2013 International Workshop on Nonlinear Circuits,
Communications and Signal Processing
NCSP'13, Isrand of Hawaii, Hawaii, USA. March 4-7, 2013

Research of Cellular Neural Networks Inspired by Actual Friendship

Yoshihiro Kato, Yoko Uwate and Yoshifumi Nishio

Tokushima University
2–1 Minami Josanjima, Tokushima, Japan
Phone: +81-88-656-7470, FAX: +81-88-656-7471
E-mail: {kkato, uwate, nishio}@ee.tokushima-u.ac.jp

Abstract

In previous study, we have proposed "Cellular neural networks with effect from friend having most different values and its friends" (DF-CNN). The DF-CNN is the method of constituting a new connection of among cells in CNN. Especially in edge detection, the DF-CNN could detect most edge including indistinct parts. In this study, we research about edge detection by using the DF-CNN. First, we research about how effective does the DF-CNN detect edge of indistinct part compare with the conventional CNN using gradient images. Moreover, we discuss about relationship between function of the DF-CNN and results of edge detection.

1. Introduction

Cellular Neural Networks (CNN) [1] were introduced by inspired from the cellular automata and the neural networks. CNN has local connectivity property and it makes the CNN tailor made for VLSI implementation. Then, the CNN has the features of time continuity, spatial discreteness, nonlinearity and parallel processing capability. Furthermore, the structure of CNN resembles that of animals' retina. Hence, CNN can be used for various image processing applications.

In a previous study, we have proposed "Cellular neural networks with effect from friend having most different values and its friends" (DF-CNN) [2]. The DF-CNN is the method of constituting a new connection of among cells in CNN. In particular, connection of control template is determined by input and output values and connection of feedback template is determined by output values. Then, we confirmed that the DF-CNN could perform edge detection for general image and divide domains. Especially in edge detection, the DF-CNN could detect most edges including indistinct parts.

In this study, we research about edge detection by using the DF-CNN. We research about how effective does the DF-CNN detect edge of indistinct part compare with the conventional CNN. In this comparison, we use gradient images within black to white. Moreover, we discuss about relationship between function of the DF-CNN and result of edge detection.

2. Cellular Neural Networks

The CNN has M by N processing unit circuits called cells. The cell contains linear and nonlinear circuit elements.Cells are arranged in a reticular pattern to M line N row. We represent a cell $C(i, j)$ using a variable i which denotes vertical position and a variable j which denotes horizontal position. Each cell is connected to its neighboring cells according to a template. Usually, the template is the same for all cells except for boundary cells. The state equation and the output equation of the CNN are shown as follows.

State equation :

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

Output equation :

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

where v_x, v_y and v_u represent a state, an output and an input of cell, respectively. In the Eq. (1), A is the feedback template and B is the control template. These and the constant bias I are collectively called general template. The output equation is a piece-wise linear function. In fact, the output value of CNN is within of -1 to 1. When the CNN is used for image processing, values of black and white are treated as 1 and -1, respectively. r is a positive integer number. Each cell has $(2r + 1)^2$ neighborhood cells. Hence, the size of template is defined by r. If r is 1, the size of template is 3×3 matrix. Additionally, if r is 2, the size of template is 5×5 matrix.

3. Cellular Neural Networks with Effect from Friend Having Most Different Values and Its Friends

In this section, we explain the algorithm of the cellular neural networks with effect from friend having most different values and its friends (DF-CNN). The DF-CNN is the method of constituting new combinations of among cells. In this paper, we consider the local connection of CNN as friendship. Then, in each cell, the neighboring cells are defined as friend cells. Based on this, the algorithm of DF-CNN is described as follows.

3.1 In the Case of Feedback Template

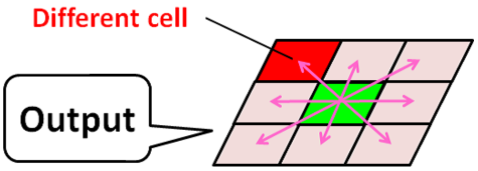

Figure 1: Search the different cell in output image.

Step 1: In each cell of output image, a cell with the most different of the value of center cell $C(i, j)$ is defined as “Different cell” from eight neighboring cells like Fig. 1.

