Proceedings of the 8th International Symposium on Mathematical Morphology,
Rio de Janeiro, Brazil, Oct. 10–13, 2007, MCT/INPE, v. 1, p. 375–386.
http://urlib.net/dpi.inpe.br/ismm@80/2007/03.20.00.50

Oversegmentation control for inexact graph matching: First results

Luís A. Consularo[1], Roberto M. Cesar-Jr[2], Luiz H. Figueiredo[3] and Isabelle Bloch[4]

[1] *Universidade Metodista de Piracicaba (UNIMEP), SP, Brazil*
laconsul@unimep.br

[2] *Departmento de Ciência da Computação, Instituto de Matemática e Estatística (IME), Universidade de São Paulo, SP, Brazil*
cesar@ime.usp.br

[3] *Instituto de Matemática Pura e Aplicada (IMPA), Rio de Janeiro, RJ, Brazil*
lhf@impa.br

[4] *Ecole Nationale Supérieure des Télécommunications (GET - Télécom Paris), CNRS UMR 5141 LTCI, Paris, France*
Isabelle.Bloch@enst.fr

Abstract An interactive image segmentation method based on structural pattern recognition has been recently introduced. A model graph is generated from an oversegmentation of the image and from traces provided by the user. An input graph is generated from the oversegmented image. Image segmentation is then obtained by matching the input graph to the model graph. An important problem that should be addressed is how to control the size of the input graph. This size is given by the number of regions provided by the oversegmentation. To address this problem, we propose to control the maximum number of regions provided by the oversegmentation by using watershed with markers. The markers are given automatically by using two approaches: quadtrees and centroidal Voronoi diagrams. Results on real images are discussed.

Keywords: inexact graph matching, oversegmentation control, image segmentation quadtrees centroidal Voronoi diagrams.

1. Introduction

Image segmentation is a key problem in most situations in image processing, analysis and computer vision. From the mathematical morphology point-of-view, there are two main paradigms: the flat zone approach using connected filters [4, 8] and the watershed-based methods [15]. Two key problems that often arise in the context of the watershed are regularization and oversegmentation. Methods that address the regularization problem include the viscous watershed transform [14] and watersnakes [10]. On the other hand, oversegmentation may be addressed by watershed with markers [13] (the most popular approach) or by a hierarchical approach [9].

Proceedings of the 8th International Symposium on Mathematical Morphology,
Rio de Janeiro, Brazil, Oct. 10–13, 2007, MCT/INPE, v. 1, p. 375–386.
http://urlib.net/dpi.inpe.br/ismm@80/2007/03.20.00.50

Inexact graph matching represents a structural alternative that has been used for image segmentation [2,3,12]. Since it is a model-based approach, it solves simultaneously the image segmentation and parts recognition problems. In the case of image segmentation, two attributed relation graphs (ARGs) are required. A *model graph* G_m should be available. There are different ways to obtain such models, depending on the application. On the other hand, an *input graph* G_i is generated from an oversegmented image, e.g., by using watershed. Image segmentation is then carried out by matching the input graph to the model graph. The possible graph matches may be shown to be equivalent to cliques of the association graph between input and model graphs. There are $|V_m|^{|V_i|}$ possible solutions for this problem, $|V_m|$ and $|V_i|$ denote the number of vertices of G_m and G_i, respectively. Because of the large number of possible matches, an objective function to assess the quality of each clique must be defined and optimized in order to provide the most suitable image segmentation.

The size of the model graph is controlled by the operator that creates the model. On the other hand, a key problem is how to control the size of the input graph, i.e., $|V_i|$, which is normally given by the number of regions of the oversegmented image. As mentioned above, watershed oversegmentation may be controlled either by markers or by a hierarchical approach. This paper adopts watershed with markers, which is the key difference from the previous methods [2,3,12]. Because the main goal of using the markers in this case is not to identify the desirable objects in the image, but only to control the number of segmented regions, the markers are generated automatically from the image, the only parameter being the maximum number of image regions in the oversegmented image. The other alternative, i.e., hierarchical watershed, will be explored and compared in future research.

