

Selection of Optimum Rule Set of Two Dimensional Cellular Automata for Some Morphological Operations

Anand Prakash Shukla

Computer Science and Engineering Department, Motilal Nehru National Institute of Technology, Allahabad, India
Email: anandskla@gmail.com

Suneeta Agarwal

Computer Science and Engineering Department, Motilal Nehru National Institute of Technology, Allahabad, India
Email: suneeta@mnnit.ac.in

Abstract—The cellular automaton paradigm is very appealing and its inherent simplicity belies its potential complexity. Two dimensional cellular automata are significantly applying to image processing operations. This paper describes the application of cellular automata (CA) to various morphological operations such as thinning and thickening of binary images. The description about the selection of the optimum rule set of two dimensions cellular automata for thinning and thickening of binary images is illustrated by this paper. The selection of the optimum rule set from large search space has been performed on the basis of sequential floating forward search method. The misclassification error between the images obtained by the standard function and the one obtained by cellular automata rule is used as the fitness function. The proposed method is also compared with some standard methods and found suitable for the purpose of morphological operations.

Index Terms—Cellular Automata, Misclassification Error, Sequential Floating Forward Search, Thinning, Thickening, Morphological Operations.

I. INTRODUCTION

Cellular Automata is also called systems of Finite Automata, i.e. Deterministic Finite Automata (DFA) arranged in an infinite, regular lattice structure [4]. In cellular automata state of a cell at the next time step is determined by the current states of a surrounding neighborhood of cells along with its own state and is updated synchronously in discrete time steps.

Formally, a (bi-directional, deterministic) cellular automaton is a triplet

$$A = (S; N; \delta) \quad (1)$$

Where, S is an non-empty state set, N is the neighborhood system, and

$$\delta: S^N \rightarrow S \quad (2)$$

is the local transition function (rule).

Due to simple structure and its property to model the complex systems, cellular automata has attracted various researchers from different areas. Cellular automata primarily announced by Ulam[1] and Von Neuman [2] in 1950's with the purpose of obtaining models of biological self-reproduction and also discussed in the book of Wolfram 'A New Kind of Science'[3]. Now a day's Cellular Automata became very popular because of its diverse function and utility as a discrete model for many processes. Cellular automata make up a very important class of completely discrete dynamical systems. Cellular Automata also provide a concept for computational automata.

Commonly used neighborhood systems are the von Neumann and Moore neighborhoods.

A. Von Neumann Neighborhood

A diamond shaped neighborhood that can be used to define a set of cells surrounding a given cell (x_0, y_0) that may affect the evolution of a two dimensional cellular automaton on a square grid. The von Neumann neighborhood of range r is defined by the equation 3.

$$N_{x_0, y_0} = \{(x, y) : |x - x_0| + |y - y_0| \leq r\} \quad (3)$$

The von Neumann neighborhood is illustrated in Fig 1.

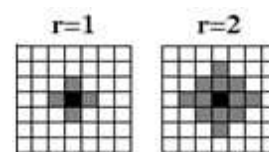


Fig. 1. Von Neumann Neighborhood

B. Moore Neighborhood

A square-shaped neighborhood that can be used to define a set of cells surrounding a given cell (x_0, y_0) that may affect the evolution of a two-dimensional cellular automaton on a square grid. The Moore neighborhood of range r is defined by the equation 2

$$N_{x_0, y_0} = \{(x, y) : |x - x_0| \leq r, |y - y_0| \leq r\} \quad (4)$$

Moore neighborhoods is illustrated in Fig. 2

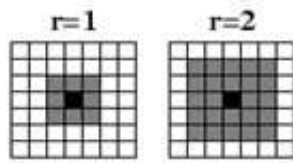


Fig. 2. Moore Neighborhood

C. Application of Cellular Automata

Some applications of cellular automata in modeling, controlling and analyzing of the behavior of complex are: The Games of Life [5], biological systems [6], Cellular automata in environmental system and ecological system [7], [8], CA in edge deduction [9], [10], [11], Cellular automata in the traffic system [12], CA in image processing [13]. Cellular automata in machine learning and control [14] and CA in cryptography [15].

Digital image processing(DIP) has very wide applications and almost all of the technical fields are impacted by DIP and it plays an important role in real life applications such as traffic analysis [32], satellite television, computer tomography and magnetic resonance imaging as well as in areas of research and technology such as biological information systems and astrophysics[16].Also it is used in Image sharpening and restoration, feature selection[31], Medical field, Remote sensing, Transmission and encoding, Machine/Robot vision, Color processing[30], Pattern recognition, Video processing, Microscopic Imaging etc.

