Proceedings of International Joint Conference on Neural Networks, Atlanta, Georgia, USA, June 14-19, 2009

Community Self-Organizing Map and its Application to Data Extraction

Taku Haraguchi, Haruna Matsushita and Yoshifumi Nishio

Abstract— **The Self-Organizing Map (SOM) is a famous algorithm for the unsupervised learning and visualization introduced by Teuvo Kohonen. One of the most attractive applications of SOM is clustering and several algorithms for various kinds of clustering problems have been reported and investigated. This study proposes the Community Self-Organizing Map (CSOM) algorithm which reflects the community in the human society. In CSOM algorithm, the neurons create some communities according to their winning frequency. We apply CSOM to various input data for clustering and data extraction, and we investigate its behaviors. We confirm that CSOM creates some communities and obtain efficient results for data extraction.**

I. Introduction

IN data mining, clustering is one of typical analysis techniques and is studied for many applications, such as a statement, a pattern recognition, an image analysis and so on. Then, the Self-Organizing Map (SOM) [1] has attracted attention for the study on clustering in recent years. SOM is an unsupervised neural network introduced by Kohonen in 1982 and is a simplified model of the self-organization process of the brain. SOM obtains statistical feature of input data and is applied to a wide field of data classifications [2]−[8]. SOM can classify input data according to similarities and patterns which are obtained by the distance between neurons and some visualization methods based on SOM were proposed [9]-[13]. On the other hand, in the real world, the amount and the complexity of data increase from year to year. Therefore, it is important to classify the data and to exactly recognize condition of the data.

Meanwhile, in human society, human-beings belong to sub-society, which is called community, such as company, laboratory and circle of a hobby and so on (as Fig. 1). It is based on definition that the human-beings are social animals introduced by Aristotle. The social animal creates a society and lives in the society. Furthermore, in the community, it is also believed that the animals are centered around a leader and have the society. In other words, the community is created as core on the leader of the community. In addition, in creating the community, the human-beings have some tendencies that human-beings easily gather around the leader, such as a CEO controlling the company, a supervisor controlling the laboratory and a chair keeping a circle together, and so on. Besides the human-beings belong to the community, there are also human-beings excluded from the community.

Taku Haraguchi, Haruna Matsushita and Yoshifumi Nishio are with the Department of Electrical and Engineering, University of Tokushima, Tokushima, Japan (phone: +81–88–656–7470; fax: +81–88–656–7471; email: {taku, haruna, nishio}@ee.tokushima-u.ac.jp).

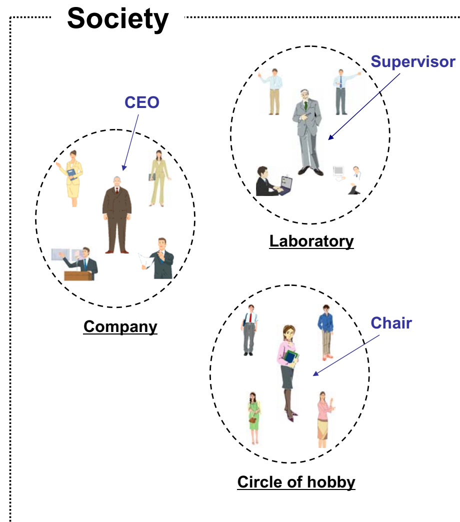

Fig. 1. Some communities in human society. A central person is a "leader" in each community.

In this study, we propose a new type of SOM algorithm, which is called Community SOM (CSOM) algorithm. The important feature of CSOM is that neurons create some neuron-community according to their winning frequency, and the neurons, which is not satisfied the condition, are removed from the community including these after all leaning. We apply CSOM for clustering and data extraction to various input data. We confirm the efficiency of CSOM for the clustering and the data extraction.

We explain the learning algorithm of CSOM in detail in Section III. In Section IV, we apply CSOM to various 2-dimensional and 3-dimensional input data. Furthermore, we explain the learning behaviors of CSOM in detail and confirm the efficiency of CSOM for the clustering and the data extraction.

II. Self-Organizing Map (SOM)

SOM has a two-layer structure of an input layer and a competitive layer. In the input layer, there are d-dimensional input vectors $\boldsymbol{x}_j = (x_{j1}, x_{j2}, \cdots, x_{jd})$ $(j = 1, 2, \cdots, N)$. In the competitive layer, M neurons are arranged on the 2-dimensional grid. Each neuron has a weight vectors $\boldsymbol{w}_i = (w_{i1}, w_{i2}, \cdots, w_{id})$ $(i = 1, 2, \cdots, M)$ with the same dimension as the input vector. The range of the input vector is assumed to be between 0 and 1. The initial values of all the weight vectors are given between 0 and 1 at random.

