AUTOMATIC ADAPTATION OF SEGMENTATION PARAMETERS APPLIED TO INHOMOGENEOUS OBJECTS DETECTION

C. M. B. Fredrich[*], R. Q. Feitosa[*]

[*]Dept. of Electrical Engineering, Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, Brazil
(buarque, raul)@ele.puc-rio.br

KEY WORDS: Parameter Adaptation, Parameter Tuning, Object Detection, Segmentation, Genetic Algorithms, Hough Transform

ABSTRACT:

Virtually all segmentation methods require parameter tuning, quite a difficult task, mostly performed manually through a troublesome trial-and-error process. To overcome this difficulty, an earlier work describes an automatic parameter adjustment method using Genetic Algorithms (GAs), given an initial set of reference object samples. The method performs well only for homogeneous objects. However, in most real applications, the meaningful image objects are actually non-homogeneous, or rather, an ensemble of usually few homogeneous segments. This work addresses this issue and proposes a supervised GA-based method to automatically adjust the values of segmentation parameters in applications where meaningful objects are inhomogeneous, though formed by an assembly of homogeneous parts. Moreover the work introduces a post-segmentation procedure that merges adjacent segments into single units, which match the geometric form of the interest image objects. Specifically, a metric for detection of polygonal arrangements of segments is proposed herein. Experimental analyses evidence the higher performance of the new method for adjusting segmentation parameters in comparison with the earlier approach. The experiments also attest the ability of the proposed post-segmentation metric to detect polygonal shapes.

1. INTRODUCTION

Segmentation is the first and most important step in object based image analysis. This is not a simple task due to a number of reasons. One of them refers to the determination of parameter values for the segmentation algorithm. These should yield segments that are consistent with the meaningful objects in that particular application. However, the relation between the parameter values and the segmentation outcome is far from being obvious. Hence, tuning accordingly often incurs a time-consuming frustrating series of trials and errors.

Feitosa et. al. (2006) propose an automatic method, in which a Genetic Algorithm (GA) searches the parameter space for the (near) optimum values. Optimality is then established by a fitness function that measures the level of agreement between the segmentation outcome and a set of analyst provided sample objects. The reported experiments attest that, indeed, the method performs well for homogeneous objects.

In real applications, however, meaningful objects are often non-homogeneous, though formed by an assembly of homogeneous parts. This paper addresses the detection of such objects. A new fitness function was also designed to favour segmentation results that match the sample objects with a minimum of segments.

Once the (nearly) optimum set of parameter values was found and the image was segmented, a so-called post-segmentation procedure analyses the groups of adjacent segments that correspond to each of the interest objects. This issue has been addressed in Korting (2006), where a self-organizing map (SOM) infers from examples if a group of adjacent segments fits the type of image objects the analyst is after.

A heuristic search for groups of adjacent segments that match an explicit object description is the alternative proposed herein. This method can potentially be applied to a variety of object types, as long as an appropriate description in terms of measurable attributes can be provided. However, the forthcoming analysis concentrates on the problem of detecting residences. The paper presents two novel geometric metrics to describe rooftops appearing in high-resolution images.

The contribution of this work is twofold. First, a novel GA-based algorithm is proposed to adjust the values of segmentation parameters in applications where the interest objects are inhomogeneous.

Second, a new post-segmentation algorithm is introduced for determining the arrangement of segments produced by a prior segmentation step that best fits the interest objects.

The overall accuracy of the post-segmentation – as well as its processing time – is intimately dependent on the actual segmentation quality, thus justifying the joint approach to both issues.

This paper is organized as follows: the next section introduces the technique applied to the automatic adaptation of segmentation parameters; section 3 addresses the segmentation quality assessment measure; sections 4 to 6 present, respectively, the integral post-segmentation procedure, the methodology implemented on the operator design and the experimental analyses together with their respective results. All of which followed by some concluding remarks.

2. EVOLUTIONARY TUNING OF SEGMENTATION PARAMETERS

The key to a high segmentation quality lies in determining suitable values for each one of the parameters of a given technique. However, stating the relation between those and the algorithm's outcome is hardly ever possible.

The user can undertake this tuning step manually. Nevertheless, the massive number of possible parameters configurations enforces the adoption of an automatic search algorithm for this matter.

