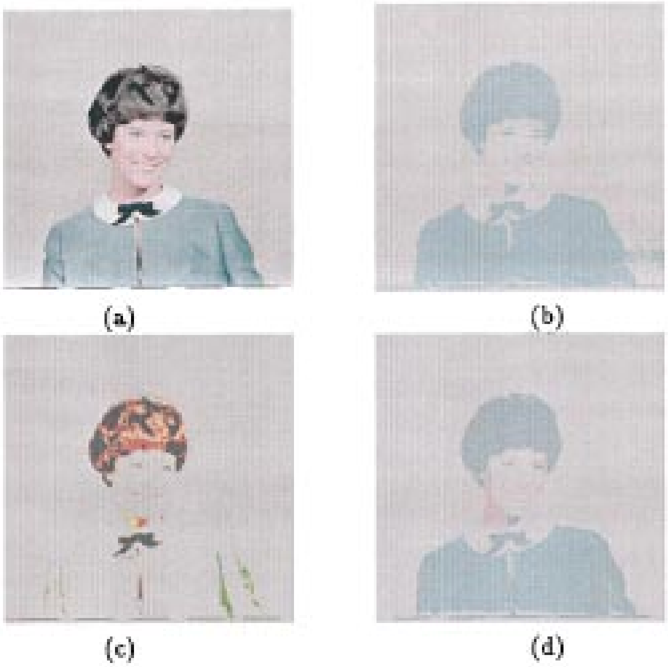


Fig. 4. (a) Original image, (b) result using 1-D fuzzy approach, (c) result using 2-D nonfuzzy approach, and (d) result using proposed 2-D fuzzy approach.

TABLE I
COMPARISONS OF THE COMPUTATIONAL TIME

Images	Size	1D fuzzy	2D Nonfuzzy	2D fuzzy
Tower	512×768	3 sec.	4 sec.	13 sec.
Cornfield	256× 256	8 sec.	3 sec.	14 sec.
Girl	256×256	7 sec.	3 sec.	14 sec.

hair, and face features are all the same. In Fig. 4(c), the dress has been misclassified as the background, therefore, it cannot be distinguished from the background. We use HP DCE/9000 to conduct the experiments, and the computation times are listed in Table I.

IV. CONCLUSIONS

A novel 2-D fuzzy partition characterized by parameters a, b, and c is proposed which divides a 2-D histogram into two fuzzy subsets "dark" and "bright." For each fuzzy subset, one fuzzy entropy and one nonfuzzy entropy are defined based on the fuzziness of the regions. The best fuzzy partition was found based on the maximum fuzzy entropy principle, and the corresponding parameters a, b, and c determines the fuzzy region $[a, c]$. The threshold is selected as the crossover point of the fuzzy region. The experimental results show that the spatial information of pixels should be considered in the selection of thresholds and the 2-D fuzzy approach outperforms the 2-D crisp approach and the 1-D fuzzy approach.

REFERENCES

[1] P. K. Sahoo, S. Soltani, A. K. C. Wong, and Y. Chen, "A survey of thresholding techniques," *Comput. Vis., Graph., Image Process.*, vol. 41, pp. 233–260, 1988.

[2] S. K. Pal and D. K. D. Majumder, *Fuzzy Mathematical Approach to Pattern Recognition*. New York: Wiley, 1986.

[3] C. A. Murthy and S. K. Pal, "Histogram thresholding by minimizing graylevel fuzziness," *Inform. Sci.*, vol. 60, pp. 107–135, 1992.

[4] L. A. Zadeh, "Probability measures of fuzzy events," *J. Math. Anal. Applicat.*, vol. 23, pp. 421–427, 1968.

[5] Y. H. Chen, "Thresholding of gray-level pictures using two-dimensional histograms and fuzzy entropy principle," M.S. thesis, Dept. Comput. Sci., Utah State Univ., Logan, 1997.

[6] H. D. Cheng, J. R. Chen, and J. Li, "Threshold selection based on fuzzy c-partition entropy approach," *Pattern Recognit.*, vol. 31, pp. 857–870, July 1998.

[7] A. D. Brink, "Thresholding of digital images using two-dimensional entropies," *Pattern Recognit.*, vol. 25, pp. 803–808, Aug. 1992.

Regions Adjacency Graph Applied to Color Image Segmentation

Alain Trémeau and Philippe Colantoni

***Abstract*—The aim of this paper is to present different algorithms, based on a combination of two structures of graph and of two color image processing methods, in order to segment color images. The structures used in this study are the region adjacency graph and the line graph associated. We will see how these structures can enhance segmentation processes such as region growing or watershed transformation. The principal advantage of these structures is that they give more weight to adjacency relationships between regions than usual methods. Let us note nevertheless that this advantage leads in return to adjust more parameters than other methods to best refine the result of the segmentation. We will show that this adjustment is necessarily image dependent and observer dependent.**

***Index Terms*—Color image segmentation, region adjacency graph, region growing process, watershed process.**

I. INTRODUCTION

Even if many algorithms are available for color image segmentation [1]–[7], the literature is not as rich as that for grey level images [7], especially when we refer to segmentation algorithms based on region growing processes. Yet, it has been demonstrated that, for several sets of images, region growing processes best perform than clustering or thresholding approaches because they deal with spatial repartition of color information. Region growing algorithms typically start with seed pixels, then iteratively add to regions unassigned neighboring pixels which satisfy one or several homogeneity criteria. Thus, we can define a region as being a set of connected pixels which satisfy some homogeneity criteria. Several criteria linked to color similarity or spatio-color similarity can be used to analyze if a pixel belongs or not to a region [8], [9]. These criteria can be defined according to local, regional and global relationships. In a previous approach, we have shown that three criteria of homogeneity must be used to perform a relevant segmentation [10]. These criteria are as follows:

Manuscript received April 8, 1998; revised September 7, 1999. This work was supported in part by the Région Rhône-Alpes through the ACTIV program. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Robert J. Schalkoff.

The authors are with the Institut d'Ingnierie de la Vision-LIGIV, Universit Jean Monnet, BP 505, 42007 Saint-Etienne Cedex 01, France (e-mail: tremeau@vision.univ-st-etienne.fr; colanton@vision.univ-st-etienne.fr).

Publisher Item Identifier S 1057-7149(00)02684-1.

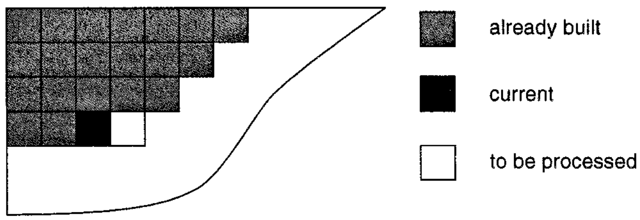


Fig. 1. Region growing process.

