



Simple Template Design Method of Cellular Neural Networks by Combining Templates

Takashi Inoue[†] and Yoshifumi Nishio[†]

[†]Tokushima University
2-1 Minami-Josanjima, Tokushima, Japan
Phone:+81-88-656-7470, FAX:+81-88-656-7471
Email: : {takashi, nishio}@ee.tokushima-u.ac.jp

Abstract

In this study, we propose a simple template design method of cellular neural networks by combining two existing templates. In this article, we investigate the output characteristics of simple image processings using the proposed template design method and show its effectiveness with some simulation results.

1. Introduction

Cellular Neural Networks (CNN) [1] were introduced by Chua and Yang in 1988. The idea of the CNN was inspired from the architecture of the cellular automata and the neural networks. Unlike the conventional neural networks, the CNN has local connectivity property. Since the structure of the CNN resembles the structure of animals' retina, the CNN can be used for various image processing application [2]-[4] including character extractions [5].

Wiring weights of the cells are established by parameters called the template. The performance of the CNN is decided by the template. If the templates of all the cells in the CNN are identical, the system is called space-invariant, while if the templates of all the cells in the CNN are not identical, the system is called space-varying. Typically, space-invariant system is mainly used in the studies of the CNN, because the implementation cost is cheap. There are some major template design methods such as template learning [6]. However, the template leaning takes a long time to find desired template.

In the previous study, we have proposed the cellular neural networks using mixture template [7]. In the first mixing method, we prepared two different CNN templates and chose one of them for each cell at random. In the second mixing method, we prepared two different CNN templates, but chose some elements of them for the matrices and the threshold I at random. As the first step of the investigation of the mixing two templates, we could confirm the effects of the both templates in the proposed two mixing methods. In order to broaden this concept, we should consider another type of

mixture template using two or more different CNN templates.

In this study, we propose a simple template design method of CNN by combining two existing templates. We investigate the output characteristics of simple image processing using the proposed template design method and show its effectiveness with some simulation results. Because the produced templates have the effects of the two original templates, it may be useful to reduce the number of the successive processing for some kinds of image processing tasks.

In the Sec. 2, we review the basic of the standard CNN. In the Sec. 3, we show the algorithm of the propose template design method. In the Sec. 4, we show some computer simulated results. Section 5 concludes the article.

2. Cellular Neural Networks [1]

In this section, we describe the basic structure of the CNN. The CNN has M by N processing unit circuits called cells as Fig. 1. 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. The cell contains linear and nonlinear circuit elements. The CNN is an array of cells. 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 CNN has the features of time continuity, spatial discreteness, nonlinearity and parallel processing capability.

The state equation and the output equation of the cell are shown as follows.

State equation:

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

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 equation (1), A is the feedback template and B is the control template. These and bias I are collectively called general template.

The r -neighborhood of the cell $C(i, j)$ in the CNN is defined by

$$Nr(i, j) = \{C(k, l) \mid \max\{|k - i|, |l - j|\} \leq r, \\ 1 \leq k \leq M; 1 \leq l \leq N\} \quad (3)$$

where r is a positive integer number. In our study, we fix the value of r as 1.

3. Template Design Method

In this section, we propose a template design method of the CNN.

In our template design method, we simply prepare two kinds of 3×3 different templates;

Template 1 :

$$A_1 = \begin{bmatrix} a_{11}^1 & a_{12}^1 & a_{13}^1 \\ a_{21}^1 & a_{22}^1 & a_{23}^1 \\ a_{31}^1 & a_{32}^1 & a_{33}^1 \end{bmatrix}, \quad B_1 = \begin{bmatrix} b_{11}^1 & b_{12}^1 & b_{13}^1 \\ b_{21}^1 & b_{22}^1 & b_{23}^1 \\ b_{31}^1 & b_{32}^1 & b_{33}^1 \end{bmatrix}, \quad I_1. \quad (4)$$

Template 2 :

$$A_2 = \begin{bmatrix} a_{11}^2 & a_{12}^2 & a_{13}^2 \\ a_{21}^2 & a_{22}^2 & a_{23}^2 \\ a_{31}^2 & a_{32}^2 & a_{33}^2 \end{bmatrix}, \quad B_2 = \begin{bmatrix} b_{11}^2 & b_{12}^2 & b_{13}^2 \\ b_{21}^2 & b_{22}^2 & b_{23}^2 \\ b_{31}^2 & b_{32}^2 & b_{33}^2 \end{bmatrix}, \quad I_2. \quad (5)$$

