GENERALIZED CELLULAR AUTOMATA FOR EDGE DETECTION

ALINA ENESCU

ABSTRACT. Cellular Automata (CA) are simple, easily parallelizable models that have been used extensively for various computational tasks. Such models are especially useful for image processing, as mapping automaton cells to image pixels is straightforward and intuitive. This paper proposes a novel optimization framework for CA rules based on evolutionary algorithms and used in edge detection. This approach addresses the problem of optimizing an individual CA rule for one image and extends it to the optimization of a generic CA rule for a set of similar images. In order to maximize the transferability of the optimized rule, the algorithm is trained on sets of images using a curriculum learning approach. A study of the supervised fitness function as well as batch optimization experiments show that the algorithm is robust and competitive with the state-of-the-art methods.

1. INTRODUCTION

Cellular Automata (CA) are simple models of parallel computation easy to be mapped on images which makes them suitable to be used in image processing. The image processing task that will be tackled in the paper is edge detection in greyscale images. The scope is to find the most suitable edge detector for sets of images. The proposed approach uses a Genetic Algorithm (GA) to evolve the CA's rule to detect edges both on images individually and on sets of images using a curriculum learning setting.

The proposed approach is an improved version of a previous approach (7). Compared to this approach where only two parameters of the CA's rule are

optimized, in the proposed approach all three parameters of the CA's rule are optimized by the means of a GA.

Many edge detectors were proposed in the literature, among which several based on gradient and also several based on CA. As gradient based methods, the ones that are the most known are Canny detetctor (1), Sobel operator (22), while as CA based methods, there are two categories: simple CA (15; 5; 3) and evolved CA (26; 13; 8). The main difference between the proposed approach and the gradient based methods is that the proposed approach adapts to each image or sets of images, while the gradient based methods do not. Adaptability to images individually or to sets of images represents an advantage in image processing, leading to shorter time spent on new batches of images. The same property of adaptability may be seen at the evolved CA compared to simple CA where the rule is usually fixed and directly applied to the images. Compared to a similar evolved CA method (26) where one parameter of the CA's rule, the linear rule, is fixed and only the other two parameters are evolved by the means of a Particle Swarm Optimization (PSO) algorithm, in the proposed approach all three parameters are evolved by the means of a GA. Moreover, there are differences in the methodologies chosen to search for the optimal CA's rule compared to similar evolved CA approaches, GA instead of Learning Automata (LA) or PSO.

The experiments along with the results shown in this paper are meant to assess the capabilities of the proposed edge detector based on evolved CA.

The rest of the paper is structured as follows: Section 2 is a theoretical overview on the edge detection problem, cellular automata and the quality metric; related work in the field is briefly presented in section 3; section 4 details the proposed edge detection approach and the results are presented in Section 5; a summary of the experiments together with future improvements are discussed in Section 6.

2. Theoretical background

Before describing the methods proposed in the current literature, the problem of edge detection is formulated, followed by a brief introduction to Cellular Automata (CA) and some theoretical aspects regarding the performance measure of the methods.

2.1. **Edge detection problem.** A given point P_0 of a greyscale image having values within $\{0, 1, \ldots, 255\}$ and eight neighbours $P_i, i \in \{1, 2, \ldots, 8\}$ as in Figure 1a, may be classified as an edge (11) if a significant local change in the intensity of the neighbours $P_i, i \in \{1, 2, \ldots, 8\}$ is observed. The curve that contains the points classified as edges is called a contour.

The system meant to extract the set of edge points for a given image is an algorithm called edge detector. Given a set of images, finding an edge detector for them is what the edge detection problem refers to.
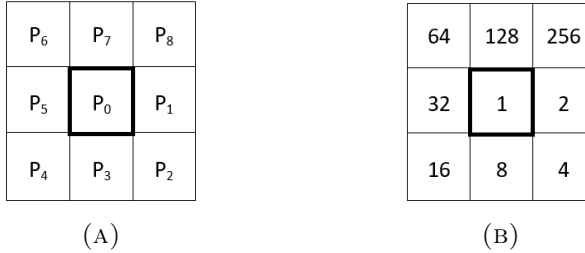
P_6	P_7	P_8
P_5	P_0	P_1
P_4	P_3	P_2

(A)

64	128	256
32	1	2
16	8	4

(B)

FIGURE 1. Visual representations of Moore neighbourhood. (A) A visual representation of a point P_0 and its eight surrounding neighbors $P_i, i \in \{1, 2, \ldots, 8\}$. (B) A visual representation of the nine fundamental rules for a 2D cellular automaton with a Moore neighbourhood, based on which a linear rule is obtained.

2.2. **Cellular Automata.** Cellular Automata (CA) are simple models of parallel computation (16). A CA is usually represented by a five tuple

$$CA = \{C, N, S, s_0, F\},$$

where C is the set of cells, N is the set of neighbors of size n: $n =\mid N \mid$, S is the set of states, s_0 is the initial state of the CA and $F : S^n \to S$ is the transition rule. The state of the cells are synchronously (or asynchronously) updated by the transition rule according to the state of the neighbor cells and the state of the cells themselves. The equation of the transition rule is:

$$s_i(t + 1) = F(s_i(t), S_i(t)),$$

where $s_i(t)$ and $s_i(t+1)$ are the states of the ith cell at time t and time $t + 1$, respectively, and $S_i(t)$ is the state set of the ith cell's neighbors at time t.

