Combining boundary and region features inside the combinatorial pyramid for topology-preserving perceptual image segmentation

Esther Antúnez[a], Rebeca Marfil[a], Antonio Bandera[a,*]

[a]*Department of Electronic Technology, University of Málaga, Higher Technical School of Telecommunication Engineering, Málaga, Spain*

Abstract

Combinatorial pyramids represent the image as a stack of successively reduced combinatorial maps, which encode the whole image at different levels of abstraction. Within this framework, this paper proposes to conduct the perceptual organization of the image content in two consecutive stages. The first stage builds the lower set of levels of the hierarchy according to simple face (regions) features (colour and size). On the top of this hierarchy, the second stage will mainly employ boundary features, encoded in the darts of the combinatorial maps, to obtain a second set of levels of abstraction. The Berkeley data set BSDS300 is used to quantitatively compare the performance of the proposal to a number of perceptual grouping approaches, showing that it yields better or similar results than most of these algorithms while offering two interesting features: computation at multiple image resolutions and preservation of the image topology.

Keywords: Image segmentation, Contour detection, Irregular pyramid, Combinatorial maps, Perceptual segmentation

1. Introduction

When the goal of an image processing algorithm is to divide the input image in a manner similar to human beings, the adopted strategy cannot simply be the grouping of image pixels into clusters (regions or boundaries) taking into account low-level photometric properties (Martin et al., 2004, Arbeláez et al., 2011). Natural images

*Corresponding author

Email addresses: eantunez@uma.es (Esther Antúnez), rebeca@uma.es (Rebeca Marfil), ajbandera@uma.es (Antonio Bandera)

are generally composed of physically disjoint objects whose associated groups of image pixels may not be visually uniform. Hence, it is very difficult to formulate what should be recovered as a region or boundary from an image or to separate complex objects from a natural scene (Lau and Levine, 2002). With the aim of organizing low-level image features into higher level relational structures, the perceptual organization of the image content is usually thought as a process of grouping visual information into a hierarchy of levels of abstraction. Starting from the lower level of the hierarchy (i.e. the input image or an initial partition), each new layer groups the regions of the level below into a reduced set of regions. This grouping needs to define a region model (the features that describe each image region) and a dissimilarity measure (the metric on those features) (Brox and Farin, 2001). Moreover, it is interesting for efficiency reason that the grouping can simultaneously merge more than two regions.

According to the aforementioned properties, many heuristics have been proposed. The simplest model describes region by luminance and size (Beaulieu, 1989). On this model, the dissimilarity measure is usually the squared difference or the Ward-criterion. Regions can be also described by information of their boundaries. Thus, the gPb-owt-ucm approach (Arbeláez et al., 2011) transforms the output of the gPb contour detector into a hierarchical region tree. The approach employs the Oriented Watershed Transform (OWT) to obtain a set of initial regions from the output of the contour detector, and builds an Ultrametric Contour Map (UCM) from the boundaries of these initial regions. The dissimilarity measure between two regions is defined by the average strength of their common boundaries. The initial segmentation can be also obtained through a watershed (Meyer, 2005). Watershed algorithms presents the advantage of providing closed contours, which lead to a proper definition of regions (Brun et al., 2005). The hierarchical watershed approaches assume that the over-segmentations usually produced by the watershed algorithms include the correct boundaries on the image. Then, if these boundaries are properly valuated, the initial partition provided by the over-segmentation of the input image can be decimated to build the hierarchy of levels (Najman and Schmitt, 1996, Brun et al., 2005). Information of the basins (regions) is typically conjointly used with the contour attributes to perform this decimation. Once the region model and dissimilarity measure have been defined, the algorithm can proceed by continuously searching for the lowest dissimilarity value and merging the two corresponding regions until a stopping criterion is satisfied or there is only one region (Arbeláez et al., 2011). If the hierarchy of partitions is encoded using irregular pyramids, several regions can be simultaneously merged between two consecutive layers (Brun et al., 2005).

Irregular pyramids represent the image as a stack of graphs with decreasing number of vertices. Some irregular pyramids use a simple graph (i.e. a region adjacency graph

(RAG)) to encode each level of the hierarchy. Region adjacency graphs do not permit to know if two adjacent regions have one or more common boundaries, and they do not allow to differentiate an adjacency relationship between two regions from an inclusion relationship. Instead of simple graphs, each level of the hierarchy can be represented using a pair of dual graphs or a combinatorial map. Thus, the combinatorial pyramid (Brun and Kropatsch, 2001) is defined by an initial combinatorial map that can be successively reduced using the general scheme proposed by Kropatsch (1995). In the multiscale framework provided by the combinatorial pyramid, this paper presents an approach to perceptual image segmentation that combines information coming from regions and boundaries. Contributions include:

- A novel, multi-stage algorithm to combine boundary and region information inside the hierarchy of the combinatorial pyramid.