We have recently introduced a semi-automated approach for model initialization to guide the graph matching segmentation procedure [3]. The input image to be segmented is decomposed into regions using watershed, as shown in Figure 1. Some regions of the oversegmented image are manually labeled by traces drawn on the main structures to be segmented. The model graph is automatically derived from the image and the watershed regions intersected by the label traces provided by the user.

This paper is organized as follows. Section 2 reviews our method. The main novelty of the present paper, detailed in Section 3, consists of the automatic generation of markers for watershed using quadtrees and the centroidal Voronoi diagrams. Experimental results are described in Section 4. This paper is concluded with some comments on our ongoing work in Section 5.

2. Model-based image segmentation

We follow the notation described in [3], where more detailed information may be found. $G = (V, E)$ denotes a directed graph where V represents the

Proceedings of the 8th International Symposium on Mathematical Morphology,
Rio de Janeiro, Brazil, Oct. 10–13, 2007, MCT/INPE, v. 1, p. 375–386.
http://urlib.net/dpi.inpe.br/ismm@80/2007/03.20.00.50

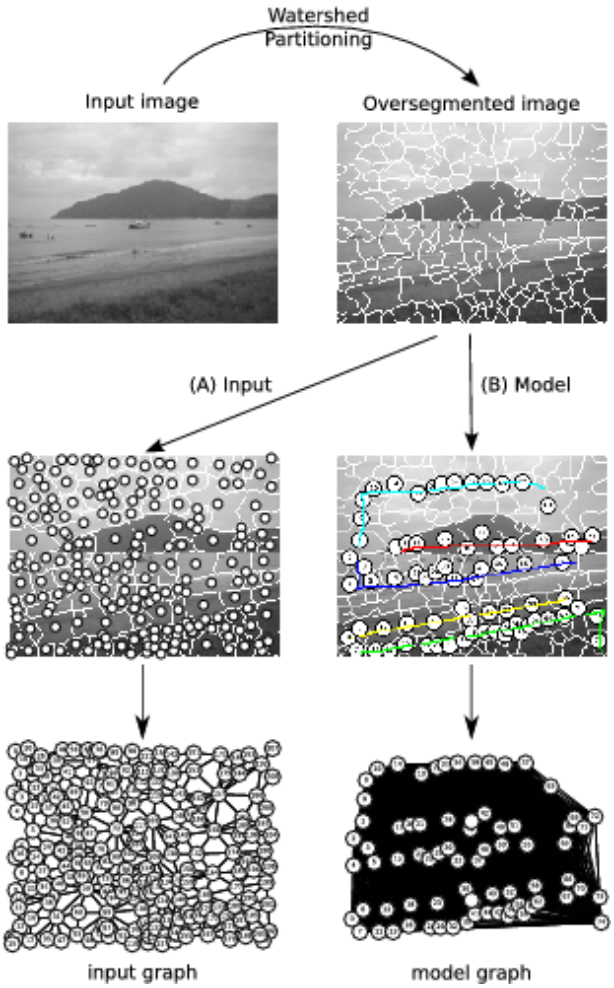

Figure 1. The input and model graphs formation process: the input image is oversegmented by a watershed procedure. Each region is represented as an input graph vertex. An adjacency graph is then generated. The user defines the model graph vertices by drawing label traces on some structurally important regions. The model graph is created as a complete graph. (Adapted from [3]).

set of vertices of G and $E \subseteq V \times V$ the set of edges. An *attributed relational graph* (ARG) is defined as $G = (V, E, \mu, \nu)$, where $\mu : V \rightarrow L_V$ assigns an attribute vector to each vertex of V. Similarly, $\nu : E \rightarrow L_E$ assigns an attribute vector to each edge of E. The vertices and the edges attributes are called *object* and *relational* attributes, respectively. Two ARGs $G_i = (V_i, E_i, \mu_i, \nu_i)$ and $G_m = (V_m, E_m, \mu_m, \nu_m)$ are adopted, i.e., the *input* (i.e.,

