15

Kohonen Feature Map Associative Memory with Refractoriness based on Area Representation

Tomohisa IMABAYASHI, Yuko OSANA
Tokyo University of Technology
Japan

1. Introduction

Recently, neural networks are drawing much attention as a method to realize flexible information processing. Neural networks consider neuron groups of the brain in the creature, and imitate these neurons technologically. Neural networks have some features, especially one of the important features is that the networks can learn to acquire the ability of information processing.

In the field of neural networks, a lot of models have been proposed such as the Back Propagation algorithm (Rumelhart et al., 1986), the Kohonen Feature Map (KFM) (Kohonen, 1994), the Hopfield network (Hopfield, 1994), and the Bidirectional Associative Memory (Kosko, 1988). In these models, the learning process and the recall process are divided, and therefore they need all information to learn in advance.

However, in the real world, it is very difficult to get all information to learn in advance, so we need the model whose learning process and recall process are not divided. As such model, Grossberg and Carpenter proposed the ART (Adaptive Resonance Theory) (Carpenter & Grossberg, 1995). However, the ART is based on the local representation, and therefore it is not robust for damaged neurons in the Map-Layer. While in the field of associative memories, some models have been proposed (Watanabe et al., 1995; Osana & Hagiwara, 1999; Kawasaki et al., 2000). Since these models are based on the distributed representation, they have the robustness for damaged neurons. However, their storage capacities are small because their learning algorithm is based on Hebbian learning.

On the other hand, the Kohonen Feature Map associative memory (KFM associative memory) (Ichiki et al., 1993) has been proposed. Although the KFM associative memory is based on the local representation as similar as the ART (Carpenter & Grossberg, 1995), it can learn new patterns successively (Yamada et al., 1999), and its storage capacity is larger than that of models in refs.(Watanabe et al., 1995; Osana & Hagiwara, 1999; Kawasaki et al., 2000). It can deal with auto and hetero associations and the associations for plural sequential patterns including common terms (Hattori et al., 2002). Moreover, the KFM associative memory with area representation (Abe & Osana, 2006) has been proposed. In the model, the area representation (Ikeda & Hagiwara, 1997) was introduced to the KFM associative memory, and it has robustness for damaged neurons. However, it can not deal with one-to-many associations and associations of analog patterns.

In this research, we propose a Kohonen Feature Map Associative Memory with Refractoriness based on Area Representation (KFMAM-R-AR). The proposed model is based on the KFM associative memory with area representation (Abe & Osana, 2006) and the neurons in the Map-Layer have refractoriness. In the proposed model, one-to-many associations are realized by refractoriness of neurons. Moreover, by improvement of the calculation of the internal states of the neurons in the Map-Layer, the proposed model has enough robustness for damaged neurons when analog patterns are memorized.

2. KFM Associative Memory with Area Representation

Here, we briefly review the KFM Associative Memory with Area Representation (KFMAM-AR) (Abe & Osana, 2006) which is used in the proposed model.

2.1 Structure

Figure 1 shows the structure of the KFM associative memory with area representation (Abe & Osana, 2006). As shown in Fig.1, the KFMAM-AR has two layers; (1) Input/Output (I/O)-Layer and (2) Map-Layer, and the I/O-Layer is divided into some parts.
In the KFMAM-AR, since one concept is expressed by the winner neuron and some neurons located adjacent to the winner neuron, it has the robustness for damaged neurons in the Map-Layer.

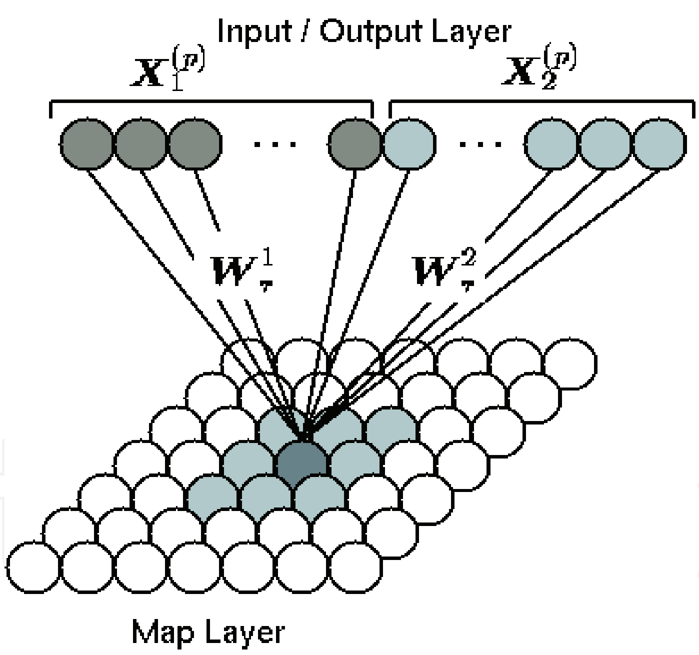


Fig. 1. Structure of KFMAM-AR.

