Binary partition tree construction from multiple features for image segmentation

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

Binary Partition Tree Construction from Multiple Features for Image Segmentation[☆]

Jimmy Francky Randrianasoa[a,b,*], Camille Kurtz[c], Éric Desjardin[b], Nicolas Passat[b]

[a] *Université de Strasbourg, CNRS, ICube, France*
[b] *Université de Reims Champagne-Ardenne, CReSTIC, France*
[c] *Université Paris Descartes, LIPADE, France*

Abstract

In the context of image analysis, the Binary Partition Tree (BPT) is a classical data structure for the hierarchical modeling of images at different scales. BPTs belong both to the families of graph-based models and morphological hierarchies. They constitute an efficient way to define sets of nested partitions of image support, that further provide knowledge-guided reduced research spaces for optimization-based segmentation procedures. Basically, a BPT is built in a mono-feature way, i.e. for one given image, and one given metric, by merging pairs of connected image regions that are similar in the induced feature space. Our goal is to design a new family of BPTs, dealing with the need to directly manage multiple features within its building process. Then, we propose a generalization of the BPT construction framework, allowing one to embed multiple features. The cornerstone of our approach relies on a collaborative strategy used to establish a consensus between different metrics, thus enabling to obtain a unified hierarchical segmentation space. In particular, this provides alternatives to the complex issue of metric construction from several —possibly non-comparable— features. To reach that goal, we first revisit the BPT construction algorithm to describe it in a graph-based formalism. Then, we present the structural and algorithmic evolutions and impacts when embedding multiple features in BPT construction. Final experiments illustrate how this multi-feature framework can be used to build BPTs from multiple metrics computed through the (potentially multiple) image content(s).

Keywords: binary partition tree, morphological hierarchies, multiple features, graph-based image processing, image segmentation

1. Introduction

1.1. Context

In image processing and analysis, segmentation is a crucial task. The concept of segmentation is also quite generic from various points of views: in terms of

[☆] This work was supported by the French *Agence Nationale de la Recherche* under Grants ANR-12-MONU-0001, ANR-15-CE23-0009 and ANR-17-CE23-0015.

[*] Corresponding author

Email address: `randrianasoa@unistra.fr` (Jimmy Francky Randrianasoa)

semantics (from low-level definition of homogeneous areas to high-level extraction of specific objects), in terms of definition (object vs. background or total partition of the image support), and in terms of algorithmics (region-based or contour-based approaches).

In this context, morphological hierarchies propose a wide range of data structures for modeling images at various scales, allowing for the definition of connected operators [1]. Mainly based on the theoretical frameworks of graphs and mathematical morphology [2] [3, Chapters 3, 7, 9], these approaches have already proved their efficiency in many imaging applications. (The algorithms to build and handle them are generally of linear or quasi-linear time and space complexity.) Their very principle is to embed images in a dual spatial / spectral representation space, composed of shapes (i.e. spectrally homogeneous and spatially coherent regions) together with their spatial (neighbouring) and hierarchical (inclusion) relations. These representations offer a structured space to find the best regions / scales according to the applicative objective using, e.g. high-level features to describe the image regions and their content.

Among these representations, the Binary Partition Tree (BPT) [4] is a hierarchical representation of an image modeled as a tree structure, where each node is a connected region. Each of these nodes is either a leaf —an elementary region— or models the union of the regions of its two children nodes. The root is the node corresponding to the entire 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. This is done by iteratively choosing and merging two adjacent regions which minimize a merging criterion, based on a given metric, computed between them. The BPT structure allows users to explore the image at different scales. It can be used for various tasks such as segmentation, image retrieval, object recognition and visual browsing.

As other hierarchical structures, the BPT was mainly designed to process one image at a time. Furthermore, in contrast with most of them (e.g. component-trees, trees of shapes) that are intrinsically defined from the image content, the BPT is also designed to embed an extrinsic metric that is used, together with the image, to build a mixed image / knowledge model. In other words, a BPT is generally built for one image and one metric.

1.2. Motivations and Contributions

The BPT is already known as a relevant data structure for the design of image processing and analysis tools, e.g. for video analysis, remote sensing or medical imaging. Until now, the way to build this data structure has remained mostly limited to a *one image, one metric* paradigm.

Indeed, on the one hand, the metric —i.e. the merging criterion for successive node merging— is a scalar function. When several kinds of information (e.g. colorimetric and geometric heterogeneity) are relevant, it is then required to find a way of fusing them as a unique metric. This complex task has a strong influence on the data structure construction. Thus, it has to be carefully carried out by the expert-user before the very construction process.

On the other hand, the BPT construction deals with one input image. This means that the handling of several images generally has to be dealt with by artificially creating a "super-image", or by fusing beforehand multiple information from various spectral bands into a single metric.

Our purpose is to provide an algorithmic process for BPT construction that goes beyond these current —image and metric— limitations, leading to a notion of multi-feature BPT (MBPT, for brief).

The main idea of our approach is to consider that k images, coupled with l metrics, lead to $n = k.l$ features. The cornerstone of the construction of a multi-feature BPT then consists of providing a way to involve all these features into the construction process, i.e. to make them interact for the definition of a relevant node merging order, and to obtain as output a unified hierarchical segmentation space.

In other words, our main contribution is the proposal of a generalized algorithmic framework for (M)BPT construction, for handling *many metrics and / or many images*. This algorithmic contribution extends the standard BPT construction (retrieved by setting $k = 1$ and $l = 1$). It also encompasses a wide range of cases, e.g. various metrics on one image ($k = 1$ and $l > 1$), one metric on many images ($k > 1$ and $l = 1$), or any combination of metrics on many images. Although being algorithmic, it is important to notice that our contribution is not directly related to image processing. We aim at providing an extended way to build a data structure that describes a hierarchy of partitions of an image. In other words, this provides a reduced, knowledge-guided research space, that can be further involved in various kinds of image analysis procedures: segmentation, classification, simplification, browsing... Such procedures fall, however, out of the scope of this article, and indeed constitute application cases of our proposal.

To reach our goal, some secondary contributions are proposed:

- First, we revisit the standard algorithmic process of BPT construction. In particular, we split its fundamental graph-based expression (namely a graph-collapsing procedure) from its knowledge-based layers (image topology, metric definition, merging policies, etc.) This preliminary analysis is developed in Section 3.

- Second, we describe how the basic BPT construction framework can be generalized to handle multiple features. To this end, we identify the data structure requirements (Section 4.1), and their algorithmic side effects (Section 4.2). This leads to a theoretical algorithmic framework.

- Third, we propose a practical description of this algorithmic framework. Some technical implementation details are provided in Section 5. For the sake of reproducibility, we also provide an open-source library[1] for the creation of MBPTs, which constitutes a technological contribution of this work. A complexity analysis (Section 5.2) describes the induced space and time cost increases of MBPT vs. BPT construction.

This work is concluded by application examples in the domain of satellite image analysis, in Section 6. In particular, we compare the behaviour of standard BPTs to MBPTs handling several images or several metrics.

The remainder of this article —which is an extended and improved version of the conference paper [5]— is composed of a synthetic state of the art of

[1] `https://bitbucket.org/agat-team/agat-v0.3`

graph-based, hierarchical and multi-image segmentation, in Section 2; and a conclusion that emphasizes the perspectives of this work, in Section 7.

2. Related works

2.1. Graph-based and Hierarchical Image Segmentation

Image analysis problems, and in particular segmentation, are often considered in a discrete way via some concepts of graph theory. Practically, image points (i.e. pixels, voxels) are considered as the vertices of a graph, whereas the spatial / neighbouring relations between them are modeled by graph edges. This paradigm, democratized since the early 1970's [6], led to the development of a wide range of segmentation approaches, based on basic graph manipulations.

In this context, image segmentation could be viewed as a partial (e.g. region growing [7]), or total partitioning problem with monotonic (e.g. watersheds [8]) or non-monotonic transformations (e.g. split and merge [9]). Some of these approaches led to the development of optimization schemes (e.g. graph-cuts [10], random walks [11], power watersheds [12]). In the framework of mathematical morphology, these graph-based approaches gave rise to the notion of connected operators [13].

Graph-based segmentation allows us to obtain one segmentation result from a given image. In order to tackle the ill-posed problem of segmentation, hierarchical approaches were developed to compute families of nested partitions, providing adapted solutions at different scales. These notions of hierarchies take their origin in image models initially dedicated to optimize the access and space cost of the carried information (e.g. octrees [14]). These regular models were progressively shifted toward image / content-guided, irregular hierarchies [15].

From this point on, several hierarchical image models were developed, mainly in the framework of mathematical morphology. The most popular are component-trees [16], trees of shapes [17, 18], hierarchical watersheds [19], hyperconnected component-trees [20], and binary partition trees [4] (see Section 2.2). Since they provide a space of potential segmentations, instead of a single result, these hierarchical models were progressively involved in attribute-based [21] or optimization schemes [22, 23] for segmentation purpose.

Based on these image models, generally designed as trees, further developments were proposed to allow for a better flexibility in image and parameter handling. The case of multiband images was considered, leading to data structures such as component-graphs [24], multivalued component-trees [25] or multivariate trees of shapes [26]. Topological handling was also investigated, by allowing connectivity hierarchies in component-hypertrees [27], or dealing with asymmetric hierarchies [28] allowing for non-directed graphs as image models. The way to embed semantic information as image values was also pioneered via the notion of shaping [29].

2.2. Binary Partition Trees

Most of the hierarchical structures proposed in the literature are models intrinsically deriving from the image signal. For instance, component-trees represent the inclusion of the successive level-sets; trees of shapes represent the image level-lines; and hierarchical watersheds rely on saliency measures similar to gradients. These models strongly rely on a functional, often derived from the image intensity. Such regions may not correspond to objects of interest in image content, in particular when dealing with complex images.

The binary partition tree (BPT) [4] relies on a mixed image / knowledge model. It can then easily embed expert-defined knowledge, in addition to image-based information. From a structural point of view, BPTs present similar properties with binary space partition trees [30], designed to efficiently model an image space, mainly in computer graphics. Indeed, BPTs provide hierarchies of nested total partitions of an image.

From an algorithmic point of view, a BPT is built by progressively merging elementary image segments (e.g. flat zones or superpixels), based on a given metric. In particular, the choice of this metric is crucial, as it has a direct impact on the adequacy of the built BPT with the targeted application. This gave rise to several works, ranging from theoretical contributions [31] to experimental assessments [32]. The basic criteria used in most of segmentation approaches are generally radiometric or geometric region similarities (or their fusion into a single metric).

