Binary-Partition-Tree creation using a quasi-inclusion criterion

Christian Ferran Bennström and Josep R. Casas
Signal Theory and Communications Department.
Universitat Politécnica de Catalunya, Barcelona, Spain.
{cferran, josep}@gps.tsc.upc.es

Abstract

The aim of this paper is to present how structural relationships among regions can be used in the context of a general merging algorithm based on texture. In [1] and [2], the notions of merging order, merging model and merging criterion were introduced for the creation of Binary-Partition-Trees (BPTs). We extend this framework to syntactic features, representing geometric relationships between image regions, which are taken into account for the creation of the BPT. In particular, we perform structure analysis on the shapes and the spatial configuration of image regions. In order to test the syntactic features approach, a proof of concept has been carried out following a quasi-inclusion criterion.

1. Introduction

This paper[1] deals with the the introduction of geometric properties into the segmentation process. For this purpose, we assume that segmentation methods can be classified according to two main approaches, *top-down* (model-based) or *bottom-up* (visual feature-based). Top-down methods link complex visual concepts (semantics) to regions (low-level features) and bottom-up approaches infer meaningful objects (semantics) from regions. In the first case, geometry can help in the search of pre-defined object features; while in the second case, geometry can be used in the construction of high order primitives by grouping low-level primitives (i.e. regions). Usually top-down approaches suffer from application-dependent limitations and *bottom-up* approaches tend to increase in complexity when infering semantic objects from low-level image features.

Region-oriented merging algorithms can be classified into the bottom-up methods and are a convenient framework to investigate how to introduce geometric features in the segmentation process while keeping the complexity low.

In this framework relationships between image regions, such as homogeneity, compactness, regularity, inclusion or symmetry, are called syntactic features. Syntactic features can be found by structure analysis (or syntax analysis), see [3], and are based on shapes and spatial configuration of spatially homogeneous regions in the image.

In this paper, image regions are obtained in a first stage with a color-based segmentation and the syntactic segmentation is applied in a second stage, see Fig.1. We focus on a simple geometric problem –quasi-inclusion of neighboring regions– in order to establish and evaluate a new syntactic segmentation framework derived from the *General Merging Algorithm* presented in [1] and the *BPT merging scheme* exposed in [2].

Object representation based on structural properties is an important research area. For example, in [6], the convex-hull is used for the construction of a concavity tree representing the object. In this paper, we introduce a BPT representation based on a quasi-inclusion criterion.

The paper is organized as follows: Section 2 presents firstly the general merging algorithm (section 2.1), secondly, the color-based approach (section 2.2) and finally the extension to quasi-inclusion segmentation (section 2.3). In section 3, a Binary Partition Tree (BPT) is built allowing a region-oriented image representation based on the newly introduced criterion. Results are presented in section 4, and finally we will draw our conclusions and we will plan our future work.

2. Graph-based segmentation framework

Merging algorithms are based on the concept of homogeneity, in terms of texture, motion or structure, for example. The merging criterion decides whether two regions have to be merged. In this paper we use a color-based homogeneity criterion to create a set of initial regions allowing the evaluation of a structure-based criterion. Thus, we introduce a new structure-based homogeneity criterion, which is

1 This material is based upon work partly supported by the IST program of the EU in the project IST-2000-32795 SCHEMA and by the grant TIC2001-0996 of the Spanish Government.

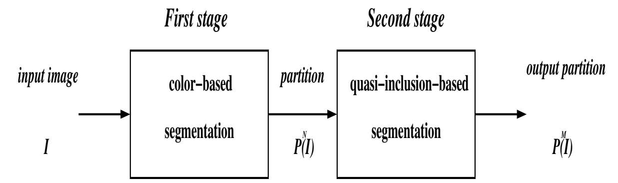

Figure 1. Bloc diagram for the segmentation process. The color-based segmentation leading to an over-segmented partition with N regions. This fine partition is used as input for quasi-inclusion segmentation. Structure-based merging algorithm is applied to the over segmented partition.The final partition presents M regions, where $M \leq N$.

applied on pairs of neighborhood regions by evaluating a quasi-inclusion value. With this criterion we want to assess whether one region is included into the other by computing its quasi-inclusion percentage.