- local homogeneity criterion (LHC) which enables to analyze the color difference between two adjacent pixels;
- average homogeneity criterion (AHC1) which enables to analyze the color difference between a pixel and the set of pixels, belonging to a definite region, which are adjacent to this pixel;
- average homogeneity criterion (AHC2) which enables to analyze the color difference between a pixel and the mean value representative of a definite region.

More precisely, we can consider that a pixel could be associated to a region R_i if

1) it is similar enough to at least A_p adjacent pixels belonging to the region R_i;
2) it is similar enough to the mean value of the set of adjacent pixels belonging to the region R_i;
3) it is similar enough to the mean value representative of the region R_i.

Indeed, we have shown in [10] that these three criteria are really complementary and absolutely necessary to minimize errors of region assignment. Nevertheless, we can note that the adjustment of the A_p parameter is not easy to define because it is image dependent. We have led without success several tests to adjust automatically this parameter according to images studied. Most of the time the segmented images was satisfying, nevertheless for some sets of images, such as those illustrated in this article, we have obtained either an under-segmentation, or an over-segmentation. This can be explained by the fact that these images are not as simple as they seem to be [see Fig. 3(b) and Fig. 13(a)]. That is the reason why we have searched to define, in this study, other algorithms more robust and more relevant, especially to analyze spatial relationships between adjacent regions.

For most of region growing processes growing goes too far, consequently growing processes have to use a de-growing stage to recover optimal regions. Rather than using a post-process, we propose to use growing criteria, more sophisticated than those previously introduced, which take into account the color contrast which separate two adjacent regions.

Let X denotes the grid of sample point of a picture, and P be a logical predicate defined on a set of contiguous picture points. Then we can define the segmentation as being a partition of X into disjoint non-empty subsets $R_1, \cdots, R_N$ such as

1) $\bigcup_{i=1}^{N} R_i = X$;
2) $R_i \; i = 1, \cdots, N$ is connected;
3) $P(R_i) = TRUE$ for $i = 1, \cdots, N$;
4) $P(R_i \bigcup R_j) = FALSE$ for $i \neq j$ where R_i and R_j are adjacent.

Let us note that the predicate P associated both to the third and to the fourth conditions determines what kind of properties the segmented regions should have, for example an homogeneous color distribution. The second condition implies that regions must be connected, i.e., composed of contiguous lattice points. This requirement is very important; it affects the central structure of segmentation algorithms, especially region growing processes, because the sequence in which picture points are processed often proceeds according to neighboring relationships [9].

It is essential to note that the adjacency relationships between regions are not really taken into account in this definition, at the exception of the fourth predicate which stipulates that two adjacent regions can not be similar. Most of segmentation processes do not give enough weight to this last predicate because they only take into account local neighboring relationships. As they do not integrate adjacency relationships between regions, they do not perform as well as we would wish. In order to compensate this lack, we suggest to use region adjacency graphs.

We will see in the Section III how to process a segmentation from the region adjacency graph resulting of a coarse pre-segmentation. Before that, we will introduce, in Section II, the two segmentation processes that we have used in our study to pre-segment a color image. In Section IV, we will give some examples to show the effectiveness and the performance of the proposed two-step process. These examples will point out the improvements that region information may provide to color image segmentation. Several figures will illustrate the different configuration alternatives that we have tested in this study. These alternatives are linked to the selection of one process, among the two proposed, to perform the pre-segmentation and to the selection of an another process to perform the final segmentation. Finally, conclusions will be drawn in Section V.

II. First Step of the Segmentation Process

In order to coarsely pre-segment the image, we propose to use either a region growing process, either a watershed process based on the computation of the image of local second order moments.

A. Region Growing Process (RGP)

The principles of region growing is as follows. At a given step of the procedure, we have (see Fig. 1)

- R set of pixels already gathered in the region under study;
- V set of pixels, neighboring the current one, already gathered in the region under study;
- P next pixel to be processed.

Considering the three merging predicates previously introduced in the introduction, we can consider in first approximation that a pixel P can be merged with one of its neighbors $P' \in V$ if

1) $d^2(\vec{c_p}, \vec{c_{p'}})$ is sufficiently small;
2) $d^2(\vec{c_p}, \vec{\mu_V})$ is sufficiently small;
3) $d^2(\vec{c_p}, \vec{\mu_R})$ is sufficiently small;

where $\vec{\mu_V}$ (resp. $\vec{\mu_R}$) is the mean of the set of colors $\{\vec{c_0}, \cdots, \vec{c_N}\}$ belonging to the neighborhood V (resp. in the region being process R) (see Fig. 1)

$$\vec{\mu_V} = \frac{1}{N_V} \sum_{(x,\,y) \in V} \vec{c}(x,\,y)$$

$$\vec{\mu_R} = \frac{1}{N_R} \sum_{(x,\,y) \in R} \vec{c}(x,\,y).$$

Region growing algorithms typically scan the image, in some predetermined manner, such as left to right and from top to bottom, in order to compare the current pixel to the already existing but not necessarily completed neighboring segment [8]. If the value of this pixel and the values of the segment are closed enough then the pixel is added to the segment and this latter is updated. If there is more than one region which is closed enough then it is added to the closest region. If no neighboring region is closed enough then a new segment is defined. This process is next reiterated from the next pixel until scanning all the image. One of the disadvantages of such region growing algorithms is

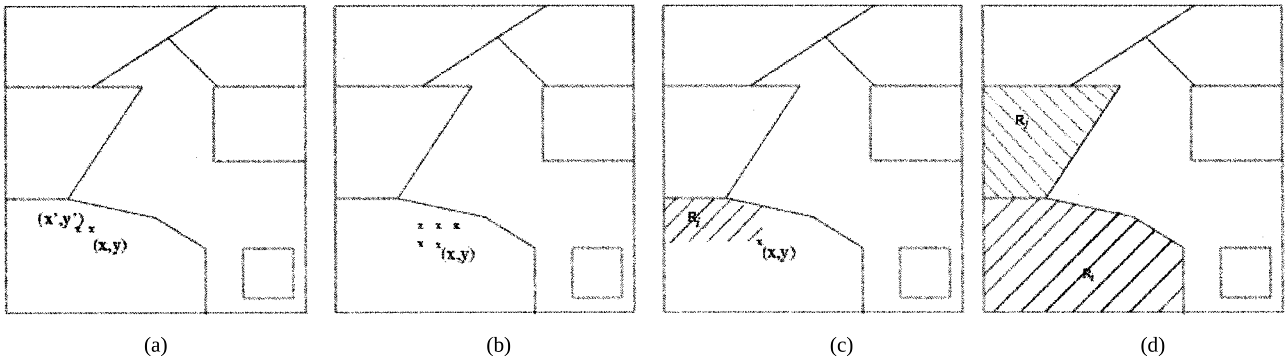


Fig. 2. Computation of color differences in reference to adjacency relationships between image elements. (x, y) represents the next pixel to be processed: (a) between two adjacent pixels, (b) between a pixel and neighborhood pixels belonging to the same region, (c) between a pixel and a region being processed, and (d) between two adjacent regions.