Proceedings of the 8th International Symposium on Mathematical Morphology,
Rio de Janeiro, Brazil, Oct. 10–13, 2007, MCT/INPE, v. 1, p. 375–386.
http://urlib.net/dpi.inpe.br/ismm@80/2007/03.20.00.50

derived from the image) and the *model* graphs, respectively. $|V_i|$ denotes the number of vertices in V_i, while $|E_i|$ denotes the number of edges in E_i. We use a subscript to denote the corresponding graph, e.g., $a_i \in V_i$ denotes a vertex of G_i, while $(a_i, b_i) \in E_i$ denotes an edge of G_i. We define in this paper $\mu(a) = (g(a))$, where $g(a)$ denotes the average gray-level of the image region associated to vertex $a \in V$, normalized between 0 and 1 with respect to the minimum and maximum possible gray-levels. Let $a, b \in V$ be two vertices of G, and p_a and p_b be the centroids of the respective corresponding image regions. The relational attribute $\nu(a, b)$ of $(a, b) \in E$ is defined as the vector $\nu(a, b) = (p_b - p_a)/(2d_{max})$, where d_{max} is the largest distance between any two points of the input image region.

An inexact match between G_i and G_m may be represented as an approximate homomorphism between G_i and G_m and is searched in the corresponding association graph [2]. The *association graph* G_A between G_i and G_m is defined as the complete graph $G_A = (V_A, E_A)$, with $V_A = V_i \times V_m$. An inexact match between G_i and G_m can be expressed as a clique (i.e., a complete subgraph) $G_S = (V_S, E_S)$ of the association graph G_A between G_i and G_m with $V_S = \{a_{im} = (a_i, a_m), a_i \in V_i, a_m \in V_m\}$ such that $\forall a_i \in V_i, \exists a_m \in V_m, a_{im} \in V_S$ and $\forall a_{im} \in V_S, \forall b_{im} \in V_S,\ a_i = b_i \Rightarrow a_m = b_m$ which guarantees that each vertex of the image graph has exactly one label (i.e., it is mapped onto a single vertex of the model graph) and $|V_S| = |V_i|$.

There is a huge number of cliques that represent possible inexact matches between G_i and G_m, namely $|V_m|^{|V_i|}$. The evaluation of the quality of a solution expressed by G_S is performed through an objective function:

$$f(G_S) = \frac{\alpha}{|V_S|} \sum_{a_{im} \in V_S} c_V(a_{im}) + \frac{(1-\alpha)}{|E_S|} \sum_{e \in E_S} c_E(e), \tag{1}$$

where $c_V(a_{im})$ is a measure of dissimilarity between the attributes of a_i and a_m. Similarly, if $e = (a_{im}, b_{im})$, $c_E(e)$ is a measure of the dissimilarity between edge (a_i, b_i) of the image and edge (a_m, b_m) of the model. The dissimilarity objective function should therefore be minimized. Let $a_{im} \in V_A$, $a_i \in V_i$ and $a_m \in V_m$. The dissimilarity measure $c_V(a_{im})$ is defined as $c_V(a_{im}) = |g_i(a_i) - g_m(a_m)|$, where $g_i(a_i)$, $g_m(a_m)$ are the object attributes of vertices $a_i \in G_i$, $a_m \in G_m$, respectively. Let $e = (a_{im}, b_{im}) \in E_A$. We compute the modulus and angular differences between $\nu(a_i, b_i)$ and $\nu(a_m, b_m)$ as $\phi_m(e) = |\|\nu(a_i, b_i)\| - \|\nu(a_m, b_m)\||$ and $\phi_a(e) = \frac{|\cos(\theta)-1|}{2}$, respectively, where θ is the angle between $\nu(a_i,\ b_i)$ and $\nu(a_m, b_m)$, i.e., $\cos(\theta)$ is calculated as:

$$\cos(\theta) = \frac{\nu(a_i, b_i) \cdot \nu(a_m, b_m)}{\|\nu(a_i, b_i)\| \|\nu(a_m, b_m)\|}.$$

In order to define the dissimilarity measure $c_E(e)$, we need an auxiliary function: $\hat{c}_E(e) = \gamma_E \phi_a(e) + (1-\gamma_E)\phi_m(e)$. The parameter γ_E $(0 \leq \gamma_E \leq 1)$ controls the weights of ϕ_m and ϕ_a. It is important to note that $\nu(a, a) = \vec{0}$.

