



PERGAMON

Available at
www.ElsevierComputerScience.com

POWERED BY SCIENCE @ DIRECT®

Pattern Recognition 37 (2004) 1785–1795

**PATTERN
RECOGNITION**

THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY

www.elsevier.com/locate/patcog

Image segmentation functional model

T. Zouagui*, H. Benoit-Cattin, C. Odet

CREATIS, UMR CNRS #5515, affiliated to INSERM INSA de Lyon, Bat. B. Pascal, Villeurbanne Cedex 69621, France

Received 15 November 2002; received in revised form 25 July 2003; accepted 10 December 2003

Abstract

We propose a new approach of the image segmentation methods. This approach is based on a functional model composed of five elementary blocks called in an iterative process. Different segmentation methods can be decomposed with such a scheme and lead to elementary building blocks with unified functionality and interfaces. We present the decompositions of three segmentation methods and the implementation results, which illustrate the potential of the proposed model. This generic model is a common framework, which makes segmentation techniques more readable and offers new perspectives for the development, the comparison and the implementation of segmentation methods.

© 2004 Pattern Recognition Society. Published by Elsevier Ltd. All rights reserved.

Keywords: Image segmentation; Functional model; Framework; Unified approach; Segmentation methodology

1. Introduction

Image segmentation is the process of assigning pixels to regions having common properties. It is one of the fundamental process in computer vision and pattern recognition because further processing steps have to rely on the segmentation results. Despite the numerous segmentation techniques, image segmentation is still a subject of on-going investigations and it cannot be conclusively stated that the segmentation problem has been solved because of the application's diversity. As a consequence, the task of choosing the best method for a specific application is still a difficult challenge. Several survey papers [1–3] cover the major image segmentation techniques available. Most of the segmentation techniques can be roughly categorized into two approaches: Boundary-based methods and region-based methods.

Basically, the first approach is based on discontinuity and tends to partition an image by detecting isolated points, lines and edges according to abrupt changes of local properties. The regions are then deduced from their boundary. The usual tools that are employed in boundary-based methods

include local filtering approaches such as Canny edge detector [4] or energy minimization like the active contour model (i.e. snake model) [5] and balloon models [6]. The algorithms from the second approach exploit the homogeneity of spatially dense information (e.g. intensity, color, texture properties, etc.) to produce the segmented image. It includes thresholding [7], clustering [8], region-growing [9], region splitting [10] and merging [11]. Both types of approaches have their advantages, drawbacks and limitations. To improve the segmentation results, a strategy consists in combining these two approaches in order to obtain a robust segmentation by exploiting the advantages of one method to reduce the drawbacks of the second one. Different frameworks have been proposed [12,13]. In [14], Zhu develops a unifying framework that combines the attractive geometrical features of deformable models and the statistical techniques of region growing. Germond et al. [15] propose to mix in a cooperative framework several types of information and knowledge provided and used by complementary individual systems like a multi-agent system, a deformable model and an edge detector. Other authors, like Geiger [16], propose an approach based on mathematical models. This is an attempt to unify different methods of image segmentation under a common framework based on the Bayesian theory.

Even when the goal of segmentation appears to be relatively modest, a unified approach is still not available today.

* Corresponding author. Tel.: +33-472-436-467; fax: +33-472-436-312.

E-mail address: zouagui@creatis.insa-lyon.fr (T. Zouagui).

Since a common mathematical framework seems to be unachievable, we propose to define a functional model of the segmentation methods. This model is a general framework that integrates a maximum number of segmentation techniques. It is implemented in a software tool which demonstrates the practical validity of this approach. By offering a unified view of image segmentation, such a framework can also be used further as a tool that could facilitate the structural comparison of methods and evaluate the original aspect of a segmentation method. Finally, it would be helpful for software implementation and test of segmentation techniques.

A preliminary overview of this model has been given in Ref. [17] with a focus on its C++ implementation. In this paper, we detail the functional model in Section 2. Section 3 deals with the decomposition through the functional model of some existing image segmentation methods and gives some experimental results to validate the proposed framework. In Section 4, we present the programming paradigm associated to the model. Section 5 describes and illustrates the chaining of segmentation operators and its efficiency.

2. The functional model

2.1. Model overview

The core of our functional model (FM) is a segmentation operator (SO) presented in Fig. 1. This SO is a functional structure that represents the image segmentation process. It is always composed of five elementary blocks, which are named Measure, Criterion, Control, Modification and Stop. The segmentation process is achieved through one or more iterations of these blocks. Only one SO is generally sufficient to define and build a “simple” segmentation method. Complex methods, like co-operative ones [15], are represented by a composition of similar or different segmentation operators. This will be discussed in Section 5.

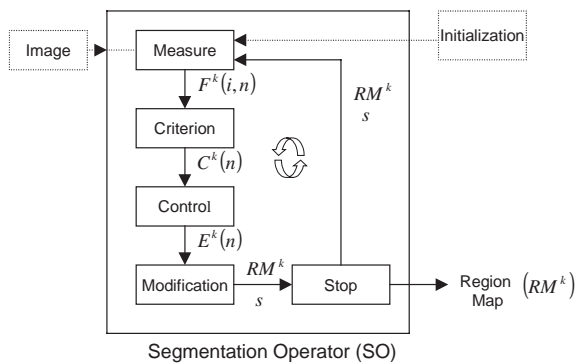


Fig. 1. Overall structure of the functional model of image segmentation.

2.2. Theory and description of the FM components

In this section, we describe the elementary blocks and present the rules that must be applied to enforce the FM.

2.2.1. The Measure block

The feature calculation process is the initial task that is required by the FM. The role of this block is to create at each iteration k a set of M scalar measures $\{F^k(i, n), i = 1..M, n = 1..N\}$ for each region n among the N regions of the image. It needs the original image and the current segmented image (region map at iteration k , noted RM^k). Generally, these measures are related to the region homogeneity and the boundary gradients, or the neighborhood relationship between adjacent regions, or the edge properties. The same measures can be used in different segmentation strategies. For example, the variance of a region can be found in a thresholding method, a splitting one, or an active contour one.