2.2 Learning Process

The learning algorithm for the KFMAM-AR is based on the conventional sequential learning algorithm for the KFM associative memory (Ichiki et al., 1993).
Let us consider the case where the I/O-Layer composed of N parts corresponding to the pattern $X_1^{(p)}, X_2^{(p)}, \cdots, X_N^{(p)}$ in the KFMAM-AR. In this case, the learning vector $X^{(p)}$ is given by

$$\boldsymbol{X}^{(p)} = \begin{pmatrix} \boldsymbol{X}_1^{(p)} \\ 0 \\ \vdots \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ \boldsymbol{X}_2^{(p)} \\ \vdots \\ 0 \end{pmatrix} + \cdots + \begin{pmatrix} 0 \\ \vdots \\ 0 \\ \boldsymbol{X}_N^{(p)} \end{pmatrix} \tag{1}$$

where $\boldsymbol{X}^{(p)} \in \{0,1\}^M$, $\boldsymbol{X}_j^{(p)} \in \{0,1\}^{M_j}$ and M_j is the number of neurons corresponding to the j th partial pattern in the $\boldsymbol{X}^{(p)}$. In addition, M is the number of neurons in the I/O-Layer.
Let $\boldsymbol{W}_i$ be the connection weights between the neurons in the I/O-Layer and the neuron i in the Map-Layer:

$$\boldsymbol{W}_i = \begin{pmatrix} \boldsymbol{W}_i^1 \\ 0 \\ \vdots \\ \vdots \\ 0 \end{pmatrix} + \cdots + \begin{pmatrix} 0 \\ \vdots \\ \boldsymbol{W}_i^j \\ \vdots \\ 0 \end{pmatrix} + \cdots + \begin{pmatrix} 0 \\ \vdots \\ \vdots \\ 0 \\ \boldsymbol{W}_i^N \end{pmatrix} \tag{2}$$

where $\boldsymbol{W}_i^j$ is the connection weight between the neuron i in the Map-Layer and the neurons of the j th part in the I/O-Layer.
In the sequential learning algorithm for the KFMAM-AR, the connection weights are learned as follows:
(1) The initial values of weights are chosen randomly.
(2) The Euclid distance between the learning vector $\boldsymbol{X}^{(p)}$ and the connection weights vector $\boldsymbol{W}_i$, $d(\boldsymbol{X}^{(p)}, \boldsymbol{W}_i)$ is calculated by

$$d\left(\boldsymbol{X}^{(p)}, \boldsymbol{W}_i\right) = \sqrt{\sum_{k=1}^{M} \left(X_k^{(p)} - W_{ik}\right)^2}\,. \tag{3}$$

(3) The winner neuron r whose Euclid distance is minimum is found.

$$r = \underset{i}{\operatorname{argmin}}\ d\left(\boldsymbol{X}^{(p)}, \boldsymbol{W}_i\right) \tag{4}$$

(4) If $d(\boldsymbol{X}^{(p)}, \boldsymbol{W}_i) > \theta^l$, the connection weights of the winner neuron r are fixed. The connection weights except those of fixed neurons are changed by

$$\boldsymbol{W}_i(t+1) = \boldsymbol{W}_i(t) + H(d_i)\eta(t)h_{ri}\left(\boldsymbol{X}^{(p)} - \boldsymbol{W}_i(t)\right) \tag{5}$$

where h_{ri} is the neighborhood function and is given by

$$h_{ri} = \exp\left(\frac{-\|\boldsymbol{r}-\boldsymbol{i}\|^2}{2\sigma(t)^2}\right) \tag{6}$$

where $\sigma(t)$ is the following decreasing function:

$$\sigma(t) = \sigma_i \left(\frac{\sigma_f}{\sigma_i}\right)^{\frac{t}{T}} \tag{7}$$

In this equation, σ_i is the initial value of $\sigma(t)$ and $\sigma(t)$ varies from σ_i to $\sigma_f(\sigma_i > \sigma_f)$. T is the upper limit of the learning iterations.
In Eq.(5), $\eta(t)$ is the learning rate and is given by

$$\eta(t) = \frac{-\eta_0(t-T)}{T} \tag{8}$$

where η_0 is the initial value of $\eta(t)$, and $H(d_i)$ is calculated by

$$H(d_i) = \frac{1}{1+\left(-\frac{d_i - D}{\varepsilon}\right)} \tag{9}$$

In this equation, d_i is the Euclid distance between the neuron i and the nearest weights fixed neuron in the Map-Layer, D is the constant and ε is the steepness parameter of the function $H(d_i)$. Owing to $H(d_i)$, weights of neurons close to the fixed neurons are semi-fixed, that is, they become hard to be learned.
(5) (2)~(9) are iterated until $d\left(\boldsymbol{X}^{(p)}, \boldsymbol{W}_i\right) \le \theta^l$ is satisfied.
(6) The connection weights of the winner neuron r, W_r are fixed.
(7) (2)~(6) are iterated when a new pattern set is given.

2.3 Recall Process

Since the conventional KFM associative memory is based on the local representation, it has not the robustness for damaged neurons in the Map-Layer. In contrast, the KFMAM-AR is based on the area representation (Ikeda & Hagiwara, 1997) and has the robustness for damaged neurons. The area representation is an intermediate representation of the local representation and the distributed representation. In the area representation, one concept is expressed by the winner neuron and some neurons located adjacent to the winner neuron.

