

Image Processing by Cellular Neural Networks with Reinforcement Learning

Kazushige Natsuno, Yoko Uwate and Yoshifumi Nishio
Dept. of Electrical and Electronic Engineering, Tokushima University,
2-1 Minami-Josanjima, Tokushima, 770-8506 JAPAN
Email: {natsushige, uwate, nishio}@ee.tokushima-u.ac.jp

Abstract—Generally, performance of Cellular Neural Networks depend on template. Additionally, the CNN can process varied applications by the template. However, performance of the CNN is restricted to an accuracy of the processing by a design of the template. In this study, we propose Cellular Neural Networks with Reinforcement Learning (RL-CNN). In the RL-CNN, the templates are updated by comparison between center and neighboring output values. From simulation results, we confirm that the proposed method is effective for image processing.

I. INTRODUCTION

Our society is called advanced information society. Because of this, a processing method demanded high speed processing for large amount of data. However, general digital computation methods have problem of processing speed by sequential dynamics. On the other hand, Neural Networks (NN) were proposed based on the human's nervous system. The concept of neural networks is neurobiology and adapted to integrated circuits. The key features of neural networks are asynchronous parallel processing, continuous-time dynamics and global interaction of network elements.

Cellular Neural Networks (CNN) were introduced by Chua and Yang in 1988 [1]. The idea of CNN was inspired from the architecture of the Cellular Automata and Neural Networks. The structure of CNN is constructed from simple analog circuit. The circuit is called a cell that has an effect on each other. Each cell are interacted with each other by the parameter is called template. Performance of CNN depends on template. Moreover, the CNN can be varied processing by template. However, a design of template is spatially and temporal uniform. Furthermore, each cell are put a various states by values of neighboring cells. From the feature of the CNN, performance of the CNN is restricted to an accuracy of the processing by a design of the template. Therefore, we focus on degree of similarity input values in neighboring cells.

In this study, we propose the new system of CNN with Reinforcement Learning. About the concept of Reinforcement Learning (RL), a type of machine learning and an algorithm for adapt to the system by reward [2]. In the RL-CNN, the template are updated by RL. The template of all cells are updated at every iterations by the center and neighboring output values. Also, updating of templates are decided by some parameter which are the output values, the reduction rate and learning rate. From simulation results, we confirm that the

RL-CNN is effective for image processing.

II. CELLULAR NEURAL NETWORKS

In this section, we explain about the structure of the conventional CNN. The conventional CNN has M by N processing unit circuits called cells. The structure of cell circuit is simple analog circuit. The cell circuit is constructed from linear capacitor, linear resistors, independent voltage source, linear and nonlinear controlled sources. The conventional CNN has the features of time continuity, spatial discreteness, nonlinearity and parallel processing capability. Additionally, the local connectivity property makes the CNN tailor made for VLSI implementation. The structure of CNN resembles that of animals' retina. Therefore, CNN can be used for various image processing applications [3]-[8]. The state equation and output equation are described as follows.

State equation :

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

Output equation :

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

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 expressed with a piece-wise linear function. Therefore, the output value of the conventional CNN is within of +1 to -1. When the conventional CNN is used for image processing, black and white are treated as +1 and -1, respectively.

III. CNN WITH REINFORCEMENT LEARNING

In this section, we explain the algorithm of the Cellular Neural Networks with Reinforcement Learning (RL-CNN). In the RL-CNN, templates of each cell are updated at every iterations by RL. The learning steps in the RL-CNN are

described as follows.

Step 1 : State and output values are updated according to Eqs. (1) and (2).

Step 2 : The output values of the center cell and neighbor cells are compared. The comparison equation for the $cell(i, j)$ is described Eq. (3).

Comparison equation:

$$D(i, j; k, l) = |v_y^{center}(i, j) - v_y^{neighbor}(k, l)|. \quad (3)$$

Step 3 : Among the eight calculated values of $D(i, j; k, l)$, the difference values of comparison between center and neighbor cells are ranked based on the small value, respectively. In ordered eight cells, the neighbor cells with the smallest and the most different values are defined as “RL-CNN(S)” and “RL-CNN(D)”, respectively.

Step 4 : Templates of RL-CNN(S) and RL-CNN(d) are updated according to Eqs. (4) and (5).