- Region merging is conducted using two different metrics inside the same hierarchy, generating a representation of the image at different levels of abstraction or scales. At low scales, only region features (size and colour information) are considered in the model. The resulting blobs or superpixels (Ren and Malik, 2003) reduce image complexity while avoiding undersegmentation. These superpixels are then grouped into larger structures using boundary and region properties.

- The proposed approach has been extensively evaluated using the precision-recall framework introduced by Martin et al. (2004) on the Berkeley Segmentation Data Set (BSDS300). Results show that it can be favorably compared with other leading approaches.

The main advantage of the proposed framework is that the combinatorial pyramid preserves at all levels of the hierarchy the topological relationships of the original image. Thus, the decomposition of the image into regions at each level is represented by a combinatorial map that encodes correctly these relationships (Brun and Kropatsch, 2001, Brun and Kropatsch, 2006). It should be noted that this paper improves a previous version proposed by the authors (Antúnez, 2011a), where only face attributes of the combinatorial maps were used for segmentation. With respect to this first algorithm, the new algorithm directly associates the darts of the combinatorial map with edge information. It should be noted that the use of edges in a hierarchy is not completely new. They were used, for example, by Burge and Kropatsch (1999) in the dual graph-based irregular pyramid framework. In our case, this will be the main factor employed to perform the perceptual grouping at high scales of the hierarchy. The rest of the paper is organized as follows: an overview of the approach is presented in Section 2. Section 3 describes it in detail. Experimental results revealing the efficiency of the

proposed method are presented in Section 4. Finally, the paper concludes along with discussions and future work in Section 5.

2. Overview of the proposed approach

The key idea in the proposed approach is to reduce the perceptual grouping computation to an efficiently solvable clustering problem. This clustering process will be hierarchically conducted in two sequentially conducted stages (Antúnez, 2011a):

- A pre-segmentation stage that accumulates local evidences from the original image (level 0 of the hierarchy) to a combinatorial map (level l_p). This map will encode a decomposition of the image into superpixels. This initial stage of the clustering process is guided by the principles described by Levinshtein et al. (2009). Thus, blobs represent connected sets of pixels without overlapping among them. They are compact and their boundaries coincide with the main image edges when the pre-segmentation stops. Additionally, they correctly encode the topological relationships of the original image.

- A perceptual grouping stage whose aim is to hierarchically merge the previously obtained blobs into a reduced set of perceptually significant components, using the level l_p of the hierarchy as its base level. The principles that drive this perceptual grouping are similar to the ones employed at the first stage of the approach (connectivity, compactness, topology preservation). However, this stage must preserve image boundaries, i.e. the changes in pixel ownership from one object or surface to another (Martin et al., 2004). The key point is here the use of image edge evidences, which are complemented with the local intra-region attributes employed at the pre-segmentation stage. The upper level of the hierarchy is a combinatorial map, which preserves the topological information of the original input image.

These two steps will be discussed in detail in Section 3. The whole approach employs three main parameters. Two values are used to threshold the minimum allowed arc weight at the two different stages of the approach. The other parameter is the maximum scale l_p allowed for the pre-segmentation stage. There is also a fourth set of parameters that is used to adjust the global arc weight at the perceptual grouping stage. In this paper, and in order to design a generic approach for segmentation, these internal parameters will be learnt by taking into account the F-measure provided by the training images and corresponding ground truth of the BSDS300 (Section 4). However, the preferences of the user could impose other values, according to their requirements on storage or computational costs.

3. The perceptual image segmentation approach

A combinatorial map is a mathematical model describing the subdivision of a space. It encodes all the vertices which compound this subdivision and all the incidence and adjacency relationships among them. A combinatorial pyramid is a hierarchical stack of combinatorial maps successively reduced by a sequence of contraction or removal operations (see (Brun and Kropatsch, 2001, Brun and Kropatsch, 2006) for further details). In our implementation, two-dimensional (2D) combinatorial maps are defined with the triplet $G = (D, \sigma, \alpha)$, where D is the set of darts, σ is a permutation in D encoding the set of darts encountered when turning (counter) clockwise around a vertex, and α is an involution in D connecting two darts belonging to the same arc:

$$\forall d \in D, \alpha^2(d) = d \tag{1}$$

Fig. 1a shows an example of combinatorial map. In Fig. 1b the set D and the permutations α and σ for such a combinatorial map can be found. In the proposed approach, counter-clockwise orientation (ccw) for α is chosen.