2.3.1 Recall Process for Binary Patterns

In the recall process of the KFMAM-AR, when the binary pattern $\boldsymbol{X}$ is given to the I/O-

Layer, the output of the neuron i in the Map-Layer x_i^{map} is calculated by

$$x_i^{map} = \begin{cases} 1, & \text{if } d(\boldsymbol{X}, \boldsymbol{W}_i) < \theta_b^{map} \\ 0, & \text{otherwise} \end{cases} \tag{10}$$

where θ_b^{map} is the threshold of the neuron in the Map-Layer as follows:

$$\theta_b^{map} = d_{min} + a(d_{max} - d_{min}) \tag{11}$$

$$d_{min} = \min_i d(\boldsymbol{X}, \boldsymbol{W}_i) \tag{12}$$

$$d_{max} = \max_i d(\boldsymbol{X}, \boldsymbol{W}_i) \tag{13}$$

In Eq.(11), $a\ (0 < a < 0.5)$ is the coefficient.
Then, the output of the neuron k in the I/O-Layer x_k^{in} is calculated as follows:

$$x_k^{in} = \begin{cases} 1, & \text{if } u_k^{in} \geq \theta_b^{in} \\ 0, & \text{otherwise} \end{cases} \tag{14}$$

$$u_k^{in} = \frac{1}{\sum_i x_i^{map}} \sum_{i:x_i=1} W_{ik} \tag{15}$$

where θ_b^{in} is the threshold of the neuron in the I/O-Layer, u_k^{in} is the internal state of the neuron k in the I/O-Layer.

2.3.2 Recall Process for Analog Patterns

In the recall process of the KFM-AR, when the analog pattern $\boldsymbol{X}$ is given to the I/O-Layer, the output of the neurons i in the Map-Layer x_i^{map} is calculated by

$$x_i^{map} = \begin{cases} 1, & \text{if } d(\boldsymbol{X}, \boldsymbol{W}_i) < \theta_a \\ 0, & \text{otherwise} \end{cases} \tag{16}$$

where θ_a is the threshold of the neuron in the Map-Layer.
Then, the output of the neuron k in the I/O-Layer x_k^{in} is calculated as follows:

$$x_k^{in} = \frac{1}{\sum_i x_i^{map}} \sum_{i:x_i=1} W_{ik} \,. \tag{17}$$

3. KFM Associative Memory with Refractoriness based on Area Representation

The conventional KFM associative memory (Ichiki et al., 1993) and KFMAM-AR (Abe & Osana, 2006) cannot realize one-to-many associations. In this paper, we propose the Kohonen Feature Map Associative Memory with Refractoriness based on Area Representation (KFMAM-R-AR) which can realize one-to-many associations. The proposed model is based on the KFMAM-AR, and the neurons in the Map-Layer have refractoriness. In the proposed model, one-to-many associations are realized by the refractoriness of neurons.
On the other hand, although the conventional KFMAM-AR can realize associations for analog patterns, it does not have enough robustness for damaged neurons. In this research, the model which has enough robustness for damaged neurons when analog patterns are memorized is realized by improvement of the calculation of the internal states of neurons in the Map-Layer.

3.1 Learning Process

In the proposed model, the patterns are trained by the learning algorithm of the KFMAM-AR described in **2.2**.

3.2 Recall Process

In the recall process of the proposed model, when the pattern $\boldsymbol{X}$ is given to the I/O-Layer, the output of the neuron i in the Map-Layer x_i^{map} is calculated by

$$x_i^{map}(t) = H^{recall}(d(\boldsymbol{r},\boldsymbol{i})) f(u_i^{map}(t)) \tag{18}$$

$$H^{recall}(d(\boldsymbol{r},\boldsymbol{i})) = \frac{1}{1+\exp\left(\dfrac{d(\boldsymbol{r},\boldsymbol{i})-D}{\varepsilon}\right)} \tag{19}$$

where D is the constant which decides area size, ε is the steepness parameter. $d(\boldsymbol{r},\boldsymbol{i})$ is the Euclid distance between the winner neuron r and the neuron i and is calculated by

$$r = \underset{i}{\operatorname{argmax}}\; u_i^{map}(t) \tag{20}$$

Owing to $H^{recall}(d(\boldsymbol{r},\boldsymbol{i}))$, the neurons which are far from the winner neuron become hard to fire. $f(u_i^{map}(t))$ is calculated by

$$f(u_i^{map}(t)) = \begin{cases} 1, & \text{if } u_i^{map}(t) > \theta^{map} \text{ and } u_i^{map}(t) > \theta^{min} \\ 0, & \text{otherwise} \end{cases} \tag{21}$$

where $u_i^{map}(t)$ is the internal state of the neuron i in the Map-Layer at the time t, θ^{map} and θ^{min} are the thresholds of the neuron in the Map-Layer. θ^{map} is calculated as follows:

$$\theta^{map} = u_{min} + a(u_{max} - u_{min}) \tag{22}$$

$$u_{min} = \min_i u_i^{map}(t) \tag{23}$$

$$u_{max} = \max_i u_i^{map}(t) \tag{24}$$

where $a(0.5 < a < 1)$ is the coefficient.
In Eq.(18), when the binary pattern $\boldsymbol{X}$ is given to the I/O-Layer, the internal state of the neuron i in the Map-Layer at the time t, $u_i^{map}(t)$ is calculated by

$$u_i^{map}(t) = 1 - \frac{d^{in}(\boldsymbol{X}, \boldsymbol{W}_i)}{\sqrt{N^{in}}} - \alpha \sum_{d=0}^{t} k_r^d x_i^{map}(t-d) \tag{25}$$

where $d^{in}(\boldsymbol{X}, \boldsymbol{W}_i)$ is the Euclid distance between the input pattern $\boldsymbol{X}$ and the connection weights $\boldsymbol{W}_i$. In the recall process, since all neurons in the I/O-Layer not always receive the input, the distance for the part where the pattern was given is calculated as follows:

