connectivity measure.

These recent works show the interest of considering a multi-image paradigm to enhance the hierarchical segmentation of complex structures from remote sensing images. Multi-image fusion segmentation methods take advantage of the complementarity of available data, and can be adapted for morphological hierarchies by interpreting the "n images, one algorithm" as "n images, one hierarchy". By performing collaborations where the fusions occur relatively to the different image contents, it is then possible to gradually build a unique consensual hierarchy, which can be used to detect complex patterns while avoiding (spectral, semantic, ...) noise appearing in a single image.

Based on these considerations, we propose in the next section a hierarchical collaborative segmentation method, extending the BPT to deal with multi-images. By contrast with classical segmentation fusion approaches, we do not intend to fuse several segmentation maps, but we directly develop a consensus during the construction of the hierarchy that models the n images. To build this consensual hierarchy, we propose different algorithmic consensus strategies (inspired from ensemble clustering strategies previously developed by the machine learning community) that seek widespread agreement among the content of different images of a same scene.

3 Building a Binary Partition Tree From Several Images

We first recall the BPT construction algorithm. In particular, we focus on the data-structure point of view, which is the cornerstone of our contribution. Then, we describe our generalization of this algorithm to deal with several images. Various families of consensus strategies for instantiating this framework are finally discussed.

3.1 The Standard BPT Construction

Summary of the algorithm [3] A BPT is a hierarchical representation of an image. More precisely, it is a binary tree, whose each node is a connected region. Each of these nodes is either a leaf – then corresponding to an "elementary" region – or an internal

node, modelling the union of the regions of its two children nodes. The root is the node corresponding to the support of the image. Practically, a BPT is built from its leaves – provided by an initial partition of the image support – to its root, in a bottom-up fashion, by iteratively choosing and merging two adjacent regions which minimize a merging criterion that reflects, *e.g.*, the spectral and / or geometrical likenesses of the regions.

Structural description of the algorithm An image is a function $I : \Omega \to V$ that associates to each point x of the finite set Ω a value $I(x)$ of the set V. A crucial point for building a BPT – and more generally for performing connected filtering – is to take into account the structure of Ω. More precisely, it is mandatory to model the fact that two points x and y of Ω are neighbours. This is done by defining an adjacency A_Ω, *i.e.*, an irreflexive, symmetric, binary relation on Ω. In other words, (Ω, A_Ω) is a graph that models the structure of the image space.

Let us now consider an initial partition $\mathcal{L}$ of Ω. (Each node $L \subseteq \Omega$ of $\mathcal{L}$ is generally assumed to be connected with respect to A_Ω.) This partition $\mathcal{L}$ defines the set of the leaves of the BPT we are going to build (*e.g.*, $\mathcal{L}$ can be the set of the image flat zones).

For any partition $\mathcal{P}$ of Ω (and in particular for $\mathcal{L}$) we can define an adjacency inherited from that of Ω. More precisely, we say that two distinct nodes $N_1, N_2 \in \mathcal{P}$ are adjacent if there exist $x_1 \in N_1$ and $x_2 \in N_2$ such that (x_1, x_2) is an edge of A_Ω, *i.e.*, x_1 and x_2 are adjacent in (Ω, A_Ω). This new adjacency relation $A_\mathcal{P}$ is also irreflexive and symmetric. In the case of $\mathcal{L}$, it allows us to define a graph $\mathfrak{G}_\mathcal{L} = (\mathcal{L}, A_\mathcal{L})$ that models the structure of the partition of the image I.

The BPT is the data-structure that describes the progressive collapse of $\mathfrak{G}_\mathcal{L}$ onto the trivial graph $(\Omega, \emptyset)$. This process consists of defining a sequence $(\mathfrak{G}_i = (\mathcal{N}_i, A_{\mathcal{N}_i}))_{i=0}^n$ (with $n = |\mathcal{L}| - 1$) as follows. First, we set $\mathfrak{G}_0 = \mathfrak{G}_\mathcal{L}$. Then, for each i from 1 to n, we choose the two nodes N_{i-1} and N'_{i-1} of $\mathfrak{G}_{i-1}$ linked by the edge $(N_{i-1}, N'_{i-1}) \in A_{\mathcal{N}_{i-1}}$ that minimizes a merging criterion, and we define $\mathfrak{G}_i$ such that $\mathcal{N}_i = \mathcal{N}_{i-1} \setminus \{N_{i-1}, N'_{i-1}\} \cup \{N_{i-1} \cup N'_{i-1}\}$; in other words, we replace these two nodes by their union. The adjacency $A_{\mathcal{N}_i}$ is defined accordingly from $A_{\mathcal{N}_{i-1}}$: we remove the edge (N_{i-1}, N'_{i-1}), and we replace each edge (N_{i-1}, N''_{i-1}) and / or (N'_{i-1}, N''_{i-1}) by an edge $(N_{i-1} \cup N'_{i-1}, N''_{i-1})$ (in particular, two former edges may be fused into a single).