Proceedings of the 8th International Symposium on Mathematical Morphology,
Rio de Janeiro, Brazil, Oct. 10–13, 2007, MCT/INPE, v. 1, p. 375–386.
http://urlib.net/dpi.inpe.br/ismm@80/2007/03.20.00.50

This fact means that, when two vertices in G_i are mapped onto a single vertex of G_m, we have $c_E(e) = \|\nu(a_{i_1}, a_{i_2}) - \vec{0}\| = \|\nu(a_{i_1}, a_{i_1})\|$, which is proportional to the distance between the centroids of the corresponding regions in the oversegmented image (in such cases, we define $\cos(\theta) = 1$). Therefore, $\hat{c}_E$ provides large dissimilarity values when assigning the same label (i.e., the target vertex in G_m) to distant regions and lower values when assigning the same label to near regions, which is intuitively desirable in the present application.

Let $a_{i_1}, a_{i_2} \in V_i$ and $a_{m_1}, a_{m_2} \in V_m$ be vertices of G_i and G_m, respectively. Suppose that a_{i_1} and a_{i_2} are matched to a_{m_1} and a_{m_2}, respectively. In this case, the edge (a_{i_1}, a_{i_2}) should be matched to (a_{m_1}, a_{m_2}) and the dissimilarity measure between them should be evaluated. However, depending on the adopted graph topology, it is possible that one or both edges do not actually exist and the dissimilarity measure should properly deal with such situations. The edge dissimilarity measure is therefore defined by Equation 2. It is important to highlight the case of $e' = (a_{i_1,m_1}, a_{i_2,m_2})$. The edge comparisons depend on the graph topology adopted for G_m and G_i. In the present case, an adjacency graph has been adopted for G_i whereas a complete graph is generated for G_m. Therefore, there are matching situations where there exists an edge in G_m but not on G_i (i.e., $(a_{i_1}, a_{i_2}) \notin E_i, (a_{m_1}, a_{m_2}) \in E_m$) because of the different adopted graph topologies. In this case, the (a_{i_1}, a_{i_2}) features are calculated on-the-fly using the same procedure to calculate the edge features, thus allowing the comparison to (a_{m_1}, a_{m_2}). This problem could be solved by adopting a complete graph for G_i, as in our previous works, but this would naturally increase the memory costs.

$$c_E(e) = \begin{cases} \hat{c}_E(e), & \text{if } (a_{i_1}, a_{i_2}) \in E_i, (a_{m_1}, a_{m_2}) \in E_m, \\ \hat{c}_E(e'), & \text{if } (a_{i_1}, a_{i_2}) \notin E_i, (a_{m_1}, a_{m_2}) \in E_m, \\ \infty, & \text{if } (a_{i_1}, a_{i_2}) \in E_i, (a_{m_1}, a_{m_2}) \notin E_m, \\ 0, & \text{if } (a_{i_1}, a_{i_2}) \notin E_i, (a_{m_1}, a_{m_2}) \notin E_m. \end{cases} \quad (2)$$

The objective function (Equation 1) should be optimized in order to find a suitable inexact match between G_i and G_m. There are many different optimization algorithms that may be used and the reader is referred to [2] for a comparative review that includes beam-search, genetic algorithms and Bayesian networks. The results presented in this paper have been obtained using the clique-search algorithm described in [3]. The optimization algorithm starts with an empty clique G_S and incrementally increases it by evaluating the objective function (Equation 1). The cheapest clique is chosen and a new vertex is added to it at each iteration. The algorithm stops when a clique that represents a complete solution is found. The vertices of G_A are of the form $a_{im} = (a_i, a_m), a_i \in V_i, a_m \in V_m$. For each $a_i \in V_i$ there is a set of vertices $a_{im} = (a_i, a_m), a_m \in V_m$ that represents all possible labels to which a_i may be assigned. Each of these sets is called a *supervertex*

