

BORG : a Knowledge-based system for the automation of image segmentation tasks

Régis Clouard, Abderrahim Elmoataz, Christine Porquet, Marinette Revenu

► **To cite this version:**

Régis Clouard, Abderrahim Elmoataz, Christine Porquet, Marinette Revenu. BORG : a Knowledge-based system for the automation of image segmentation tasks. 5th Int. Conf. on Image Processing and its applications, 1995, Edinburgh, United Kingdom. pp.524-528, 1995. <hal-00962424>

HAL Id: hal-00962424

<https://hal.archives-ouvertes.fr/hal-00962424>

Submitted on 24 Mar 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

BORG: A KNOWLEDGE-BASED SYSTEM FOR THE AUTOMATION OF IMAGE SEGMENTATION TASKS

R. Clouard, A. Elmoataz, C. Porquet & M. Revenu

GREYC (Research Group in Computer Science, Image and Instrumentation of Caen) - France

1. INTRODUCTION

In this paper, a knowledge-based system that automates the execution of image segmentation processes is presented. Given an image and a set of segmentation tasks, our system, named BORG, generates, performs and optimises the processing of classical segmentation operators, in order to produce new images that satisfy these tasks. Such a system must have capabilities for self-configuration to different segmentation tasks and to different image classes.

Non specialised image segmentation is a complex task because processing techniques are very different from one task to another and from one class of images to another (Matsuyama (1)). It is difficult to conceive general enough algorithms to deal with images that come from various origins.

We are interested in solving segmentation problems by semantic integration of image segmentation procedures (Clément and Thonnat (2)). The system reasoning is based on explicit knowledge about expertise in image segmentation, in order to generate a graph of operators. The purpose of this graph is to produce new images from an input image, so that they match as closely as possible the request that the user has specified beforehand. This is done by *selection*, *parameter adjustment* and *linking* of classical image segmentation operators stored into a run-time library, according to the specific features of the problem and the characteristics of images.

The first part of this paper describes design principles for building image segmentation applications based on the generation of a graph of operators that can deal with the whole set of images belonging to some given class. In the second part, a model of the hierarchical and incremental planning process used for building this graph is presented. The third section deals with the system architecture. Finally, in the fourth one, an application dealing with biomedical images is described.

2. DESIGN PRINCIPLES

The objective of the BORG system is to dynamically build a graph of operators, the execution of which answers an image segmentation request specified by some user, which is valid on a whole set of images belonging to some class. A request submitted to the system includes a list of goals and constraints, a digital image and a description of the contents of the images of that class.

A graph of operators is a linking of operators with adjustable parameters (Figure 2). An operator is an executable program, that takes images as its input, and produces images and/or other data as its output. Its parameters are a means to adjust its behaviour to the distinctive features of images and tasks. Within the graph, operators are linked in such a way that the outputs of some operator constitute the inputs of those that are following it in the graph.

Because it is almost impossible to completely specify all the terms of a request beforehand, the generation of the suitable graph of operators for a whole set of images of one given class is done in three steps:

1) *Initialisation* : The user chooses an image and specifies the task to be performed, as well as boundary-conditions and the quality criteria required on results. These data are used by the system to generate a first graph of operators, the execution of which produces the first resulting images.

2) *Optimisation* : By examining these first results, the user can change some terms of his/her request, so as to refine unsatisfying results.

3) *Validation* : The third step consists in validating the graph on a set of images that best characterize the class under study, by eventually modifying the request again, until complete satisfaction. The resulting graph can be seen as the best answer to that given application.

This graph of operators is then transformed into a C++ program, by retrieving the code of each operator of the graph from the run-time library, where they are stored, and linking these codes thanks to additional functions.

3. REASONING MODEL

In this section, we are now going to describe the planning model, as well as the various types of knowledge that are needed for building a graph of operators, according to the specifications of the user's request.

3.1 Model of Reasoning

The reasoning takes advantage of general knowledge about image segmentation techniques and characteristics of image data to generate the graph of operators. This is done in four steps :

(1) *Plan generation* : First, an appropriate plan to guide the segmentation of a given image is built. This planning phase can only be hierarchical, because segmentation problems cannot be solved following a linear reasoning. The only way is to determine a global strategy from the characteristics of the request, then to choose primitive actions to refine this strategy.

We have chosen to study the segmentation domain under four levels of abstraction corresponding to the various steps of developing a solution starting from the definition of the problem down to the fixing of primitive actions (Clouard et al (3)). We distinguish in decreasing order : the *request* level, the *task* level, the *design* level and the *procedure* level. The former two levels describe desirable actions whereas the latter two describe feasible actions. The planning phase uses the description of the image given by the user (e.g. noise rate, contrast, texture, shape, size of objects) and knowledge about standard image segmentation processes to produce a hierarchical tree of actions by successive refinement : the request is elicited as a sequence of tasks, in which each task can be achieved by a suitable sequence of design decisions; and finally, each design decision is implemented as a sequence of procedures.