This block has two operating modes:

- (1) A local mode, where the measures are evaluated at the position (site) s which is specified by the Modification block (see Section 2.2.4). Usually, in this mode the measures are computed in a neighborhood of the point s , and they take into account all the pixels belonging to this neighborhood. For example, a local mean measure is done by averaging the gray levels of all the pixels inside a square window of width W centered at position s .
- (2) A global mode, where the measures are evaluated on the whole image for each region. For example, we can compute the gray level variance of a region, the surface, or a contour energy.

2.2.2. The Criterion block

Segmentation is the process of dividing up an image into a set of regions which are uniform and homogeneous according to some characteristic. This homogeneity is generally defined by a criterion. In the FM, the Criterion block receives all the measures from the measure block and builds a scalar criterion $C^k(n) = f(F^k(i, n))$ for each region n . This homogeneity criterion is called energy, potential (...) depending on the segmentation method context.

In our FM, the criterion $C^k(n)$ is used to identify significant changes in the segmentation results from one iteration to the next one. We define that it must decrease when the segmentation map approaches “a good result”.

One of the most common criteria is an additive combination of measures weighted by hyper parameters $w^k(i, n)$. These hyper parameters can be fixed or evolve through the iterations k . This criterion is given by the following equation:

$$C^k(n) = \sum_{i=1}^M w^k(i, n) F^k(i, n), \tag{1}$$

where n is the region number and i the measure number.

Table 1
List of blocks obtained after several methods implementation

Block type (class)	Functions
Measure	1. Gray level variance 2. Gray level mean 3. Area 4. Local deviation from mean 5. Clique energy 6. Mean square error 7. Local distance 8. Point displacement
Criterion	1. Additive combination 2. Inverse 3. Embedded 4. Magnitude
Control	1. Thresholding 2. Derivative 3. Maximum 4. Minimum 5. Linear
Modification	1. Orthogonal splitting 2. Fixed control point displacement 3. Pixel labeling 4. Histogram thresholding 5. Merging 6. Dilation 7. Contraction 8. Adaptive control point displacement

Criteria can be also non-linear and can contain Min/Max operators or logical operators [18].

2.2.3. The Control block

The Control block evaluates for each region the “need” of evolution of a segmentation map. It takes as input the criterion values $C^k(n)$ and produces the control value $E^k(n)$ for each region n . This value is normalized between -1 and 1 . A positive control value $E^k(n)$ means that the region n must be modified because it doesn’t reach the required quality. A null control value of a region n means that this region has reached the required quality. A negative control value means that the considered region has been modified beyond the expected quality.

Different kinds of Control blocks can be implemented (Table 1). For example, a differential control is given by the equation below:

$$E^k(n) = \frac{C^k(n) - C^{k-1}(n)}{C^k(n) + C^{k-1}(n)}, \quad (2)$$

where k represents the current iteration and n is the region number.

2.2.4. The Modification block

The Modification block contains the strategy for the modification of the segmentation map. It can be considered as the core of the segmentation process. It can be very complex but, in every case, it has to respect the Control block requirements $E^k(n)$. To modify the region map, this block can do all the measures it needs on the original image and on the current segmentation map.

By now, we have identified three kinds of Modification blocks:

- (1) Modification with a constant number of regions (region growing, active mesh, ...).
- (2) Modification with creation of new regions (splitting, ...).
- (3) Modification with deletion of regions (merging, ...).

For the methods working in local mode, like Markovian based ones, this block transmits to the Measure block the site position s to be used in the next iteration. This block gives as output the new segmentation map RM^{k+1} build from its modification strategy.

2.2.5. The Stop block

The Stop block has to stop the iterative process using strategies not compulsory linked to the region measures (homogeneity...). An example of such a block is a detection of the stability of the segmentation map between two successive iterations. Note that “stopping” criterion based on an evaluation of the quality of the segmentation map should be implemented preferably into the Measure/Criterion/Control blocks. Indeed, if the control value $E^k(n)$ is equal to zero, then the segmentation map will be stable. Nevertheless, this is not necessary because one can choose to stop a segmentation process with different criteria than the one used for the evolution control, like a maximal number of iterations.

3. Methods decompositions

Experiments were carried out to validate the proposed FM. Tests were conducted on different segmentation methods [8–10,18–27]. To illustrate this, we present the decompositions and the implementation results of a split method, a Markov random field (MRF)-based method and an active contour method.

3.1. Decomposition of a splitting based segmentation method

The basic idea of the quadtree region splitting [10] is to recursively break the image into a set of quadrangular disjoint regions until each sub region becomes homogeneous. The decomposition of this method according to the FM is shown in Fig. 2.

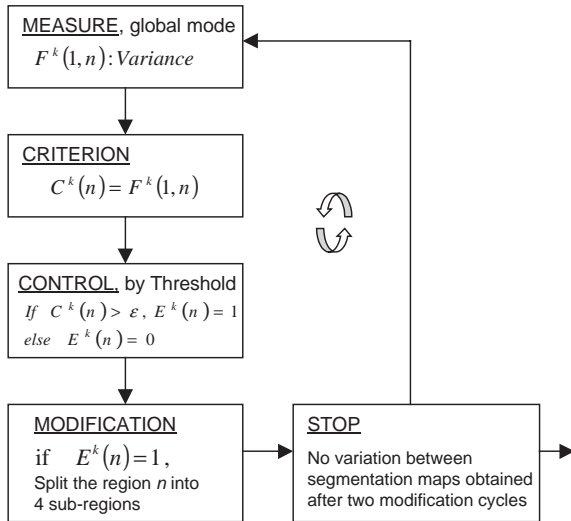


Fig. 2. Functional model of a Split method.

