

Exploring Hierarchy Simplification for Non-Significant Region Removal

Isabela Borlido Barcelos, Gabriel Barbosa da Fonseca, Laurent Najman, Yukiko Kenmochi, Benjamin Perret, Jean Cousty, Zenilton Do Patrocinio, Silvio Jamil F. Guimaraes

▶ **To cite this version:**

Isabela Borlido Barcelos, Gabriel Barbosa da Fonseca, Laurent Najman, Yukiko Kenmochi, Benjamin Perret, et al.. Exploring Hierarchy Simplification for Non-Significant Region Removal. 2019 32nd SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI), Oct 2019, Rio de Janeiro, Brazil. pp.100-107, 10.1109/SIBGRAPI.2019.00022 . hal-03132023

HAL Id: hal-03132023
https://hal.science/hal-03132023

Submitted on 4 Feb 2021

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

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

Exploring hierarchy simplification for non-significant region removal

Isabela Borlido Barcelos†, Gabriel Barbosa da Fonseca†, Laurent Najman*, Yukiko Kenmochi*, Benjamin Perret*, Jean Cousty*, Zenilton K. G. do Patrocínio Jr.† and Silvio Jamil F. Guimarães†

*Université Paris-Est, LIGM UMR 8049, UPEMLV
ESIEE Paris, ENPC, CNRS, F-93162 Noisy-le-Grand France
†Audio-Visual Processing Laboratory (VIPLAB)
Pontifical Catholic University of Minas Gerais, Brazil, 31980–110
E-mail: sjamil@pucminas.br

***Abstract*—Image segmentation is a classic subject in the field of digital image processing, and it can be used to solve a large variety of problems or serve as preprocessing for other methods of image analysis. Hierarchical image segmentation methods provide a multiscale representation, therefore they produce a nested set of image segmentations in which a result at a given level can be produced by merging regions of the segmentation at its previous level. However, a hierarchical representation may produce small components at its coarser levels, leading to oversegmentations on such scales. To solve this problem, we explore strategies to simplify hierarchies in order to remove non-significant regions, in terms of area, while trying to preserve the hierarchical structure. We evaluate the proposed simplification strategies with different hierarchical segmentation methods on the Pascal Context dataset by using precision-recall measures and fragmentation curves, along with a qualitative assessment showing that the simplification of hierarchies can lead to visually better image segmentations.**

I. Introduction

Image segmentation is one of the most elemental topics on image processing and analysis. The task itself consists in partitioning (or grouping) the pixels of an image into perceptually coherent parts. It can be used as preprocessing for other high-level tasks of image analysis [1], [2], or used as main method in various areas such as sedimentary petrography [3] and medical image analysis [4]–[7]. An image segmentation can be simplistically given as a single set of grouped pixels of an image, but many consider that this task is in fact a multi-scale problem, where the resulting segmentation can have different levels of refinement. Aiming at working with different scales of a segmentation, hierarchical methods are proposed.

A hierarchical image segmentation is a series of image segmentations at different detail levels, such that the segmentation at a given level can be produced by merging regions of the segmentation at its previous level. Usually, the pipeline for segmenting an image in a hierarchical way can be outlined as: (i) transformation of the image into a graph (*e.g.*, 4-adjacent graph); (ii) computation of a hierarchy from the graph (*e.g.*, quasi-flat zone hierarchy (QFZ) [8]); and (iii) computation of the segmentation from the hierarchy according to a given criterion (*e.g.*, number of regions). As stated in [9], [10], the saliency map of a hierarchy is represented in the 2D Khalimsky grid, in which the brightness of a contour is inversely proportional to the number of partitions of the hierarchy this contour belongs to, *i.e.*, dark contours are the strongest ones. Unfortunately, as it can be seen in Figure 1, the QFZ hierarchy computed from a graph that represents the original image has the majority of its strong contours representing very small regions, and using this hierarchy as it is can lead to an oversegmentation.

There are several hierarchical segmentation methods in the literature that use some kind of simplification, such as the HGB [11], HSRG [12] and HPRT [13], but differently from this work, they have used just one order of edges for traversing the tree representation. In a certain sense, the hierarchy simplification tries to remove small connected components as in the area opening [14] and the hierarchical simplification proposed by [15]. The idea of area opening, and its dual operation, called area closing, is to remove from a binary image its connected components with area smaller than a given threshold. This strategy is easily extended to grayscale

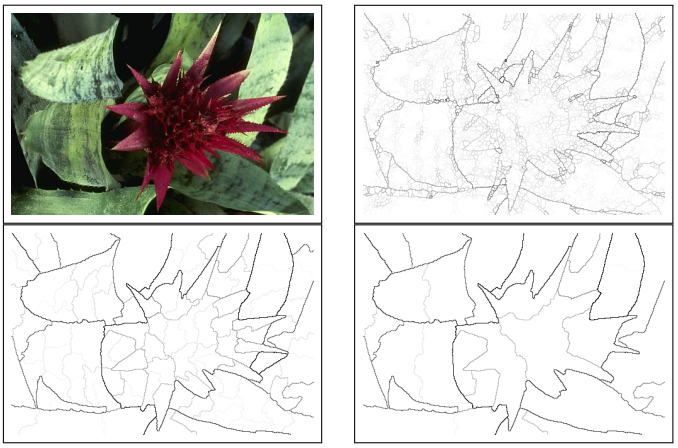


Fig. 1. An example of hierarchy simplification with two different traversal strategies by removing regions smaller than 1000 pixels. In first row, the QFZ hierarchy (right) of the image (left) is presented. In the second row, edges are browsed in a non-increasing (left) and non-decreasing (right) order of their weight.

