Cellular automata rule characterization and classification using texture descriptors

Jeaneth Machicao+, Lucas C. Ribas*, Leonardo F. S. Scabini+, Odermir M. Bruno+*

(*) Institute of Mathematics and Computer Science, University of São Paulo, São Carlos, SP, Brazil.

(+) Scientific Computing Group. São Carlos Institute of Physics, University of São Paulo, São Carlos - SP, PO Box 369, 13560-970, Brazil.

Abstract

The cellular automata (CA) spatio-temporal patterns have attracted the attention from many researchers since it can provide emergent behavior resulting from the dynamics of each individual cell. In this manuscript, we propose an approach of texture image analysis to characterize and classify CA rules. The proposed method converts the CA spatio-temporal patterns into a gray-scale image. The gray-scale is obtained by creating a binary number based on the 8-connected neighborhood of each dot of the CA spatio-temporal pattern. We demonstrate that this technique enhances the CA rule characterization and allow to use different texture image analysis algorithms. Thus, various texture descriptors were evaluated in a supervised training approach aiming to characterize the CA's global evolution. Our results show the efficiency of the proposed method for the classification of the elementary CA (ECAs), reaching a maximum of 99.57% of accuracy rate according to the Li-Packard scheme (6 classes) and 94.36% for the classification of the 88 rules scheme. Moreover, within the image analysis context, we found a better performance of the method by means of a transformation of the binary states to a gray-scale.

Keywords: elementary cellular automata, texture analysis

[*] Corresponding author
Email address: bruno@ifsc.usp.br (Odermir M. Bruno+)

1. Introduction

Currently, many CA researchers have put many efforts on the development of CA theoretical models and to analyze the behavior and training patterns revealed by certain models of CA. The evolution of CAs generate spatio-temporal patterns from prior knew rules that governs their local behavior. However, the problem of CA rule characterization and classification based on the observation of spatio-temporal patterns has received minor attention. In fact, few works have explored the classification of CA's rules. For instances, Wuensche [1] attempt to classify one-dimensional CA rules by means of an entropy-related measure over time. Kunkle [2] presents an algorithm based on neural networks to classify ECAs and one family of totalistic CAs. Kunkle's algorithm involves seven parameters, being activity, reverse determinism, sensitivity, absolute activity, neighborhood dominance, activity propagation, and incompressibility. In these former works, the complexity of the generated patterns was examined in the local time, i.e. not considering the CA global evolution. On the other hand, a more interesting approach was recently proposed by Da Silva et al. [3]. They classify ECAs and four families of totalistic CAs through a texture analysis approach that rely on the spatio-temporal patterns.

In this manuscript, we also use an image analysis approach under a supervised training strategy to tackle the CA classification problem. For this purpose, we explored directly the CA's spatio-temporal patterns. As CA consists basically of two parts: the cellular space and the transition rule, therefore we trace correspondence within an image analysis context. Thus, the spatio-temporal pattern is the image to be analyzed and the rule is the class that generates it. In a previous work, Da Silva et al. [3] used texture images to characterize the CA rules successfully using texture image analysis. In this work, we propose an enhancement of the CA rules characterization using texture analysis. Instead of using directly the binary states of the CA spatio-temporal, we transform it into a gray-scale image considering the neighborhood of each state. Furthermore, apart from the CA classification task, we include the rule characterization task, i.e. to characterize the spatio-temporal patterns regarding the rule that generated it.

For our experiments, we explored the spatio-temporal patterns generated by the well-known elementary CA (ECA), which consists of a regular array of cells N (pixels) compiled in time steps t to form an image of $N \times t$ pixels. Thus, we aim to classify ECA rule space within the Li-Packard classification scheme and to characterize all rules within the 88-class scheme. We compare accuracies rates through well-known texture descriptors, namely Local Binary Pattern (LBP), Fourier descriptors, Gabor descriptors and Gray Level Co-occurrence Matrix (GLCM), to extract the intrinsic attributes from these spatio-temporal diagrams evaluated through a supervised learning classifier algorithm.

2. Background

2.1. Cellular automata

Formally, a CA is represented by $\langle \mathcal{T}, S, s_0, N, \phi \rangle$, where $\mathcal{T}$ is the tessellation, S is a finite set of states or values for the cells, N is the finite cell neighborhood, s_0 is the initial configuration of the states of all the cells $c_i \in \mathcal{T}$, and ϕ the local transition rule [4]. The state of a cell (c_i) is determined by the evaluation of the transition rule on the cell's neighborhood at discrete time t. The transition rule can be totalistic or outer-totalistic, the former only depends on the participation of the neighboring states, while the latter depends also on the cell's state itself. Moreover, the initial configuration of the automaton is parametrized by ρ, which represents the probability of having "alive" cells ($c_i = 1$) at $t = 0$.

The well-known elementary CA is one of the most studied CAs, which is defined on a one-dimensional tessellation $\mathcal{T}$ and by its binary states $S = \{0, 1\}$. The values of the cells are updated synchronously, in discrete time steps t, according to a local rule, which is identical to all cells. The output state of each automaton depends on the state of its neighbors and the value of the cell itself within a radius $r = 1$ [5]. Moreover, some boundary conditions must be considered in order to be implemented computationally, for example, a periodic, reflective or fixed. The well-known elementary rules are given by a transition table, that lists each possible neighborhood, in binary code [5], regard to its output state. The elementary space is composed of 256 rules, due to the size of the neighborhood is $2r + 1$, leading to 2^{2^3} combinations. Furthermore, the spatial-temporal patterns formed by CAs can be visualized as a two-dimensional image. Particularly ECA is formed by compiling the evolution of the cells over time t, thus representing an image of $N \times t$ cells.