The similarity constraint in this method is the gray level variance. So, the Measure block computes the variance of all the regions in the image at each iteration. The Criterion value is equal to the measure value itself. The Control block is a control by threshold where the control value $E^k(n)$ is set to one when the criterion for region n is greater than a threshold ϵ , and zero otherwise.

The Modification block represents the splitting process. If the control value of a particular region is positive, then the region is split into four quadrants. This process is repeated for all the regions of the image. When the control value is equal to zero, it means that the square region created in this way is homogeneous.

The Stop block simply observes the segmentation map at the current iteration and compares it with the one obtained

one iteration before. If there is no variation, the segmentation process is stopped.

Fig. 3 presents the results obtained after several iterations of the split process implemented by the FM of Fig. 2. In Fig. 4, we represent the evolution of the global criterion value with a number of iterations for the split method. The global criterion GC^k is defined by Eq. (3) where N is the total number of regions in the image, k is the iteration number and $C^k(n)$ the criterion value of region n .

$$GC^k = 1/N \sum_{n=1}^N C^k(n) \tag{3}$$

From Fig. 4, we note that the global criterion is decreasing while the segmentation process progresses. A good segmentation result is obtained after 7 iterations. All the resulting regions are homogeneous compared to the given variance threshold ($\epsilon = 20$). This is in total accordance with our FM criterion rule.

3.2. Decomposition of a MRF-based segmentation method

As an example of MRF-based segmentation methods, the method proposed by Pappas [8] is decomposed and implemented with our FM working in local mode. This segmentation method uses a maximum a posteriori (MAP) criterion optimized by iterated conditional modes (ICM). The a priori model used to characterize the segmented image follows a Gibbs distribution with one or two-point clique. A white Gaussian noise is chosen for the conditional probability modeling the noise. Fig. 5 represents the resulting blocks.

A local mean m_s^n (gray level mean of the neighbor pixels of the site s having label n) is computed inside the Measure block. The first measure $F^k(1,n)$ is a local deviation from the mean and is defined by

$$F^k(1,n) = (y_s - m_s^n)^2, \tag{4}$$

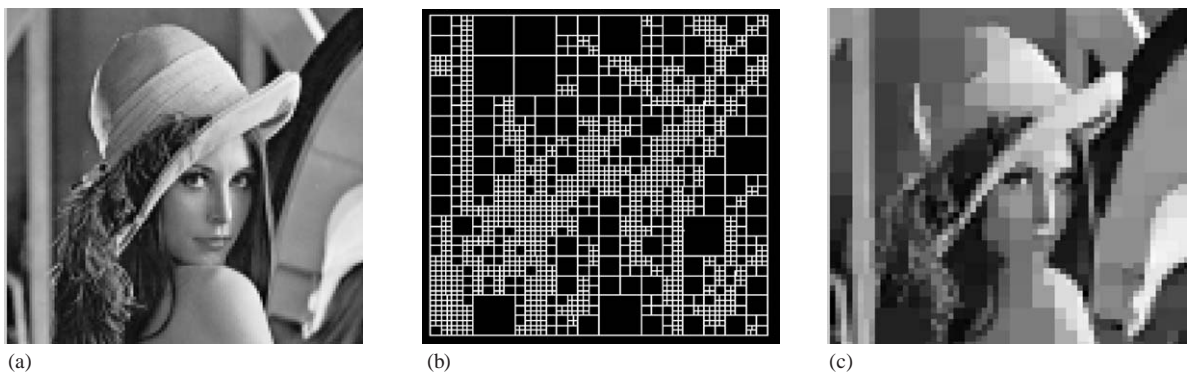


Fig. 3. Results of the split method: (a) original image, (b) polygonal structure of the resulted image, and (c) regions of the image filled by their mean intensity.

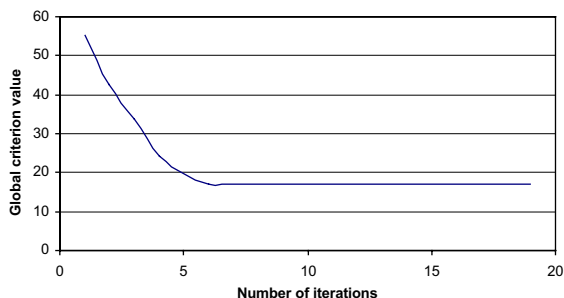


Fig. 4. Global criterion evolution through iterations for the split method decomposition.

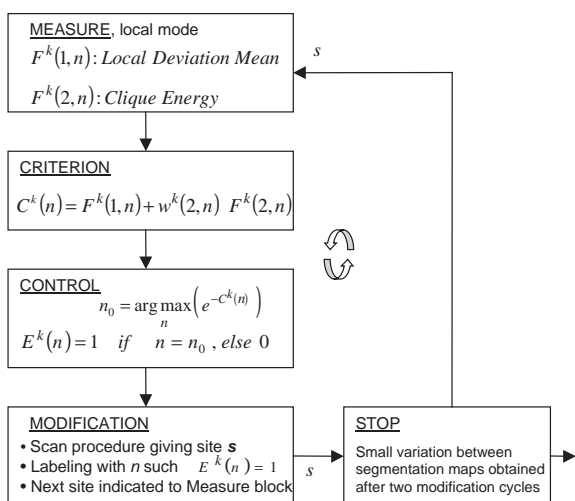


Fig. 5. Functional model of a Markovian based segmentation.

where y_s is the gray level of the site s . The second measure $F^k(2, n)$ associated with the a priori model and the clique potential $V_c(s)$ is given by

$$F^k(2, n) = \sum_{c \in CQ} V_c(s). \tag{5}$$

The summation is over all the cliques CQ and the clique potentials V_c [8] depend only on the pixels with label n in the segmentation map RM^k that belong to clique c .

