Supervised Assessment of Segmentation Hierarchies

Jordi Pont-Tuset and Ferran Marques*

Universitat Politècnica de Catalunya BarcelonaTech,
Jordi Girona, 1-3, 08034, Barcelona, Spain
http://imatge.upc.edu

Abstract. This paper addresses the problem of the supervised assessment of hierarchical region-based image representations. Given the large amount of partitions represented in such structures, the supervised assessment approaches in the literature are based on selecting a reduced set of representative partitions and evaluating their quality. Assessment results, therefore, depend on the partition selection strategy used. Instead, we propose to find the partition in the tree that best matches the ground-truth partition, that is, the upper-bound partition selection. We show that different partition selection algorithms can lead to different conclusions regarding the quality of the assessed trees and that the upper-bound partition selection provides the following advantages: 1) it does not limit the assessment to a reduced set of partitions, and 2) it better discriminates the random trees from actual ones, which reflects a better qualitative behavior. We model the problem as a Linear Fractional Combinatorial Optimization (LFCO) problem, which makes the upper-bound selection feasible and efficient.

1 Introduction

Region-based hierarchical image representations have proven their applicability in many fields such as segmentation, filtering, information retrieval [1]; object detection [2–4], contour detection [5, 6], etc.

Any hierarchy of nested regions based on a set of non-overlapping regions can be represented by a binary tree of regions (such as *Binary Partition Trees* (BPT) [1] or *Ultrametric Contour Map* (UCM) trees [5]), so although this work is focused on this type of trees, the results are generalizable to any hierarchy of regions such as quad trees [7].

A supervised assessment has been the most used to prove the validity of these representations, that is, comparing the results to a set of manually-generated partitions known as *ground truth*. However, comparing the large collection of partitions represented in a hierarchy to a non-hierarchical partition is not straightforward.

The approaches found in the literature consist in selecting a set of *representative* partitions from the tree and comparing them to the ground-truth partitions. This way, for each partition of the ground-truth database, there will be a set of values that indicate the quality of that particular tree.

* This work has been partially supported by the Spanish *Ministerio de Ciencia e Innovación*, under project TEC2010-18094 and FPU grant AP2008-01164.

To average these results on a whole database, the representative partitions of the tree on each image of the database have to be put in correspondence (*align*) with the representative partitions of the trees of the rest of images, that is, there has to be a common parameter that *indexes* each set of representative partitions (e.g. their number of regions). Overall, aggregate results depend on a **partition selection algorithm** and an **alignment** procedure.

For instance, in [8, 9], the set of selected partitions are the ones formed in the merging sequence, aligned by their number of regions. The latter proposes a second alignment based on the accumulated merging cost threshold. In [5], the selected regions are also the ones in the merging sequence, but the alignment parameter is the confidence threshold on the ultrametric contour map.

Ideally, assessment results should depend mainly on the trees themselves, otherwise it is not clear whether the obtained results are due to the tree itself, or to the alignment and partition selection algorithms. To make results independent of the former, [5] proposes the *Optimal Image Scale* (OIS) analysis, which averages the best result in the representative set of each tree.

This paper proposes a technique to make the assessment results independent of the partition selection algorithm via the **upper-bound partition selection**, that is, computing the optimal results that can be achieved by any partition selection procedure.

In the case of the OIS, the maximum performance for each image is searched by brute force among all possible partitions in the merging sequence [5]. However, exhaustively searching the upper-bound performance among all possible partitions in a tree to make results independent of the partition selection algorithm is not computationally feasible. To overcome this limitation, we propose to model the problem of finding the best partition selection as a *Linear Fractional Combinatorial Optimization* (LFCO), which can be efficiently solved by the procedure presented in [10].

We show that the upper-bound partition selection has the following advantages in the assessment of region-based hierarchies. First, it expands the range of partitions assessed beyond the merging sequence. Note that this is a relevant feature, since there are image analysis works such as [2, 1] that extract partitions that are not in the merging sequence, and so such partitions would not be covered by the previous selection approaches. Second, we demonstrate that the partition selection technique may mislead the assessment of the tree quality, in the sense that the ranking between results is different from the upper-bound in a significant number of cases of the experiments. Finally, we show that the upper-bound partition selection has a better discriminative power between the baseline method of computing the hierarchy randomly and actual hierarchies.

The remainder of the paper is organized as follows: Section 2 presents the different trees that are used in this work and Section 3 expounds on the supervised techniques found in the literature to assess these hierarchies. Then, in Section 4 we present the step-by-step deduction and motivation of the LFCO model that we propose to find the upper-bound partition selection. Section 5 presents the experiments performed to evaluate and compare our algorithm and in Section 6 we draw the conclusions.