(SOM1) Input an input vector $\boldsymbol{x}_j$ to all the neurons simultaneously in parallel.
(SOM2) Find a winner c by calculating a distance between the input vector $\boldsymbol{x}_j$ and the weight vector $\boldsymbol{w}_i$ of each neuron i;

$$c = \arg\min_i \{\|\boldsymbol{w}_i - \boldsymbol{x}_j\|\}, \tag{1}$$

where $\|\cdot\|$ is the distance measure, in this study, we use Euclidean distance.
(SOM3) Update the weight vectors of all the neurons as

$$\boldsymbol{w}_i(t+1) = \boldsymbol{w}_i(t) + h_{c,i}(t)(\boldsymbol{x}_j - \boldsymbol{w}_i(t)), \tag{2}$$

where t is the learning step. $h_{c,i}(t)$ is called the neighborhood function and is described as

$$h_{c,i}(t) = \alpha(t) \exp\left(-\frac{\|\boldsymbol{r}_i - \boldsymbol{r}_c\|^2}{2\sigma^2(t)}\right), \tag{3}$$

where $\boldsymbol{r}_i$ and $\boldsymbol{r}_c$ are the vectorial locations on the display grid, $\alpha(t)$ is called the learning rate, and $\sigma(t)$ corresponds to the width of the neighborhood function. Both $\alpha(t)$ and $\sigma(t)$ decrease monotonically with time, in this study, we use

$$\alpha(t) = \alpha(0)\left(1 - \frac{t}{T}\right), \quad \sigma(t) = \sigma(0)\left(1 - \frac{t}{T}\right), \tag{4}$$

where T is the maximum number of the learning.
(SOM4) The steps from (SOM1) to (SOM3) are repeated for all the input data.

III. Community Self-Organizing Map (CSOM)

In this study, we propose a Community SOM (CSOM) algorithm. The important feature of CSOM is that the neurons create some neuron-community according to their winning frequency. In other words, in CSOM algorithm, the winner, which satisfies the condition for the winning frequency, and its neighborhood neurons, which satisfy the same condition, create $k_{\rm th}$ community C_k. In the community C_k, a leader l_k is a neuron that has become the winner most frequently among the all neurons belonging to C_k. Because in the human society, the human-beings also creates some community. This phenomenon has tendency that human-beings gather around a leader.

A. Learning Algorithm

We explain the learning algorithm of CSOM in detail. In CSOM, M neurons are arranged as a regular 2-dimensional grid. A winning frequency W_i is associated with each neuron and is set to zero initially: $W_i = 0$. The number of members in each community C_k and the number of community n are zero. Before learning, the all neurons do not belong to any community, however, they gradually belong to some community with learning.

(CSOM1) Input an input vector $\boldsymbol{x}_j$ to all the neurons simultaneously in parallel.
(CSOM2) Find a winner c according to Eq. (1). Increase the winning frequency of the winner c by

$$W_c^{\rm new} = W_c^{\rm old} + 1. \tag{5}$$

(CSOM3) Updated the weight vectors of all the neurons according to Eq. (2). If $t \geq T_{\rm min}$ is satisfied, perform (CSOM4). If not, perform (CSOM9). $T_{\rm min}$ is fixed parameter and the minimum number of the learning in creating community.
(CSOM4) Evaluate whether the winner c satisfies the conditions of the winning frequency to update the community informations. If $W_c > W_{\rm th}(t)$ is satisfied, perform (CSOM5). If not, perform (CSOM9) without updating the community. $W_{\rm th}(t)$ is the threshold value and increases with learning as

$$W_{\rm th}(t) = \frac{t}{M}. \tag{6}$$

(CSOM5) Find the community C_k including the winner c. If winner c does not belong to any community, create a new community, $n^{\rm new} = n^{\rm old} + 1$, and affiliate the winner c to new community C_k as $c \in C_k$ (where $k = n^{\rm new}$). If not, c remains in its community C_k.
(CSOM6) Find a leader l_k which has become the winner most frequently among the all neurons belonging to C_k, according to Eq. (7) as Fig. 2.

$$l_k = \arg\max_i \{W_i\}, \quad i \in C_k. \tag{7}$$

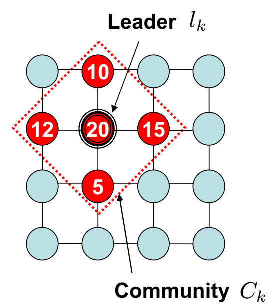

Fig. 2. How to update leader l_k in community C_k. Number in each neuron denotes its winning frequency W_i. The neuron with $W_i = 20$, which is the highest winning frequency among the neurons in the community C_k, becomes the leader l_k.