Cellular automata are used in various image processing tasks such as image filtering in a better way than some existing filters in denoising process [17], [18], connected set morphology thinning and thickening of images [17], edge detection in digital images that provide boundaries of images[19], image segmentation which is an integral part of image processing applications like medical image analysis and photo editing[20], [19] and in Image Enhancement because of its dynamic behavior[21]. The advantage of cellular automata is that it provides simplicity in complexity i.e. though each cell has an extremely limited view of the system (just its immediate neighbors) and each cell generally contains simple rules, the combination of an array of cells with their local interaction leads to more sophisticated emergent global behavior of the system. This paper concentrates on the selection of the set of rules among the all possible combination of two dimensional cellular automata rules for the thinning and thickening process with their applications in different fields.

II. OPTIMIZATION OF CELLULAR AUTOMATA RULES

Cellular automata rule optimization is basically used to select the best possible rule set, among the all possible combination of rules, to perform the particular task, such

as performance of cellular automata in image processing like- calculating distance features, Template Matching, noise filtering, thinning thickening, Image Sharpening, Simple Object Recognition using rule sets which provide specific operations to their states at each step of time.

A. Related Work

To find out the optimum rule set, feature selection is acquiescent and can be performed using branch and bound algorithms [22]. In automatically optimization of rules most of the researchers focus in the density classification problem[20] that is recognized as a standard for exploring cellular automata rules with universal properties. Some authors further applied standard genetic algorithm (GA) for learning rules [20] [23]. A standard genetic programming platform is used in learning and training of cellular automata[24]. There are various feature selection methods that were introduced also consider GA and SFFS (sequential floating forward search)[24], [25], [17], where SFFS have better characteristics over the GA approach. Rule set automatically generated by evolutionary algorithms but probably this procedure was applied to artificial problems like- Majority Problem or for boundary detection in binary images[26], [27]. It depends on a rule set produced exact target output by increasing the size of neighborhood [28]. A form of hill climbing and backtracking algorithms also used to identify the rules. To optimize the cellular automata rules for image's thinning and thickening process Paul Rosin[17] has been used root mean square(RMS) error criterion and Hausdorff distance error measures with SFFS procedure as objective function. But this approach did not produce satisfactory results and produced some lines in fragmented way. Fragmented problem overcame by an algorithm and two cycle cellular automata to produce better results in learning of rules[17].

III. OPTIMIZATION OF CELLULAR AUTOMATA RULES FOR IMAGE THINNING AND THICKENING

Thinning and Thickening words come under the morphology operations that refer to form and structure; in computer visualization it can be used to refer to the shape of a region.

Thinning algorithms on binary images have a long history in image processing, because of their value in deriving higher representations (and compressed encodings) of the information in a bitmap. Thinning is a procedure to remove selected foreground pixels from binary images like erosion or opening. It is normally applied to binary images and produces another binary image as output. It can be used for several applications, but is particularly useful for skeletonization.

Thickening is a morphological operation that is used to breed particular regions of forefront pixels in binary images, that could be seem as dilation or closing. Basically Thickening process only applied to binary images, and by this it produces another binary image as output. The thickening operation is related to the hit-and-miss transform, There are various applications in image

processing where thinning and thickening are used for different purposes as- Thinned images possess a subset of the original information that is useful for applications such as segmentation, feature extraction, vectorization, and pattern identification and thickening includes determining the approximate convex hull of a shape, to study any disease growing cell in spreading manner and determining the skeleton by zone of influenced region.

IV. METHODOLOGY

In this paper the binary images are considered i.e. cells have two states(white or black) and Moore neighborhood is considered in which the cells are eight ways connected and fixed value boundary conditions are applied which means transition rules are only applied to non boundary cells. The initial cell values are considered as the pixel values of the input image. As the Moore neighborhood is considered there are 2^8 rules are possible which are reduced to 51 by considering the 45^0 rotational symmetry and bilateral reflections [17]. Now the aim is to find the rule or set of rules among these selected 51 rules which provides the best thinning and thickening of the given image. Before applying the rules, the thinning and thickening of input image is obtained by the standard

function `bwmorph()` available in the MATLAB and the resulting images are saved for the further references.

In order to select the best rule set for thinning of white portion, first the 3×3 neighborhood pattern of rule, as shown in Fig. 3, is compared with the 3×3 neighbors of the current pixel of input image. If both are not same then the central pixel is inverted in case of central white pixel. This operation is performed for all white pixels of the image. Each rule among 51 rules is applied at one time and the resulting image is compared with the image obtained by the standard function `bwmorph()`. Then the misclassification error[29] has been consider which is best suited for the classification of the thinned(thickened) white portion with standard function and rule set. The misclassification error is defined as

$$\text{error} = 1 - \frac{|B_o \cap B_T| + |F_o \cap F_T|}{B_o + F_o} \quad (5)$$

Where, F_o and B_o shows the white and black pixels of the image obtained by the cellular automata rule whereas F_T and B_T are the white and black pixels of target image obtained by the standard function.