From a structural point of view, the BPT $\mathfrak{T}$ is the Hasse diagram of the partially ordered set $(\bigcup_{i=0}^n \mathcal{N}_i, \subseteq)$. From an algorithmic point of view, $\mathfrak{T}$ is built in parallel to the progressive collapse from $\mathfrak{G}_0$ to $\mathfrak{G}_n$; in other words, $\mathfrak{T}$ stores the node fusion history. More precisely, we define a sequence $(\mathfrak{T}_i)_{i=0}^n$ as follows. We set $\mathfrak{T}_0 = (\mathcal{N}_0, \emptyset) = (\mathcal{L}, \emptyset)$. Then, for each i from 1 to n, we build $\mathfrak{T}_i$ from $\mathfrak{T}_{i-1}$ by adding the new node $N_{i-1} \cup N'_{i-1}$, and the two edges $(N_{i-1} \cup N'_{i-1}, N_{i-1})$ and $(N_{i-1} \cup N'_{i-1}, N'_{i-1})$. The BPT $\mathfrak{T}$ is finally defined as $\mathfrak{T}_n$.

Remark The classical – image and application-oriented – description of the BPT construction algorithm considers as input: the image I (*i.e.*, the geometrical embedding of Ω, and the value associated to each point of Ω); a region model, that allows us to "describe" the nodes; and a merging criterion, that allows us to quantify the homogeneousness of nodes before and after a putative fusion. These information are important from

an applicative point of view. However, from an algorithmic point of view, their only use is to define a valuation on the edges that allows us to *choose* which nodes to fuse at any given step. In the sequel, we will then consider – without loss of correctness – that a BPT is fully defined by only two input information: (1) the graph $\mathfrak{G}_{\mathcal{L}} = (\mathcal{L}, A_{\mathcal{L}})$ that models the initial partition of the image; (2) a valuation function $W : (2^{\Omega})^2 \times V^{\Omega} \to \mathbb{R}$ that allows us to choose, at each step of the process, the next pair of nodes to be merged.

Data-structures The above description of the BPT construction algorithm implies to define and update, during the whole process, several data-structures, namely: the graph $\mathfrak{G}$, that allows us to know what nodes remain to be merged and what are their adjacency links; and the tree $\mathfrak{T}$ that is progressively built. In order to efficiently compute the valuation W, it is also important to associate each node of $\mathfrak{G}$ to the corresponding part of the image I, *e.g.*, via a mapping between $\mathfrak{G}$ (actually, $\mathcal{N}$) and Ω.

The last – but not least – required data-structure is a sorted list $\mathcal{W}$ that gathers the valuations of each remaining edge of $\mathfrak{G}$. This list contains the information that will authorise, at each of the n iterative steps of the process, to choose the couple of nodes to be merged. This choice is made in constant time $O(1)$, since $\mathcal{W}$ is sorted. After the merging operation, $\mathcal{W}$ has to be updated: (1) to remove the edge between the two nodes; (2) to update the edges affected by the merging operation; and (3) to re-order these updated edges. Operation (1) is carried out in constant time $O(1)$. Operation (2) is carried out in $O(\alpha.T_W)$, where T_W is the cost of the computation of W for an edge, and α is the number of neighbours of the merged nodes (α is generally bounded by a low constant value). Operation (3) is carried out in $O(\alpha.\log_2|\mathcal{W}|)$.