Next, we choose the mixing rate;

Mixing Rate:

$$p \quad (0 \leq p \leq 1.0) \quad (6)$$

Finally, we create a new 5×5 combination template from the two 3×3 templates with the mixing rate;

Combination Template:

$$A = \begin{bmatrix} pa_{11}^1 & 0 & pa_{12}^1 & 0 & pa_{13}^1 \\ 0 & (1-p)a_{11}^2 & (1-p)a_{12}^2 & (1-p)a_{13}^2 & 0 \\ pa_{21}^1 & (1-p)a_{21}^2 & pa_{22}^1 + (1-p)a_{22}^2 & (1-p)a_{23}^2 & pa_{23}^1 \\ 0 & (1-p)a_{31}^2 & (1-p)a_{32}^2 & (1-p)a_{33}^2 & 0 \\ pa_{32}^1 & 0 & pa_{32}^1 & 0 & pa_{33}^1 \end{bmatrix},$$

$$B = \begin{bmatrix} pb_{11}^1 & 0 & pb_{12}^1 & 0 & pb_{13}^1 \\ 0 & (1-p)b_{11}^2 & (1-p)b_{12}^2 & (1-p)b_{13}^2 & 0 \\ pb_{21}^1 & (1-p)b_{21}^2 & pb_{22}^1 + (1-p)b_{22}^2 & (1-p)b_{23}^2 & pb_{23}^1 \\ 0 & (1-p)b_{31}^2 & (1-p)b_{32}^2 & (1-p)b_{33}^2 & 0 \\ pb_{32}^1 & 0 & pb_{32}^1 & 0 & pb_{33}^1 \end{bmatrix},$$

$$I = pI_1 + (1-p)I_2. \quad (7)$$

The schematic picture of the procedure is shown in Fig. 1. In Fig. 1(c), the white elements are 0.

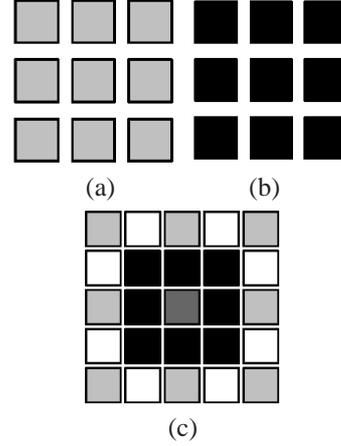


Figure 1: Combining two templates. (a) Template 1 (3×3). (b) Template 2 (3×3). (c) Combination template (5×5).

The following is an example of the procedure.

Template 1 :

$$A_1 = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{bmatrix}, \quad B_1 = \begin{bmatrix} 1 & 2 & 1 \\ 2 & 0 & 2 \\ 1 & 2 & 1 \end{bmatrix}, \quad I_1 = 0. \quad (8)$$

Template 2:

$$A_2 = \begin{bmatrix} 0 & 0.5 & 0 \\ 0.5 & 4 & 0.5 \\ 0 & 0.5 & 0 \end{bmatrix}, \quad B_2 = \begin{bmatrix} 0.2 & 0 & 0.2 \\ 0 & 1 & 0 \\ 0.2 & 0 & 0.2 \end{bmatrix},$$

$$I_2 = 4.5. \quad (9)$$

Mixing Rate:

$$p = 0.8. \quad (10)$$

Combination Template:

$$A = \begin{bmatrix} 0.8 & 0 & 0.8 & 0 & 0.8 \\ 0 & 0 & 0.1 & 0 & 0 \\ 0.8 & 0.1 & 2.4 & 0.1 & 0.8 \\ 0 & 0 & 0.1 & 0 & 0 \\ 0.8 & 0 & 0.8 & 0 & 0.8 \end{bmatrix},$$

$$B = \begin{bmatrix} 0.8 & 0 & 1.6 & 0 & 0.8 \\ 0 & 0.04 & 0 & 0.04 & 0 \\ 1.6 & 0 & 0.2 & 0 & 1.6 \\ 0 & 0.04 & 0 & 0.04 & 0 \\ 0.8 & 0 & 1.6 & 0 & 0.8 \end{bmatrix},$$

$$I = 0.9. \quad (11)$$

4. Simulation results

In this section, we show some simulation results. All the original templates in this section are found in [8].