2.2. Rule space classifications based on dynamic behavior

More than one rule space scheme has been proposed in the literature to classify CAs based on their dynamics and behavior. For instances, Wolfram [5] classified the 256 elementary rules into four qualitative classes (Type 1, Type 2, Type 3 and Type 4) based on the dynamic behavior exhibited by the spatio-temporal patterns observed on the average of many initial conditions. Later on, Li & Packard [6] proposed a modified version of Wolfram's classification, by refining specifically Wolfram's Type 2, thus dividing the rule space into six classes (null, fixed-point, two-cycle, periodic, complex and chaotic). Should be noted that, among the 256 elementary rules, some rules are equivalent to other ones, due to the symmetry properties of the state transitions [7]. For example, the behavior is equivalent to each other through a black-white reflection, negation, or both. Thus according to this specificity, e.g. rule 30 is equivalent to rules 86, 135 and 149. Therefore, the elementary rule space is reduced to 88 equivalent rules. Table 1 shows the rule equivalences and the corresponding Li-Packard classification scheme.

3. Texture Analysis

The goal of texture analysis is to characterize spatial patterns of intensity variation in a given image. This is a classical image analysis approach, and many texture analysis methods were proposed along the years [13]. Four well-known texture descriptors that are applied to this work are described as follows.

3.1. Local Binary Pattern (LBP)

Local Binary Pattern (LBP) is a simple and efficient method, widely used in several image analysis problems [9]. Given an image I, the method capture spatial patterns of intensity changes to each pixel by analyzing its 8-connected neighborhood. Thus,

Table 1: Elementary rule and their equivalent rules grouped by the six classes of Li-Packard scheme (Ref. [8]).

Rule	Equiv. rule	Dynamics	Rule	Equiv. rule	Dynamics	Rule	Equiv. rule	Dynamics
0	255	Null	35	49,59,115	Two-Cycle	108	201	Two-Cycle
1	127	Two-Cycle	36	219	Fixed-Point	110	124,137,193	Complex
2	16,191,247	Fixed-Point	37	91	Two-Cycle	122	161	Chaotic
3	17,63,119	Two-Cycle	38	52,155,211	Two-Cycle	126	129	Chaotic
4	223	Fixed-Point	40	96,235,249	Null	128	254	Null
5	95	Two-Cycle	41	97,107,121	Periodic	130	144,190,246	Fixed-Point
6	20,159,215	Two-Cycle	42	112,171,241	Fixed-Point	132	222	Fixed-Point
7	21,31,87	Two-Cycle	43	113	Two-Cycle	134	148,158,214	Two-Cycle
8	64,239,253	Null	44	100,203,217	Fixed-Point	136	192,238,252	Null
9	65,111,125	Two-Cycle	45	75,89,101	Chaotic	138	174,208,244	Fixed-Point
10	80,175,245	Fixed-Point	46	116,139,209	Fixed-Point	140	196,206,220	Fixed-Point
11	47,81,117	Two-Cycle	50	179	Two-Cycle	142	212	Two-Cycle
12	68,207,221	Fixed-Point	51		Two-Cycle	146	182	Chaotic
13	69,79,93	Fixed-Point	54	147	Complex	150		Chaotic
14	84,143,213	Two-Cycle	56	98,185,227	Fixed-Point	152	188,194,230	Fixed-Point
15	85	Two-Cycle	57	99	Fixed-Point	154	166,180,210	Periodic
18	183	Chaotic	58	114,163,177	Fixed-Point	156	198	Two-Cycle
19	55	Two-Cycle	60	102,153,195	Chaotic	160	250	Null
22	151	Chaotic	62	118,131,145	Periodic	162	176,186,242	Fixed-Point
23		Two-Cycle	72	237	Fixed-Point	164	218	Fixed-Point
24	66,189,231	Fixed-Point	73	109	Chaotic	168	224,234,248	Null
25	61,67,103	Two-Cycle	74	88,173,229	Two-Cycle	170	240	Fixed-Point
26	82,167,181	Periodic	76	205	Fixed-Point	172	202,216,228	Fixed-Point
27	39,53,83	Two-Cycle	77		Fixed-Point	178		Two-Cycle
28	70,157,199	Two-Cycle	78	92,141,197	Fixed-Point	184	226	Fixed-Point
29	71	Two-Cycle	90	165	Chaotic	200	236	Fixed-Point
30	86,135,149	Chaotic	94	133	Periodic	204		Fixed-Point
32	251	Null	104	233	Fixed-Point	232		Fixed-Point
33	123	Two-Cycle	105		Chaotic			
34	48,187,243	Fixed-Point	106	120,169,225	Chaotic			

to each central c pixel located at a given position (x_c, y_c) with intensity $g_c = I(x_c, y_c)$ and a neighbor p, a binary value is obtained:

$$LBP(g_c, g_p) = s(g_p - g_c), s(x) = \begin{cases} 1, x \geq 0 \\ 0, x < 0 \end{cases} \tag{1}$$

where g_p represent the gray level of p. This process is repeated to the 8 equally-spaced neighbors of c, according to the 8-connected neighborhood. This results in a 8-bit binary sequence which describes the pattern of intensity changes around each pixel of the image. This binary sequence is then encoded in a decimal number k, where the maximum k value is 2^8 = 256. The occurrence of each k on the transformed image is computed with an histogram H:

$$H(k) = \sum_{i=1}^{M} \sum_{j=1}^{N} q(LBP_{P,r}(i,j), k) \tag{2}$$

with $k \in [0, 256]$, q$(x, y) = 1$ if $x = y$ or q$(x, y) = 0$ otherwise, and M and N are the sizes of the image I. This histogram represents a global statistic of local intensity change patterns captured by LBP, and is employed as the image descriptor.