3.2 Generalizing the BPT Construction to Several Images

Let us now consider $k > 1$ images I_j, instead of one. Assuming that they correspond to a same scene, we consider – up to resampling – that all are defined on the same support. We then have a set of images $\{I_j : \Omega \to V_j\}_{j=1}^{k}$. The purpose is now to build a BPT from these k images, by generalizing the algorithmic framework described in Section 3.1. A step of this algorithmic process is illustrated in Figure 1.

Structural evolutions As stated above, we first need a graph that models the initial partition $\mathcal{L}$ of the image(s). Since all the I_j share the same support Ω, such a graph can still be obtained easily, either by subdividing Ω into one-point singleton sets – the induced graph $\mathfrak{G}_{\mathcal{L}}$ is then isomorphic to (Ω, A_{Ω}) – or by considering flat zones, *e.g.*, maximal connected sets of constant value with respect to the Cartesian space $\Pi_{j=1}^{k} V_j$.

The k images I_j share the same support, but they take their values in different sets V_j. As a consequence, following the standard BPT construction paradigm, each of them is associated with a specific valuation function $W_j : (2^{\Omega})^2 \times V_j^{\Omega} \to \mathbb{R}$ that is defined in particular with respect to the value set V_j.

From a data-structure point of view, the generalized BPT construction algorithm will still handle one graph $\mathfrak{G}$, that will be progressively collapsed; and one tree $\mathfrak{T}$ that will be built to finally provide the BPT. A unique mapping between $\mathcal{N}$ and Ω will still allow to have access to the values of a node for the k images. The main difference now

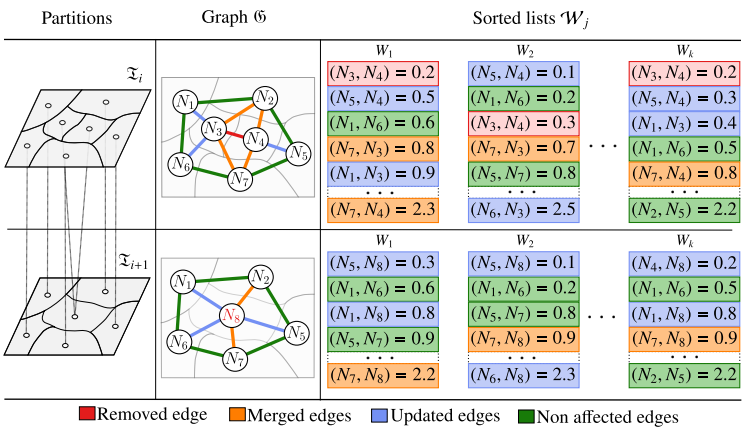


Fig. 1. One step of the building of a BPT from k images. Left: the partition of Ω before and after the fusion of two nodes. Center: the associated graph $\mathfrak{G}$, before and after the fusion of N_3 and N_4, forming the new node N_8. The red edge is removed. The blue and orange edges are updated, *e.g.*, (N_1, N_3) becomes (N_1, N_8); the orange are merged by pairs, *e.g.*, (N_7, N_3) and (N_7, N_4) become (N_7, N_8). The green edges are not affected. Right: the k lists $\mathcal{W}_j$ each corresponding to an image. The red cells are removed, as the edge (N_3, N_4) is suppressed; this edge had been chosen according to a given consensus policy, due to its "high" position in the k lists. The scores of blue and orange cells are updated with respect to N_8; the orange cells are merged by pairs. The positions of the blue and orange cells are updated with respect to their new scores. The scores of the green cells are not affected.

lies into the fact that each function W_j induces – in first approximation – a specific sorted list $\mathcal{W}_j$ to gather the valuation of the remaining edges of $\mathfrak{G}$, with respect to W_j.

Algorithmic consequences From an algorithmic point of view, each iteration of the construction process preserves the same structure. An edge is chosen and the two incident nodes of the graph are merged. This operation leads to update the nodes and edges of $\mathfrak{G}$, and adds a new node plus two edges in $\mathfrak{T}$. The main differences are that: (*i*) several sorted lists then have to be updated instead of one; and (*ii*) the choice of the optimal edge has to be made with respect to the information carried by these k sorted lists instead of only one, for a standard BPT.

From a computational point of view, choosing the edge to remove is no longer a constant time operation, but will depend on the way information are used and compared. Afterwards, operations (1–3) described in the standard BPT construction algorithm, for the sorted list maintenance, have to be duplicated for each list. These operations are then carried out in $O(k)$, $O(k.\alpha.T_{W_\star})$ and $O(k.\alpha.\log_2|\mathcal{W}_\star|)$, respectively.