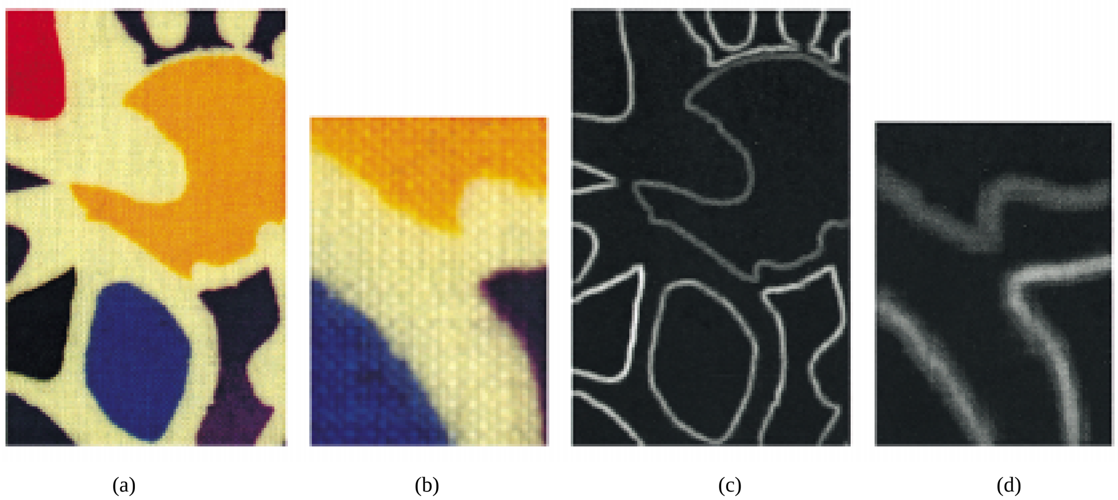


Fig. 3. Image tested (of size 230×360) and image of locally computed variances (with a mask 7×7): (a) original image, (b) area zoomed $\times 4$, (c) image of contours, and (d) area zoomed $\times 4$.

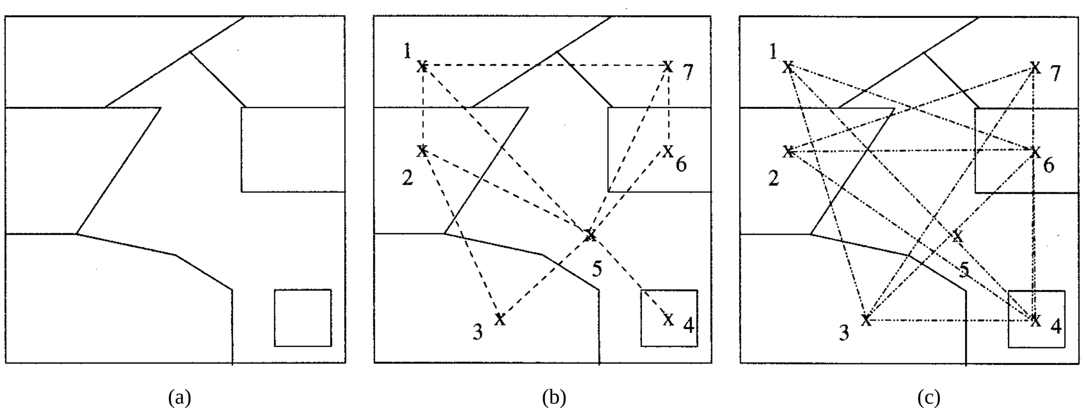


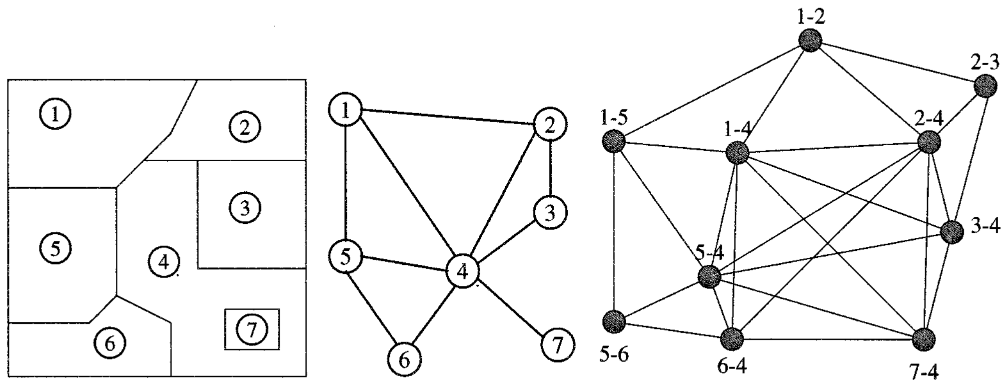
Fig. 4. The two first levels of the RAG: (a) original image, (b) "simple-connectivity relationships," and (c) "second-order connectivity relationships."

that they are inherently sequential. The regions produced depend both on the order in which pixels are scanned and on the value of pixels first scanned and gathered to define each new segment. Rather than scanning one image from left to right and from top to bottom, it may be more suitable to scan the image in all directions from seeds which best characterize regions, such as centroid seeds, as this is done with watershed algorithms. The most important aspect of these algorithms consists then in localizing these seeds. We will illustrate this notion with examples in Section IV.

B. Watershed Process

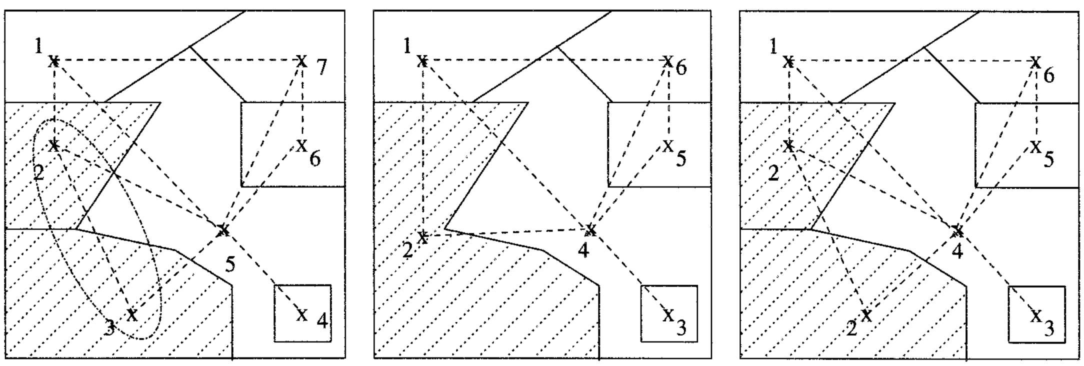
The watershed transformation is a morphological tool defined to segment image, it can also be seen as a region growing process. In the watershed transformation pixel values are considered as topographic data characteristic of a relief; the value of each pixel denotes then the elevation of the point [11], [12].