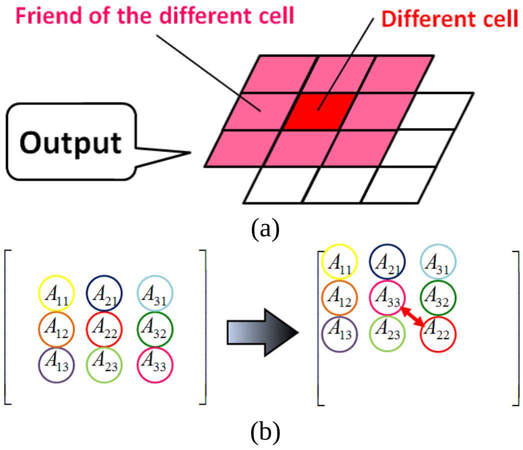
Friend of the different cell
Different cell
Output
(a)
A_{11} A_{21} A_{31}
A_{12} A_{22} A_{32}
A_{13} A_{23} A_{33}
A_{11} A_{21} A_{31}
A_{12} A_{33} A_{32}
A_{13} A_{23} A_{22}
(b)

Figure 2: New combination of feedback template. (a) The combination on centering on the different cell. (b) Elements of new feedback template.

Step 2: The combination centering on a cell $C(i, j)$ is changed into a combination centering on the different cell like Fig. 2(a).

Moreover, the element of the feedback template of the different friend and the center cell are replaced. With this, a new combination is constituted like Fig. 2(b).

Step 3: The value of each cell is updated according to the Eqs. (1) and (2) using the new combinations of feedback templates.

Step 4: Steps 1 to 4 are repeated every 0.005 [τ].

3.2 In the Case of Control Template

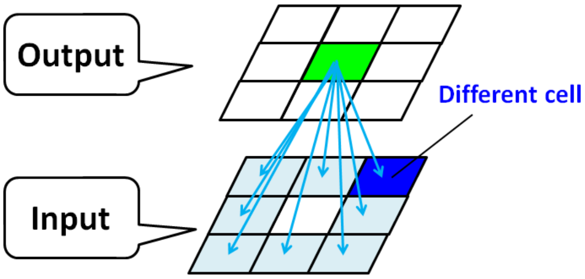

Figure 3: Search the different cell in input image.

Step 1: In each cell of input image, a cell with the most different of the value of center cell $C(i, j)$ in output image is defined as “Different cell” from eight neighboring cells in input image like Fig. 3.

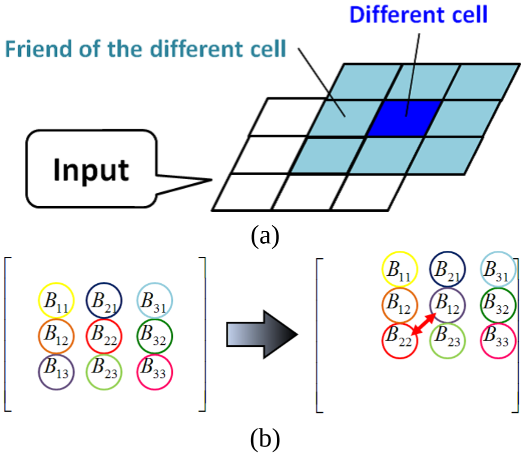
Different cell
Friend of the different cell
Input
(a)
B_{11} B_{21} B_{31}
B_{12} B_{22} B_{32}
B_{13} B_{23} B_{33}
B_{11} B_{21} B_{31}
B_{12} B_{12} B_{32}
B_{22} B_{23} B_{33}
(b)

Figure 4: New combination of control template. (a) The combination on centering on the different cell. (b) Elements of new control template.

Step 2: The combination centering on a cell $C(i, j)$ is changed into a combination centering on the different cell like Fig. 4(a).

Moreover, the element of the control template of the different friend and the center cell are replaced. With this, a new combination is constituted like Fig. 4(b).

Step 3: The value of each cell is updated according to the Eqs. (1) and (2) using the new combinations of control templates.

Step 4: Steps 1 to 4 are repeated every 0.005 [τ].

Therefore, the value of each cell is updated by receiving influences from the different cell and its neighboring cells.

4. Evaluation of Edge Detection

In the previous study, we have confirmed that the DF-CNN could perform edge detection for general image. The DF-CNN could detect most edge including indistinct parts. In this section, we evaluate performance of the DF-CNN for edge detection. We compare the DF-CNN and two types of the conventional CNNs with "*Edgedetection*" templates of 3×3 and 5×5 matrices. In this evaluation, we call the conventional CNN with "*Edgedetection*" template of 3×3 matrix and the conventional CNN with "*Edgedetection*" template of 3×3 matrix as "*Type A* (3×3)" and "*Type B* (5×5)", respectively. Input images include gradient and fixed values parts which are shown Figs. 5(a) to (d). In every image, boundary of two parts become indistinct as it goes top of the image.

Figure 6 shows simulation results of using the "*Type A* (3× 3)". In any cases, edge of indistinct boundary is difficult to detected. However, detected edge is fine line as 1-pixel-wide. Figure 7 shows simulation results of using the "*Type B* (5 × 5)". The boundary of two parts is detected better than using the "*Type A* (3×3)". However, detected edge is bold line as 2-pixel-wide. We take this reason as that un-boundary cells are detected as edge by using "*Edgedetection*" template of 5×5 matrix. On the other hand, Fig. 8 shows simulation results of using the DF-CNN. We can see that the DF-CNN detect indistinct portion better than the "*Type A* (3 × 3)" and the detected edge is fine line.