Note that the Measure block gives a set of two measures for each possible label (region). The current site s is indicated to the Measure block by the Modification block, which corresponds to the FM local mode.

The Criterion block is simply an additive combination of the two measures. The parameter $w^k(2, n)$ is used to control the relative weight of the two measures.

The Control block contains the optimization method, which is deterministic in our case (ICM). The ICM leads to a Control block which finds the label $n = n_0$ for which $e^{-C^k(n)}$ is maximum over n and then “tells” to the Modifi-

cation block to expand the region n_0 . “Expanding” means that the considered site s must take the label n_0 so “the region” n_0 wins one pixel.

The Modification block simply gives to the current site the label n for which $E^k(n)$ is equal to 1. Then, it chooses the next site to be considered according to its internal scanning procedure and sends its coordinates to the Measure block. At the end of each scanning, the Modification block updates the region map RM^k .

The Stop block simply looks at the number of modified pixels at the end of each scanning cycle and stops the iteration if this number is under a pre-defined threshold.

Note that a Control block, which computes the minimum of the criterion value, can replace the ICM Control block.

Examples of images obtained with the FM implementation of the Markovian method are shown in Fig. 6. The curve in Fig. 7 illustrates the evolution of the global energy over the whole segmentation process. The aim in this method is to minimize this energy at every point in the image. This global energy decreases until convergence.

3.3. Decomposition of an active contour method

The third method corresponds to an active contour method [28]. We remind that the snakes or active contours [25] are curves defined by a set of control points within an image domain that can move under the influence of internal forces coming from the curve itself and external forces computed from the image data.

The functional decomposition of this method using the local mode produces the blocks represented in Fig. 8.

The Measure block computes two measures (Eq. (6)) corresponding to the displacement of one snake control point at site s given by the Modification block in local mode.

$$\begin{bmatrix} F^k(1, n) \\ F^k(2, n) \end{bmatrix} = \begin{bmatrix} \Delta Ux^k(s, n) \\ \Delta Uy^k(s, n) \end{bmatrix}, \tag{6}$$

where $\Delta Ux^k(s, n), \Delta Uy^k(s, n)$ are the displacement in the x and y directions, respectively.

The displacement vector of the control point s at the iteration k , is noted $\Delta U^k(s, n) = (\Delta Ux^k(s, n), \Delta Uy^k(s, n))$. It is obtained from the $P \times 2$ matrix $\Delta U^k(n)$, which contains the displacement of the P control points of region n (snake), given by the following equation [28]:

$$\Delta U^k(n) = [\gamma I + A]^{-1} [wF_{ext}(U^{k-1}(n)) + \gamma U^{k-1}(n)] - U^{k-1}(n), \tag{7}$$

where $U^k(n)$ is the $P \times 2$ coordinates matrix of all the control points. F_{ext} represents the external forces [25]. w is a weighting parameter. γ is the damping coefficient. A is a $P \times P$ pentadiagonal matrix. I is the identity matrix.

The Criterion block computes the magnitude of the displacement (Eq. (8)). The criterion value $C^k(n)$ decreases

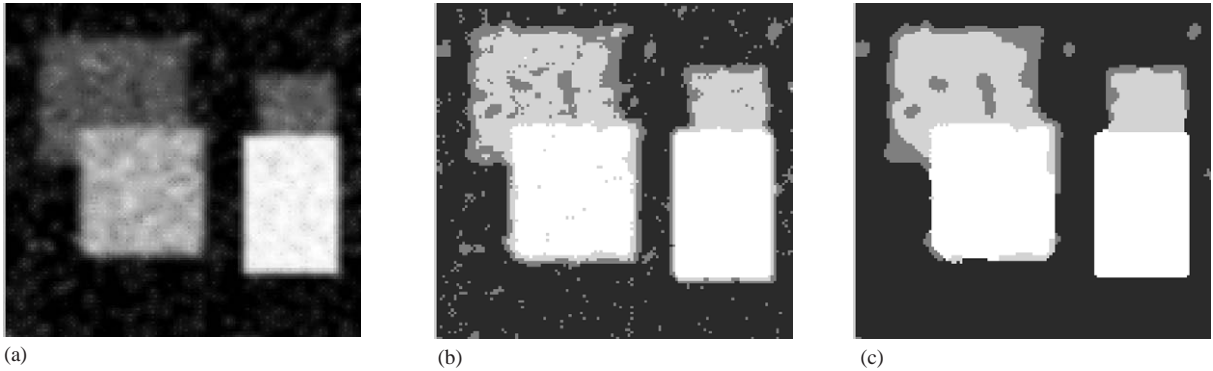


Fig. 6. Results of the Markovian method: (a) original image, (b) initial image with 4 regions, and (c) final result of the Markovian segmentation.

through iterations until the snake reaches its final position.

$$C^k(n) = \sqrt{(F^k(1,n))^2 + (F^k(2,n))^2}. \tag{8}$$

We use a Control block by threshold like in the quadtree method (Section 3.1). It gives a control value equal to zero when the displacement's magnitude is under a fixed threshold ε (Fig. 8).

The Modification block contains the strategy to modify the segmentation map. In this case, it moves the snake control points. This block calculates the new coordinates of a control point belonging to the snake defining region n , by the following equation:

$$U^k(s,n) = U^{k-1}(s,n) + \Delta V^k(s,n), \tag{9}$$

where $\Delta V^k(s,n)$ is a displacement, which must be computed in the Modification block. In this specific case [28], the displacement $\Delta V^k(s,n)$ is equal to $\Delta U^k(s,n)$ obtained from $\Delta U^k(n)$ in Eq. (7).¹

The Stop block combines two ways for stopping the segmentation process. The first way is based on a sufficient number of iterations. The second one stops the process when the region map do not change between two modification cycles.