Within this approach, the first level of abstraction is obtained afer the first stage and is the representation of the image through regions which are obtained by merging pixels using a color homogeneity criterion. The resulting partition is used as starting region configuration for the syntactic analysis in the second stage, as shown in Fig.1. Region relationships are then evaluated with a Region Adjacency Graph (RAG) where each node of the RAG models a region contour allowing convex-hull computation for the quasi-inclusion criterion. Image segmentation is an iterative merging process performed over the RAG nodes by applying this syntactic criterion.

The quasi-inclusion value is derived from the region convex-hull (CH). Therefore, the CH of the involved regions needs to be calculated, see [5]. In this context, computational geometry techniques require the appropriate selection of algorithms and data structures. Since we are dealing with the study of quasi-inclusion properties of image regions we have to choose an image representation which emphasizes region connectivity while allowing efficient shape extraction.

In the next section, we present the algorithms and data structures that have been used in this paper for syntactic-based and color-based segmentation.

2.1. RAG: spatial structured description

Graphs are one of the most widely used relational structures for image analysis and representation. Indeed, they offer a rich and compact representation for structural relationships. In this paper we are interested in spatial region configurations, which are very well synthesized using the RAG. A RAG is a set of nodes and links. Nodes represent regions from the image, and links connect each pair of neighboring regions. We consider that two regions, R_1 and R_2, are adjacent when there is a common contour segment between them in the partition image. Moreover, each node models the syntactic or color feature of the region and each link models the result of merging the two nodes it connects.

In this framework, segmentation is performed by merging regions satisfying a pre-defined criterion. Merging of neighboring regions is done in the RAG by:

- Removing the link connecting those regions.
- Merging the two associated nodes in order to create a new merged region with the model specified in the link.
- Updating the links to the neighboring regions.

According to [1], every concept will be associated to a different data structure resulting in the following entities:

Region model $\mathcal{M}_R$**:** concept related to the region (R) representation.

Merging order $O(R_1, R_2)$**:** concept related to the region homogeneity definition, that is, the likelihood that two regions have to be merged.

Merging criterion $C(R_1, R_2)$**:** concept related to the termination criteria.

This representation clearly separates each segmentation concept. All RAG nodes are firstly initialized with their corresponding region model. Then, links are placed in a hierarchical queue and extracted following the priority given by the merging order. The merging criterion is boolean and states that two nodes will be merged only if the criterion is true. When two nodes are merged the queue is updated. This is an iterative process.

Note that the merging order defines which regions should be merged first. Therefore, the introduction of higher level primitives for the merging process, such as structural criteria, has to be done by expressing them properly through the merging order.

2.2. First stage: Color-based segmentation

Segmentation in the color space has been performed in the first stage using the following values for the *general merging algorithm*,

$\mathcal{M}_{R_i}$**:** Region is modeled by a vector in the color space. This vector is the mean color value of the pixels belonging to the *i-th* region (R_i).

$O(R_1, R_2)$**:** Using a zero-order model the color homogeneity order is a linear combination of the values defined for each color component (c). Thus region similarity is defined as follows:

$$O_{color} = \sum_c \omega_c O_c(R_1, R_2)$$

where, $O_c(R_1, R_2) = A_{R_1}||M_{R_1} - M_{R_1 U R_2}||_2 + A_{R_2}||M_{R_2} - M_{R_1 U R_2}||_2$, $||.||_2$ is the $\mathcal{L}_2$ norm and A_{R_i} is the area of $R_i, i = 1, 2$.

$C(R_1, R_2)$: The termination criterion is the desired final number of regions.

We choose as merging criterion the final number of regions (N) leading to the creation of over-segmented partitions, see Fig.2. A large number of initial regions allows us to assume that the desired object can be obtained by merging some of this set of regions with the appropriate criterion.