Given a dart d and a permutation β, the β-cycle of d denotes the series of darts defined by the successive application of β on the dart d (Brun and Kropatsch, 2001). The σ and α cycles of a dart d are, respectively, denoted by $\sigma^*(d)$ and $\alpha^*(d)$. In this case, the cycle $\sigma^*(d)$ encodes the sequence of darts encountered when turning counter-clockwise around the vertex encoded by the dart d. The cycle $\alpha^*(d)$ encodes darts that belong to the same arc. Given a combinatorial map, its dual is defined by $\hat{G} = (D, \varphi, \alpha)$, with $\varphi = \sigma o \alpha$. The cycle concept allows labelling darts as belonging to a vertex, arc or face of the graph. If vertices, arcs and faces of the graph are respectively encoded by the sets V, E and F, then a labelled combinatorial map (Brun et al., 2003) can be defined as an n-tuple $G = (D, V, E, F, \sigma, \alpha, \mu, v, \pi)$,

Thus, in the example of Fig. 1, all darts in $\varphi^*(1) = (1, 5, 7, 10)$ are related with the graph face A.

When a combinatorial map is built from an image, the vertices of such a map G could be used to represent the pixels (regions) of the image. Then, in its dual $\hat{G}$, instead of vertices, faces are used to represent pixels (regions). Both maps store the same information and there is not so much difference in working with G or $\hat{G}$. In the proposed approach, the base level of the pyramid will be a combinatorial map where each face represents a pixel of the image as a homogeneous region. The combinatorial pyramid is built reducing this initial combinatorial map successively by a sequence of contraction or removal operations (Brun and Kropatsch, 2001, Kropatsch, 1995).

3.1. Arcs and faces description

In our approach, the combinatorial pyramid associated to the input image is built using two different strategies, which constitute the pre-segmentation and perceptual grouping stages. However, faces and arcs of the combinatorial maps encoding each level of the hierarchy are attributed by the same set of descriptors in both stages. Faces are attributed by the mean colour of their corresponding pixels in the input image, $colour(f_k)$. The CIELab space has been used for colour encoding. They are also attributed with their sizes at the base level (i.e., the number of pixels of their receptive fields). On the other hand, each arc of the map e_k is attributed with three descriptors:

- The length of the arc, $length(e_k)$
- The difference on colour of the regions (faces) it separates, $colour(e_k)$
- The mean edge gradient along this boundary, $strength(e_k)$. This value is initialized from an edge image at the base level, attributing the arc set E_0.

As we will show in next subsections, the pre-segmentation stage is driven by the $colour(e_k)$ descriptor, meanwhile the perceptual grouping stage is mainly driven by the $strength(e_k)$ descriptor. In any case, all descriptors are updated when a new level is built. Thus, at the pre-segmentation stage, the $strength(e_k)$ descriptor is updated when arcs are contracted, i.e. in case of concatenation of boundaries, where $CK1_{k,k+1}$ defines a contraction kernel (Brun and Kropatsch, 2001), i.e. the set of arcs e_k that generates e_{k+1}. When arcs are contracted, the $length(e_k)$ value is also updated from the previous level, as the sum of the lengths of the concatenated boundaries.

3.2. Pre-segmentation stage

Let G_0 be a given labelled combinatorial map with the vertex set V_0, the arc set E_0 and face set F_0 on the base level (level 0) of the pyramid. In the same way, the combinatorial map on level k of the pyramid is denoted by G_k. As it was aforementioned, each face of the base level represents a pixel of the image. Faces are attributed with the colour and brightness of the corresponding pixel. The hierarchy of graphs is built using the algorithm proposed by Haxhimusa and Kropatsch (2004), which is based on a spanning tree of the initial graph obtained using the algorithm of Boruvka (1926). However, in the proposed approach, the method to merge two faces is different from the one used in (Haxhimusa and Kropatsch, 2004). Thus, two faces are now merged if the difference of colour between them is smaller than a given threshold U_p. That is, the attribute of each arc of the graph $colour(e_k)$ is compared with the threshold U_p

