Investigation of Recall Image by Partitioned Hopfield Neural Network

Tomoya Shima, Chihiro Ikuta, Yoko Uwate and Yoshifumi Nishio
Dept. Electrical and Electronic Eng, Tokushima University
Tokushima 770-8506, JAPAN
Email: {s-tomoya, ikuta, uwate, nishio}@ee.tokushima-u.ac.jp

Abstract—**In this study, we propose Partitioned Hopfield Neural Network (PHNN) to realize the memory mechanism of the human brain. The PHNN is realized by arranging cell which has a small HNN in the whole image regularly without overlap. By computer simulations, we confirm that the PHNN recalls the whole image from the partial characteristic of the stored images.**

I. Introduction

Computer can execute processing which requires a huge amount of computational cost. Human brain can not execute such processing. However, in other cases, the human brain can be superior to the computer, for example, the pattern recognition, the data mining and so on. Furthermore, the human brain have broad utility, which can apply in various fields. Hence, it is imperative that the human brain functions are clarified. In this study, we focus on "memory" which is one of the human brain functions. The memory mechanism of the human brain consists of three processes; "memorization" is to commit new experience to memory, "store" is to retain the memorized information, and "recall" is to remember the stored information. The memory mechanism is interesting information-processing capability [1]. In order to show a specific example of the memory mechanism, we use image A and B (see Fig. 1). The image A includes landscape and a triangle pattern. Meanwhile, the image B includes some patterns. However a triangle pattern in the image B is exactly same as one of the image A.

We assume that a human memorizes the image A, after that the human watches the image B. Then, the human remembers the whole image A by watching the triangle pattern in the image B. The example relates to three processes of the memory mechanism. First, when the human watches the image A, information of the image A is transmitted to the human brain. Then, the human brain "memorizes" and "stores" the information by watching the image A. Second, the human remembers the image A from the triangle pattern in the image B. Namely, the human can recall the whole image from the partial image. If such memory mechanism will be applied in engineering, we expect that much information will be able to be obtained from little information [2]-[5].

We consider that the memory mechanism can be represented by Hopfield Neural Network (HNN) [6]. Because the HNN operating for an associative memory is similar to the memory mechanism [7]. The HNN memorizes and stores some patterns by determining the weight parameter, and recalls a similar pattern to an input pattern from some stored patterns by repeating update of the internal state of all neurons such that the energy of the network becomes minimum. Thereby, the HNN recalls a similar image to the whole input image from some stored images. However, the HNN operating for the associative memory alone can not realize the specific example of the memory mechanism.

In order to realize the memory mechanism, we propose Partitioned HNN (PHNN). In the field of image processing, the method of the divided HNN has been already proposed [8]. However, in our study, the PHNN is used as the associative memory modeled the high-functional information processing of the human brain. In the PHNN, several small HNNs are arranged in the whole image regularly without overlaps. The PHNN recalls similar patterns to partial input image only on small separate regions. By computer simulations, we confirm that the PHNN can recall the whole image from the partial characteristic of the images.

Fig. 1. Example images for the memory mechanism. (a) Image A. (b) Image B.

II. HNN Working as Associative Memory

The HNN based on the model of biological neurons was proposed by J.J. Hopfield in 1982 [6]. Every neurons in the HNN is connected to each other neuron in the network with no self-connection.

The HNN is used as the associative memory by exploiting the property that the network has multiple stable states. Namely, if the parameters of the network can be decided in such a way that the patterns to be stored become stable states of the network, the network produces a stored pattern that is similar to an input pattern. The process of the associative memory by the conventional HNN is described as follows.

[Step1 (HNN)]: If we apply the conventional HNN for the associative memory, one pixel in an image corresponds with one neuron.

[Step2 (HNN)]: The HNN stores images by determining the weight parameter. The weight parameter is given by

$$w_{ij} = \begin{cases} \frac{1}{P} \sum_{p=1}^{P} x_i^{(p)} x_j^{(p)} & (i \neq j) \\ 0 & (i = j), \end{cases} \quad (1)$$

where P is the number of patterns, w is the weight parameter and x is the pattern of memory. It turns out that the weight parameter is symmetrical by Eq. (1).