There are different watershedding approaches in the literature ranking from iterative methods to sequential, these latter gather



(a) Region partitionning. (b) Region Adjacency Graph. (c) Associated Line Graph.

Fig. 5. Spatial views of region relationships.



(a) Vertices to be merged. (b) Merging of 2 vertices. (c) Gathering of 2 vertices with the same label.

Fig. 6. Simplification of the RAG after having merged two adjacent regions.

arrowing and flooding methods. The iterative methods represent the earliest form used for performing a watershedding. The algorithms using this principle are very slow and not wholly accurate. Sequential methods are based on a pixels predefined order scanning according with each new value given to a pixel is immediately taken into account in the processing of subsequent pixels. That is the reason why these algorithms are faster and produce more accurate watersheds than iterative methods. However, the localizing of the watershed lines may not be centered on the ridge points of the image. The flooding watershedding methods are the fastest and the most accurate of all watershed methods. By definition, these methods process according to the principle of a slow immersion of the grey-scale landscape in an imaginary body of water. There are several flooding simulation algorithms [11], [12]. In our study, we have performed the watershed transformation from the local moments image by implementing the Vincent and Soille algorithm, which is described in [11].

C. Image of Local Second Order Moments

Several investigations have been done to provide a good pre-segmentation which will differentiate homogeneous regions from textured and noisy regions. While there are many efficient methods for grey-tone images, problems are encountered with color images because regions partitioning depends both of color and spatio-color distributions [13]. Nevertheless, we have demonstrated in a previous work that statistical features based on local and contextual properties could be used with success to pre-segment a color image [13]. Moreover, we have also shown that the second-order moment (the variance) was the most interesting of these features because it is easier and quicker to compute and gives more relevant results [14]. This feature can be processed simply by a linear combination of higher order prefix sums which can be obtained by iterating the prefix sum computation on previous prefix sums. Indeed, by applying this prefix summation method to the two dimensions of the picture, we obtain a fast process to realize the computation of a two-dimensional moment [14].

This feature can be defined locally in each pixel (x, y), according to a 7×7 neighborhood mask $V(x, y)$ centered in (x, y), by

$$\sigma^2_{(x,y)} = \frac{1}{N_V} \sum_{(x', y') \in V(x, y)} \left\| \vec{c}(x', y') - \vec{\mu}_{V(x,y)} \right\|^2$$

where $\vec{\mu}_{V(x,y)}$ is the mean color value of the set of pixels neighboring (x, y)

$$\vec{\mu}_{V(x,y)} = \frac{1}{N_V} \sum_{(x', y') \in V(x, y)} \vec{c}\,(x', y').$$

This feature can be computed in each pixel; it performs then an image relatively closed to the image of contours (see Fig. 3). Rather than using directly this image to perform the pre-segmentation, we have refined this latter from a watershed process in order to better taking into account spatio-color relationships between pixels.

D. Computation of the Watershed Process from Local Moments Image

To compute the watershed transformation (with the Vincent and Soille algorithm [11]), we have scanned and partitioned all pixels of the contours image, considering successively all grey-scale values, according to a two-step process: a sorting step followed by a flooding

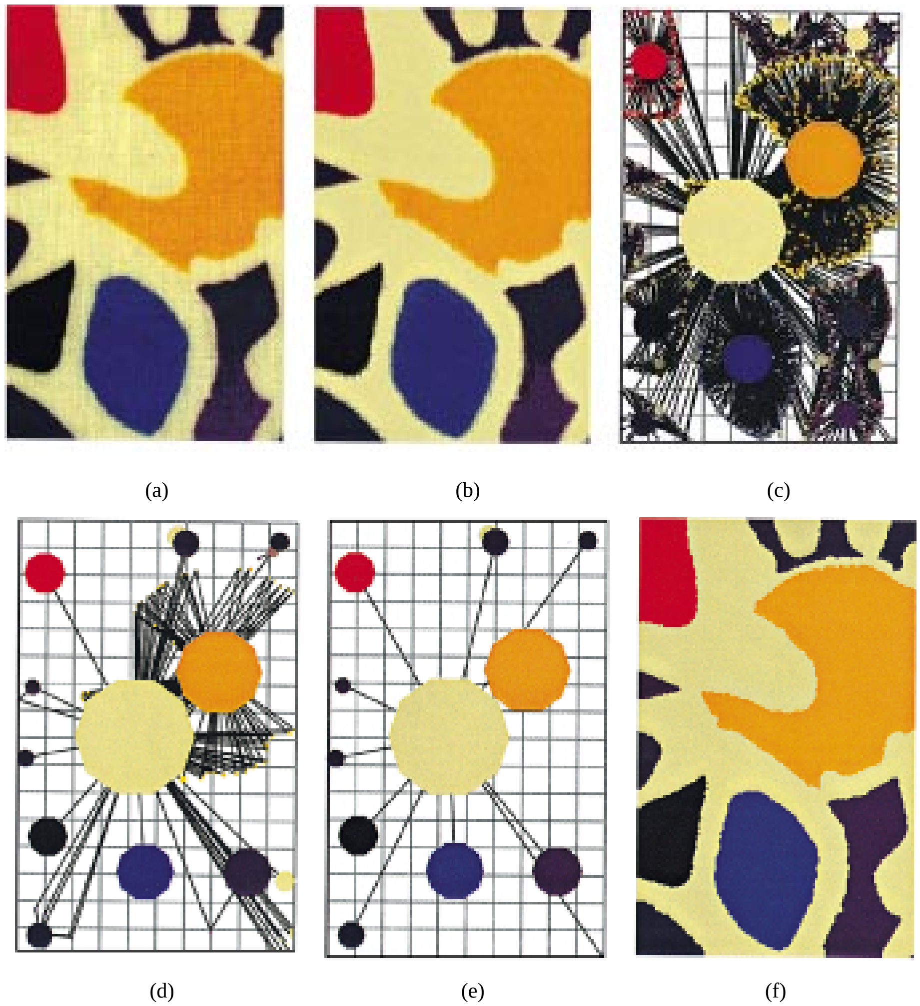


Fig. 7. Partitioning of RAG from vertices growing process: (a) original image, (b) image presegmented from growing process, (c) RAG of the presegmented image, (d) intermediate result, (e) final result, and (f) final image.

step. The sorting step produces a list of the pixels sorted according to an ascending grey-values scale. Once the pixels have been sorted, it then becomes possible to gather them according to successive grey-scale levels, as if a flood was progressing up through the levels of the image. We note that this algorithm is not directly optimal because it produces an incomplete watershed for pixels which are at the boundary of different minima. To be relevant, this algorithm needs to be followed by a second process which first detects pixels which are at the boundary of different minima and second adjusts their watershed value to the most relevant minima. This has been done by analyzing the pixels of contours relatively to homogeneity criteria previously defined.