2.3. **Performance measure.** The Baddeley's Delta Metric (BDM), presented in (19) measures the dissimilarity between two binary images, but besides other metrics, it considers the spatial distribution of the pixels that differ.

Given I_1 and I_2, two binary images with the same sizes $M \times N$ and $\Omega = \{1, .., M\} \times \{1, .., N\}$ the set of each pixel's position in the images, one may define the l-BDM between these two images for a $0 < l < \infty$ as Equation 1 (denoted $\Pi^l(I_1, I_2)$):

$$(1) \qquad \Pi^l(I_1, I_2) = \left[\frac{1}{|\Omega|} \sum_{t \in \Omega} |w(d(t, b(I_1))) - w(d(t, b(I_2)))| \right]^l$$

where $b(I_i)$ is the set of feature (edge) points in I_i, $d(t, b(I_i))$ represents the distance between a position t and the closest feature point and $w : [0, \infty] \to [0, \infty]$ is a concave function. In the experiments the Euclidean distance was used, while $w(x) = x$ and $l = 2$.

3. Related work

There are multiple edge detection approaches in the literature which involve the use of Cellular Automata (CA). For greyscale edge detection, there are approaches such as (18) which applies on only one image the 512 linear rules and split them in three groups: no edge detection rules, strong edge detection rules and weak edge detection rules. In (14), the authors identifies only 4 rules out of the 512 linear rules as being the best for edge detection.

EAs have also been proposed for optimizing CA rules for edge detection. An edge detection approach based on CA, fuzzy logic and Particle Swarm Optimization (PSO) is presented in (26). Similar to (21), the authors uses a heuristic membership function to decide the cell's state according to its neighbours. Then a threshold is used to classify the cell as an edge or a non–edge by comparing it with the value obtained by the heuristic membership function.

Apart from these methods, we mention the classic well-known methods based on computing the gradient: Canny edge detector (1), Sobel operator and Prewitt operator (17). The Canny edge detector presented in (1) is a method characterized by three principles: detection, localization and single edge response. The first step of the Canny method is to smooth the greyscale image by Gaussian filtering, then to compute the first derivative in both vertical and horizontal directions. As the second step, it finds the magnitude and direction of each pixel. As a final step, it uses a non-maximal suppression to ensure that the edges have one-pixel width and it uses two thresholds to select the final edge points and to trace them.

4. Proposed approach

4.1. **Cellular Automata Model.** As described in the introduction, the proposed approach is an edge detector based on evolved CA. This method uses a 2D-CA for which the cells are arranged as a grid, the states values are $S = \{0, 1, ..., 255\}$, the neighborhood N is the Moore neighborhood and the initial state s_0 is extracted from the input image. To compute the next state,

first an *edge membership function* (24; 25) is applied. The *edge membership* for a pixel $P_0 \in \{0, 1, ..., 255\}$ and its eight neighbours $P_i \in \{0, 1, ..., 255\}$ is defined as:

$$(2) \qquad \mu(P_0) = \frac{\sum_i |P_0 - P_i|}{\Delta + \sum_i |P_0 - P_i|},$$

where $i \in \{1, 2, ..., 8\}$ is selected according to a linear rule. The parameter $\Delta \in \{0, ..., 255\}$ (one of the possible gray values for a pixel represented on 8 bits) is inversely proportional to the number of detected edges. The higher its value, the fewer edges will be detected. The transition rule is a function $F : P \mapsto \{0, 1\}$ given by Equation 3 (26) which gives the final edge map. The value of the threshold $\tau \in [0, 1)$ controls the number of pixels that pass as edges, therefore the higher the threshold the fewer values will pass.

$$(3) \qquad F(P_0) = \left\{ \begin{array}{ll} 1, & \text{if } \mu(P_0) > \tau \\ 0, & \text{if } \mu(P_0) \leq \tau \end{array} \right.$$

The quality of the edge detection is influenced by the Δ and τ parameters as well as the linear rule. Choosing these values, however, is a non-trivial task, therefore automatizing the search for the best values is necessary.

4.2. **Evolved Cellular Automata.** The GA used to evolve the CA's rule to detect edge points is presented in the following.

The chromosome encodes the three parameters that need to be optimized: the gray encoding (9; 2; 20) of the two parameters, Δ and τ, and the binary encoding of a linear rule. In order to use the gray encoding for the real parameter τ, it was encoded an integer value within $\{0, 1, \ldots, 127\}$ and divided it by 128 to obtain the real value when the Equation 3 is applied. The choice of using the gray encoding of the two parameters was to keep the genetic operators, crossover and mutation, used in the binary encoding and to overcome the Hamming Cliff (20) problem when using the binary encoding of an integer. Figure 2 shows an example of a chromosome encoding the binary representation of linear rule 35, highlighted in blue, the grey code of threshold, highlighted in purple and the grey code of parameter Δ, highlighted in red.