$$d^{in}(\boldsymbol{X}, \boldsymbol{W}_i) = \sqrt{\sum_{\substack{k=1 \\ k \in C}} (X_k - W_{ik})^2} \tag{26}$$

where C shows the set of the neurons in the I/O-Layer which receive the input. In Eq.(25), N^{in} is the number of neurons which receive the input in the I/O-Layer, α is the scaling factor of the refractoriness and $k_r (0 \le k_r < 1)$ is the damping factor. The output of the neuron k in the I/O-Layer at the time t, $x_k^{in}(t)$ is calculated by

$$x_k^{in}(t) = \begin{cases} 1, & \text{if } u_k^{in}(t) \ge \theta_b^{in} \\ 0, & \text{otherwise} \end{cases} \tag{27}$$

$$u_k^{in}(t) = \frac{1}{\sum_i x_i^{map}(t)} \sum_{i: x_i > \theta^{out}} W_{ik} \tag{28}$$

where is the threshold of the neuron in the I/O-Layer, is the threshold for the

output of the neuron in the Map-Layer.
On the other hand, when the analog pattern is given to the I/O-Layer at the time t , $u_i^{map}(t)$ is calculated by

$$u_i^{map}(t) = \frac{1}{N^{in}} \sum_{\substack{k=1 \\ k \in C}}^{N^{in}} g(X_k - W_{ik}) - \alpha \sum_{d=0}^{t} k_r^d x_i^{map}(t-d). \tag{29}$$

Here, $g(\cdot)$ is calculated as follows:

$$g(b) = \begin{cases} 1, & |b| < \theta^b \\ 0, & \text{otherwise} \end{cases} \tag{30}$$

where θ^b is the threshold.
In the conventional KFMAM-AR, the neurons whose Euclid distance between the input vector and the connection weights are not over the threshold fire. In contrast, in the proposed model, the neurons which have many elements whose difference between the weight vector and the input vector are small can fire. The output of the neuron k in the I/O-Layer at the time t , $x_k^{in}(t)$ is calculated as follows:

$$x_k^{in}(t) = \frac{1}{\sum_i x_i^{map}(t)} \sum_{i: x_i^{map} > \theta^{out}} W_{ik}. \tag{31}$$

4. Computer Experiment Results

In this section, we show the computer experiment results to demonstrate the effectiveness of the proposed model. Table 1 shows the experimental conditions.

4.1 Association Result for Binary Patterns

Here, we show the association result of the proposed model for binary patterns. In this experiment, the number of neurons in the I/O-Layer was set to 800(= 400 × 2) and the number of neurons in the Map-Layer was set to 400. Figure 2 (a) shows an example of stored binary pattern pairs.
Figure 3 shows the association result of the proposed model when "lion" was given. As shown in this figure, the proposed model could realize one-to-many associations.

4.2 Association Result for Analog Patterns

Here, we show the association result of the proposed model for analog patterns. In this experiment, the number of neurons in the I/O-Layer was set to 800(= 400 × 2) and the number of neurons in the Map-Layer was set to 400. Figure 2 (b) shows an example of stored analog pattern pairs.

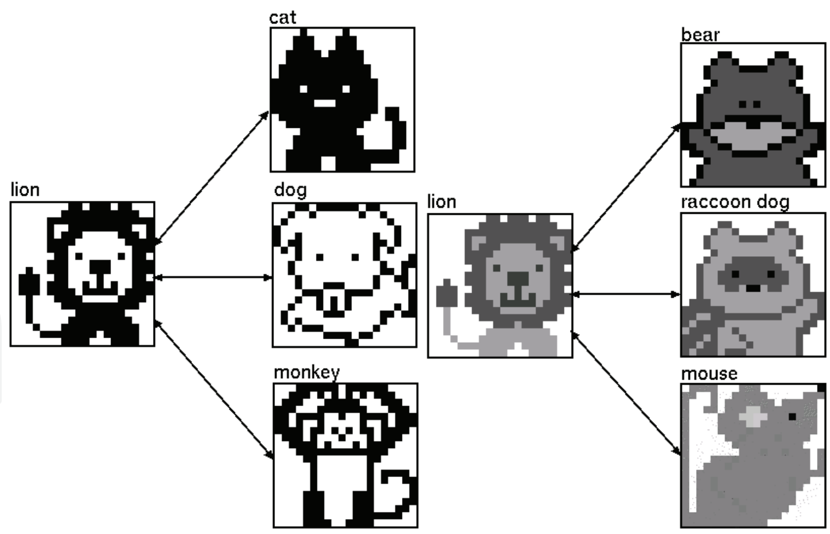
Figure 4 shows the association result of the proposed model when "lion" was given. As shown in this figure, the proposed model could realize one-to-many associations for analog patterns.

4.3 Storage Capacity

Here, we examined the storage capacity of the proposed model. In this experiment, we used the proposed model which has 800(= 400 × 2) neurons in the I/O-Layer and 400/800 neurons in the Map-Layer. We used random patterns and Figs.5 and 6 show the average of 100 trials. In these figures, the horizontal axis is the number of stored pattern pairs, and the vertical axis is the storage capacity. As shown in these figures, the storage capacity of the proposed model for the training set including one-to-many relations is as large as that for the training set including only one-to-one relations.

