Color Segmentation Of 2D Images With Thresholding

Hepzibah A. Christinal[1,2], Daniel Díaz-Pernil[1],
Pedro Real Jurado[1],S. Easter Selvan[3]

[1]Research Group on Computational Topology and Applied Mathematics
University of Sevilla, Avda. Reina Mercedes s/n, 41012, Sevilla, Spain

[2]Karunya University
Coimbatore, Tamilnadu, India

[3]Université catholique de Louvain
Louvain-la-Neuve, Belgium

{sbdani,real}@us.es, hepzia@yahoo.com, easterselvans@gmail.com

Abstract. Membrane Computing is a biologically inspired computational model. Its devices are called P systems and they perform computations by applying a finite set of rules in a synchronous, maximally parallel way. In this paper, we follow a new research line using tissue-like P systems to do a parallel color segmentation of images using a thresholding to look for edge pixels. We have chosen this variant of P systems because it uses a less number of computational ingredients with respect to classical variants.

1 Introduction

Natural Computing studies new computational paradigms inspired from Nature. It abstracts the way in which Nature "computes", conceiving new computing models. There are several fields in Natural Computing that are now well established as are Genetic Algorithms ([1]), Neural Networks ([2]), DNA-based molecular computing ([3]).

Membrane Computing is a theoretical model of computation inspired by the structure and functioning of cells as living organisms able to process and generate information. The computational devices in Membrane Computing are called *P systems* [4]. Roughly speaking, a P system consists of a membrane structure, in the compartments of which one places multisets of objects which evolve according to given rules. In the most extended model, the rules are applied in a synchronous non-deterministic maximally parallel manner, but some other semantics are being explored.

According to their architecture, these models can be split into two sets: cell-like P systems and tissue-like P systems. In the first type of systems, membranes are hierarchically arranged in a tree-like structure. The inspiration for such architecture is the set of vesicles inside the cell. All of them perform their biological processes in parallel and life is the consequence of the harmonious conjunction of such processes. This paper is devoted to the second approach: tissue-like P systems.

Digital Image Processing ([**?**]) try to obtain a set of characteristics or parameters related with an image. Segmentation in computer vision (see [5]), refers to the process of partitioning a digital image into multiple segments (sets of pixels). The goal of segmentation is to simplify and/or change the representation of an image into something that is more meaningful and easier to analyze. Image segmentation is typically used to locate objects and boundaries (lines, curves, etc.) in images. More precisely, image segmentation is the process of assigning a label to every pixel in an image such that pixels with the same label share certain visual characteristics.

Recently, Membrane Computing techniques have been used for solving problems from Digital Image. Different P systems models have been used for dealing with images, as in [6] where cell-like P systems are used for computing the thresholding of 2D images; [7–10] where tissue-like P systems are used, or even [11], where the *symmetric dynamic programming stereo* (SDPS) algorithm [12] for stereo matching was implemented by using simple P modules with duplex channels. But in these cases, it is usual to do a pre-processing to the input images. So, they are transformed from color images in gray scale images or even binarized images.

We give a new researching step in this paper. We develop the first solution in Membrane Computing to the color segmentation problem. We show, from a theoretical point of view, it is possible to do a parallel segmentation of an image considering a general color alphabet. Here, the edges are not selected because a pixel has a different color than a neighbor pixel. We decide a pixel is an edge because it is enough different with a neighbor pixel.

The paper is structured as follows: in the next section we present the definition of array tissue-like P systems with input. In section 3, we design a family of systems for edge-based segmentation in 2D image ($n \times m$) using thresholding. Finally, some conclusions and future work are given in the last section.

2 Formal Framework

An *Array tissue-like P system* of degree $q \geq 1$ with input is a tuple of the form

$$\Pi = (\Gamma, V, \mathcal{E}, w_0, w_1, \ldots, w_q, A_1, \ldots, A_q, \mathcal{R}, i_\Pi, o_\Pi),$$

where

1. Γ is a finite *alphabet*, whose symbols will be called *objects*,
2. V is the *alphabet of colors* verifying $V \cap \Gamma = \emptyset$.
3. $\mathcal{E}$ is a finite subset of arrays on V.
4. $w_0, w_1, \ldots, w_q$ are strings over Γ representing the multisets of objects associated with the cells at the initial configuration,
5. $A_1, \ldots, A_n$ are arrays on V, placed on the corresponding cells at the initial configuration.