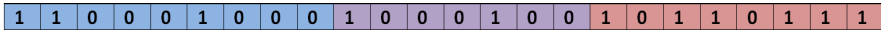| 1	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	1	0	1	1	0	1	1	1

FIGURE 2. An example of a chromosome encoding linear rule 35, the threshold $\tau = 120$ (for Equation 3 will be $120/128 = 0.9375$) and the parameter $\Delta = 218$.

In order to assess the quality of the individuals a fitness function is needed. The fitness function is computed as the Dice Similarity Coefficient (DSC)

(4; 23) from Equation 4. Because DSC quantifies the correctly detected edge points more precisely, it becomes more effective.

$$(4) \qquad DSC = \frac{2 \cdot TP}{2 \cdot TP + FP + FN},$$

where TP is the number of true positives, FP is the number of false positives and FN is the number of false negatives.

For crossover, the binary tournament selection is used to select two chromosomes and three random cross–point are chosen such as all three components to be crossed. The three–points crossover applied to the chromosomes is expressed in Figure 3. Then the bit flip mutation is applied to each offspring: for each component of the offspring, a random bit of the gene is selected and inverted by a probability. Both offspring are added to the new population. These genetic operators are applied to the population until the new population reaches the same size as the old one and replaces it completely.
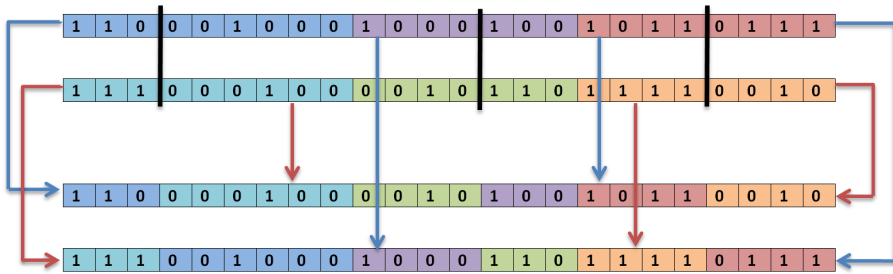

FIGURE 3. Three-points crossover between two chromosomes encoding linear rule 35 with threshold $\tau = 120$ and $\Delta = 218$, and linear rule 452 with threshold $\tau = 27$ and $\Delta = 163$, respectively. The two offspring that are obtained encode the linear rule 388 with threshold $\tau = 24$ and $\Delta = 220$, and linear rule 456 with threshold $\tau = 123$ and $\Delta = 165$, respectively.

4.3. **Framework.** The input of the proposed approach is a greyscale image on which the evolving process is performed.

Firstly, the initial configuration of the CA, called the input state, is extracted from the input image. The state of each CA's cell is given by the pixel value, while the neighbours of each CA's cell is a list of its eight surrounding pixels.

Further, the steps of the GA that optimizes the rules to detect edge points are described. The first step is generating the initial population: *PopSize* chromosomes, each encoding the binary representation of a linear rule, the gray code of a threshold and the gray code of Δ, are generated. Each chromosome is evaluated by the means of fitness function, then selection, crossover and mutation are applied producing a new population to which the best chromosome is added. These steps are repeated for G generations. At the end of the execution of the GA, one best chromosome is obtained.

5. Experiments and comparisons

In the following experiments, 22 images of sizes 321×481 (or 481×321) from Berkely dataset (BSD) (12), along with the available ground truth, and a synthetic data set containing geometric shapes, were used. Only 22 images from Berkely dataset (12) were used in these experiments, the same subset used in *Uguz et al.* (26), so that the proposed approach could be compared with the one presented in (26).

In the first part of experiments, the 22 images of the BSD subset are given as input to the GA that evolves the CA's rule and several runs are performed for each image. The GA runs for 100 generations, having a population of 50 chromosomes.
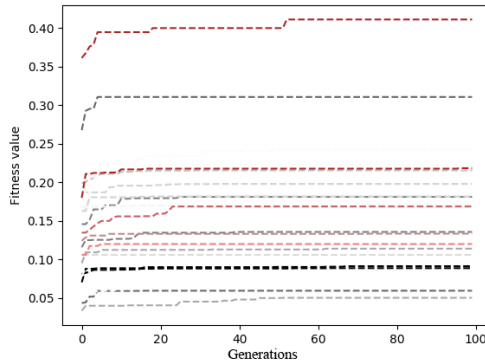

Figure 4. Best fitness values for the first 100 generations. Each line corresponds to a different image from the BSD subset.

A first analyse is aimed towards the convergence of the GA that evolves the CA's rule to detect edge points. The scope of this analysis is to emphasize that the GA proposed for this task keeps improving during the evolving process and reaches a maximum stable state. In the graph displayed in Figure 4, the best

values of the fitness were extracted for the first 100 generations of the GA run on each of the 22 images of the BSD subset.