2 Hierarchy Creation Algorithm

In this paper we explore a region-based hierarchical image representation consisting of a binary tree, where each node represents a region in the image, and the parent node of a pair of regions represents their merging. This structure is referred to as *Binary Partition Tree* (BPT) in [1, 2, 8, 9]. The *Ultrametric Contour Map* (UCM) [5] hierarchy of regions is also a binary tree.

The algorithm to build both BPT and UCM is a greedy region merging algorithm that, starting from an initial partition P_0, iteratively merges the most similar pair of neighboring regions. The concept of region *similarity* is what makes the difference between both approaches.

In the case of the BPT, each region is represented by a model such as the color mean and contour complexity [8] or the color histogram [9], and the region similarity is obtained comparing their models. The UCM [5], in contrast, defines the dissimilarity between two neighboring regions as the strength of the *Oriented Watershed Transform* (OWT) of the *globalized Probability of boundary* (gPb) in the common boundary.

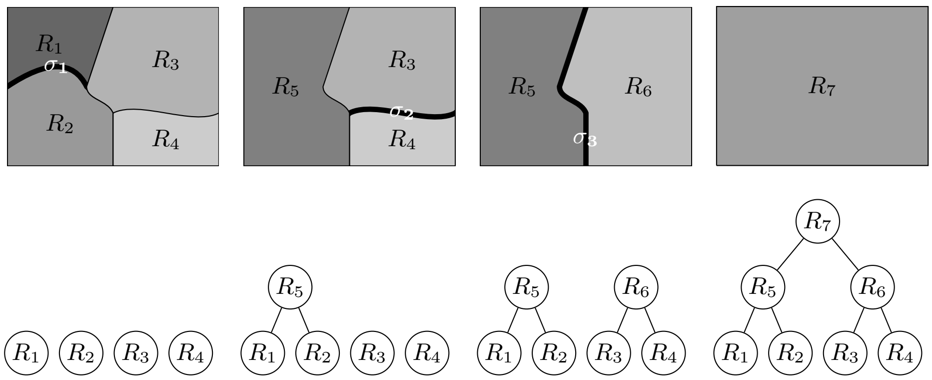

Fig. 1. BPT creation process: Above, the merging sequence partition set where, from left to right, two neighboring regions are merged at each step. The common boundary between them is highlighted. Below, the BPT representation depicted by a tree, where the region formed from the merging of two segments is represented as the parent of the two respective nodes

The merging process ends when one single region remains, the whole image, which is represented by the root of the tree. The set of mergings that create the tree, from the starting partition to the whole image, is usually referred to as *merging sequence* and the set of partitions that are iteratively formed in the process is known as *merging-sequence partition set*.

To illustrate the process of creation of the hierarchies, Figure 1 shows the tree and the partition at each step of the merging sequence. In this example, the merging sequence is $\{R_1+R_2\rightarrow R_5\ , R_3+R_4\rightarrow R_6\ , R_5+R_6\rightarrow R_7\}$, and the merging-sequence partition set is formed by the four represented partitions.

3 Hierarchy Quality Assessment

The quality of a hierarchical region-based image representation is usually assessed in a supervised environment, that is, comparing how *accurate* the representation is with respect to a human-generated ground truth. Given that a hierarchical region-based image representation is a structured set of image partitions from the most detailed ones (more regions) to the coarsest ones (less regions), an intuitive approach to assess the representation could be to compare *a set* of representative partitions *selected* from the hierarchy and *aligned* by an index that represents the level of detail.

This is the approach followed in [8, 9], where the quality of the tree is represented by the assessment of the set of selected partitions in the so-called *merging-sequence partition set*, that is, the partitions formed at each step of the tree merging sequence. The number of regions is the index that represents the level of detail of the partitions. In other words, to assess the trees for various ground-truth images, the sets of partitions are put in correspondence by their number of regions to obtain an average result.

In [5], the same selection approach applies, but in this case the partitions to be assessed are selected via the thresholding of the Ultrametric Contour Map (UCM) at different levels of confidence. Therefore, the difference here is that the partitions are *aligned* with respect to this threshold value.

In other words, the same threshold for two UCM trees can correspond to different number of regions. This way, the aggregate results on a database will be different depending on the alignment used.

The strategy to average the results *aligning* them with respect to a certain parameter is referred to as Optimal Dataset Scale (ODS) in [5]. To avoid the dependence of the results of an alignment process, the same work proposes the Optimal Image Scale (OIS) which, in contrast, averages the quality of the best partition *in the selected set* for each image. That is, the rationale behind the OIS is to average the upper-bound performance of the UCM trees, avoiding the use of an *alignment*.

However, limiting the partitions assessed to those of a reduced set among all found in a hierarchy is also masking the real upper-bound performance of the technique, since this approach is not assessing all the partitions represented in the tree.