We have carried out the above experiment with a smooth object as shown in Fig. 9(a). Fig. 9(b) shows the initial position of the snake. It can be seen from Fig. 9(c) that the final snake configuration closely approximates the final boundary.

The evolution of the global criterion through a number of iterations is represented in Fig. 10. In this case, the global criterion is the sum of the magnitude of the displacement of each snake control point. We can see that the criterion value, which corresponds to the final configuration of the snake, is the lowest over the segmentation process.

¹ Note that if some of the measures computed by the Modification block are similar to those calculated in the Measure block for a particular segmentation methods, then the programming strategy we use allows the Modification block to get them in the Measure block for saving execution time.

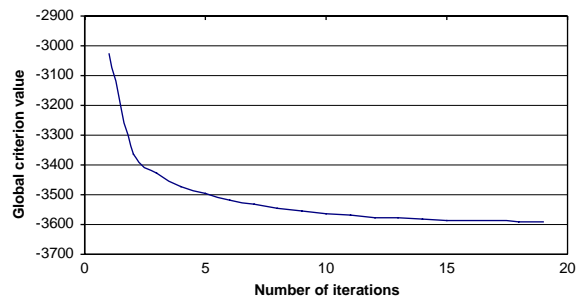


Fig. 7. Global criterion evolution through iterations for a Markovian method.

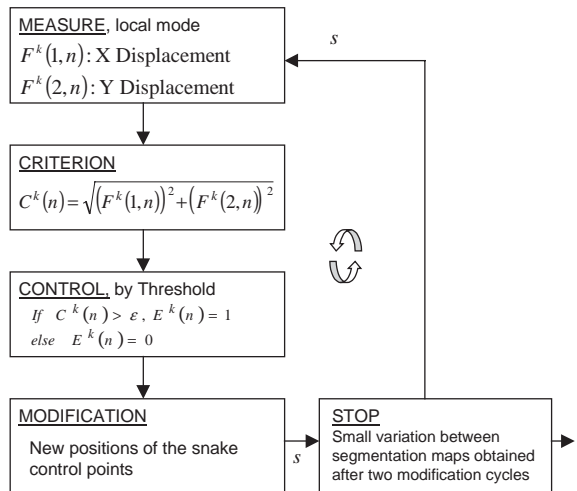


Fig. 8. Functional model of an active contour based segmentation.

3.4. Decomposition strategy

The formalism proposed by our FM is very flexible since it is adapted for many segmentation methods. Nevertheless the decomposition of some methods according to the FM is not always trivial. To overcome the difficulty, it is necessary

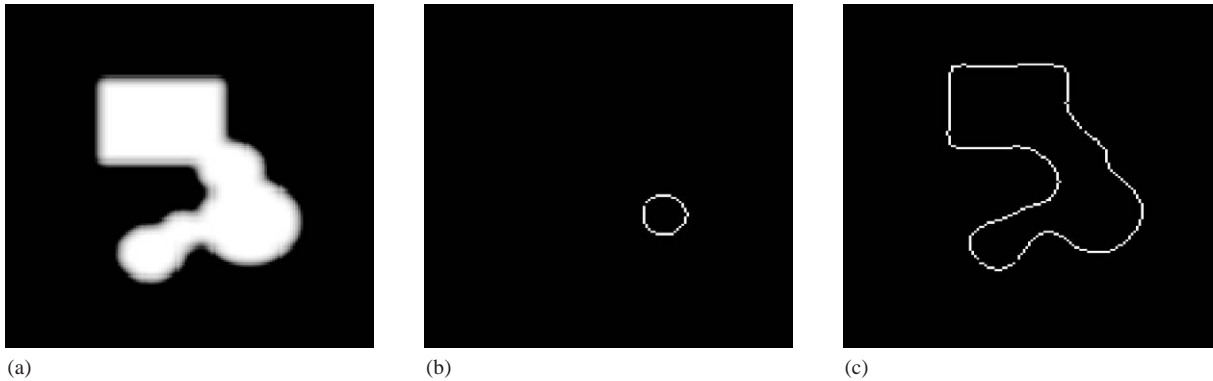


Fig. 9. Results of the snake method: (a) original image, (b) initial contour, and (c) final contour after the convergence of the snake.

to analyze functionally the method with the following strategy. First, identify all the measures and the mode (global or local) in which they are made. These can be of two kinds. The first type of measures are combined in the Criterion block to evaluate the segmentation quality. The second kind of measures are used by the Modification block to change the segmentation map. After this, a control must be defined to assess the evolution quality of the segmentation map. Finally, the segmentation map is modified in the sense given by the original algorithm in the Modification block. Note that this analysis and the corresponding FM implementation often lead to a new insight of the initial algorithm.

The proposed FM is naturally suitable for iterative segmentation. For non-iterative segmentation techniques (like non-adaptive thresholding algorithms), the final segmentation map is directly obtained without any iteration. This is done in the Modification block.

We have shown with the previous examples that the proposed model can decompose different segmentation techniques within the same framework. This allows us to give the term ‘generic’ to our model. We note also that the FM has the same behavior than the chosen original algorithm. We can say that our FM offers a unified vision to image segmentation techniques which can be represented by one SO, and improves the functional comprehension of the methods.

4. Generic model and programming paradigm

The programming paradigm associated to the FM is defined by a set of virtual base classes corresponding to the block types (Measure, Criterion, Control, Modification and Stop). These base classes embed the theoretical constraints of the model, define the interface between the blocks and enforce the block chaining and the corresponding information flow. The building of a new specific block is done by inheritance of the corresponding base class, so no modification of the block interface is allowed. This way, only the



Fig. 10. Global criterion evolution through iterations for an active contour method.

specificity of the new block has to be written and the new block is usable with all the others. The C++ allows such an implementation of the functional model and guarantees the reusability of each specific block with any others.

We have decomposed with our FM several different methods and we have obtained a number of blocks, which are listed in the Table 1. For example, the first method (Section 3.1) is built with Measure 1, Criterion 1, Control 2 and Modification 1.