[Step3 (HNN)]: The neurons is initialized by an unknown image. Namely, the network is given the input. The state of each neuron is determined by Eq. (2).

$$u_i(t) = \sum_{j=1}^{n} w_{ij}(t)x_j(t), \qquad (2)$$

where u is the internal state of the neuron and x is the input or output. The value of u is binary.

[Step4 (HNN)]: Neuron's output is determined by a linear function. The equation of the determining Neuron's output is given as follows.

$$x_i(t+1) = \begin{cases} 1, & (u_i(t) \geq 0) \\ -1, & (u_i(t) < 0). \end{cases} \qquad (3)$$

[Step5 (HNN)]: The HNN recalls a similar image to the input image from the stored images by repeating Step3 to Step4.

III. PHNN Working as Associative Memory

In this section, we explain the process of the PHNN for associative memory. One pixel in an image corresponds with one neuron as same as the conventional HNN. In the PHNN, we separate an image into small square regions. We call one small square region, "cell". Every cell is composed of the pixels and has one small HNN. The cells are arranged in the whole image regularly without overlap. The every cell in the PHNN is not connected each other. Figure 2 shows the composition of the PHNN. In Fig. 2, every cell has 2×2 pixels. This means every cell has the small HNN of 2×2 neurons.

□ **is cell**
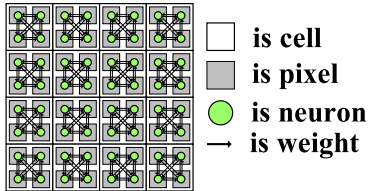▦ **is pixel**
● **is neuron**
→ **is weight**

Fig. 2. The composition of the PHNN.

The process of the PHNN for the associative memory is explained as follows.

[Step1 (PHNN)]: One pixel in an image corresponds with one neuron in the PHNN.

[Step2 (PHNN)]: The PHNN memorizes and stores the partial images on every cell by determining the weight parameter as shown in Fig. 3.

[Step3 (PHNN)]: An initial image is inputted to the PHNN as shown in Fig. 4.

[Step4 (PHNN)]: The PHNN recalls similar patterns to the partial input image on every cell from the partial stored images by repeating updating rule as shown in Fig. 5. Thereby, the PHNN can work associative memory locally. The recalled image by the PHNN is composed patterns of the local stored
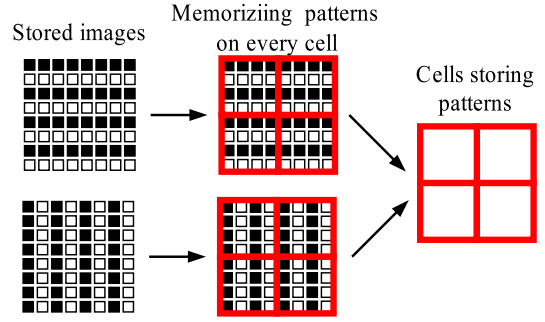

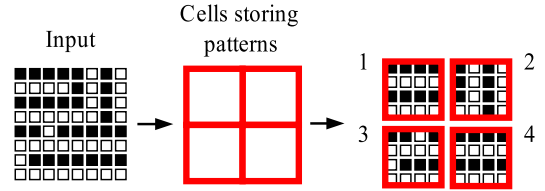Fig. 3. Step2: Memorizing and storing images.

Fig. 4. Step3: Inputting an initial image into the PHNN.

images. Therefore, while the recalled image by the HNN includes only information of one stored image, the recalled image by the PHNN includes information of a number of stored images.
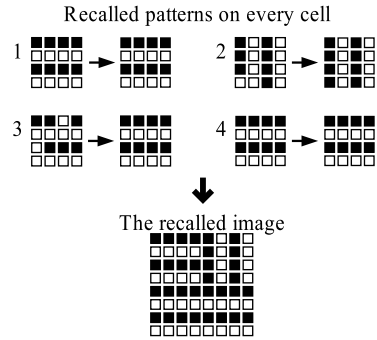

Fig. 5. Step4: Recalling patterns on every cell.

