2018 RISP International Workshop on Nonlinear Circuits,
Communications and Signal Processing (NCSP2018)
Honolulu, Hawaii, USA, March 4-7, 2018

Cellular Neural Networks with Partial Delay Output for Image Processing

Fumiki Sugano, Takahisa Ando, Yoko Uwate and Yoshifumi Nishio

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

Abstract

In the previous study, some researches were reported that the Cellular Neural Networks with Delay Output (DCNN) was applied to the image processing. However, that method has a problem of the noise in image processing. In this study, we propose a CNN method of adding partial delay output for edge detection. This is a new algorithm of adding delay output (DCNN) to the conventional CNN partially. We investigate the performance of the proposed method in edge detection by some input images.

1. Introduction

In recent years, the amount of information is increasing. Therefore, we need to process big data. However, conventional digital computation methods have problem of processing speed. On the other hand, analog computation methods have ability of high speed processing. Therefore, the Neural Networks was proposed. The Neural Networks was based on human's nervous system. The Cellular Neural Networks (CNN) was introduced by Chua and Yang [1]. The idea of the CNN was inspired from the architecture of the Cellular Automata and Neural Networks. The structure of the CNN is cells connected each other and resembles the structure of the animal's retina. Hence, the CNN has been used for various image processing. Performance of the CNN depends on the parameters which called the template. The template is the strength of the connection between the cells.

Delayed Cellular Neural Network (DCNN) has proposed to solve some dynamic image processing and it was applied to the image processing [2]-[4]. In that method, image processing of the DCNN contained past information and it was obtained good results compared to the conventional CNN method. However, it had problem that noise effect is observed in the output image.

In this study, we propose a method of the CNN with partial delay output (DCNN). We investigate the performance of the proposed method in edge detection.

2. Cellular Neural Networks [1]

In this section, we show the structure of the CNN. The basic circuit unit of the CNN is called cell. All cells are connected only to its neighboring cells. We show array composed of $M \times N$ cells arranged in M rows and N columes. The array of the CNN is shown in Fig. 1.

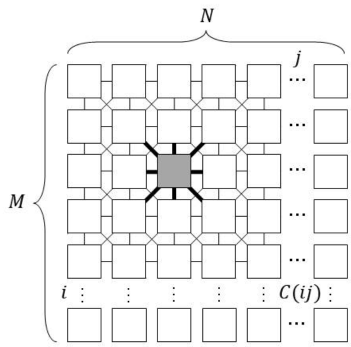

Figure 1: The structure of the CNN.

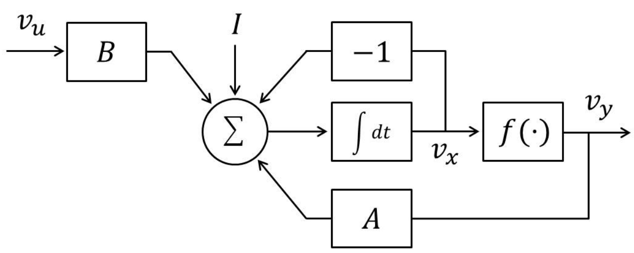

Figure 2: Block diagram of the conventional CNN.

Figure 2 shows the block diagram of the conventional CNN. In image processing of the conventional CNN, two templates are used. Template A is feedback template and template B is control template.

The state equation and output equation are described as follows.

State equation of the conventional CNN :

$$\frac{dv_{xij}}{dt} = -v_{xij} + \sum_{k=i-r}^{i+r} \sum_{l=j-r}^{j+r} A_{(i,j;k,l)} v_{ykl}(t) + \sum_{k=i-r}^{i+r} \sum_{l=j-r}^{j+r} B_{(i,j;k,l)} v_{ukl}(t) + I \quad (1)$$

Output equation of the conventional CNN :

$$v_{yij}(t) = \frac{1}{2}(|v_{xij}(t)+1| - |v_{xij}(t)-1|) \quad (2)$$

v_x, v_y and $v_u$are state value, output value and input value. Figure 3 shows piece-wise linear function. This output equation expressed with such piece-wise linear function. Then, the output value of the CNN is within of +1 to -1.

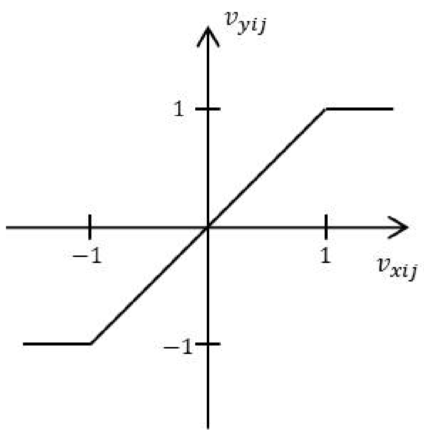

Figure 3: Piece-wise linear function.

3. Delayed Cellular Neural Networks [2][3]

In this section, we show the structure of the DCNN. Figure 4 shows the block diagram of the DCNN. Template D is Delayed feedback template and τ is delay time.

The state equation and output equation are described as follows.