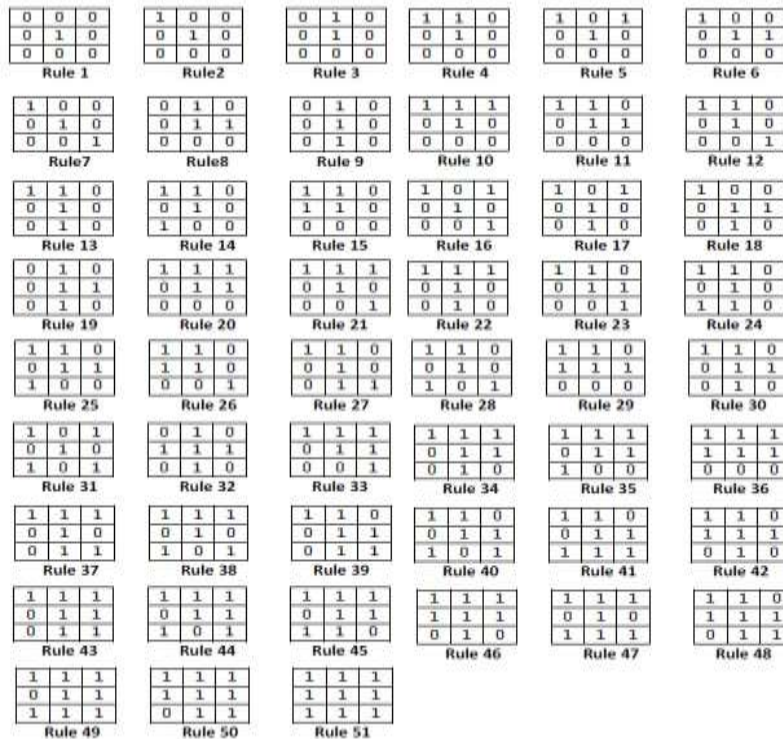


Fig. 3. Rule Set under Consideration

To compute the rule set for the thickening of the image the same procedure is used for the central black pixel i.e. if the central pixel is black and the encoded neighborhood pattern does not match with the encoded rule then the central pixel is inverted.

Proposed approach is providing Thinning and Thickening of white region on different images taking the advantage of cellular automata rules. The current paper shows only a subpart of test images to shown a fine visibility of the result.

Algorithm 1 Algorithm for Thining by Cellular Automata Rules

```

1: PROCEDURE CA_THIN(A)
2:   Input : image A
3:   m ← 1
4:   B ← A      // Save image A secondary image B
5:   E ← bwmorph(B) // thin the images by standard function bwmorph(),
6:   repeat
7:     C ← B
8:     For every pixel C[i,j] and Rule number = m
9:     begin
10:      S1 ← 3 × 3 neighbors of C[i][j].
11:      S2 ← 3 × 3 matrix of Rule m
12:      if C[i, j] = white and S1 != S2 then
13:        B[i, j] ← Invert (C[i, j])
14:      end for
15:      error[m] = 1 -  $\frac{|Bo \cap BT| + |Fo \cap FT|}{Bo + Fo}$ 
16:      m ← m+1
17:    until m = 51

```

Algorithm 2 Sequential Floating Forward Search

```

1: procedure SFFS
2: Step 1:
3:   Y ← { φ }
4: Step 2:
5:   Select the best feature
6:   x+ ← argmaxJ (Zk + x) | x ∉ Zk
7:   Zk ← Zk + x+
8:   k=k+1
9: Step 3:
10:  Select the worst feature
11:  x- ← argmaxJ (Zk - x) | x ∈ Zk
12: Step 4:
13:  if J (Zk - x-) > J (Zk) then
14:    Zk ← Zk - x-
15:    k = k + 1
16:    Go to Step 3
17:  else
18:    Go to Step 2

```

The proposed method is shown as the procedure CA_THIN in algorithm 1. The rules are referred in the procedure by the rule number. The rules are numbered from 1 to 51 in Fig. 3 from left to right and upwards down manner. The proposed algorithm is initialized by setting the value of m as 1, here m shows the rule number. To find the thinning of the white portion, only white portion of the image is considered. For every white pixel, the 3x3 neighbors of pixel is considered and it is matched with the rule selected. If the 3x3 pattern of the pixel does not match with the rule under consideration then the central pixel is inverted otherwise it remains unchanged. This is applied for whole white portions of the image. When the whole image has been processed the misclassification error shown by the equation 5 is computed and stored in an array of error, where index of the array shows the error found by the particular rule number. Note that the black portion of the image is not considered here.