(2) *Operator selection*. During this second step, the reasoning deals with the instantiation of the segmentation plan into a graph of specific image segmentation operators by using knowledge about the decomposition of a procedure into a sequence of operators of the library.

(3) *Execution of the plan and parameter adjustment*. Finding the optimal values of parameters can only be performed during the execution phase : each operator is run with various values of parameters so as to optimise a given evaluation function. First, an interval of acceptable values for each parameter must be determined, according to the specific features of tasks and images, then a cost function must be proposed, that will have to be optimised. At this step, intermediate images now exist, that can be evaluated.

(4) *Assessment of the plan*. The quality of the plan must be assessed after each execution and this verification can lead, if necessary, to correcting some parts of the plan. This assessment must be done by taking into account our hierarchical approach, because each action at the first three levels is decomposed into a sequence of sub-actions at the next lower level. Assessment is first a local assessment of the result of each sub-action, and secondly a global assessment of the result of the sequence. As we have chosen to distribute the assessment process on each action of the plan, the results of the execution of operators must be collected and taken back up towards upper levels. Assessment rules are applied to measure the difference between intermediate results and the current goal. The rules at the lower levels deal with technical details whereas those at the upper levels deal with general goals. In the last resort, the assessment of the final result falls within the competence of the user.

3.2. Model of Knowledge

A plan is represented as an AND/THEN tree with five levels of abstraction. This tree represents the successive decomposition of a task at one level, into a sequence of sub-tasks at the next lower level. If tasks are linked by THEN-relations, they are considered as sequential tasks; if they are linked by AND-relations, they are considered as independent and can be executed in parallel.

Four types of knowledge are needed for building the plan, instantiating it, controlling its execution and finally assessing its quality :

(1) *A description of the user's intentions and the context of the application* are essential to direct and control the resolution process. This knowledge constitutes the user's request and the input of the system. In a request, goals describe the nature of the problem, whereas constraints define the quality of expected results. The user has the possibility to select segmentation tasks among a given set of tasks (e.g. isolating objects from the background, extracting boundaries, delineating areas of interest), and then, to precise the context of these tasks thanks to specific constraints also selected among a given set of constraints (e.g. localizing boundaries precisely, extracting step-edge boundaries only). Because characteristics of images are essential to direct treatments, a symbolic description of the input images is given by means of three attribute-value lists describing *physics* about image formation (e.g. type of camera, acquisition conditions), *perceptual* information (e.g. boundary type, size and form of regions) and knowledge about the *semantics* of the scene (e.g. form, size of objects, and relations between objects).

(2) *Knowledge about the domain of application* is a means to associate a sense and a subject to the image under study, in order to understand the request. Not only can this knowledge be acquired from the user at the beginning of each new application, but also interactively owing to ambiguous and incomplete points that must be clarified by trial-and-error processes in the course of resolution.

(3) *Knowledge about Image Processing* is in charge of finding out the right methods to perform Image Processing tasks, as well as evaluating their results. We distinguish knowledge about Image Processing methods on the one hand, and knowledge about the way to use them on the other hand.

(4) *Knowledge about operators* is used to direct the selection, parameter adjustment and linking of operators to achieve Image Processing methods. We distinguish syntactical knowledge, used to build syntactically correct sequences of operators, and semantic knowledge used for selecting and controlling these operators.

As we want to build a general-purpose system, non dedicated to any application, the vocabulary we use must be strictly limited to the segmentation and mathematics domains, i.e. independent of any domain of application. So, the user must be an expert in image segmentation, because the definition of the request requires from him to specify the right objectives and to describe the relevant features.

The problem is solved by a cooperative process between several knowledge sources. A knowledge source can be seen as a module coding a unique know-how, whether it deals with decomposing a task into sub-tasks, executing an operator or assessing results. There are as many knowledge sources as ways of decomposition, execution or assessment. A knowledge source is composed of two parts: condition and action. The condition part is in charge of recognizing a task it can perform, whereas the action part proposes an action to be applied to the plan.

With such a knowledge organization, processing an image consists in building a specific plan of actions, in an incremental way, i.e. by successive execution of knowledge sources, and by taking advantage of the hierarchical levels of abstraction.

4. ARCHITECTURE OF THE SYSTEM

The architecture of the BORG system is based on the BB1 blackboard model (Hayes-Roth (4)). This model defines a database - the Blackboard - and Knowledge Sources (Figure 1).