III. Second Step of the Segmentation Process

The second step of our segmentation process consists of refining the pre-segmentation previously obtained. In order to give, to the second step, more weight to adjacency relationships between regions, than to the first step, we have first computed the region adjacency graph corresponding to the partition given by the first step, then we have simplified this graph by merging regions which need it.

A. Region Adjacency Graph

One of the advantages of the region adjacency graphs (RAG) [15] is that they provide a "spatial view" of the image [1]. One way to represent

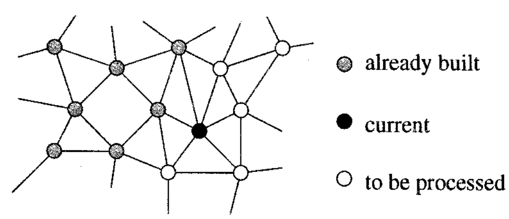


Fig. 8. Vertices-graph growing process.

a region adjacency graph consists of associating a vertice at each region and an edge at each pair of adjacent regions. By definition, this region adjacency graph provides a "simple-connectivity view" of the image, such as this illustrated in Fig. 4(b).

Beyond this "simple-connectivity view," this graph gives also a "high-level connectivity view" of the image. Thus, considering the Fig. 4, we can observe that regions 1 and 2 are adjacent, likewise regions 2 and 3 are adjacent, that leads us to note that regions 1 and 3 present a "second-order connectivity" relationships. In fact the "simple-connectivity view" contains inherently all "high-level connectivity relationships" of the image, moreover it can be considered as the dual of the regions partitioning of the image (see Fig. 5).

At each vertice v_i corresponds a region R_i and two color values $\vec{\mu_i}$ and σ_i representatives of the color distribution of this region.

Fig. 9. RAG of presegmented image from growing process or watershed process: (a) image presegmented in 2500 regions from region growing process, (b) RAG of the presegmented image from region growing process, (c) image presegmented in 1540 regions from watershed process, and (d) RAG of the presegmented image from watershed process.

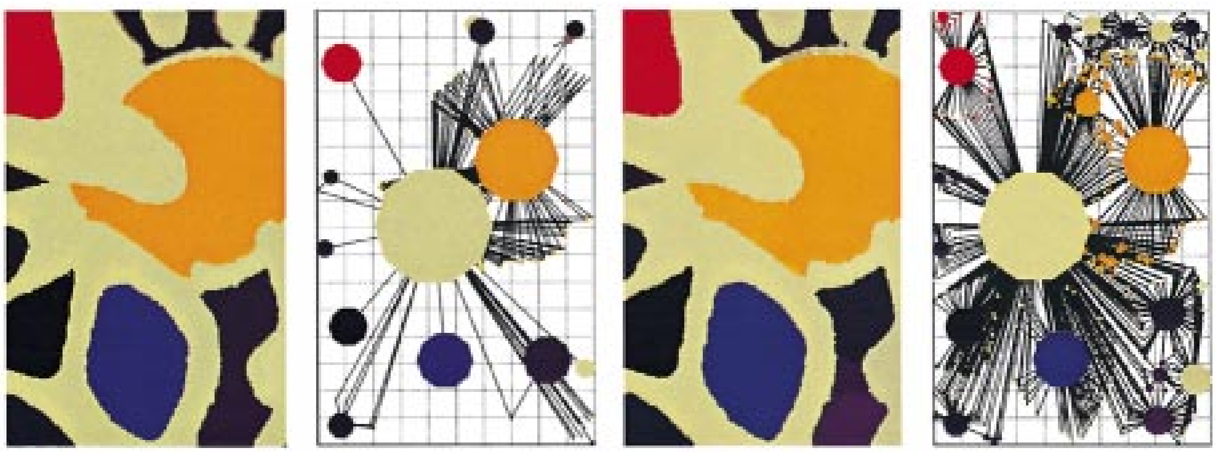


Fig. 10. Different line graph segmentations after a presegmentation by a region growing process: (a) 115 regions from the line graph growing process, (b) RAG of the segmented image from the line graph growing process, (c) 540 regions from the line graph watershed process, and (d) RAG of the segmented image from the line graph watershed process.

At each edge e_{ij} corresponds a pair of adjacent regions $\{R_i, R_j\}$ and a color distance $d^2(R_i, R_j)$ which can be used to compare the color distribution of these two regions [see Fig. 2(d)].

$$d^2(R_i, R_j) = d^2(\vec{\mu_i}, \vec{\mu_j}).$$

In this study, we have used the Fisher distance as was done in [1]. Thus

$$d^2(\vec{\mu_i}, \vec{\mu_j}) = \frac{(N_{R_i} + N_{R_j})\,\|\vec{\mu_i} - \vec{\mu_j}\|^2}{N_{R_i}\sigma_i^2 + N_{R_j}\sigma_j^2} \qquad \text{if } \sigma_i \neq 0 \text{ and } \sigma_j \neq 0$$

$$d^2(\vec{\mu_i}, \vec{\mu_j}) = \|\vec{\mu_i} - \vec{\mu_j}\|^2 \qquad \text{if } \sigma_i = 0 \text{ and } \sigma_j = 0.$$

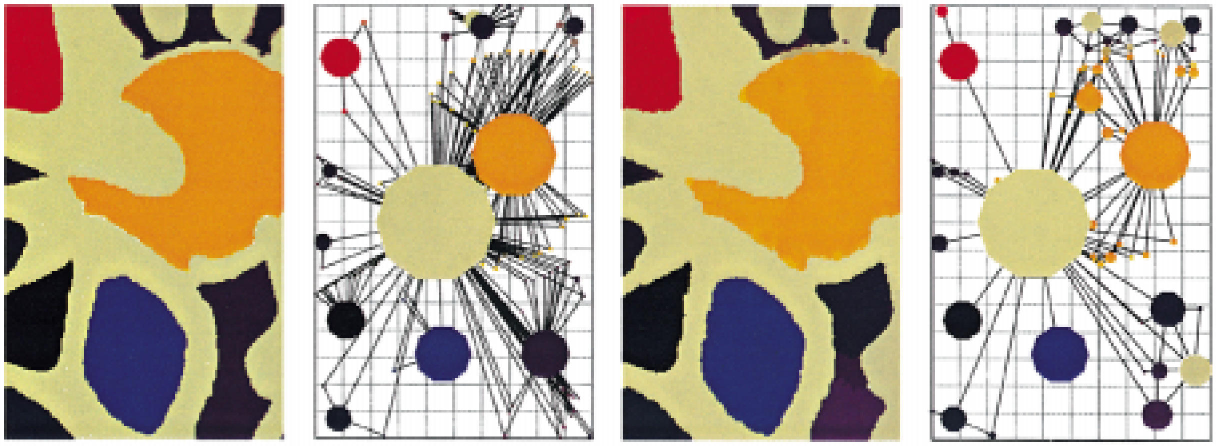
Considering the fourth merging predicate previously introduced, we can consider that a region R_i can not be merged with an another region R_j if $d^2(R_i, R_j)$ is sufficiently high.