In each step each image pixel should be processed in

parallel. However, since it is used as sequential operation, repeated until the maximum number of pixels matched to the standard method, the processed pixels are stored in a secondary image C. At the end of every iteration this processed pixel is copied back to original image.

In order to find the thickening of the white portion the same procedure with slight change can be used. To find the thickening, the algorithm 1 is applied to the black portion of the image. It should be noted that the rules shown in Fig. three are only for central white pixels. For central black pixel the rules shown in Fig. 3 are inverted.

To find the optimum set of the rules the sequential floating forward search(SFFS) method is used. The SFFS is a deterministic search method which provide the effectiveness equivalent to the genetic algorithm (GA)[23]. The algorithm 2 shows the SFFS algorithm.

Let Y_k denote the rule set at iteration and its score be $J(Y_k)$. Here, It is defined by the result of the algorithm by applying the CA rule set to the input image and computing the fitness function. Step 1 show that the

initial set is empty. In step 2 at each iteration, all rules are considered for addition to the rule set and only the rule giving the maximum score is added the resulting rule set. This process is repeated until no improvements in score are gained by adding rules. In step 3, each rule in rule set found in step 2 is removed to find the rule whose removal provides the resulting rule set with the improved value of objective function. As shown in step 4 if removal of the rule cause the better score of the objective function then it is discarded from the rule set and again next rule is tried for the deletion and process go to step 3. Otherwise, the process go to step 2 for the addition of new rule to the rule set.

V. EXPERIMENTAL RESULTS

Following experiments are performed by using MATLAB 2014a. Considering 51 rules, which are used to provide thin or thick images at different iteration. Aforementioned procedure has been implemented and one pixel thinned image generated by the function `bwmorph()` available in MATLAB has been considered for the comparison purpose. In order to train the rule set 20 binary sample images are carefully selected in such a way that the images contains all kind of shapes such as lines, circles, cones and some complicated structures. Then the rules are trained by using the objective function misclassification error with the thinned image by the SFFS method as described in the previous section. In all the images the trained rule set which gives the best thinning result is rule number 51 as shown in Fig. 4.

1	1	1
1	1	1
1	1	1

Rule 51

Fig. 4. Rule learned for Thinning and Thickening

Fig. 5 shows the original input image. Then learned rule 51 is applied to these images which produced result in Fig. 6 that are compared to thinned images produced by standard function `bwmorph()` in shown in Fig. 7. Similarly, Fig. 8 and 9 show the results for thickening. It is also found that rule number 51 have the minimum misclassification error for the thickening operation and the images of Fig. 8 shows the result of the thickening performed by the rule 51 and Fig. 9 shows the thickening result of `bwmorph()`. Besides these images experiment of optimization of cellular automata for thinning and thickening of images using best rules set at which minimum misclassification error is occurred in corresponding to standard function is done on several other images and in all the cases the rule number 51 shown in Fig. 4 is found best for the thinning and thickening operations.

Since the proposed algorithm is simulated on the sequential machine in MATLAB 2014a the time required to find the result is high. But once the rule set is trained for the thinning and thickening, then it is quite easier to apply the trained rule set to the images to find the desired result.