4.1. Noise Reduction

First, the effective noise reduction results using combination template are shown. The original templates are “Small Object Remover” template and “Patch Maker” template. We set the “Small Object Remover” template as Template 1 and the “Patch Maker” template as Template 2.

Small Object Remover (Template 1) :

$$A_1 = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{bmatrix}, B_1 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, I_1 = -1. \quad (12)$$

By using this template, large objects remain and small objects are removed from input binary images.

Patch Maker (Template 2):

$$A_2 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 2 & 1 \\ 0 & 1 & 0 \end{bmatrix}, B_2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}, I_2 = 4.5. \quad (13)$$

By using this template, black objects are expanded without limit.

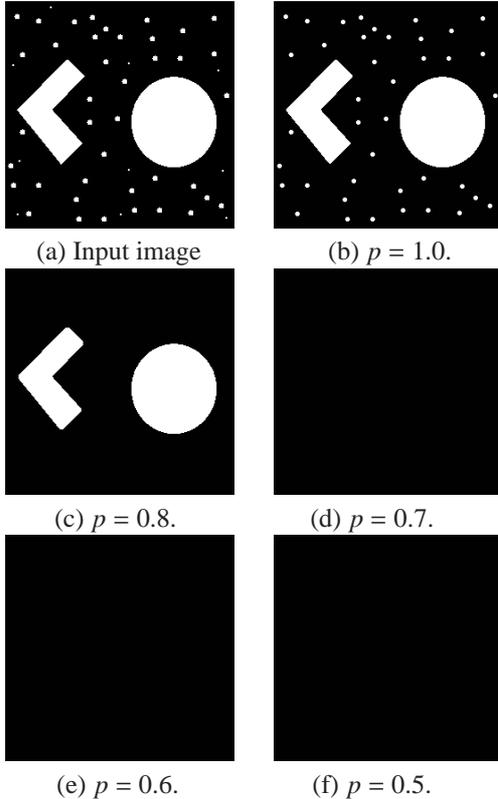


Figure 2: Simulation results using combination template of Small Object Remover Template (Template 1) and Patch Maker Template (Template 2) with different mixing rates.

Figure 2 shows the simulation results using combination template. Figure 2(b) shows the simulation results using only “Small Object Remover” template for an input image with various size of spot noises. We can see that the “Small Object Remover” template cannot remove relatively large spot noises at all. Figures 2(c), (d), (e) and (f) show simulation results using combination template with different mixing rates. In Fig. 2(c), all noises removed and the two large objects are little distorted. However, when the ratio of the “Patch Maker” template is larger than 30%, the output image becomes completely black.

Remark: In this simulation, the mixing rate $p=1.0$ gives almost similar identical output image to the case that the standard CNN with the “Small Object Remover” template is used. However, the output is not always the same even if we choose $p=1.0$. Because the template 1 is expanded to 5x5 size and this can cause a different effect to the image. This means that the choice of template 1 or 2 is important in this template design method. In this noise reduction performance, we consider that the large noises can be removed by setting the “Small Object Remover” template outside of the combination template.

4.2. Edge Detection

Secondly, the effective edge detection results using combination template are shown. The original templates are “Edge Detection” template and “Optimal Edge Detector” template. We set the “Edge Detection” template as Template 1 and the “Optimal Edge Detector” template as Template 2.

Edge Detection (Template 1) :

$$A_1 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}, B_1 = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}, I_1 = -1. \quad (14)$$

By using this template, the edges in binary images can be detected.

Optimal Edge Detector (Template 2):

$$A_2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, B_2 = \begin{bmatrix} -0.11 & 0 & 0.11 \\ -0.28 & 0 & 0.28 \\ -0.11 & 0 & 0.11 \end{bmatrix}, I_2 = 0. \quad (15)$$

By using this template, output images become gray-scale and edges in horizontal direction are enhanced.