(CSOM7) Find neurons, whose winning frequency are higher than $W_{\rm th}(t)$, in 1-neighborhoods of the winner c, then consider whether they belong to any community. If this neighborhood neuron belongs to any community, perform (CSOM8). If not, affiliate it to the community C_k including the winner c in Fig. 3, update the leader l_k as (CSOM6), and perform (CSOM9).
(CSOM8) Compare the winning frequencies of two leaders between the community including the winner and the community including winner's neighborhood neuron. Loss of generality, assume that the winner c belongs to C_1 and

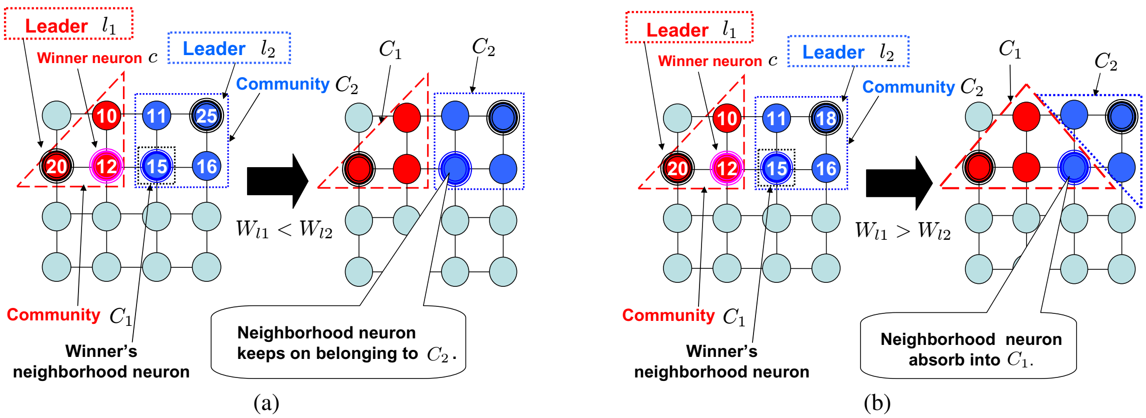

Fig. 4. How to determine whether a community C_1 absorb a community C_2 including a neighborhood neuron. Number in each neuron denotes its winning frequency W_i. The leader l_1 and l_2 are a neuron with the highest winning frequency in the community C_1 and in the community C_2, respectively. (a) As the winning frequency $W_{l_1} = 20$ of the leader l_1 is lower than the winning frequency $W_{l_2} = 25$ of the leader l_2, the neighborhood neuron keeps on belonging to C_2. (b) As $W_{l_1} = 20$ is higher than $W_{l_2} = 18$, the neighborhood neuron is absorbed into C_1.

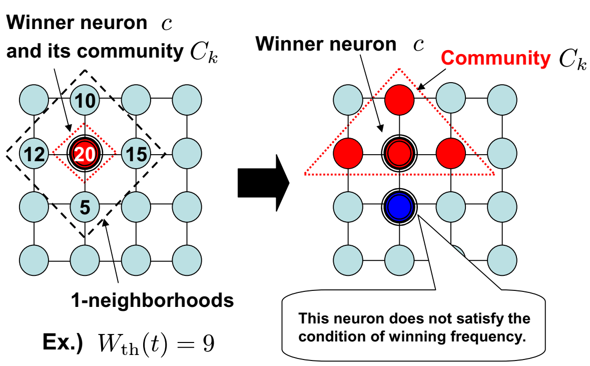

Fig. 3. How to update community C_k. Number in each neuron denotes its winning frequency W_i. The winner's 1-neighborhood neurons with higher winning frequency than $W_{\rm th}$(t) belong to community C_k. The neuron with $W_i = 5$, which is lower winning frequency than $W_{\rm th}(t)$, belongs to no community.