The proposal of this work is to find the upper-bound performance regardless also of the representative partition set selection, that is, independently of whether we assess those partitions from a thresholding of the UCM, those forming the set of merging-sequence partitions, etc. We will refer to the resulting selection strategies as upper-bound ODS and upper-bound OIS (ubODS and ubOIS, respectively).

The number of partitions represented in a binary tree, however, grows rapidly with respect to the number of initial regions, so it would not be feasible to assess all of them using brute force. To do so, the main objective of this paper is to model the problem as a *Linear Fractional Combinatorial Optimization* (LFCO) problem [10], which allows us to find the partitions that entail the upper-bound quality using a feasible algorithm.

The *F measure for boundary detection* (F_b) [5] measures the trade-off between the precision and recall of the matching between the boundary pixels of the ground truth and the assessed partition. Although this measure was initially designed to assess contour detectors, [5] states that: “While the relative ranking of segmentation algorithms remains fairly consistent across different benchmark criteria, the boundary benchmark appears most capable of discriminating performance."

As we will present on the following section, F_b can be written in the fractional form of an LFCO, and thus fulfills the objective of feasibility. Therefore, adding the good behavior of this measure perceived by [5], we will base our assessment on the F measure for boundary detection F_b.

4 Upper-Bound Partition Selection

The computation of F_b is based on a global optimal matching between the set of boundary pixels of the partition to be assessed and those of the ground truth. To avoid performing a matching for each of the partitions represented in a tree, which is computationally prohibitive, we propose an algorithm that performs a local matching between the ground truth and each of the *pieces* of region boundaries of the tree. This allows us to efficiently find the upper bound of the optimal global matching for any represented partition.

Formally, let P_0 be the partition on which a hierarchy H is built and $R_1 \ldots R_n$ its regions. Let $\{R_{i_1} + R_{i_2} \rightarrow R_{i_3}\}, i = 1 \ldots n-1$ be the merging sequence that forms H. We define σ_i as the common boundary between the regions that are merged at step i of the merging sequence. (Figure 1 depicts σ_1, σ_2, and σ_3 of the example tree.) Note that this set of common boundaries is not the full set of common boundaries between pairs of regions of P_0, but only those between regions merged in the hierarchy H.

Let $\mathcal{P}$ be the set of all partitions represented in the hierarchy H. Any partition $P \in H$ can be unequivocally described by the set of σ_i that forms its boundaries. Let $\mathbf{p} \in \{0,1\}^{n-1}$ be a binary vector such that $p_i = 1$ if the boundaries of P contain σ_i. In Figure 1, for example, the set of merging-sequence partitions can be identified by the vectors: $\mathbf{p} = (1,\ 1,\ 1)$, $(0,\ 1,\ 1)$, and $(0,\ 0,\ 1)$.

This way, one can define a bijection between the set of partitions $\mathcal{P}$ and a subset $\chi \subset \{0,1\}^{n-1}$. Our approach to find the partition that entails the best matching relies on modeling the problem as a binary search in χ and solving it using computationally feasible techniques.

Specifically, we will model the upper-bound partition selection as a Linear Fractional Combinatorial Optimization (LFCO) problem [10]:

$$\text{LFCO:} \quad \underset{\mathbf{x}}{\text{maximize}}\ \frac{\mathbf{t} \cdot \mathbf{x}^T}{\mathbf{f} \cdot \mathbf{x}^T} \qquad \text{s.t. } \mathbf{x} \in \chi \subset \{0,1\}^{n-1} \tag{1}$$

being $\mathbf{f}, \mathbf{t} \in \mathbb{R}^{n-1}$ and all the constraints that define χ linear.

Section 4.1 explores the constraints that have to be put to the vector $\mathbf{p}$ in order for the corresponding partition to be valid within the hierarchy (that is, define χ) and how to make them linear. Next, Section 4.2 presents how the F_b of a partition with respect to a ground truth can be obtained from $\mathbf{p}$ in the form of an LFCO such as that of Equation 1. Finally, Section 4.3 adds the needed constraints to be able to find the ubODS.

4.1 Forcing the partition to be in the hierarchy

Not all combinations of boundaries σ_i form a valid partition of the hierarchy and thus not all $\mathbf{p} \in \{0,1\}^{n-1}$ correspond to feasible solutions of our problem. Recalling the example of Figure 1, for instance, the partition corresponding to $[1\ 0\ 1]$ is a valid partition in the hierarchy, while the ones corresponding to $[1\ 0\ 0]$ or $[1\ 1\ 0]$ are not.

Let $\Sigma_i = \{i_j | j = 1 \dots n_i\}$ be the indices of the set of boundaries σ_{i_j} between pairs of regions among the children of the two regions that define σ_i. In the example, for σ_3, $\Sigma_3 = \{1, 2\}$.