In this study, the way of dealing with the parameters tuning issue for the segmentation stage is as formerly introduced in Feitosa et. al. (2006), a publication addressed to the description of an early version of the method adopted and briefly depicted in the remaining of this section. The tuning quality evaluation itself, nevertheless, deserved an individual section, since a novel measurement was specially devised for this particular problem of inhomogeneous objects detection.

The standard classical definition of Genetic Algorithms (GAs) states that they are stochastic algorithms of search and optimization based on genetic inheritance and Darwinian strife to survival. They should be perceived, nonetheless, as a heuristic for finding the optimal solution to a problem, conducted by parallel search, rather than by exhaustion or a troublesome and time consuming trial-and-error process.

In this specific application, the optimal solution is the set of parameters values, which *minimises* the function that represents how well the segmentation outcome fits a group of reference segments provided by the user. Furthermore, the searching heuristic is compounded of several (genetic) operators. The candidates for the (near) best solution – or chromosomes – must carry the information corresponding to the segmentation parameters in some designer-defined representation.

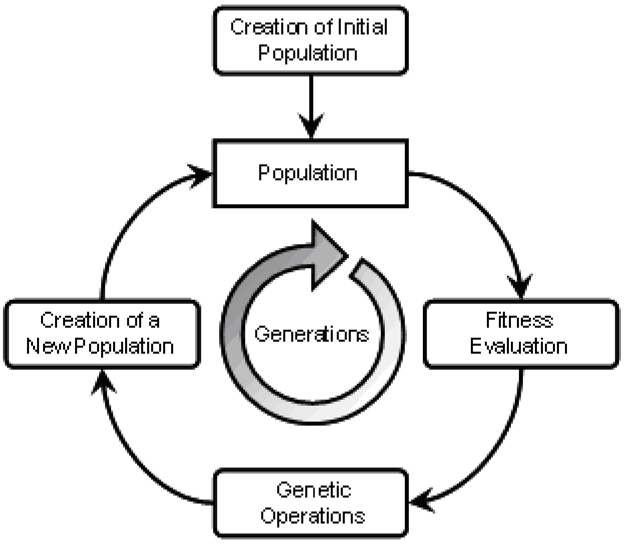


Figure 1. Basic GA architecture

Figure 1 displays the basic architecture of a generic GA. Additionally, Figure 2 presents a schematic of the whole segmentation evaluation process underwent by every single one of the chromosomes in the population created – or renewed – at each generation. The schematic is actually the detailed procedure concerning the "Fitness Evaluation" block (Figure 1), when applied to the particular segmentation parameters calibration problem.

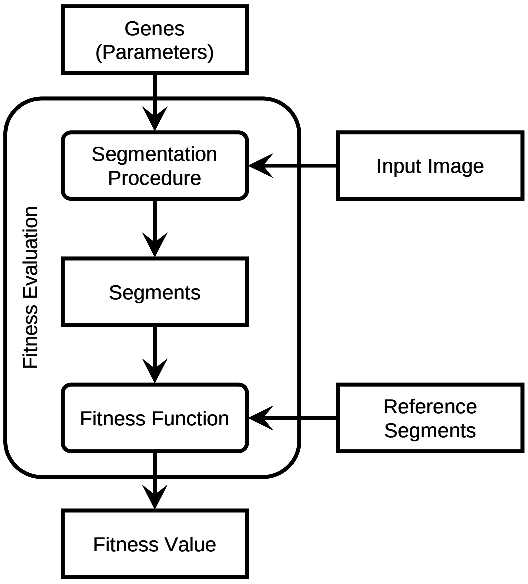


Figure 2. Segmentation evaluation block diagram

3. SEGMENTATION EVALUATION

Since the candidates for optimal solution are matched against each other by the fitness function – which is actually the optimization problem statement itself – it must draw special designer's attention.

Figure 3 depicts all entities featured in the function devised for the parameters tuning and given by equation (1).