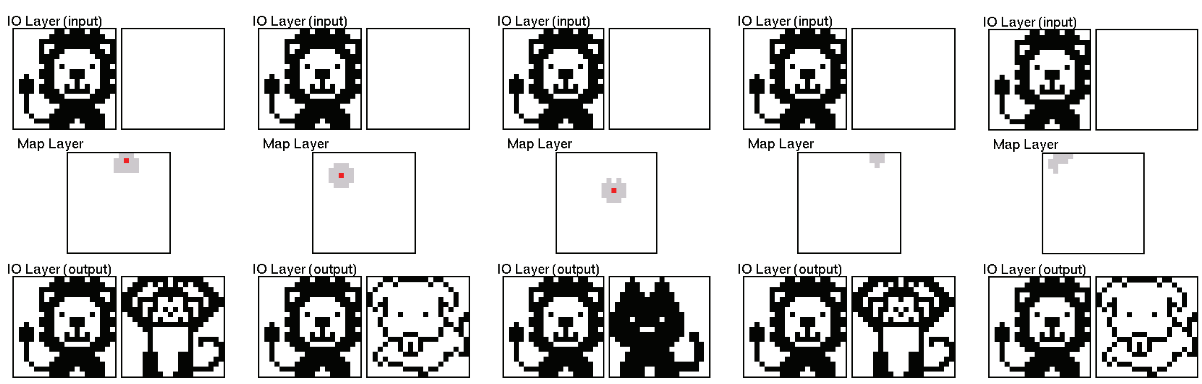
Parameters for learning		
threshold(learning)	θ^l	10^{-7}
initial value of η	η_0	0.1
initial value of σ	σ_i	3.0
last value of σ	σ_f	0.5
steepness parameter	ε	0.01
coefficient (range of semi-fixed)	D	3.0
Parameters for recall (common)		
scaling factor of refractoriness	α	1.0
damping factor	k_r	0.9
steepness of H^{recall}	ε	0.01
coefficient (size of area)	D	3.0
threshold (minimum)	θ^{min}	0.5
threshold (output)	θ^{out}	0.99
Parameters for recall (binary)		
coefficient (threshold)	a	0.9
threshold in the I/O-Layer	θ_b^{in}	0.5
Parameter for recall (analog)		
threshold (difference)	θ^d	0.1

Table 1. Experimental Conditions.



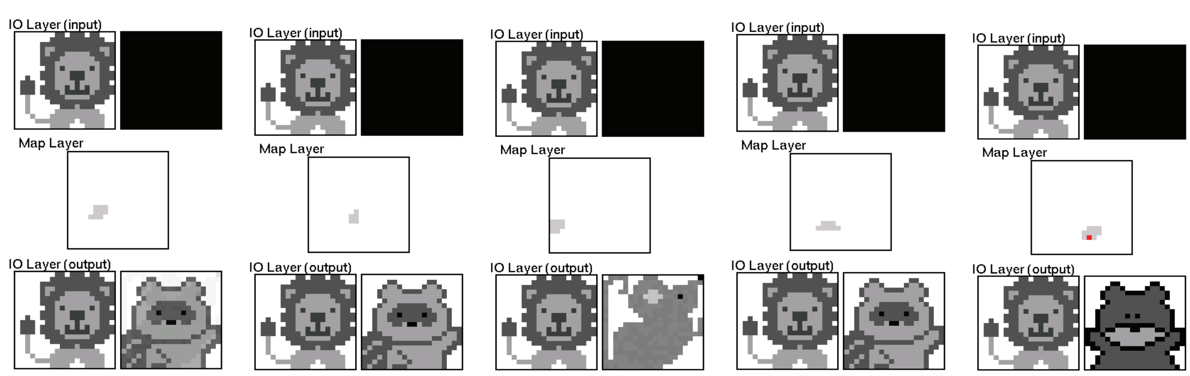
(a) Binary Pattern (b) Analog Pattern

Fig. 2. An Example of Stored Patterns.



$t = 1$ $t = 2$ $t = 3$ $t = 4$ $t = 5$

Fig. 3. Association Result for Binary Patterns.



$t = 1$ $t = 2$ $t = 3$ $t = 4$ $t = 5$

Fig. 4. Association Result for Analog Patterns.

4.4 Recall Ability for One-to-Many Associations

Here, we examined the recall ability in one-to-many associations of the proposed model. In this experiment, we used the proposed model which has 800(= 400 × 2) neurons in the I/O-Layer and 400 neurons in the Map-Layer. We used one-to-P(P = 1, 2, · · · , 30) random patterns and Fig.7 shows the average of 100 trials. In Fig.7, the horizontal axis is the number of stored pattern pairs, and the vertical axis is the recall rate. As shown in Fig.7, the proposed model could recall all patterns when P is smaller than 15 (binary patterns) / 4 (analog patterns). Although the proposed model could not recall all patterns corresponding to the input when P was 30, it could recall about 25 binary patterns / 17 analog patterns.

4.5 Noise Reduction Effect

Here, we examined the noise reduction effect of the proposed model.

Figure 8 shows the noise sensitivity of the proposed model for analog patterns. In this experiment, we used the proposed model which has 800(= 400 × 2) neurons in the I/O-Layer and 400 neurons in the Map-Layer and 9 random analog patterns (three sets of patterns in one-to-three relations) were stored. Figure 8 shows the average of 100 trials.

In the proposed model, the minimum threshold of the neurons in the Map-Layer θ^{min} influences the noise sensitivity. As shown in Fig.8, we confirmed that the proposed model is more robust for noisy input when θ^{min} is small.

