IMAGE PROCESSING USING TOPOLOGICAL OPTIMIZATION

Pham The Bao (1), **Image processing Group of Workshop** (2)
(1) Faculty of Mathematics & Infomatics, University of Natural Sciences – VNU-HCM
(2) Workshop on Mathematical Modeling in Life and Materials Science and Technology – Trieste, Italy
(Received 30 November 2004, Revised 09 March 2005)

ABSTRACT: *Image processing is very important, how choosing better algorithms that you have good application so is necessary. This paper, I introduce brief about topological optimization for image processing and the result of my group (working in Workshop on Mathematical Modeling in Life and Materials Science and Technology, from 08/03/2004 to 02/04/2004, ICTP). We use topological optimization to restore and segment not only gray image but also color image (RGB). We implement algorithm by MathLab, and test for some different images.*

Introduction:

Image processing [1] is a fundamental tool in many applications that range from satellite data interpretation to medical imaging. The whole processing of an image can be divided in several stages, namely those of acquisition, enhancement, restoration, segmentation, compression and finally recognition and classification. In what follows, we will be concerned about restoration and segmentation of images. By restoration we understand the removal of unwanted noise in the image while preserving the features of interest, while segmentation means splitting the image in meaningful parts. Several algorithms [2] are available for these processes, with different advantages and drawbacks. In this work we developed ideas involving tools from topological optimization [3] in achieve both restoration and segmentation at once.

This document is organized in the following way. First, we describe a method to denoise or restore images by means of the Tikhonov[4] regularization. Then, the topological gradient is described and the basic algorithm is presented. We present the segmentation algorithm in the next section and show the results on some test images. Some conclusions are drawn from this work in the last section.

Computer Image Representation:

An image is represented in terms of its intensity function

$$v \in L^2(\Omega), \quad \Omega \subset \Re^2$$

which in the case of gray images takes values from 0 to 255, where 0 represents black and 255 white color. In case of a RGB image, we have three components for the intensity function giving the red, blue and green colors.

$$v \in (L^2(\Omega))^3, \quad \Omega \subset \Re^2$$

Image Restoration and Tikhonov Regularization:

Image restoration consists of eliminating the noise of an image, while at the same time preserving the desired features. One way to restore images is based on the following idea. The noisy image v can be thought of as belonging to some functions space without any regularity conditions, $L^2(\Omega)$ for instance, while the denoised one u belongs to some space of more regular functions, $H^1(\Omega)$ for instance. The denoising process is then described as looking for a way in which to approximate the given $L^2(\Omega)$ function by some representative in $H^1(\Omega)$.Tikhonov approach consists of choosing u as a minimize of the following functional

$$\min_u \int_\Omega |v - \mathrm{K}u|^2\, dx + c\int_\Omega |\nabla u|^2\, dx$$

Where K is the canonical embedding of $H^1(\Omega)$ into $L^2(\Omega)$,

$$K : H^1(\Omega) \rightarrow L^2(\Omega)$$

An intuitive way of looking at this formulation is the following. The first term in the functional is trying to force the denoised image u to be as close to the original noisy one v as possible, while the second term is forcing to do so without large gradients. In more formal terms, the second term has a regularizing effect that allows obtaining a well posed problem.

The corresponding Euler-Lagrange equation for the minimize is given by

$$\begin{aligned} u - div(c\nabla u) &= v \text{ in } \Omega \\ \partial_n u &= 0 \text{ in } \partial\Omega \end{aligned} \qquad (1)$$

The regularization parameter c is assumed to be constant in the domain. The solution of this equation is the restored image in this approach. As the restored image is smoother than the input image, the noise in the image is removed as can be seen in figure (1). The main problem with this approach is that together with the noise it removes edges and other features of the original image which should be retained, producing an unwanted blurring.