Starting from individual pixels the first stage yields partitions whose regions are homogeneous in color.

2.3. Second stage: Quasi-Inclusion-based segmentation

The previous algorithm is used to obtain the initial partition for a quasi-inclusion-based segmentation. Let us first introduce some basic definitions, extracted from [4], which might simplify the notation and allow the definition of the quasi-inclusion concept in the context of the syntactic segmentation.

Region boundary parametrization: Γ, the region boundary, is a 2D simple closed curve parametrized in a clockwise manner such as $\{(x(t), y(t)), t \in [a, b]\}$.

Position function z of a curve Γ: In $\mathbb{C}^2$ Γ can be represented as $\{z = z(t) = x(t) + jy(t), t \in [a, b]\}$.

Convex-hull: The CH of a set S is the intersection of all the convex sets that contain S.

Quasi-inclusion Q^{inc}: Quasi-inclusion is the percentage of R_1 included in the convex-hull of the neighboring region R_2.

In this paper the region boundary has been extracted with 4-connectivity and parametrized by the position function in an interpolated image in order to simplify their extraction, see [7]. The initial region boundaries are derived from the image presented in Fig.2. This representation allows us to directly obtain the relevant contour points, which are defined as,

Relevant contour point (RP): is a point P belonging to a contour Γ and having 3 or more different neighboring region labels in the partition, or equivalently, having 3 or more contour elements.

To introduce syntactic properties into the merging algorithm the following new structures, directly related to region shape description, have been implemented:

$\mathcal{M}_{R_i}$: The region model is the region boundary Γ and the list of the relevant points belonging to the contour.

$O(R_1, R_2)$: The merging order is the percentage of R_1 quasi-included in the convex-hull of R_2 ($CH(R_2)$).

$$Q^{inc} = \frac{|CH(R_2) \cap R_1|}{A_{R_1}} \tag{1}$$

$C(R_1, R_2)$: The termination criterion is simply the quasi-inclusion threshold.

The merging of two regions (R_1, R_2) is performed by combining the two models ($\mathcal{M}_{R_1}$, $\mathcal{M}_{R_2}$). In this new framework this means:

1. Fusing the two contours Γ_1 and Γ_2 using the information associated to the list of RP.
2. Updating the list of RP with the existing items.

Notice that, contour and RP extraction is only performed once, for the initial creation of the RAG. During the merging process, the models are simply combined reducing the computational cost.

The syntactic approach is based on the assumption that the desired object boundary is a sub-set of the initial region contours, this condition should be guaranteed by using as initial partition an over-segmented image. Therefore, when the desired object can be well modeled by quasi-inclusion properties, segmentation should result in a single region representing the object at some stage of the merging process.

In this section we have presented the extension of the *general merging algorithm* with a new region model, merging order and merging criterion within a syntactic approach. This is the necessary base for undertaking the quasi-inclusion-based BPT construction.

3. Binary-Partition-Tree representation

The Binary-Partition-Tree is a region-based image representation, usually obtained from a segmentation procedure. The BPT allows a multi-scale representation and the region connectivity is invariant to translation, see [2]. BPTs have a wide range of applications such as, filtering, segmentation, information retrieval, visual browsing, etc.

In this paper we will take advantage of the BPT's capability of tracking the sequence of mergings performed during a merging process. The BPT construction is based on the merging order and the region model, whereas the merging criterion is trivial. The previous merging algorithm, presented in section 2, with a trivial color or structure criterion leads to a BPT representation. The BPT presents a compromise between representation accuracy and processing efficiency. That means that only the most likely feasible mergings in the RAG are represented in the BPT.

Within the BPT framework, leaves of the tree are in both cases –color and structure– the regions belonging to the initial color partition. The remaining nodes represent the regions resulting from the successive mergings according to the pre-defined homogeneity criterion. For example, applying a color homogeneity criterion each node represents a region with the color mean of its two children. In a similar way, using structure homogeneity each node represents a region with a contour encompassing the outer contours of its children. Note that in both cases the root node represents the image support.

