2018 International Symposium on Nonlinear Theory and Its Applications,
NOLTA2018, Tarragona, Spain, September 2-6, 2018

Cellular Neural Networks with Two Templates Switched by Local Features of Input Images

Takahisa Ando, Yoko Uwate and Yoshifumi Nishio

Dept. of Electrical and Electronic Engineering, Tokushima University
2-1 Minami-Josanjima, Tokushima 770–8506, Japan
Email: {ando,uwate,nishio}@ee.tokushima-u.ac.jp

Abstract—In 1988, Cellular Neural Networks (CNN) was introduced. A different point from the conventional Neural Network is that CNN has local connectivity property. The performance of CNN depends on the parameters which are called the template. CNN consists of cells which contain analog circuit and each cell connects to only its neighboring cells according to the template. In this study, we focus on the local features of the input image and propose a CNN method of switching two templates. We apply the proposed method to some simulations and investigate its performance.

1. Introduction

In recent years, our life teems with information by growth of high information society. In order to conduct parallel signal and flexible processing like human, Neural Network was proposed. Neural Network is devised based on the nervous system of human. Additionally, Neural Network can perform expression of nonlinear operating characteristics. Then, Cellular Neural Networks (CNN) was introduced by L. O. Chua and L. Yang in 1988 [1]. The idea of CNN was inspired from the architecture of the cellular automata and Neural Network. Different points from Neural Network are that CNN has local connectivity property and its structure resembles the retina. CNN has been successfully used for various high-speed parallel signals processing applications such as image processing, pattern recognition and so on [2]. CNN is composed of the basic analog circuit units called cell and each cell is connected to its neighboring cells according to the template. The performance of CNN depends on the parameters which are called the template. Each cell is influenced and its value is updated by neighboring cells. Additionally, various applications for image processing and pattern recognition of CNN have been reported [3]-[6].

In image processing of CNN, it is difficult to process complex parts of the input image; edge, background, etc. In order to process complex parts of the input image, we have proposed a CNN method of using two templates which have different features [7]. Cell needs to obtain the information from neighboring cells to process complex parts. In the conventional CNN, the information that cell can obtain from its neighboring cells is limited. There is an approach of extending size of the template to increase information from neighboring cells. However, this approach has the potential to collapse the local connectivity property of CNN by uniformly using that template .

In this study, in order to keep the local connectivity property of CNN, two templates are applied to each cell according to the feature of cell in the neighborhood. We focus on the local features of the input image and two templates are switched. In the proposed method, appropriate template is chosen for the cell and two templates are switched by maximum and minimum output values of cell in the neighborhood. We apply the proposed method to some simulations and investigate its performance.

2. Cellular Neural Networks

In this section, we show the conventional CNN. Basic unit circuit of CNN is called cell. The cell consists of linear element and nonlinear element. CNN contains an array in a reticular pattern of many cells. In Fig. 1, we show a two dimensional array composed of M × N identical cells arranged in M rows and N columns.

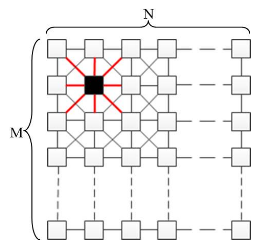


Figure 1: The structure of CNN.

Each cell is coupled to its neighboring cells according to the template. In image processing of CNN, cells correspond to pixels of the input image. State and output equation of cell are described as follows.
State Equation :

$$\frac{dv_{x(ij)}}{dt} = -v_{x(ij)} + \sum_{k=i-r}^{i+r}\sum_{l=j-r}^{j+r} A_{(i,j;k,l)}v_{y(kl)}(t) + \sum_{k=i-r}^{i+r}\sum_{l=j-r}^{j+r} B_{(i,j;k,l)}v_{u(kl)}(t) + T. \quad (1)$$

Output Equation :

$$v_{y(ij)}(t) \;=\; \frac{1}{2}(|v_{x(ij)}(t)+1| - |v_{x(ij)}(t)-1|). \qquad (2)$$

v_x, v_y and v_u are state value, output value and input value.

In Eq. (1), A, B and T are feedback template, feedforward template and threshold. Each cell updates the values by the given template. The output equation is piece-wise linear as shown in Fig. 2.