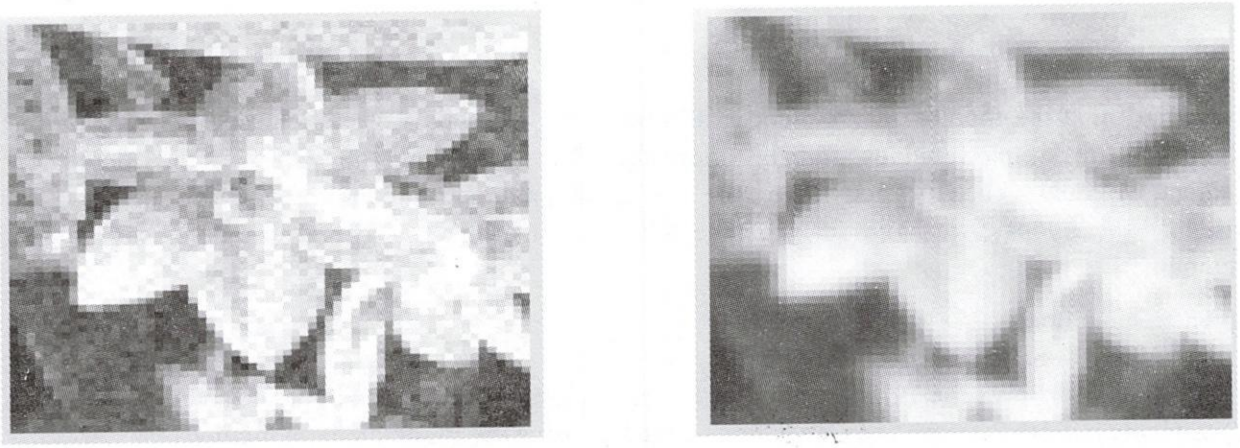


Figure 1. Initial noisy image, and its Tikhonov regularization for c=1.

A way of trying to smoothen and at the same time, to retain the edges is by taking a variable c instead of a constant c. In particular, using the fact that edges correspond to large gradients in the image, we can choose c to be a decreasing function of the modulus of the gradient. This approach, which amounts to solving a nonlinear and possibly degenerate elliptic equation, is in general successful, but we used a different approach.

The starting point is to use the linear elliptic equation (1) with a variable c, where we define c to take only two values –a constant c_{max} and a small value ε. We will try to define c to be small at the edges, as this will ensure that the smoothing is turned off and we retain the sharp features of the image. At the same time, the fact that c will be c_{max} on the other part of the domain ensures smoothing and the removal of noise. The main difficulty now is to define where c is going to be small and that is done by using ideas coming from topological optimization.

The Topological gradient:

In topological optimization, we have a certain domain Ω, and a certain cost functional J defined on a class of functions that solve a partial differential equation in the domain. The aim is to decrease the cost functional by drilling a small hole in the domain. We have to locate the optimal position of this hole. This is done by measuring the variation of the functional after creation of a hole. Since the topology of the domain changes, we speak of the topological gradient or derivative instead of the classical derivative (which cannot be defined). We have the following asymptotic expansion

$$j(\rho) = j(0) + f(\rho)g(x_0) + o(f(\rho))$$

Where $j(0) = J(u_\Omega)$, $j(\rho) = J(u_{\Omega_\rho})$. Here $f \geq 0$, and $\lim_{\rho \to 0} f(\rho) = 0$.