Note that the same blocks can be used by different segmentation methods. As a consequence the implementation of a new segmentation technique needs only the development of a limited number of new blocks. Furthermore, beginning with an existing method, a new one can be easily implemented and tested by changing some blocks like the addition of a new measure or the change of the modification strategy.

To easily benefit of the FM advantages, we have developed a software (Fig. 11) which allows interactive building of segmentation methods by block selection. Nevertheless, it is never guaranteed that combining arbitrary blocks lead to a useful effective segmentation method. Each button (Measure, Criterion, Control or Modification) allows the user to choose and to configure one of the blocks given in the Table 1. The connection between the blocks is done through

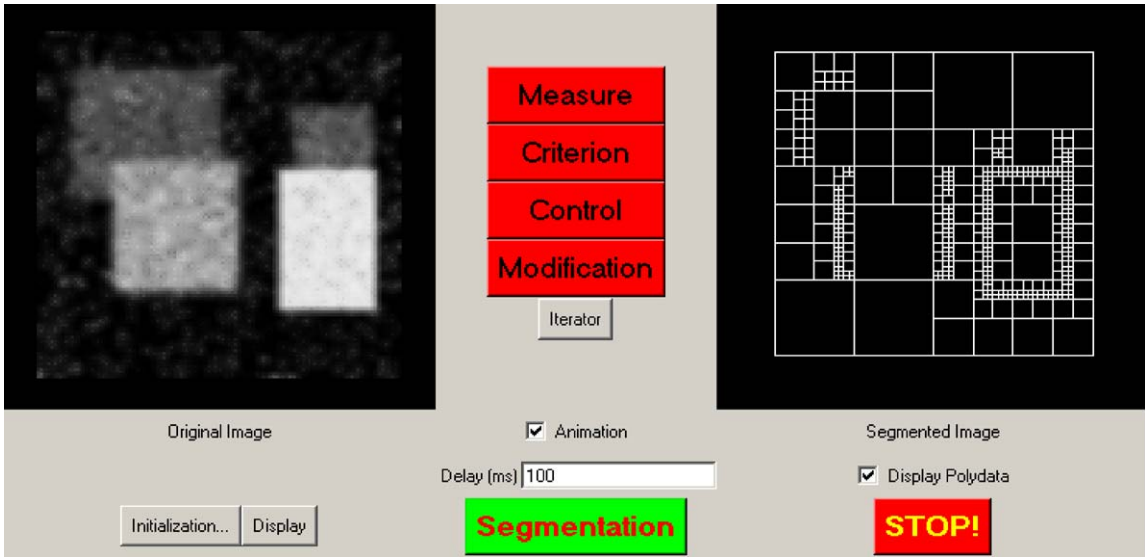


Fig. 11. Segmentation's graphic interface based functional model of the quadtree method.

the base classes and the iteration process is activated by the 'Segmentation' button. The 'iterator' button permits a user to configure the Stop block. The left window shows the original image and the right one presents the segmented image, which evolves through iterations. Note that this software is not dedicated to a specific segmentation technique. The same interface is used for any segmentation methods implemented with the paradigm defined by the FM.

The segmentation model software [17] is based on C++ implementation of the base classes built over an open source library VTK (<http://www.kitware.com>). We chose VTK because of its extensibility and its ease of use with many programming language (C++, Tcl, Python, ...). VTK also gives a pipelining structure well adapted to the FM block chaining. All the results presented in Section 3 have been obtained with the software interface shown in Fig. 11.

5. Chaining segmentation operators

Up to this point, we did not deal with complex segmentation methods because we knew that they could be represented by more than one SO. The study of different segmentation algorithms led us to identify three ways to combine the segmentation operators. These ways are similar to those reported in Ref. [29]. So combination of segmentation operators can be performed through initialization, retroaction or fusion of information.

To illustrate the chaining of SO in the case of integration of segmentation techniques, we present two complex methods: The split-and-merge [10,11] and the region growing method [9,30]. The split-and-merge decomposed by our FM can be represented by a combination of two SO through initialization (Fig. 12). The first SO implements the split

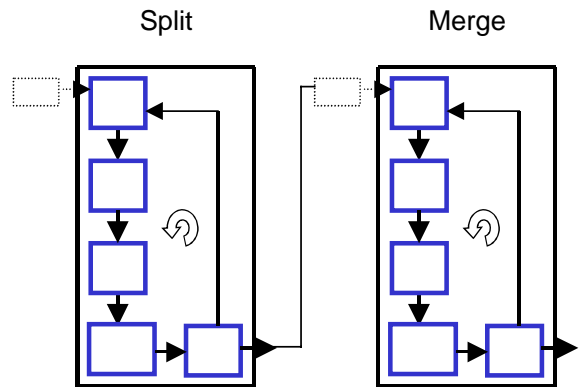


Fig. 12. Serial chaining of the split operator and the merge one.

method and the second one the merge method. When the split operator has finished, it transmits its results to the merge operator which begins its process. An example of segmentation results of the split-and-merge method is shown in Fig. 13.

The second method decomposed is a region growing method [9,30]. It is initially described through four steps. These could a priori lead to four SO but a careful analysis of the method through the FM shows that two SO are sufficient (Fig. 14). The first one is a dilation SO and the second one is a contraction SO. Note that the iteration is repeated over the two SO, which corresponds to a combination through retroaction.

Fig. 15 illustrates the segmented image obtained by the region-growing algorithm implemented by the FM presented in Fig. 14.