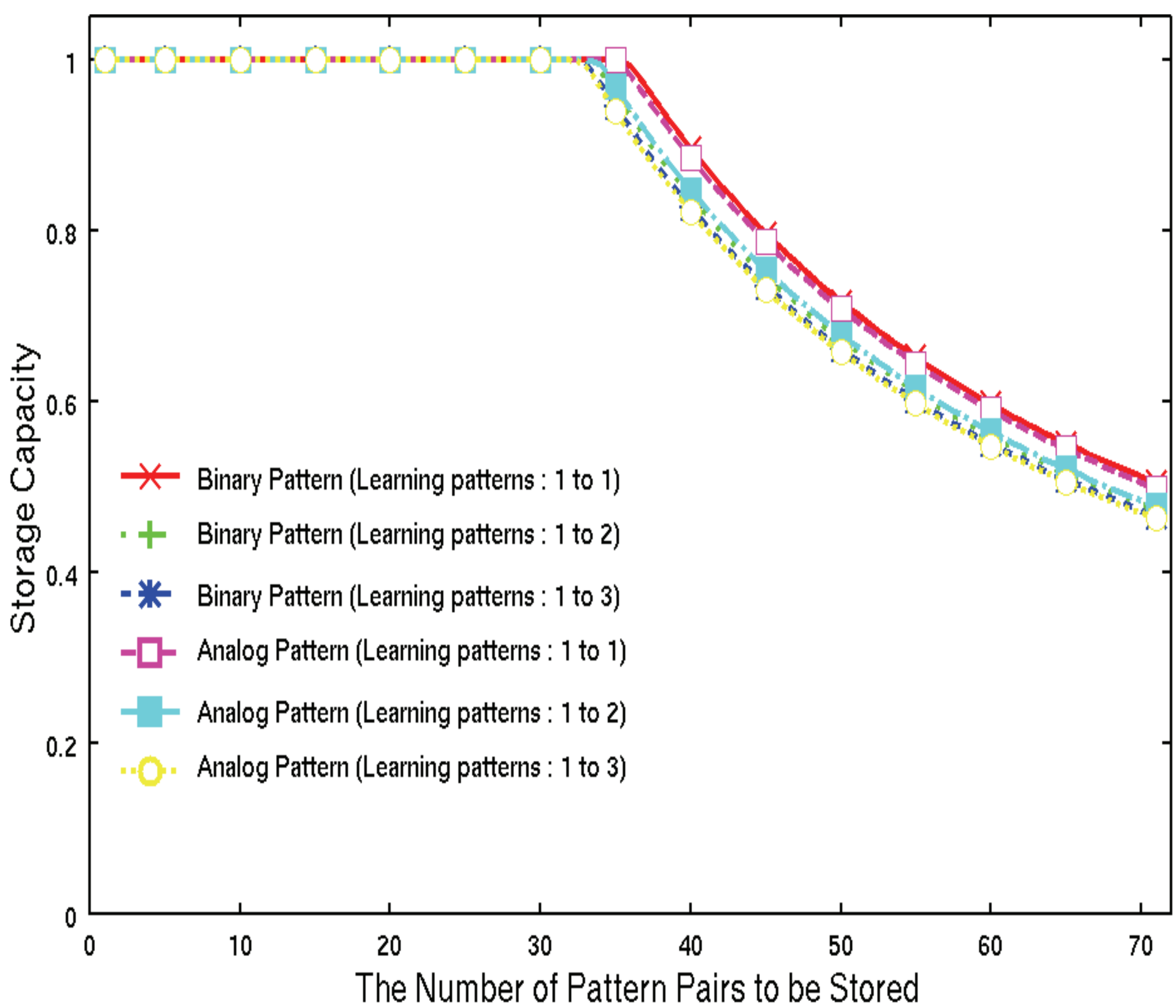


Fig. 5. Storage Capacity (400 neurons in the Map-Layer).

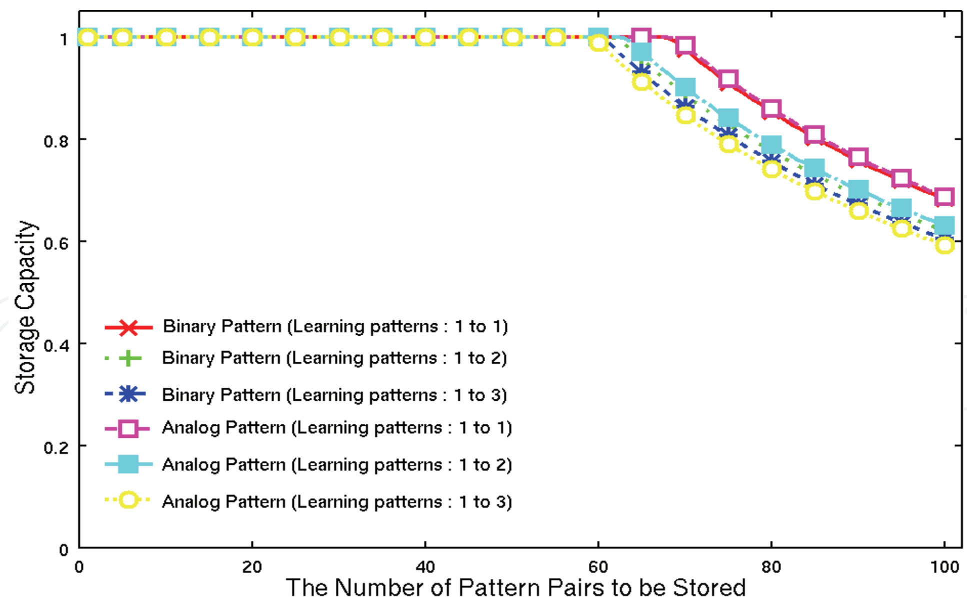


Fig. 6. Storage Capacity (800 neurons in the Map-Layer).