In order to apply this method for the image restoration, we adopt the following framework. We consider as our domain the domain of the image and we take the equation (1) given by Tikhonov regularization. We assume a cost functional of the form

$$J(u) = \alpha \int_{\Omega_\rho} |\nabla u|^2 \, dx + \beta \int_{\Omega_\rho} |u - v|^2 \, dx$$

Notice that the parameters α and β are regularization parameters and that the functional actually consists of the first two terms of the Mumford Shah functional. In this setting, the topological gradient at x_0 is given by:

$$g(x_0) = -\alpha \nabla u(x_0).\nabla p(x_0) - \alpha |\nabla u(x_0)|^2 - \beta |u - v|^2$$

Where $p(x)$ is the solution to the adjoin problem for the PDE:

$$\partial_n p = 0 \text{ in } \partial\Omega_\rho \qquad (2)$$

$$p - div\,(c \nabla p) = -D_u J(u) \text{ in } \Omega_\rho$$

We remark that the topological gradient is different from the standard gradient and depends on the adjoining state which provides global information. After obtaining the value of the topological gradient, we drill holes at locations where the topological gradient is the most negative.
Based on these ideas, we have designed the following algorithm based on the topological gradient method:

Given an input image v,

Step 1 - Start with a constant $c = c_{max}$

Step 2 - Solve the PDE (1) in order to obtain u, compute the solutions of the adjoin equation (2) to obtain p. Both the equations have been solved by using the Finite element method.

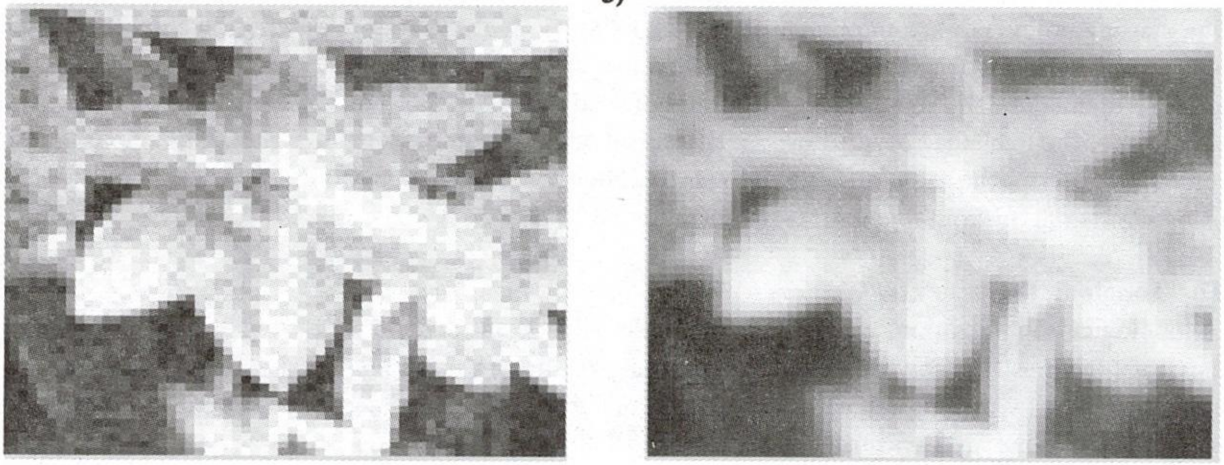
Step 3 - Compute the topological gradient g at each point and find the locations where it is most negative and insert a hole at those points

Step 4 - Define $c = \varepsilon$ at the holes and retain $c = c_{max}$ otherwise

Step 5 - If the stopping a criterion is satisfied stops the process, else, go to 2.

The results of this algorithm are presented in **figure 2**.

a) b)



c) d)