However, this initial generalization of the BPT construction algorithm can be refined by studying more precisely the policies that are considered to choose an edge, with respect to the information carried by the W_j valuation functions and / or the $\mathcal{W}_j$ sorted lists.

3.3 Consensus Strategies

At each iteration, the choice of the optimal edge to remove, leading to the fusion of its two incident nodes, depends on a consensus between the information of the k images. For each image I_j, useful information are carried, on the one hand, by the valuation function $W_j : (2^\Omega)^2 \times V^\Omega \to \mathbb{R}$ that gives an *absolute* value to each edge and, on the other hand, by the sorted list $\mathcal{W}_j$, that gives a *relative* information on edges, induced by their ordering with respect to W_j. These information are of distinct natures; we study their relevance according to the kinds of considered consensus policies.

Absolute information consensus Let us consider that the consensus policy consists of choosing the edge of lowest mean valuation among the k images, or the edge of minimal valuation among all images. The first consensus (namely *min of mean*) is defined by a linear formulation: $\arg_{(N,N')\in\mathcal{N}} \min \sum_{j=1}^{k} W_j((N,N'))$, while the second (namely *min of min*) is defined by a non-linear formulation: $\arg_{(N,N')\in\mathcal{N}} \min \min_{j=1}^{k} W_j((N,N'))$. However, in both cases the decision is made by considering the absolute information carried by the edges. In other words, it is sufficient to know the k values of each point of Ω with respect to the images I_j. Then, the k sorted lists $\mathcal{W}_j$ are useless, and a single sorted list $\mathcal{W}$ that contains the information of these – linear or non-linear – formulations is indeed sufficient. The construction of a BPT from k images is then equivalent to that from one image defined as $I : \Omega \to \Pi_{j=1}^{k} V_j$.

Relative local information consensus Let us now consider that the consensus policy consists of choosing the edge that is the most often in first position in the k sorted lists $\mathcal{W}_j$, or the most frequently present in the $r \ll |\mathcal{W}_\star|$ first positions in the k sorted lists $\mathcal{W}_j$. These consensus (namely, *majority vote* and *most frequent*, potentially weighted) policies do not act on the absolute valuations of the edges, but on their relative positions in the lists. In such case, it is then mandatory to maintain k sorted lists. However, the decision process does not require to explicitly access the whole lists, but it can be restricted to the first (or the first r) element(s) of each, leading to a *local* decision process.

Relative global information consensus Let us finally consider that the consensus policy consists of choosing the edge that has the best global ranking among the k sorted lists $\mathcal{W}_j$. Such consensus (*e.g.*, *best average*, or *best median ranking*) policy, also acts on the relative positions of the edges in the lists. By contrast with the above case, the decision process requires to explicitly access to the whole content of all these lists, leading to a *global* decision process of high computation cost. (Such cost may be reduced by maintaining, in favourable cases, a $(k+1)$-th list that summarises the global information, and / or by adopting heuristic strategies that update the lists only after a given number of steps).

Algorithmic and structural consequences The choice of a consensus strategy is strongly application-dependent. As a consequence, it is important to consider a trade-off between the structural and computational cost of the approach versus the benefits in terms of results accuracy. In particular, these costs are summarized in Table 1.

Table 1. Space and time cost of the BPT construction for various families of consensus policies. For the sake of readability, r, α and $T_{W_\star}$, which are practically bounded by low constant values have been omitted here.

Consensus policies	$\#\mathcal{W}_\star$	Edge choice	Edge removal	Edges update	Edges sorting
Absolute information	1	$\Theta(1)$	$\Theta(1)$	$\Theta(1)$	$\Theta(\log_2 \lvert\mathcal{W}_\star\rvert)$
Relative local inf.	k	$\Omega(k)$	$\Theta(k)$	$\Theta(k)$	$\Theta(k.\log_2 \lvert\mathcal{W}_\star\rvert)$
Relative global inf.	k	$\Omega(k.\lvert\mathcal{W}_\star\rvert)$	$\Theta(k)$	$\Theta(k)$	$\Theta(k.\log_2 \lvert\mathcal{W}_\star\rvert)$

4 Experiments

To experiment our framework, two applications have been considered in the context of remote sensing: one-time, one-sensor, several (noisy) images, to assess the ability to retrieve information despite image degradation; and multi-time, one-sensor, one image per date, to assess the ability to capture time-independent and redundant information.