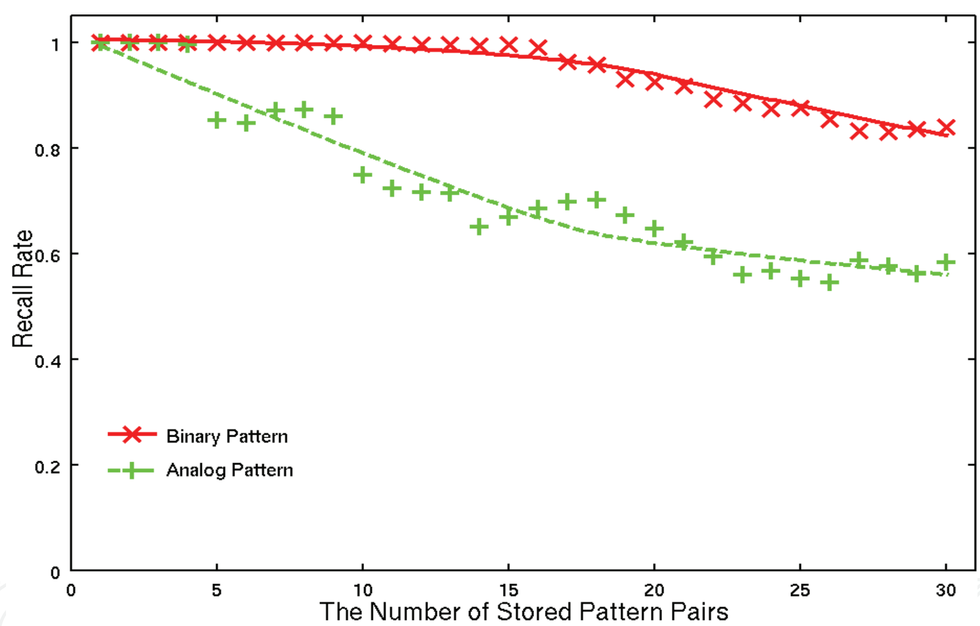


Fig. 7. Recall Ability in One-to-Many Associations.

4.6 Robustness for Damaged Neurons

Here, we examined the robustness for damaged neurons of the proposed model.
Figure 9 shows the robustness for damaged neuron of the proposed model. In this experiment, we used the proposed model which has 800(= 400 × 2) neurons in the I/OLayer and 400 neurons in the Map-Layer and 9 random patterns (three sets of patterns in one-to-three relations) were stored. In this experiment, n% of the neurons in the Map- Layer were damaged randomly. Figure 9 shows the average of 100 trials. In this figure, the results of the conventional KFMAM-AR were also shown.
From this result, we confirmed that the proposed model has the robustness for damaged neurons.

5. Conclusion

In this research, we have proposed the KFM Associative Memory with Refractoriness based on Area Representation. The proposed model is based on the KFMAM-AR (Abe & Osana, 2006) and the neurons in the Map-Layer have refractoriness. We carried out a series of computer experiments and confirmed that the proposed model has following features.

(1) It can realize one-to-many associations of binary patterns.
(2) It can realize one-to-many associations of analog patterns.
(3) It has robustness for noisy input.
(4) It has robustness for damaged neurons.

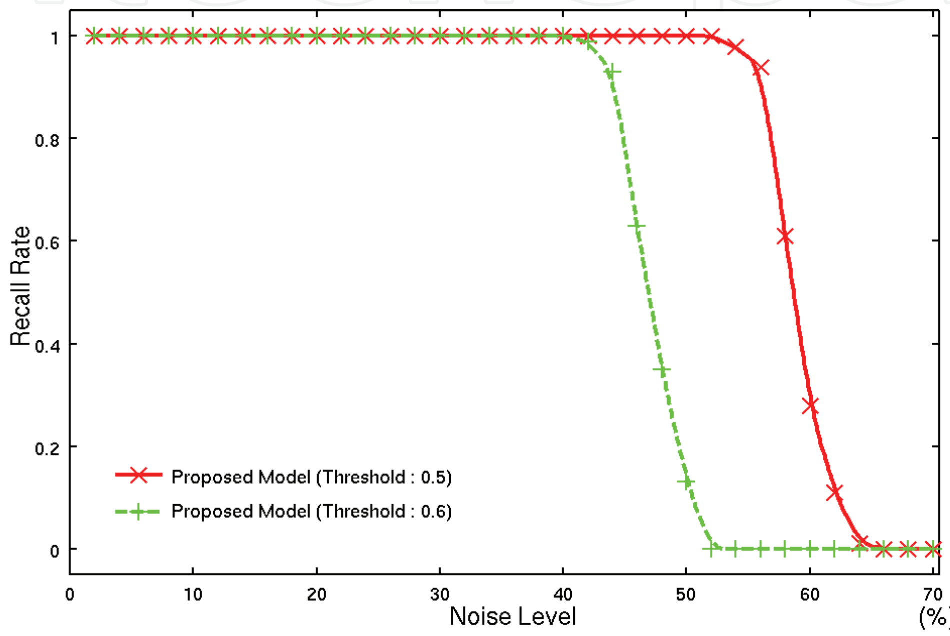


Fig. 8. Sensitivity to Noise (Analog Pattern).