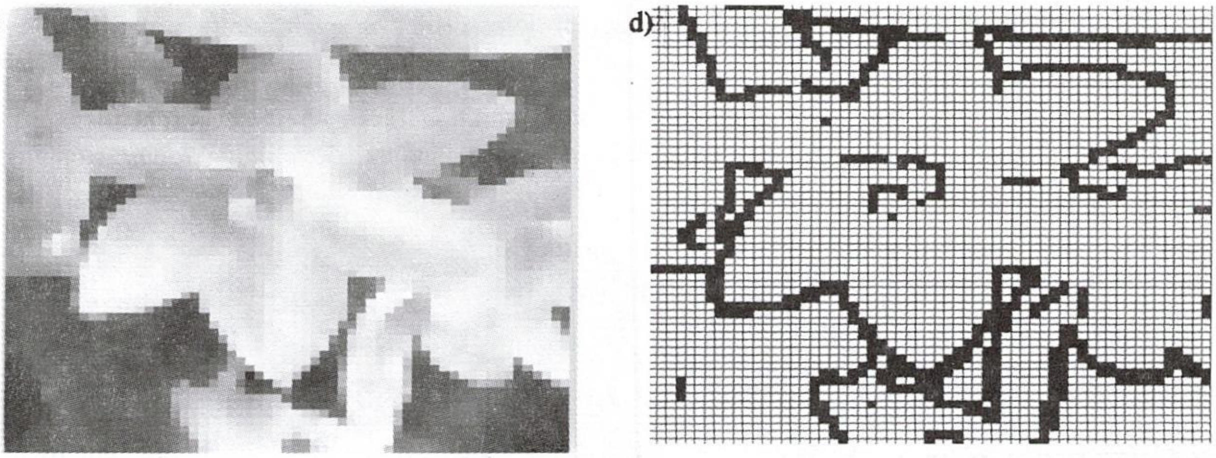


Figure 2. a) Initial image, b) Tikhonov regularization with $c_{max}=1$ (step 2), c) final restored image, after 20 iterations, d) final value of $c(x)$ (white: $c=c_{max}$, black: $c=\varepsilon$).

Observe that while the initial Tikhonov regularization is very smooth and blurred the final restored image presents at the same time sharp edges and good noise removal. Also, notice that the final set where $c=\varepsilon$ gives a good indication on the location of the edges. This remark is the basis of the application of this technique to image segmentation.

Segmentation:

As mentioned before, the image segmentation goal is to fragment the image into meaningful parts. This is traditionally achieved either by grouping together parts of the image with similar pixel properties, or by identifying the edges of the image and filling the regions thus determined.

The method presented before has two features that help in this process: on the one hand, as already stressed, the set where $c=\varepsilon$ provides an indication of the edge location; on the other hand smoothing inside the regions where $c=c_{max}$ provides better pixel uniformity inside each. However, if the hole creation process is pursued too little, or too much, these advantages are lost because edges are improperly defined (in the first case), or because too much detail is being captured and edges become too thick (in the second one). This is clearly seen in figure 3, where the evolution of the set where $c=\varepsilon$ is shown for increasing number of iteration of the main algorithm. For 5 or 10 iterations, the edges are not yet fully defined; 15 or 20 iterations allow a good identification of regions inside the original image, while 25 to 30 iterations are capturing lots of small lighting changes inside the regions, and edges are becoming too thick.

In order to overcome these problems, the following criterion based on region homogeneity for stopping the hole creation process was devised. After every 5 iterations, a region filling algorithm is used to identify all maximal connected regions (for a given connectedness definition) determined in the set where $c=c_{max}$ (i.e. outside the already identified edges). For each region detected by this process, we check whether the standard deviation of the intensity function for the pixels in the region is within some prescribed tolerance and declare the region to be homogenous if this criterion is satisfied. If any of the maximal connected regions is still not homogeneous, the hole creation process is continued [5]. This stopping criterion has worked well in a large number of cases. Results for there are shown in figure (4).

Summary:

The topological gradient method has been successfully adapted to solve restoration and segmentation problems in image processing. An interesting feature of the technique is that restoration and segmentation are achieved at the same time. Matlab routines were developed that can handle both gray and RGB color images, and region filling and merging algorithms were devised and implemented, that allowed to analyze several images with different levels of complexity. The main tuning parameters of the developed technique are the values of α and β in the cost functional, the value of the diffusivity c in the smooth regions (c_{max}), the number of holes drilled in each iteration and the total number of holes.

Further research is deserved in several areas, like including additional terms in the cost functional, use of small cracks instead of holes in the topological gradient technique, developing robust stopping criteria, etc.