IV. Simulation Results

We apply the PHNN for two types of simulations, "recalling image" and "detecting sign". For comparison, we also investigate the performance of the conventional HNN. The conventional HNN has no cells, namely every neuron of the HNN are connected each other. We focus on the Hamming distance (HD) for the two simulations. The HD between two images is the number of corresponding pixels. The HD is obtained between the stored image and the input image in simulations. Thereby, if the HD is small, the stored image is similar to the input image. Hence, the conventional HNN recalls the stored image which has the smaller HD than other stored images. In this study, we use the HD as the method of confirmation for recalling ability. In the two simulations,

images size is set to 64×64 pixels and every cell has 4×4 or 8×8 neuron units.

A. Recalling Image

In this section, we investigate whether the PHNN can recall the whole image from the partial image. In this simulation, we add two more steps for the PHNN after [Step4 (PHNN)] in order to make a judgment about what stored image is recalled by the PHNN as the whole image. The additional steps for the PHNN is described as follows.

[StepI (PHNN)]: We compare the recalled image at [Step4 (PHNN)] with the stored images for every cell and count the number of cells which have the exact same pattern as each partial stored image in the recalled image as shown in Fig. 6.
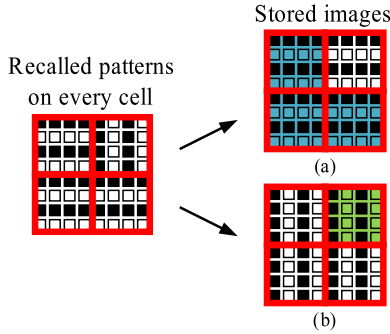

Fig. 6. StepI: Comparing the recalled image with each partial stored images.

[StepII (PHNN)]: The PHNN recalls the whole stored image (a) in Fig. 6 which has more same cells as the recalled image from some stored images.

For the simulation of "recalling image", three binary images are prepared as shown in Fig. 7. Figures 7 (a) and (b) are stored images and Fig. 7 (c) is an input image. Each stored image includes a pattern of sign. The sign in Fig. 7 (c) is the exact same pattern as partial Fig. 7 (a). The HD between Fig. 7 (a) and (c) is 2161, and the HD between Fig. 7 (b) and (c) is 1297. In this simulation, every cell has 4×4 neuron units.
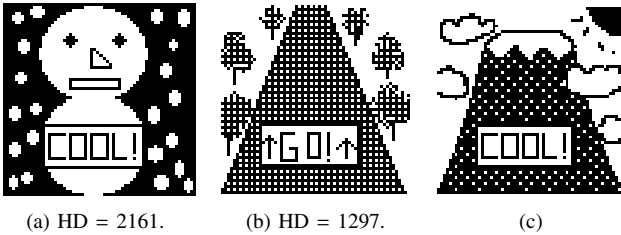

(a) HD = 2161.   (b) HD = 1297.   (c)

Fig. 7. Simulation images for "recalling images". (The number of stored images is two.) (a) Stored image (COOL!). (b) Stored image (GO!). (c) Input image.

The simulation results of "recalling image" is shown in Fig. 8. Figure 8 (a) is recalled image by the conventional HNN. We can see that the conventional HNN recalls stored image which has smaller HD than the others. Hence, the ability of

the conventional HNN depends on the HD. Figure 8 (b) is recalled image by the PHNN. The PHNN clearly recalls the sign in Fig. 7 (a).

By applying [StepI (PHNN)] and [StepII (PHNN)], we obtain the recalled whole image as shown in Fig. 7 (c) Therefore, Fig. 8 (b) has more cells of the same patterns as Fig. 7 (a) than Fig. 7 (b) and the PHNN recalls Fig. 7 (a). Hence, the PHNN can recall Fig. 7 (a) by the sign in Fig. 7 (c) without dependence of the HD.
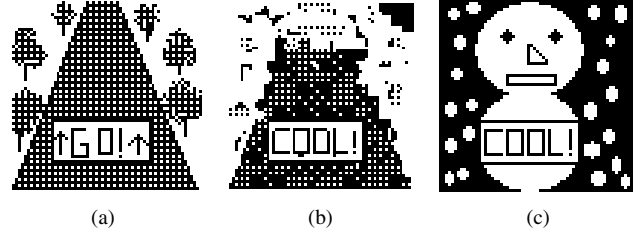

(a)   (b)   (c)

Fig. 8. Simulation results for "recalling image". (The number of stored images is two.) (a) Recalled image by the HNN. (b) Recalled partial image by the PHNN. (c) Recalled whole image by the PHNN.