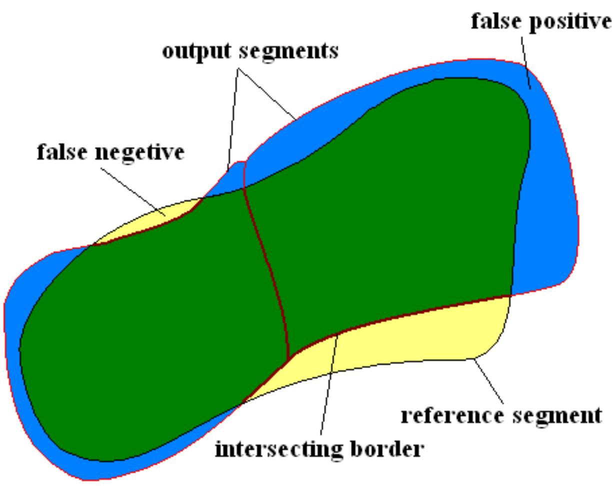


Figure 3. Graphical representation of all entities featured in the fitness function

Let us first assume that there are N reference segments delineated by the analyst. We denote by RSA_i (for $i=1,\ldots N$) the area of the i^{th} reference segment, expressed in pixels. Let also Ω_i be the set of segments produced by the segmentation algorithm and that have at least half of their pixels in the RSA_i domain. With some abuse of notation, the symbol Ω_i stands, as well, for the set of pixels belonging to any segment in Ω_i. Let us further consider

- fp_i as the number of pixels in Ω_i that do not belong to the i^{th} reference segment; the so-called false-positives;
- fn_i as the number of pixels in the i^{th} reference segment that do not belong to Ω_i; the so-called false-negatives;
- b_i as the number of border pixels in Ω_i that intersect the i^{th} reference segment area , and
- NS as the number of non-empty Ω_i.

The fitness function then used by the GA to measure the segmentation quality is defined as

$$F = \begin{cases} \dfrac{1}{NS} \displaystyle\sum_{\Omega_i \neq \varnothing} \dfrac{fp_i + fn_i + b_i}{RSA_i}, & \text{if } NS > 0 \\ 1, & \text{otherwise} \end{cases} \tag{1}$$

The behaviour of the function given above is worth being considered with care. First, both fp_i and fn_i terms favour solutions with a tight overlap with the reference segments, which most likely leads to solutions consisting of numerous small (in the limit of single-pixel) segments. The b_i term counterbalances this effect by granting a lower score to solutions with few, larger, segments.

Second, it can be easily demonstrated that F lies within [0 1], whereby $F=0$ corresponds to a perfect match between a solution and the set of reference segments. A noteworthy fact concerning the NS term is that it equals zero, when all segments produced by the segmentation algorithm have less than 50% overlap with the reference segments. In such cases, F is set to 1.

4. POST-SEGMENTATION PROCEDURE

This section presents an overview of the post-segmentation procedure.

4.1 Terminology

In order to ease the understanding of this and the subsequent sections, some terms are defined bellow.

Segment Sub-Cluster: Group of segments, sharing a specific property, whose union forms a single connected component.

Segment Cluster: A segment sub-cluster containing no further adjacent segments with the property that establishes the membership to the sub-clusters.

Winner Sub-Cluster: The most suitable segment sub-cluster within a cluster, according to pre-defined criteria.

4.2 General Description

The block diagram in Figure 4 illustrates the whole object detection method. The bold frame surrounds the stages concerning post-segmentation procedure and the rectangular boxes denote the processing blocks.

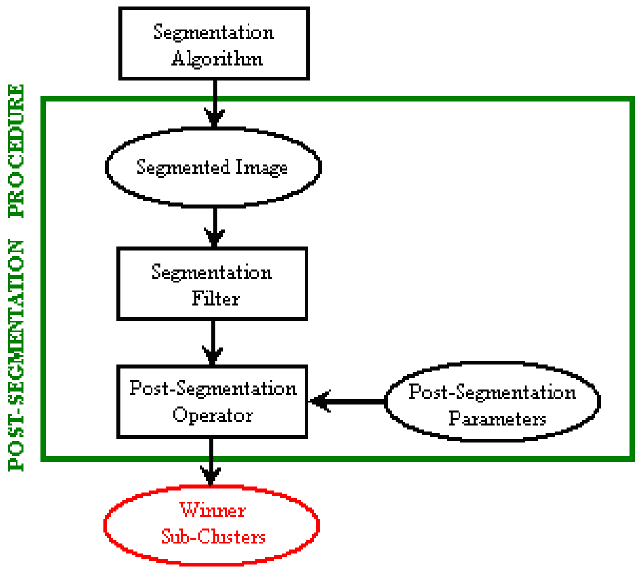


Figure 4. Inhomogeneous objects detection system

The accuracy and the computational performance of the post-segmentation procedure depend essentially on the quality of the segmented image produced in the foregoing step. In this context, quality is an assessment of how well each interest object matches some arrangement of few adjacent segments.