6. $\mathcal{R}$ is a finite set of communication rules of the following form: $(i, u_i W_i / u_j W_j, j)$, for $i, j \in \{0, 1, 2, \ldots, q\}, i \neq j$, $u_i, u_j \in \Gamma^*$ and W_i, W_j two arrays on V.
7. $i_\Pi \in \{0, 1, 2, \ldots, q\}$ is the input cell.
8. $o_\Pi \in \{0, 1, 2, \ldots, q\}$ is the output cell.

In a similar way to tissue-like P systems, an *array tissue-like P system* of degree $q \geq 1$ can be seen as a set of q cells (each one consisting of an elementary membrane) labeled by $1, 2, \ldots, q$. We will use 0 to refer to the label of the environment, i_Π and o_Π denote the input region and the output region (which can be the region inside a cell or the environment), respectively.

The strings $w_1, \ldots, w_q$ describe the multisets of objects placed in the q cells of the system. We interpret that w_0 is the set of objects placed in the environment, each one of them available in an arbitrary large amount of copies.

For each $i \in \{1, \ldots, q\}$, each A_i is an array placed in the cell i in the initial configuration and $\mathcal{E}$ is the set of arrays placed in the environment, each one of them available in an arbitrary large amount of copies. The empty array $\emptyset$ always belongs to $\mathcal{E}$. For all the non-empty copies, we will consider that the leftmost pixel of the bottom row in the array corresponds to the coordinates $(0, 0)$.

Rules are used as usual in the framework of membrane computing, that is, in a maximally parallel way (a universal clock is considered), regardless whether the environment is involved or not. In one step, each object in a membrane can only be used for one rule (non-deterministically chosen when there are several possibilities), but any object which can participate in a rule of any form must do it, i.e., in each step we apply a maximal set of rules.

3 Segmentation

We have segmented 2D digital images using array tissue-like P systems based in *edge-based segmentation* and thresholding. We consider an input 2D digital image, and an ordered color alphabet of the image. We define a family of P systems to do this task and show an overview of a computation of any system of the family. We do a complexity study for these systems. Finally, we see how one of our systems works with an example (see Fig. 1).

So, we look for pairs of adjacent pixels such that the distance between their colors is greater than a fixed threshold. We call these pixels *boundary pixels*. For each pair of border pixels, the smallest of them is called an *edge pixel*. But, we have a problem with this definition of edge pixels. The set of this type of pixel is not connected. So, we can consider a pixel adjacent to two edge pixels with near colors. If its color is near to the colors of the two edge pixels we say it is an edge pixel too.

Definition 1. The problem of segmentation of 2D digital images using thresholding (*2DEST problem*) is defined as follows:

Given a 2D digital image with pixels of (possibly) different colors establish the edge pixels (defined above).

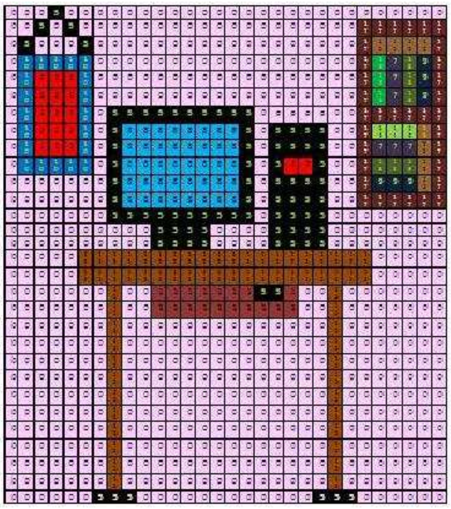

Fig. 1. An image of size 30 × 30

3.1 A family of array tissue-like P systems

Given a digital image with $n \times m$ pixels $(n, m \in \mathbb{N})$ we define an array tissue-like P system whose input is given by the pixels of the image encoded by the objects a_{ij}, where $1 \leq i \leq n$ and $1 \leq j \leq m$.

The key idea of how a system of this family works is the following. First, the system marks the boundary pixels. Next, it marks the necessary pixels to connect all the boundary pixels of the same color. Finally, the system uses a counter (z_i, whose number of initial copies in the system is $\lceil r_1^{1/2^7} \rceil$ and $r_1 = max(n, m)$) to send the marked objects to the environment. The output of the system is given by the objects that appear in the output cell when it stops.