Except for a few contributions in the field of object recognition [33, 34, 35], BPTs were mainly used for segmentation / classification tasks, where such total partitions make sense from a semantic point of view. More precisely, the wider application field of BPTs is remote sensing [36]. In this context, BPTs were involved in multiresolution / multiscale image segmentation and classification [37, 38]; for coupled optical / LIDAR data analysis [39]; hyperspectral images [40, 41]; polarimetric SAR [42, 43]; mixed SAR / hyperspectral images [44]; or multi-temporal SAR image analysis [45].

2.3. Discussion – Contributions

Binary partition trees are built from one given metric. This paradigm is particularly well-fitted when the prior semantic information required for the construction of a mixed image / knowledge image model is indeed unique. (This is for instance the case when we are interested by patterns in satellite images that present a specific colorimetric property, such as agricultural crops with the vegetation index like the NDVI.)

By contrast, when several kinds of prior semantic information are potentially useful for the construction of the BPT, one has to find a way to relevantly embed them into a unique metric. This may be a complex task, in particular when such information are not easily comparable. Until now, the most popular solution for dealing with that case consisted of defining a linear combination of various metrics, leading to a unified one. (This was, for instance, the case in [37] where a linear combination was proposed for mixing spectral homogeneity and compactness criteria, via an evolving trade-off parameter.)

In order to propose an alternative solution to the issue of handling simultaneously many metrics, we consider a new way of building BPTs. Our approach is built on the standard "mono-metric" BPT construction; in particular, it generalizes it.

We do not intend to regroup different criteria into a unique metric. By contrast, we consider each metric individually. Our purpose is then to involve consensual strategies (classically used in machine learning applications) dedicated to manage the different and possibly antagonistic information carried by each metric. By side effect, this consensual paradigm also allows us to encompass the case of processing several images / modalities of a same scene, with similar or specific metrics in each.

In this generalized framework, for a given set of chosen metrics and images, the user's task does no longer consist of defining a complex combination of metrics but, more generally, of choosing a consensus policy, that is the way the different features will define a common result. This paradigm provides a wide range of possibilities —including those previously offered by standard BPTs— and may then improve the adequacy of the constructed BPTs with the user's needs for a particular application.

However, the construction process of such multi-feature BPTs is more complex than for standard BPTs. In particular, the intermediate data structures involved in the construction algorithmics must now allow for the representation and interactions of various features. This leads to more complex data structures, and a higher computational cost.

In the sequel, we first analyze the intrinsic properties of the standard BPT construction algorithm, in order to emphasize the parts impacted by the generalization to the multi-feature case. Then, we propose a theoretical algorithmic framework for MBPT construction, that generalizes the standard BPT algorithm. Finally, we propose a practical description and an implementation of this MBPT construction algorithm.

3. Structural Description of the BPT Construction

3.1. Definitions and Notations

This section provides formal definitions and notations mandatory to make this work self-contained.

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.

To model the fact that two points x and y of Ω are neighbours, let A_Ω be an adjacency (i.e. irreflexive, symmetric) binary relation on Ω. In other words, $\mathfrak{G}_\Omega = (\Omega, A_\Omega)$ is a graph that models the structure of the image space.

For any partition $\mathcal{P}$ of Ω, we define an adjacency inherited from that of Ω. We say that two distinct sets $N_1, N_2 \in \mathcal{P}$ 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.

3.2. The Standard BPT Construction

A BPT is a hierarchical representation of an image. More precisely, it is a binary tree whose each node is a connected region. A node can be either a leaf representing an "elementary" region, or the union of two neighbouring regions. The root node corresponds to the whole image support.

We recall the classical way of building a standard BPT, as proposed in the pioneering article [4]. The BPT construction is a bottom-up process. It proceeds from the determination of the leaves —provided by an initial partition of the image— to the root. This is done by choosing and merging iteratively two adjacent regions that minimize a criterion reflecting their likeness. This merging sequence is stored in a hierarchical structure, namely the BPT, that models the image at different scales. This is illustrated in the right part of Figure 1.

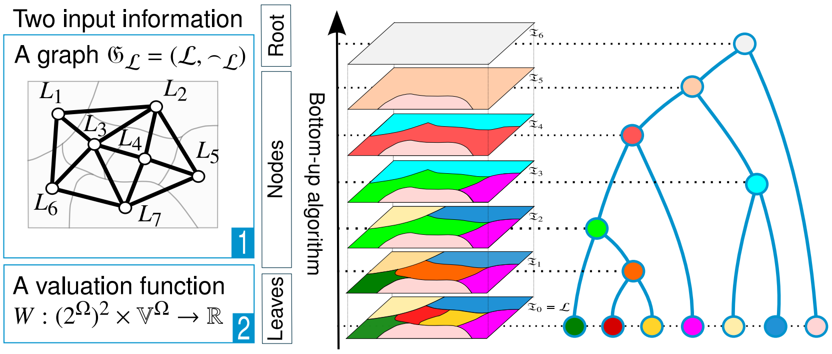


Figure 1: Illustration of the construction of a BPT from one image. Left (input): (1) a graph $\mathfrak{G}_{\mathcal{L}} = (\mathcal{L}, A_{\mathcal{L}})$ models the initial partition of the image; (2) a valuation function $W : (2^{\Omega})^2 \times V^{\Omega} \to \mathbb{R}$ allows us to iteratively choose the pair of nodes to be merged (see Section 3.3). Right: progressive, bottom-up, creation of the tree from $\mathfrak{T}_o$ to $\mathfrak{T}_6$ by iterative fusions of two neighbouring regions.

A huge number of distinct BPTs may be obtained for a given initial partition of Ω. In order to determine a relevant BPT, it is then necessary to decide of the priority of the fusions between nodes. Let $N_i, N_j \in \mathcal{P}$ be two distinct and adjacent regions / nodes. A BPT generation relies on two main notions: a *region model* $M_r(N_i)$ which specifies how a region N_i is characterized (e.g. colour, shape); and a *merging criterion* $O_r(N_i, N_j)$ which defines the similarity of neighbouring regions N_i, N_j and thus the merging order.

A strategy commonly adopted for representing each region is to consider their average colour in a given space (e.g. RGB, HSV), and to merge pairs of adjacent regions either similar one to each other, or similar to the region model of the novel region built from their union. Another strategy [37] considers as region model a linear combination of radiometric and geometric features, with a merging criterion where the trade-off between these features evolves during the construction of the BPT. In any case, the merging criterion is a scalar function, which requires fusing various elements of expert knowledge in a single metric.

3.3. Structural Description: A Graph-based Point of View

In this section, we now revisit the BPT construction process. In particular, we focus on its algorithmic core, that essentially consists of a graph collapsing approach. We then rewrite the BPT construction process in a graph-based formalism, thus emphasizing the involved data structures mandatory to carry out the collapsing process, modeled by the output tree.

3.3.1. Graph and valuation function

The way to describe the construction of a BPT is generally considered from spatial (the way regions are built) and descriptive (the way regions are characterized and how they can be considered similar) points of view. Indeed, the classical description of the BPT construction 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 "describes" the nodes; and a merging criterion, that quantifies the homogeneousness of nodes before and after fusion. These information are important from the point of view of a considered application.

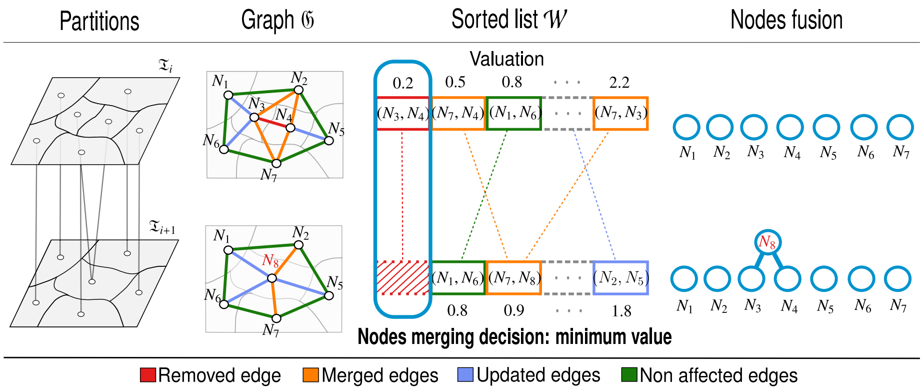


Figure 2: One step of the BPT construction from one image. From the left to the right: First: the partition of Ω before and after the fusion of two nodes. Second: the associated graph $\mathfrak{G}$, before and after the fusion of N_3 and N_4, forming the new node N_8. The red edge is removed. The blue and orange edges are updated, e.g. (N_1, N_3) becomes (N_1, N_8). The orange edges 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. Third: the sorted list that gathers the scalar valuations of each remaining edge of $\mathfrak{G}$. The red cells are removed, as the edge (N_3, N_4) is suppressed (this edge had been chosen due to its highest position in the list). 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. Last: a new part of the BPT $\mathfrak{T}$ is created by adding the new node N_8, and linking it to its two children nodes N_3 and N_4.

However, beneath these image and knowledge-based notions, the construction of a BPT is intrinsically a graph collapsing process. From an algorithmic point of view, the region models and merging criteria 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 generality, that a BPT is fully defined by only two input information (see left part of Figure 1):

1. a graph $\mathfrak{G}_\mathcal{L} = (\mathcal{L}, A_\mathcal{L})$ that models the initial partition of the image;
2. a valuation function $W : (2^\Omega)^2 \times V^\Omega \to \mathbb{R}$ for choosing the pair of nodes to be merged.

3.3.2. Structural description of the algorithm

We now consider an initial partition $\mathcal{L}$ of Ω. Each node $L \subseteq \Omega$ of $\mathcal{L}$ is assumed to be connected with respect to A_Ω. This partition $\mathcal{L}$ defines the set of the BPT leaves we are going to build. For instance, $\mathcal{L}$ can be the set of the image flat zones. It is equipped by the adjacency $A_\mathcal{L}$ inherited from A_Ω. This leads to a graph $\mathfrak{G}_\mathcal{L} = (\mathcal{L}, A_\mathcal{L})$ modeling the structure of the partition of 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 the valuation function W. 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 one.

From a structural side, the BPT $\mathfrak{T}$ is the Hasse diagram of the partially ordered set $(\bigcup_{i=0}^{n} \mathcal{N}_i, \subseteq)$. From an algorithmic side, $\mathfrak{T}$ is built in parallel to the progressive collapse from $\mathfrak{G}_0$ to $\mathfrak{G}_n$. In particular, $\mathfrak{T}$ 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 then defined as $\mathfrak{T}_n$.

Remark. In [46], a graph-based definition of BPT construction is also proposed, that relies on a minimum spanning tree paradigm. However, this formalization is valid only if the merging order is associated to a valuation of the edges that is fixed *a priori* on the initial partition. This is not the case, in general.

3.3.3. Data structures

The above description of the BPT construction algorithm implies to define and update the following data structures:

- the graph $\mathfrak{G}$ which stores nodes that remain to be merged, and 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 $\mathfrak{G}$ and Ω.