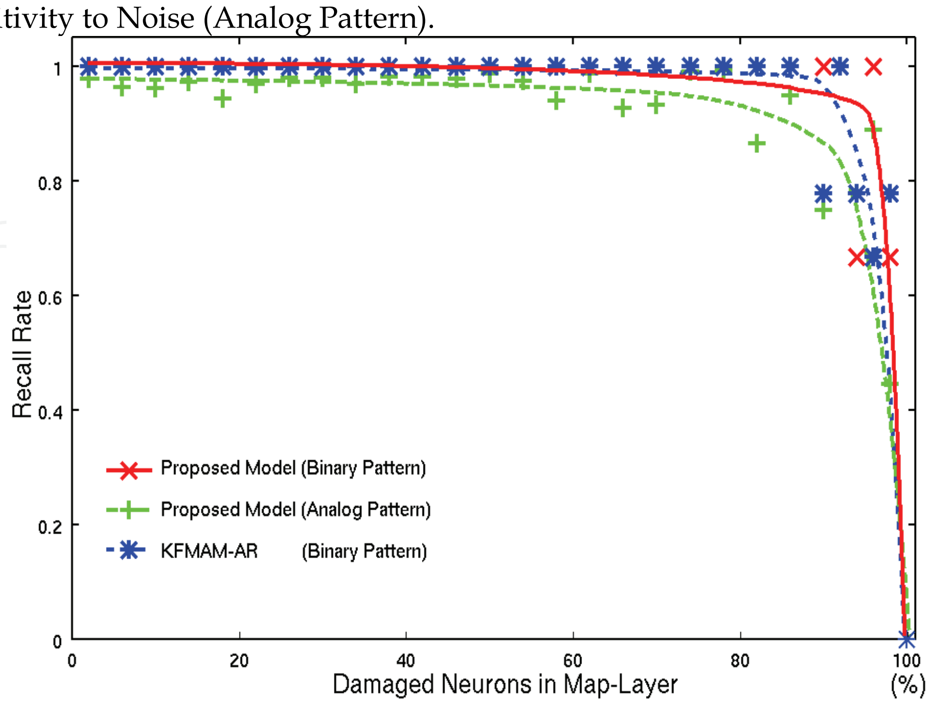


Fig. 9. Robustness for Damaged Neurons.

6. References

Abe, H. & Osana, Y. (2006). Kohonen feature map associative memory with area representation. *Proceedings of IASTED Artificial Intelligence and Applications*, Innsbruck.

Carpenter, G, A. & Grossberg, S. (1995). Pattern Recognition by Self-organizing Neural Networks. *The MIT Press*.

Hattori, M. Arisumi, H. & Ito, H. (2002). SOM Associative Memory for Tempral Sequences. *Proceedings of IEEE and INNS International Joint Conference on Neural Networks*, pp. 950-955, Honolulu.

Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proceedings of National Academy Scienses USA*, Vol.79, pp.2554–2558.

Ichiki, H. Hagiwara, M. & Nakagawa, M. (1993). Kohonen feature maps as a supervised learning machine. *Proceedings of IEEE International Conference on Neural Networks*, pp.1944–1948.

Ikeda, N. & Hagiwara, M. (1997). A proposal of novel knowledge representation (Area representation) and the implementation by neural network. *International Conference on Computational Intelligence and Neuroscience*, III, pp. 430–433.

Kawasaki, N. Osana, Y. & Hagiwara, M. (2000) Chaotic associative memory for successive learning using internal patterns. *IEEE International Conference on Systems, Man and Cybernetics*.

Kohonen, T. (1994). *Self-Organizing Maps*, Springer.

Kosko, B. (1988). Bidirectional associative memories. *IEEE Transactions on Neural Networks*, Vol.18, No.1, pp.49–60.

Osana, Y. & Hagiwara, M. (1999). Successive learning in chaotic neural network. *International Journal of Neural Systems*, Vol.9, No.4, pp.285–299.

Rumelhart, D, E. McClelland, J, L. & the PDP Research Group. (1986). Parallel Distributed Processing. *Exploitations in the Microstructure of Cognition*, *Foundations, The MIT Press*, Vol.11.

Watanabe, M. Aihara, K. & Kondo, S. (1995). Automatic learning in chaotic neural networks. *IEICE-A*, Vol.J78-A, No.6, pp.686–691.

Yamada, T. Hattori, M. Morisawa, M. & Ito, H. (1999). Sequential learning for associative memory using Kohonen feature map. *Proceedings of IEEE and INNS International Joint Conference on Neural Networks*, paper no.555, Washington D.C.

New Developments in Robotics Automation and Control
Edited by Aleksandar Lazinica

ISBN 978-953-7619-20-6
Hard cover, 450 pages
Publisher InTech
Published online 01, October, 2008
Published in print edition October, 2008

This book represents the contributions of the top researchers in the field of robotics, automation and control and will serve as a valuable tool for professionals in these interdisciplinary fields. It consists of 25 chapter that introduce both basic research and advanced developments covering the topics such as kinematics, dynamic analysis, accuracy, optimization design, modelling , simulation and control. Without a doubt, the book covers a great deal of recent research, and as such it works as a valuable source for researchers interested in the involved subjects.

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821