The BPT construction and segmentation approaches were voluntarily chosen as very simple, in order to avoid any bias related to these choices, thus better focusing on the actual structural effects of multi-image BPT versus standard BPT.

At this stage, these experiments have to be considered as toy-examples, since neither quantitative validation nor fine parameter tuning were carried out. Our purpose is mainly to give the intuition of potential uses of such BPTs in the field of remote sensing.

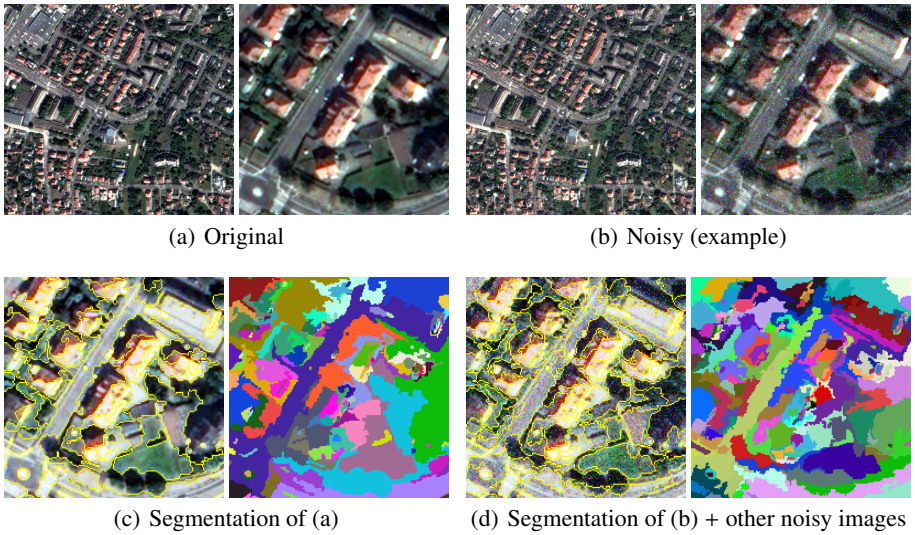
4.1 Urban Noisy Images

Data The dataset used here was sensed over the town of Strasbourg (France). The original sample (Figure 2(a)) is an urban image (1024×1024 pixels) acquired by the Pléiades satellite in 2012 (courtesy LIVE, UMR CNRS 7263). It is a pansharpened image at a spatial resolution of 60 cm with four bands (NIR, R, G, B). From this image, a series of 7 noisy images was generated by adding Gaussian and speckle noise (Figure 2(b)).

Method and results The BPTs are built from the trivial partition $\mathcal{L}$ composed by all singleton sets, *i.e.*, one pixel per region. The valuation function $W_\star : (2^\Omega)^2 \times V^\Omega \to \mathbb{R}$ is defined as the increase of the ranges of the intensity values (for each radiometric band), potentially induced by the fusion of the incident regions. In the case of multi-images, the relative local information consensus policy *most-frequent*, weighted according to the position of the edges within the lists is applied for the first 10% of the lists $\mathcal{W}_\star$.

The non-noisy image of Figure 2(a) is first segmented by extracting a user-defined horizontal cut from its "standard" BPT (Figure 2(c)). According to a same number of regions, a cut is then selected from the multi-images BPT leading to a comparable segmentation result (Figure 2(d)).

From these figures, we observe that the results obtained from noisy images are slightly degraded, but of comparable quality, with respect to Figure 2(c). This tends to confirm the ability of the multi-image BPT-based segmentation to generate accurate results, by discriminating relevant information from noise thanks to the consensus operated between the various images, even in the case of low signal-to-noise ratio.



(a) Original (b) Noisy (example)

(c) Segmentation of (a) (d) Segmentation of (b) + other noisy images

Fig. 2. (a) Initial image, PLÉIADES, 2012; and a zoomed sample, 200×200 pixels. (b) An example of noisy image generated from (a), Gaussian ($\sigma = 10\%$) and speckle noise (5%), and a zoomed sample, 200×200 pixels. (c) Segmentation from the mono-image BPT of (a). (d) Segmentation from the multi-image BPT of (b) plus 6 other noisy images.