The last required data structure is a sorted list $\mathcal{W}$ that gathers the scalar valuations of each remaining edge of $\mathfrak{G}$. This list contains the information for choosing the couple of nodes to be merged relatively to a given metric. One iteration of this algorithm is illustrated in Figure 2.

This choice is made in constant time $\mathcal{O}(1)$, 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 $\mathcal{O}(1)$. Operation (2) is carried out in $\mathcal{O}(\alpha.T_W)$, 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 $\mathcal{O}(\alpha. \log_2 |\mathcal{W}|)$.

4. Multi-feature Generalization of the BPT Construction

In this section, we now describe our multi-feature generalization of the BPT construction algorithm. We propose a description of the algorithm and associated data structures in the same graph-based formalism used in the previous section for describing the standard BPT construction. In particular, we still consider that this construction —viewed as a graph collapsing problem— takes as input the graph $\mathfrak{G}_{\mathcal{L}} = (\mathcal{L}, A_{\mathcal{L}})$. However, we now have to use *several* valuation functions $W_\star : (2^\Omega)^2 \times V^\Omega \to \mathbb{R}$. We then discuss the structural and algorithmic side effects of using several such valuation functions "at the same time", instead of only one.

Our purpose is still to build *one* BPT from these input information. Introducing several valuation functions allows us to embed several features in a more independent way in the construction process. These features can represent several metrics associated to a same image; a same metric associated to several images of a same scene; or various metrics on various images of a same scene.

4.1. Structural Evolutions

4.1.1. What does not change...

The proposed generalization deals with the "feature" part of the construction. As stated in Section 3.3.1, we need a graph that models the initial partition $\mathcal{L}$ of the image(s). Our purpose remains to collapse this unique graph associated to a unique spatial scene. Practically, it implies that the (potentially multiple) images involved in the BPT construction are defined in the same spatial reference, i.e. the same support Ω. This spatial coherence assumption is generally a standard requirement in image processing (e.g. in medical imaging, via registration; in remote sensing via georeferencing; etc.). A graph $\mathfrak{G}_{\mathcal{L}}$, which is isomorphic to (Ω, A_Ω), can be obtained easily, either by subdividing Ω into one-point singleton sets or by considering flat zones or superpixels.

Then, the "graph" part of the BPT construction process remains unchanged. In terms of data structures, the generalized BPT construction will still handle one graph $\mathfrak{G}$, that will be progressively collapsed; and one tree $\mathfrak{T}$ that will be built to provide the BPT. A unique mapping between $\mathcal{N}$ and Ω will still allow us to access to the values of a node for the different images.

4.1.2. ... and what does

Let us now consider the "feature" part of the data structure. In the initial BPT construction approach, the valuation function $W : (2^\Omega)^2 \times V^\Omega \to \mathbb{R}$ was explicitly modeled by a sorted list $\mathcal{W}$ of the values of all graph edges. This list was updated during the progressive collapsing of $\mathfrak{G}$, by removing elements from the list; updating the values of some edges; and re-sorting edges with respect to their updated values.

We now consider $n \geq 1$ valuation functions $W_\star : (2^\Omega)^2 \times V^\Omega \to \mathbb{R}$, which means that each edge is associated to n values, one for each function. By considering k distinct images and l distinct metrics, we may have up to $n = k.l$ such valuation functions. This leads to define no longer one, but n sorted lists $\mathcal{W}_i$ $(1 \leq i \leq n)$. Each list $\mathcal{W}_i$ is associated with a specific valuation function $W_i : (2^\Omega)^2 \times V_j^\Omega \to \mathbb{R}$, defined with respect to a value set V_i (see Figure 3). The handling of these sorted lists remains the same in terms of removal, value updating / resorting, as for one list.

4.2. Algorithmic Evolutions

Our purpose is now to build a BPT from these n lists by generalizing the algorithm described in Section 3.2, which initially depended on a unique list $\mathcal{W}$.

From an algorithmic point of view, each iteration of the construction preserves the same organization. An edge is chosen and the two incident nodes of the graph are merged. This operation updates the nodes and edges of $\mathfrak{G}$, and adds a new node plus two edges in $\mathfrak{T}$. The main differences are now that:

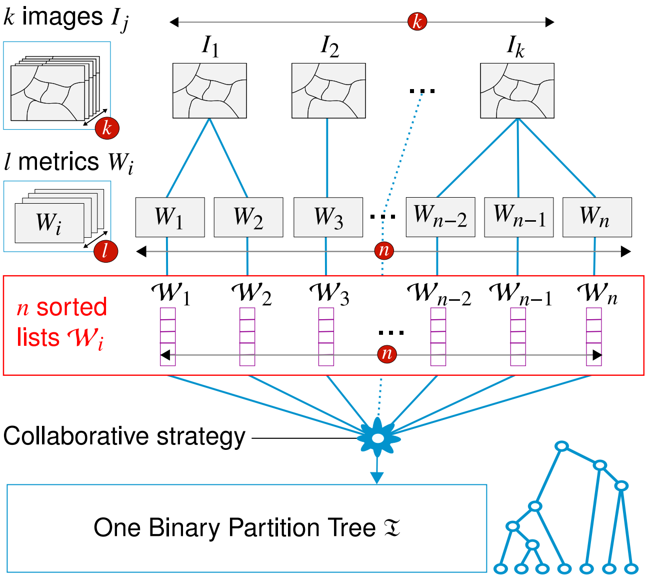


Figure 3: General structure of a BPT construction involving n sorted lists. Each list $\mathcal{W}_i$ gathers information about edges from one valuation function associated to a given metric on a given image. A BPT is obtained by applying a collaborative strategy between the information carried by these n lists (see Section 4.2).

1. $n \geq 1$ sorted lists are updated; and
2. the choice of the optimal edge is made with respect to the information carried by these n sorted lists instead of only one.

At each iteration of the algorithm (see Figure 4), the choice of the optimal edge to remove, leading to the node fusion, depends on a decision. This decision is a consensus, made with respect to the information provided by these n lists. On the one hand, information are carried by the lists $\mathcal{W}_i$, that give *relative* information on edges, induced by their ordering with respect to W_i. On the other hand, information are carried by the valuation functions $W_i : (2^\Omega)^2 \times V_j^\Omega \to \mathbb{R}$ that give an *absolute* value to each edge. These information are of distinct natures. We study their relevance according to various kinds of consensus policies. In particular, we identify three families of consensus strategies.

4.2.1. Absolute information consensus

Let us consider, for instance, that the consensus policy consists of choosing the edge of lowest mean valuation among the n lists $\mathcal{W}_i$, or the edge of minimal valuation among all lists. The first consensus (namely *min of mean*) is defined by a linear formulation: $\arg_{(N,N')\in\mathcal{N}} \min \sum_{i=1}^{n} W_i((N,N'))$. The second (namely *min of min*) is defined by a non-linear formulation: $\arg_{(N,N')\in\mathcal{N}} \min \min_{i=1}^{n} W_i((N,N'))$. However, in both cases, the decision is made by considering *absolute* information carried by the edges. More generally, when the information carried by the values of each edge is a sufficient knowledge, n sorted lists $\mathcal{W}_i$ are not necessary. In such a case, a single sorted list $\mathcal{W}$ that contains the information of these, linear or non-linear, formulations is

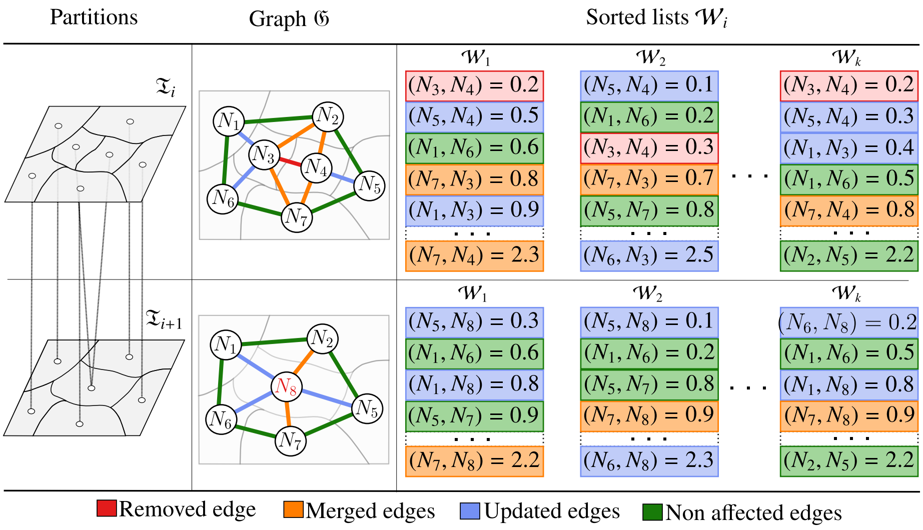


Figure 4: One step of a BPT construction involving n lists $\mathcal{W}_i$. Left: the partition of Ω before and after the fusion of two nodes. Center: the associated graph $\mathfrak{G}$, before and after the fusion of N_3 and N_4, forming the new node N_8. The red edge is removed. The blue and orange edges are updated, e.g. (N_1, N_3) becomes (N_1, N_8). The orange edges 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 n lists $\mathcal{W}_i$, each corresponding to a valuation function W_i. The red cells are removed, as the edge (N_3, N_4) is suppressed (this edge had been chosen due to its “optimal” position and / or valuation in the n 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.

indeed sufficient. The BPT construction involving n lists is then equivalent to that from one list.

The main difficulty raised by this policy derives from the potential heterogeneity of the values carried by the different $W_\star$ valuation functions. Indeed, when the features are not of same nature, their values should be normalized and / or weighted to allow for the definition of adequate fusion / comparison operators. This issue is mainly the same that occurs in most optimization schemes where a given metric is built from several terms of varying semantics. Although this remains tractable, at the cost of a certain expertise, it argues in favour of considering the next proposed policies as relevant alternatives.

4.2.2. Relative local information consensus

Let us now consider, for instance, that the consensus policy consists of choosing the edge that is the most often in first position in the n sorted lists $\mathcal{W}_i$, or the most frequently present in the $r \ll |\mathcal{W}_i|$ first positions in the n lists $\mathcal{W}_i$. 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. Another strategy can also consist of choosing the edge with the lowest *mean of ranks*, according to the position of the edges within the lists. In such cases, it is then mandatory to maintain n sorted lists, but the decision process does not require to access the whole lists. It can be restricted to the first (or the first r) element(s) of each, leading to a *local* decision process.

4.2.3. 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 n sorted lists $\mathcal{W}_i$. As previously, such consensus (e.g. *best average* or *best median ranking*) policy, also acts on the relative positions of the edges in the lists, and does not consider the absolute values of the edges. However, in contrast with the above case, the decision process requires to explicitly access the whole content of all these lists, leading to a *global* decision process of higher computational cost.