Next, we investigate the recalling ability when the number of stored image is increased. We add the two stored images as shown in Figs. 9 (c) and (d). Figure 10 shows the simulation result for four stored images. Even if stored image increases, the simulation result is same as previous case (Fig. 8). From these results, we can confirm that the PHNN can recall the whole image from the partial image.
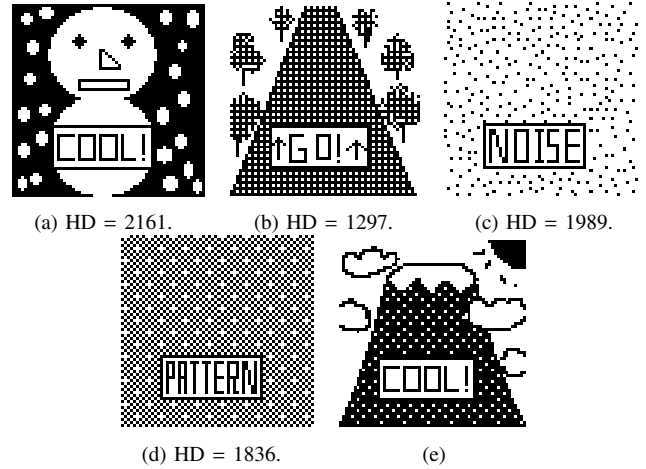

(a) HD = 2161.   (b) HD = 1297.   (c) HD = 1989.

(d) HD = 1836.   (e)

Fig. 9. Simulation images for "recalling image". (The number of stored images is four.) (a) Stored image (COOL!). (b) Stored image (GO!). (c) Stored image (NOISE). (d) Stored image (PATTERN). (e) Input image.

B. Detecting Sign

Next, we investigate whether the PHNN can recall the sign in the stored image even if position of the sign in the input image is changed. In this simulation, we do not add two more steps for the PHNN after [Step4 (PHNN)]. We prepare four stored images and two input images to confirm the behavior of the PHNN as shown in Fig. 11. Figure 11 (a) shows the stored image which is disposed same sign on the image space.
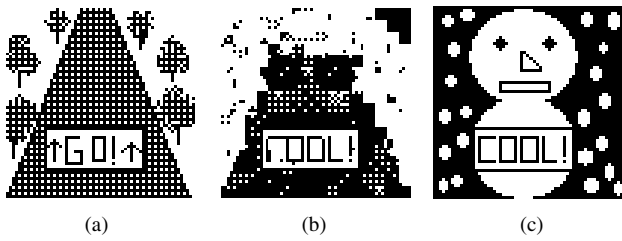
Fig. 10. Simulation results for "recalling image". (The number of stored images is four.) (a) Recalled image by the HNN. (b) Recalled partial image by the PHNN. (c) Recalled whole image by the PHNN.

Figure 11 (b) is the stored image which is a similar image to the input image 1. Figure 11 (c) is the stored image which is a similar image to the input image 2. Figure 11 (d) is the stored image with regular pattern.

In this simulation, two types of input images are prepared as shown in Figs. 11 (e) and (f). Figure 11 (e) is the input image 1 which has the smallest HD to Fig. 11 (b) and the exactly same sign of Fig. 11 (a) is included on the outside of the object. Figure 11 (f) is the input image 2 which has the smallest HD to Fig. 11 (c) and the exactly same sign of Fig. 11 (a) is included on the inside of the object. We compare the recalled images obtained by the PHNN and the conventional HNN for the two input images. In the case of the PHNN, every cell has 8×8 neuron units.

(a) HD1 = 2294. HD2 = 2105.
(b) HD1 = 839. HD2 = 1640.
(c) HD1 = 1756. HD2 = 355.
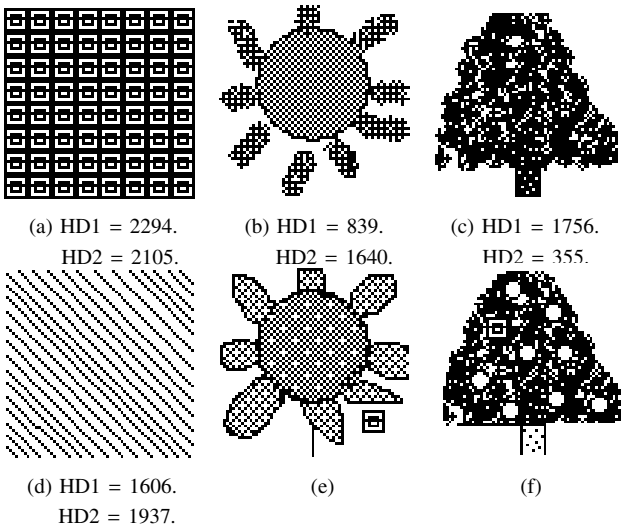(d) HD1 = 1606. HD2 = 1937.
(e)
(f)