Proceedings of the 8th International Symposium on Mathematical Morphology,
Rio de Janeiro, Brazil, Oct. 10–13, 2007, MCT/INPE, v. 1, p. 375–386.
http://urlib.net/dpi.inpe.br/ismm@80/2007/03.20.00.50

of G_A, defined as:

$$s_i = \{a_{im} = (a_i, a_m) \in V_s, a_i \in V_i, \forall a_m \in V_m\}.$$

A clique G_S that represents a valid solution is composed by one single vertex a_{im} of each supervertex s_i in G_A. For each supervertex, the association vertex a_{im} with the best node cost defines the supervertex cost. The proposed algorithm selects the cheapest supervertex s_i at each iteration. All vertices a_{im} of the selected supervertex s_i are considered in order to identify the one minimizing the objective function (Equation 1) when added to the solution clique. This idea is inspired by the Sequential Forward Search (SFS) algorithm for feature selection [7]. The final solution produced by the matching procedure may be represented as a labeled image where a label associated to the model vertices is assigned to each pixel (actually, to all pixels of each watershed connected region). A mode filter is applied to the labeled image to smooth the produced boundaries and to eliminate small noisy labels.

3. Markers detection

In order to find markers suitable for the watershed, we use quadtrees and centroidal Voronoi diagrams, an approach originally proposed for generating mosaic effects [6]. The simplest method for finding markers is by sampling the image adaptively using a quadtree. The stop criterion in the quadtree is that the intensity of all pixels in a cell is close to the average intensity in the cell. To avoid getting to single-pixel cells, we also stop the subdivision when cells get too small. The markers are the centers of the leaf cells in the quadtree. This quadtree sampling method generates two kinds of points: points that are clustered around the image edges, and points in the middle of homogeneous regions (having a low level of detail). Both kinds of points are needed for a fair sampling of the image. The important aspect is that the center of different cells tends to be on different image basins, thus defining potentially good markers for the watershed. Because only a maximum number of quadtree cells are available, oversegmentation may be controlled by the allowed total number of cells.

A more sophisticated method for finding markers is to use a centroidal Voronoi diagram whose sites are adjusted to the image features. The *Voronoi diagram* of a set of points, called *sites*, is a decomposition of the space into cells, one cell for each site, such that the cell corresponding to a site p is the set of all points in space which are closer to p than to any other site [1, 11]. A *centroidal Voronoi diagram* is a Voronoi diagram in which each site is the centroid of its cell [5].

Centroidal Voronoi diagrams are very rare. However, any Voronoi diagram can be transformed into a centroidal Voronoi diagram by a simple relaxation procedure known as *Lloyd's algorithm*: replace each Voronoi site

Proceedings of the 8th International Symposium on Mathematical Morphology,
Rio de Janeiro, Brazil, Oct. 10–13, 2007, MCT/INPE, v. 1, p. 375–386.
http://urlib.net/dpi.inpe.br/ismm@80/2007/03.20.00.50

by the centroid of its Voronoi cell, recompute the Voronoi diagram for the new sites, and repeat until convergence.

A major ingredient in the definition of a centroidal Voronoi diagram is an underlying *density function* with respect to which the centroids of the Voronoi cells are found. More precisely, the centroid of a region V with respect to a density function μ is the point z given by

$$z = \frac{\int_V x\mu(x)\,dx}{\int_V \mu(x)\,dx}\ .$$

Naturally, for images we use sums over pixels as the discrete analogues of these integrals. The density function μ does not enter in the computation of the Voronoi diagram, which is still computed using the Euclidean metric.

Centroidal Voronoi diagrams adapt themselves to the mass distribution implied by the density function, having larger cells where the density is low and smaller cells where the density is high. As in the mosaic approach described in [6], we use the Euclidean norm of the gradient of the image as density function. The gradient is computed using central differences. We start from the markers found in the quadtree sampling step and compute a centroidal Voronoi diagram using a few iterations of Lloyd's algorithm. The new markers are the centroids of the final Voronoi diagram.

4. Experimental results

The proposed approach has been applied to different images of the Berkeley Image Segmentation Database[1]. The traced strokes manually defined for three test images are shown in Figure 2. Each color represents a different label to be recognized by the matching procedure. In order to compare the different approaches, the same set of traced strokes for each test images have been used to generate the models for the three approaches assessed in the present paper: (i) watershed without markers; (ii) watershed with markers generated by quadtrees; (iii) watershed with markers generated by the centroidal Voronoi diagrams. It is worth noting that the image regions may present similar gray-level and belong to different model classes defined by the user labels. Also, there are some image regions with substantial gray-level variation because of belonging to non-homogeneous textured regions, which are traditionally very difficult to segment. The structural information leads to a robust segmentation performance even in such cases.