5. Technical Discussion

In this section, we finally propose a technical description of the MBPT construction algorithm. This description is associated to a Java library implementing this algorithm and that we distribute online[2].

5.1. Details of the proposed algorithm

To build a MBPT, the main structure of the classical BPT construction remains unchanged. Indeed, the MBPT is still built from its leaves to its root, by choosing and merging adjacent regions that minimize a merging criterion. However, instead of dealing with a classical merging criterion on one image and one metric, we may now consider many images, different metrics, and a consensus strategy as input parameters. The core of the algorithm is divided as follows:

1. preparation of the lists $\mathcal{W}_i$ based on each image and metric couple;
2. preparation of the graph $\mathfrak{G}$ modeled as a Region Adjacency Graph (RAG);
3. creation of the MBPT $\mathfrak{T}$ by merging pairs of nodes.

The general structure of the proposed algorithm is presented in Algorithm 1.

5.1.1. Preparation of the lists $\mathcal{W}_i$ and the RAG

As already mentioned in Section 4.1, by considering k distinct images and l distinct metrics, we may have up to $n = k.l$ valuation functions. This leads us to prepare no longer one, but n sorted lists $\mathcal{W}_i$. Each list $\mathcal{W}_i$ contains all the edges of the graph $\mathfrak{G}$ and is maintained ordered during the whole process. An object representing the metric is instantiated by the algorithm and linked to each list $\mathcal{W}_i$. At this stage, these n lists are still empty.

When all the lists $\mathcal{W}_i$ are prepared and linked to the right couple of image and metric, a graph $\mathfrak{G}$ (implemented as a RAG) has to be built. The initial nodes of the graph are created from the pixels or from an initial partition of the image support. They constitute the leaves of the MBPT. Once these leaves are prepared, the edges of the graph are created and their values are computed by invoking particular methods of the corresponding metric. These edges represent the adjacencies between neighbouring regions of the image. For each edge, n computed metric values are associated. In parallel, we use these metric values to fill the n lists $\mathcal{W}_i$ that will be further maintained sorted.

[2] `https://bitbucket.org/agat-team/agat-v0.3`

Algorithm 1: MBPT construction

```
Data: Image $I[\,]$; Metric $W[\,]$; Consensus strategy $Consensus$;
Result: A Multi-feature Binary Partition Tree $MBPT$;
/* Variables                                                              */
Region Adjacency Graph $\mathfrak{G}$;                                    // the RAG
List $\mathcal{W}[\,]$;                                      // the list of lists $\mathcal{W}_i$
Image $preSeg$;                                    // initial partition of the image
/* Initialization                                                         */
prepareLists($I, W$);                // associates metrics, images & adjacency lists
prepareRAG($\mathcal{W}$, $preSeg$);                // builds initial leaves & adjacencies
fillLists($\mathcal{W}, \mathfrak{G}$);               // fills the lists with the initial adjacencies
/* Node fusion                                                            */
while $\mathfrak{G}.tabAdja$ is not empty do
    Adjacency $chosenAdja = Consensus.apply(\mathcal{W})$;
    Node $n1 = chosenAdja.region1$;
    Node $n2 = chosenAdja.region2$;
    Node $newNode$ = fusionOf($n1, n2$);
    $\mathfrak{G}.tabNode.add(newNode)$;
    $\mathfrak{G}.tabAdja.update()$;                   // adds, removes and updates values
    /* Updates the content of the lists                                   */
    for $i = 0; i < \mathcal{W}.size(); i{+}{+}$ do
        $\mathcal{W}[i].update()$;                     // adds, removes and updates values
$MBPT.tabNode = \mathfrak{G}.tabNode$;
return $MBPT$
```

5.1.2. Creation of the multi-feature BPT by merging nodes

Once the lists $\mathcal{W}_i$ and the graph $\mathfrak{G}$ are ready, an iterative fusion of the nodes is operated to build the MBPT. This iterative process stops when the lists $\mathcal{W}_i$ are empty. The root of the tree is then reached. Each iteration is mainly composed of:

1. the choice of the next two adjacent nodes of $\mathfrak{G}$ to be merged, based on the consensus strategy;
2. the handling of the n lists $\mathcal{W}_i$ by removing, in each list, the adjacency linking the two chosen nodes, and updating the values of all adjacencies that link them to their neighbours;
3. the resorting of the lists.

5.2. Complexity Analysis

In the multi-feature paradigm, choosing the edge to remove is no longer a constant time operation. Indeed, it depends 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 $\mathcal{O}(n)$, $\mathcal{O}(n.\alpha.T_{W_\star})$ and $\mathcal{O}(n.\alpha.\log_2|\mathcal{W}_\star|)$, respectively, where T_{W_i} is the cost for computing W_i for a given edge, while α is an upper bound for the nodes degree within the graph $\mathfrak{G}$.

This analysis can be refined with respect to the policies considered for choosing an edge, and the information carried by the valuation functions W_i and / or the sorted lists $\mathcal{W}_i$. Besides, the choice of a consensus strategy is strongly application-dependent. It is then important to consider a trade-off between the structural and computational cost of the approach versus the benefits in terms of results accuracy. These costs are summarized in Table 1.

	# $\mathcal{W}_\star$	Edge choice	Edge removal	Edge update	Edge sorting
Stand. BPT [4]	1	$\mathcal{O}(1)$	$\mathcal{O}(1)$	$\mathcal{O}(1)$	$\mathcal{O}(\log_2 \lvert\mathcal{W}\rvert)$
Absolute inf.	1	$\mathcal{O}(1)$	$\mathcal{O}(1)$	$\mathcal{O}(1)$	$\mathcal{O}(\log_2 \lvert\mathcal{W}\rvert)$
Relative local inf.	n	$\mathcal{O}(n)$	$\mathcal{O}(n)$	$\mathcal{O}(n)$	$\mathcal{O}(n.\log_2 \lvert\mathcal{W}_\star\rvert)$
Relative global inf.	n	$\mathcal{O}(n.\lvert\mathcal{W}_\star\rvert)$	$\mathcal{O}(n)$	$\mathcal{O}(n)$	$\mathcal{O}(n.\log_2 \lvert\mathcal{W}_\star\rvert)$

Table 1: Cost of the BPT construction for various families of consensus. For readability purpose, α and $T_{W_\star}$, practically bounded by low constant values, have been omitted.

This table provides the cost of an elementary step of the BPT construction. The number of these steps is equal to the size of the initial graph $\mathfrak{G}_\mathcal{L}$, namely $|\mathcal{L}|$. More precisely, it is equal to the number of vertices in this initial graph (minus one). Indeed, each step merges two of these vertices, until obtaining a graph formed by exactly one node. At each step, at least one edge is removed from the graph. The number of remaining edges is then equal to the size of the list(s) $|\mathcal{W}_i|$. We can assume that the number of edges is bounded by the number of vertices of the graph, up to a multiplicative constant α, generally low for images defined on discrete grids. Based on these assumptions, an upper bound for the overall computational cost of the standard BPT construction [4] is $\sum_{i=1}^{|\mathcal{L}|}(\mathcal{O}(1)+\mathcal{O}(\log_2|\mathcal{W}|)) = \sum_{i=1}^{|\mathcal{L}|}\mathcal{O}(1) + \sum_{i=1}^{|\mathcal{L}|}\mathcal{O}(\log_2(\alpha.i)) = \mathcal{O}(|\mathcal{L}|) + \mathcal{O}(\log_2(|\mathcal{L}|!)) = \mathcal{O}(|\mathcal{L}|\log_2|\mathcal{L}|)$, and so is the cost for the first consensus policy (Section 4.2.1). Following the same kind of calculation, the cost for the second consensus policy (Section 4.2.2) is $\mathcal{O}(n.|\mathcal{L}|\log_2|\mathcal{L}|)$. That of the third (Section 4.2.3) is $\mathcal{O}(n.|\mathcal{L}|^2)$.

6. Experimental studies

6.1. Experimental protocol – Motivation and choices

Multi-feature BPTs do not define, by themselves, a segmentation method. Indeed, MBPTs —such as BPTs— are data structures, that provide a reduced space of partitions of the image support, hierarchically organized with respect to the (partial order) refinement relation. Such (M)BPTs can then further be involved in segmentation methods (but also in other tasks: detection, image simplification, classification, etc.).

Based on these considerations, our purpose is not to prove that MBPTs lead to better results for segmentation tasks compared to other segmentation methods. Indeed, this would require to involve subsequent segmentation processes on MBPTs, whose own quality evaluation could not be easily deconvoluted from that of the very data structure. By contrast, our purpose is to focus on the quality of the data structure itself. This is why we choose to compare the MBPTs with the standard notion of BPT (built from one feature) and with BPTs built from one feature obtained as the linear combination of many.

Other tree-based data structures exist for image modeling, for instance component-trees [16], trees of shapes [17] and hierarchical watersheds [19]. However, these trees are not built from features, and they require information provided beforehand. Indeed, component-trees and trees of shapes require the definition of a total order on the image values, whereas hierarchical watersheds require the definition of a saliency measure (namely, a gradient) on the image. By contrast, the (M)BPTs rely on relative information between nodes, that evolve while the organization and geometry of the nodes changes from the leaves up to the root. For these reasons, related to the different nature of the involved information, a direct comparison between (M)BPTs and such other tree structures would not provide relevant clues with respect to the advantages of a multi-feature paradigm in BPTs.

In the sequel, we then compare MBPTs with various kinds of BPTs. As stated above, this cannot be done by simply observing segmentation results, but by observing more generally the ability of (M)BPTs to potentially lead to good segmentations. Then, we need a specific validation framework dedicated to the evaluation of hierarchical image representations such as trees. Such a framework, recently proposed in [47], is described in Section 6.2; it is used for our experiments.

Since the BPTs have been mainly involved in remote sensing image processing and analysis over the past 20 years, we choose to consider two case studies within this application field. In Section 6.3, we show how a MBPT can be built from a single satellite image, by considering simultaneously various metrics. In Section 6.4, we show how a MBPT can be built from multiple satellite images sensed over the same area.

6.2. BPT quality evaluation

Many works have been dedicated to improve the construction of BPTs. However, only few works were dedicated to quantitatively evaluate the "quality" of these trees, (i.e. their ability to allow for a satisfactory segmentation), and then to determine if they were "correctly" constructed. In the case of supervised evaluation, the segmentation results are generally compared to a ground-truth (GT) map with standard quality indices based on spatial overlapping information. In the case of BPTs, such frameworks can not be easily considered, since several segmentations can be obtained from one single BPT.