State equation of the DCNN :

$$\frac{dv_{xij}}{dt} = -v_{xij} + \sum_{k=i-r}^{i+r} \sum_{l=j-r}^{j+r} A_{(i,j;k,l)} v_{ykl}(t) + \sum_{k=i-r}^{i+r} \sum_{l=j-r}^{j+r} B_{(i,j;k,l)} v_{ukl}(t) + \sum_{k=i-r}^{i+r} \sum_{l=j-r}^{j+r} D_{(i,j;k,l)} v_{ukl}(t-\tau) \quad (3)$$

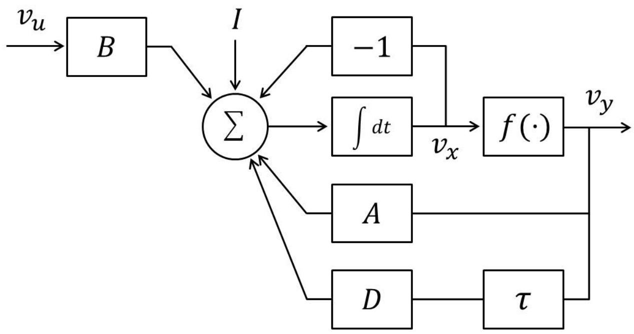

Figure 4: Block diagram of the DCNN.

Output equation of the DCNN :

$$v_{yij}(t) = \frac{1}{2}(|v_{xij}(t)+1| - |v_{xij}(t)-1|) \quad (4)$$

4. Proposed method

In this section, we show the system of the proposed method. The algorithm of the proposed system is shown as follows.

Step 1 : First, calculate difference values between the output value of the center cell and the output values of the neighboring cells. In Fig. 5, we show the concept of calculation and each number is represented the output value of each cell. (①-② , ①-③ , · · · ,①-⑨).

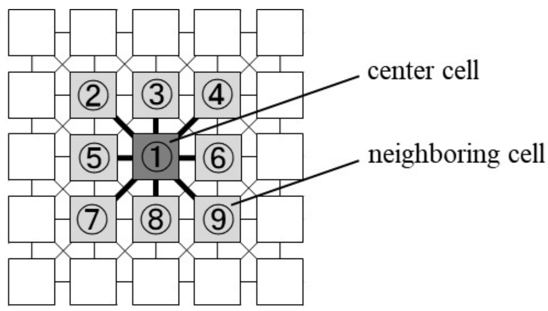

Figure 5: The concept of calculation.

Step 2 : Second, count number of cells which have the difference value smaller than threshold value a. Then, let N be the number of cells satisfying the following equation (5).

$$\begin{cases} |\ \textcircled{1} - \textcircled{2}\ | < a \\ |\ \textcircled{1} - \textcircled{3}\ | < a \\ \quad\quad \vdots \\ |\ \textcircled{1} - \textcircled{9}\ | < a \end{cases} \tag{5}$$

Step 3 : Finally, determine the method according to the following equations. x is arbitrary value.

$$\begin{cases} DCNN : N \geq x \\ CNN : N < x \end{cases} \tag{6}$$

5. Simulation results

In this section, we show simulation results for edge detection by using the proposed method. In this simulation, we use the 3×3 edge detection template and the DCNN edge detection template.

Templates A, B, threshold I are described as follows [4][5].

Edge detection templates of the conventional CNN :

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$
$$B = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}, I = -1. \tag{7}$$

Edge detection templates of the DCNN :

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}, B = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix},$$
$$D = \begin{bmatrix} -0.1 & -0.1 & -0.1 \\ -0.1 & 0.1 & -0.1 \\ -0.1 & -0.1 & -0.1 \end{bmatrix}, I = -1. \tag{8}$$

In Fig. 6, we show the input image and the simulation results of the edge detection. Figure 6(a) shows the input image. In the input image, indistinct parts are the right side of the red pepper and the vegetable in the upper left. In Fig. 6(b), we show the simulation result of the conventional CNN. The conventional CNN cannot detect edge lines of the indistinct parts. In Fig. 6(c), we show the simulation result of the DCNN (τ = 10). The DCNN can detect edge lines of indistinct parts, however, remains the noise effect. In Fig. 6(d), we show the simulation result of the proposed method (τ = 10). The proposed method can detect edge lines of indistinct parts and receive less noise effect compared to the DCNN by applying the DCNN to indistinct parts.

