Image Segmentation using Tissue-like P Systems with Multiple Auxiliary Cells

Raúl Reina-Molina, Javier Carnero, Daniel Díaz-Pernil

[1]Research Group on Computational Topology and Applied Mathematics
Department of Applied Mathematics I
University of Seville
javier@carnero.net, raureimol@alum.us.es, sbdani@us.es

Abstract. We present a solution of the segmentation problem using a distributed, non deterministic and parallel computational model known as tissue-like P systems. We present a new technique to segment images with respect to the algorithm appeared in [1], where we use multiple auxiliary cells and not only one.

1 Introduction

Membrane systems are distributed and parallel computing devices processing multisets of objects in compartments delimited by membranes. Computation is carried out by applying given rules to every membrane content, usually in a maximal non-deterministic way, although other semantics are being explored.

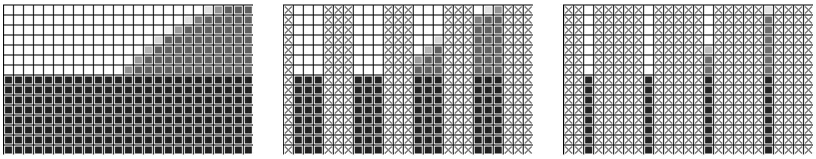
In our work, we present a family of tissue-like P systems (tlP systems) which solves the Segmentation Problem in Digital Imagery using multiple cells. Segmentation in computer vision (see [4]) refers to the process of partitioning a digital image into multiple segments (sets of pixels). Image segmentation is typically used to locate objects and boundaries (lines, curves, etc.). 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.

In the literature, one can find several attempts for bridging problems from Digital Imagery with Membrane Computing. We can cite the works by K.G. Subramanian *et al.* [**?**] or recently some problems from Digital Imagery have been solved in the framework of Membrane Computing (see [3]).

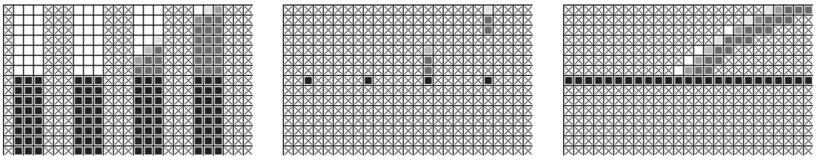
2 Image Processing: Segmentation Problem

The *m-D Segmentation Problem with k auxiliary cells (mDSP-kC)* can be settled as follows: given an m-D digital image, I of size n^m, to determine the edge pixels of this image using k auxiliary cells.

The usual definition of edge pixel presents problems from a practical point of view with the noise and the degradation of colours, because we take as border points a lot of pixels that are not edge pixels from a practical point of view.



(a) Initial content (b) Content of second cell (c) Content of second cell after cleaning stage



(d) Content of second cell before edge detection stage (e) Content of second cell after edge detection (f) Content of first cell once halting condition is reached

Fig. 1: Full segmentation process zoomed.

Next, we will show that *2DSP-kC* can be solved in linear time (in the number of pixels of the image) by a family of tlP systems (see Fig.1). To this aim, let us construct a family $\mathbf{\Pi} = \{\Pi(n,k) : n, k \in \mathbb{N}\}$ where each system of the family will process every instance u of the problem (a 2D image I with n^2 pixels) and using k auxiliary cells. More formally, we define the size of the instance as $s(u) = \langle n, k\rangle$, where $\langle x, y\rangle = (x+y)(x+y+1)/2 + x$ is the Gödel mapping. In order to provide a suitable encoding of this instances into the systems, we will use the objects $I(ij)_{ij}$, with $1 \leq i, j \leq n$, to represent the pixels of the graph, and we will provide $cod(u)$ as the initial multiset for the system, where $cod(u)$ is the multiset of objects $I(ij)'_{ij}$ for $1 \leq i, j \leq n$.