We designed in [47] a supervised framework for BPT quality evaluation. Our approach is not to match an ideal partition of the image. We aim to evaluate the ability of a BPT to construct nodes that match at best with a subset of expert-defined segments. Then, we assume that a GT map, composed of reference segments $S_i \subseteq \Omega$ ($i \in [\![1, k]\!]$), is provided by an expert. Such GT segments can be labelled, allowing us to embed semantic criteria in the framework. We recall the three main points of this framework for BPT quality evaluation (see [47] for more details).

6.2.1. Node / segment matching metrics

First, we choose a local metric to evaluate the matching degree between a GT segment S and a node N of the BPT. We define a local score $\Lambda(N, S) \in \mathbb{R}^+$, where Λ is a metric evolving monotonically with respect to this similarity. The most common quality indices are based on a combination of the true positives (TP), false positives (FP) and false negatives (FN) computed between N and S. Examples of Λ functions, considered in this work, are the Jaccard index J' [48]:

$$J'(N, S) = \frac{|N \cap S|}{|N \cup S|} = \frac{TP}{TP + FP + FN} \tag{1}$$

and the Dice coefficient D [49]:

$$D(N,S) = \frac{2.|N \cap S|}{|N| + |S|} = \frac{2.TP}{2.TP + FP + FN} \tag{2}$$

In particular, these two classical quality functions were chosen as they are quite often considered for remote sensing image analysis evaluation, which is coherent with the context of the following two case studies.

6.2.2. Finding matching nodes in the BPT

Then, we determine the nodes within the BPT that maximize matching with the GT segments. Given k GT segments S_i ($i \in [\![1,k]\!]$), and a BPT $\mathfrak{T}$ composed of n nodes N_j ($j \in [\![1,n]\!]$ with $n = 2.|\mathcal{L}| - 1$), we aim to solve k times the optimization problem: $N_\star^i = \arg\max_{j \in [\![1,n]\!]} \Lambda(N_j, S_i)$. A brute-force approach would be to browse, for each GT segment S_i, the whole BPT in a bottom-up way. This would involve useless comparisons. To reduce this cost, we consider two strategies.

We use a spatial constraint to "vertically" restrict the search space within the BPT. In particular, for finding an optimal node $N_\star$ it is sufficient, for a given GT segment S, to restrict the study to the BPT branches whose ending leaves intersect S. We also use a quantitative heuristic to "horizontally" restrict the BPT browsing. Indeed, to correctly match a GT segment S, a node N has to be of comparable size. Practically, it is relevant to restrict the actual computation of the Λ values to nodes N with size within a confidence interval $[\kappa^-, \kappa^+] \subseteq \mathbb{N}$, with $\kappa^- < |S| < \kappa^+$.

At this stage, the scores Λ are computed. The score $\Lambda(N_\star, S)$ of the best node is then set as the quality score $\lambda(S)$ of $\mathfrak{T}$, for each GT segment S.

6.2.3. Global quality score

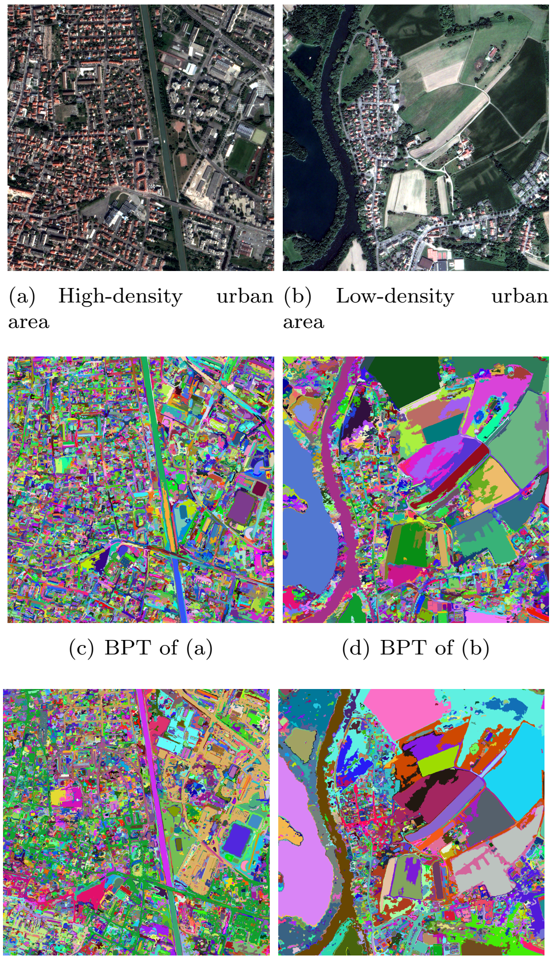
Finally, we define a global quality measure that merges the local scores for the different GT segments. We define a weighted formulation of such global quality score Γ as

$$\Gamma = \sum_{\ell \in L} w_\ell \sum_{S_i \in \mathcal{C}_\ell} w_i.\lambda(S_i) \tag{3}$$

with $\sum_{\ell \in L} w_\ell = 1$, $\sum_{S_i \in C_\ell} w_i = 1$, and $w_\star \geq 0$, where L is the label set and $\mathcal{C}_\ell$ are the different semantic classes of GT segments. The weights w_ℓ can be used to assess the relative importance of each semantic class $\mathcal{C}_\ell$. In particular, $w_\ell = 1/|L|$ if each class has the same importance. The weights w_i can be used to discriminate the importance of each GT segment, i.e. the necessity to correctly segment it.

6.3. Case study 1: Multi-criteria segmentation

The analysis of very high spatial resolution (VHSR) remote sensing images is a challenging task. In this context, the segmentation of satellite images using classical mono-metric BPTs has been widely investigated [36, 37, 38, 40, 41]. This motivates the following experimentation on multi-criteria segmentation procedures on such images.



(a) High-density urban area

(b) Low-density urban area

(c) BPT of (a)

(d) BPT of (b)

(e) MBPT of (a)

(f) MBPT of (b)

Figure 5: (a–b) Two VHSR satellite images (2 000 × 2 000 pixels) at a spatial resolution of 60 cm sensed by the Pléiades satellite and covering different geographical areas. (c–d) Segmentation results from standard BPTs computed from (a) and (b), respectively (23 500 and 5 000 regions, respectively). (e–f) Segmentation results from MBPTs computed from (a) and (b), respectively (23 500 and 5 000 regions, respectively), using 4 features: W_{colour}, W_{elong}, W_{ndvi}, W_{ndwi}.

6.3.1. Data

The dataset used in this study (courtesy LIVE, UMR CNRS 7263) was sensed over the town of Strasbourg (France) by the Pléiades satellite, in 2012. The first sample is a VHSR image (2 000 × 2 000 pixels) representing a complex high-density urban area (Figure 5(a)) composed of different urban objects (e.g. individual houses, industrial buildings, parking lots, roads, shadows, water canals). The second sample is a VHSR image (2 000 × 2 000 pixels) that represents a typical low-density urban area (Figure 5(b)) composed of different geographical objects (e.g. crop fields, forests, bare soils, rivers). Both are multispectral images at a spatial resolution of 60 cm with 4 spectral bands (R, G,

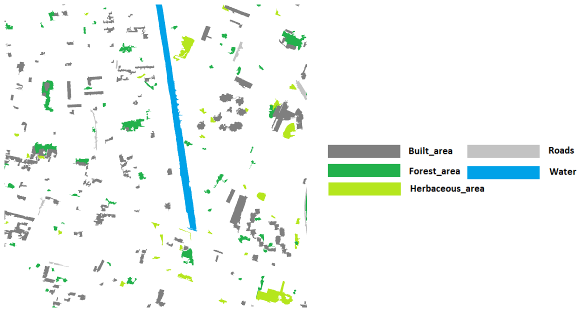


Figure 6: Partial ground-truth map representing urban objects of the VHSR image of Figure 5(a). This map contains 228 segments delivered by a crowd-sourcing campaign in the context of the COCLICO research project. $L = 5$ different semantic classes are available: *built area*, *forest area*, *herbaceous area*, *roads*, *water*.

B, NIR).

Ground-truth maps of different urban objects represented in the scene are also available. These maps were derived from a public crowd-sourcing campaign in the context of the COCLICO research project[3]. We only retained the reference segments that led to the highest consensus between the crowders. The segments are labelled with $L = 5$ different semantic classes: *built area*, *forest area*, *herbaceous area*, *roads*, *water*. Figure 6 presents the partial GT map of the first image (Figure 5(a)).

6.3.2. Experiments

To reduce the patial complexity, all (M)BPTs are built from an initial partition $\mathcal{L}$ composed of 200 000 regions obtained from a cut on a standard BPT, thus providing superpixels composed each of 5 pixels, in average.

We consider five valuation functions $W_\star$. Each models either radiometric or geometrical information related to region dissimilarities.

The first valuation function W_{colour} is defined as the increase of the range of the pixel intensity values for each radiometric band, induced by the putative fusion of incident regions. Let $N_i, N_j \in \mathcal{P}$ be two adjacent regions / nodes. A multispectral image $\mathcal{I} : \Omega \to V$ associates, to each point $x \in \Omega$, a s-uple of spectral intensities $\mathcal{I}(x) = \prod_{b=1}^{s} \mathcal{I}_b(x)$. Then, W_{colour} is computed as $\frac{1}{s}\sum_{b=1}^{s} \max\{v_b^+(N_i), v_b^+(N_j)\} - \min\{v_b^-(N_i), v_b^-(N_j)\}$, where $v_b^\star$ provides the extremal values for the b-th spectral band in $\mathcal{I}$ (i.e. in $\mathcal{I}_b$).

The second valuation function W_{ndvi} quantifies the difference of NDVI between two adjacent regions. The NDVI (Normalized Difference Vegetation Index) is a standard indicator for the presence of green vegetation. For each point x, it is computed from the R and NIR channels as the ratio $(I_{NIR}(x) - I_R(x))/(I_{NIR}(x) + I_R(x))$. Practically, the W_{ndvi} valuation is computed for two nodes N_i, N_j as the absolute difference between the mean NDVI scores associated to these two regions.