Table 1: Detection Rate [%]

	DF-CNN	Type A (3×3)	Type B (5×5)
Figure 5(a)	56.25	25.00	73.34
Figure 5(b)	67.18	33.59	82.03
Figure 5(c)	64.84	33.59	80.47
Figure 5(d)	69.53	33.59	83.59

Table 1 shows the detection rate of each input image using the DF-CNN, the "*Type A* (3 × 3)" and the "*Type B* (5 × 5)". Ideal edge is assumed as a 1-pixel-wide line on the boundary of two domains. And we evaluate each types by the ideal edge as 100 [%] of baseline. From Tab. 1, the "*Type B* (5 × 5)" has detection rate by it can detect indistinct portion in every cases. However in fact, detected edge is bold line as 2-pixel-wide. Then, the detection rate will decrease considering this point. On the other hand, we can say that the detection rates of the DF-CNN are improving about two times compared with the "*Type A* (3 × 3)".

5. Change the Rank of Determine Different Cell

In the proposed algorithm, a cell with the most different of the value of center cell $C(i, j)$ is defined as different cell. In this section, we investigate output characteristics for the cases of other cell is defined as the different cell like a cell with 2nd to 8th different values, respectively.

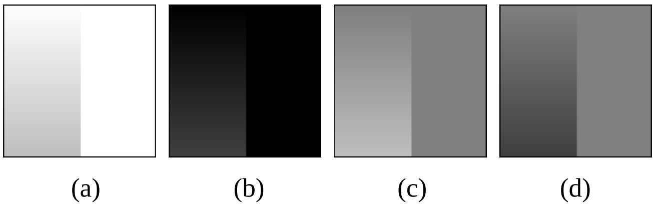

Figure 5: Input images. (a) Gradient (white to whitish gray) and white. (b) Gradient (black to blackish gray) and black. (c) Gradient (gray to whitish gray) and gray. (d) Gradient (gray to blackish gray) and gray.

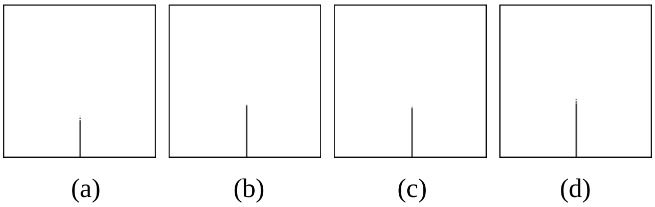

Figure 6: Simulation results using the "*Type A* (3 × 3)". (a) Output image for Fig. 5(a). (b) Output image for Fig. 5(b). (c) Output image for Fig. 5(c). (d) Output image for Fig. 5(d).

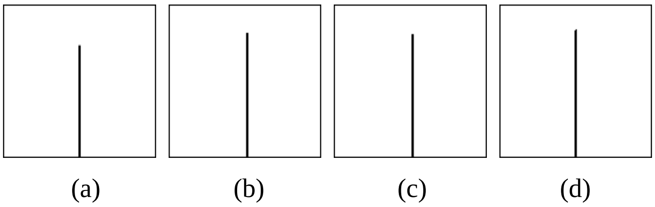

Figure 7: Simulation results using the "*Type B* (5 × 5)". (a) Output image for Fig. 5(a). (b) Output image for Fig. 5(b). (c) Output image for Fig. 5(c). (d) Output image for Fig. 5(d).

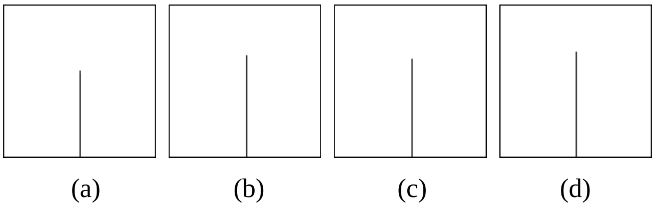

Figure 8: Simulation results using the DF-CNN. (a) Output image for Fig. 5(a). (b) Output image for Fig. 5(b). (c) Output image for Fig. 5(c). (d) Output image for Fig. 5(d).