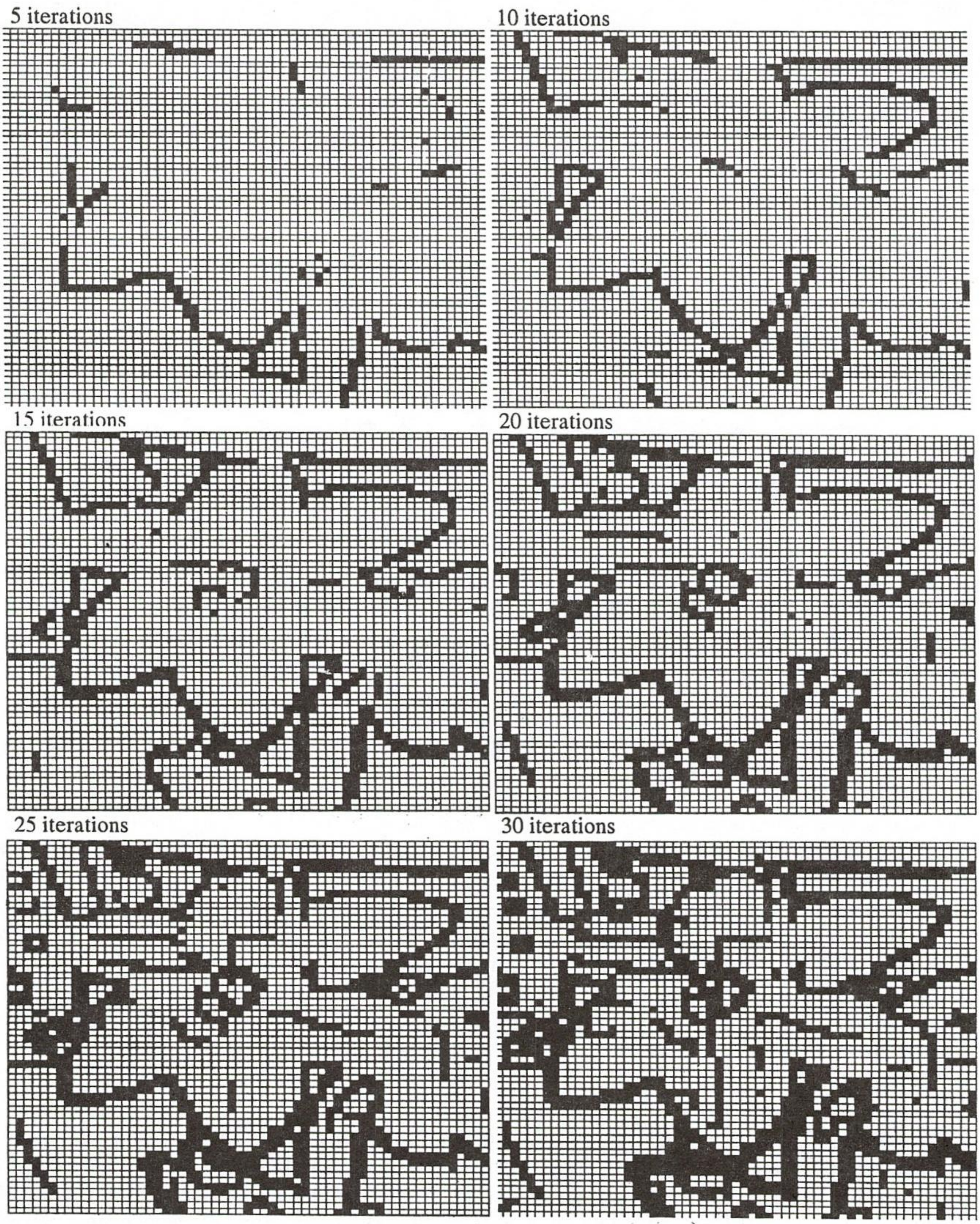
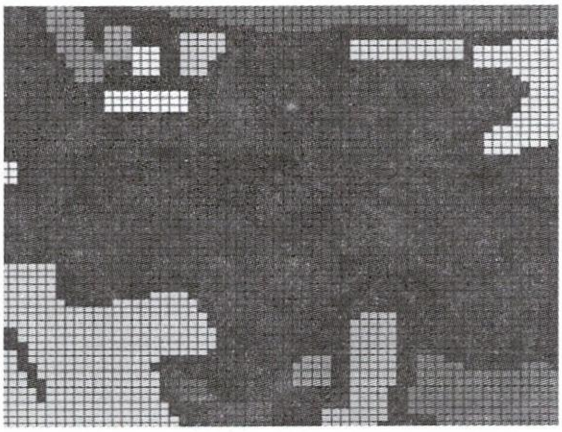