In the next section we present some results illustrating the concepts related to these segmentation approaches.

4. Results

In this section we first present the results of the segmentation process for the color and the structure criteria. Since, BPTs are a powerful tool for region-oriented representation, we build the BPTs associated to this two different homogeneity approaches in order to compare them in the second part of this section.

The original image of a butterfly[2], the resulting color-based partition with $N = 15$ regions and the initial region boundaries obtained after the first stage is shown in Fig. 2. The initial partition is used as input for the second stage with both criteria: color and structure. Moreover, this partition should include the relevant regions to generate convenient color and structure BPT representations. As we can see, the desired object regions (or boundaries) set are present after the first stage.

Performing a quasi-inclusion segmentation using the previous partition with different termination criteria (Q^{inc}) allows us to verify that the new merging procedure performs well. As it can be seen in Fig.3, with $Q^{inc} = 100\%$ all the regions fully included into other regions are merged resulting in a partition with M=11 regions. Relaxing the criterion, for example to $Q^{inc} = 50\%$, all the regions having more that 50% of his area into the CH of another region are included into other regions are merged and the number of regions after the second stage is now M=8. We have confirmed that this criterion can correctly deal with this geometric region relationships. Moreover, the quasi-inclusion merging algorithm can be used for post-processing purposes.

To compare color and structure approaches we construct two BPTs, which can be seen as a tracking of all the mergins performed in the two different second stages. Moreover, both BPT representations are obtained using the same color-based initial partition with $N = 15$ shown in Fig.2.

2 This image is copyright Corel Corp.

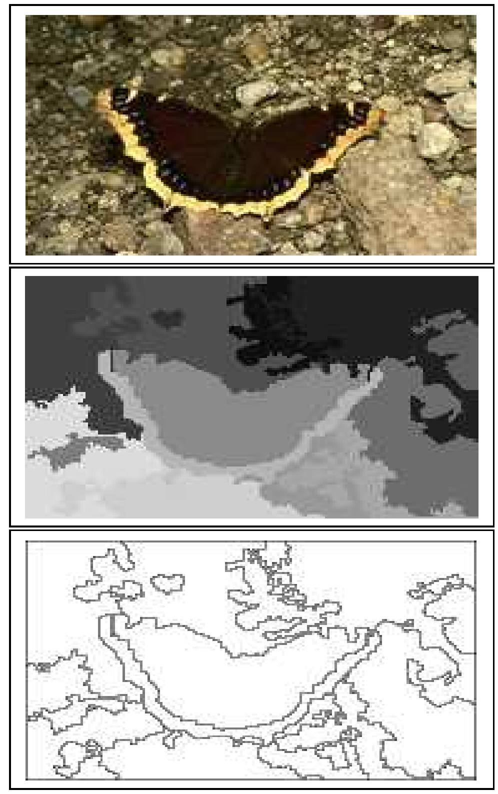

Figure 2. Original image, color-based partition after the first stage with N=15 regions and their associated region boundaries.

Firstly, a BPT is build using color homogeneity. In this case, the first and the second stage are performed with the same color-based criterion. Secondly, a structure-based BPT representation is build. For this purpose, the first stage is color-based , while the second stage is performed with a quasi-inclusion criterion.

Figure 4 shows the BPT associated to the color-based segmentation of the butterfly image. This representation is based on the same merging order defined as in the first stage. As we can see in Fig.5 and Fig.6, the black part of the butterfly (node 9) is merged with the dark part of the background (node 27), while the outside part, which is yellow (node 12), is merged with the bright part of the background (node 18). Therefore, the color homogeneity criterion is unable to segment the desired object because the outer part of the objects is more similar to the background than to the image parts.