Then, if the two regions that form σ_i are merged ($p_i = 0$), all the pairs of regions that form the boundaries indexed by Σ_i are forced to be also merged ($p_{i_j} = 0$). Formally $p_i = 0 \ \Rightarrow\ p_{i_j} = 0 \quad \forall\, i_j \in \Sigma_i$, or equivalently the following constraints:

$$p_i = 1 \quad \text{or} \quad \sum_{i_j \in \Sigma_i} p_{i_j} = 0 \tag{2}$$

In other words, if the boundary between two regions is not in the partition, the boundaries between any pair of their children cannot be in the partition either.

The binary search problem we are modeling will be much more efficient to solve if it is linear. The following linear constraint is equivalent to Equation 2:

$$\sum_{i_j \in \Sigma_i} p_{i_j} \ \leq\ K\, p_i \tag{3}$$

where K is a *sufficiently large* constant, which in our problem can be set to n, the number of regions.

To conclude, the set of partitions represented in the hierarchy H can be identified with the set:

$$\chi = \left\{ \mathbf{p} \in \{0,1\}^{n-1} \middle| \sum_{i_j \in \Sigma_i} p_{i_j} \ \leq\ n\, p_i \right\}.$$

In the sequel, any partition P in the hierarchy H will be identified by its corresponding binary vector $\mathbf{p} \in \chi$.

4.2 Upper-bound partition selection as an LFCO

For a given partition $P \in H$ ($\mathbf{p} \in \chi$), let TP be the set of matched boundary pixels with the boundary pixels of a ground truth partition, i.e. true positives, and FP the false positives set. We can write that $|TP| = \sum_{i=1}^{n-1} p_i\, |\sigma_i^m|$, $|FP| = \sum_{i=1}^{n-1} p_i\, |\sigma_i^u|$, where $\sigma_i = \sigma_i^m \cup \sigma_i^u$ is a division of the boundary pixels between *matched* and *unmatched*, respectively.

The first approach we propose is to perform a single matching between the boundary pixels of the original partition P_0 and those of the ground-truth partition, and define σ_i^m and σ_i^u as those sets of pixels of σ_i matched or unmatched, respectively.

If we define $\boldsymbol{\sigma}^m = \left(|\sigma_1^m|, \dots, |\sigma_{n-1}^m|\right) \in \mathbb{N}^{n-1}$, $\boldsymbol{\sigma} = (|\sigma_1|, \dots, |\sigma_{n-1}|) \in \mathbb{N}^{n-1}$, the problem of finding the partition in the hierarchy with the best F_b with respect to the ground truth can be written as:

$$\mathcal{F}: \underset{\mathbf{p}}{\text{maximize}} \;\; F_b = 2\frac{(\boldsymbol{\sigma}^m,0)\cdot(\mathbf{p},1)^T}{(\boldsymbol{\sigma},|P_{gt}|)\cdot(\mathbf{p},1)^T}, \quad \text{s.t.} \quad (3)$$

This type of problem is referred to as a *Linear Fractional Combinatorial Optimization* (LFCO) problem in [10], which also presents an efficient way to solve it. The remainder of this section is devoted to present the limitation of this approach: the ground-truth multi-matching and the solution we propose.

Ground-truth multi-matching

The previous matching strategy presents the following problem: the matching is carried out at the level of the original partition P_0, assuming it is optimal for all possible combinations of pieces of boundaries in the hierarchy. More precisely, this approach assumes that the sets σ_i^m and σ_i^u do not depend on the partition $\mathbf{p}$ being analyzed; or in other words, that the optimal matching for any partition $\mathbf{p}$ can be obtained from the initial matching on P_0.

In order to illustrate this problem, Figure 2 depicts an example partition (a) and the correspondent ground truth (b). If pixels are matched globally, as presented in the previous section, let us assume that all pixels in σ_1 are matched to all M ground-truth pixels. Then, we would have that $\sigma_1^m = M$ and $\sigma_2^m = 0$, that is, no pixel in σ_2 would be matched at the level of P_0. When computing the number of matched ground-truth pixels for the partition identified by $\mathbf{p} = (0, 1)$, we would find that the number of matched pixels is $\boldsymbol{\sigma}^m \cdot \mathbf{p} = 0$, but the right portion of the ground-truth boundary should be matched to σ_2, that is, the correct result should be $\boldsymbol{\sigma}^m \cdot \mathbf{p} = M/2$.