According to the graph, the GA used in the optimization process of the CA's rule, converges due to the elitism that keeps at a high level the quality of the chromosomes. To ensure the diversity of the population, the mutation probability was set to 0.7 and the crossover operator was applied to each parameter encoded of the chromosome (see Figure 3).
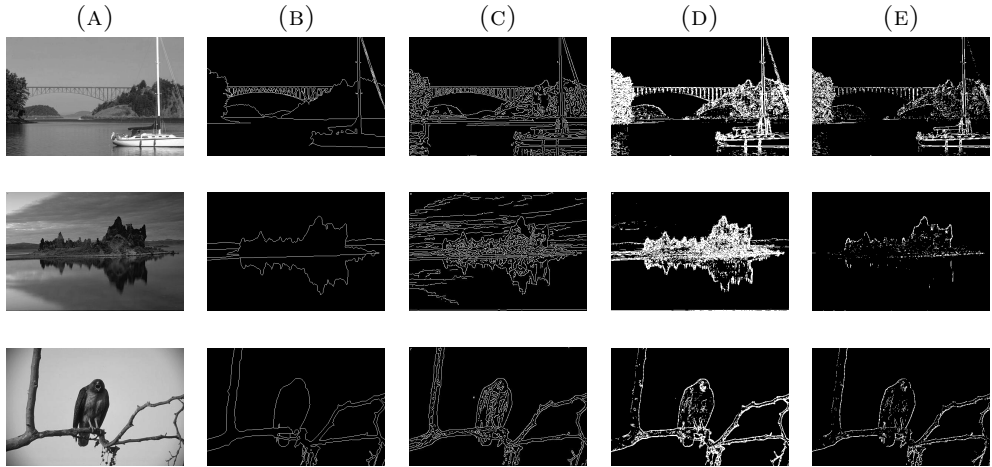

FIGURE 5. Images obtained by individual parameter optimization: (A) Input image. (B) Ground truth. (C) Canny (1) result. (D) *Uguz et al.* (26) result. (E) Proposed approach result.

A second analysis is performed on the performance of the proposed approach compared with the methods proposed in the current literature. The BDM values computed for the images obtained by applying the best parameters obtained by the proposed method, for the images obtained by applying the Canny edge detector (1) and for the images obtained by *Uguz et al.* (26) are compared. Best parameters found by this method along with the best parameters proposed by *Uguz et al.* (26) and BDM values obtained for the three methods are displayed in Table 1. The differences in the obtained values are explained by the differences in the proposed method compared to *Uguz et al.* (26) and *Canny* (1). Firstly, in the experiment conducted on the proposed approach, the Δ and τ parameters are optimized as well as the linear rule while *Uguz et al.* (26) only optimizes the first two. Secondly, the fitness

function used in this experiment is the Dice Similarity Coefficient (DSC). On average, the proposed method performs better than *Uguz et al.* (26) and *Canny* (1), however measuring the results on a larger data set would provide more significant insight. Three examples of the obtained images can be seen in Figure 5.

TABLE 1. All parameters values for *Uguz et al.* (26) and the ones found as best for the proposed approach, and BDM values for our proposed approach versus Canny (1) and *Uguz et al.* (26) on the 22 selected images in (26), with p-values computed based on Wilcoxon Signed-Ranks Test.

	Canny	Uguz et al.				Proposed approach				
Img.	BDM	Δ	τ	rule	BDM	Δ	τ	rule	BDM
Im 1	101.7416	110	0.62	47	74.9449	192	0.4687	436	74.5924
Im 2	36.2963	39	0.472	35	33.5381	181	0.0625	258	35.0743
Im 3	16.1734	59	0.414	139	5.1064	172	0.4140	156	4.4463
Im 4	41.01326	68	0.6078	154	24.2914	218	0.2578	132	31.9772
Im 5	28.7809	136	0.3889	60	41.8038	85	0.3437	285	27.911
Im 6	25.467	83	0.6642	194	72.0194	190	0.3515	150	19.7797
Im 7	15.956	104	0.194	50	14.6908	159	0.164	4	14.0289
Im 8	40.6465	107	0.4209	265	37.5357	113	0.414	4	35.4066
Im 9	26.6485	98	0.6854	309	20.9226	99	0.4218	134	19.0317
Im 10	19.8602	54	0.203	140	20.2130	174	0.3593	200	23.3961
Im 11	100.6888	87	0.45	66	86.2903	199	0.4296	468	81.0943
Im 12	29.0242	92	0.628	87	29.2202	110	0.4687	18	28.6511
Im 13	43.9345	29	0.3673	19	31.1133	251	0.3437	180	16.1749
Im 14	22.4603	76	0.3506	89	22.0495	218	0.125	204	21.4574
Im 15	13.892	103	0.5097	434	7.2010	106	0.3281	394	7.9008
Im 16	66.2486	70	0.55	75	63.2492	124	0.3984	286	60.9771
Im 17	78.2544	103	0.3818	104	3.5905	154	0.2656	30	13.6351
Im 18	21.0141	62	0.6186	46	17.8882	78	0.5546	157	16.7753
Im 19	42.8603	47	0.336	136	54.2173	79	0.164	6	37.0839
Im 20	22.1191	64	0.6141	195	19.2414	154	0.2656	30	21.4092
Im 21	33.3709	112	0.65	252	25.2466	80	0.375	6	25.9750
Im 22	24.0112	76	0.54	412	21.7873	187	0.2421	8	21.9363
Avg.	38.6574				33.0073				29.0325
p-value	0.1886				0.2234				