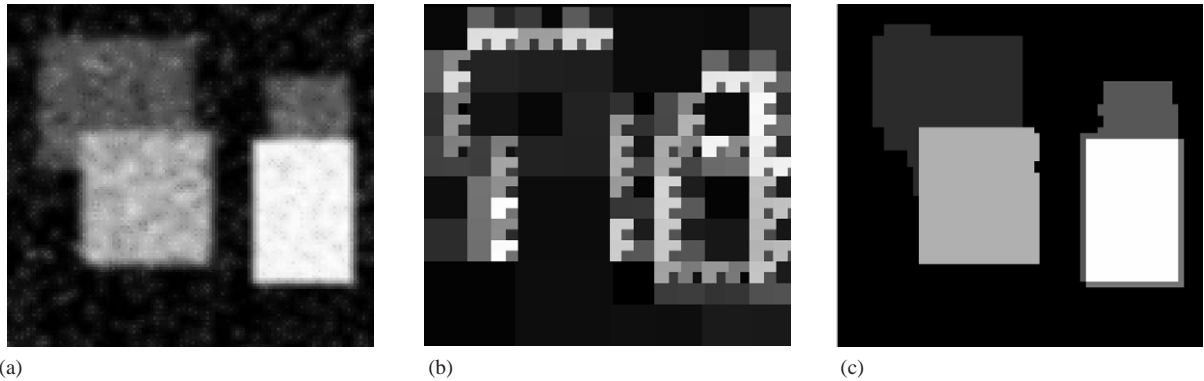


Fig. 13. Results of the split-and-merge method: (a) original image, (b) result of the split SO, and (c) result of the merge SO.

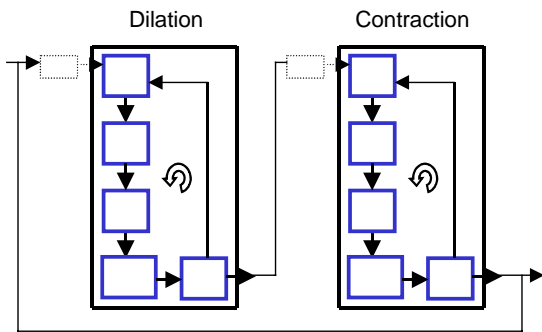


Fig. 14. Combination through retroaction of a dilation operator and a contraction one.

6. Conclusion and perspectives

We have proposed a novel approach that unifies the segmentation techniques under the same framework. This approach is based on a functional modeling of the

segmentation process. Only five elementary blocks are chained in an iterative way within a so-called segmentation operator (SO). These blocks correspond to functions appearing in every segmentation method. Simple segmentation methods can be decomposed with one SO. The chaining of SO allows to build cooperative segmentation methods. Nevertheless, further work has to be done on the cooperation strategies and on the control of multi SO methods.

The FM can be used as a programming paradigm. This is demonstrated with a generic interface which allows to build and control any segmentation method implemented through the FM. The FM can also be used to understand an existing method, to compare the structure of different methods and to define new methods. Obviously, it is not possible to prove that every segmentation methods will fit in the FM, but our experience in integrating a large number of known methods in the model is encouraging. Indeed, our experimental results clearly indicate the efficiency of our model and point the way toward a number of future developments. For example, experiments in progress on various multi-resolution image segmentation techniques show that the FM is still valid in the context of complex methods.

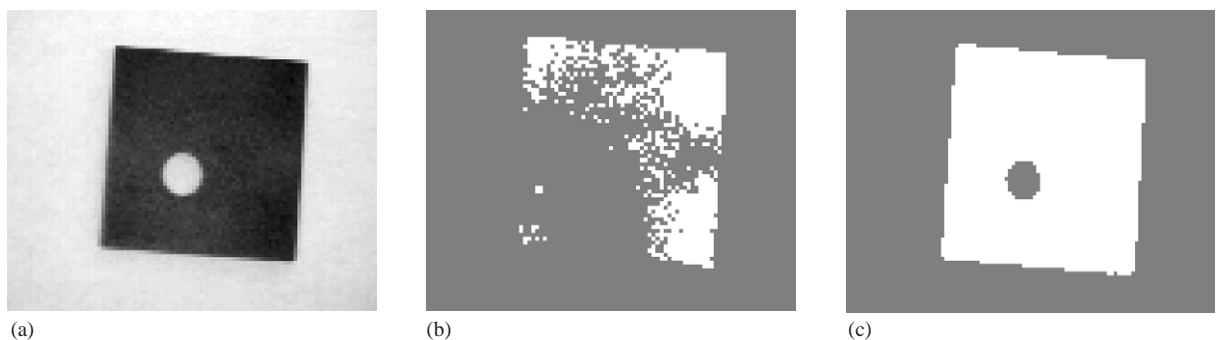


Fig. 15. Results of the region growing method: (a) original image, (b) initial seed by arbitrary thresholding, and (c) segmented image using the FM.

Acknowledgements

This research work is in the scope of the scientific topics of the PRC-GdR ISIS research group of the French National Center for Scientific Research CNRS.