The database contains the plan of treatments and the graph of operators. Both the plan and the graph are

represented as a unique tree of goals, the intermediate nodes of which are tasks at various levels of abstraction and the leaves are the operators that form the graph. Each node describes a goal to be reached at a specific level of abstraction, and it is linked to a sequence of sub-goals at the next lower level.

The expert knowledge is divided into four types of Knowledge Sources : planning Sources, instantiation Sources, execution Sources and assessment Sources. Each Knowledge Source contains expertise to solve one part of the global problem.

The plan and the graph are built in an incremental and opportunistic way on the database, thanks to the cooperation of Knowledge Sources, the sequence of which is chosen by a control mechanism also based on the blackboard model. The basic cycle is composed of three steps:

- 1) *Updating* an agenda containing all the Knowledge Sources that can contribute to make the present state of the plan progress.
- 2) *Selection* of a Knowledge Source among the agenda.
- 3) *Execution* of that Knowledge Source, which modifies the state of the plan, thus triggering new Knowledge Sources that are placed in turn into the agenda.

This cycle goes on until complete resolution or failure.

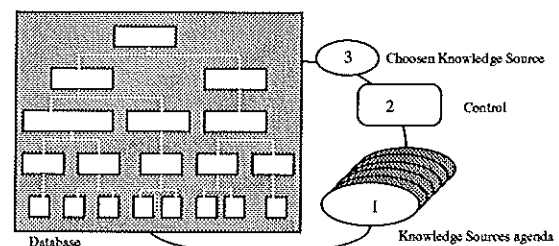


Figure 1 : Architecture of the BORG system

Within the blackboard model, the control problem consists in selecting the next Knowledge Source to be executed among a set of potential Knowledge Sources. To that purpose, each Knowledge Source is characterized by four ratings that are computed according to the present state of the plan: importance, efficiency, credibility, triggering date. The next Knowledge Source to be chosen is the one with the highest value computed from a weighted sum of these four ratings (the highest weight being given to the "importance" factor). In BB1, as well as in BORG model, the control problem is solved also thanks to a blackboard (Clouard et al. (5)).

5. APPLICATIONS AND RESULTS

The system has been used for the segmentation of biomedical images, more precisely microscopic images of human oesophagus cells. We are concerned with the segmentation of cytological images in order to automatically classify cellular categories identified thanks to their nucleus. In a preliminary study, Elmoataz et al. (6) proposed a suitable plan - manually built - to solve this problem. Here, we show the results that were completely automatically obtained on that same application.

The request is to *isolate objects from the background*. Constraints specify that boundaries must be precisely localized and clusters of cells should not be separated.

This request is translated into the task (Isolate objects from the background) with the constraints ((precise boundaries) (under-segmentation)).

The description of the class of images is done by the means of three attribute-value lists dealing with physical, perceptual, and semantic aspects :

- (physical
 - (type-of-scene microscopic)
 - (image-quality sharp)
 - (type-of-image density))
- (perceptual
 - (background yes)
 - (type-of-regions homogenous)
 - (type-of-boundaries step-edge)
 - (aspect-of-objects non-textured)
 - (repartition dissociated))
- (semantic
 - (color-of-background light)
 - (color-of-objects dark)
 - (size-of-objects medium))

The plan generated by the system is shown on figure 3 (1). This plan is based on a pre-segmentation consisting in a background elimination, so as to delineate regions of interest, followed by a re-segmentation of these regions by a division method, in order to get objects corresponding to cells.

The graph obtained by instantiation of that plan is given on figure 3 (2). Its execution on the image of figure 2 (1) is shown on figure 2 (2).

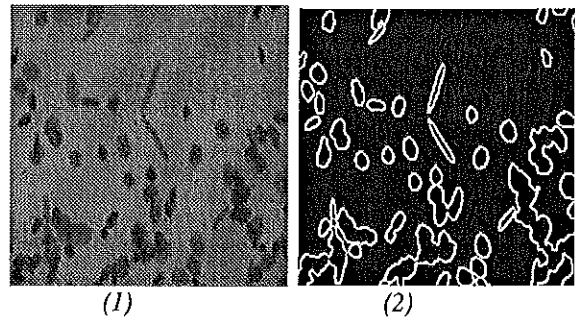


Figure 2 : A cytological image (1) and the final result automatically obtained by BORG (2).