and if this value is smaller, that arc is added to a removal kernel. Then, in a second step, hanging arcs are contracted. Finally, a contraction kernel is applied to contract vertex chains of order two, obtaining the new level of the pyramid. This process is iteratively repeated until the level l_p is reached or no more removal/contraction operations are possible. At this stage, the information about the image content goes up from level k to level $k+1$ by updating the attributes of the new faces. The attribute of a new face is the weighted mean colour of the faces that have been merged. This weighted mean colour takes into account the colour of the regions as well as the size of their receptive fields. After setting the attributes of the faces, the algorithm sets the $colour(e_k)$ attributes of the arcs. The attributes of the arcs are update with the colour difference of the new faces they separate. As aforementioned in Section 4.1, $strength(e_k)$ and $length(e_k)$ attributes are also updated. Algorithm 1 shows how to build the levels of the combinatorial pyramid associated to the pre-segmentation stage. Building the spanning tree allows to find the region borders quickly and effortlessly based on local differences in a colour space (Ion et al., 2006). This process results in an oversegmentation of the image into a set of superpixels (regions with homogeneous colour). Besides, the topology is preserved (Brun and Kropatsch, 2001, Brun and Kropatsch, 2006). These superpixels will be the input of the perceptual grouping stage. Fig. 2 gives a qualitative feed for the superpixels provided by the proposed stage for several images from the Berkeley database. It can be noted that the blobs respect the salient boundaries, while remaining compact in colour. The obtained superpixels do not exhibit an uniform size.

3.3. Perceptual grouping stage

After the pre-segmentation stage, the perceptual grouping stage aims for simplifying the content of the obtained colour-based image partition. To join pre-segmentation and perceptual grouping stages, the last level of the combinatorial pyramid associated to the pre-segmentation stage will constitute the first level of the hierarchy associated to the perceptual grouping stage. Next, successive levels will be built using the decimation scheme described in Section 3.2. However, the perceptual stage is mainly driven by boundary evidences, encoded in the attributes of the arcs of the combinatorial map.

With respect to the measure of dissimilarity between two adjacent faces, it is initially defined as the minimal edge strength of their common boundaries. Thus, the algorithm is very similar to the one used at the pre-segmentation stage, employing a different threshold value, U_s. As different attributes are employed, some minor differences also appear. Thus, it should be noted that this stage merges two regions if at least one of their common boundaries have a strength value below the threshold (in the previous stage, the colour value is always the same for all common boundaries between

two adjacent regions). On the other hand, when this approach has been evaluated using the BSDS300 dataset, it has been found that there existed certain arcs that were not contracted despite the colours of the faces they separated are very similar. As it was pointed out by Arbeláez et al. (2011), this problem arises from the adopted strategy for arc weighting. Edge detectors typically produce spatially extended responses around strong edges. Those arcs of the combinatorial map that lie near but not on these strong edges will be then erroneously upweighted. Fig. 3 shows an example of this problem. Fig. 3b illustrates the edge map associated to the central part of Fig. 3a. The three yellow bands are correctly delimited by strong edges (dark values in this image). In Fig. 3c, we have coloured each pixel of the boundaries generated by the pre-segmentation stage with its associated edge weight. It can be noted that short boundaries inside the bands also present dark values, which are specially significant in their end-points. These end-points increase the average edge strength values of these short boundaries, avoiding its removal. This issue was even worse when we tested to use edge information at lower levels of the hierarchy. In the proposed approach, this negative effect has been reduced at this stage by including the face descriptors in the evaluation of the strength of the arcs. Thus, if the $colour(e_k)$ is very low (under a fixed value u_r), the strength of the arc will not be equal to $strength(e_k)$, but to $strength(e-k) - \alpha(1 - colour(e_k)/\eta$, where η defines the maximum distance between two colour values in the chosen colour space. The pair of values (u_r, α) has been heuristically set from the experiments on the BSDS300 dataset. Fig. 4 shows several examples of segmentation results on the BSDS300. The algorithm proposes image partitions at different scales, which have been illustrated on the figure (whiter colour values correspond to higher scales). To present a single segmentation as output, the scale that provides the better performance on the BSDS300 has been employed. For edge detection, the mPb detector (Arbeláez et al., 2011) has been employed. Further evaluation is provided in Section 4.

4. Experimental results

4.1. Quantitative evaluation of the pre-segmentation stage

In order to evaluate how well superpixel boundaries align to image edges, the Berkeley Segmentation Dataset and Benchmark (BSDB300) 1 (Martin et al., 2001) has been used. The methodology for evaluating the performance of segmentation techniques using this dataset is mainly based in the comparison of machine detected boundaries with respect to human-marked boundaries (ground truth data) using the precision-recall framework (Martin et al., 2004). This technique considers two quality measures: precision and recall. The precision is defined as the fraction of boundary detections that are true positives rather than false positives. Thus, it quantifies the amount of

noise in the output of the boundary detector approach. The recall is defined by the fraction of true positives that are detected rather than missed. Then, it quantifies the amount of ground truth detected. In the proposed approach, in order to evaluate how well superpixel boundaries align to image edges the recall measure has been used. Then, given a boundary in the ground truth, a search is made for a boundary in the superpixel segmentation within a distance of a small number of pixels (2 pixels in these experiments). The recall value is the percentage of length of ground truth boundary that is also present in the pre-segmentation decomposition within this threshold of 2 pixels.