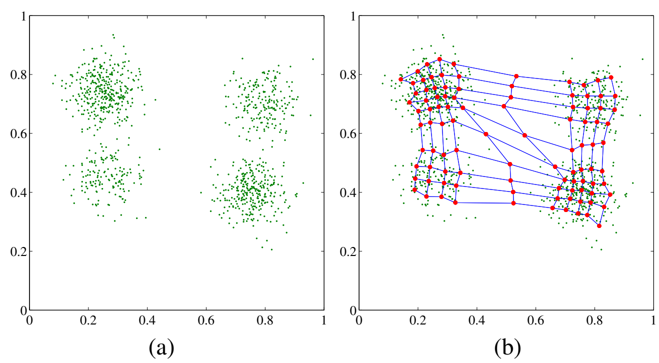

Fig. 5. Clustering for 2-dimensional input data. (a) Input data. (b) Learning result of the conventional SOM.

its neighborhood neuron belongs to C_2. The leaders of C_1 and C_2 are assumed as l_1 and l_2, respectively. If $W_{l_2} \geq W_{l_1}$, the neighborhood neuron keeps on belonging to C_2 as Fig. 4(a). If not, the neighborhood neuron belonging to C_2 are absorbed into C_1 as Fig. 4(b). Then, in a specific case, if the neighborhood neuron is the leader l_2 in the community C_2, all the neurons belonging to C_2 are absorbed into C_1 and decrease the number of communities as $n^{\rm new} = n^{\rm old} - 1$.

(CSOM9) Repeat the steps from (CSOM1) to (CSOM8) for all the input data.

(CSOM10) After all learning are finished, check whether $W_i > T/2M$ for each particle i. If it is not satisfied, remove the particle i from the community including it.

IV. Learning Results of CSOM

A. For 2-dimensional input data (1)

We consider the 2-dimensional input data generated by the probability density function as shown in Fig. 5(a). This data has four clusters, however, it is difficult to find four clusters because the each cluster is close to each other and the density of each cluster is different. The total number of the input data N is 1000. The top-left cluster has 350 data, the top-right cluster has 200 data, left lower cluster has 150 data, right lower cluster has 300 data, and the variance of all the clusters are the same values. The conventional SOM and CSOM have 100 neurons (10×10). We repeat the learning 15 times for all the input data, namely $T = 15000$. The parameters for the learning for two algorithm are chosen as follows;

$$\alpha(0) = 0.3, \;\; \sigma(0) = 3.0, \;\; T_{\rm min} = \frac{T}{3}.$$

The learning result of the conventional SOM is shown in Fig. 5(b). We can see that the conventional SOM has some inactive neurons, which are on a part without the input data, between clusters and the boundary line between clusters are not clear.

1) Behavior of CSOM: Let us consider the learning process and behaviors of CSOM in detail as Fig. 6. In the early stage of learning as Fig. 6(b), all the neurons do not belong to any communities, In Fig. 6(c), some neurons