Fig. 5. Original Images

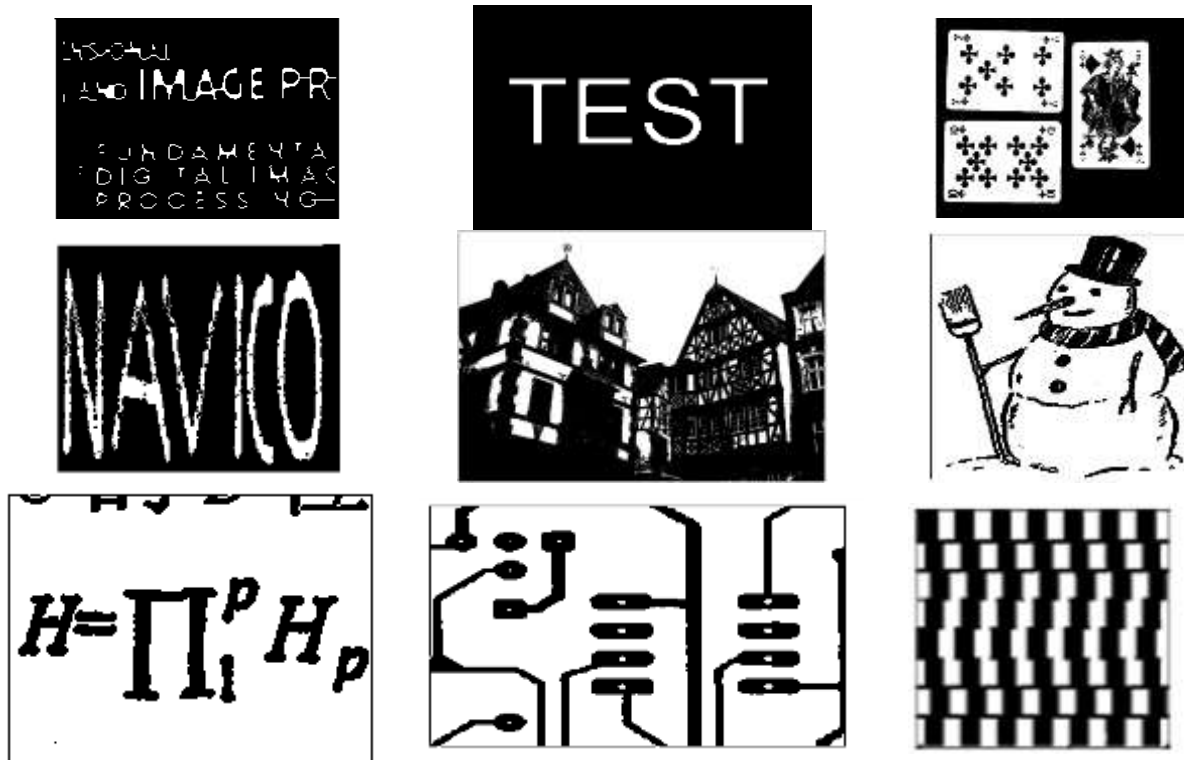


Fig. 6. Image Thinned by cellular automata rule.