Figure 3 shows the simulation results using combination template. Figure 3(b) shows the simulation result using only the “Optimal Edge Detector” template. In this result, we can see that the edge of the mast in the left region of the input image can be detected. Figure 3(c) shows the simulation result using only the “Edge Detection” template. In this result,

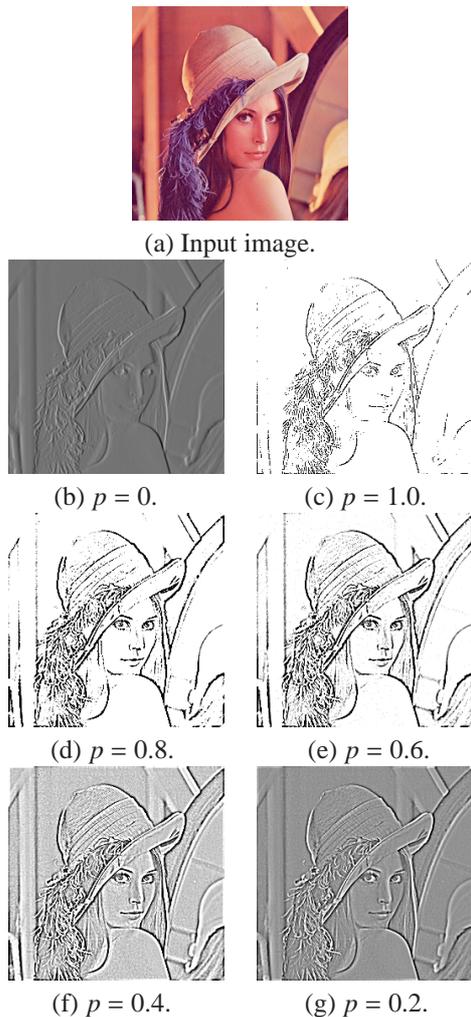


Figure 3: Simulation results using combination template of Edge Detection Template (Template 1) and Optimal Edge Detector Template (Template 2) with different mixing rates.

we can see that the edges of the input image are not visible clearly. Figures 3(d), (e), (f) and (g) show simulation results using combination template with different mixing rates. In Fig. 3(e), the edge of the mast in the left region of the input image can be detected more clearly than using only the “Edge Detection” template. From these results, we can say that the edge detection performance is improved by the both effects of the two templates.

We should mention that the proposed designing method create new templates but the created templates are normal 5x5 space-invariant template. This means that the theory of the standard CNN can be applied to the CNN using proposed template design method.

5. Conclusions

In this study, we have proposed the template design method of the CNN by combining existing templates. By computer simulations of simple image processing tasks for both binary images and gray-scale images, we investigated the basic output characteristics of the proposed systems. As we expected, we could confirm the effects of both templates by combining them. At the moment, we do not say that the proposed method exhibited a superior performance than the original CNN. However, we feel that we obtained some results to broaden the research on the CNN design.

Acknowledgment

This research was partly supported by the found of Open Research Center Project from MEXT of Japanese Government (2007-2011).

References

- [1] L.O. Chua and L. Yang, “Cellular Neural Networks:Theory,” IEEE Trans. Circuits Syst., vol. 32, pp. 1257-1272, Oct. 1988.
- [2] 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.
- [3] 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.
- [4] H. Koeppl and L.O. Chua, “An Adaptive Cellular Non-linear Network and its Application,” Proc. of NOLTA’07, pp. 15-18, Sep. 2007.
- [5] J. Kishida, C. Rekeczky, Y. Nishio and A. Ushida, “Feature Extraction of Postage Stamps Using an Iterative Approach of CNN,” IEICE Trans. on Fundamentals, vol. E79-A, no. 10, pp. 1741-1746, Oct. 1996.
- [6] Á. Zarándy, “The Art of CNN Template Design,” Int. J. Circ. Theor. Appl., vol. 27, no. 1, pp. 5-23, Jan.-Feb. 1999.
- [7] T. Inoue, M. Nakano and Y. Nishio, “Output Characteristics of Cellular Neural Networks Using Mixture Template,” Proc. of IJCNN’08, pp. 4124-4129, Jun. 2008.
- [8] 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”