The first post-segmentation step is carried out by a segmentation filter, which eliminates from the succeeding analysis all segments that do not meet a set of necessary conditions to be part of an interest object. Such conditions are formulated primarily by the characteristics of the interest objects as well as their context in the image. The segmentation filter may be viewed as a two-class classifier whose design and optimal parameter values are application dependent. In effect, it plays an important role in the whole procedure by reducing the problem complexity and consequently the processing time associated to the next step.

The filter outputs the segment clusters to be processed by the post-segmentation operator, which yields the winner sub-clusters, i.e. the final result of the whole post-segmentation procedure.

5. POST-SEGMENTATION OPERATOR

Apart from the new approach for the adaptation of segmentation parameters introduced in sections 2 and 3, a major contribution of this paper relates to the post-segmentation operator, which receives as inputs the filtered label images.

A further input to the post-segmentation operator is the set of its own parameters values. The optimum set depends on both the characteristics of the image to be post-processed and the interest objects themselves. The post-segmentation parameters calibration is analogous to the problem addressed in section 2. This task can be performed either manually or by using GA or yet some other optimization method. Despite of its importance, a detailed discussion on this issue cannot be properly conducted in this paper due to space restrictions. Thus, it is assumed henceforth that suitable parameters values have been provided somehow.

The post-segmentation operator processes all segment clusters separately. For each cluster, the shapes of their sub-clusters are individually evaluated on how well they fit the geometric form of the interest objects.

Thus, the evaluation metric must be devised as to compare a given sub-cluster to a mold that is representative of the objects. The operator then applies it to all segment sub-clusters and elects a winner.

5.1 Fitness Metric

This subsection describes the metric devised for detecting polygonal objects whose edges are mainly parallel or perpendicular to one another.

The Hough Transform for line fitting is applied to find the edges contouring each segment sub-cluster. The sub-cluster fitness metric M relates to the number of existing parallel and perpendicular edges detected by the Hough Transform, weighted by their length. It is given by equation (2).

$$M = -\frac{\sum_i \left[\ell_i \left(\min_{j \neq i}\left(\left| \theta_j - \theta_i \right| \right) + \left| 90° - \max_{j \neq i} \left| \theta_j - \theta_i \right| \right| \right) \right]}{\sum_i \ell_i} \qquad (2)$$

where ℓ_i is the i^{th} edge length in pixels, and θ_i (θ_j) is the i^{th} (j^{th}) edge angle in degrees. Only significant edges are considered in the above equation. To qualify as significant, an edge must fulfill a number of parameterized requirements, which are given by part of the post-segmentation parameter set.

The following considerations help to understand the reasoning behind this equation. The $|\theta_j - \theta_i|$ term is the angle formed by the i^{th} and j^{th} edges. Thus, a low value of

$$\min_{j \neq i}\left(\left| \theta_j - \theta_i \right| \right)$$

implies there is another edge in the sub-cluster that is nearly parallel to the i^{th} edge. Analogously a low value of

$$90^\circ - \max_{j \neq i}\left(\left| \theta_j - \theta_i \right| \right)$$

means that there is another edge in the sub-cluster nearly orthogonal to the i^{th} edge. In consequence, M is expected to be close to zero if the sub-cluster contour consists mostly of parallel and perpendicular straight lines. After the clusters processing is completed, all winners have their evaluations normalised between 0 and 1.

This metric can be easily adapted for detecting other regular polygonal shapes. For hexagonal objects, for instance, the constant 90° is replaced by 120° in (2). In such case, parallel edges play the same role as before.

The lower threshold (M_{min}) is another parameter of the post-segmentation operator. If a winner sub-cluster is given an evaluation lower then M_{min}, it is cast aside.

6. EXPERIMENTAL ANALYSIS

A software prototype implementing the proposed method was built for validation and applied to the problem of detecting residential rooftops on high-resolution optical satellite images cropped from Google Earth®.