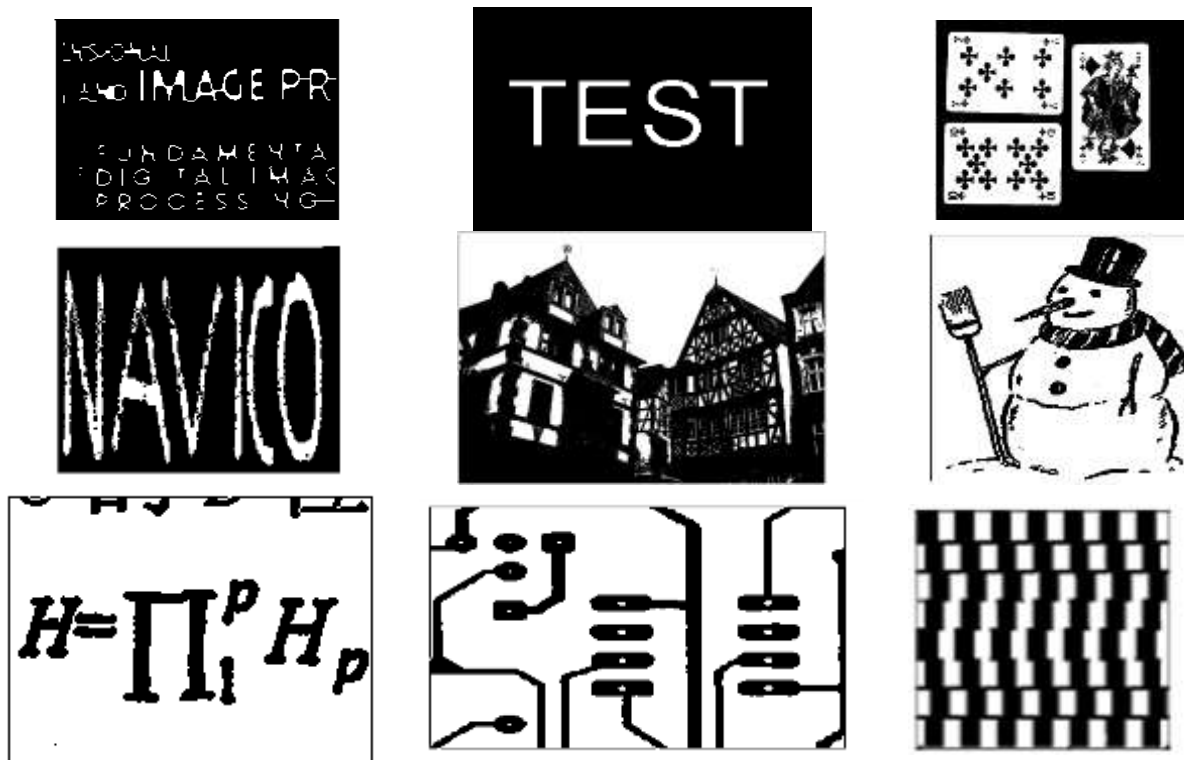


Fig. 7. Image Thinned by Standard Function

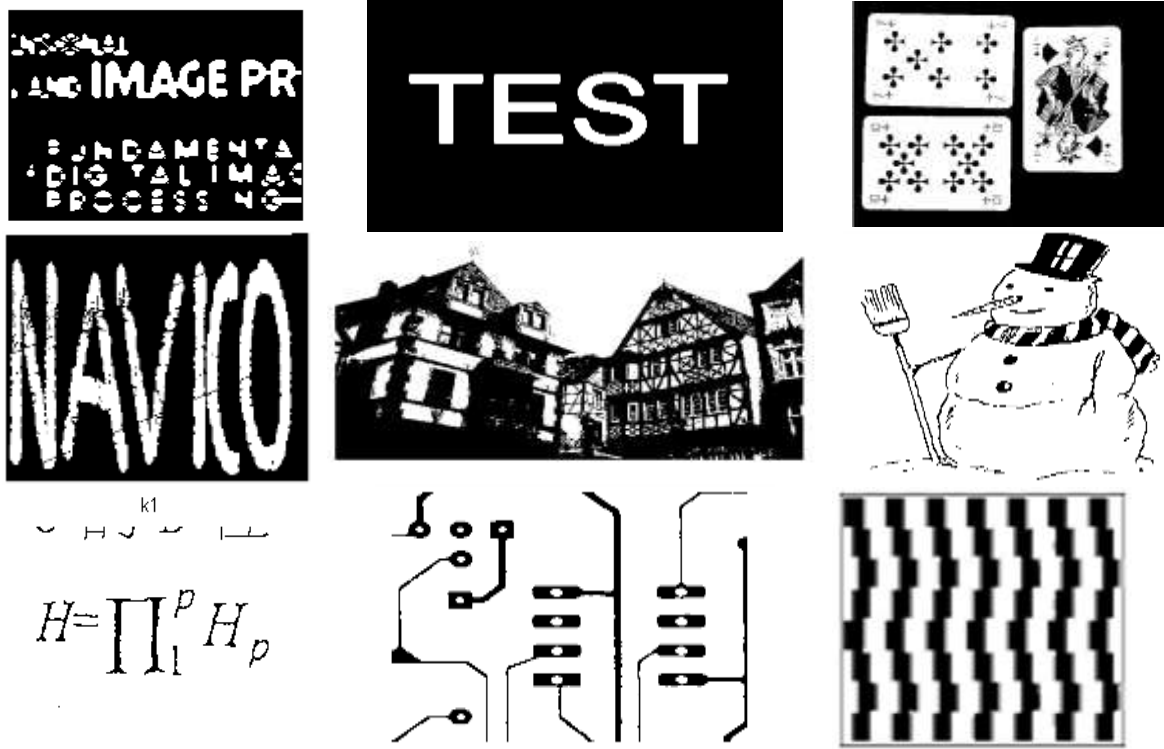


Fig. 8. Images Thickend by cellular automata rule.