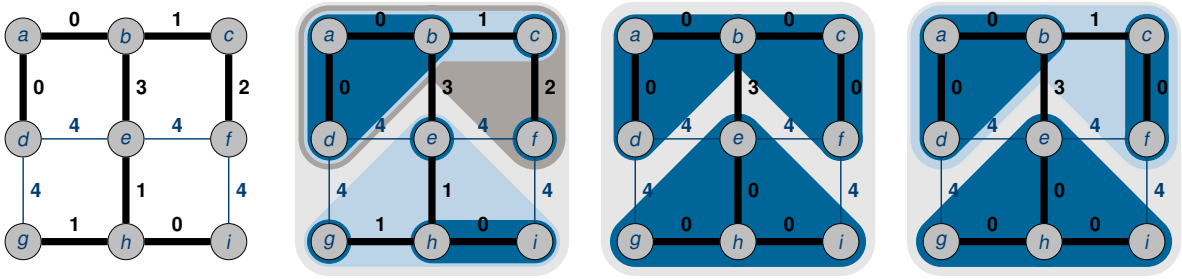


Fig. 2. An example of hierarchy simplification in terms of region size, in which, the minimum region size is 2 elements: (a) the original graph; (b) the initial hierarchy; (c)-(d) the simplified hierarchy taking into account the non-decreasing (non-increasing) order of edges for their weights. In (a)-(d), the black-solid edges represent the MST.

image in which the images can be seen as a stack of binary images. For the hierarchy simplification for non-significant region removal, the idea is to filter out small components, on a graph representation, by merging them to their adjacent components. The simplification proposed by [15] is based on the minimization of an energy functional. The basic idea of this method is to simplify the image taking into account stacked level lines considering their "importance" while preserving the protruding structures intact.

In this work, we explore several strategies to simplify hierarchies in order to remove, in terms of area, non-significant regions. These strategies consist on applying a simplification criteria over a tree representation of a hierarchy, performing a single traversal per simplification. This leads to a simple, but still effective hierarchy simplification method. It is showed that oversegmentation can be avoided by using the proposed simplification strategies, producing image segmentations that are visually better. Additionally, depending on the used strategy, it is possible to obtain segmentations with coarser or finer regions. This observation can be seen on a toy example illustrated in Figure 2, in which there are no regions with area smaller than 2 after the hierarchy simplification. Moreover, we study the behavior of the simplification on the HGB [11] and QFZ hierarchy [8] hierarchical segmentation methods, following the supervised assessment framework proposed in [16], [17]. In summary, the main goal of this work is to simplify a hierarchy without computing the segmentation beforehand. In other words, instead of computing the segmentations and then applying any kind of simplification, we simplify the hierarchy in order to remove the non-relevant regions, and the segmentation without these regions can be easily inferred. Moreover, we propose a general method for non-significant region removal that can be used to any hierarchical segmentation which is represented by a binary partition tree [18].

This paper is organized as follows. In Section II, notions of hierarchies are given. In Section III, we present the method for hierarchy simplification in order to remove non-significant regions. In Section IV, we evaluate the proposed method in terms of quantitative and qualitative assessments. And, finally, some conclusions and future works are drawn in Section V.

II. Hierarchical graph-based image segmentation

This section aims at briefly explaining the method of hierarchical graph-based image segmentation (HGB) [11]. However, we first give a series of necessary notions such as quasi-flat zones hierarchies [8], and then describe the HGB method.

A. Hierarchies

Given a finite set V, a *partition* of V is a set $\mathbf{P}$ of non-empty disjoint subsets of V whose union is V. Any element of $\mathbf{P}$ is called a *region* of $\mathbf{P}$. Given two partitions $\mathbf{P}$ and $\mathbf{P}'$ of V, $\mathbf{P}'$ is said to be a refinement of $\mathbf{P}$, denoted by $\mathbf{P}' \preceq \mathbf{P}$, if any region of $\mathbf{P}'$ is included in a region of $\mathbf{P}$. A hierarchy on V is a sequence $\mathcal{H} = (\mathbf{P}_0, \ldots, \mathbf{P}_\ell)$ of partitions of V, such that $\mathbf{P}_{i-1} \preceq \mathbf{P}_i$, for any $i \in \{1, \ldots, \ell\}$.

B. Graph and connected-component partition

A *graph* is a pair $G = (V, E)$ where V is a finite set and E is a subset of $\{\{x, y\} \subseteq V | x \neq y\}$. Each element of V is called a *vertex* of G, and each element of E is called an *edge* of G. A subgraph of G is a graph (V', E') such that $V' \subseteq V$ and $E' \subseteq E$. If X is a graph, its vertex and edge sets are denoted by $V(X)$ and $E(X)$, respectively.