3.2. Fourier Descriptors

Fourier transform changes the domain of a signal from spatial to frequency [10]. The 2-Dimensional Fourier transform can be applied to obtain texture descriptors. The feature extraction consists of analyzing sections of the image spectrum. Consider x and y as the positions in the spatial domain. The 2D Fourier transform of an image $|I| = NxM$ defined by the function $f(x, y)$ is:

$$F(u, v) = \frac{1}{NM} \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} f(x, y) \mathrm{e}^{j2\pi(ux/N+vy/M)} \tag{3}$$

Then we multiply the pixel values by $(-1)^{x+y}$ considering circular regions, and the frequency (0,0) will appear in the middle of its spectrum. For characterization, the sections of the spectrum can be analyzed using two types of characteristics: the energy E_a contained in concentric annuli and the energy E_b contained in circular regions:

$$E_{a_{ij}} = \int_{\lambda_i}^{\lambda_{i+1}} \int_{\theta_j}^{\theta_{j+1}} |F(\lambda, \theta)|^2 \, d\theta d\lambda \tag{4}$$

$$E_{b_i} = \int_{\lambda 0}^{\lambda_i} \int_{0}^{2\pi} |F(\lambda, \theta)|^2 d\theta d\lambda \tag{5}$$

where i and j are the parameters of the descriptor related to the number of orientations and sections used (7 and 4, respectively). For a best region analysis, the polar coordinates (λ, θ) are used in the frequency domain. Finally, the image descriptor consists of the concatenation of the computed energies.

3.3. Gabor Descriptor

In opposition the Fourier transform, the Gabor filters use both spatial and frequency domain to feature extraction of the image texture. In this method, filters are constructed by a sinusoidal wave localized by a Gaussian kernel. The real part of Gabor filters is produced by the following equation:

$$g\big(x, y; \lambda, \theta, \psi, \sigma, \gamma\big) = \exp\Big(-\frac{(x')^2 + \gamma\,(y')^2}{2\sigma^2}\Big)\cos(2\pi\frac{x'}{\lambda} + \psi) \qquad (6)$$

where $x' = x\cos\theta + y\sin\theta, , y' = -x\sin\theta + y\cos\theta$, λ is the wavelength of the sinusoidal, θ is the orientation of the filter, ψ denotes the phase offset, σ is the standard deviation of the Gaussian kernel, and γ is its spatial aspect ratio.

Different filters can be obtained changing the parameters of the Gabor function. Thus, the texture image is convolved with a bank of filters built with different parameters, resulting in a set of filtered images c_i. On this work, we apply a filter bank composed of 8 scales (3x3, 5x5, ..., 17x17) and 5 orientations ($\theta = \{0°, 36°, 72°, 108°, 144°\}$). For feature extraction, four measurements are extracted from each image c_i: contrast, correlation, energy, and homogeneity. The final image descriptor is the concatenation of measures extracted from each image c_i.

3.4. Gray Level Co-occurrence Matrix

Gray Level Co-occurrence Matrix (GLCM) [11] is a well-known statistical texture analysis method that counts the joint occurrence of intensity values. An image is represented by a co-occurrence matrix that is calculated using two parameters, d and θ. These parameters are defined based on the distance and angle of the pixels analyzed. Consider L the maximum gray level of the image. A matrix of size $L \times L$ is built where the position (i, j) contains the frequency of intensities i and j that occurs in the image with a certain distance and angle. To address that, two position operators are employed,

Δx and Δy to indicate where to check on the image to count the co-occurrence of pixel intensities. For instance, $(\Delta x, \Delta y) = (2, 1)$ indicates ”two down, one right”. Considering an image $I(x, y)$ and a given pair of position operators, the GLCM matrix of I is defined by:

$$\text{GLCM}_{\Delta x,\Delta y}(i,j) = \sum_{p=1}^{n}\sum_{q=1}^{m} \begin{cases} 1, \text{ if } I(p,q) = i \text{ and } I(p+\Delta x, q+\Delta y) = j \\ 0, \text{ otherwise} \end{cases} \tag{7}$$

Different statistical measures can then be obtained from the co-occurrence matrix $GLCM_{\Delta x,\Delta y}$ as image texture descriptors. The most common measures are contrast, correlation, energy, and homogeneity. A more robust representation is done by combining different values to Δx and Δy, resulting in multiple co-occurrence matrices. In this work, the following position operators are used: $(0, 1)$, $(-1, 1)$, $(-1, 0)$, $(-1, -1)$, $(0, 2)$, $(-2, 2)$, $(-2, 0)$ and $(-2, -2)$. Also, the number of gray levels L is reduced from 256 to 64 to minimize computations cost [11]. Intuitively, the final texture descriptor is the combination of statistical measures extracted from each GLCM.