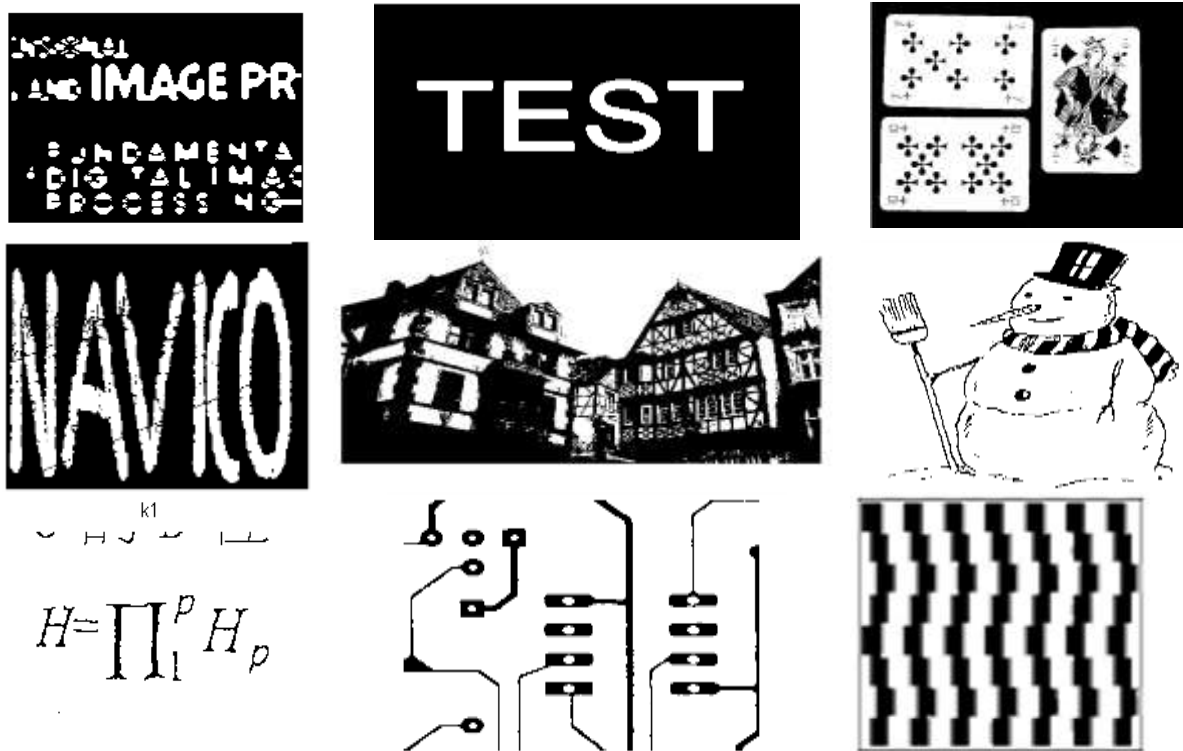


Fig. 9. Images Thickend by standard function.

VI. CONCLUSION AND FUTURE WORK

Results for the thinning and thickening of binary images through cellular automata rules are encouraging. The resulting rule sets provides comparable thinning and

thickening as compared with the standard function `bwmorph()`. Implementation and use of cellular automata rules is easier and straight forward as compare to other methods and also the size and the dimensions of images does not affect the complexity of the cellular automata. It is also required to mention that the aim to use the cellular

automata is to provide simplicity to solve the complex processes which is fulfilled in our experiment and clearly depicted in the results section. As the work presented in the paper is limited in thinning and thickening with the use of the sequential floating forward search method. It can be extended for some other morphological and image processing operations.