The watershed-based segmentation algorithm described in Mota et al. (2007), was used in all experiments. The GA fitness function for determining the (near) optimal segmentation parameters values is given by equation (1).

In Experiments 2 and 3, the segmentation filter relied exclusively on radiometric attributes. A detailed description of such filter can not be accommodated herein. Basically, it was designed to ideally suppress all segments that weren't brownish to reddish.

The accuracy of the post-segmentation outcome is given by a metric computed in the following way. A logical matrix having the same number of rows and columns as the input image represents the post-segmentation output. Its elements are equal to "1" in positions corresponding to pixels of the winner sub-clusters and "0" otherwise. A target matrix with the same format was created manually to represent the ground truth, i.e., the ideal post-segmentation result, regarding the prior segmentation outcome. The performance quality is expressed by the similarity, given by the proportion of positions where the output and the target matrices agree and that do not correspond to filtered areas (i.e., positions equal to "0" in both matrices).

The processing time measures reported in Experiments 2 and 3 refer only to the post-segmentation operator – implemented in the MATLAB® environment – and to tests conducted in a Pentium Duo Core Processor (2.80 GHz – 2 Gb RAM Memory). The GA was implemented in C# and Experiment 1 was conducted in the same computer.

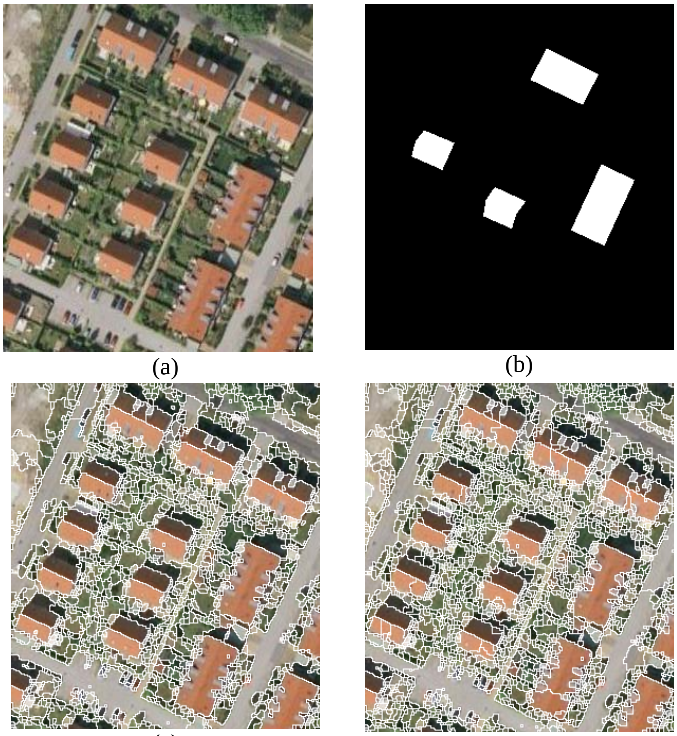
6.1 Experiment 1

This subsection describes the experiments conducted to validate the proposed extension to the GA-based method for automatic tuning of segmentation parameter values.

The input data for this experiment is the 402×382 RGB image of Figure 5a and the reference sample segments of Figure 5b.

Figures 5c and 5d show, respectively, the segmentation results, if applying the parameter values tuned by equation (1) and the ones resulting from the fitness function proposed in Feitosa et al. (2006). The same C# implemented GA was adopted in both tests. Each segmentation evaluation was completed in approximately 280 ms.

A comparison between Figures 5c and 5d shows that the new proposed fitness function yields fewer segments for each interest object appearing in the image. This typical behaviour may be attributed to the b_i term, added to penalise segment borders that cross the interior of reference objects. This was the fitness function adopted in both subsequent experiments.



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

Figure 5. Experiment on segmentation parameters tuning: a) input image; b) reference objects; c) proposed method outcome; d) original method outcome

6.2 Experiment 2

This experiment investigates the performance of the post-segmentation operator for detecting rectangular rooftops. Figure 6a shows the input image (386×240 pixels).