4. Proposed method

In this manuscript, we propose a texture image analysis approach to characterize the CA’s spatio-temporal patterns for both purposes: i) automatic classification of CAs into the Li-Packard classification scheme and ii) to characterize the CA’s rules independently. More specifically, we focused on the family of rules of the elementary CA. We then propose a technique to transform an original binary CA into a gray-scale image by considering its spatial binary patterns. Each cell of the resulting $N \times N$ binary CA is converted into a value between 0 and 255 (gray-scale space) by considering its 8-connected neighborhood. These neighbors then compose a binary chain of size 8, which can be converted into a decimal value. Each step of the proposed algorithm is illustrated in Figure 1. As we consider the cell neighborhood to compute its gray-scale value, the method performs a slight smoothing of borders on the image. The result is a more consistent and realistic pattern of texture with longer gradients if compared to the raw binary image. Intuitively, traditional texture descriptors should perform better with the gray-scale conversion of the CA image.

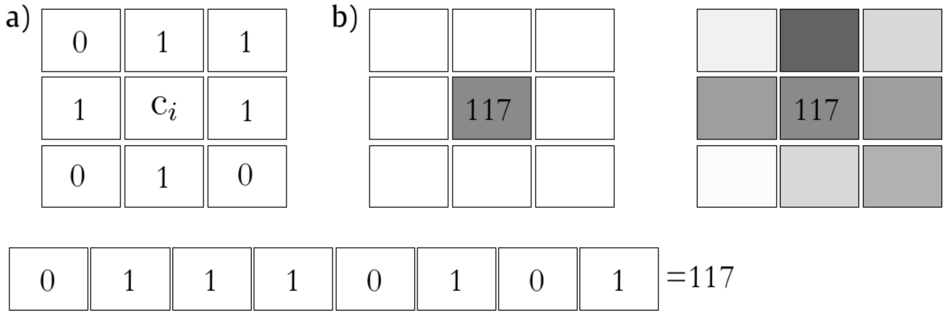
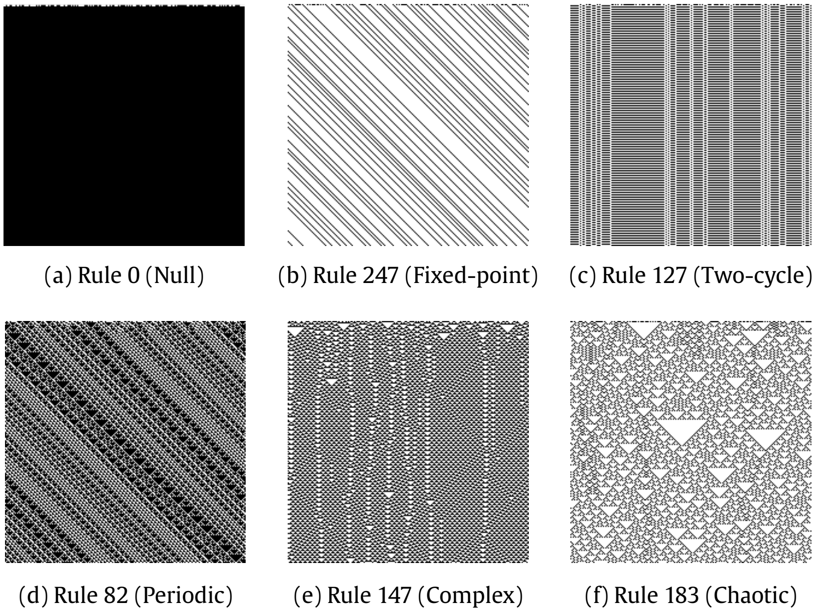


Figure 1: a) Transformation of a binary cell (center) to gray-scale by considering the binary states from the 8-connected neighborhood, following a clockwise direction, starting from top-left cell b) to then be codified as a decimal value. The resultant CA image into gray-scale considers periodic boundary.

Considering a resulting gray-scale CA image, our approach relies on different texture descriptor algorithms to characterize the CA visual patterns. We tested the following traditional texture descriptors: Local Binary Patterns (LBP), Fourier spectrum, Gabor filters, and Gray Level Co-occurrence Matrix (GLCM). These algorithms can also be applied on the binary CA, as was done on a previous work [3]. However, we show in our experiments that using the binary image is not the best approach to characterize the CA visual patterns.

4.1. CA dataset

The image dataset was constructed by evolving the ECAs using the 88 rules considered in this study (see Table 1). Here, we considered spatio-temporal images with different configurations of the number of pixels $N \times N$ with N between 200, 250 and 300 pixels evolved for N time steps, respectively. In all these cases, the cells started with random initial states $\rho = \{0.1, 0.3, 0.5, 0.7, 0.9\}$ generated by a uniform distribution. Then, we constructed 50 images per class for training with each combination of the size N and the initial configuration ρ per rule. Following this idea, we have 4500 and 66000 images for the Li-Packard and the 88 class scheme, respectively. Some samples of the generated dataset are shown on Figure 2. The test set is obtained in the same way according to each experiment. Moreover, we also generate the gray-scale correspondent of each binary image according to the proposed method (described in Fig. 1), resulting in a separated equal-sized dataset for the gray-scale experiments. A sample of a converted image from binary to gray-scale is shown on Figure 3. As the initial CAs cells are randomly generated according to ρ, we generate 10 different train and test sets.