The segmentation results are shown in Figures 3, 4 and 5. These figures present, for each image, the results using the watershed without markers

[1] http://www.eecs.berkeley.edu/Research/Projects/CS/vision/grouping/segbench/

Proceedings of the 8th International Symposium on Mathematical Morphology,
Rio de Janeiro, Brazil, Oct. 10–13, 2007, MCT/INPE, v. 1, p. 375–386.
http://urlib.net/dpi.inpe.br/ismm@80/2007/03.20.00.50

(top row), the watershed with markers provided by the quadtree (middle row) and the watershed with markers provided by the centroidal Voronoi diagram (bottom row). The first column shows the oversegmentation in the case of the watershed without markers and the corresponding quadtree and Voronoi partitions with the cells seeds, which are used as markers for the watershed. The middle column shows the corresponding watershed partitions (red) and the final segmentation result in green. Finally, the last row shows the final segmentation results with the region labels provided by the inexact matching procedure.

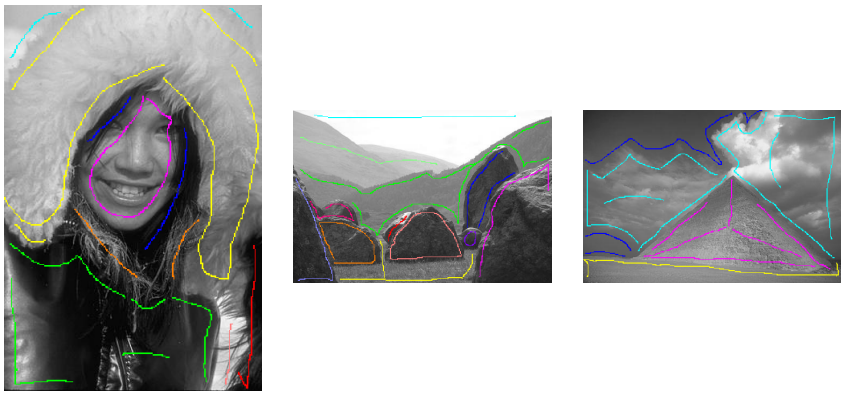

Figure 2. Strokes traced by the user on each test image are used to generate the graph models. Each color is associated to a different label, i.e., a different class to be recognized by the matching procedure.

As it can be seen, though a smaller number of larger regions are used by the matching procedure in the case of the watershed with markers (both quadtree and Voronoi), the final segmentation results are comparable. In some cases, the segmentation actually improves, once the watershed without markers lead to too many regions, some very small, which makes more difficult the correct matching and may lead to misclassifications. Therefore, in general, the final segmentation is comparable for any of the three assessed approaches. Nevertheless, the main difference lies in the memory and running time costs, as shown in Table 1. It can be seen from that table that the input and model graphs are substantially smaller than with the previous approach without markers. The running time differences decrease from nearly an hour to some seconds. This important decrease in running time is explained because the total number of possible solutions is exponential in the sizes of the graphs ($|V_m|^{V_i|}$) and, once both $|V_i|$ and $|V_m|$ decrease, the search space is considerably shrinked. It is also important to note that, although the centroidal Voronoi method performs a kind of fine tuning in the position of the markers, there is apparently no strong differences between

Proceedings of the 8th International Symposium on Mathematical Morphology,
Rio de Janeiro, Brazil, Oct. 10–13, 2007, MCT/INPE, v. 1, p. 375–386.
http://urlib.net/dpi.inpe.br/ismm@80/2007/03.20.00.50

the final result produced by it and the results produced by the quadtrees.

Figure 3. (a) Graph matching without markers; (b) Graph matching with quadtree markers; (c) Graph matching with Voronoi markers.