If two vertices of a graph G are joined by an edge, we say that they are adjacent. From the reflexive–transitive closure of this adjacency relation on a finite set $V(G)$, we derive the connectivity relation on $V(X)$. It is an equivalence relation, whose equivalence classes are called connected components of G. We denote by $\mathbf{C}(G)$ the set of all connected components of G. Note that $\mathbf{C}(G)$ is a partition of $V(G)$, called the *connected-component partition induced by* G.

C. Quasi-flat zone hierarchies

Given a graph $G = (V, E)$, let w be a map from E into the set $\mathbb{R}$ of real numbers. For any edge u of G, the value $w(u)$ is called the weight of u (for w), and the pair (G, w) is called an *edge-weighted graph*. We now make from an edge-weighted graph a series of connected-component partitions, which constitutes a hierarchy. Such a hierarchy is called a quasi-flat zone hierarchy (QFZ) of (G, w), and the quasi-flat zone hierarchy transform is a bijection between the hierarchies and a subset of the edge-weighted graphs called the saliency

maps [8]. Hence, any edge-weighted graph induces a quasi-flat zone hierarchy and any hierarchy $\mathcal{H}$ can be represented by an edge-weighted graph whose quasi-flat zone hierarchy is precisely $\mathcal{H}$ [8]. This bijection allows us to handle quasi-flat zone hierarchies through edge-weighted graphs.

Given an edge-weighted graph (G, w), let X be a subgraph of G and let λ be a value of $\mathbb{R}$. The *λ-level edge set* of X for w is defined by $w_\lambda(X) = \{u \in E(X) \mid w(u) < \lambda\}$, and the *$\lambda$-level graph* of X for w is defined as the subgraph $w^V_\lambda(X)$ of X such that $w^V_\lambda(X) = (V(X), w_\lambda(X))$. Then, the connected-component partition $\mathbf{C}(w^V_\lambda(X))$ induced by $w^V_\lambda(X)$ is called the *λ-level partition* of X for w.

As we consider only finite graphs and hierarchies, the set of considered level values is reduced to a finite subset of $\mathbb{R}$ that is denoted by $\mathbb{E}$ in the remaining parts of this article. In order to browse the values of this set and to round real values to values of $\mathbb{E}$, we define, for any $\lambda \in \mathbb{R}$: $\mathrm{p}_\mathbb{E}(\lambda) = \max\{\mu \in \mathbb{E} \cup \{-\infty\} \mid \mu < \lambda\}$ and $\mathrm{n}_\mathbb{E}(\lambda) = \min\{\mu \in \mathbb{E} \cup \{\infty\} \mid \mu > \lambda\}$.

Let (G, w) be an edge-weighted graph and let X be a subgraph of G. The sequence of all λ-level partitions of X for w, ordered by increasing value of λ, is a hierarchy, defined by $QFZ(X, w) = (\mathbf{C}(w^V_\lambda(X)) \mid \lambda \in \mathbb{E} \cup \{\infty\})$, and called the *quasi-flat zone hierarchy of X for w*. Let $\mathcal{H}$ be the quasi-flat zone hierarchy of G for w. Given a vertex x of G and a value λ in $\mathbb{E}$, the region that contains x in the λ-level partition of the graph G is denoted by $\mathcal{H}^\lambda_x$.

Let us consider a minimum spanning tree T of (G, w). It has been shown in [8] that $QFZ(T, w)$ of T for w is the same as $QFZ(G, w)$ of G for w. This indicates that the quasi-flat zone hierarchy for G can be handled by its minimum spanning tree.

D. HGB method

The method HGB [11] does not explicitly produce a hierarchy of partitions, but instead it produces a weight map w' (scales of observations) from which the desired hierarchy $\mathcal{H}$ (for w') can be inferred on a given T of (G, w). It starts from a minimum spanning tree T of an edge-weighted graph built from the image. In order to compute the scale $L(u = \{x, y\})$ associated with each edge of T, the HGB method iteratively considers the edges of T in a non-decreasing order of their original weights w.

The hierarchical scale $w'(u)$ is simply set to:

$$w'(u) = \max\{S_Y(X), S_X(Y)\}, \tag{1}$$

in which,

$$S_Y(X) = [Dif(X, Y) - Int(X)] \times |X| \tag{2}$$

with $Dif(X, Y)$ and $Int(X)$ defined analogously to [19]. Thus, the *internal difference* $Int(X)$ of a region X is the highest edge weight among all the edges linking two vertices of X in the minimum spanning tree of (G, w); and the *difference* $Dif(X, Y)$ between two neighboring regions X and Y is the smallest edge weight among all the edges that link X to Y. Additional details were omitted and the reader should refer to [11] for more information.

III. Hierarchy simplification

The seminal work of the hierarchy simplification was proposed in [11]. Differently from that proposed method, here we explored two strategies to organize the edges according to their weights: (i) non-increasing order; and (ii) non-decreasing order. The main idea for this simplification is to transform an initial hierarchy into a new one in which: (i) the new hierarchy does not contain any region with area below a given threshold; and (ii) the regions of the new hierarchy are either regions of the initial hierarchies or regions obtained by merging adjacent regions of the initial hierarchy. In other words, if a region R of the initial hierarchy is not adjacent to other regions with area smaller than the threshold, this region will be in the simplified hierarchy, but if an adjacent region S does not reach the threshold, the region S will be merged to R or to another adjacent region to S.