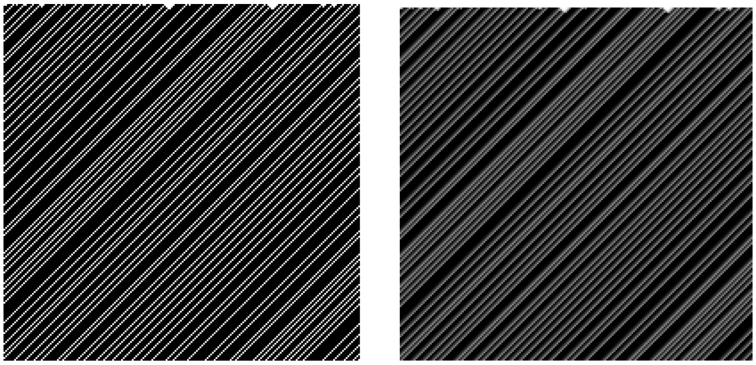

(a) Rule 0 (Null) (b) Rule 247 (Fixed-point) (c) Rule 127 (Two-cycle)

(d) Rule 82 (Periodic) (e) Rule 147 (Complex) (f) Rule 183 (Chaotic)

Figure 2: Sample of spatio-temporal image ECA samples according to the Li-Packard scheme.

4.2. Experimental Setup

We perform classification experiments with four well-known texture descriptors over the two CAs datasets previously described. On the first dataset, images are labeled according to the Li-Packard scheme, and the second dataset follows the 88-class scheme. The classification is performed using, separately, the original binary CA image or its gray-scale conversion using the proposed method. As previously mentioned concerning the randomness on the CAs construction, 10 different train and test sets are built for each experiment.



(a) Original ECA rule 130 and $p = 0.5$

(b) Gray-scale conversion of (a)

Figure 3: An original binary image of a CA (a) and its corresponding gray-scale conversion using the proposed method.

A SVM (*Support Vector Machine*) [12] classifier is considered with the following parameters: The penalty parameter (c) of the error term with $c = 1$ and an RBF (*Radial Basis Function*) kernel with $\gamma = 0.01$. In all experiments, 50 samples per class are used for both training and test steps. Results are measured by the accuracy rate of the classifier, which is the fraction of CAs from the test set correctly classified according to the correspondent class scheme (Li-Packard or 88-class). As we use 10 different train and test sets for each experiment, the mean and the standard deviation of the accuracy rate are considered.

5. Results and discussion

The results for the classification of ECAs within the Li-Packard (6 class) and the 88-class schemes are shown as follows.

5.1. Classification of ECA rule as Li-Packard scheme

We performed an experiment for the ECA classification within the Li-Packard scheme considering different probabilities of the CA initial configuration (ρ) and the size of the CA's tessellation, i.e. the spatio-temporal patterns ($N \times N$). Table 2 shows a comparison of the accuracy rates between the four texture descriptors studied here.

Within these parameters, the highest accuracy rates are obtained with the LBP descriptor, reaching 98.53% and 99.57% for the binary and gray-scale images, respectively. The overall performance considering the binary images and LBP is 95.16% (± 3.45), and the proposed conversion to gray-scale improved it to 97.49%(± 2.26). The proposed method also improved the overall performance of Gabor and GLCM descriptors. On the other hand, the influence of the gray-scale conversion on the overall performance of Fourier descriptors is a small reduction of 0.49%. We also observed a small performance fluctuation regarding the size of the automaton (N), which is a good indicator of the efficiency of the proposed method. Finally, we also observed that the values of the initial condition $\rho = 0.5$ provide the highest accuracies in all the cases. This variation is expected because ρ values around the extremes (≈ 0 or ≈ 1) tend to produce CAs with dynamics that escape its expected behavior, i.e. the occurrence of not expected patterns. On the other hand, CAs built with $\rho \approx 0.5$ are more likely to reproduce spatio-temporal patterns related to the classes in which the rules were associated.

5.2. Characterization of ECA's rule into 88 class scheme

For the characterization of the 88 ECA rules, we performed the same experiment as shown previously. Here we considered different initial configuration (ρ) and different values for the size of the CA's tessellation (N). Table 3 shows a comparison of the accuracy rates between the four texture descriptors studied here for the classification of the 88 ECA rules. In terms of classification, this is a harder problem because the CAs rules are divided into more classes, which is more likely to result in similar CAs from different classes. The highest accuracies obtained under these two parameters (ρ and N) yielded 92.28% and 94.36%, for the binary images and gray-scale images respectively, using the LBP descriptor. The gray-scale conversion worked in the same way as in the Li-Packard experiment, improving the overall performance of LBP, Gabor and GLCM. In contrast to the classification of the six classes of Li-Packard scheme shown in Section 5.1, we observed a significant influence of ρ on the accuracy rate of all descriptors, mainly around extreme values (≈ 0 or ≈ 1). Here we also observe that the highest accuracy rates appear with $\rho = 0.7$, instead of $\rho = 0.5$ as observed in the

Table 2: Accuracy rate (%) of texture descriptors on binary and gray-scale CA images for the Li-Packard scheme of class experiment by varying the ECA initial configuration (ρ) and the number of pixels (N).