Figure 6b presents the filtering step result. The post-segmentation operator used the M metric exactly as stated by equation (2). Figure 6c shows the winner sub-clusters. The inhomogeneous objects were overall successfully detected, exception made to the fourth rooftop from top to bottom in the middle column. Notice that the left out segments form a rectangular object themselves. Although the winner election method prioritises larger sub-clusters, the one without those segments was still granted a higher grade.

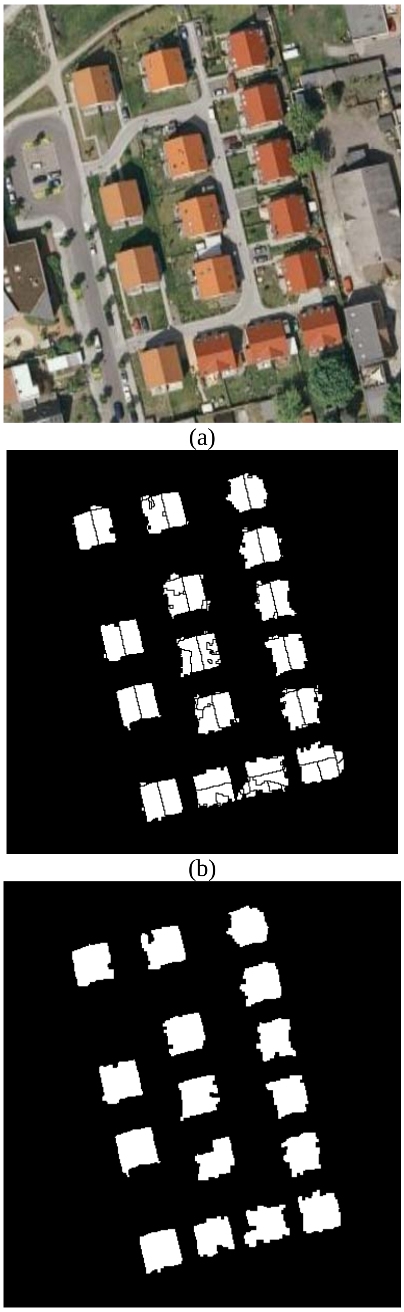
Table 1 presents the attained performance in terms of similarity to the target result and processing time.

Performance	
similarity	95.9 %
processing time	28min 46sec

Table 1. Post-Segmentation performance in Experiment 2

6.2 Experiment 3

In order to test the performance of the method for objects of different forms, an image containing hexagonal rooftops was taken under consideration. The M metric of equation (2) underwent the alteration suggested in 5.1 for hexagonal objects and the procedure was applied as in the previous experiment, without any further changes.



(a) (b) (c)

Figure 6. Post-segmentation results; a) input image; b) segmented image; c) detected objects

The 292×325 pixels input image is shown in figure 7a. Clusters obtained after filtering are depicted in Figure 7b.

Figure 7c presents the winner sub-clusters. Notice that the little protrusions on top of both hexagons are part of the larger segments and, therefore, couldn't possibly be eliminated by the operator; despite the fact that they are not parts of the real objects. Nevertheless, the operator was able to find the best sub-cluster for both rooftops appearing in the input image. It is also interesting to note that the amorphous segments seen in Figure 7b are not present in the final outcome shown in Figure 7c. Since they are clusters themselves, each of them should be the only possible winner sub-cluster within their cluster. However, they were all discarded because the (normalised) evaluations returned by the M metric were below a user-selected threshold (M_{min}).

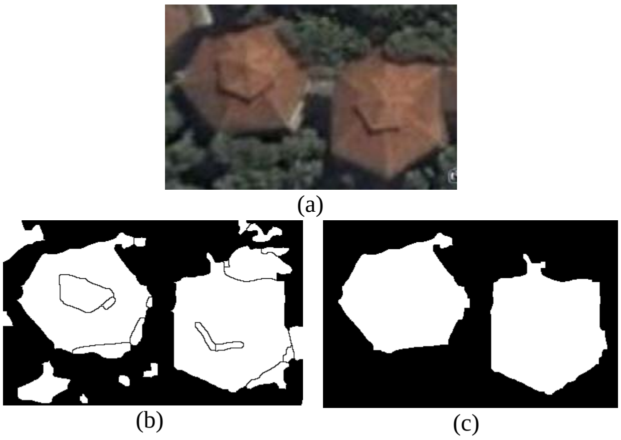


Figure 7. Experiment on post-segmentation: a) input image; b) segment clusters; c) post-segmentation result

Table 2 shows the performance achieved in terms of similarity and processing time.