The third valuation function W_{ndwi} quantifies the difference of NDWI between two adjacent regions. The NDWI (Normalized Difference Water Index)

[3] http://icube-coclico.unistra.fr/index.php/Coclico

Table 2: Scores of various (M)BPTs for built areas (see Section 6.3.3). BPT-1: W_{colour}. BPT-2: W_{comp}. BPT-3: W_{ndvi}. BPT-4: linear combination of W_{colour}, W_{comp}. BPT-5: linear combination of W_{colour}, W_{ndvi}. MBPT-1: $\{W_{colour}, W_{comp}\}$ with *most-frequent.* MBPT-2: $\{W_{colour}, W_{ndvi}\}$ with *most-frequent.*

Index	BPT-1	BPT-2	BPT-3	BPT-4	BPT-5	MBPT-1	MBPT-2
Dice (D)	**0.561**	0.432	0.393	0.466	0.376	0.548	0.547
Jaccard (J')	**0.417**	0.283	0.276	0.313	0.266	0.403	0.409

Table 3: Scores of various (M)BPTs for forest areas (see Section 6.3.3). BPT-1: W_{colour}. BPT-2: W_{ndvi}. BPT-3: linear combination of W_{colour}, W_{ndvi}. MBPT-1: $\{W_{colour}, W_{ndvi}\}$ with *mean-of-rank.* MBPT-2: $\{W_{colour}, W_{ndvi}\}$ with *most-frequent.*

Index	BPT-1	BPT-2	BPT-3	MBPT-1	MBPT-2
Dice (D)	0.612	0.684	0.679	0.685	**0.715**
Jaccard (J')	0.464	0.553	0.541	0.548	**0.573**

Table 4: Scores of various (M)BPTs for roads (see Section 6.3.3). BPT-1: W_{colour}. BPT-2: W_{comp}. BPT-3: linear combination of W_{colour}, W_{comp}. MBPT-1: $\{W_{colour}, W_{comp}\}$ with *mean-of-rank.* MBPT-2: $\{W_{colour}, W_{comp}\}$ with *most-frequent.* MBPT-3: $\{W_{colour}, W_{elong}, W_{ndvi}\}$ with *most-frequent.*

Index	BPT-1	BPT-2	BPT-3	MBPT-1	MBPT-2	MBPT-3
Dice (D)	0.633	0.443	0.466	0.509	**0.697**	0.515
Jaccard (J')	0.490	0.295	0.310	0.352	**0.564**	0.357

is a standard indicator for the presence of water. It is computed similarly to W_{ndvi}, by substituting the G channel to the R channel.

The fourth valuation function W_{elong} is defined as the change of geometrical elongation, potentially induced by the fusion of two regions. The geometrical elongation of a node N, noted $elong(N)$, is approximated as the ratio of the height and width of its bounding box. The W_{elong} valuation function is then computed for two adjacent nodes N_i, N_j as $|elong(N_i \cup N_j) - \frac{elong(N_i)+elong(N_j)}{2}|$.

The fifth valuation function W_{comp} is defined as the change of the geometrical compactness values, potentially induced by the fusion of two incident regions. The geometrical compactness of a node N, noted $comp(N)$, is approximated as the ratio of its border length and the root square of its size. The W_{comp} valuation function is computed similarly to W_{elong} by substituting *comp* to *elong*.

As a baseline, standard BPTs are built for each of the two satellite images of Figure 5(a–b), by using individually the 5 considered valuation functions. Some standard BPTs are also built in a mono-feature way, by considering linear combinations of several valuation functions.

For MBPTs, we consider the relative local information consensus policies *mean-of-ranks* and *most-frequent*, according to the position of the edges within the lists $\mathcal{W}_\star$ (applied to the first 20% of their contents). To evaluate the impact of the different valuation functions on the segmentation results, the MBPTs are then built by considering various combinations of valuation functions.

6.3.3. Quantitative analysis

For quantitative analysis, we use the evaluation framework presented in Section 6.2. Our purpose is to assess the adequacy of the BPTs and MBPTs

Table 5: Scores of various (M)BPTs for water zones (see Section 6.3.3). BPT-1: W_{colour}. BPT-2: W_{ndwi}. BPT-3: linear combination of W_{colour}, W_{ndwi}. MBPT-1: $\{W_{colour}, W_{ndwi}\}$ with *most-frequent*. MBPT-2: $\{W_{colour}, W_{elong}, W_{ndvi}\}$ with *most-frequent*.

Index	BPT-1	BPT-2	BPT-3	MBPT-1	MBPT-2
Dice (D)	0.803	0.845	0.906	0.906	**0.940**
Jaccard (J')	0.671	0.732	0.829	0.828	**0.887**

Table 6: Scores of various (M)BPTs for the five thematic classes (see Section 6.3.3). BPT-1: W_{colour}. BPT-2: W_{ndvi}. BPT-4: W_{ndwi}. BPT-6: W_{comp}. BPT-5: linear combination of W_{colour}, W_{ndwi}. MBPT-1: $\{W_{colour}, W_{ndvi}\}$ with *most-frequent*. MBPT-2: $\{W_{colour}, W_{ndwi}\}$ with *most-frequent*. MBPT-3: $\{W_{colour}, W_{ndvi}, W_{ndwi}\}$ with *most-frequent*. MBPT-4: $\{W_{colour}, W_{ndwi}\}$ with *most-frequent*. MBPT-5: $\{W_{colour}, W_{elong}, W_{ndvi}\}$ with *most-frequent*.

Index	BPT-1	BPT-2	BPT-4	BPT-6	BPT-5	MBPT-1	MBPT-2	MBPT-3
Dice (D)	0.630	0.559	0.580	0.446	0.568	**0.658**	0.656	0.656
Jaccard (J')	0.489	0.439	0.445	0.293	0.453	**0.529**	0.525	0.526

relatively to the reference segments of the GT maps. Our framework is parameterized as follows.

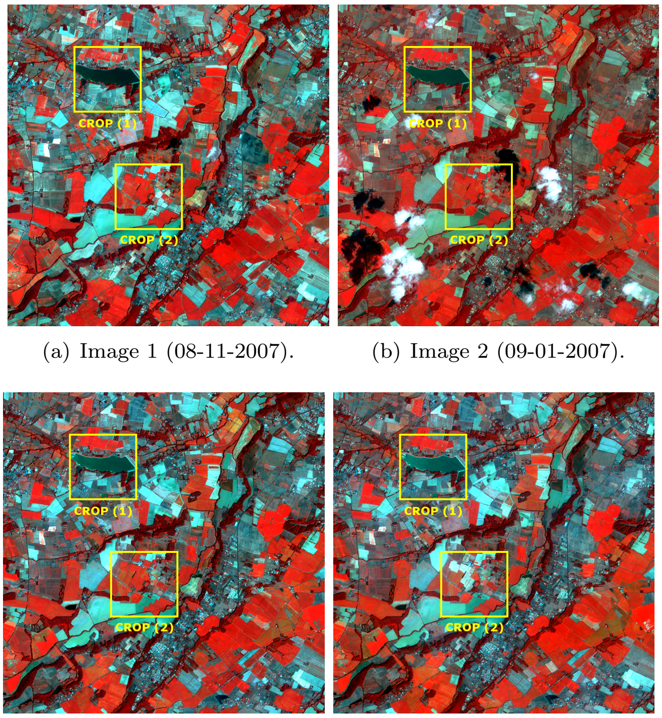
To compare nodes and GT segments, we use the Jaccard index J' and the Dice coefficient D (see Equations (1–2)). As quantitative heuristics for reducing the search space (horizontal selection), we only evaluate (M)BPT nodes whose size lies within $[0.5.|S|, 1.5.|S|]$ for a given GT segment S. The weights involved in the computation of the global quality scores are set as $w_\ell = 1/|L|$ and $\omega_i = |S_i| / \sum_{S_j \in \mathcal{C}_\ell} |S_j|$, for a GT segment S_i belonging to the semantic class C_ℓ.

In order to highlight the efficiency of BPTs (with one metric, or one linear combination of metrics) vs. MBPTs, we first compute the scores obtained for different semantic classes. In particular, Tables 3–5 provide scores for *built area*, *forest area*, *roads* and *water*, respectively. For the sake of readability, only the most representative –and most efficient– (M)BPTs are provided.

We can observe that for 2 of the 4 tested classes, the MBPTs improve both Dice and Jaccard indices. In particular, for *forest area* and *water*, the results are better than for BPTs. In the case of *roads*, one of the MBPTs provides better results than BPTs, while other MBPTs provide worse results. This emphasises the fact that MBPTs do not systematically provide better results than standard BPTs, especially when a given metric is already well fitted to a given class. For instance, in the fourth case, namely for *built area*, a standard BPT provides better results than the MBPTs, whose scores remain however fairly close. The scores obtained by BPTs and MBPTs when considering all the GT segments finally provide index values that are slightly better for MBPTs, compared to BPTs. This study tends to prove that well-chosen metrics, involved in the construction of a MBPT can allow us to slightly improve the quality of the built tree structures for segmentation purpose.

6.4. Case study 2: Multi-image segmentation

We now illustrate the interest of MBPTs for multi-image segmentation, still in the context of satellite imaging.



(a) Image 1 (08-11-2007).

(b) Image 2 (09-01-2007).

(c) Image 3 (09-08-2007).

(d) Image 4 (09-15-2007).

Figure 7: Four satellite images (1 000 × 1 000 pixels) sensed by the FORMOSAT-2 satellite covering the same geographical area (see Section 6.4.1). Note that the Image 2 is partially affected by the presence of clouds.

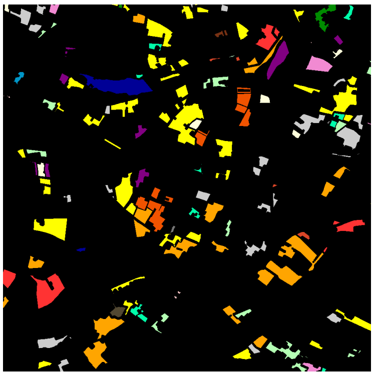


Figure 8: Ground-truth map representing partial crop fields of the studied area corresponding to the 4 images of Figure 7. This map contains 198 segments of reference delivered by the European Environment Agency.

6.4.1. Data

The dataset used here is a time series of images (1 000 × 1 000 pixels), sensed over an area located near Toulouse (France). This area is a typical agricultural

Table 7: Scores assessing the capacity of each (M)BPT to provide nodes similar to the 198 segments of reference of the ground-truth map of Figure 8 (see Section 6.4.3).

Index	BPT-1	BPT-2	BPT-3	BPT-4	MBPT-1	MBPT-2
Dice (D)	0.651	0.647	0.669	0.674	0.677	**0.700**
Jaccard (J')	0.499	0.500	0.521	0.524	0.532	**0.559**

zone, composed of different types of crop fields and vegetations. Images were acquired by the Formosat-2 in 2007. They were ortho-rectified and have a spatial resolution of 8 m, with 4 spectral bands (R, G, B, NIR)[4].

From this dataset, we selected 4 images (Figure 7). They were all acquired in late August and September. This choice allows us to reduce temporal evolution effects, since we do not aim at carrying out time-based analysis. The purpose of this experiment is to assess the ability of our MBPT framework to capture time-independent and redundant information from the content of multiple images representing a same scene.

A ground-truth map of different crop fields represented in the scene is also available (Figure 8). This map is derived from a field survey produced by the European Environment Agency (`http://ec.europa.eu/agriculture`). The semantic classes associated to this reference map reflect the temporal behavior of the considered crops over the 2007 cultural year. However, we only consider the delineation of the 198 segments of reference in the image for the quantitative evaluation of the quality of the produced BPTs.