B. Partitioning of the RAG

The region adjacency graph can be used to merge adjacent regions provided that these regions have a color distribution sufficiently closed. This merging introduces then a reorganization of the region adjacency graph which can be managed (see Fig. 6) by merging the vertices v_i and v_j into one vertice v_k and by merging the corresponding edges or by gathering the vertices v_i and v_j into one set of vertices $\{v_i, v_j\}$ having the same label v_k.

This merging provides *a fortiori* a new partition of the image plane. Rather than simplifying the RAG by merging identical vertices, we have retained the second method which consists of associating at each vertice a label and of managing these labels according to operations under process. This second method enables to preserve the initial structure of the graph (the only difference from one operation to the other concerns label values), in order to come back if necessary to previous partitioning(s) before last merging(s). Nevertheless, so as to better display the result of each operation on the RAG, our illustrations represent each set of vertices by only one vertice and all edges associated at each set by the corresponding edges, as for the merging method. This process gives a better “spatial view” of the partitioning under process (see Fig. 7).

In order to perform the merging of regions of the RAG which need it, we have developed two algorithms: vertices-graph growing process and vertices-graph watershed process. These two algorithms are based on the same rules than the two image partition algorithms previously



(a) (b) (c) (d)

Fig. 11. Different segmented images after a presegmentation by region growing process: (a) 135 regions from vertices graph growing process, (b) RAG of the segmented image from vertices graph growing process, (c) 70 regions from the line graph watershed process, and (d) RAG of the segmented image from the line graph watershed process.

TABLE I
RUN TIMES OF EACH TWO-STEP ALGORITHMS COMBINATION ON A SUN ULTRA SPARC 1. EACH ALGORITHM INCLUDES FIRST A PRE-SEGMENTATION PROCESS (RGP OR WP), NEXT AN ADJACENCY-GRAPH COMPUTATION PROCESS (RAG OR LG), LAST A SEGMENTATION PROCESS (RGP OR WP) BASED ON A GRAPH PARTITIONING PROCESS (RGP OR WP)

Algorithm		Execution Time of each step (s)		Image size
step 1	step2	step 1	step2	
RGP	RAG + RGP	4.1	2.2 + 0.4	230 × 360
RGP	RAG + WP	4.1	2.2 + 0.6	
WP	RAG + RGP	11.6	1.0 + 0.2	
WP	RAG + WP	11.6	1.0 + 0.3	
RGP	LG + RGP	4.1	7.7 + 1.0	230 × 360
RGP	LG + WP	4.1	7.7 + 2.0	
WP	LG + RGP	11.6	3.3 + 0.6	
WP	LG + WP	11.6	3.3 + 1.2	
RGP	RAG + RGP	16.8	6.5 + 1.0	591 × 591
RGP	RAG + WP	16.8	6.5 + 1.4	
WP	RAG + RGP	62.0	3.5 + 0.4	
WP	RAG + WP	62.0	3.5 + 0.7	
RGP	LG + RGP	16.8	24.5 + 4.3	591 × 591
RGP	LG + WP	16.8	24.5 + 7.0	
WP	LG + RGP	62.0	13.4 + 3.1	
WP	LG + WP	62.0	13.4 + 5.2	

introduced in Section II-A and Section II-B, except that they apply to vertices and not to pixels (see Fig. 8).

C. Vertices-Graph Growing Process

At each vertice v_i corresponds a region R_i. Let us note V the set of vertices v_j which have a "simple-connectivity relationship" to the vertice v_i, i.e., the vertices v_j which have an edge e_{ij} which links them directly to the vertice v_i. Then, we can consider that this set of vertices defines the set of neighbors of vertice v_i (see Fig. 8), that is equivalent to say that the set of regions R_j surrounding the region R_i defines the set of neighbors of region R_i.

Considering the four merging predicates previously introduced; we have postulated that a vertice v_i could be merged with one of its neighbor $v_j \in V$ if

1) $d^2(R_i, R_j)$ is sufficiently small;
2) $d^2(R_i, V)$ is sufficiently small;
3) $d^2(R_i, S)$ is sufficiently small;

where S is the set of vertices being gathered which contains the vertice v_j.

These three conditions determine the good arrangement of vertices in sets of nearly similar vertices, that implies also regions underlying of vertices. Thus, the first condition implies that two adjacent regions can be merged only if they are closed enough. Likewise, the second and the third condition stipulate that this merging can not be performed at the disadvantage of sets of regions already done. That is to say local and global homogeneity criteria prevail to punctual homogeneity criterion. These three criteria enable us to gather the current region to a set of regions being processed, not because only one region of this set is adjacent and sufficiently similar to the current one, but above all because most of adjacent regions already belonging to the set of regions being processed, are also similar to the current one. Thanks to this rule, we give more weight to contextual connectivity relationships than to punctual connectivity relationships, so regions gathering is more consistent.

D. Line Graph of the Region Adjacency Graph

Another way to study regions connectivity relationships consists of analyzing the line graph (LG) of the RAG (see Fig. 5). The line graph G' of a graph G is defined as follows:

- at each vertice of G' corresponds an edge of G;
- at each edge of G' corresponds a pair of adjacent vertices in G'.

This structure enables to compare each pair of adjacent regions $\{R_i, R_j\}$ to any other pair of adjacent regions $\{R_j, R_k\}$ which have a region R_j in common. At each region R_i corresponds a color value $\vec{\mu}_i$ and a size parameter s_{R_i}. Consequently, at each vertice v_{ij} of G' corresponds a pair of adjacent region $\{R_i, R_j\}$, a color distance $d_{ij} = d^2(R_i, R_j)$, and a weight $w_{ij} = \min(s_{R_i}, s_{R_j})/(s_{R_i} + s_{R_j})$, functions of adjacent regions R_i, and R_j [1]. Other weight parameters, such as spatio-color parameters defined in [1], could also be associated to these vertices. Nevertheless, we have observed that, in our case of study, these parameters do not enough increase the effectiveness of our segmentation algorithms, inversely they substantially increase their computing time.