In order to cope with such transformation, the hierarchies, which are represented by weight maps (see [8] for more details), are manipulated as binary partition trees since the computation of the area for a region in this representation can efficiently be done. The main idea of the Algorithm 1 is to identify the edges that merge two connected components in which at least one of them has area smaller than a given threshold (*line 4*). If so, the new edge weight will be set to zero (*line 5*) since both connected components must be merged together instead of being separated.

As one can see, our method does not explicitly produce a hierarchy of partitions, but instead it produces a new weight map from which the desired hierarchy is simply the QFZ hierarchy on T with the new edge weights. Generally, it starts from a minimum spanning tree T of an edge-weighted graph built from the image. In order to compute the weight associated with each edge of T, our method iteratively considers the edges of T in a given order of their original weights w. For every edge u, the new weight is either the original weight or zero.

Algorithm 1: Hierarchy simplification for non-significant region removal.

Data: A minimum spanning tree $T = (V, E')$ of w
Data: An array A_{MST} with the edges of the MST T in a given order of weights with respect to w
Data: A minimum area size M
Result: A map w' from E' to $\mathbb{R}^+$

1 $w' \leftarrow w$;
2 **for** *i ← 1 to* $|E'|$ **do**
  `/* Assuming u = {x,y} as an edge */`
3 $u = A_{MST}[i]$;
4 **if** $|[\mathbf{C}(w'^V_{f(u)}(T))]_x| \geq M$ *and* $|[\mathbf{C}(w'^V_{f(u)}(T))]_y| \geq M$ **then** $w'(u) \leftarrow w(u)$;
5 **else** $w'(u) \leftarrow 0$;



(a) Original image (b) HGB hierarchy (c) Non-increasing order (d) Non-decreasing order

Fig. 3. Comparison between HGB method (non-simplified) (b) with simplification by using non-increasing order of the edge according to their weights (c) with simplification by using non-decreasing order of the edge according to their weights (d). On lines, from top to bottom, the simplifications have threshold of 400, 800, 400, and 500 pixels and the number of components on the segmentations with simplification is 13, 24, 22, and 20 respectively. On segmentations non-simplified, the number of components is 16, 29, 28, and 22.

An example of a hierarchy simplification is given in Figure 2. This example is a bit tricky as there are several valid orderings. The initial hierarchy (Figure 2(b)), which is computed from the original graph, is simplified in order to remove all regions with area smaller than 2. The *4-level partition* of T for w contains 2 (resp, 3) regions when the edges are considered in a non-decreasing (resp, non-increasing) order, illustrated in Figure 2(c) (resp, (d)).

By using this hierarchy simplification method, we can argue that when the edges are taken according to a non-decreasing order of their weights, the simplified hierarchy has coarser regions when compared to the simplification according to non-increasing order that has finer regions. This observation can be easily seen in Figure 3 and Figure 4.

IV. Experimental Results

This section presents the qualitative and quantitative assessments of the results obtained with the non-increasing and non-decreasing order simplifications.

A. Qualitative assessment

In this section we present a qualitative evaluation of the hierarchical simplifications. The images used on this assessment are from the BSDS 500 test subset [10]. As a first experiment, shown in Figure 3, we computed the HGB segmentation without simplification and with a simplification taking into account the non-increasing order, hereafter NI for short, and with a simplification taking into account the non-decreasing order, hereafter ND for short, using the same simplification threshold and number of components on the segmentation for each image. As a second experiment, shown in Figure 4, there are two QFZ segmentations of the same image with the NI and ND simplifications, using three different levels of area threshold and showing the lowest λ-level segmentation (the one corresponding to the set with the largest number of components). For the third batch of experiments, shown



(a) Original image (b) QFZ hierarchy (c) Threshold 500 (d) Threshold 1000 (e) Threshold 1500

Fig. 4. Comparison of simplifications by using non-decreasing and non-increasing orders from QFZ hierarchy. Non-increasing order is used on the first line and non-decreasing order on the second. All the slices presented are of the lowest λ-level segmentation. In the columns, from left to right, we have the original image, the QFZ hierarchy and the simplifications with a threshold of 500, 1000, and 1500 pixels, respectively.

on Figure 5, we paired the two hierarchical segmentations computed from HGB and QFZ hierarchies, showing the non-simplified saliency map, the NI and ND simplified saliency maps, and the NI and ND segmentations with same number of components for each image.

On Figure 3, we can observe the effect of the hierarchy simplification by analyzing the saliency maps and segmentations. Without using one of the simplification methods, the produced hierarchies contain numerous small components which belong to the same object on the image. This is not desirable, since finding a level on such hierarchies that would not produce an oversegmented image can be quite hard. After applying the simplification, hopefully the small components that belong to a same object are merged, and creating segmentations in the new hierarchies is eased. It can be observed in these images that, when using the same number of components, a segmentation done after the simplification represents better the original images.