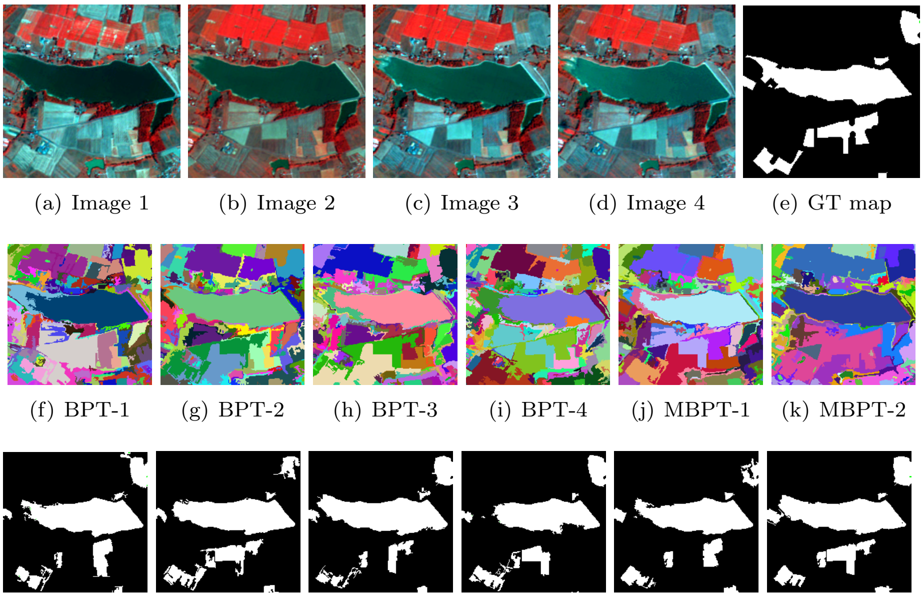
6.4.2. Experiments

To reduce the spatial complexity, all (M)BPTs are built from an initial partition $\mathcal{L}$ composed of 200 000 regions obtained from a cut on a standard BPT, thus providing superpixels composed each of 5 pixels, in average. This partition was produced from Image 1 (Figure 7(a)), which is not affected by clouds.

As a baseline, a standard BPT is built for each satellite image presented in Figure 7, with the intensity value criterion W_{colour} (denoted as W^{img1}_{colour}, W^{img2}_{colour}, W^{img3}_{colour}, W^{img4}_{colour} depending on the image where this criterion is computed). These BPTs are denoted as BPT-1, BPT-2, BPT-3 and BPT-4, respectively.

For multi-image BPTs, we consider the relative local information consensus policy *mean-of-ranks*, according to the position of the edges within the lists. It is applied to the first 15% of the lists $\mathcal{W}_\star$. To evaluate the impact of the different valuation functions on the segmentation results, two MBPTs are built. The first, denoted as MPBT-1, considers information extracted from the contents of the four satellite images. In other words, it is built from W^{img1}_{colour}, W^{img2}_{colour}, W^{img3}_{colour}, W^{img4}_{colour}. The second, denoted as MBPT-2, is built from only 3 of the 4 images, but 2 metrics (colour and NDVI). In other words, it is built from W^{img1}_{colour}, W^{img2}_{colour}, W^{img3}_{colour}, W^{img1}_{ndvi}, W^{img2}_{ndvi}, W^{img3}_{ndvi}.

[4] The authors thank the researchers from CESBIO for providing the geometrically and radiometrically corrected Formosat-2 images.



(a) Image 1 (b) Image 2 (c) Image 3 (d) Image 4 (e) GT map

(f) BPT-1 (g) BPT-2 (h) BPT-3 (i) BPT-4 (j) MBPT-1 (k) MBPT-2

(l) BPT-1/GT (m) BPT-2/GT (n) BPT-3/GT (o) BPT-4/GT (p) MPBPT-1/GT (q) MPBPT-2/GT

Figure 9: Segmentation results from standard (i.e., mono-image) BPTs and multi-image/metric MBPTs. (a–d) Crop 1, from Figure 7. (e) Crop 1, from the ground-truth map of Figure 8. (f–i) Segmentation results from the standard BPTs (BPT-1 to BPT-4), on Images 1 to 4. (j) Segmentation result from MBPT-1 with 4-images/1-metric on Images 1–4. (k) Segmentation results from MBPT-2 with 3-images/2-metrics on Images 1–3. (l–o) Best-matching nodes of the BPTs with respect to the ground-truth map segments. (p–q) Best-matching nodes of the MBPTs with respect to the ground-truth map segments.

6.4.3. Quantitative analysis

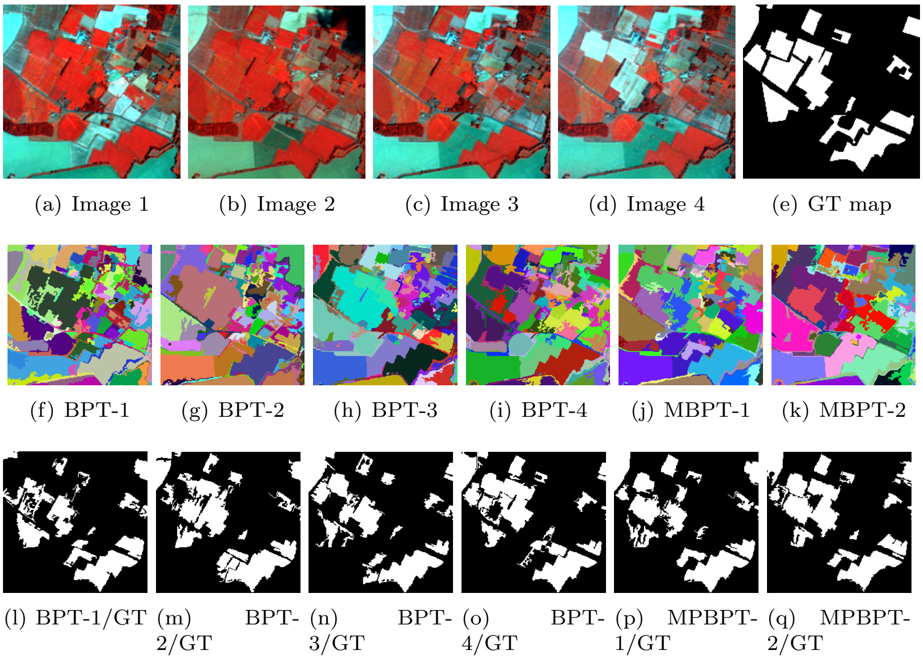
For quantitative analysis, we use the evaluation framework presented in Section 6.2. Our purpose is to assess the adequacy of the BPTs and MBPTs with respect to the reference segments of the ground-truth map. Our framework is parameterized the same way as in Section 6.3.

Table 7 presents the different scores obtained for the different BPTs and MBPTs. The best scores for both Dice and Jaccard indices are obtained by MBPT-2, built from 3 images and 2 metrics (W_{colour}, W_{ndvi}). The other MBPT, built from 4 images, presents lower scores than MBPT-2, which remain however slightly higher than those of standard BPTs for the 4 images.

This quantitative evaluation suggests that MBPTs are able to improve the quality of segmentation results obtained from standard BPTs. The increase of the index values is slight, but indeed sufficient to show that MBPTs constitute a fair structure for gathering and merging information collected from various images of a same scene.

6.4.4. Qualitative analysis

Beyond the quantitative assessment of the adequacy of (M)BPTs with ground-truth segments, it is also relevant to provide qualitative illustration of



(a) Image 1 (b) Image 2 (c) Image 3 (d) Image 4 (e) GT map

(f) BPT-1 (g) BPT-2 (h) BPT-3 (i) BPT-4 (j) MBPT-1 (k) MBPT-2

(l) BPT-1/GT (m) BPT-2/GT (n) BPT-3/GT (o) BPT-4/GT (p) MPBPT-1/GT (q) MPBPT-2/GT

Figure 10: Segmentation results from standard (i.e., mono-image) BPTs and multi-image/metric MBPTs. (a–d) Crop 2, from Figure 7. (e) Crop 2, from the ground-truth map of Figure 8. (f–i) Segmentation results from the standard BPTs (BPT-1 to BPT-4), on Images 1 to 4. (j) Segmentation result from MBPT-1 with 4-images/1-metric on Images 1–4. (k) Segmentation results from MBPT-2 with 3-images/2-metrics on Images 1–3. (l–o) Best-matching nodes of the BPTs with respect to the ground-truth map segments. (p–q) Best-matching nodes of the MBPTs with respect to the ground-truth map segments.

the potential of MBPTs versus standard BPTs. These qualitative analysis is indeed complementary to the above analysis.

On the one hand, the different (M)BPTs are segmented by a user-defined horizontal cut. We chose a simple segmentation paradigm, in order to avoid any bias related to the segmentation process. Such tree-cut produces a partition with a scale adapted, according to the user, to segment the geographical objects covering the sensed areas (e.g. agricultural crop fields, wide forest areas, lake).

On the other hand, we illustrate the nodes of the (M)BPTs that provide the best-matching with the segments of reference of the ground-truth map. These nodes are indeed those involved in the computation of the Dice and Jaccard indices, in the previous quantitative analysis.

For a better visualization, we focus on two image crops representing different parts of the geographical scene. Crops 1 and 2 are illustrated in Figures 9(a–d) and Figures 10(a–d), respectively.

In Crop 1, we observe that, despite of variations due to time evolutions, the global structure of the crop fields does not evolve much. Based on a visual interpretation of these images, it appears that the majority of observed geographical areas preserve the same crop field partition. However, each segmentation obtained from the standard BPTs (BPT-1 to BPT-4) presents different partitions, see Figure 9(f–i). The results obtained from the multi-image MBPT-1 and MBPT-2, see Figure 9(j–k), arise from a consensus between the results of the 4 standard BPT results.

A better observation of this phenomenon can be observed in Figure 9(p–q) which gathers consensual information between the relevant segments found in the Figure 9(l–o). The best-matching nodes obtained from our multi-image/metric MBPTs are visually close to the segments of reference of the GT map presented in the Figure 9(e). Such nodes are not necessarily the same as in the arbitrary cuts of Figure 9(f–k). However, they are likely to be obtained in a segmentation result induced by a more sophisticated tree-cut process, compared to the trivial user-defined horizontal cut.

The same behaviour can be observed in Crop 2, see Figure 10. These qualitative results corroborate the previous quantitative analysis and emphasise the relevance of MBPTs, which provide different behaviour than standard BPTs, with slightly better results for relevant choices of images and metrics.

7. Conclusion

In this article, we proposed a generalization of the BPT construction framework. While BPTs are classically built in a mono-feature way, we considered here a multi-feature paradigm.