We define a family of tissue-like P systems to do the edge-based segmentation with the thresholding of a 2D image. For each $n, m \in \mathbb{N}$, we consider the array tissue-like P system with input of degree 2:

$$\Pi = (\Gamma, \Sigma, V, \mathcal{E}, w_0, w_1, w_2, A_1, A_2, \mathcal{R}, i_\Pi, o_\Pi),$$

defined as follows

(a) $\Gamma = \Sigma \cup \{a'_{ij} \,:\, a \in \mathcal{C}_S,\ 1 \leq i \leq n,\ 1 \leq j \leq m\} \cup \{\bar{z}_1\}$,
(b) $\Sigma = \{a_{ij} \,:\, a \in \mathcal{C}_S,\ 1 \leq i \leq n,\ 1 \leq j \leq m\}$,
(c) $V = \mathbb{N}$
(d) $\mathcal{E} = \{\, a'\, b\,,\, b\, a'\,, \begin{array}{c} a' \\ b \end{array}, \begin{array}{c} b \\ a' \end{array}, \begin{array}{cc} a & b' \\ c' & d' \end{array}, \begin{array}{cc} a' & b \\ c' & d' \end{array},$

$$\begin{array}{cc} a' & b' \\ c & d' \end{array}, \begin{array}{cc} a' & b' \\ c' & d' \end{array}, \begin{array}{cc} a & b' \\ c' & d \end{array}, \begin{array}{cc} a' & b \\ c & d' \end{array} \;:\; a, b, c, d \in V\},$$

(e) $w_0 = \{\bar{z}_i \,:\, 2 \leq i \leq 9\}$,
$w_1 = \bar{z}_1^{\lceil r^{1/2^7} \rceil}$, where $r = max(n, m)$,
$w_2 = \bar{z}_1^{\lceil r^{1/2^7} \rceil}$,

(f) $A_1, A_2 = \emptyset$

(g) R is the following set of communication rules:

Type 1.
$$(j, \bar{z}_i / \bar{z}_{i+1}^2, 0), \text{ for } i = 1, \ldots, 8,\ j = 1, 2$$
In this rule, we are working with a counter that it is used in the output of the systems.

Type 2.
$$(1, a\ b\ /\ a'\ b\ , 0), \text{ for } a, b \in \mathcal{C}_S,\ a < b \text{ and } d(a, b) > k.$$
$$(1, b\ a\ /\ b\ a'\ , 0), \text{ for } a, b \in \mathcal{C}_S,\ a < b \text{ and } d(a, b) > k.$$
$$(1, \begin{matrix} a \\ b \end{matrix} / \begin{matrix} a' \\ b \end{matrix}, 0), \text{ for } a, b \in \mathcal{C}_S,\ a < b \text{ and } d(a, b) > k.$$
$$(1, \begin{matrix} b \\ a \end{matrix} / \begin{matrix} b \\ a' \end{matrix}, 0), \text{ for } a, b \in \mathcal{C}_S,\ a < b \text{ and } d(a, b) > k.$$
These rules are used when an image has two adjacent pixels whose associated colors have a distance greater than a threshold k (boundary pixels). Then, the pixel with the less associated color is marked (edge pixel).

Type 3.
$$(1, \begin{matrix} a & b' \\ c' & d \end{matrix} / \begin{matrix} a' & b' \\ c' & d \end{matrix}, 0), \text{ for } a, b, c, d \in \mathcal{C}_S,\ d(a, b') < k \text{ and } d(a, c') < k.$$
$$(1, \begin{matrix} a' & b \\ c & d' \end{matrix} / \begin{matrix} a' & b' \\ c & d' \end{matrix}, 0), \text{ for } a, b, c, d \in \mathcal{C}_S,\ d(b, a') < k \text{ and } d(b, d') < k.$$
$$(1, \begin{matrix} a & b' \\ c' & d \end{matrix} / \begin{matrix} a & b' \\ c' & d' \end{matrix}, 0), \text{ for } a, b, c, d \in \mathcal{C}_S,\ d(d, b') < k \text{ and } d(d, c') < k.$$
$$(1, \begin{matrix} a' & b \\ c & d' \end{matrix} / \begin{matrix} a' & b \\ c' & d' \end{matrix}, 0), \text{ for } a, b, c, d \in \mathcal{C}_S,\ d(c, a') < k \text{ and } d(c, d') < k.$$
The rules of this type are activated when a set of four pixels appears in cell 1 verifying that one no-marked pixel (a, in the first set of rules) is adjacent to two edge pixels (b', c', in the first set of rules) whose distance between them and from the first pixel to them are less than $k \in \mathbf{N}$ (our threshold). Finally, the fourth pixel (d, in the first set of rules) has to be adjacent to the previous edge pixels and distance from this last pixel to the first one is greater than k. When this pattern appears in cell 1 the first considered pixel (a, in the first set of rules) is marked and passed to be an edge pixel.