On Figure 4, it can be seen that for both simplifications, using higher thresholds lead to smaller number of regions with larger sizes, since regions with a size (in number of pixels) smaller than the threshold are merged to the one closest in the hierarchy (*i.e.*, more similar). Taking into account that the union between regions is done directly in the hierarchy, the change of the threshold preserves the hierarchical characteristics of the segmentation. One can also see in Figure 4 that, for all the thresholds used, the NI simplification produces more refined regions than those from the ND simplification, in addition to containing a greater amount of regions. This behaviour is highly related to the nature of the NI and ND order of evaluation. The NI simplification starts evaluating the larger components of a hierarchy, and it continues in non-increasing order until all elements of the hierarchy are evaluated. With that in mind and considering that merged elements are not checked twice, given the same hierarchy, the NI simplification performs less merges of bigger components when compared to the ND simplification, leading to fewer component fusions, and consequently smaller components on the lowest λ-level segmentation.

On Figure 5, we can compare the results produced by both hierarchical segmentation methods, and how they are impacted by the simplifications. For both segmentation methods, the non-simplified hierarchies contain many small objects, and although the HGB produces slightly better segmentations without hierarchy simplification (it partially separates the boat and the person from the background on Figures 5(b) and 5(c), while the QFZ does not), after the simplification the two methods produce similar segmentations. As already stated, the NI simplification can result on more refined segmentations compared to ND simplification. The choice for which variation to use is situational, as we can observe on the QFZ segmentations of Figures 5(a) and 5(b). On Figure 5(a),