		LBP		**Fourier**		**Gabor**		**GLCM**	
config.		bin	gray	bin	gray	bin	gray	bin	gray
N=200	ρ=0.1	91.57($\pm$ 0.83)	95.00($\pm$ 0.97)	84.57($\pm$ 1.54)	83.70($\pm$ 0.97)	84.00($\pm$ 0.94)	86.77($\pm$ 0.69)	87.33($\pm$ 0.68)	87.20($\pm$ 0.85)
	ρ=0.3	98.13($\pm$ 0.39)	98.93($\pm$ 0.44)	85.37($\pm$ 1.21)	86.43($\pm$ 1.51)	87.97($\pm$ 1.05)	89.30($\pm$ 0.81)	88.67($\pm$ 1.12)	92.70($\pm$ 1.08)
	ρ=0.5	98.07($\pm$ 0.62)	99.20($\pm$ 0.42)	85.80($\pm$ 1.06)	85.47($\pm$ 1.31)	87.73($\pm$ 1.11)	89.90($\pm$ 0.80)	88.63($\pm$ 1.43)	94.53($\pm$ 1.02)
	ρ=0.7	97.13($\pm$ 0.57)	98.83($\pm$ 0.67)	81.50($\pm$ 1.57)	81.40($\pm$ 1.76)	86.43($\pm$ 1.74)	88.57($\pm$ 0.90)	89.13($\pm$ 1.06)	93.33($\pm$ 0.98)
	ρ=0.9	90.27($\pm$ 1.53)	94.10($\pm$ 0.59)	75.40($\pm$ 1.36)	75.10($\pm$ 1.79)	77.63($\pm$ 1.35)	82.73($\pm$ 2.40)	83.03($\pm$ 1.97)	85.60($\pm$ 1.47)
N=250	ρ=0.1	91.67($\pm$ 0.99)	95.1($\pm$ 0.47)	83.7($\pm$ 1.35)	82.37($\pm$ 1.2)	84.3($\pm$ 0.85)	86.57($\pm$ 0.9)	87.5($\pm$ 1.1)	87.5($\pm$ 0.61)
	ρ=0.3	98.33($\pm$ 0.22)	99.33($\pm$ 0.35)	84.33($\pm$ 1.01)	83.77($\pm$ 1.69)	87($\pm$ 0.7)	89.33($\pm$ 1.07)	88.57($\pm$ 1.14)	92.3($\pm$ 0.46)
	ρ=0.5	**98.53**($\pm$ 0.39)	**99.57**($\pm$ 0.39)	82.57($\pm$ 1.92)	81.63($\pm$ 2.26)	88.4($\pm$ 1.08)	89.5($\pm$ 0.72)	88.5($\pm$ 1.03)	94($\pm$ 0.92)
	ρ=0.7	96.77($\pm$ 0.32)	99.23($\pm$ 0.42)	80.4($\pm$ 1.68)	79.4($\pm$ 1.94)	87.47($\pm$ 1.58)	88.47($\pm$ 0.55)	87.73($\pm$ 1.3)	93.9($\pm$ 0.96)
	ρ=0.9	90.5($\pm$ 0.93)	94.4($\pm$ 0.52)	74.1($\pm$ 1.66)	73.9($\pm$ 1.44)	77.37($\pm$ 0.92)	82.93($\pm$ 1.71)	84.9($\pm$ 2.23)	86.77($\pm$ 0.94)
N=300	ρ=0.1	91.77($\pm$ 0.97)	95.43($\pm$ 0.59)	84.13($\pm$ 1.44)	83.67($\pm$ 0.9)	84.67($\pm$ 0.83)	86.83($\pm$ 0.55)	87.83($\pm$ 0.81)	87.97($\pm$ 0.79)
	ρ=0.3	98.33($\pm$ 0.52)	99.5($\pm$ 0.28)	85.73($\pm$ 1.65)	85.1($\pm$ 1.5)	87.17($\pm$ 0.97)	89.23($\pm$ 0.7)	88.8($\pm$ 1.55)	92.7($\pm$ 0.76)
	ρ=0.5	98.4($\pm$ 0.52)	99.43($\pm$ 0.27)	85.93($\pm$ 1.19)	85.43($\pm$ 1.22)	87.6($\pm$ 0.75)	89.47($\pm$ 0.89)	88.97($\pm$ 0.69)	94.93($\pm$ 1.22)
	ρ=0.7	96.77($\pm$ 0.45)	99.23($\pm$ 0.22)	82.27($\pm$ 1.68)	81.73($\pm$ 1.83)	86.23($\pm$ 1.98)	88($\pm$ 1.4)	88.77($\pm$ 0.97)	94.1($\pm$ 1.1)
	ρ=0.9	91.17($\pm$ 1.21)	95($\pm$ 0.35)	75.1($\pm$ 0.92)	74.47($\pm$ 1.49)	77.5($\pm$ 1.34)	82.9($\pm$ 2.85)	85.9($\pm$ 1.49)	87.97($\pm$ 1)
mean acc.		95.16 ($\pm$ 3.45)	**97.49** ($\pm$ 2.26)	82.06 ($\pm$4.07)	81.57 ($\pm$ 4.11)	84.76 ($\pm$ 3.99)	87.37 ($\pm$ 2.56)	87.62 ($\pm$ 1.74)	91.03 ($\pm$ 3.38)

experiment within the Li-Packard scheme. It suggests that for the 88-class scheme, the spatio-temporal pattern tend to be more discriminative when the number of alive cells increase on the initial condition ($0.5 < \rho < 0.9$ with maximum ≈ 0.7).