Performance	
similarity	99,8 %
processing time	1min 7sec

Table 2. Post-Segmentation performance in Experiment 3

Tables 1 and 2 show that results attained in Experiments 2 and 3 were quite similar. The experiment series conducted so far is, however, still insufficient for a definitive judgment of the proposed methods. Nevertheless, these results encourage this investigation to go on. Taking into account the image sizes in Experiments 2 and 3 and the associated processing times, the demand for a more efficient search algorithm to find winner sub-clusters becomes evident.

7. OVERALL EVALUATION AND CONCLUDING REMARKS

This paper proposes an extension to a previous work for the automatic adaptation of segmentation parameters. Specifically, an alternative fitness function – which copes with inhomogeneous objects composed of few homogeneous parts – was embedded in the new Genetic Algorithm. The proposal has been tested for a number of images and types of objects and the result of a sample experiment was reported. Generally, the proposed extension outperforms the original method, when applied to non-homogeneous objects.

A second contribution is a post-segmentation procedure that automatically finds, among segments provided by a foregoing step, the arrangements of homogeneous segments that best fit the meaningful image objects. The experimental analysis conducted so far and partially reported in this document, showed encouraging results.

Nevertheless, design alternatives weren't yet exhausted and must be approached in the ongoing research. Despite the fact that the software environment used for the experimental analysis is computationally inefficient, the processing time observed for some experiments was still significant. The exhaustive search as considered in this paper is acceptable, regarding such terms, only in cases where the number of segments to be arranged into a meaningful object is low. In this respect, the most important issue is the development of a non-exhaustive search method for segment sub-cluster computation.

The segmentation filtering technique is also worthy of further investigation. Naturally, the number of segments to be processed by the post-segmentation operator might drop significantly with the increase of its efficiency.

Another issue concerns the automatic parameter tuning for the post-segmentation operator. A solution similar to the one presented in this text for the segmentation algorithm is an approach worth being considered.

Finally, variants of the proposed Hough Transform-based metric for detecting meaningful objects of different forms should be taken under consideration, as well.

REFERENCES

Baatz, M. and Schäpe, A., 2000. Multiresolution Segmentation – an optimization approach for high quality multi-scale image segmentation. In: *Strobl J, Blaschke T, Griesebner G (eds) Angewandte Geographische Informationsverarbeitung*, Heidelberg, Vol. XII, pp. 12-23.

Feitosa, R. Q., Costa, G. A. and Cazes, T., 2007. *Sensoriamento Remoto e SIG Avançados.* Oficina de Textos, São Paulo, pp. 129-144.

Feitosa, R. Q., Costa, G. A. O. P., Cazes, T. B. and Feijó, B., 2006. A genetic approach for the automatic adaptation of segmentation parameters. In: *1st International Conference on Object Based Image Analysis*, Salzburg. Also available in http://www.commission4.isprs.org/obia06/Papers/11_Adaption%20and%20further%20development% (accessed 16 May 2008)

Forsyth, D. A. and Ponce J., 2003. *Computer Vision*, Prentice Hall, New Jersey, pp. 329-333.

Gonzalez, R. C. and Woods, R. E., 2008. *Digital Image Processing*, Prentice Hall, New Jersey, pp. 774-776.

Korting, T. S., Fonseca, L. M. G., Dutra, L. V. and Silva, F. C., 2008 Image Re-Segmentation - A New Approach Applied to Urban Imagery. In: *3rd International Conference on Computer Vision Theory and Applications (VISAPP)*, Madeira, Portugal.

Michalewicz, Z., 1999. *Genetic Algorithms + Data Structures = Evolution Programs*, Springer, Heidelberg.

Mota, G. L. A., Feitosa, R. Q., Coutinho, H. L. C., Liedtke, C. E., Müller, S., Pakzad, K. and Meirelles, M. S. P., 2007. Multitemporal fuzzy classification model based on class transition possibilities. *ISPRS Journal of Photogrammetry and Remote Sensing* 62, pp. 186-200.

"Sistema de Processamento de Informações Georreferenciadas", GIS developed. http://www.dpi.inpe.br (accessed 16 May 2008)