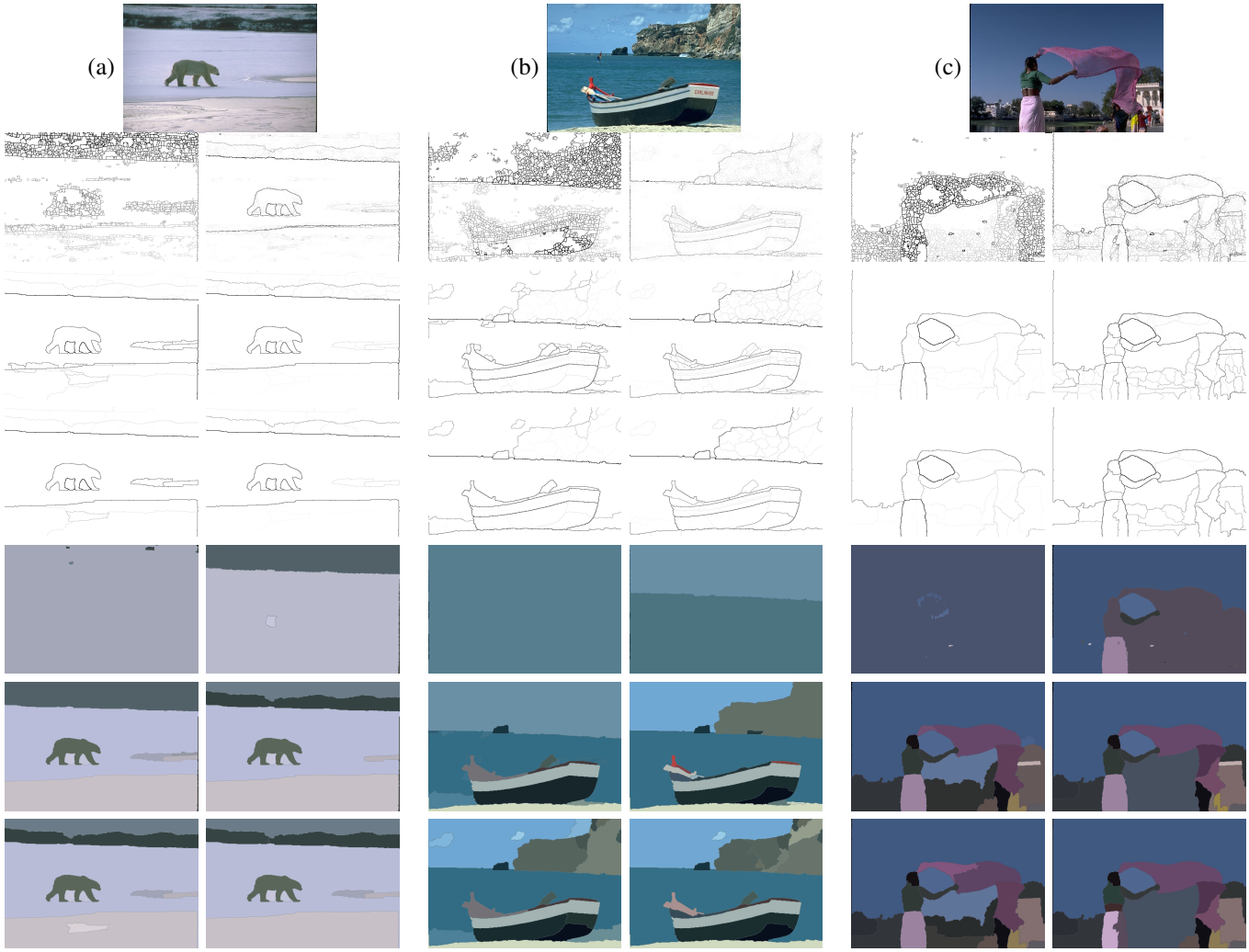


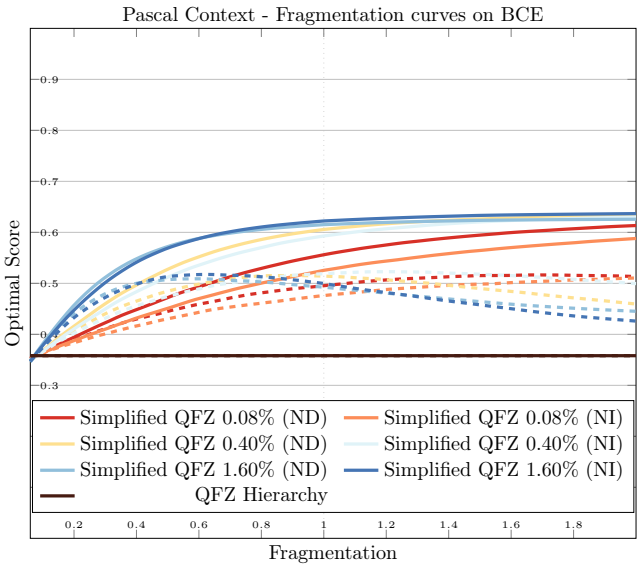
Fig. 5. Saliency maps and segmentations for each segmentation and simplification method. For each original image (on top), there are two columns regarding the HGB and QFZ segmentation methods on the left and right respectively. On each column, the first image (from top to bottom) is the saliency map without simplification, followed by the NI and ND simplified saliency maps. Following, there are HGB and QFZ segmentation without simplification and segmentations for the NI and ND simplifications with the same number of regions for each original image used (in NI and ND). The threshold used in simplifications are 400, 200, and 500 pixels and the number of components on the segmentations is 10, 22 and 20 for images (a), (b), and (c) respectively. On the non-simplified segmentations, the number of components is 11, 24, and 25 (HGB); and 10, 21, and 19 (QFZ), respectively.

the bear is segmented in two regions when using the QFZ with NI simplification, which does not happen with the ND counterpart. On Figure 5(b), the NI simplification results on a segmentation where the fine details of the boat are preserved, while the ND merges these details and split the mountain into multiple regions.

At the saliency maps with NI and ND simplifications, Figures 5(b) (QFZ and HGB) and 5(c) (QFZ only), we can see several small regions that are not at a low levels in the hierarchy. However, only the segmentations with NI and ND simplifications on 5(b) preserved these regions. This occurred due to the different λ-levels chosen for each method to enforce them to produce the same number of regions for the analysis. While in Figure 5(b) the connected-component partitions were induced by λ-levels of 14 (HGB) and 27 (QFZ), in Figure 5(c) the partitions were taken at the level 92 (QFZ). The difference in the levels of λ taken to induce a partition can also influence the results, since small λ-levels produce partitions in which separated components can have high similarities.

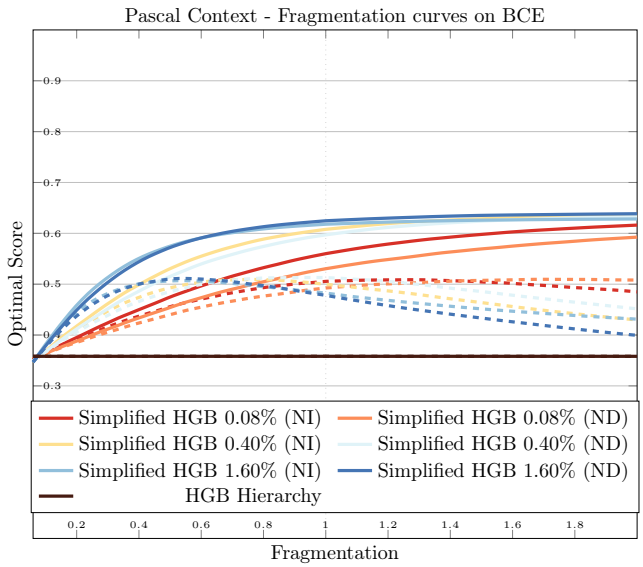
B. Quantitative assessment

In this section, a quantitative assessment of the proposed method and its variations is presented. The hierarchy simplification methods are evaluated on a dataset of natural image analysis, following the metrics of an image segmentation evaluation framework. The database, metrics and framework are detailed, and then the results for the QFZ [17] and HGB [11], [20] hierarchies are shown and compared to their simplified counterparts. For computing both methods, we have used Structured Edge [21] as gradient estimator. In both cases, we used the two different strategies for organizing the edges according to their weights: (i) non-decreasing order; and (ii) non-increasing order. In terms of area size, we remove the



	FB		BCE	
	ODS	OIS	FOC	FHC
Simplified QFZ 0.08% (ND)	0.537	0.588	0.533	0.478
Simplified QFZ 0.08% (NI)	0.520	0.574	0.508	0.464
Simplified QFZ 0.40% (ND)	**0.548**	0.590	0.571	0.480
Simplified QFZ 0.40% (NI)	0.545	**0.592**	0.562	**0.491**
Simplified QFZ 1.60% (ND)	0.531	0.556	0.586	0.474
Simplified QFZ 1.60% (NI)	0.543	0.578	**0.590**	0.472
QFZ Hierarchy	0.479	0.477	0.358	0.358