Fig. 5 shows a comparison of the proposed method with the algorithms by Felzenszwalb and Huttenlocher (2004) (FelzH), Levinshtein et al. (2009) (TurP), Yu and Shi (2003) (NCut), Christoudias et al. (2002) (Edison), Veksler et al. (2010) (EnO), Achanta et al. (2010) (SLIC) and Haxhimusa et al. (2006) (CPcon). Source codes have been downloaded from the web sites provided by the authors. The approach by Felzenszwalb and Huttenlocher (2004) (FelzH) is a graph-based segmentation method that performs an agglomerative clustering of pixel nodes on the graph. Thus, each region is the shortest spanning tree of the constituent pixels. It does not offer an explicit control on the number or compactness of superpixels. The TurboPixel algorithm (TurP) by Levinshtein et al. (2009) employs a gradient-based affinity function of a gray-scale image to grow superpixels from seeds placed regularly in the image. It offers the compactness of superpixels, but it also aligns the superpixel boundaries with image edges when they are present. The Normalized cut approach (Yu and Shi (2003)) (NCut) is another graph-based method, which conducts a recursive partition of the input graph using boundary and texture features. It globally minimizes a cost function defined on the arcs at the partition boundaries. It provides control about the compactness of superpixels. The Edison algorithm (Christoudias et al., 2002) integrates the confidence-based edge detector with the mean-shift based image segmentation. The approach by Veksler et al. (2010) (EnO) regularly covers the image with square patches of fixed size. Then, the partitioning problem is stated as a energy minimization problem optimized with graph cuts. Superpixels cannot be extended out of the original square patches. The authors provide two versions of the approach. In this comparison, we have used the formulation that provides constant intensity superpixels. Using this version, less regular space tessellation and more accurate boundaries are provided. Achanta et al., 2010 propose to obtain superpixels using a simple linear iterative clustering (SLIC). This algorithm performs a local clustering of pixels in the 5-dimensional space defined by the values of the CIELab colour space and the image pixel coordinates. The proposed distance measure enforces compactness and regularity in the shapes of superpixels. Finally, the algorithm by Haxhimusa et al. (2006) (CP-

con) is the first version of the MST-combinatorial pyramid. It uses the difference in image colour proposed by Felzenszwalb and Huttenlocher (2004) as affinity function in all levels of the hierarchy. In order to set the internal parameters of these algorithms for comparison, we have imposed that they must partition the image into a specific set of superpixels. Several approaches only require to set this parameter to provide the tessellation (e.g. NCut, TurP or SLIC). In other cases, we had to perform a search on the parameter space to achieve this control (sometimes it was not possible to exactly obtain the desired value, but nearby values were used to interpolate the desired ones). Two measures have been employed to perform this comparison: Fig. 5 a shows the dependency of the recall value on the number of superpixels for each algorithm, and Fig. 5 b shows the undersegmentation error, which measures the percentage of area that the regions output by an algorithm differs from the ground-truth regions (Levinshtein et al., 2009 , Achanta et al., 2010).

The results of Fig. 5 have been obtained by averaging over 100 images in the dataset. It can be noticed from Fig. 5a that when the number of superpixels increases, there are more boundaries and the recall is better for all algorithms. In fact, for a high number of superpixels, the performance of all approaches is similar. However, for smaller number of superpixels, there exists significant differences. From the figure, it can be concluded that the proposed algorithm has a performance comparable to the ones provided by the FelzH, the Edison and the EnO (in this case, the 'constant intensity superpixels' has been employed for comparison (see Veksler et al., 2010 for further details)). The proposed approach outperforms the performance of those approaches whose aim is to decompose the image into superpixels of uniform size, such as CPcon, SLIC, TurP or NCut. On the contrary, undersegmentation errors (Fig. 5b) are typically lower in this last group of approaches, as they do not generate large superpixels. However, the nature of the proposed approach imposes that superpixels will be initially uniformly distributed on the image, in a way that resembles how seeds are initially scattered in the TurP approach. After the hierarchical grouping process is conducted through several levels, it can be appreciated that small superpixels are concentrated on the image boundaries and large ones in uniform image regions. But this process is not as noticeable as in other algorithms such as the Edison or the FelzH. Hence, undersegmentation errors are significantly lower in the proposed algorithm, being comparable to the ones provided by the EnO, SLIC or TurP approaches. Fig. 6 shows the superpixels provided by several algorithms for the same image. As foremen-tioned, some of these algorithms do not allow to control the number of superpixels. In the figure, the image has been approximately segmented in 400 superpixels.