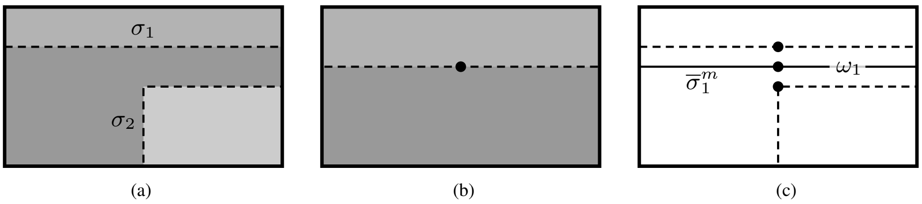

Fig. 2. Ground-truth multi-matching representation: (a) Partition being assessed, (b) ground truth, (c) both partitions overlaid. The points are plotted to highlight the division of the boundary pixels into sets

The approach we propose to solve this issue is to perform $n-1$ matchings between the pixels of the ground-truth partition and those of each σ_i, and define σ_i^m and σ_i^u as those sets of pixels locally matched or unmatched, respectively. In other words, some pixels of the ground truth can be matched with more than one boundary segment, and thus we call it multi-matching.

Formally, once performed the $n-1$ matchings between the ground-truth boundary pixels and each σ_i, each boundary pixel of the ground truth may be matched to a boundary pixels of some σ_i (from 1 to $n-1$) of the partition. Understanding the set of indices of each σ_i involved in the multi matching as a *signature* of each of the ground-truth boundary pixels, we divide these ground-truth boundary pixels into groups of equal signature.

This way, for instance, we will have a set of unmatched pixels, $n-1$ sets of single-matched pixels which we will denote as $\overline{\sigma}_i^m$ (see Figure 2), and the rest will have more than one index in the signature. Intuitively, we will count a ground-truth boundary pixel as matched only if any of the σ_i in its signature is in the partition but not counting it more than once.

To do so, and in order to have a compact modeling, let us group the set of ground-truth boundary pixels with equal signature and define the set as ω_j. Moreover, let us assume we have m different multiple-index sets of pixels ω_j with signatures $\Omega_j = \{s_1^j, \ldots, s_k^j\}$. For instance, the pixels in the set ω_1 of the example of Figure 2 (see (c)) are each of them multi-matched to pixels in σ_1 and σ_2, then their signature is $\Omega_j = \{1, 2\}$. Let $\mathbf{q} \in \{0,1\}^m$ be a vector such that $q_j = 1$ if the set of pixels in ω_j should be considered as matched. The value of q_j is function of the values in the signature, that is, $q_j = 1$ if any $p_{s_i^j} = 1$ and 0 otherwise. Mathematically:

$$q_j = p_{s_1^j} \text{ or } p_{s_2^j} \text{ or } \cdots \text{ or } p_{s_{k_j}^j}$$

The equivalent linear constraints that define this equation are:

$$q_j \leq \sum_{s \in \Omega_j} p_s \tag{4}$$

$$q_j \geq p_s \qquad \forall\, s \in \Omega_j \tag{5}$$

Let us define $\overline{\boldsymbol{\sigma}} = \left(|\overline{\sigma}_1^m|, \ldots, |\overline{\sigma}_{n-1}^m|\right) \in \mathbb{N}^{n-1}$ be the vector of single-matched number of ground-truth boundary pixels for each σ_i, and $\boldsymbol{\omega} = (|\omega_1|, \ldots, |\omega_m|) \in \mathbb{N}^m$ the vector of the number of ground-truth boundary pixels with equal signature. Then, the problem $\mathcal{F}$ can be rewritten as:

$$\begin{aligned} \mathcal{F}: \ \underset{\mathbf{p},\mathbf{q}}{\text{maximize}} \ \ & F_b = 2\frac{(\overline{\boldsymbol{\sigma}}, \boldsymbol{\omega}, 0) \cdot (\mathbf{p}, \mathbf{q}, 1)^T}{(\boldsymbol{\sigma}, \mathbf{0}, |P_{gt}|) \cdot (\mathbf{p}, \mathbf{q}, 1)^T} \\ \text{subject to} \ \ & (3), (4), (5) \end{aligned} \tag{6}$$

which, as wanted, fulfills the form of an LFCO as in Equation 1, identifying $\mathbf{x} = (\mathbf{p}, \mathbf{q}, 1)$ as the binary-valued variable of the problem.

4.3 ubODS: Sweeping the number of regions

The problem 6 finds the optimal single partition in terms of F_b so, in other words, it finds the upper-bound Optimal Image Scale (ubOIS) partition. Given that a hierarchy represents a collection of partitions of varying number of regions, it would also be desirable to explore the upper-bound Optimal Dataset Scale (ubODS) from sweeping a varied range of number of regions.

To do so, we add the following constraint, that forces the result to have a specific number of regions N: $\sum_i p_i = N - 1$, and sweep all the values of N between 1 and n.

5 Experiments

We compare the upper-bound partition selection technique against the merging-sequence partition analysis on four different hierarchies. The first one is the Ultrametric Contour Map (UCM) tree [5]. Then, two different BPT: the Normalized Weighted Euclidean distance between Models with Contour complexity (NWMC) tree [8], and the Independent Identically Distributed - Kullback Leibler (IID-KL) tree [9]. As a baseline we use a randomly-generated tree (Random), that is, a tree that is formed by iteratively merging random pairs of neighboring regions.