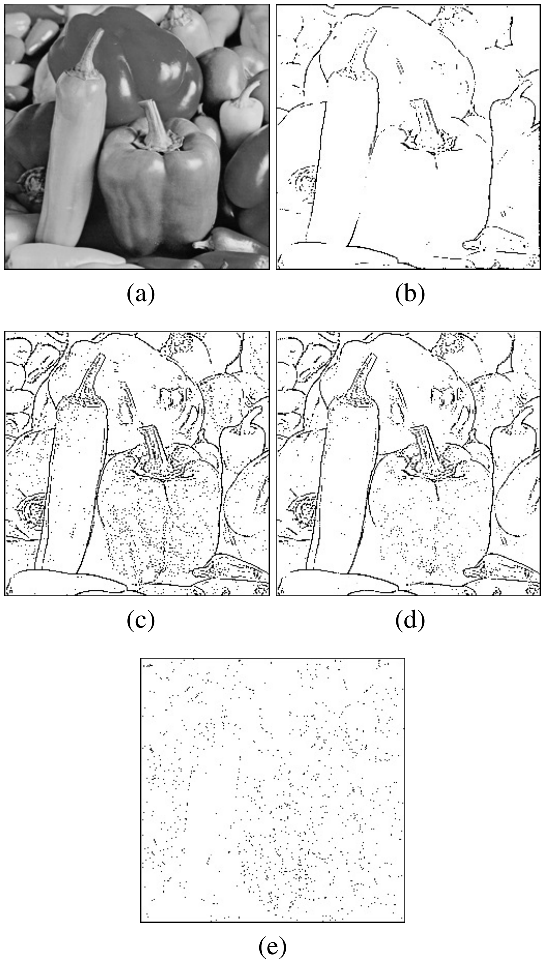

Figure 6: Simulation results 1. (a) Input image. (b) Simulation result of the conventional CNN. (c) Simulation result of the DCNN ($\tau = 10$). (d) Simulation result of the proposed method ($\tau = 10$, $a = 0.003$, $x = 5$). (e) Simulation result of difference (c) and (d).

We apply the proposed method to another input image. In Fig. 7, we show the input image and the simulation results of the edge detection. Figure 7(a) shows the input image. The input image is woman's face. In Fig. 7(b), we show the simulation result of the conventional CNN. The conventional CNN cannot detect edge lines of woman's face. In Fig. 7(c), we show the simulation result of the DCNN (τ = 10). The DCNN can detect edge lines of indistinct parts, however, remains the noise effect. In Fig. 7(d), we show the simulation result of the proposed method (τ = 10). The proposed method can detect edge lines of indistinct parts and receive less noise effect compared to the DCNN.

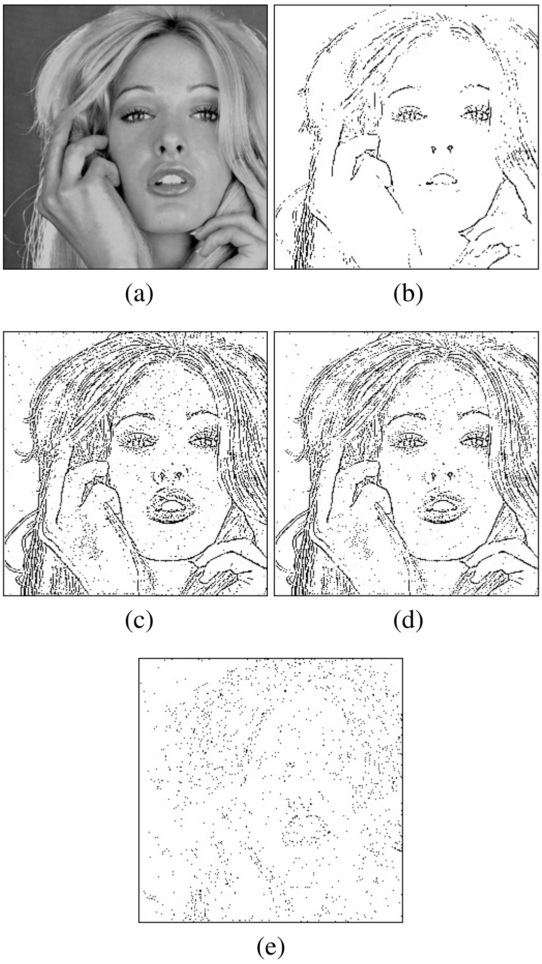

Figure 7: Simulation results 2. (a) Input image. (b) Simulation result of the conventional CNN. (c) Simulation result of the DCNN ($\tau = 10$). (d) Simulation result of the proposed method ($\tau = 10$, $a = 0.003, x = 5$). (e) Simulation result of difference (c) and (d).

6. Conclusion

In this study, we have proposed a CNN method of adding delay output (DCNN) to the conventional CNN partially. From the simulation results, the proposed method can detect edge lines of indistinct parts and decrease effect of noise compared to the DCNN. Therefore, the proposed method is more effective than the conventional CNN and the DCNN in the edge detection. In the future works, we will investigate more effective values a and N.

Acknowledgment

This work was partly supported by JSPS Grant-in-Aid for Scientific Research 16K06357.

References

[1] L. O. Chua and L. Yang, "Cellular Neural Networks Theory." IEEE Trans. Circuits Syst., vol. 35, pp. 1257–1272, 1988.

[2] T. Roska and L. O. Chua, "Cellular Neural Networks with Nonlinear and Delay-type Template Elements," Int. J. Circuit Theory Applications. 20. pp. 463–481, 1992.

[3] N. Takahashi, "On the Complete Stability or Cellular Neural Networks with Delay," NLP Nonlinear Problem, vol. 98, pp. 31–38, 1999.

[4] T. Nakano, Y. Uwate and Y. Nishio. "Edge Detection by Using Switching System with Cellular Neural Networks and Delayed Cellular Neural Networks," SJCIEE, pp. 3, 2015.

[5] 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.