From Fig. 5, Fig. 6, it can be concluded that the proposed algorithm provides a smaller undersegmentation error and a larger recall value than the MST Pyramid

based on Combinatorial Maps (CPcon) by Haxhimusa et al. (2006). This improving is specially important when the number of superpixels is more reduced. The first row of the Fig. 7 shows the image partitions provided by the CPcon at different levels of the hierarchy (different number of superpixels). Marked image regions show irreparable undersegmentation errors that happen when the number of superpixels is reduced. The second row of the Fig. 7 shows the image partitions provided by the proposed algorithm at different levels of the hierarchy. Image boundaries are respected despite of the severe reduction on the number of superpixels. On the other hand, the proposed algorithm is also conducted at a lower computational cost. Although both implementations have not been optimized (e.g. being designed to work in parallel, they currently run in a sequential manner), they share the same software structure and obtained processing times can be thus compared. From these comparisons (conducted over the test set of the BSDS300), it can be concluded that the proposed algorithm works approximately in less than a tenth of the time than the CPcon in an Intel (R) Core (TM)2 Duo CPU T8100 2.10 GHz. This is due to the use of a simple thresholding algorithm to perform the pre-segmentation stage instead of the algorithm based on external/internal differences. The building of the first levels of the hierarchy is the most computationally expensive step on the CPcon. However, it must be noted that this time estimation does not include the computation of the edge map. On the contrary, it can be also found disadvantages on the proposed technique. As faces are only characterized by the weighted mean colour of their receptive fields, there is an important lost of information. The adopted strategy for the pre-segmentation stage should not be extended to the higher layers of the hierarchy because wrong merges will occur. Thresholds should be set to avoid the presence of regions with high colour variance.

4.2. Quantitative evaluation of the perceptual grouping stage

The performance of the proposed image segmentation approach is conducted using the precision-recall framework over the BSDB300 (Martin et al., 2001, Arbeláez et al., 2011). In the performed experiments, best results are typically obtained using $\{U_p, l_p\} = \{100, 6\}$ and $\{U_s, u_r, \alpha\} = \{0.2, 15, 0, 02\}$. The best scale for partition on the perceptual grouping stage ranges from 4 to 6, depending on the chosen edge detector. On the other hand, although the perceptual grouping stage can employ any source of edges for the edge map, the best results have been obtained by using the variants of the Pb detector by Martin et al. (2004).

Regarding to the sensibility of the algorithm to changes on the parameters, the pre-segmentation stage exhibits a strong behaviour, being relatively easy to find a good pair of $\{U_p, l_p\}$ values. Thus, we have conducted several trials over the test set of the BSDS300, changing the $\{U_p, l_p\}$ values. For U_p values ranging from 25 to 100

and l_p from 4 to 7, the obtained recall value for the boundaries provided by the pre-segmentation output is always over 0.9. Higher l_p values induce a decreasing on the recall value. As it was pointed out by Ion et al. (2006), the first edge selection step (the Boruvka's algorithm at Algorithm 1) ensures that the approach may then obtain regions with small variations surrounded by borders with large variation (Ion et al. (2006)). These results confirm this assertion: the recall value is mainly a function of the level l_p. Experimental results also show that the time consumed for the whole algorithm is not largely dependent on the l_p value. As in the CPcon, the time is mainly consumed in the generation of the first levels of the hierarchy. On the other hand, the best scale for partition on the perceptual grouping stage can be also usually chosen from a wide range of valid values. In our tests on the BSDS300, the F-measure typically remains constant for a large range of scales. On the contrary, it is not easy to determine the best values for the parameters $\{u_r, \alpha\}$ and several trials have been conducted to find them. When the mPb detector is used, and depending on the choices, F-measures can vary between 0.612 to 0.651 for small variations on this pair of parameters.