The trees are built on the 200 test images of the BSDS500 [5]. Each tree is compared with each of the multiple ground-truth partitions available and the result averaged, as proposed by [11] to handle multiple-partition ground truths. In order for the comparison to be fair, the base partition P_0 on which the tree is built is the same for the four techniques: the one obtained with the UCM with 100 regions.

The upper-bound partition selection algorithm is implemented in MATLAB, publicly available at `https://imatge.upc.edu/web/?q=node/1352`. The optimization itself of the LFCO is done by the IBM ILOG CPLEX Optimizer (free of charge for academic use), which is called directly by the MATLAB code. The scripts to fully reproduce the experiments and figures of this paper are also released.

In turn, the boundary matching code used in all the experiments has been obtained from [12]. Note that, this original code represents the boundaries of a partition in the *pixel grid*, that is, as a mask in which the pixels swept by the boundaries moved half pixel up and left are activated, which leads to an ambiguous representation. This ambiguity can be solved using the *contour grid* [13], which we use in our code. The numerical impact of this change of representation is not significant but the code obtained is much simpler and more readable.

5.1 ODS and OIS

Table 1 shows the mean Optimal Dataset Scale (ODS) and Optimal Image Scale (OIS) F_b values for the 200 image ground-truth pairs, and for the four compared hierarchies. The two first columns refer to the merging-sequence partition selection technique and the two last columns show the values for the upper-bound partition selection technique.

Comparing the quality of the hierarchies, the UCM tree presents better results than the rest of hierarchies. However, the main objective of this paper is not to compare the hierarchies themselves, but the partition selection techniques on which the assessment is based.

Regarding the comparison between ODS and OIS, the latter is coherently higher than the former. An improvement is also observed between the merging-sequence and the upper-bound techniques, which is, again, coherent with the theory.

Moreover, what really makes the difference between comparison techniques is their relative values, that is, how well the assessment discriminates the quality between the different hierarchies. In particular, good assessment techniques should be able to correctly discriminate between a random tree and the other techniques. To evaluate this aspect, Table 2 shows the relative values of ODS and OIS, that is, assigning 0 to the random tree, 1 to UCM, and scaling the rest of the values accordingly.

	Merging sequence ODS	OIS	Upper bound ubODS	ubOIS
UCM	0.587	0.622	0.669	0.695
NWMC	0.542	0.581	0.658	0.684
IID-KL	0.538	0.571	0.634	0.654
Random	0.523	0.537	0.589	0.603

Table 1. ODS and OIS F_b values for the merging-sequence and upper-bound partition selection techniques

	Merging sequence ODS	OIS	Upper bound ubODS	ubOIS
UCM	1	1	1	1
NWMC	0.30	0.51	0.86	0.89
IID-KL	0.23	0.40	0.56	0.56
Random	0	0	0	0

Table 2. Relative ODS and OIS F_b values for the merging-sequence and upper-bound partition selection techniques

If the measurement techniques were equivalent, the relative values should not change, but there are significant differences in these relative values, meaning that the conclusions extracted from the assessment can vary depending on the criterion used.

As introduced previously, a desirable property of the assessment techniques is a high discrimination of the random tree. In other words, it is obvious that the random trees must be far away from any *real* hierarchy. Under this point of view, the upper-bound assessment provides much better behavior.

As an example, the IID-KL tree is much closer to the random tree than to the UCM tree for ODS, while for the upper-bound ODS (ubODS), the IID-KL tree is halfway between the two, which is qualitatively more accurate.

The improvement obtained in the OIS with respect to the ODS highlights the relevance of the alignment algorithm on the results obtained. The same way, the improvement of the upper-bound analysis ubODS and ubOIS with respect to ODS and OIS is an indicator of the impact of the partition selection algorithm on the assessment.

Focusing on the sorting of the algorithm quality for the two top-rated hierarchies (UCM and NWMC), in 529 of the 1800 cases studied (9 parameterizations on 200 images), the ranking provided by the merging sequence analysis is not coherent with the one provided by the upper-bound. In other words, different partition selection strategies can lead to different decisions with respect to which is the best hierarchy based on a supervised assessment.

5.2 Upper-bound precision-recall curves

A region-based hierarchy is a structured set of image partitions at different scales, and thus comparing them to non-hierarchical partitions via the OIS and ODS F_b may obviate the assessment of some parts of the tree. The precision recall curves on boundary detection, instead, can give us a global picture of the quality of the hierarchy, sweeping the partitions at different scales.

Figure 3 shows the precision-recall curves for the four hierarchies studied. In the figure of the left, the points have been obtained using the merging-sequence partition selection whereas, in the figure of the right, the proposed upper-bound partition selection has been used.