Then, given an instance u of the *2DSP-kC* problem, the system $\Pi(s(u))$ with input $cod(u)$ give a solution to this problem, implemented in the following stages:

- Cleaning noise.
- Homogenize colours using a general thresholding in colour space.
- Segmenting image process.

The family $\mathbf{\Pi} = \{\Pi(n,k) : n, k \in \mathbb{N}\}$ of tlP systems of degree $k+1$ is defined as follows: for each $n, k \in \mathbb{N}$,

$$\Pi(n,k) = (\Gamma, \Sigma, \mathcal{E}, w_1, \ldots, w_{k+1}, \mathcal{R}, i_\Pi, o_\Pi),$$

defined as follows:

- $\Gamma = \Sigma \cup \{a_{ij}a''_{ij}, \bar{a}_{ij}, A_{ij}, A'_{ij}, A''_{ij}, \bar{A}_{ij} : 1 \leq i, j \leq n,\ a \in \mathcal{C}\} \cup \{*_{ij}, *_{ji} : i = 0, n+1, 0 \leq j \leq n+1\}$, $\Sigma = \{a'_{ij} : 1 \leq i, j \leq n,\ a \in \mathcal{C}\}$, $\mathcal{E} = \Gamma - \Sigma$,

– $w_1 = *_{ij}, *_{ji}$ with $i = 0, n+1, 0 \le j \le n+1$, $w_2 = \cdots = w_{k+2} = T^{\lceil n^2/k \rceil}$,
– R is the following set of communication rules:
  - $(1, a'_{ij}/a^8_{ij} A_{ij}, 0)$ for $0 \le i, j \le n+1$ and $a \in \mathcal{C} \cup \{*\}$
  - $\left(\begin{array}{llll} & c_{i-1j-1} & d_{i-1j} & e_{i-1j+1} \\ 1, & b_{ij-1} & A_{ij} & f_{ij+1} \quad / \ T, t \\ & o_{i+1j-1} & h_{i+1j} & g_{i+1j+1} \end{array} \right)$
    for $1 \le i, j \le n$, $a, b, c, d, e, f, g, h, o \in \mathcal{C} \cup \{*\}$ and $2 \le t \le k+1$ indicating an auxiliary working cell.
    These rules are used to generate new elements. The P system uses these elements to work with the noise of our image.
  - $\left(\begin{array}{llll} & c_{i-1j-1} & d_{i-1j} & e_{i-1j+1} \\ t, & b_{ij-1} & A_{ij} & f_{ij+1} \quad / \ z'_{ij}, 0 \\ & o_{i+1j-1} & h_{i+1j} & g_{i+1j+1} \end{array} \right), \left(\begin{array}{llll} & c_{i-1j-1} & d_{i-1j} & e_{i-1j+1} \\ t, & b_{ij-1} & A_{ij} & f_{ij+1} \quad / \ a'_{ij}, 0 \\ & o_{i+1j-1} & h_{i+1j} & g_{i+1j+1} \end{array} \right)$
    for $1 \le i, j \le n$, $a, b, c, d, e, f, g, h, o \in \mathcal{C} \cup \{*\}$. We take μ as the number of pixels adyacents to the ij possition with colours in $\mathcal{C}$ and $* = 0$. Then, $av = (b + c + d + e + f + g + h + o)/\mu$ and $z = \max\{s \in \mathcal{C} : s \le av\}$ and $|a - av| \le \rho_1$, where $\rho_1 \in (0, +\infty)$.
    This set of rules is used to detect the noise and correct it with the average of colours of its adjacent pixels. We find here a local thresholding (with respect to the colours) with predefined threshold ρ_1. The use of $*$ grants a simple working for pixels in the border of I.
  - $(t, b'_{ij}/A'_{ij}, 0)$ for $1 \le i, j \le n$, $\nu = (|\mathcal{C}|/\rho_2)$, $l = 0, 1, 2, \ldots, \rho_2$. If $b \in \mathcal{C}$ then $a \in \mathcal{C}$ ($a < b \le a + (\nu - 1)$ and $a = \nu \cdot l$) or ($b = a = \nu \cdot l$) and, if $b = *$ then $A = *$.
    These rules are used to discretize the colours dividing the set of colours in ρ_2 subsets of length ν. We find here a general thresholding (with respect to the colours) with predefined threshold ν.
  - $(t, A'_{ij}/T, 1)$ for $a \in \mathcal{C}$, $0 \le i, j \le n+1$ and $2 \le t \le k+1$.
    This set of rules are used to send our transformed image to the cell 1. Now, the objects A'_{ij} codify the pixels of our image.
  - $(1, A'_{ij}/A''_{ij}\overline{a}^8_{ij}, 0)$ for $a \in \mathcal{C} \cup \{*\}$ and $0 \le i, j \le n+1$.
    The P system uses these rules to generate an number of copies of our image to do the segmentation process in the cells $2, \ldots, k$ and $k+1$.
  - $\left(\begin{array}{llll} & \overline{c}_{i-1j-1} & \overline{d}_{i-1j} & \overline{e}_{i-1j+1} \\ 0, & \overline{b}_{ij-1} & A''_{ij} & \overline{f}_{ij+1} \quad / \ T, t \\ & \overline{i}_{i+1j-1} & \overline{h}_{i+1j} & \overline{g}_{i+1j+1} \end{array} \right)$
    for $1 \le i, j \le n$ and $a, b, c, d, e, f, g, h, i \in \mathcal{C} \cup \{*\}$.
    These rules are defined to send to the remainder of cells the new objects.
  - $(t, A''_{ij}\overline{b}_{kl}/\overline{A}_{ij}, 0)$, for $1 \le i, j, k, l \le n$, $(i, j), (k, l)$ adjacent pixels, $a, b \in \mathcal{C}$ and $a < b$.
    These rules are used to mark edge pixels. When we have two adjacent pixels with different colour, we take as edge pixel the pixel with less associated colour. In fact, the P system brings from the environment an object A_{ij} (in this case).
  - $(t, \overline{A}_{ij}/T, 1)$ for $a \in \mathcal{C}$ and $1 \le i, j \le n$.
    These rules send to the cell 1 the output objects.