Fig. 8 summarizes the main results obtained on the BSDS300 using the precision-recall framework. Fig. 8a represents the dependence of the proposed approach with the input provided by the chosen edge detector. Similar to the results by Arbeláez et al. (2011), it can be noted that the performance of the approach improves when an edge detector that exhibits a better behaviour on this database is employed. Fig. 8b shows the evaluation of multiple image segmentation approaches. The proposed approach is only superseded by the gPb-owt-ucm (Arbeláez et al., 2011) and the UCM (Arbeláez, 2006), providing better results than other approaches (Cour et al., 2005, Felzenszwalb and Huttenlocher, 2004, Comaniciu and Meer, 2002). With respect to the segmentation results provided by these approaches, it can be noted that the graph-based approach by Felzenszwalb and Huttenlocher (2004) and the Mean-Shift by Comaniciu and Meer (2002) produce segmentations that usually capture small, high-contrast regions. They tend to produce oversegmentations. On the contrary, the Normalized Cuts by Cour et al. (2005) typically produces under-segmentations. The gPb-owt-ucm is a very robust approach, which only suffers from those problems inherited from the edge detector (strong and weak intra-region variations can cause over-segmentations and under-segmentations, respectively). Similar problems affect our proposed approach.

4.3. Importance of preserving the image topology

Some applications like object detection or image correspondence are, generally, based on finding correspondences between image regions. Such correspondences are usually based on photometric or geometric image features like shape, colour or tex-

ture. However, on a real-world scenario, these features change when rotation, scale, illumination or 3-dimensional pose vary. Adding information about the topological relationships among the regions of the image can be very helpful in these cases. Thus, the objects are not only characterized by features or parts, but also by the spatial relationships among these features or parts. In that way two regions of different images can be matched if they have similar features (i.e. similar colour or texture) and they also present similar topological relationships with their neighbour regions (Brun and Pruvot, 2008, Antúnez et al., 2011b). Region adjacency graphs (RAG) do not always encode all the necessary information. Fig. 9 shows a very simple example. The aim is to find the object template in Fig. 9a into the image at Fig. 9d. Fig. 9e presents the segmentation of this image. As it is illustrated in Fig. 9g, it is possible to find two sub-RAGs inside Fig. 9f whose colour values and adjacency relationships are the same that the ones of the template at Fig. 9c. If the template is encoded using a combinatorial map, there is only one possible option for matching on the scene because the green region should include one gray region (an error-tolerant method for sub-map isomorphism should be used (Wang et al., 2011)).

5. Conclusions and future work

This paper presents a new perception-based segmentation approach which consists of two stages: a pre-segmentation stage and a perceptual grouping stage. In our proposal, both stages are conducted in the framework of a hierarchy of successively reduced combinatorial maps. Experimental results conducted on the BSDS300 shows that the performance of the proposed approach is good, although it is still under the values provided by the current state-of-art on the literature: the UCM (Arbeláez, 2006) and, specially, the gPb-owt-ucm (Arbeláez et al., 2011). As these approaches, the described approach represents the whole image at multiple levels of abstraction. This allows the final application to choose the best level of abstraction according to the task to solve. On the contrary, the proposed approach presents an interesting, promising advantage over the referred approaches: it is able to preserve the image topology at all levels of the hierarchy.

Future work will focus on improving the performance of the perceptual grouping stage, including texture descriptors at this stage that emanate from the input image. We will also change the encoding of the edge evidences on the structure, testing the possibility of including information about their orientations (Maire, 2009).

Acknowledgements

We would like to thank the people at PRIP for providing us the source code of the MST Pyramid and for their help and useful comments. We would also like to thank the reviewers for their constructive and detailed comments.

References

Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., Susstrunk, S., 2010. Slic superpixels. EPFL Tech. Report, 149300.

Antúnez E., Marfil R., Bandera A., 2011a, Region correspondence using Combinatorial Pyramids, In: Proc. 1st Workshop on Recognition and Action for Scene Understanding (REACTS2011), SPICUM, pp. 13–24.

Antúnez, E., Marfil, R., Bandera, A., 2011b. A new perceptual-based segmentation approach using Combinatorial Pyramids. Lect. Notes Computer Sci. 6978, 327-336.

Arbeláez, P., 2006. Boundary extraction in natural images using ultrametric contour maps. In: Proc. 5th IEEE Workshop on Perceptual Organization in Computer Vision 1, pp. 82–189.

Arbeláez, P., Maire, M., Fowlkes, C., Malik, J., 2011. Contour detection and hierarchical image segmentation. IEEE Trans. Pattern Anal. Machine Intell. 33 (5), 898–916.

Beaulieu, J.M., Goldberg, M., 1989. Hierarchy in picture segmentation: A stepwise optimization approach. IEEE Trans. Pattern Anal. Machine Intell. (11), 150–163.

Boruvka, O., 1926. O jistem problemu minimalnim. Prace Mor. Prirodved. Spol. v Brne (Acta Societ. Scienc. Natur. Moravicae), 3, p. 1926.

Brox, T., Farin, D., de With, P., 2001. Multi-stage region merging for image segmentation. In: Proc. 22nd Symposium on Information Theory in the Benelux, 189-196, Werkgemeenschap voor Informatie– en Communicatietheorie.