(a) QFZ



	FB		BCE	
	ODS	OIS	FOC	FHC
Simplified HGB 0.08% (NI)	0.408	0.408	0.536	0.476
Simplified HGB 0.08% (ND)	0.330	0.330	0.512	0.472
Simplified HGB 0.40% (NI)	0.480	0.479	0.573	0.468
Simplified HGB 0.40% (ND)	0.425	0.426	0.565	**0.477**
Simplified HGB 1.60% (NI)	**0.512**	**0.508**	0.588	0.466
Simplified HGB 1.60% (ND)	0.485	0.483	**0.592**	0.456
HGB Hierarchy	0.207	0.208	0.358	0.358

(b) HGB

Fig. 6. Comparison between the QFZ hierarchy and simplified QFZ hierarchy at a given threshold. **Fragmentation–Optimal Cut score curves (FOC)** for BCE on Pascal Context (second column): each plain curve represent the upper-bound score achievable for a given fragmentation value. The corresponding dashed curves represent the score obtained by horizontal cuts. The principal performance measures are summarized in the table: F-Measure of FB at ODS and OIS, area under the curve for BCE for optimal and horizontal cuts (fragmentation curves).

regions that area smaller than 0.08%, 0.40%, and 1.60% of the original size of the image. Thus, we have compared the initial hierarchy to these simplified versions.

Some aspects related to the evaluation of image segmentation in the form of hierarchies of partitions have to be taken in consideration for a proper assessment. Even though doing a qualitative analysis of a hierarchy can be eased by the use of contour maps such as the saliency maps obtained from the hierarchies, the quantitative analysis of hierarchies are more complicated. Since the image segmentation datasets do not present ground-truths for hierarchical segmentations, a common strategy is to perform cuts on different scales, *i.e.* all possible horizontal scales of the hierarchy, and evaluate the created segmentations with common image segmentation metrics. To perform the assessment, we follow the supervised assessment framework proposed in [17].

In this paper, we have used only fragmentation curves on the bidirectional-consistency error (BCE) [17]. The fragmentation level of a partition is defined as the number of regions in the partition divided by the number of regions in the ground-truth. The fragmentation curve on BCE then evaluates the quality of the regions of partitions of the hierarchy as the fragmentation level increases, also with respect to a ground-truth segmentation. We consider two categories of partitions that can be extracted from a hierarchy: the partitions of the hierarchy (horizontal cuts), and the optimal partitions that can be constructed from regions taken from any partition of the hierarchy (the optimal non-horizontal cuts). Two aggregated measures are defined: the area under the curve for the optimal cuts (FOC) and the area under the curve for horizontal cuts (FHC).

The fragmentation curves are evaluated on the test set of the Pascal Context dataset [22]. Pascal Context test set consists of a pixel-wise segmentation of the last 2,498 images of the Pascal VOC'10 [23] validation set.

Observing the results, one can conclude that the hierarchy simplifications do improve the results for both segmentation methods compared to their non-simplified versions, specially for the HGB method. Also, at lower thresholds, the NI variation curves usually outperform the ND curves, but as the threshold grows, the methods tend to have closer and closer results. This is expected, because as the thresholds get larger, some regions are merged regardless of the traversal strategy chosen.

V. Conclusion

In this paper we have explored two strategies to simplify hierarchies in order to remove, in terms of area, non-significant regions. We have shown that these studied strategies may obtain different results with different features. Generally, when the edges are considered in a non-decreasing order of their weights, the segmentation is coarser than the non-increasing order. This behaviour is highly related to the nature of the order of evaluation. The NI simplification starts evaluating the larger components of a hierarchy, and continues in non-increasing order until all elements of the hierarchy are evaluated. With that in mind and considering that merged elements are not checked twice, given the same hierarchy, the NI simplification performs less merges of bigger components when compared to the ND simplification, leading to fewer component fusions, and consequently smaller components on the lowest λ-level segmentation.

According to our experiments, the use of hierarchy simplification outperforms the initial hierarchy for the both studied methods, QFZ and HGB. Also, it is observed that the NI simplification outperforms the ND at lower thresholds, but as the threshold increases, the two simplifications produce more similar results. It is important to observe that the fragmentation measures are better when the minimum sized regions increase. For future works, we aim at studying how to combine both strategies for new hierarchy simplification, and how to create new traversal criteria to perform the hierarchical simplification.

Acknowledgments

The authors are grateful to CNPq (Universal 421521/2016-3 and PQ 307062/2016-3), FAPEMIG (PPM-00006-16), CAPES (Grant PVE 88887.125000/2014-00, COFECUB 88887.191730/2018-00) and PUC Minas for the financial support to this work. This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001.

References

[1] J. E. Vargas, P. T. M. Saito, A. X. Falcão, P. J. d. Rezende, and J. A. d. Santos, "Superpixel-based interactive classification of very high resolution images," in *2014 27th SIBGRAPI Conference on Graphics, Patterns and Images*, Aug 2014, pp. 173–179.

[2] M. F. Alcântara, T. P. Moreira, and H. Pedrini, "Real-time action recognition based on cumulative motion shapes," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2014, pp. 2917–2921.

[3] I. Mingireanov Filho, T. V. Spina, A. X. Falcão, and A. C. Vidal, "Segmentation of sandstone thin section images with separation of touching grains using optimum path forest operators," *Computers & geosciences*, vol. 57, pp. 146–157, 2013.