TD-error :

$$TD - error = \{r_t + \gamma V(x_t + 1)\} - V(x_t). \quad (4)$$

Updated equation :

$$V(x_t) \leftarrow V(x_t) + \alpha [TD - error]. \quad (5)$$

where $a(i, j; k, l)$, α and γ represent the template, a learning rate ($0 \leq \alpha \leq 1$) and a reduction rate ($0 \leq \gamma \leq 1$), respectively.

Step 5 : Values of each cell are updated according to Eqs. (6) and (7).

State equation :

$$\begin{aligned} \frac{dv_{xij}}{dt} = & -v_{xij} + \sum_{k=i-r}^{i+r} \sum_{l=j-r}^{j+r} A_{(i,j;k,l)}^{update} v_{ykl}(t) \\ & + \sum_{k=i-r}^{i+r} \sum_{l=j-r}^{j+r} B_{(i,j;k,l)}^{update} v_{ukl}(t) + I \\ & (|i-k| \leq 1, |j-l| \leq 1). \end{aligned} \quad (6)$$

Output equation :

$$v_{yij}(t) = \frac{1}{2} (|v_{xij}(t) + 1| - |v_{xij}(t) - 1|), \quad (7)$$

Step 6 : Step 1 to Step 5 are repeated every 0.005 [τ].

In the RL-CNN, the template of each cell are changed for based on condition that are decided by ourself. In this study, we focus on degree of similarity output values in neighboring cells. Therefore, the values of each cell are updated by receiving influenced from the center and its neighboring cells.

IV. SIMULATION RESULTS

In this section, we show simulation results for edge detection by using the RL-CNN. In the edge detection, when the value of the center cell is close to that values of neighborhoods cell, the edge of indistinct portion is not detected. In the simulation, the parameters of the learning rate and the reduction rate are set to $\alpha = 0.005$ and $\gamma = 1$, respectively. Also, we use templates which are found in [9]-[10].

Edge detection template:

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}, B = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}, I = -1. \quad (8)$$

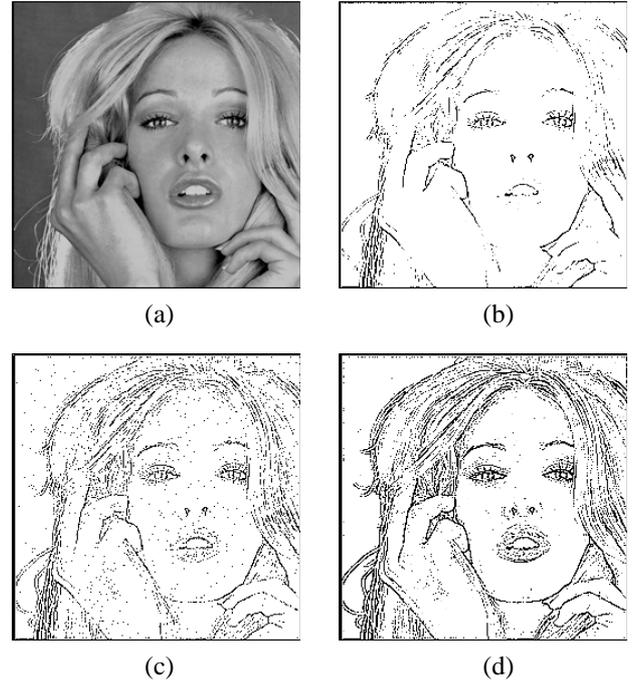


Fig. 1. Simulation results 1. (a) Input and initial state images, (b) Output image of using conventional CNN, (c) Output image of using RL-CNN(S), (d) Output image of using RL-CNN(D).

Figure 1 shows the simulation results by using RL-CNN. In Fig. 1(a), the outline of face is the indistinct portions. Figure 1(b) shows the simulation result by the conventional CNN. Figures 1(c) and (d) show the simulation results by using RL-CNN. In the Fig. 1(b), the edge of the indistinct portion cannot be mostly detected. On the other hand, in the case of using RL-CNN(S), the edge of the indistinct portion can be detected in Fig. 1(c). However, Fig 1(c) includes noises. In contrast, in the case of using RL-CNN(D), the edge of the indistinct portion can be more clearly detected and some noises are removed in Fig. 2.