The BPT representation of the butterfly image obtained

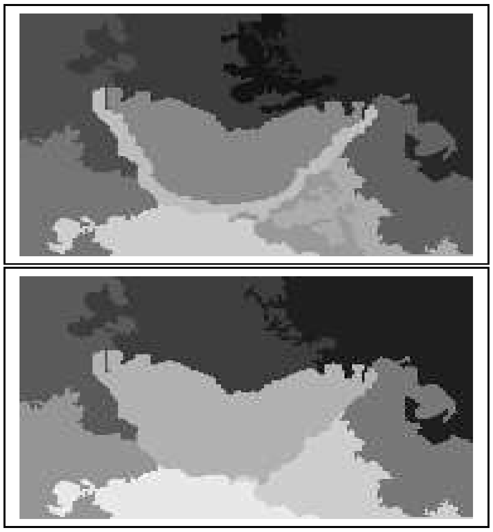

Figure 3. Structure-based segmentation. Top $Q^{inc} = 100\%$ and $M = 11$ regions. Bottom $Q^{inc} = 50\%$ and $M = 8$ regions.

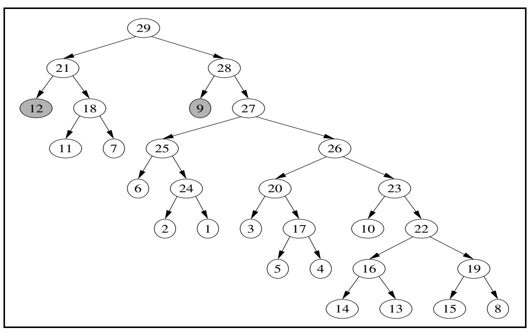

Figure 4. Color-based BPT: Example of color homogeneity criterion

with structure homogeneity can be seen in Fig.7. Regions are merged following the quasi-inclusion order leading to a representation of the image region structure. In the case of a quasi-inclusion criterion, the butterfly is obtained by merging node 9 and node 12 and represented with node 20, see Fig.8. Note that color information has not been used.

Comparing both BPT representations, we can see that the results obtained with quasi-inclusion can deal with intrinsic object relationships related to the structure, whereas a color based homogeneity is not able to manage this object properties. Even though quasi-inclusion results strongly depend on the spatial configuration of the regions belonging to object, these examples are a positive proof of concept to the

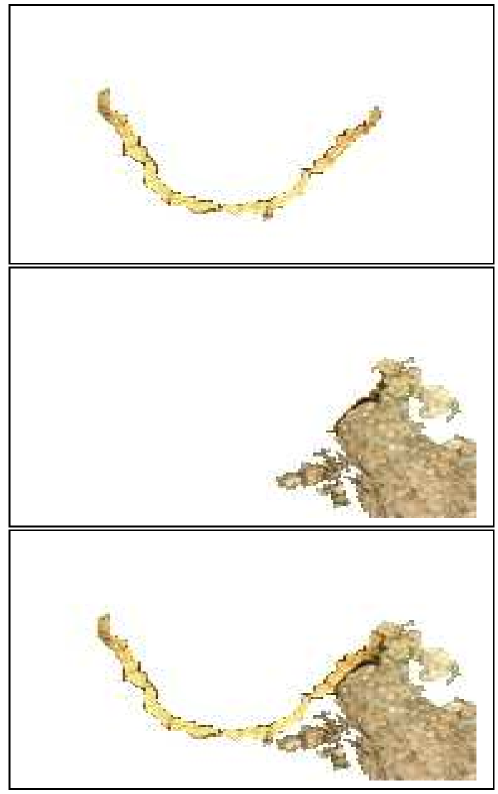

Figure 5. Color homogeneity criterion: Image regions represented by the BPT nodes 12, 18 and 21 respectively.

new approach.

5. Conclusions

In this paper we have extended the *general merging algorithm* introducing structural region relationships, such as quasi-inclusion, derived from the *general merging algorithm* and we have create the associated BPT. An effort to separate concepts, like color and structure, has been carried out during the implementation of the segmentation algorithm. Quasi-inclusion-segmentation has proved to be suitable for objects composed of overlapped convex-hull regions (structure-based homogeneity).