In this second part of the experiments, the focus is moved to evaluating the ability to generalize the behaviour of the proposed approach on multiple

images. An input dataset is selected to learn the parameters and a test dataset
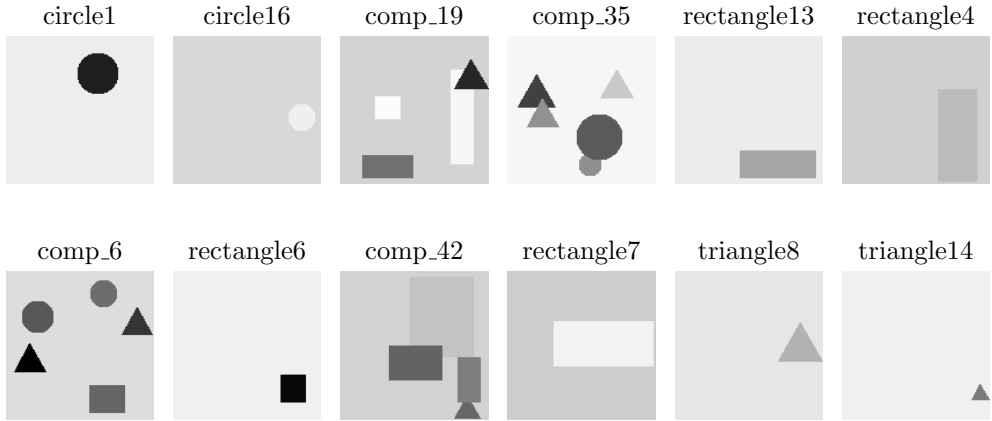to evaluate the previously optimized parameters.

circle1   circle16   comp_19   comp_35   rectangle13   rectangle4

comp_6   rectangle6   comp_42   rectangle7   triangle8   triangle14

FIGURE 6. Subset of the synthetic training set.

circle5   rectangle1   triangle3   comp_4   circle9   rectangle1

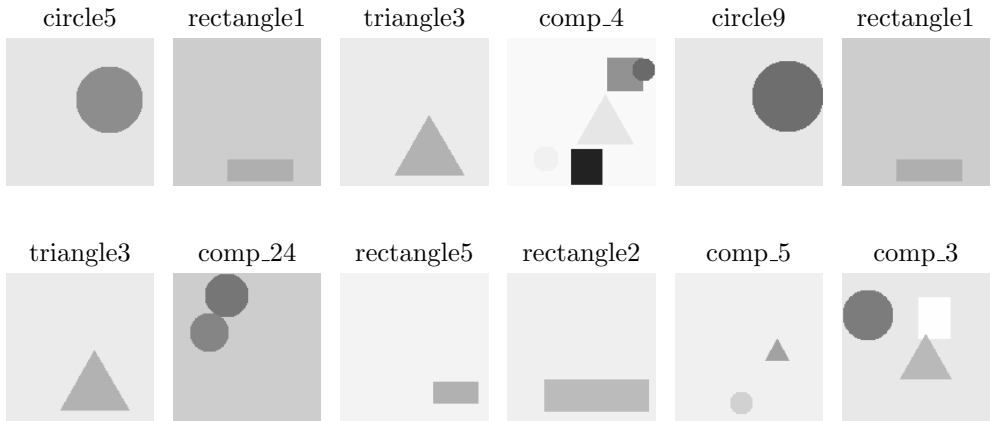triangle3   comp_24   rectangle5   rectangle2   comp_5   comp_3

FIGURE 7. Subset of the synthetic test set.

For this experiment, a dataset of images containing different geometric
shapes was generated. These images were then split into an input set (Figure
6) and a test set (Figure 7). The model was evolved using a curriculum learn-
ing strategy (6), meaning that the algorithm is fed new input examples in an

ascending order of difficulty. In this case, the difficulty was measured by the magnitude of the gradient between the shapes and the background. It was reported in the literature that feeding simpler examples first and then gradually increasing difficulty helps a learner generalize better (6; 10), which fits the aim to achieve transferability of the optimized parameters. For this reason, in the optimization process there were fed examples to the optimizer starting with high gradient images (edges easier to detect) and gradually progressing towards low gradient images (edges harder to detect).