References

- [1] K.S. Fu, J.K. Mui, A survey on image segmentation, *Pattern Recognition* 13 (1981) 3–16.
- [2] R.M. Haralick, L.G. Shapiro, Survey: image segmentation techniques, *Comput. Vision, Graphics Image Process.* 29 (1985) 100–132.
- [3] N.R. Pal, S.K. Pal, A review on image segmentation techniques, *Pattern Recognition* 26 (9) (1993) 1277–1294.
- [4] J.F. Canny, A computational approach to edge detection, *IEEE Trans. Pattern Anal. Mach. Intell.* 8 (6) (1986) 679–698.
- [5] M. Kass, A. Witkin, D. Terzopoulos, Snake: active contour models, *Int. J. Comput. Vision* 1 (1988) 312–331.
- [6] L.D. Cohen, On active contour models and balloons, *Comput. Vision, Graphics, Image Process.: Image Understanding* 53 (2) (1991) 211–218.
- [7] P.K. Sahoo, et al., A survey of thresholding techniques, *Comput. Vision Graphics Image Process.* 41 (1988) 233–260.
- [8] T.N. Pappas, An adaptive clustering algorithm for image segmentation, *IEEE Trans. Signal Process.* 40 (4) (1992) 901–914.
- [9] C. Revol, M. Jourlin, A new minimum variance region growing algorithm for image segmentation, *Pattern Recogn. Lett.* 18 (1997) 249–258.
- [10] G.M. Hunter, K.S. Steiglitz, Operations on images using quadtree, *IEEE Trans. Pattern Anal. Mach. Intell.* 1 (2) (1979) 145–153.
- [11] K. Haris, et al., Hybrid Image Segmentation Using Watersheds and Fast Region Merging, *IEEE Trans. Image Process.* 7 (12) (1998) 1684–1699.
- [12] A. Chakraborty, J.S. Duncan, Game-theoretic integration for image segmentation, *IEEE Trans. Pattern Anal. Mach. Intell.* 21 (1) (1999) 12–30.
- [13] R. Ronfard, Region-based strategies for active contour models, *Int. J. Comput. Vision* 13 (2) (1994) 229–251.
- [14] S.C. Zhu, A. Yuille, Region competition: unifying snakes, region growing and Bayes/MDL for multiband image segmentation, *IEEE Trans. Pattern Anal. Mach. Intell.* 18 (9) (1996) 884–900.
- [15] L. Germond, et al., A cooperative framework for segmentation of MRI brain scans, *Artificial Intelligence in Med.* 20 (2000) 77–93.
- [16] D. Geiger, A. Yuille, A common framework for image segmentation, *Int. J. Comput. Vision* 6 (3) (1991) 227–243.
- [17] H. Benoit-Cattin, T. Zouagui, C. Odet, Functional model and object oriented programming for image segmentation methods, *Proceedings of the International Conference on Quality Control by Artificial Vision, QCAV, Le Creusot, France, 2001*, pp. 391–394.
- [18] O. Monga, B. Wrobel, Segmentation d'images: vers une méthodologie, *Traitement de signal* 4 (3) (1987) 169–193.
- [19] F.R.D. Velasco, Thresholding using the Isodata clustering algorithm, *IEEE Trans. Syst. Man Cybern.* 10 (1980) 771–774.
- [20] S.E. Hernandez, K.E. Barner, Tactile Imaging Using Watershed-Based Image Segmentation, *Proceedings of the Fourth ACM Conference on Assistive Technologies, Arlington, VI, 2000*, pp. 26–33.
- [21] P.T. Sander, et al., Hierarchical Region Based Stereo Matching, *Proceedings of the Conference on Computer Vision and Pattern Recognition, San Diego, CA, 1989*, pp. 416–421.
- [22] M. Spann, R. Wilson, A quad-tree approach to image segmentation which combines statistical and spatial information, *Pattern Recognition* 18 (3/4) (1985) 357–269.
- [23] M. Spann, C. Horne, Image segmentation using a dynamic thresholding pyramid, *Pattern Recognition* 22 (6) (1989) 719–732.
- [24] P.J. Burt, T. Hong, A. Rosenfeld, Segmentation and estimation of image region properties through cooperative hierarchical computation, *IEEE Trans. Syst. Man Cybern.* 11 (12) (1981) 802–809.
- [25] C. Xu, J.L. Prince, Snakes, Shapes, and Gradient Vector Flow, *IEEE Trans. Image Process.* 7 (3) (1998) 359–369.
- [26] Y.L. Chang, X. Li, Fast image region growing, *Image Vision Comput.* 13 (7) (1995) 559–571.
- [27] H.S. Park, J.B. Ra, Homogeneous region merging approach for image segmentation preserving semantic objects contours, *Proceedings of the International Workshop on Very Low Bitrate Video Coding, Chicago, IL, 1998*, pp. 149–152.
- [28] A. Anwander, Segmentation d'images couleur par un opérateur gradient vectoriel multiéchelle et contour actif: Application à la quantification des phases minéralogiques du cliker de ciment, Ph.D. Thesis. INSA de Lyon, Lyon, 2001.
- [29] L. Germond, et al., Trois principes de coopération pour la segmentation en IRM, *Proceedings of the Colloque Reconnaissance de Formes et Intelligence Artificielle, RFIA-2000, Paris, 2000*, pp. 69–78.
- [30] C. Revol-Muller, et al., Automated 3D region growing algorithm based on an assessment function, *Pattern Recognition Lett.* 23 (2002) 137–150.

About the Author—TARIK ZOUAGUI received the Engineer and Magistere degrees both in Electrical Engineering from the University of Science and Technology of Oran (Algeria), and is currently a Ph.D. student at the National Institute of Applied Sciences (INSA) of Lyon, France. His research interests are in the area of image processing with particular focus on image segmentation.

About the Author—HUGUES BENOIT-CATTIN received in 1992 the Engineer Degree (Electrical Engineering) and in 1995 the Ph.D. Degree (Wavelet image coding of medical images), both from the National Institute of Applied Sciences (INSA) of Lyon (France). He is Assistant Professor at INSA Lyon, at the Telecommunications Department. His teaching activities mainly concern Information Theory, Signal and Image Processing. He is in charge of the international relations of this Department. He worked since 1992 at CREATIS Laboratory (CNRS #5515). Member of the Volumic Imaging Team, his research interests include image coding and image segmentation as well as MRI image acquisition.

About the Author—CHRISTOPHE ODET received in 1979 the engineering degree (electrical engineering) and in 1984 the Ph.D. degree (Non-Destructive Testing), both from the National Institute of Applied Sciences (INSA) of Lyon (France). He worked since 1984 at CREATIS as Assistant-Professor. His primary research interests included image segmentation and classification based on texture analysis and signal and image restoration. He has been involved in the application of the previous methods to linear and 2D images sensors for the control of high-speed moving products. He is now Professor at CREATIS and is mainly involved in image processing and segmentation for medical applications such as bone analysis.