Likewise, at each edge e_{ijk} of G' corresponds a pair of color distances (d_{ij} and d_{jk}) and a pair of weights (w_{ij} and w_{jk}), functions of the two pairs of adjacent regions $\{R_i, R_j\}$ and $\{R_j, R_k\}$. Then, considering the previous predicates used to perform the vertices-graph growing process and the rules of organizing of the line-graph, we can postulated that a region R_j could be merged to an adjacent region R_i,

(a) (b) (c) (d)

Fig. 12. Different segmentations linked to the value of growing threshold used at the presegmentation stage: (a) 2500 regions linked to a low growing threshold, (b) segmentation in 115 regions, (c) 540 regions linked to a high growing threshold, and (d) segmentation in 50 regions.

and not to any other adjacent region R_k, if $d^2(R_i, R_j)$ is minimal and $\min(s_{R_i}, s_{R_j})/(s_{R_i} + s_{R_j})$ is maximal.

Consequently, one vertice v_{jk} of G' could be merged to an adjacent vertice v_{ij} of G' if the color distance associated to the vertice v_{jk} is sufficiently similar to the color distance associated to the vertice v_{ij}, that is to say $d^2(R_j, R_k)$ is closer to $d^2(R_i, R_j)$ than any other $d^2(R_j, R_l)$, whereas $d^2(R_i, R_j)$ is minimal.

This merging introduces then a re-organization of the line-graph which provides *a fortiori* a new partition of the image plane. This process can be iterated sequentially, such as a line-graph growing process for which seeds will be the vertices of minimal color distance and of maximal weight.

This line-graph can be considered as the edge graph of the region adjacency graph (i.e., the dual of the region adjacency graph) (see Fig. 5). Consequently, this *line-graph growing process* can be seen as the dual of the *region adjacency graph growing process* previously defined.

From a computational point of view the line graph appears more sophisticated and more complex to realize than the region adjacency graph because it contains more information on region relationships. We will analyze in the following section if this representation improves, substantially or not, the effectiveness of the two proposed segmentation processes, without increasing too much their performance in terms of execution times.

IV. Results and Discussion

By definition, each image area which verifies the following segmentation predicate can be considered as a region. In this study, we have retained two sets of three homogeneity criteria to define our segmentation predicate

$$1.\ d^2(\vec{c_{v_i}}, \vec{c_{v_j}}) \leq d_1,\ \ 2.\ d^2(\vec{c_{v_i}}, \vec{\mu_V}) \leq d_2,\ \ 3.\ d^2(\vec{c_{v_i}}, \vec{\mu_S}) \leq d_3$$
$$1.\ d^2(R_i, R_j) \leq d'_1,\ \ 2.\ d^2(R_i, V) \leq d'_2,\ \ 3.\ d^2(R_i, S) \leq d'_3.$$

The first set of criteria has been defined to pre-segment the image, whereas the second set has been used to refine the segmentation from the region adjacency graphs.

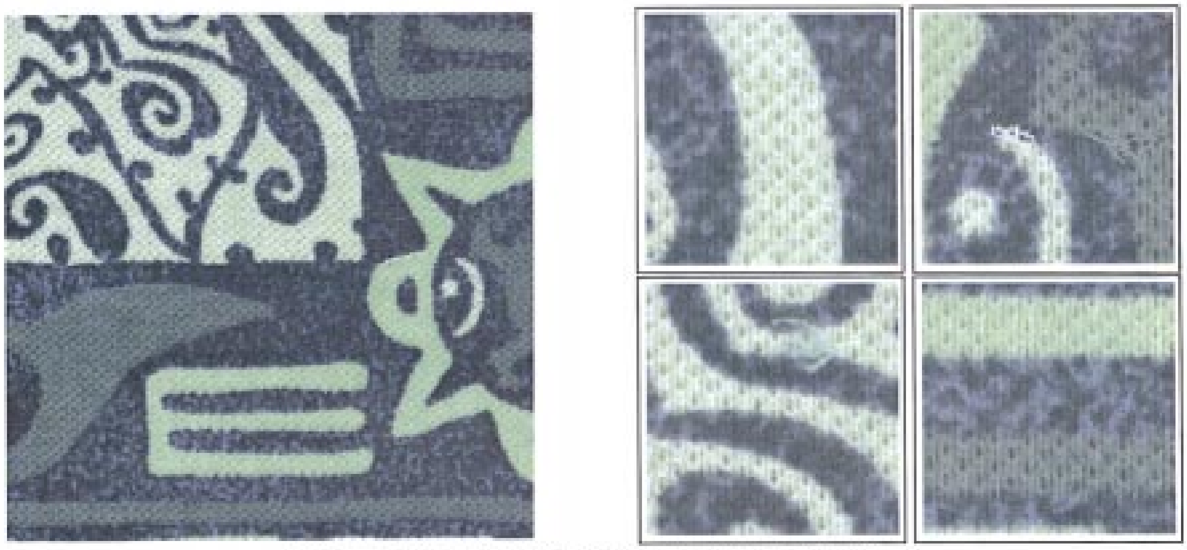
The main difficulty which we are faced consists of weighting the three parameters associated to these three criteria. In order to perform a sufficiently fine pre-segmentation of the image, during the first step of our algorithm we have used relatively low homogeneity parameters. Our aim was to not forget any image area which could be considered as a component key to define a region and to not gather by error two adjacent, and relatively similar, areas which could generate two different regions. Inversely, for the second step of our algorithm, we have used relatively high homogeneity parameters to merge adjacent, and relatively similar, regions. Thus, meanwhile the first step of the process multiplies the number of related areas, which can be considered as component keys to define a region, the second step enables to gather these areas in a single region. Likewise, meanwhile the first step of the process minimizes errors of pixels assignments, the second step minimizes errors of areas assignments. That justifies that this two-step process is more adapted to minimize the errors of partitioning than other commonly used sequential scanning and merging processes which operate pixel by pixel.

Four kinds of two-step processes have been proposed in previous sections. They are based on the combination of two segmentation processes: the first process performs a pre-segmentation from pixels gathering and the second process refines this segmentation from region gathering. The two segmentations algorithms used to perform these gatherings are: the Region Growing Process (RGP) and the Watershed Process (WP). Whereas during the first step of the process these algorithms perform on pixels, during the second step these algorithms perform on regions previously defined and organized according to their adjacency relationships. The two data structures used to describe these adjacency relationships are the RAG LG.

Obviously all these algorithms combinations in a two-step process do not give the same segmentation (see Figs. 9–11). As example, consider the result of the first step in terms of image segmentation, this relatively to the image illustrated by Fig. 3. With the watershed process we have obtained a set of regions which are quite homogeneous in size [see Fig. 9(d)], this is due to the content of the image studied and to the homogeneity criteria used. Inversely, with the region growing process, we have obtained a set of regions which are quite different in size [see Fig. 9(b)]. When these regions represent noisy areas they are quite small, inversely when these regions represent homogeneous areas, where contours are well defined, they are quite big. Indeed the region growing process is by definition more sensitive to noise than the watershed process. Consequently, according to the content of the images to be segmented, we will use either the former process either the latter process, to best perform the pre-segmentation step.