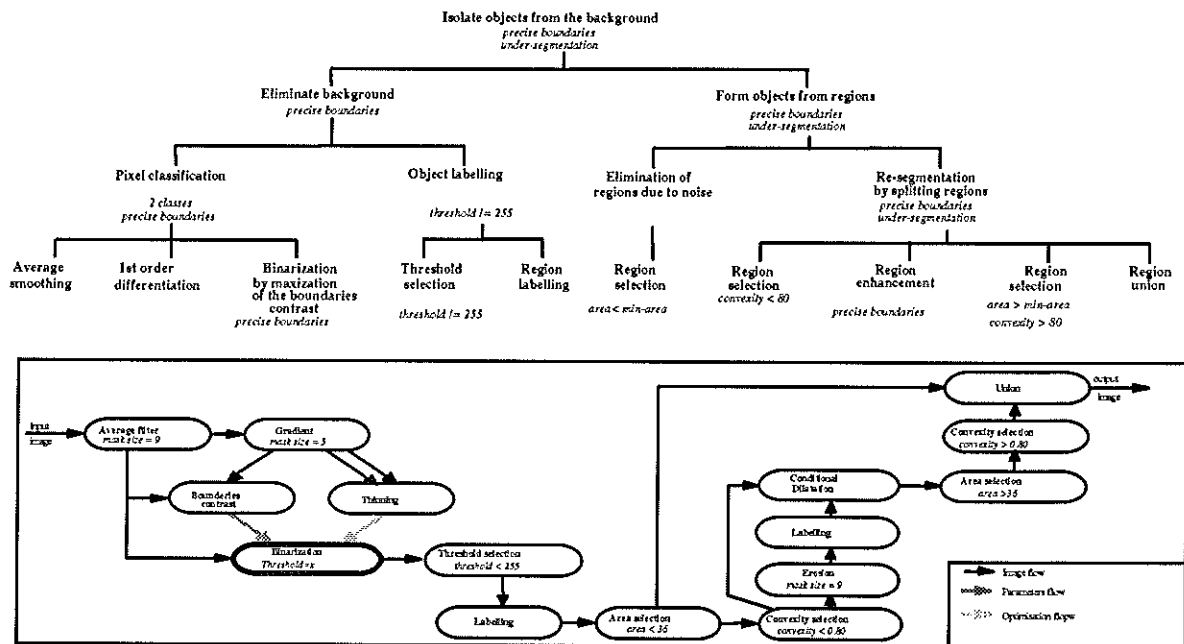


Figure 3 : The plan. (1) and the graph (2) for the image of figure 2 (1).

6. CONCLUSION

A system such as BORG can be seen as an experimenting tool for studying new applications. Even if the user no longer has to directly program Image Processing operators, he/she still has to be an expert in Image Processing when it comes to describe the right tasks and choose the right visual primitives.

We are presently interested in increasing the cooperative aspects of our system, in order to enable various categories of experts - expert of the domain of application, expert in Image Processing and programmer-designer - to cooperate when tackling new applications (Clouard et al. (7)). Two programming modes are available in cooperative BORG: in the manual mode, an Image Processing expert can build his/her own plans; in the automatic mode, the resolution is achieved through collaboration between experts and the system.

Our long-term objective is, through the resolution of various applications thanks to cooperative BORG, to try to propose some kind of computational theory of Image Processing, analogous to Marr and Poggio's theory (8).

ACKNOWLEDGEMENTS

This project is developed within the framework of the "Image Processing and Analysis" Center of Caen. This center covers a wide domain of applications, dealing with biomedical, as well as material or geographic images. The authors are grateful to Mrs P. Herlin of the pathology department of the cancer-research centre F. Baclesse of Caen for providing biomedical images.

REFERENCES

1. Matsuyama, T., 1989, "Expert Systems for Image Processing: Knowledge-based Composition of Image Analysis Processes", *CVGIP*, 48, 22-49.
2. Clément, V. and Thonnat, M. (1993). A Knowledge-Based Approach to Integration of Image Processing Procedures, *CVGIP: Image Understanding*, 57(2), 164-184.
3. Clouard, R., Porquet, C., Elmoataz, A. & Revenu, M., 1993; "Resolution of image processing problems by dynamic planning within the framework of the Blackboard model"; *Proceedings of the SPIE's international symposium: Intelligent Robot and Computer Vision XII*, 2056; Boston, September, 419-429.
4. Hayes-Roth, B. 1985, "A blackboard architecture for control", *Artificial Intelligence*, 26, 251-321.
5. Clouard, R., 1994, "Raisonnement incrémental et opportuniste appliqué à la construction dynamique de plans de traitement d'images" PhD thesis, University of Caen, France.
6. Elmoataz A., M. Revenu and C. Porquet, 1992, "Segmentation and classification of various types of cells in cytological images", *Image Processing and its Applications*, Maastricht, 385-388.
7. Clouard, R., Revenu, M., Elmoataz, A., & Porquet, C, 1995, "A software workbench for knowledge acquisition and integration in Image Processing", *Int. Workshop on the Design of Cooperative Systems*, Juan-Les-Pins, January, 298-313.
8. Marr, D. & Poggio, T. (1979). A computational theory of human stereo vision; *Proc. of the Royal Society of London*, serie B(204), England, pp. 301-328.