Brun, L., Kropatsch, W., 2001. Introduction to combinatorial pyramids. Lect. Notes Comput. Sci. 2243, 108–128.

Brun, L., Kropatsch, W., 2006. Contains and Inside relationships within combinatorial Pyramids. Pattern Recognit. 39 (4), 515–526.

Brun, L., Pruvot, J., 2008. Hierarchical matching using combinatorial pyramid framework. In: Elmoataz, A., Lezoray, O., Nouboud, F., Mammass, D. (Eds.), Lecture Notes in Computer Science, vol. 5099. Springer, pp. 346–355.

Brun, L., Mokhtari, M., Domenger, J.P., 2003. Incremental modifications on segmented image defined by discrete maps. J. Visual Comm. Image Represent. 14, 251–290.

Brun, L., Mokhtari, M., Meyer, F., 2005. Hierarchical watersheds within the combinatorial pyramid framework. Lect. Notes Comput. Sci. 3429, 34–44.

Burge, M., Kropatsch, W., 1999. A minimal line property preserving representation of line images. Computing 62, 355–368.

Christoudias, C., Georgescu, B., Meer, P., 2002. Synergism in low level vision. In: Proceedings of the 16th International Conference on Pattern Recognition. IEEE, pp. 150–155.

Comaniciu, D., Meer, P., 2002. Mean shift: A robust approach toward feature space analysis. IEEE Trans. Pattern Anal. Mach. Intell. 24 (5), 603–619.

Cour, T., Benezit, F., Shi, J., 2005. Spectral segmentation with multiscale graph decomposition. Proc. IEEE CS Conf. Comput. Vision Pattern Recognition 2, 1124-1131.

Felzenszwalb, P., Huttenlocher, D., 2004. Efficient graph-based image segmentation. Internat. J. Comput. Vision 59 (2), 167–181.

Haxhimusa, Y., Kropatsch, W., 2004. Segmentation graph hierarchies. Lect. Notes Comput. Sci. 3138, 343–351.

Haxhimusa, Y., Ion, A., Kropatsch, W., 2006. Evaluating graph-based segmentation algorithms. Proc. 18th Internat. Conf. on Pattern Recognition, vol. 2. IEEE, pp. 195–198.

Ion, A., Kropatsch, W., Haxhimusa, Y., 2006. Considerations regarding the minimum spanning tree pyramid segmentation method. Lect. Notes Comput. Sci. 4109, 182–190.

Kropatsch, W., 1995. Building irregular pyramids by dual graph contraction. IEEE Proc. Vision Image Signal Process. 142, 366–374.

Lau, H., Levine, M., 2002. Finding a small number of regions in an image using low level features. Pattern Recognition 35, 2323–2339.

Levinshtein, A., Stere, A., Kutulakos, K., Fleet, D., Dickinson, S., Siddiqi, K., 2009. Turbopixels: Fast superpixels using geometric flows. IEEE Trans. on Pattern Analysis Machine Intell. 31 (12), 2290–2297.

Maire, M.R., 2009. Contour Detection and Image Segmentation. PhD Thesis EECS Department, University of California, Berkeley.

Martin, D., Fowlkes, C., Tal, D., Malik, J., 2001. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. Proc. Eighth IEEE Internat. Conf. on Computer Vision, vol. 2, pp. 416–423.

Martin, D., Fowlkes, C., Malik, J., 2004. Learning to detect natural image boundaries using local brightness, color, and texture cues. IEEE Trans. Pattern Anal. Machine Intell. 26 (1), 1–20.

Meyer, F., 2005. Morphological segmentation revisited. Space, Structure and Randomness. Springer.

Najman, L., Schmitt, M., 1996. Geodesic saliency of watershed contours and hierarchical segmentation. IEEE Trans. Pattern Analysis Machine Intell. 18 (2), 1163–1173.

Ren, X., Malik, J., 2003. Learning a classification model for segmentation. Proc. Ninth IEEE Internat. Conf. on Computer Vision, vol. 1, pp. 10–17.

Veksler, O., Boykov, Y., Mehrani, P., 2010. Superpixels and supervoxels in an energy optimization framework. European Conf. on Computer Vision (ECCV), vol. 5. Springer, pp. 211–224.

Wang, T., Dai, G., Xu, D., 2011. A polynomial algorithm for submap isomorphism of general maps. Pattern Recognition Lett. 32, 1100–1107.

Yu, S., Shi, J., 2003. Multiclass spectral clustering. Proc. IEEE Internat. Conf. on Computer Vision, vol. 1, pp. 313–319