Once the quadtrees approach is computationally cheaper, it may be more

Proceedings of the 8th International Symposium on Mathematical Morphology,
Rio de Janeiro, Brazil, Oct. 10–13, 2007, MCT/INPE, v. 1, p. 375–386.
http://urlib.net/dpi.inpe.br/ismm@80/2007/03.20.00.50

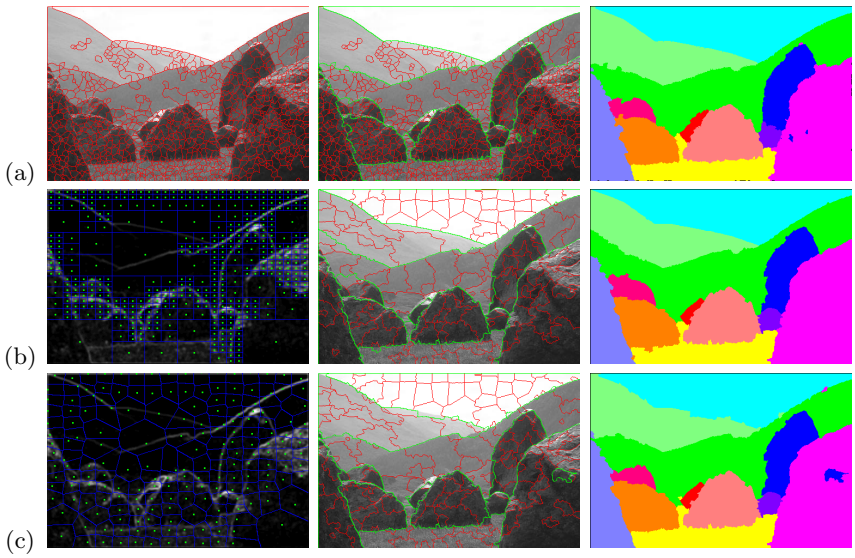

Figure 4. (a) Graph matching without markers; (b) Graph matching with quadtree markers; (c) Graph matching with Voronoi markers.

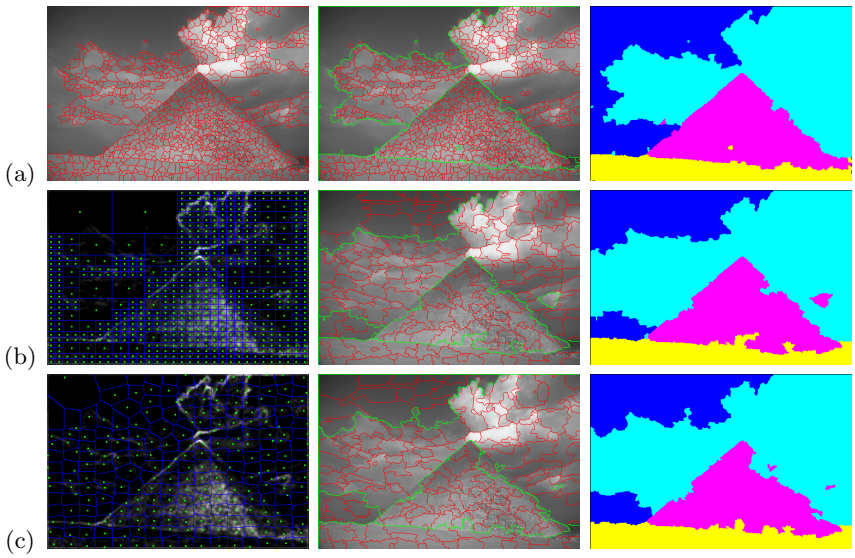

Figure 5. (a) Graph matching without markers; (b) Graph matching with quadtree markers; (c) Graph matching with Voronoi markers.

Proceedings of the 8th International Symposium on Mathematical Morphology,
Rio de Janeiro, Brazil, Oct. 10–13, 2007, MCT/INPE, v. 1, p. 375–386.
http://urlib.net/dpi.inpe.br/ismm@80/2007/03.20.00.50

advantageous. However, because different density functions may be adopted for the centroidal Voronoi diagram, we feel that a good research topic is to look for more suitable functions that could improve its performance. For instance, an obvious drawback of the quadtree/Voronoi approaches is that, because the oversegmented image is partitioned in a smaller number of regions, some true edges are lost. It would be nice to have a method to avoid such a problem.

Table 1. The table summarizes the size of the input and model graphs ($|V_i|$ and $|V_m|$, respectively) and the running time to segment the image.