Figure 3. Evolution of the set where $c=\varepsilon$, for an increasing number of iterations.



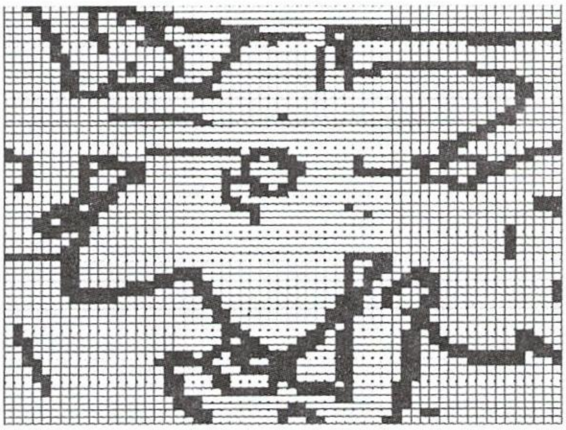


Figure 4: Image segmentation achieved after 20 iterations, with a tolerance for the standard deviation of the pixels of any successfully segmented region equal to 30.

If you need this MathLab source code, you can email me to request.

XỬ LÝ ẢNH VỚI TOPOLOGICAL OPTIMIZATION

Phạm Thế Bảo [1], Nhóm xử lý ảnh của hội thảo [2]

[1]Khoa Toán – Tin học, Trường Đại học Khoa học Tự nhiên – ĐHQG-HCM

[2]Workshop on Mathematical Modeling in Life and Materials Science and Technology – Trieste, Italy

TÓM TẮT: Xử lý ảnh là một lãnh vực nghiên cứu rất quan trọng, việc chọn lựa các thuật toán tốt hơn cho các ứng dụng khác nhau cũng rất quan trọng. Trong phần này, xin trình bày ngắn gọn về topological optimization trong xử lý ảnh và kết quả của nhóm xử lý ảnh trong hội thảo "Workshop on Mathematical Modeling in Life and Materials Science and Technology", được tổ chức từ ngày 08 tháng 03 năm 2004 đến ngày 02 tháng 04 năm 2004 tại ICTP. Nhóm xử lý ảnh đã giải quyết bài toán: khôi phục và phân đoạn ảnh với kỹ thuật topological optimization, kết quả có được rất tốt không những cho ảnh xám mà còn cho cả ảnh màu (RGB) và thuật toán được thử nghiệm trên nhiều loại ảnh khác nhau, thuật toán được cài đặt trên MathLab.

REFERENCE

[1]. Rafael C.Gonzales and Richard E.Woods, *Digital Image Processing*, Addison Wesley 1993.

[2]. Alan Watt and Fabio Policarpo, *The Computer Image*, acm Press – Addison Wesley 1998.

[3]. Amstutz Samuel, *PhD. Thesis*, University Paul Sabatier Toulouse – France 2003.

[4]. Internet, http://www.acm.org

[5]. Internet, http://www.ph.tn.tudelft.nl/Courses/FIP/noframes/fip-Morpholo.html