REFERENCES

- [1] S. Ulam, "Some ideas and prospects in biomathematics," *Annual review of biophysics and bioengineering*, vol. 1, no. 1, pp. 277–292, 1972.
- [2] J. Von Neumann, A. W. Burks et al., "Theory of self-reproducing automata," 1966.
- [3] S. Wolfram, *A new kind of science*. Wolfram media Champaign, 2002, pp. 5.
- [4] M. Sipper, "The evolution of parallel cellular machines: Toward evol- ware," *BioSystems*, vol. 42, no. 1, pp. 29–43, 1997.
- [5] M. Gardner, "Mathematical games: The fantastic combinations of john conways new solitaire game life," *Scientific American*, vol. 223, no. 4, pp. 120–123, 1970.
- [6] H. De Garis, "Cam-brain the evolutionary engineering of a billion neuron artificial brain by 2001 which grows/evolves at electronic speeds inside a cellular automata machine (cam)," in *Towards evolvable hardware*. Springer, 1996, pp. 76–98.
- [7] S. Aassine and M. C. El Jai, "Vegetation dynamics modelling: a method for coupling local and space dynamics," *Ecological modelling*, vol. 154, no. 3, pp. 237–249, 2002.
- [8] R. Smith, "The application of cellular automata to the erosion of landforms," *Earth Surface Processes and Landforms*, vol. 16, no. 3, pp. 273–281, 1991.
- [9] C.-I. Chang, Y.-j. Zhang, and Y.-Y. Gdong, "Cellular automata for edge detection of images," in *Machine Learning and Cybernetics, 2004. Proceedings of 2004 International Conference on*, vol. 6. IEEE, 2004, pp. 3830–3834.
- [10] T. Kumar and G. Sahoo, "A novel method of edge detection using cellular automata," *International Journal of Computer Applications*, vol. 9, no. 4, pp. 0975–8887, 2010.
- [11] S. Wongthanavasuu and R. Sadananda, "Pixel-level edge detection using a cellular automata-based model," *Advances in Intelligent Systems: Theory and Applications*, vol. 59, pp. 343–351, 2000.
- [12] M. Fukui and Y. Ishibashi, "Traffic flow in 1d cellular automaton model including cars moving with high speed," *Journal of the Physical Society of Japan*, vol. 65, no. 6, pp. 1868–1870, 1996.
- [13] A. Popovici and D. Popovici, "Cellular automata in image processing," in *Fifteenth International Symposium on Mathematical Theory of Networks and Systems*, vol. 1, 2002.
- [14] F. M. Marchese, "A directional diffusion algorithm on cellular automata for robot path-planning," *Future Generation Computer Systems*, vol. 18, no. 7, pp. 983–994, 2002.
- [15] S. Nandi, B. Kar, and P. Pal Chaudhuri, "Theory and applications of cellular automata in cryptography," *Computers, IEEE Transactions on*, vol. 43, no. 12, pp. 1346–1357, 1994.
- [16] L. S. Davis, "A survey of edge detection techniques," *Computer graphics and image processing*, vol. 4, no. 3, pp. 248–270, 1975.
- [17] P. L. Rosin, "Training cellular automata for image processing," *Image Processing, IEEE Transactions on*, vol. 15, no. 7, pp. 2076–2087, 2006.
- [18] A. P. Shukla, S. Chauhan and S. Agarwal "Training of cellular automata for image filtering," in *Proc. Second International Conference on Advances in Computer Science and Application - CSA 2013, LNCS*, pp. 86–95, 2013.
- [19] Y. Y. Boykov and M.-P. Jolly, "Interactive graph cuts for optimal boundary & region segmentation of objects in nd images," in *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, vol. 1. IEEE, 2001, pp. 105–112.
- [20] M. Mitchell, J. P. Crutchfield, R. Das et al., "Evolving cellular automata with genetic algorithms: A review of recent work," in *Proceedings of the First International Conference on Evolutionary Computation and Its Applications (EvCA96)*, 1996.
- [21] G. Sahoo, T. Kumar, B. Raina, and C. Bhatia, "Text extraction and enhancement of binary images using cellular automata," *International Journal of Automation and Computing*, vol. 6, no. 3, pp. 254–260, 2009.
- [22] P. Somol, P. Pudil, and J. Kittler, "Fast branch & bound algorithms for optimal feature selection," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 26, no. 7, pp. 900–912, 2004.
- [23] A. Jain and D. Zongker, "Feature selection: Evaluation, application, and small sample performance," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 19, no. 2, pp. 153–158, 1997.
- [24] D. Andre, F. H. Bennett III, and J. R. Koza, "Discovery by genetic programming of a cellular automata rule that is better than any known rule for the majority classification problem," in *Proceedings of the First Annual Conference on Genetic Programming*. MIT Press, 1996, pp. 3–11.
- [25] H. Hao, C.-L. Liu, and H. Sako, "Comparison of genetic algorithm and sequential search methods for classifier subset selection." in *ICDAR. Citeseer*, 2003, pp. 765–769.
- [26] A. Adamatzky, "Automatic programming of cellular automata: identification approach," *Kybernetes*, vol. 26, no. 2, pp. 126–135, 1997.
- [27] M. Mitchell, P. Hraber, and J. P. Crutchfield, "Revisiting the edge of chaos: Evolving cellular automata to perform computations," *arXiv preprint adap-org/9303003*, 1993.
- [28] B. Straatman, R. White, and G. Engelen, "Towards an automatic calibration procedure for constrained cellular automata," *Computers, Environment and Urban Systems*, vol. 28, no. 1, pp. 149–170, 2004.
- [29] W. A. Yasnoff, J. K. Mui, and J. W. Bacus, "Error measures for scene segmentation," *Pattern Recognition*, vol. 9, no. 4, pp. 217–231, 1977.
- [30] Heidari, Hadis, Abdolah Chalechale, and Alireza Ahmadi Mohammadabadi. "Parallel Implementation of Color Based Image Retrieval Using CUDA on the GPU." *International Journal of Information Technology and Computer Science (IJITCS)* 6.1 (2013): 33.
- [31] Goswami, Mr Saptarsi, and Amlan Chakrabarti. "Feature Selection: A Practitioner View." (2014).
- [32] Olutayo, V. A., and A. A. Eludire. "Traffic Accident Analysis Using Decision Trees and Neural Networks." *International Journal of Information Technology and Computer Science (IJITCS)* 6.2 (2014): 22.

Authors' Profiles



Anand Prakash Shukla is born in year 1979. Obtained B.E Degree in Computer Science and Engineering in 2000 and M.Tech in Computer Science and Engineering in 2005 and currently pursuing Ph.D from Computer Science and Engineering department, Motilal Nehru National Institute of Technology, Allahabad and working as Associate Professor in Department of Computer Science and Engineering, Krishna Institute of Engineering and Technology(KIET), Ghaziabad, India. His current research interest includes Cellular Automata, Image Processing and Pattern Recognition.



Suneeta Agarwal received B.Sc. degree in 1973 from University of Allahabad, M.Sc. degree in 1975 from University of Allahabad, Ph.D. in 1980 from IIT Kanpur and M.Tech. degree in 2007 from AAIDU. She is having 31 years of Teaching Experience and currently Professor in the Computer Science and Engineering Department, Motilal Nehru National Institute of Technology, Allahabad. Her current research interest includes Pattern Recognition, Computer Vision, Theory of Computation Science, Algorithms, Automata Theory, Compression, Patten matching, Finger print recognition.

How to cite this paper: Anand Prakash Shukla, Suneeta Agarwal, "Selection of Optimum Rule Set of Two Dimensional Cellular Automata for Some Morphological Operations", International Journal of Information Technology and Computer Science(IJITCS), vol.7, no.12, pp.50-58, 2015. DOI: 10.5815/ijitcs.2015.12.06