Watershed	Figure	$\|V_i\|$	$\|V_m\|$	Running time
Without markers	3(a)	1482	352	1h34m20s
Quadtree	3(b)	226	152	25s
Voronoi	3(c)	226	133	24s
Without markers	4(a)	1102	373	59m30s
Quadtree	4(b)	187	121	14s
Voronoi	4(c)	187	117	13s
Without markers	5(a)	1003	317	35m10s
Quadtree	5(b)	232	152	30s
Voronoi	5(c)	232	134	27s

5. Conclusion

We have recently introduced an interactive image segmentation approach based on inexact graph matching. The present paper improves our previous works by exploring the watershed with markers automatically generated by using quadtrees and centroidal Voronoi diagrams. The results are equivalent in quality, but the improvement expressively decreased running time and memory requirements. Our ongoing work includes research to find more suitable density functions for the centroidal Voronoi diagram and the inclusion of additional object attributes such as color and texture. Also, we intend to explore the method for the recognition of object parts by using different model and input images.

References

[1] F. Aurenhammer, *Voronoi diagrams—a survey of a fundamental geometric data structure*, ACM Computing Surveys **23** (1991), no. 3, 345–405.

[2] R. M. Cesar-Jr., E. Bengoetxea, I. Bloch, and P. Larrañaga, *Inexact graph matching for model-based recognition: Evaluation and comparison of optimization algorithms*, Pattern Recognition **38** (2005), no. 11, 2099–2113.

Proceedings of the 8th International Symposium on Mathematical Morphology,
Rio de Janeiro, Brazil, Oct. 10 –13, 2007, MCT/INPE, v. 1, p. 375–386.
http://urlib.net/dpi.inpe.br/ismm@80/2007/03.20.00.50

[3] L. A. Consularo, R. M. Cesar-Jr., and I. Bloch, *Structural image segmentation with interactive model generation*, Proc. IEEE International Conference on Image Processing (ICIP-07), 2007.

[4] J. Crespo, R. W. Schafer, J. Serra, C. Gratin, and F. Meyer, *The flat zone approach: a general low-level region merging segmentation method*, Signal Process. **62** (1997), no. 1, 37–60.

[5] Q. Du, V. Faber, and M. Gunzburger, *Centroidal Voronoi tessellations: Applications and algorithms*, SIAM Review **41** (1999), no. 4, 637–676.

[6] G. M. Faustino and L. H. de Figueiredo, *Simple adaptive mosaic effects*, Proceedings of SIBGRAPI'05, 2005, pp. 315–322.

[7] A. Jain and D. Zongker, *Feature selection - evaluation, application, and small sample performance*, IEEE Transactions on Pattern Analysis and Machine Intelligence **19** (1997), no. 2, 153–158.

[8] A. Meijster and M. H. F. Wilkinson, *A comparison of algorithms for connected set openings and closings*, IEEE Trans. Pattern Anal. Mach. Intell. **24** (2002), no. 4, 484–494.

[9] L. Najman and M. Schmitt, *Geodesic saliency of watershed contours and hierarchical segmentation*, IEEE Trans. Pattern Anal. Mach. Intell. **18** (1996), no. 12, 1163–1173.

[10] H. T. Nguyen, M. Worring, and R. van den Boomgaard, *Watersnakes: Energy-driven watershed segmentation.*, IEEE Trans. Pattern Anal. Mach. Intell. **25** (2003), no. 3, 330–342.

[11] A. Okabe, B. Boots, and K. Sugihara, *Spatial tessellations: Concepts and applications of Voronoi diagrams*, John Wiley & Sons, Inc., 1992.

[12] A. Perchant and I. Bloch, *A New Definition for Fuzzy Attributed Graph Homomorphism with Application to Structural Shape Recognition in Brain Imaging*, Imtc'99, 16th ieee instrumentation and measurement technology conference, 1999, pp. 1801–1806.

[13] P. Soille, *Morphological image analysis: Principles and applications*, Springer Verlag, 1999.

[14] C. Vachier and F. Meyer, *The viscous watershed transform*, J. Math. Imaging Vis. **22** (2005), no. 2-3, 251–267.

[15] L. Vincent and P. Soille, *Watersheds in digital spaces: An efficient algorithm based on immersion simulations*, IEEE Trans. Pattern Anal. Mach. Intell. **13** (1991), no. 6, 583–598.