The results showed that the LBP descriptor succeeded to classify in both scenarios, i.e. the set of CA rules for the Li-Packard and the characterization of rules. We assume that the LBP method outperformed the others because it encodes the occurrence of local patterns recursively, while its histogram distribution aims to characterize the CA global evolution. Moreover, the four texture descriptors (LBP, Fourier, Gabor and GLCM) shown considerably high tolerance to the size of the tessellation (N).

Although a formal comparison is not from the scope of this work, we can trace some comparisons between the highest accuracy obtained with our proposed method and the accuracies obtained by Kunkle's method [2] and Da Silva et al. work [3]. Regarding the former case, the LBP descriptor using the gray-scale proposal has provided a classification accuracy of 99.57% compared to Kunkle's accuracy of 98.3%, for the classification of ECA within the Li-Packard classification scheme. Regarding the latter case, besides Da Silva et al. work [3] classified within Wolfram's scheme, the highest accuracy achieved was 98.85% using Fourier descriptor and similar ECA configurations.

6. Conclusions

In this manuscript, we explored a texture image analysis approach to classify CA's rules based on its spatio-temporal patterns. We propose a technique to convert original binary CAs into gray-scale images by encoding the binary chain of an 8-connected neighborhood of cells into a decimal value between $[0, 255]$. We then perform a set of experiments applied to the well-known elementary CA. Thus, we explored two scenarios: i) the classification of ECA rules within the six classes of Li-Packard scheme (null, fixed point, two-cycle, periodic, complex, and chaotic) and ii) the classification of the ECA rules independently into an 88-class scheme.

Within the Li-Packard scheme, we obtained the highest accuracy rate of 99.57% using the LBP descriptor. We compare the results between the original binary CA

Table 3: Accuracy rate (%) of texture descriptors on binary and gray-scale images for the 88-class scheme of class experiment by varying the ECA initial configuration (ρ) and the number of pixels (N).

		LBP		**Fourier**		**Gabor**		**GLCM**	
config.		bin	gray	bin	gray	bin	gray	bin	gray
N=200	$\rho = 0.1$	61.47($\pm$ 0.43)	61.66($\pm$ 0.34)	53.13($\pm$ 0.9)	53.46($\pm$ 0.89)	58.23($\pm$ 0.75)	58.73($\pm$ 1.03)	58.36($\pm$ 0.46)	58.11($\pm$ 0.69)
	$\rho = 0.3$	78.46($\pm$ 0.59)	79.53($\pm$ 0.74)	70.66($\pm$ 0.47)	68.97($\pm$ 0.56)	75.02($\pm$ 0.45)	75($\pm$ 0.53)	76.94($\pm$ 0.58)	77.47($\pm$ 0.38)
	$\rho = 0.5$	87.66($\pm$ 1.03)	90.19($\pm$ 0.46)	79.9($\pm$ 0.79)	80($\pm$ 0.52)	84.14($\pm$ 0.48)	84.31($\pm$ 0.5)	83.73($\pm$ 0.48)	86.19($\pm$ 0.41)
	$\rho = 0.7$	91.1($\pm$ 0.58)	93.61($\pm$ 0.43)	84.61($\pm$ 0.64)	83.75($\pm$ 0.45)	88.75($\pm$ 0.58)	90.25($\pm$ 0.55)	86.92($\pm$ 0.44)	88.33($\pm$ 0.58)
	$\rho = 0.9$	82.58($\pm$ 0.99)	83.32($\pm$ 0.73)	79.45($\pm$ 0.61)	79.45($\pm$ 0.96)	82.45($\pm$ 0.99)	83.01($\pm$ 0.86)	79.69($\pm$ 0.67)	78.89($\pm$ 0.81)
N=250	$\rho = 0.1$	62.38($\pm$ 0.49)	62.33($\pm$ 0.64)	52.55($\pm$ 1.31)	53.88($\pm$ 0.75)	58.63($\pm$ 0.77)	59.92($\pm$ 0.58)	58.9($\pm$1.04)	59.23($\pm$ 0.83)
	$\rho = 0.3$	79.34($\pm$ 0.83)	81.27($\pm$ 0.51)	68.83($\pm$ 1.09)	67.38($\pm$ 0.94)	74.48($\pm$ 0.76)	75.37($\pm$ 0.58)	78.38($\pm$ 0.7)	78.94($\pm$ 0.37)
	$\rho = 0.5$	88.52($\pm$ 0.6)	90.63($\pm$ 0.56)	77.88($\pm$ 0.58)	76.46($\pm$ 0.3)	83.33($\pm$ 0.76)	84.1($\pm$ 0.96)	84.25($\pm$ 0.71)	86.94($\pm$ 0.77)
	$\rho = 0.7$	92.03($\pm$ 0.67)	93.88($\pm$ 0.65)	82.09($\pm$ 0.85)	81.31($\pm$ 0.77)	88.82($\pm$ 0.75)	90.63($\pm$ 0.42)	87.3($\pm$ 0.4)	88.84($\pm$ 0.81)
	$\rho = 0.9$	83.96($\pm$ 0.58)	83.8($\pm$ 0.74)	78.56($\pm$ 1.05)	78.47($\pm$ 0.98)	82.67($\pm$ 1.3)	83.2($\pm$ 0.99)	80.08($\pm$ 0.94)	80.03($\pm$ 0.83)
N=300	$\rho = 0.1$	63.58($\pm$ 0.49)	63.01($\pm$ 0.48)	53.57($\pm$ 0.64)	54.48($\pm$ 0.54)	59.11($\pm$ 0.79)	59.74($\pm$ 1.21)	60.39($\pm$ 0.92)	60.14($\pm$ 1.15)
	$\rho = 0.3$	80.96($\pm$ 0.56)	82.3($\pm$ 0.68)	70.22($\pm$ 0.84)	68.8($\pm$ 0.59)	76.03($\pm$ 0.91)	76.16($\pm$ 0.43)	80.8($\pm$ 0.57)	80.9($\pm$ 0.66)
	$\rho = 0.5$	89.44($\pm$ 0.6)	91.35($\pm$ 0.56)	80.42($\pm$ 0.6)	79.07($\pm$ 0.57)	85.28($\pm$ 0.7)	85.44($\pm$ 0.84)	85.2($\pm$ 0.42)	88.32($\pm$ 0.74)
	$\rho = 0.7$	**92.28**($\pm$ 0.6)	**94.36**($\pm$ 0.41)	82.83($\pm$ 0.4)	82.19($\pm$ 0.63)	89.35($\pm$ 0.42)	91.29($\pm$ 0.32)	88.4($\pm$ 0.53)	90.12($\pm$ 0.87)
	$\rho = 0.9$	84.56($\pm$ 0.45)	84.62($\pm$ 0.43)	78.83($\pm$ 0.68)	78.7($\pm$ 0.82)	82.93($\pm$ 1.17)	83.7($\pm$ 0.67)	81.15($\pm$ 0.81)	80.55($\pm$ 0.67)
mean acc.		81.22 ($\pm$ 10.65)	**82.39** ($\pm$ 11.45)	72.9 ($\pm$ 11.24)	72.42 ($\pm$ 11.76)	77.95 ($\pm$ 11)	78.72 ($\pm$ 11.17)	78.03 ($\pm$ 10.3)	78.87 ($\pm$ 11.02)