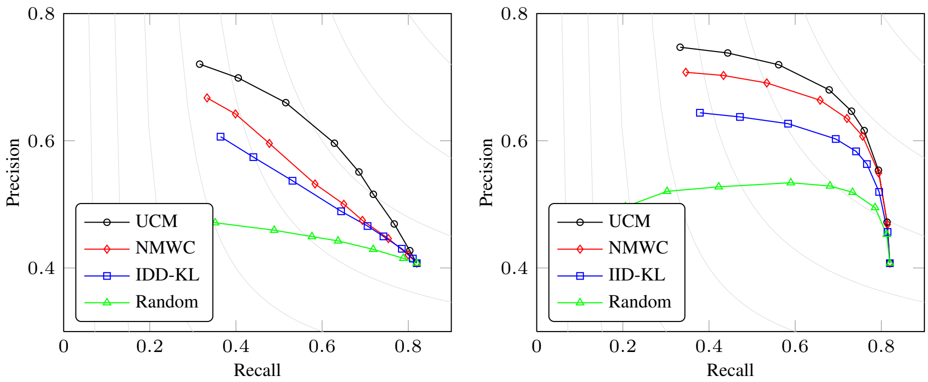

Fig. 3. Merging-sequence (left) and upper-bound (right) precision-recall curves

In the range of interest of the hierarchies, that is, the range of better F_b, similarly to the results of the previous section, the upper-bound precision-recall curves better discriminate between the random hierarchy and the rest of trees. In the range of higher number of regions, close to the leaves of the tree, the different curves are much closer than in the merging sequence, which reflects that, in this range, the original partition is more influent than the hierarchy itself. Note that, coherently, all curves meet in the point corresponding to 100 regions, because each tree contains only one partition with the maximum level of detail: P_0.

To better visualize the differences between the precision-recall curves and their upper-bound equivalents, Figure 4 shows them both in the same axis, leaving the IDD-KL tree out for the sake of clarity of the plot.

Note that, for each type of hierarchy, both curves start from the same point at high number of regions and tend to converge for few regions. In the middle range, corresponding also to the better F_b values, the gain obtained with the upper-bound assessment is much more relevant for the NWMC tree than for the rest of trees, which reinforces the possibility that different partition selection techniques can lead to discrepant results.

5.3 Computational cost

Although a supervised assessment is usually performed offline, a reduced computational cost is of paramount importance. The faster the method is, the larger the datasets in which researchers will tune and test their algorithms, which results in a more solid research. This section compares the computational cost of the proposed evaluation techniques (ubODS and ubOIS) in front of ODS and OIS.

ODS requires computing the boundary matching for k different number of regions, thus the whole time is k times the cost of a boundary matching, which we will refer to as $t(BM)$. OIS requires the ODS computation and find the F_b maximum for each image, so the cost is also $k \cdot t(BM)$. The cost of ubODS is one boundary multimatching $t(BMM)$ and k LFCO optimizations $t(LFCO)$. Finally, the cost of ubOIS

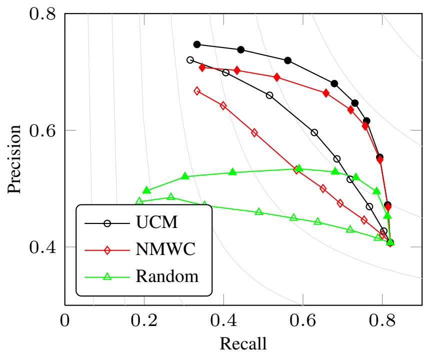

Fig. 4. Merging sequence (unfilled markers) versus upper-bound (filled markers)

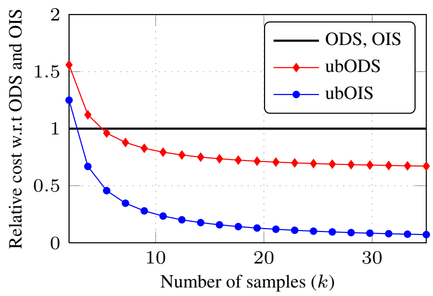

Fig. 5. Computational cost analysis for varying number of samples (k)

is $t(BMM) + t(LFCO)$, since one single optimization finds the optimal number of regions, thus not needing the computation of ubODS.

The mean values obtained in the experiments for each of these processes are: $t(BM) = 0.34\,s$, $t(BMM) = 0.64\,s$, and $t(LFCO) = 0.21s$. Thus, the mean time spent for each technique is $t(ODS) = t(OIS) = k \cdot 0.34\,s$, $t(ubODS) = 0.64 + k \cdot 0.21\,s$, and $t(ubOIS) = 0.85\,s$. Figure 5 shows the relative cost of the upper-bound techniques with respect to the cost of the merging-sequence ones. The higher the number of samples k, the lower the relative cost of the upper-bound techniques. The computation of ubODS is approximately 25% faster than ODS and OIS, while ubOIS, thanks to the fact that a single optimization is enough, is considerably faster.