4.2 Agricultural Image Time Series

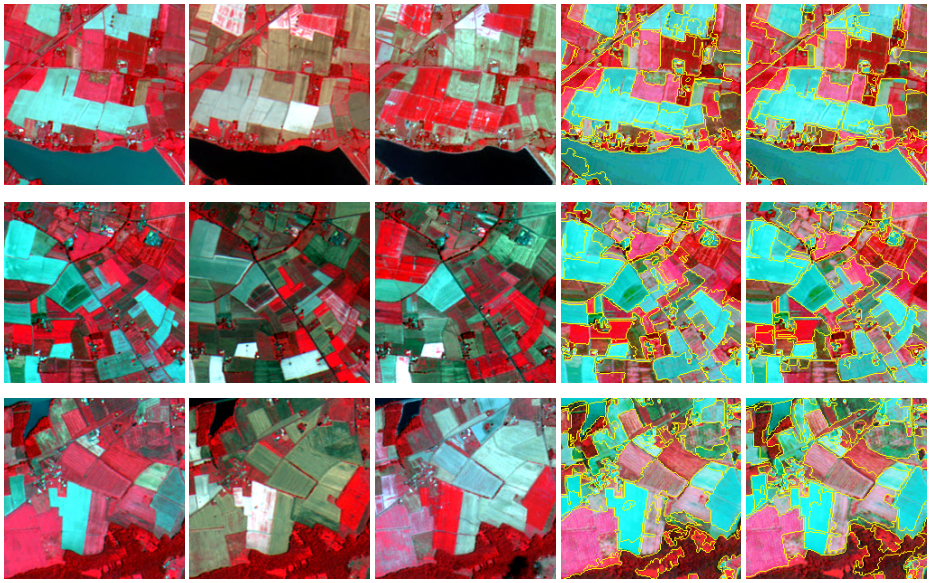
Data The dataset used here is a time series of agricultural images (1000×1000 pixels) of an area located near Toulouse (France). Images were acquired by the FORMOSAT-2 satellite over the 2007 cultural year, see Figure 3 (a–c). They were ortho-rectified and have a spatial resolution of 8 m, with four spectral bands (NIR, R, G, B).

Method and results The BPTs are built from the same partition $\mathcal{L}$ and valuation function $W_\star : (2^\Omega)^2 \times V^\Omega \to \mathbb{R}$ as in Section 4.1. They are also segmented in the same way.

The segmented results, depicted in Figure 3(e), provide regions that are not the same as those obtained from a standard mono-image BPT, computed from one of the images of the series (Figure 3(d)). On the one hand, some segmentation effects deriving from semantic noise in mono-image segmentation are sometimes corrected by the redundant information obtained from multi-images. On the other hand, the multi-image BPT focuses on time-specific details that are only accessible via a temporal analysis, providing a potentially useful tool for such data.

5 Conclusion

This article has presented a data-structure / algorithmic framework and different fusion consensus strategies for building a unique BPT from several images. This contribution



(a) May (b) July (c) August (d) Seg. mono-im. (e) Seg. multi-im.

Fig. 3. (a–c) 3 samples (200 × 200 pixels) of the image time series, FORMOSAT-2, 2007. (d) Segmentation obtained from the mono-image BPT from (a). (e) Segmentation obtained from the multi-image BPT from (a–c).

is, to the best of our knowledge, the first attempt to handle segmentation fusion in the framework of (morphological) hierarchies, in the "*n* images, one algorithm" paradigm.

The experiments carried out on satellite multi-image datasets have shown that the quality of the induced morphological hierarchies are sufficient to further perform improved segmentation, *e.g.*, from noisy or multi-temporal images of a same scene. The consensus strategies considered in this study remain, however, mostly basic (most-frequent, majority vote, *etc.*). Integrating higher-level consensus may then allow us to improve the quality of the hierarchies and the induced segmentation.

In the case of mono-date images, the fusion decisions underlying to the consensus strategies could also be guided by semantic information. Recent advances concerning the hierarchical modelling of such semantic information, in the context of classified remote sensing data [22], may facilitate such approaches.