Likewise, the same kind of analysis can be done to study the effectiveness of the second step of our process in terms of image segmentation, except that this time homogeneity criteria proceed with sets of regions and not with sets of pixels.

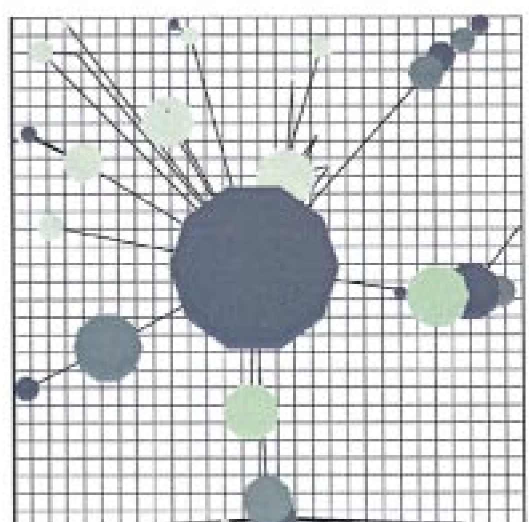
Table I gives the execution times of RGP and WP algorithms combinations for the two images illustrating this study. These execution times



(a) Image studied of size 591 × 591 (some areas are zoomed)

(b) Image segmented from the two steps method proposed.

(c) Image of contours of Fig.13(a)



(d) RAG of the image segmented

Fig. 13. Image with interconnected overlapped regions: (a) image studied of size 591×591 (some areas are zoomed), (b) image segmented from the two steps method proposed, (c) image of contours of (a), and (d) RAG of the image segmented.

also relate the computation times of graphs (RAG or LG) resulting from the pre-segmentation.

The aim of our study was to state different algorithms combinations to improve significantly the subjective quality of two commonly used color image segmentation processes. In order to demonstrate that this improvement is not expansive in terms of performance, the execution times given for each algorithms combination correspond to the best segmentation obtained with these algorithms, that is to say these times do not relate the subjective adjustment of study parameters before processing the best segmentation. It is useful to note that parameters adjustment is image dependent (see Fig. 12) and observer dependent. Of course they can be adaptively adjusted according to contextual and local criteria [10], but that is not sufficient. As the quality of a segmentation can not be objectively described, we have asked to ten subjects to evaluate the quality of segmentations resulting of our parameters adjustments. The best segmentations are those corresponding to these subjective assessments. These considerations justify why these algorithms combinations can not be compared with other existing algorithms except in terms of execution time or in terms of subjective assessment. We have already shown in Section I why these algorithm combinations could improve the quality of a segmentation process, so we do not come back on this aspect. To be convinced of this improvement, one can subjectively compare the best image obtained by the pre-segmentation process (which corresponds to a commonly used segmentation process) to the best image obtained at the end of the two-step segmentation process. As an example, we can compare Fig. 7(f) to other figures illustrating segmentations obtained by RGP or WP processes (see also Fig. 13).

V. Conclusion

The aim of this study was to present different algorithms based on a combination of different structures of graph and of different color image processing methods in order to segment color images. The structures tested in this article are the region adjacency graph and its line-graph associated. We have seen how these structures can enhance segmentation processes such as region growing or watershed transformation. The principal advantage of these structures is that they enable to give more weight to adjacency relationships between regions than usual methods. We have nevertheless observed that this advantage led in return to adjust more parameters than others methods to better refine the result of the segmentation. We have also shown that this adjustment was necessarily image dependent and observer dependent.

In order to show the effectiveness and the performance of these two-step algorithms, we have given different examples. These examples have point out the improvements that region information may provide for color image segmentation. Likewise, several figures have been done to illustrate the different configuration alternatives that we have tested in this study. These alternatives are linked to the selection of one process, among the two proposed, to perform the pre-segmentation and to the selection of an another process to perform the final segmentation.

References

[1] R. Schettini, "A segmentation algorithm for color images," *Pattern Recognit. Lett.*, vol. 14, pp. 499–506, June 1993.
[2] M. Celenk, "A color clustering technique for image segmentation," *Comput. Vis. Graph. Image Process.*, vol. 52, pp. 145–170, 1990.
[3] Y. I. Ohta, T. Kanade, and T. Sakai, "Color information for region segmentation," *Comput. Graph. Image Process.*, vol. 13, pp. 222–241, 1980.
[4] S. Tominaga, "Color classification of natural color images," *Color Res. Applicat.*, vol. 17, pp. 230–239, Aug. 1992.
[5] J. Gauch and C. W. Hsia, "A comparison of three color image segmentation algorithms in four color spaces," in *Proc. SPIE*, vol. 1818, 1992, pp. 1168–1181.
[6] F. Ferri and E. Vidal, "Color image segmentation and labeling through multiedit-condensing," *Pattern Recognit. Lett.*, vol. 13, pp. 561–568, Aug. 1992.
[7] N. R. Pal and S. K. Pal, "A review on image segmentation techniques," *Pattern Recognit.*, vol. 26, pp. 1277–1294, Sept. 1993.
[8] R. M. Haralick and L. Shapiro, "Image segmentation techniques," *Comput. Vis. Graph. Image Process.*, vol. 29, pp. 100–132, 1985.
[9] S. W. Zucker, "Region growing: Childhood and adolescence," *Comput. Graph. Image Process.*, vol. 5, pp. 382–399, 1976.
[10] A. Trémeau and N. Borel, "A region growing and merging algorithm to color segmentation," *Pattern Recognit.*, vol. 30, pp. 1191–1203, July 1997.
[11] L. Vincent and P. Soille, "Watersheds in digital spaces: An efficient algorithm based on immersion simulations," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 13, pp. 499–506, June 1991.
[12] S. Beucher and F. Meyer, "The morphological approach to segmentation: The watershed transformation," in *Mathematical Morphology in Image Processing*, E. R. Dougherty, Ed. New York: Marcel Dekker, 1993, ch. 12, pp. 433–481.
[13] A. Trémeau, P. Colantoni, and B. Laget, "On color segmentation guided by the cooccurrence matrix," in *OSA Annu. Conf. Optics Imaging Information Age*, Rochester, NY, Oct. 1996, pp. 30–38.
[14] F. Zhou and P. Kornerup, "Computing moments by prefix sums," in *IEEE Int. Conf. Image Processing*, vol. 3, Lausanne, Switzerland, Sept. 1996, pp. 619–622.
[15] T. Pavlidis, *Structural Pattern Recognition*, New York: Springer, 1980.