Type 4.
$$(1, \bar{z}_9 a' / \bar{z}_9, 2), \text{ for } a \in \mathcal{C}_S.$$
With these rules appears, the system sends the edge pixels to the output cell.

(h) $i_\Pi = 1$
(i) $o_\Pi = 2$.

An overview of the Computation: A 2D image is codified by the input array that appear in the input cell and the system begins to work with them. Rules of type 1 initiate the counter $\bar{z}$. In a parallel manner, rules of type 2 identify the boundary pixels and mark the edge pixels. These rules need 4 steps to do this. From the second step, the rules of type 3 can be used with the rules of the first type at the same time. So, in the other 4 steps we can mark the rest of the (edge) pixels adjacent to two edge pixels and the other boundary pixel with a color whose distance to the colors of the others is big enough. The system can apply the types of rules 2 and 3 simultaneously in some configurations, but it always applies the same number of these two types of rules because this number is given by the edge pixels (we consider 4-adjacency). Finally, the fourth type of rules are applied in the following step, when the system finished to mark all edge pixels in cell 1. So, with one step more the system sends the edge pixels to cell 2. Thus, we need only 9 steps to obtain an edge-based segmentation for an $n \times m$ digital image.

3.2 Complexity and necessary resources

Taking into account the size of the input data is $O(n \cdot m)$ and $|\mathcal{C}_S| = h$ is the number of colors of the image, the amount of necessary resources for defining the systems of our family and the complexity of our problem is determined in the following table:

2DEST Problem

Complexity	
Number of steps of a computation	9
Necessary Resources	
Size of the alphabet	$n \cdot m \cdot h$
Initial number of cells	2
Initial number of objects	$\lceil 2r_1^{(1/2^7)} \rceil$
Number of rules	$O(n \cdot m \cdot h^2)$
Upper bound for the length of the rules	8

3.3 Examples

In this section, we show the results obtained by the application of our method with three examples over a given image of size 30×30 (see Fig. 1). Some questions should be done when we begin to apply our algorithm. First, we need to choose a color space. But, this choice is hidden in the preprocessing of the systems, because we can choose a distance function.

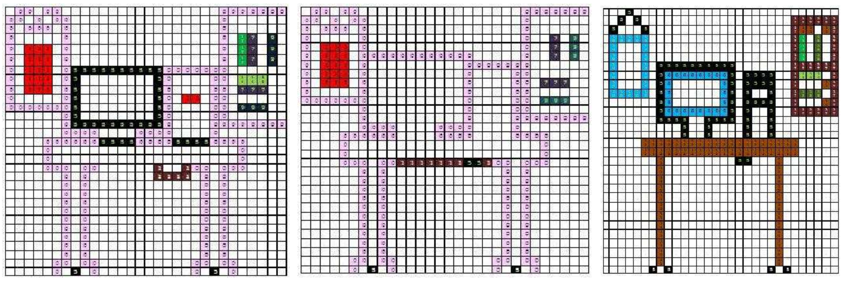

Fig. 2. (Left) Segmentation of the image of our example with thresholding $k = 3$ and considering the natural order in the colors alphabet. (Center) Segmentation of the image of our example with thresholding $k = 5$ and considering the natural order to the colors alphabet. (Right) Segmentation of the image of our example with thresholding $k = 3$ and considering the reverse order in the colors alphabet.

Once we get a distance, the second question is to know which is an appropriate *threshold*. Notice that, when we work with different thresholds we obtain different segmentations as Fig. 2 (Left) and Fig. 2 (Center) show. In the first image, the threshold is $k = 3$ and in the second one it is $k = 5$.

Finally, the las question is closely relate with the first one. We can consider an order in the *color alphabet*. The choice of the ordered color alphabet implies changes in the final results of our system. We take $k = 3$ and $\mathbb{N}$ as the color alphabet. We can consider two orders in this set. The natural order, $(\leq)$, or the reverse order, $(\leq^\star)$[1]. If we work with the first order we obtain the segmentation of Fig. 2 (Right), and if we work with the second order we obtain the segmentations of Fig. 2 (Left). Obviously, we can find a lot of differences between two segmentations.