[4] J. Cousty, L. Najman, M. Couprie, S. Clément-Guinaudeau, T. Goissen, and J. Garot, "Segmentation of 4d cardiac mri: Automated method based on spatio-temporal watershed cuts," *Image and Vision Computing*, vol. 28, no. 8, pp. 1229–1243, 2010.

[5] F. Milletari, N. Navab, and S.-A. Ahmadi, "V-net: Fully convolutional neural networks for volumetric medical image segmentation," in *2016 Fourth International Conference on 3D Vision (3DV)*. IEEE, 2016, pp. 565–571.

[6] K. Bacchuwar, J. Cousty, R. Vaillant, and L. Najman, "Scale-space for empty catheter segmentation in pci fluoroscopic images," *International journal of computer assisted radiology and surgery*, vol. 12, no. 7, pp. 1179–1188, 2017.

[7] M. Drozdzal, G. Chartrand, E. Vorontsov, M. Shakeri, L. Di Jorio, A. Tang, A. Romero, Y. Bengio, C. Pal, and S. Kadoury, "Learning normalized inputs for iterative estimation in medical image segmentation," *Medical image analysis*, vol. 44, pp. 1–13, 2018.

[8] J. Cousty, L. Najman, Y. Kenmochi, and S. Guimarães, "Hierarchical segmentations with graphs: Quasi-flat zones, minimum spanning trees, and saliency maps," *Journal of Mathematical Imaging and Vision*, vol. 60, no. 4, pp. 479–502, 2018.

[9] L. Najman and M. Schmitt, "Geodesic saliency of watershed contours and hierarchical segmentation," *Transactions on Pattern Analysis and Machine Intelligence*, pp. 1163–1173, 1996.

[10] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik, "Contour detection and hierarchical image segmentation," *Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 5, pp. 898–916, 2011.

[11] S. Guimarães, Y. Kenmochi, J. Cousty, Z. Patrocinio Jr., and L. Najman, "Hierarchizing graph-based image segmentation algorithms relying on region dissimilarity - the case of the Felzenszwalb-Huttenlocher method," *Mathematical Morphology - Theory and Applications*, vol. 2, no. 1, pp. 55–75, 2017.

[12] S. J. F. Guimarães and J. Patrocínio, Zenilton K.G., "A graph-based hierarchical image segmentation method based on a statistical merging predicate," in *Image Analysis and Processing - ICIAP 2013*, ser. Lecture Notes in Computer Science, A. Petrosino, Ed. Springer Berlin Heidelberg, 2013, vol. 8156, pp. 11–20.

[13] S. J. F. Guimarães, Z. K. G. do Patrocínio, Y. Kenmochi, J. Cousty, and L. Najman, "Hierarchical image segmentation relying on a likelihood ratio test," in *Image Analysis and Processing — ICIAP 2015*, V. Murino and E. Puppo, Eds. Cham: Springer International Publishing, 2015, pp. 25–35.

[14] L. Vincent, "Grayscale area opening and closing, their efficient implementation and applications," in *in Proc. Eurasip Workshop on Mathematical Morphology and its Applications to Signal Processing*, 1993.

[15] Y. Xu, T. Géraud, and L. Najman, "Hierarchical image simplification and segmentation based on mumford–shah-salient level line selection," *Pattern Recognition Letters*, vol. 83, pp. 278–286, 2016.

[16] B. Perret, J. Cousty, J. C. R. Ura, and S. J. F. Guimarães, "Evaluation of morphological hierarchies for supervised segmentation," in *Mathematical Morphology and Its Applications to Signal and Image Processing*, J. A. Benediktsson, J. Chanussot, L. Najman, and H. Talbot, Eds. Springer International Publishing, 2015, pp. 39–50.

[17] B. Perret, J. Cousty, S. J. GuimarÃ£es, and D. S. Maia, "Evaluation of hierarchical watersheds," *IEEE Transactions on Image Processing*, vol. 27, no. 4, pp. 1676–1688, 2018.

[18] P. Salembier and L. Garrido, "Binary partition tree as an efficient representation for image processing, segmentation, and information retrieval," *IEEE Trans. Image Processing*, vol. 9, no. 4, pp. 561–576, 2000.

[19] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient graph-based image segmentation," *International Journal of Computer Vision*, vol. 59, no. 2, pp. 167–181, 2004.

[20] E. Cayllahua-Cahuina, J. Cousty, S. Guimarães, Y. Kenmochi, G. Cámara-Chávez, and A. de Albuquerque Araújo, "A study of observation scales based on felzenswalb-huttenlocher dissimilarity measure for hierarchical segmentation," in *Discrete Geometry for Computer Imagery*. Springer International Publishing, 2019, pp. 167–179.

[21] P. Dollár and Zitnick, "Fast edge detection using structured forests," *Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 8, pp. 1558–1570, 2015.

[22] R. Mottaghi, X. Chen, X. Liu, N.-G. Cho, S.-W. Lee, S. Fidler, R. Urtasun, and A. Yuille, "The role of context for object detection and semantic segmentation in the wild," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[23] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL Visual Object Classes Challenge 2010 (VOC2010) Results," http://www.pascal-network.org/challenges/VOC/voc2010/workshop/index.html.