Figure 2 shows the simulation results for another input image. In Fig. 2(a), the almost edges are the indistinct portions. Figure 2(b) shows the simulation result by the conventional

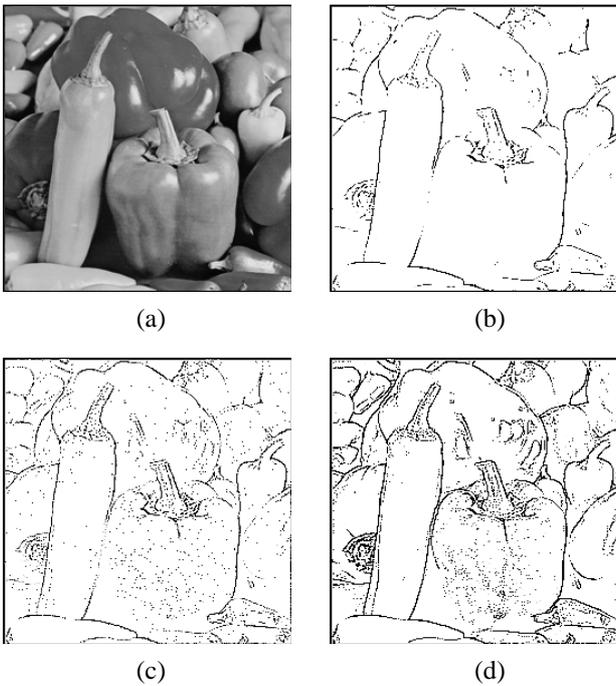


Fig. 2. Simulation results 2. (a) Input and initial state images, (b) Output image of using conventional CNN, (c) Output image of using RL-CNN(S), (d) Output image of using RL-CNN(D).

CNN. Figures 2(c) and (d) show the simulation results by using RL-CNN. In Fig. 2(b), the edge of the indistinct portion cannot be mostly detected. On the other hand, in Fig. 2(c), in the case of RL-CNN(S), the edge of the indistinct portion cannot be detected. In contrast, in the case of RL-CNN(D), the edge of the indistinct portion can be clearly detected in Fig. 2(d).

From these results, we consider that the RL-CNN(D) is more effective than the conventional CNN for the edge detection.

V. CONCLUSION

In this study, we have proposed the Cellular Neural Networks with Reinforcement Learning. The RL-CNN has a feature of changing template by cell values by center and neighboring cell values. From simulation results, we consider that the RL-CNN(D) is more effective for edge detection. In the future works, we would like to investigate the RL-CNN for a lot of applications for image processing.

ACKNOWLEDGMENT

This work was partly supported by JSPS Grant-in-Aid for Challenging Exploratory Research 26540127.

REFERENCES

- [1] L. O. Chua and L. Yang, "Cellular Neural Networks: Theory," IEEE Trans. Circuits Syst., vol. 35, pp. 1257-1272, Oct. 1988.
- [2] L. P. Kaelbling, M. L. Littman, A. W. Moore, "Reinforcement Learning: A Survey" Journal of Artificial Intelligence Research, vol. 4, pp. 237-285, 1996.
- [3] 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.
- [4] M. Namba and Z. Zhang, "Cellular Neural Network for Associative Memory and its Application to Braille Image Recognition," Proc. of IJCNN'06, pp. 4716-4721, Jul. 2006.
- [5] H. Koepl and L. O. Chua, "An Adaptive Cellular Nonlinear Network and its Application," Proc. of NOLTA'07, pp. 15-18, Sep. 2007.
- [6] T. Kozek, K. R. Crouse, 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] K. R. Crouse and L. O. Chua, "Methods for Image Processing and Pattern Formation in Cellular Neural Networks: A Tutorial," IEEE Trans. Circuits Syst., vol. 42, no. 10, pp. 583-601, Oct. 1995.
- [8] K. R. Crouse, L. O. Chua, P. Thiran and G. Setti, "Characterization and Dynamics of Pattern Formation in Cellular Neural Networks," International Journal of Bifurcation and Chaos, vol. 6, no. 9, pp. 1703-1724, 1996.
- [9] Cellular Sensory Wave Computers Laboratory Computer and Automation Research Institute Hungarian Academy of Sciences, "Cellular wave Computing Library(Template, Algorithms, and Programs) Version 2. 1"
- [10] T. Nishio and Y. Nishio, "Periodic Pattern Formation and its Applications in Cellular Neural Networks" IEEE Transactions on Circuits and Systems I, vol. 55, no. 9, pp. 2736-2742, Oct. 2008.