5.4 Worst-discrepancy graphical results

To get a qualitative idea of the type of discrepancies between the region selection techniques, Figure 6 shows the most discrepant example of partition selection on the UCM tree for 6, 10, and 20 regions selected, that is, the three results whose partition selected in the merging sequence is more dissimilar with the upper-bound one.

The differences observed between the two strategies are visually relevant. In the first and second columns (6 and 10 regions), the merging sequence analysis obviates the main object of interest or part of it, while in the upper-bound partition selection the object is present in the selected partition. In the last column (20 regions) the upper-bound selection is capable of highlighting the higher importance of the background object with respect to the background as in the ground truth.

To sum up, the upper-bound partitions (Figure 6.d) represent the quality of the tree much better than those of the merging sequence (Figure 6.c), or in other words, the region selection masks the actual quality of the tree.

Fig. 6. Worst-case results on UCM trees: (a) images of the BSDS500 test set, (b) their multiple ground-truth partitions, (c) partitions selected from the merging sequence with 6, 10, and 20 regions, respectively, and (d) the upper-bound partitions with the same number of regions

6 Conclusions

This paper presents the upper-bound partition selection algorithm as a supervised assessment of hierarchical region-based image representations. It consists in finding, among all possible partitions represented in the hierarchy, the partition that best match the ground truth, instead of assessing just a reduced set of representative partitions.

The quality assessment measure used is the so-called F measure for boundary detection (F_b), which is known to present a good behavior among the existing measures. To be able to efficiently analyze all possible partitions in a hierarchy, we model the problem as a Linear Fractional Combinatorial Optimization (LFCO) problem.

The experiments show that the ubODS and ubOIS assessment techniques better represent the quality of the tree: 1) they cover partitions that are omitted in the merging sequence (and are used in image analysis works) and reach much better F_b values, 2) their performance discrimination between the random and the actual techniques is much better. Some visual examples corroborate that the merging-sequence selected partitions are not good representatives of the quality of the tress. Overall, an assessment based on the previous techniques in the literature can mislead the conclusions that can be extracted.

We make the MATLAB code to compute the ubODS and ubOIS publicly available, as well as all the scripts to fully reproduce the experiments and figures of this paper.

References

1. Salembier, P., Garrido, L.: Binary partition tree as an efficient representation for image processing, segmentation, and information retrieval. IEEE Transactions on Image Processing **9**(4) (2000) 561–576
2. Huihai, L., Woods, J.C., Ghanbari, M.: Binary partition tree analysis based on region evolution and its application to tree simplification. IEEE Transactions on Image Processing **16**(4) (2007) 1131–1138
3. Cardelino, J., Caselles, V., Bertalmio, M., Randall, G.: A contrario hierarchical image segmentation. In: IEEE International Conference on Image Processing. (2009) 4041 –4044
4. McGuinness, K., O'connor: A comparative evaluation of interactive segmentation algorithms. Pattern Recognition **43** (2010) 434–444
5. Arbeláez, P., Maire, M., Fowlkes, C.C., Malik, J.: Contour detection and hierarchical image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence **33**(5) (2011) 898–916
6. Pont-Tuset, J., Marques, F.: Contour detection using binary partition trees. In: IEEE International Conference on Image Processing. (2010) 1609–1612
7. Hunter, G.M., Steiglitz, K.: Operations on images using quad trees. IEEE Transactions on Pattern Analysis and Machine Intelligence **1**(2) (april 1979) 145 –153
8. Vilaplana, V., Marques, F., Salembier, P.: Binary partition trees for object detection. IEEE Transactions on Image Processing **17**(11) (2008) 2201–2216
9. Calderero, F., Marques, F.: Region merging techniques using information theory statistical measures. IEEE Transactions on Image Processing **19**(6) (2010) 1567–1586
10. Radzik, T.: Newton's method for fractional combinatorial optimization. In: Symposium on Foundations of Computer Science. (1992) 659–669
11. Unnikrishnan, R., Pantofaru, C., Hebert, M.: Toward objective evaluation of image segmentation algorithms. IEEE Transactions on Pattern Analysis and Machine Intelligence **29**(6) (2007) 929–944
12. Martin, D., Fowlkes, C., Malik, J.: Learning to detect natural image boundaries using local brightness, color, and texture cues. IEEE Transactions on Pattern Analysis and Machine Intelligence **26**(5) (2004) 530–549
13. Nunes, P., Marques, F., Pereira, F., Gasull, A.: A contour-based approach to binary shape coding using a multiple grid chain code. Signal Processing: Image Communication **15**(7-8) (2000) 585–599