However, this simple criterion has obvious limitations. Therefore we plan to extend this framework by combining other homogeneity criteria and introducing other syntactic features.

References

[1] L. Garrido and P. Salembier. *Region based analysis of video sequences with a general merging algorithm.* In European

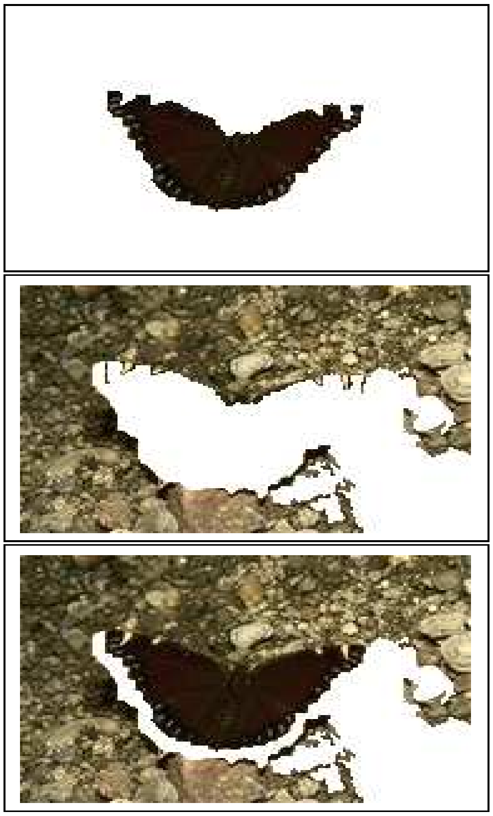

Figure 6. Color homogeneity criterion: Image regions represented by the BPT nodes 9, 27 and 28 respectively.

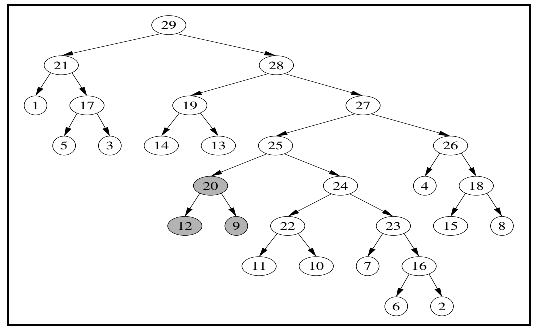

Figure 7. Structure-based BPT: Example of structure homogeneity criterion

Figure 8. Quasi-inclusion homogeneity criterion: Image regions represented by the BPT nodes 12, 9 and 20 respectively.

Signal Processing Conference (EUSIPCO), Rhodes, Greece, September 1998.

[2] P. Salembier, L. Garrido. *Binary Partition Tree as an Efficient Representation for Image Processing, Segmentation, and Information Retrieval.* In IEEE Transactions on Image Processing, 9(4):561-576, April 2000.

[3] Rafael C.Gonzalez and Michael G. Thomason. *Syntactic Pattern Recognition. An introduction.* Addison-Wesley Publishing Company, Inc, 1978. ISBN 0-201-02930-8

[4] P.J. van Otterloo. *A Contour-Oriented Approach to Shape Analysis.* Prentice-Hall, 1991.

[5] Mark de Berg, Marc van Kreveld, Mark Overmars and Otfried Schwarzkopf. *Computational Geometry: Algorithms and Application.* Springer-Verlag, Berlin 1997. ISBN 3-540-61270-X

[6] D. S. Zhang and G. Lu. *Review of Shape Representation and Description Techniques*. Pattern Recognition, 37(1):1-19, 2004.

[7] F. Marques and A. Gasull. *Partition coding using multigrid chain code and motion compensation.* In IEEE International Conference on Image Processing, ICIP'96, volume II, pages 935-938, Lausanne, Switzerland, September 1996.