	Canny	Sobel	Prop. Approach
BDM	1.08875	1.74193	0.64586
p-value	0.00002	0.15313	

(A)

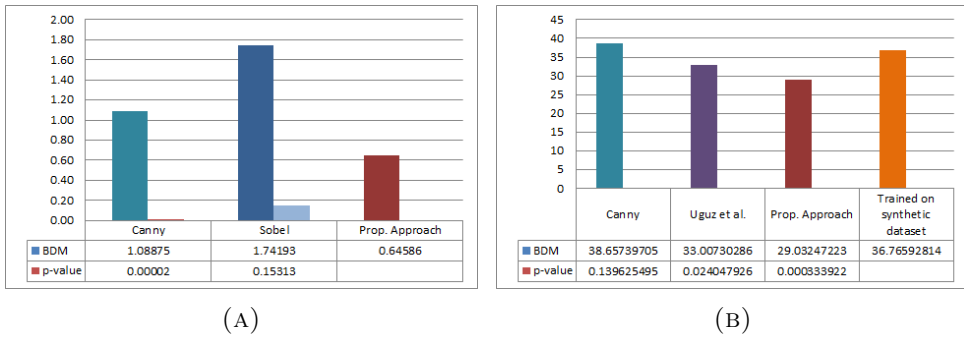|	Canny	Uguz et al.	Prop. Approach	Trained on synthetic dataset
BDM	38.65739705	33.00730286	29.03247223	36.76592814
p-value	0.139625495	0.024047926	0.000833922	

(B)

FIGURE 8. The proposed approach is compared with the other methods used in comparison in terms of a Wilcoxon Signed-Ranks Test. (A) BDM results on images used for the testing process. (B) BDM values for results obtained by applying the edge detector optimised on synthetic images versus Canny (1), Uguz et al.(26) and results obtained by individually training with the proposed approach on the 22 selected images in (26).

Figure 8a shows for the three compared methods, the average BDM values computed on the test set along with the p-values computed with the Wilcoxon Signed-Ranks Test. As may be seen, overall the proposed approach performs better than both Canny edge detector (1) and Sobel operator (17). Several examples of the obtained images can be seen in Figure 9. The main difference between the three methods may be seen in the last row, on images that contain more than one shape, where the proposed approach detects all edges, while the Canny edge detector (1) struggles to find the circle and Sobel operator (17) struggles to find both the circle and the triangle.

In order to assess the generalization capabilities of the model the parameters obtained on the synthetic dataset were also tested on the BSD subset used in *Uguz et al.* (26). One configuration used to evolve the parameters for the synthetic dataset had the best results on the BSD subset. For this

(A)   (B)   (C)   (D)   (E)
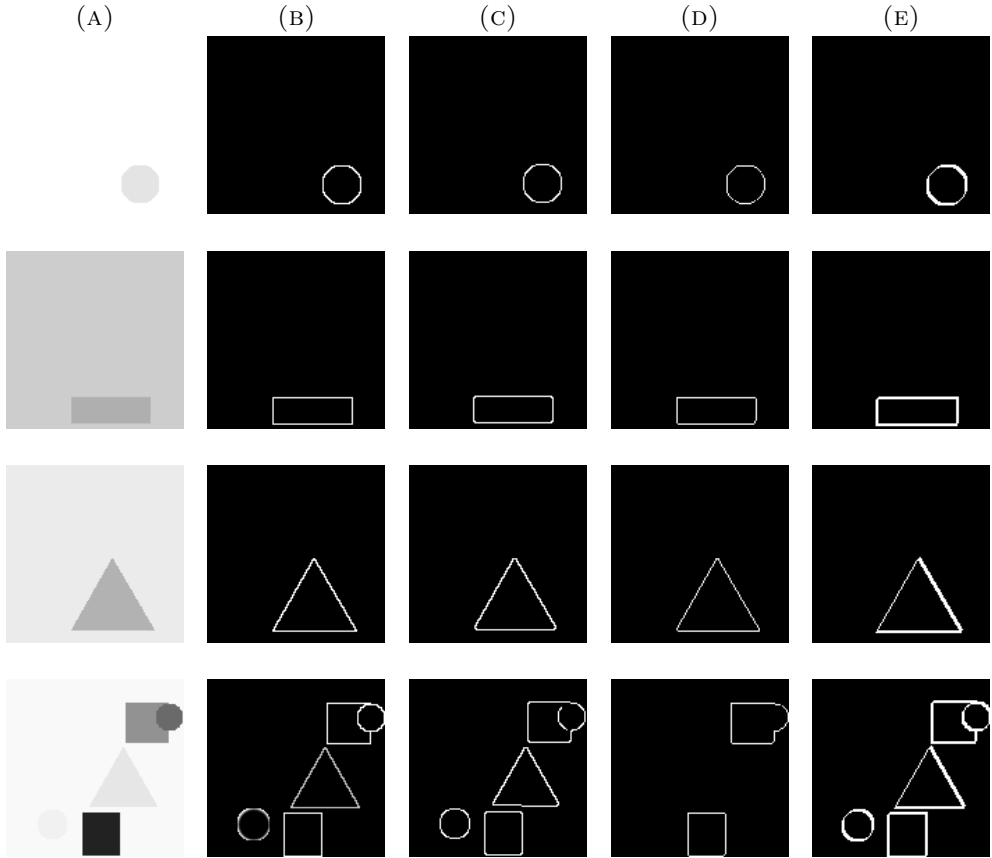

FIGURE 9. Batch optimization results on the synthetic test set. (A) Input image. (B) Ground truth. (C) Canny edge detector (1) result. (D) Sobel operator (17) result. (E) Proposed approach result.