In the case of multi-date images, we may handle the land-cover evolutions of the observed territories by considering an adequate region model and merging criterion [23]. Such spatio-temporal information could be used to follow local consensus between images leading to hyper-trees where the branches model local temporal fusion decisions.

In a next issue, the results obtained with this method will be fully assessed by quantitative comparisons (using datasets provided by different sensors) and compared to the results produced by other hierarchical and fusion-based segmentation methods.

References

1. Topchy, A., Jain, A.K., Punch, W.: Clustering ensembles: Models of consensus and weak partitions. IEEE TPAMI **27** (2005) 1866–1881
2. Salembier, P., Wilkinson, M.H.F.: Connected operators: A review of region-based morphological image processing techniques. IEEE SPM **26** (2009) 136–157
3. Salembier, P., Garrido, L.: Binary partition tree as an efficient representation for image processing, segmentation, and information retrieval. IEEE TIP **9** (2000) 561–576
4. Rohlfing, T., Maurer Jr., C.R.: Shape-based averaging. IEEE TIP **16** (2007) 153–161
5. Vidal, J., Crespo, J., Maojo, V.: A shape interpolation technique based on inclusion relationships and median sets. IVC **25** (2007) 1530–1542
6. Franek, L., Duarte Abdala, D., Vega-Pons, S., Jiang, X.: Image segmentation fusion using general ensemble clustering methods. In: ACCV. (2010) 373–384
7. Mignotte, M.: Segmentation by fusion of histogram-based K-means clusters in different color spaces. IEEE TIP **17** (2008) 780–787
8. Calderero, F., Eugenio, F., Marcello, J., Marqués, F.: Multispectral cooperative partition sequence fusion for joint classification and hierarchical segmentation. IEEE GRSL **9** (2012) 1012–1016
9. Wang, H., Zhang, Y., Nie, R., Yang, Y., Peng, B., Li, T.: Bayesian image segmentation fusion. KBS **71** (2014) 162–168
10. Chu, C.C., Aggarwal, J.K.: The integration of image segmentation maps using region and edge information. IEEE TPAMI **15** (1993) 72–89
11. Cho, K., Meer, P.: Image segmentation from consensus information. CVIU **68** (1997) 72–89
12. Angulo, J., Jeulin, D.: Stochastic watershed segmentation. In: ISMM. (2007) 265–276
13. Bernard, K., Tarabalka, Y., Angulo, J., Chanussot, J., Benediktsson, J.A.: Spectral-spatial classification of hyperspectral data based on a stochastic minimum spanning forest approach. IEEE TIP **21** (2012) 2008–2021
14. Wattuya, P., Rothaus, K., Praßni, J.S., Jiang, X.: A random walker based approach to combining multiple segmentations. In: ICPR. (2008) 1–4
15. Salembier, P., Oliveras, A., Garrido, L.: Antiextensive connected operators for image and sequence processing. IEEE TIP **7** (1998) 555–570
16. Monasse, P., Guichard, F.: Scale-space from a level lines tree. JVCIR **11** (2000) 224–236
17. Soille, P.: Constrained connectivity for hierarchical image decomposition and simplification. IEEE TPAMI **30** (2008) 1132–1145
18. Vilaplana, V., Marques, F., Salembier, P.: Binary partition trees for object detection. IEEE TIP **17** (2008) 2201–2216
19. Benediktsson, J.A., Bruzzone, L., Chanussot, J., Dalla Mura, M., Salembier, P., Valero, S.: Hierarchical analysis of remote sensing data: Morphological attribute profiles and binary partition trees. In: ISMM. (2011) 306–319
20. Kurtz, C., Passat, N., Gançarski, P., Puissant, A.: Extraction of complex patterns from multiresolution remote sensing images: A hierarchical top-down methodology. PR **45** (2012) 685–706
21. Akcay, H.G., Aksoy, S.: Automatic detection of geospatial objects using multiple hierarchical segmentations. IEEE TGRS **46** (2008) 2097–2111
22. Kurtz, C., Naegel, B., Passat, N.: Connected filtering based on multivalued component-trees. IEEE TIP **23** (2014) 5152–5164
23. Alonso-González, A., López-Martínez, C., Salembier, P.: PolSAR time series processing with binary partition trees. IEEE TGRS **52** (2014) 3553–3567