image and its gray-scale conversion using the proposed method. The gray-scale conversion improved results of LBP and Gabor descriptors to any configuration used for the CAs construction. Concerning the GLCM descriptor, the performance is also improved in most of the cases, except from extreme ρ values (≈ 0 and ≈ 1). The overall performance of LBP using the binary images was 95.16%, while using the proposed gray-scale conversion it reaches 97.49%.

For the classification of CA rules independently within the 88-class scheme, the highest accuracy rate obtained is 94.36%, also with the LBP descriptor. These results are inferior to the results obtained with the Li-Packard scheme due to the larger number of classes, which difficult the discrimination. However, the gray-scale conversion with the proposed method also improved the classification results in most cases, except for the Fourier descriptors. The mean accuracy rate of LBP considering all configurations tested with the binary images was 81.22%, while gray-scale images reaches 82.39%. This suggests that the proposed gray-scale conversion improves the characterization of the CA visual patterns, therefore helping on the classification. Thus more attention should be given to the spatio-temporal patterns of CAs since it possesses an intrinsic characteristic that allows distinguishing between CA dynamic behaviors.

Acknowledgments

J. M. acknowledges a scholarship from the Brazilian agency CAPES (PROEX). L. C. R. acknowledges support from FAPESP (Grant #16/23763-8) and CAPES (Grant Nos. 9254772/m). L. F. S. S. acknowledges support from CNPq (Grant #134558/2016-2). O. M. B. acknowledges support from CNPq (Grant #307797/2014-7 and Grant #484312/2013-8) and FAPESP (grant #14/08026-1).

References

[1] A. Wuensche, Classifying cellular automata automatically: Finding gliders, filtering, and relating space-time patterns, attractor basins, and the z parameter., Complexity 4 (3) (1999) 47–66.

[2] D. Kunkle, Automatic classification of one-dimensional cellular automata, Master's thesis, Rochester Institute of Technology (2003).

[3] N. da Silva, J. Baetens, M. Oliveira, B. D. Baets, O. Bruno, Classification of cellular automata through texture analysis, Information Sciences 370–371 (2016) 33–49.

[4] J. M. Baetens, B. D. Baets, Phenomenological study of irregular cellular automata based on Lyapunov exponents and Jacobians, Chaos 20 (3) (2010) 033112.

[5] S. Wolfram, Cellular automata, Los Alamos Science 9 (1983) 2–27.

[6] W. Li, N. Packard, The structure of the elementary cellular automata rule space, Complex Systems 4 (1990) 281–297.

[7] S. Wolfram, Cellular Automata and Complexity: Collected Papers, 1-2150-A; Louisiana Barrier Island, Addison-Wesley, 1994.

[8] G. M. Oliveira, P. P. d. Oliveira, N. Omar, Definition and application of a five-parameter characterization of one-dimensional cellular automata rule space, Artificial Life 7 (3) (2001) 277–301.

[9] T. Ojala, M. Pietikäinen, T. Mäenpää, Multiresolution gray-scale and rotation invariant texture classification with local binary patterns, Pattern Analysis and Machine Intelligence, IEEE Transactions on 24 (7) (2002) 971–987.

[10] J. Fourier, The Analytical Theory of Heat, Cambridge University Press, 1878.

[11] R. M. Haralick, Statistical and structural approaches to texture, Proceedings of the IEEE 67 (5) (1979) 786–804.

[12] C. M. Bishop, Pattern Recognition and Machine Learning (Information Science and Statistics), Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[13] J. Zhang and T. Tan, Brief review of invariant texture analysis methods, Pattern recognition volume 35 (Elsevier), 735–747, 2002.