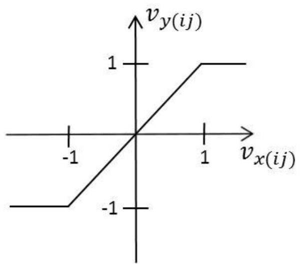


Figure 2: Piece-wise linear function.

3. Proposed Method

In this section, we show the algorithm of the proposed method. We define center cell as the cell which is applied to the template. Each cell has output value. We use two templates which are switched by using the maximum and minimum output values (v_{ymax}: maximum output value of cell, v_{ymin}: minimum output value of cell) in the n×n neighborhood including the center cell. In the neighborhood including the center cell, if cell has v_{ymax} or v_{ymin}, that cell has possibility which represents feature of the input image. In the proposed method, when center cell has that possibility, we use the template which can conduct complex processing. The algorithm of the proposed method are described as follows.

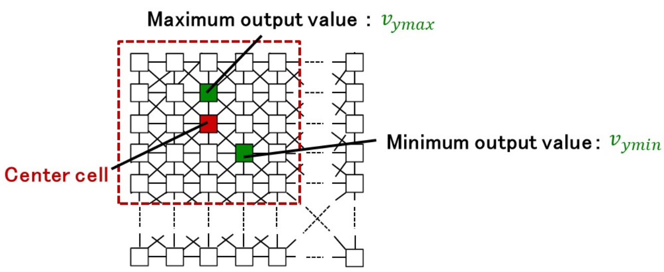


Figure 3: Concept of the proposed method.

Step 1 : First, find the v_{ymax}, and v_{ymin} from the n×n neighborhood including the center cell.

Step 2 : Secondly, determine if the center cell has the maximum or the minimum output value.

Step 3 : Thirdly, in case that the center cell has the maximum or the minimum output value, it is applied 2nd template. In the other case, 1st template is applied to the center cell. Then, the value of the cell is updated by the given template.

State equation with 1st template :

$$\begin{aligned}\frac{dv_{x(ij)}}{dt} &= -v_{x(ij)} + \sum_{k=i-r_1}^{i+r_1}\sum_{l=j-r_1}^{j+r_1} A_1(i,j;k,l)v_{x(kl)}(t)\\ &\quad + \sum_{k=i-r_1}^{i+r_1}\sum_{l=j-r_1}^{j+r_1} B_1(i,j;k,l)v_{u(kl)}(t) + T_1\\ &\quad (|i-k| \le r_1\ ,\ |j-l| \le r_1). \qquad (3)\end{aligned}$$

State equation with 2nd template :

$$\begin{aligned}\frac{dv_{x(ij)}}{dt} &= -v_{x(ij)} + \sum_{k=i-r_2}^{i+r_2}\sum_{l=j-r_2}^{j+r_2} A_2(i,j;k,l)v_{x(kl)}(t)\\ &\quad + \sum_{k=i-r_2}^{i+r_2}\sum_{l=j-r_2}^{j+r_2} B_2(i,j;k,l)v_{u(kl)}(t) + T_2\\ &\quad (|i-k| \le r_2\ ,\ |j-l| \le r_2). \qquad (4)\end{aligned}$$

These steps are applied to all cells and repeated every 0.005 [τ].

4. Simulation Results

In this section, we show some simulation results of the conventional CNN and the proposed method.

4.1. Texture Analysis

In this subsection, we show simulation results of texture analysis and using templates are described as follows [8].

Texture analysis templates :

3×3 *template* (1st template) :

$$A = \begin{bmatrix} 0.86 & 0.94 & 3.75 \\ 2.11 & -2.81 & 3.75 \\ -1.33 & -2.58 & -1.02 \end{bmatrix},$$

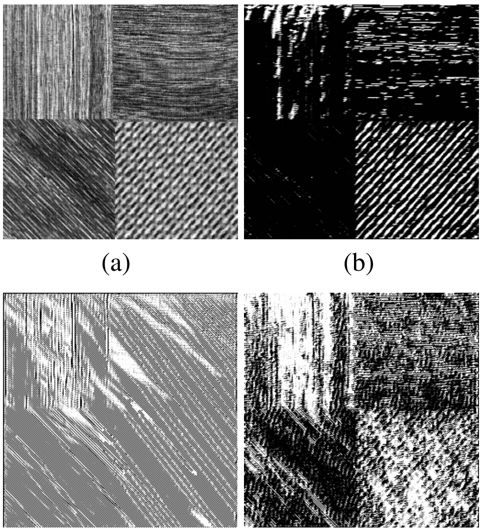
$$B = \begin{bmatrix} 0.16 & -1.56 & 1.25 \\ -2.89 & 1.09 & -3.2 \\ 4.06 & 4.69 & 3.75 \end{bmatrix},\ T = 1.8. \qquad (5)$$