configuration, 100 individuals were used. The graph from Figure 8b shows the average of the BDM values obtained for images in the BSD subset. It can be seen that, on average, the method proposed in *Uguz et al.* (26) and the proposed approach for individually evolving performs better than our model by a small margin, but our model performs better than Canny detector (1). However, the results from *Uguz et al.* (26) and of the proposed approach are obtained by supervised optimization on each individual image, whereas in the proposed model the rule is generalized for multiple images. This means that

the generalization capacity of the proposed model using a curriculum learning setting is comparable to individual optimization of the parameters on the test image. Sets of example images are presented in Figure 10. The main difference between these four methods may be seen in the first two rows, where both *Uguz et al.* (26) and proposed method on individual evolving detects the least edge points that makes them the most similar to the ground truth, while the proposed approach trained to generalize detects too many edge points in terms of BDM.
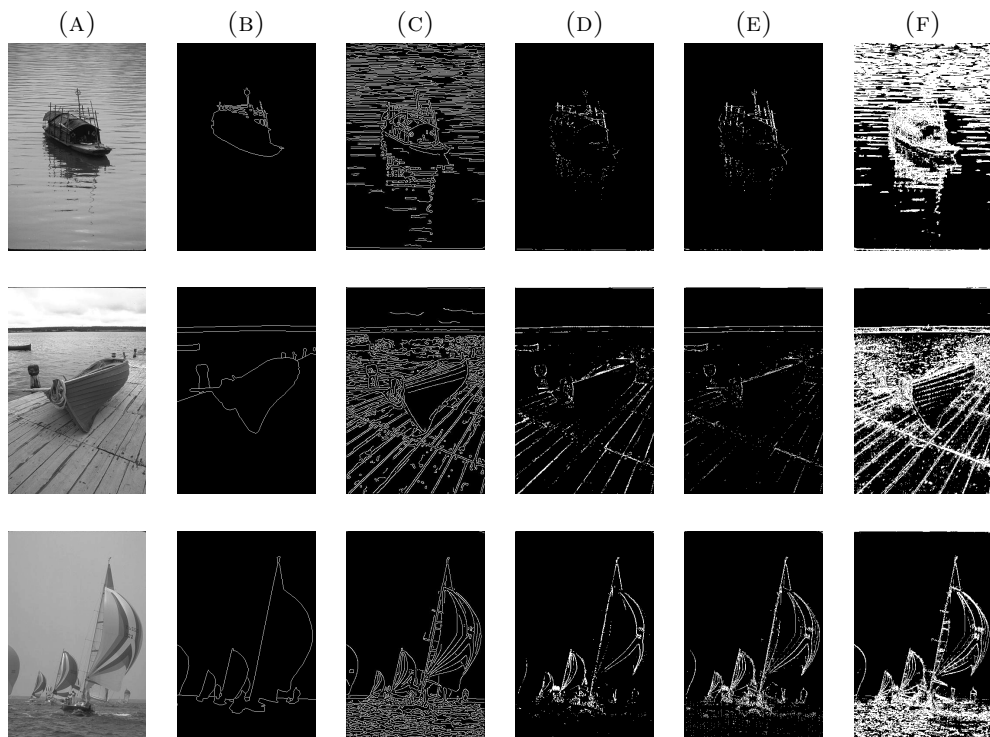

FIGURE 10. Images obtained by individual parameter optimization: (A) Input image. (B) Ground truth. (C) Canny (1) result. (D) *Uguz et al.* (26) result. (E) Proposed approach result. (F) Proposed approach results of training on the synthtic dataset.

6. Conclusions

In this paper it was presented an edge detector designed for greyscale images, based on evolved CA. The proposed method used one GA to optimize the three parameters of the CA's rule: the parameter Δ, the threshold τ and the linear rule r.

The proposed approach distinguishes from the methods presented in the literature by optimizing all the parameters by encoding all of them in a single chromosome, using both binary and grey encoding. The results obtained on both individual and batch optimization experiments indicate that the algorithm is robust and competitive with the best related method, such as Canny edge detector (1) and *Uguz et al.* (26). Moreover, the generalization capacity using a curriculum learning setting reached competitive performance related to individual optimization, which supports the idea of using transferable parameters.

As future work, more improvements can be done to both individual and batch optimization, such as using larger and more complex datasets.

References

[1] CANNY, J. A computational approach to edge detection. *IEEE Transactions on pattern analysis and machine intelligence*, 6 (1986), 679–698.

[2] CARUANA, R. A., AND SCHAFFER, J. D. Representation and hidden bias: Gray vs. binary coding for genetic algorithms. In *Machine Learning Proceedings 1988*. Elsevier, 1988, pp. 153–161.