4 Conclusions

We claim that Digital Imagery problems can be suitable for Natural Computing techniques in general and for Membrane Computing techniques in particular. Many of the problems in Digital Imagery share features that are very interesting for using these techniques.

We give, in this paper, a first theoretical step to do parallel color segmentation using P systems. Until now, when we use the membrane parallel techniques of segmentation we need to binarize the image previously (see [13]).

As a future work, we try to implement this algorithm using parallel programming, as CUDA. An interesting line of research could be to apply this method to areas as Biological and Medical Image. Even, we can translate this algorithm to be used with higher dimensions without a lot work.

[1] $\forall a, b \in \mathbb{N}$, $a \leq^\star b$ if and only if $a \geq b$.

Acknowledgement

The first author acknowledge the support of the project ”Computational Topology and Applied Mathematics” PAICYT research project FQM-296. The third author acknowledge the support of the project MTM2006-03722 of the Ministerio español de Educación y Ciencia and the project PO6-TIC-02268 of Excellence of Junta de Andalucía.

References

1. Holland, J.H.: Adaptation in natural and artificial systems. MIT Press, Cambridge, MA, USA (1992)
2. McCulloch, W.S., Pitts, W.: A logical calculus of the ideas immanent in nervous activity. Bulletin of Mathematical Biophysics **5** (1943) 115–133
3. Adleman, L.M.: Molecular computation of solutions to combinatorial problems. Science **266** (1994) 1021–1024
4. Păun, G.: Computing with membranes. Technical Report 208, Turku Centre for Computer Science, Turku, Finland (November 1998)
5. Shapiro, L.G., Stockman, G.C.: Computer Vision. Prentice Hall PTR, Upper Saddle River, NJ, USA (2001)
6. Christinal, H.A., Díaz-Pernil, D., Gutiérrez-Naranjo, M.A., Pérez-Jiménez, M.J.: Thresholding of 2D images with cell-like P systems. Romanian Journal of Information Science and Technology (ROMJIST) **13**(2) (2010) 131–140
7. Christinal, H.A., Díaz-Pernil, D., Real, P.: Segmentation in 2D and 3D image using tissue-like P system. In Bayro-Corrochano, E., Eklundh, J.O., eds.: Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications 14th Iberoamerican Conference on Pattern Recognition, CIARP 2009, Guadalajara, Jalisco, Mexico, November 15-18, 2009. Proceedings. Volume 5856 of Lecture Notes in Computer Science., Berlin Heidelberg, Springer (2009) 169–176
8. Christinal, H.A., Díaz-Pernil, D., Real, P.: Region-based segmentation of 2D and 3D images with tissue-like P systems. Pattern Recognition Letters **32**(16) (2011) 2206 – 2212 Advances in Theory and Applications of Pattern Recognition, Image Processing and Computer Vision.
9. Peña-Cantillana, F., Díaz-Pernil, D., Berciano, A., Gutiérrez-Naranjo, M.A.: A parallel implementation of the thresholding problem by using tissue-like P systems. In Real, P., Díaz-Pernil, D., Molina-Abril, H., Berciano, A., Kropatsch, W.G., eds.: CAIP (2). Volume 6855 of Lecture Notes in Computer Science., Springer (2011) 277–284
10. Peña-Cantillana, F., Díaz-Pernil, D., Christinal, H.A., Gutiérrez-Naranjo, M.A.: Implementation on CUDA of the smoothing problem with tissue-like P systems. International Journal of Natural Computing Research **2**(3) (2011) 25–34
11. Gimel’farb, G., Nicolescu, R., Ragavan, S.: P systems in stereo matching. In Real, P., Díaz-Pernil, D., Molina-Abril, H., Berciano, A., Kropatsch, W., eds.: Computer Analysis of Images and Patterns. Volume 6855 of Lecture Notes in Computer Science. Springer Berlin / Heidelberg (2011) 285–292
12. Gimel’farb, G.L.: Probabilistic regularisation and symmetry in binocular dynamic programming stereo. Pattern Recognition Letters **23**(4) (2002) 431–442
13. Sheeba, F., Thamburaj, R., Nagar, A.K., Mammen, J.J.: Segmentation of peripheral blood smear images using tissue-like p systems. International Conference on Bio-Inspired Computing: Theories and Applications, (2011) 257–261