5×5 *template* (2nd template) :

$$A = \begin{bmatrix} 4.21 & -1.56 & 1.56 & 3.36 & 0.62 \\ -2.89 & 4.53 & -0.23 & 3.12 & -2.89 \\ 2.65 & 2.18 & -4.68 & -3.43 & -2.81 \\ 3.98 & 1.56 & -1.17 & -3.12 & -3.2 \\ -3.75 & -2.18 & 3.28 & 2.19 & -0.62 \end{bmatrix},$$

$$B = \begin{bmatrix} 4.06 & -5 & 0.39 & 2.11 & -1.87 \\ 3.9 & 0.31 & -1.95 & 4.84 & -0.31 \\ 0 & -4.06 & 0.93 & -0.31 & 0.46 \\ -0.62 & -5 & 2.34 & 0.62 & -1.87 \\ 3.59 & -0.93 & 0.15 & 2.81 & -1.87 \end{bmatrix},$$

$$T = -5. \tag{6}$$



(a) (b)

(c) (d)

Figure 4: Simulation results 1. (a) Input image. (b) Simulation result of the 3 × 3 CNN. (c) Simulation result of the 5 × 5 CNN. (d) The proposed method ($n = 7$).

Figure 4 shows the input image and simulation results by using only 3×3 template, only 5×5 template, the proposed method. We define 1st template as the "3×3 Texture analysis" template and 2nd template as the "5 × 5 Texture analysis" template. In Fig. 4(b), characterizing the regions of the input image according to the texture content is conducted by using 3 × 3 template. However, the boundary of the regions is defocused. In Fig. 4(c), same characterization can be observed by using 5×5 template. In addition, the boundary of regions of the input image is indistinct. In Fig. 4(d), the proposed method, the boundary of the regions is clear and the texture content is characterized differently. In the proposed method, two templates are switched and applied to cells according to the features of texture contents.

4.2. Edge Detection

In this subsection, we show simulation results of edge detection and using templates are described as follows [8][9].

Edge detection templates :

3×3 *template* (1st template) :

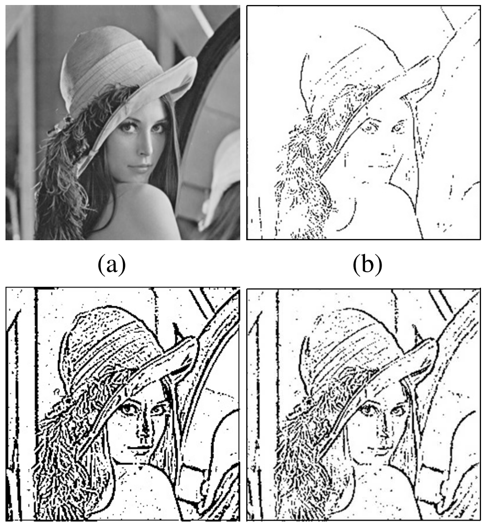
$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

$$B = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}, T = -1. \tag{7}$$

5×5 *template* (2nd template) :

$$A = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix},$$

$$B = \begin{bmatrix} -1 & -3 & -4 & -3 & -1 \\ -3 & 0 & 6 & 0 & -3 \\ -4 & 6 & 20 & 6 & -4 \\ -3 & 0 & 6 & 0 & -3 \\ -1 & -3 & -4 & -3 & -1 \end{bmatrix}, T = -1. \tag{8}$$



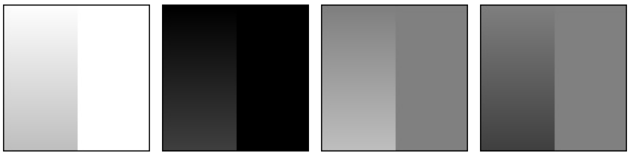
(a) (b)

(c) (d)

Figure 5: Simulation results 2. (a) Input image. (b) Simulation result of the 3 × 3 CNN. (c) Simulation result of the 5 × 5 CNN. (d) The proposed method ($n = 7$).