– $i_{\Pi} = o_{\Pi} = 1$.

A little study of the complexity aspects of this solution is given by the Fig.2 showing this is an efficient algorithm from a theoretical point of view.

mDSP-kC Problem	
Complexity	
Number of steps of computation	9
Resources needed	
Size of the alphabet	$8n^2 + 4n + 5$
Initial number of cells	$k + 1$
Initial number of objects	$(n + 2)^2$
Number of rules	$O(n^2 \cdot h^9 \cdot k)$
Upper bound for the length of the rules	10

Fig. 2: Complexity aspects, where the size of the input data is $O(n^2)$, $|\mathcal{C}| = h$ is the number of colours of the image and k is the number of working cells.

3 Final Remarks

We have designed a family of tissue-like P systems to do a segmentation of a digital image. With this algorithm we work with multiple cells, so we can obtain a bigger parallelization with respect to the design presented in [1].

Acknowledgement

DDP acknowledges the support of the projects TIN2008-04487-E and TIN-2009-13192 of the Ministerio de Ciencia e Innovación of Spain and the support of the Project of Excellence of the Junta de Andalucía, grant P08-TIC-04200.

References

1. Carnero, J., Díaz-Pernil, D., Molina-Abril, H., Real, P.: Image segmentation inspired by cellular models using hardware programming. Image-A 1(3), 143–150 (2010)
2. Ceterchi, R., Gramatovici, R., Jonoska, N., Subramanian, K.G.: Tissue-like P systems with active membranes for picture generation. Fundamenta Informaticae 56(4), 311–328 (2003)
3. Díaz-Pernil, D., Gutiérrez-Naranjo, M.A., Molina-Abril, H., Real, P.: Designing a new software tool for digital imagery based on P systems. *Natural Computing*, pages 1–6, 2011.
4. Shapiro, L.G., Stockman, G.C.: Computer Vision. Prentice Hall PTR, Upper Saddle River, NJ, USA (2001)