The induced algorithmic evolutions require the handling of more complex data structures and consensual algorithms, compared to standard BPTs. In order to tackle memory and time complexity issues, a short-term perspective of this work will be to implement distributed heuristics relying on graph-based distribution. Integrating higher-level consensus may also allow us to improve the relevance of the hierarchies and the induced segmentation.

Beyond the application examples described in this article, other relevant use-cases could be considered for the processing of different families of images. As an example, it is possible to apply MBPTs to segment hyperspectral images, by establishing a consensus between the complementary (and potentially correlated) information carried by the different spectral bands. Multi-time imaging can also be considered, by establishing a higher-level time consensus between the different image acquisitions of the same scene.

Another methodological challenge is raised by the possible divergence between the different values gathered by the metrics/images, which may lead to irrelevant consensual decisions. We plan to study how non-consensual information could be used to follow local consensus between metrics/images leading to hypertrees where the branches model local fusion decisions.

References

[1] P. Salembier, M. H. F. Wilkinson, Connected operators: A review of region-based morphological image processing techniques, IEEE Signal Processing Magazine 26 (6) (2009) 136–157.

[2] L. Najman, J. Cousty, A graph-based mathematical morphology reader, Pattern Recognition Letters 47 (2014) 3–17.

[3] L. Najman, H. Talbot (Eds.), Mathematical Morphology: From Theory to Applications, ISTE/J. Wiley & Sons, 2010.

[4] P. Salembier, L. Garrido, Binary partition tree as an efficient representation for image processing, segmentation, and information retrieval, IEEE Transactions on Image Processing 9 (4) (2000) 561–576.

[5] J. F. Randrianasoa, C. Kurtz, É. Desjardin, N. Passat, Multi-image segmentation: A collaborative approach based on binary partition trees, in: ISMM, International Symposium on Mathematical Morphology, Proceedings, Vol. 9082 of Lecture Notes in Computer Science, Springer, 2015, pp. 253–264.

[6] A. Rosenfeld, Connectivity in digital pictures, Journal of the ACM 17 (1) (1970) 146–160.

[7] R. Adams, L. Bischof, Seeded region growing, IEEE Transactions on Pattern Analysis and Machine Intelligence 16 (6) (1994) 641–647.

[8] L. Vincent, P. Soille, Watersheds in digital spaces: An efficient algorithm based on immersion simulations, IEEE Transactions on Pattern Analysis and Machine Intelligence 13 (6) (1991) 583–598.

[9] S. L. Horowitz, T. Pavlidis, Picture segmentation by a directed split-and-merge procedure, in: Second International Joint Conference on Pattern Recognition, Proceedings, Vol. 424, 1974, p. 433.

[10] Y. Boykov, O. Veksler, R. Zabih, Fast approximate energy minimization via graph cuts, IEEE Transactions on Pattern Analysis and Machine Intelligence 23 (11) (2001) 1222–1239.

[11] L. Grady, Random walks for image segmentation, IEEE Transactions on Pattern Analysis and Machine Intelligence 28 (11) (2006) 1768–1783.

[12] C. Couprie, L. Grady, L. Najman, H. Talbot, Power watershed: A unifying graph-based optimization framework, IEEE Transactions on Pattern Analysis and Machine Intelligence 33 (7) (2011) 1384–1398.

[13] P. Salembier, J. Serra, Flat zones filtering, connected operators, and filters by reconstruction, IEEE Transactions on Image Processing 4 (8) (1995) 1153–1160.

[14] M.-M. Yau, S. N. Srihari, A hierarchical data structure for multidimensional digital images, Communications of the ACM 26 (7) (1983) 504–515.

[15] A. Montanvert, P. Meer, A. Rosenfeld, Hierarchical image analysis using irregular tessellations, IEEE Transactions on Pattern Analysis and Machine Intelligence 13 (4) (1991) 307–316.

[16] P. Salembier, A. Oliveras, L. Garrido, Anti-extensive connected operators for image and sequence processing, IEEE Transactions on Image Processing 7 (4) (1998) 555–570.

[17] P. Monasse, F. Guichard, Scale-space from a level lines tree, Journal of Visual Communication and Image Representation 11 (2) (2000) 224–236.

[18] P. Monasse, F. Guichard, Fast computation of a contrast-invariant image representation, IEEE Transactions on Image Processing 9 (5) (2000) 860–872.

[19] L. Najman, M. Schmitt, Geodesic saliency of watershed contours and hierarchical segmentation, IEEE Transactions on Pattern Analysis and Machine Intelligence 18 (12) (1996) 1163–1173.

[20] B. Perret, S. Lefèvre, C. Collet, E. Slezak, Hyperconnections and hierarchical representations for grayscale and multiband image processing, IEEE Transactions on Image Processing 21 (1) (2012) 14–27.

[21] R. Jones, Connected filtering and segmentation using component trees, Computer Vision and Image Understanding 75 (3) (1999) 215–228.

[22] L. Guigues, J.-P. Cocquerez, H. Le Men, Scale-sets image analysis, International Journal of Computer Vision 68 (3) (2006) 289–317.

[23] N. Passat, B. Naegel, F. Rousseau, M. Koob, J.-L. Dietemann, Interactive segmentation based on component-trees, Pattern Recognition 44 (10–11) (2011) 2539–2554.

[24] N. Passat, B. Naegel, Component-trees and multivalued images: Structural properties, Journal of Mathematical Imaging and Vision 49 (1) (2014) 37–50.

[25] C. Kurtz, B. Naegel, N. Passat, Connected filtering based on multivalued component-trees, IEEE Transactions on Image Processing 23 (12) (2014) 5152–5164.

[26] E. Carlinet, T. Géraud, MToS: A tree of shapes for multivariate images, IEEE Transactions on Image Processing 24 (12) (2015) 5330–5342.

[27] N. Passat, B. Naegel, Component-hypertrees for image segmentation, in: ISMM, International Symposium on Mathematical Morphology, Proceedings, Vol. 6671 of Lecture Notes in Computer Science, Springer, 2011, pp. 284–295.

[28] B. Perret, J. Cousty, O. Tankyevych, H. Talbot, N. Passat, Directed connected operators: Asymmetric hierarchies for image filtering and segmentation, IEEE Transactions on Pattern Analysis and Machine Intelligence 37 (6) (2015) 1162–1176.

[29] Y. Xu, T. Géraud, L. Najman, Connected filtering on tree-based shape-spaces, IEEE Transactions on Pattern Analysis and Machine Intelligence 38 (6) (2016) 1126–1140.

[30] J. L. Bentley, Multidimensional binary search trees used for associative searching, Communications of the ACM 18 (9) (1975) 509–517.

[31] P. Soille, Constrained connectivity for hierarchical image partitioning and simplification, IEEE Transactions on Pattern Analysis and Machine Intelligence 30 (7) (2008) 1132–1145.

[32] S. Valero, P. Salembier, J. Chanussot, Comparison of merging orders and pruning strategies for binary partition tree in hyperspectral data, in: ICIP, International Conference on Image Processing, Proceedings, 2010, pp. 2565–2568.

[33] X. Giro, F. Marqués, From partition trees to semantic trees, in: MRCS, Multimedia Content Representation, Classification and Security, Proceedings, Vol. 4105 of Lecture Notes in Computer Science, Springer, 2006, pp. 306–313.

[34] V. Vilaplana, F. Marques, P. Salembier, Binary partition trees for object detection, IEEE Transactions on Image Processing 17 (11) (2008) 2201–2216.

[35] S. Valero, P. Salembier, J. Chanussot, Object recognition in hyperspectral images using binary partition tree representation, Pattern Recognition Letters 56 (2015) 45–51.

[36] J. A. Benediktsson, L. Bruzzone, J. Chanussot, M. Dalla Mura, P. Salembier, S. Valero, Hierarchical analysis of remote sensing data: Morphological attribute profiles and binary partition trees, in: ISMM, International Symposium on Mathematical Morphology, Proceedings, Lecture Notes in Computer Science, Springer, 2011, pp. 306–319.

[37] C. Kurtz, N. Passat, P. Gançarski, A. Puissant, Extraction of complex patterns from multiresolution remote sensing images: A hierarchical top-down methodology, Pattern Recognition 45 (2) (2012) 685–706.

[38] F. Calderero, F. Eugenio, J. Marcello, F. Marques, Multispectral cooperative partition sequence fusion for joint classification and hierarchical segmentation, IEEE Geoscience and Remote Sensing Letters 9 (6) (2012) 1012–1016.

[39] C. Kurtz, A. Stumpf, J.-P. Malet, P. Gançarski, A. Puissant, N. Passat, Hierarchical extraction of landslides from multiresolution remotely sensed optical images, ISPRS Journal of Photogrammetry and Remote Sensing 87 (2014) 122–136.

[40] S. Valero, P. Salembier, J. Chanussot, Hyperspectral image representation and processing with binary partition trees, IEEE Transactions on Image Processing 22 (4) (2013) 1430–1443.

[41] M. A. Veganzones, G. Tochon, M. Dalla Mura, A. J. Plaza, J. Chanussot, Hyperspectral image segmentation using a new spectral unmixing-based binary partition tree representation, IEEE Transactions on Image Processing 23 (8) (2014) 3574–3589.

[42] A. Alonso-González, C. López-Martínez, P. Salembier, Filtering and segmentation of polarimetric SAR data based on binary partition trees, IEEE Transactions on Geoscience and Remote Sensing 50 (2) (2012) 593–605.

[43] P. Salembier, Study of binary partition tree pruning techniques for polarimetric SAR images, in: ISMM, International Symposium on Mathematical Morphology, Proceedings, Vol. 9082 of Lecture Notes in Computer Science, Springer, 2015, pp. 51–62.

[44] A. Alonso-González, S. Valero, J. Chanussot, C. López-Martínez, P. Salembier, Processing multidimensional SAR and hyperspectral images with binary partition tree, Proceedings of the IEEE 101 (3) (2013) 723–747.

[45] A. Alonso-González, C. López-Martínez, P. Salembier, PolSAR time series processing with binary partition trees, IEEE Transactions on Geoscience and Remote Sensing 52 (6) (2014) 3553–3567.

[46] J. Cousty, L. Najman, B. Perret, Constructive links between some morphological hierarchies on edge-weighted graphs, in: ISMM, International Symposium on Mathematical Morphology, Proceedings, Vol. 7883 of Lecture Notes in Computer Science, Springer, 2013, pp. 86–97.

[47] J. F. Randrianasoa, C. Kurtz, P. Gançarski, É. Desjardin, N. Passat, Evaluating the quality of binary partition trees based on uncertain semantic ground-truth for image segmentation, in: ICIP, International Conference on Image Processing, Proceedings, 2017, pp. 3874–3878.

[48] P. Jaccard, The distribution of the flora in the Alpine zone, New Phytologist 11 (1912) 37–50.

[49] L. Dice, Measures of the amount of ecologic association between species, Ecology 26 (1945) 297–302.