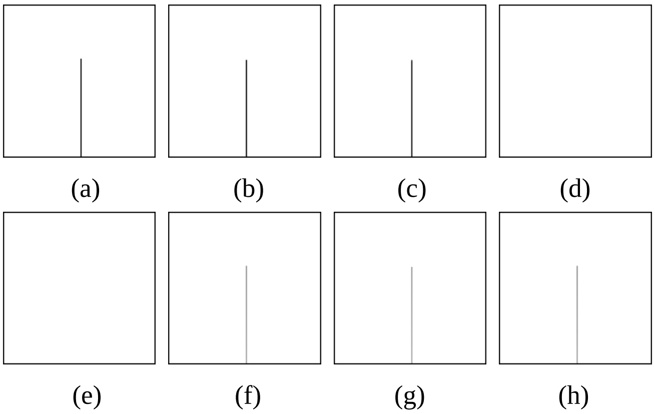

Figure 9: Simulation results for Fig. 5(c). (a) Case of the most different cell. (b) Case of 2nd different cell. (c) Case of 3rd different cell. (d) Case of 4th different cell. (e) Case of 5th different cell. (f) Case of 6th different cell. (g) Case of 7th different cell. (h) Case of 8th different cell.

Figure 9 shows simulation results of using Fig. 5(c) as input image. In Figs. 9(a) to (c), there is not much difference between output images by cases of most, 2nd and 3rd different cells, respectively. This reason is assumed be due to that values of the most to 3rd different cells are same side because of two parts are divided in to right and left side in Fig. 5(c). On the other hand, the boundary is not detected in Figs. 9(d) and (e). However, the boundary is detected in Figs. 9(f) to (h) although value of cells are not converged.

Case of using Fig. 5(c) as input image has image dependence. Then, we use general image (Fig. 10(a)) as input image. Figures 10(b) to (i) show simulation results. Figure 10(b) shows output image by using the DF-CNN that defines a cell with the most different value as the different cell. Most edge is detected and value of cell is converged. Figures 10(c) to (i) show output images by cases of other cell is defined as the different cell, respectively. In every cases, each cell can obtain informations from the 2-neighborhood. Thus, boundary of indistinct portion is estimated at edge. However, convergence of cell is difficult as the different cell is chosen from nearest value of cell. Even if all un-converged cell converged, some noises are left and detected edge is bold line.

From these investigations, we can say that the DF-CNN choose a cell with most different value is the method which effectively obtain the value from the 2-neighborhood.

6. Conclusions

In this study, we have researched about edge detection by using the DF-CNN. First, we investigated about how effective does the DF-CNN detect edge of indistinct part compare with the conventional CNN using gradient images. Moreover, we investigated output characteristics for the cases of other cell is defined as the different cell like a cell with the 2nd to 8th different value, respectively. From these investigations, we could say that the DF-CNN choose a cell with most different value is the method which effectively obtain the value from the 2-neighborhood. In the future work, we would like to try another determining condition of the “Different cell”.

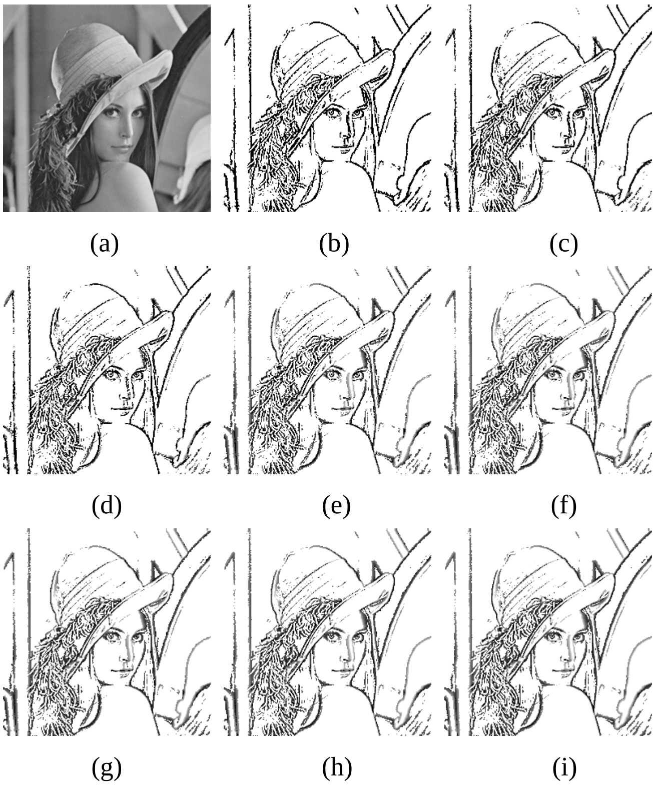

Figure 10: Simulation results for general image. (a) Input image. (b) Case of the most different cell. (c) Case of 2nd different cell. (d) Case of 3rd different cell. (e) Case of 4th different cell. (f) Case of 5th different cell. (g) Case of 6th different cell. (h) Case of 7th different cell. (i) Case of 8th different cell.

Acknowledgment

This work was partly supported by JSPS Grant-in-Aid for Young Scientists 23700269.

References

[1] L. O. Chua and L. Yang, “Cellular Neural Networks: Theory,” *IEEE Trans. Circuits Syst.*, vol. 32, pp. 1257-1272, Oct. 1988.

[2] 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.