[3] CHRISTIYANA, C. C., RAJAMANI, V., AND DEVI, A. U. Ultra sound kidney image retrieval using time efficient one dimensional glcm texture feature. *IJCA Special Issue on Advanced Computing and Communication Technologies for HPC Applications*, 4 (2012), 12–17.

[4] DICE, L. R. Measures of the amount of ecologic association between species. *Ecology 26*, 3 (1945), 297–302.

[5] DIWAKAR, M., PATEL, P. K., AND GUPTA, K. Cellular automata based edge-detection for brain tumor. In *2013 International Conference on Advances in Computing, Communications and Informatics (ICACCI)* (2013), IEEE, pp. 53–59.

[6] ELMAN, J. L. Learning and development in neural networks: The importance of starting small. *Cognition 48*, 1 (1993), 71–99.

[7] ENESCU, A., ANDREICA, A., AND DIOSAN, L. Evolved cellular automata for edge detection in grayscale images. In *2019 21st International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC)* (2019), IEEE, pp. 326–332.

[8] FARBOD, M., AKBARIZADEH, G., KOSARIAN, A., AND RANGZAN, K. Optimized fuzzy cellular automata for synthetic aperture radar image edge detection. *Journal of Electronic Imaging 27*, 1 (2018), 013030.

[9] FRANK, G. Pulse code communication, Mar. 17 1953. US Patent 2,632,058.

[10] HACOHEN, G., AND WEINSHALL, D. On the power of curriculum learning in training deep networks. *arXiv preprint arXiv:1904.03626* (2019).

[11] JAIN, R., KASTURI, R., AND SCHUNCK, B. G. Edge detection. In *Machine vision* (1995), vol. 5, McGraw-Hill New York, pp. 140–185.

[12] MARTIN, D., FOWLKES, C., TAL, D., AND MALIK, J. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on* (2001), vol. 2, IEEE, pp. 416–423.

[13] MOFRAD, M. H., SADEGHI, S., REZVANIAN, A., AND MEYBODI, M. R. Cellular edge detection: Combining cellular automata and cellular learning automata. *AEU-International Journal of Electronics and Communications 69*, 9 (2015), 1282–1290.

[14] MOHAMMED, J., AND NAYAK, D. R. An efficient edge detection technique by two dimensional rectangular cellular automata. *CoRR abs/1312.6370* (2013).

[15] NAYAK, D. R., PATRA, P. K., AND MAHAPATRA, A. A survey on two dimensional cellular automata and its application in image processing. *IJCA Proceedings on International Conference on Emergent Trends in Computing and Communication (ETCC-2014) ETCC*, 1 (2014), 78–87.

[16] NEUMANN, J., BURKS, A. W., ET AL. *Theory of self-reproducing automata*, vol. 1102024. University of Illinois press Urbana, 1966.

[17] POOBATHY, D., AND CHEZIAN, R. M. Edge detection operators: Peak signal to noise ratio based comparison. *International Journal of Image, Graphics and signal processing 6*, 10 (2014), 55.

[18] QADIR, F., PEER, M. A., AND KHAN, K. A. Efficient edge detection methods for diagnosis of lung cancer based on two dimensional cellular automata. *Advances in Applied Science Research 4*, 3 (2012), 2050–2058.

[19] RUE, H. Baddeley's delta metric. In *Banks (eds), Encyclopedia of Statistical Sciences, Vol. Update Volume* (1996), Citeseer.

[20] SCHAFFER, J. A study of control parameters affecting online performance of genetic algorithms for function optimization. *San Meteo, California* (1989).

[21] SINAIE, S., GHANIZADEH, A., MAJD, E. M., AND SHAMSUDDIN, S. M. A hybrid edge detection method based on fuzzy set theory and cellular

learning automata. In *2009 International Conference on Computational Science and Its Applications* (2009), IEEE, pp. 208–214.

[22] Sobel, I., and Feldman, G. A 3x3 isotropic gradient operator for image processing. *a talk at the Stanford Artificial Project in* (1968), 271–272.

[23] Sørensen, T. *A Method of Establishing Groups of Equal Amplitude in Plant Sociology Based on Similarity of Species Content and Its Application to Analyses of the Vegetation on Danish Commons.* Biologiske skrifter. I kommission hos E. Munksgaard, 1948.

[24] Tizhoosh, H. R. Fast fuzzy edge detection. In *2002 Annual Meeting of the North American Fuzzy Information Processing Society Proceedings. NAFIPS-FLINT 2002 (Cat. No. 02TH8622)* (2002), IEEE, pp. 239–242.

[25] Tizhoosh, H. R., Haussecker, H., Jähne, B., Haussecker, H., and Geissler, P. Fuzzy image processing: an overview. *Handbook on computer vision and applications, Academic Press, Boston* (1998).

[26] Uguz, S., Sahin, U., and Sahin, F. Edge detection with fuzzy cellular automata transition function optimized by pso. *Computers & Electrical Engineering 43* (2015), 180–192.

Babeş-Bolyai University, Department of Computer Science, 1 M. Kogălniceanu Street, 400084 Cluj-Napoca, Romania
  Email address: `aenescu@cs.ubbcluj.ro`