Figure 5 shows the input image and simulation results by using only 3 × 3 template, only 5 × 5 template, the proposed method. The defocused parts of the input image are the left-side pillar and the woman's silhouette. Therefore, it is difficult to detect the edge lines of these parts clearly. In Fig. 5(b), the defocused parts are not detected by using 3 × 3 template. On the other hand, in Fig. 5(c), the defocused parts are detected by using 5 × 5 template. However, noise effect is observed at woman's hat and silhouette and detected edge lines are thick. In Fig. 5(d), the edge lines of the defocused can be detected clearly and thinly. In addition, noise is removed especially woman's hat and silhouette. Edge detection of using the 3 × 3 template can detect thin edge lines and insusceptible to noise effect. According to switching two templates, thin edge lines are detected and noise effect is reduced compared to the conventional CNN.

We evaluate the performance of the proposed method detection for general image in edge detection. We compare the proposed method and the conventional CNN with four input images which include gradient region. In each input image, boundary of two parts become defocused as it goes top of the image. In Fig. 6, we show input images.



(a) (b) (c) (d)

Figure 6: Input images. (a) Gradient (white to whitish gray) and white. (b) Gradient (black to blackish gray) and black. (c) Gradient (gray to whitish gray) and gray. (d) Gradient (gray to blackish gray) and gray.

We define the range from the bottom to the top of the image boundary is 0[%] to 100[%] and evaluate the proposed method by detected boundary line. In Tab. 1, we show detection rate of each input image.

Table 1: Detection Rate [%].

	Proposed method	3×3 CNN	5×5 CNN
Figure 6(a)	47	25	73
Figure 6(b)	61	34	82
Figure 6(c)	55	34	80
Figure 6(d)	60	34	84

Ideal edge is assumed as a 1-pixel-wide line on the boundary of two regions. From Tab. 1, detection rate by using 5×5 template is high and it can detect boundary line in each case. However, detected edge line is thick line as 2-pixel-wide. On the other hand, we have confirmed that detection rate of the proposed method is improving compared with the performance of 3×3 template.

5. Conclusion

In this study, we applied the proposed method to some simulations and investigate its performance. In the proposed method, we focus on the local features of the input image and two templates were switched by local features of the input image. From the simulation results, we confirmed that the proposed method could process more effective than the conventional CNN. In the future works, we would like to investigate the performance of the proposed method for other simulations and combine to digital processing.

Acknowledgment

This work was partly supported by collaborative research project with Hitachi.

References

[1] L. O. Chua and L. Yang, “Cellular Neural Networks: Theory,” IEEE Trans. Circuits Syst., vol. 35, pp. 1257-1272, Oct. 1988.

[2] Z. H. Yang, Y. Nishio and A. Ushida, “Image processing of twolayer CNNs applications and stability,” IEICE Trans. Fundamentals, vol. E85-A, no. 9, pp. 2052-2060, 2002.

[3] M. Hänggi and G. S. Moschytz, “Cellular Neural Networks Analysis, Design and Optimization.” Kluwer Academic Publishers.

[4] F. Dirk and T. Ronald, “Coding of Binary Image Data using Cellular Neural Networks and Iterative Annealing,” Proc. of ECCTD’03, vol. 1, pp. 229-232, Sep. 2003.

[5] Z. H. Yang, Y. Nishio and A. Ushida, “Image Processing of two-layer CNNs -applications and stability-,” IEICE. Trans. Fundamentals, vol. E85-A, no. 9. pp. 2052-2060, Sep. 2002.

[6] T. Kozek, K. R. Crounse, T. Roska and L. O. Chua, “Smart Image Scanning Algorithms for the CNN Universal Machine,” Proc. of NOLTA’95, vol. 2, pp. 707-712, 1995.

[7] T. Ando, Y. Uwate and Y. Nishio, “Image Processing by Cellular Neural Networks with Switching Two Templates,” Proc. of PrimeAsia’17, pp. 45-48, Oct. 2017.

[8] L. Kek, K.Karacs and T. Roska, “Cellular Wave Computing Library (Templates, Algorithms, and Programs), Version 2.1, ” Cellular Sensory Wave Computers laboratory, Hungarian Academy of Sciencesk, Budapest, Hungary, 2007.

[9] Y. Kato, Y. Uwate and Y. Nishio, “Cellular Neural Networks with Effect from Friend Having Most Different Values and Its Friends,” Proc. of APCCAS’12, pp. 495-498, Dec. 2012.