Fig. 11. Simulation images for "detecting sign". (a) Stored image 1. (b) Stored image 2. (c) Stored image 3. (d) Stored image 4. (e) Input image 1. (f) Input image 2.

Figures 12 (a) and (b) are recalled by the conventional HNN from Figs. 11 (e) and (f). The conventional HNN recalls stored images which have the smallest HD to the input images. Figures 12 (c) and (d) are recalled by the PHNN from Figs. 11 (e) and (f). From this simulation results, we can see that the PHNN can recall the stored image with the sign even if the position of the sign in the input image is changed.
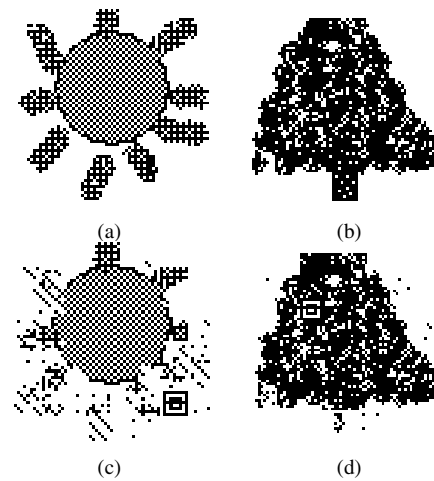

(a)
(b)
(c)
(d)

Fig. 12. Simulation results for "detecting sign". (a) Recalled image by the HNN for input image 1. (b) Recalled image by the HNN for input image 2. (c) Recalled partial image by the PHNN for input image 1. (d) Recalled partial image by the PHNN for input image 2.

V. CONCLUSIONS

In this study, we have proposed the PHNN and applied the PHNN to associative memory. In our method, the PHNN has recalled a similar pattern to the partial input image only on the cell. First simulation results showed that the PHNN recalls the whole image from the partial image. Second simulation results showed that the PHNN recalls the sign in the stored image even if position of the sign in the input image was changed. From these results, we can say that associative memory of the PHNN is closer the memory mechanism of the human brain than the HNN.

REFERENCES

[1] A. Knoblauch, "Neural Associative Memory for Brain Modeling and Information Retrieval," *Information Processing Letters*, vol. 95, pp. 537-544, September, 2005.
[2] Y.T. Zhou, R. Chellappa, A. Vaid and B.K. Jenkins, "Image Restoration Using a Neural Network," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. ASSP-36, pp. 1141-1151, 1988.
[3] Y. Chigusa, K. Suzuki, A. Hattori, M. Ikegami, M. Tanaka, "An Image Reconstruction System By Neural Network with Median Filter," *Proceedings of ISCAS'93*, pp. 2446-2449, May, 1993.
[4] G. Palm, "On Associative Memory," *Biological Cybernetics*, vol. 36, pp. 19-31, 1980.
[5] N. Ikeda, P. Watta, M. Artiklar and M. H. Hason, "A Two-Level Hamming Network for High Performance Associative Memory," *Neural Networks*, vol. 14, pp. 1189-1200, 2001.
[6] J.J. Hopfield, "Neural Networks and Physical Systems with Emergent Collective Computational Abilities," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 79, pp. 2554-2558, 1982.
[7] S.S. Young, P.D. Scott and N.M. Nasrabadi, "Object Recognition Using Multilayer Hopfield Neural Network," *IEEE Transactions on Image Processing*, vol. 6, no. 3, pp.357-372, 1997.
[8] S.W. Perry and L. Guan, "A Partitioned Modified Hopfield Neural Network Algorithm for Real-Time Image Restoration," *Real-Time Imaging*, vol. 2, pp.215-224, 1996.