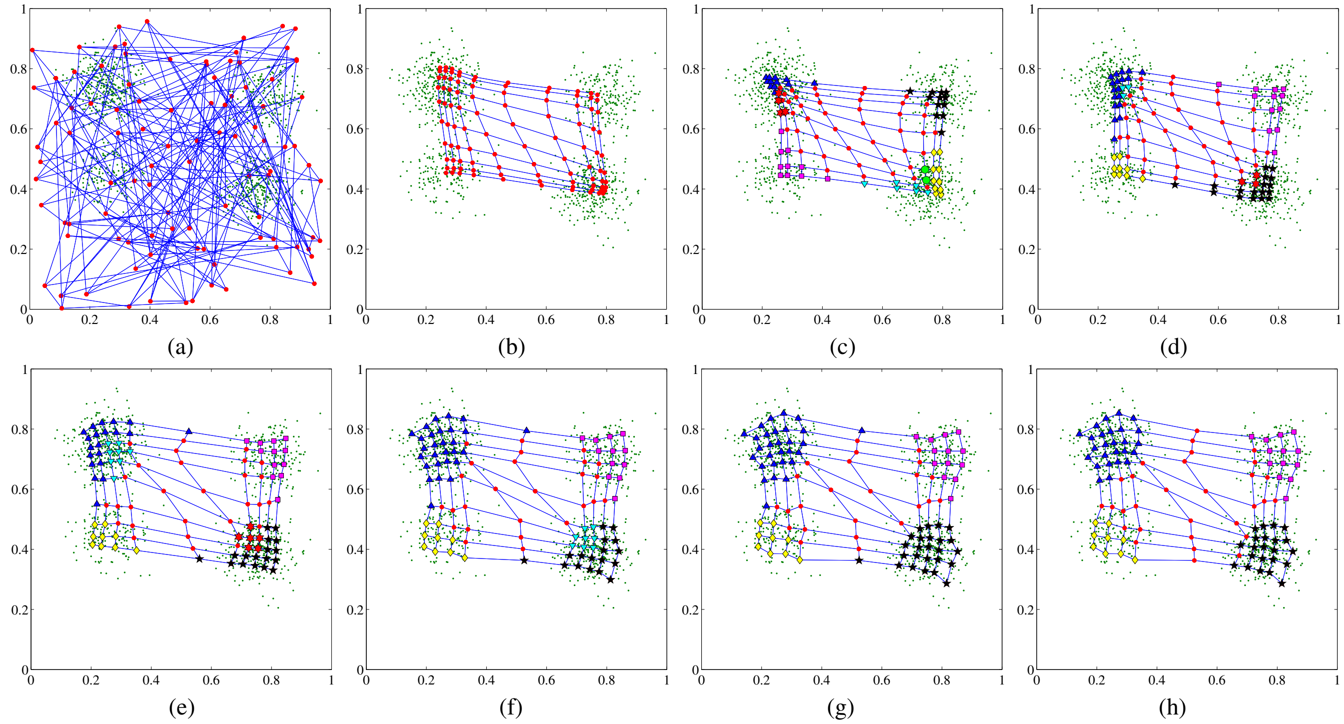

Fig. 6. Clustering by using CSOM for 2-dimensional input data. ▲, ★, ■, ♦, ▼, ✶ and ● denote from the largest community C_1 to the seventh largest community C_7, respectively. (a) Initial state (t = 0). (b) t = 4999. (c) t = 5200. (d) t = 5500. (e) t = 11000. (f) t = 13000. (g) Learning result before (CSOM9) (t = 15000). (h) Learning result after (CSOM9) (t = 15000).

belong to create small communities because $T_{\min} = 5000$, however, the number of their communities are too much. Furthermore, in Fig. 6(d), as small communities absorb some neurons according to (CSOM7), or they merge with other small community according to (CSOM8), then their sizes grow gradually. In the middle stage of learning as Figs. 6(e) and (f), the size of communities grow larger because the neurons, which are self-organizing each cluster, belong to any community on the cluster. However, some communities include not only the neurons self-organizing the cluster but also the inactive neurons. In the last stage of learning as in Fig. 6(g), the number of communities is the same as the number of clusters. Fig. 6(h) shows the learning result after the excluding process (CSOM10) that the neurons, whose winning frequency are not satisfied the condition, are excluded from the community including them. Therefore, the inactive neurons are excluded from the community because whose winning frequency is low. In other words, the communities of CSOM are composed of only the neurons which self-organize the clusters.

2) Comparison between SOM and CSOM: The learning result of CSOM after the excluding process according to (CSOM10) is shown in Fig. 6(h). The learning result of CSOM is completely same as the result of the conventional SOM, however, CSOM carries out the creating of the community in parallel. From this result, we can see that the number of communities is the same as the number of clusters. Furthermore, only the neurons, which self-organize the area where the input data are concentrated, create the communities and the inactive neurons belong to no community.

3) Recognition of data condition: Next, in order to recognize condition of the input data, we use the gray scale display method [9] for the conventional SOM as follows. Figure 7 shows distances between neighboring neurons and visualizes the cluster structure of the map after learning. Black squares on this figure mean large distance between neighboring map nodes. Clusters are typically uniform areas of white squares.

We can see that the boundary line is not clear and it is difficult to recognize the number of clusters and the density on its cluster.

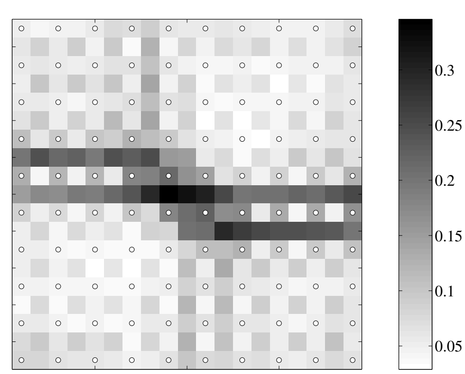

Fig. 7. Visualization of result of conventional SOM for 2-dimensional data.

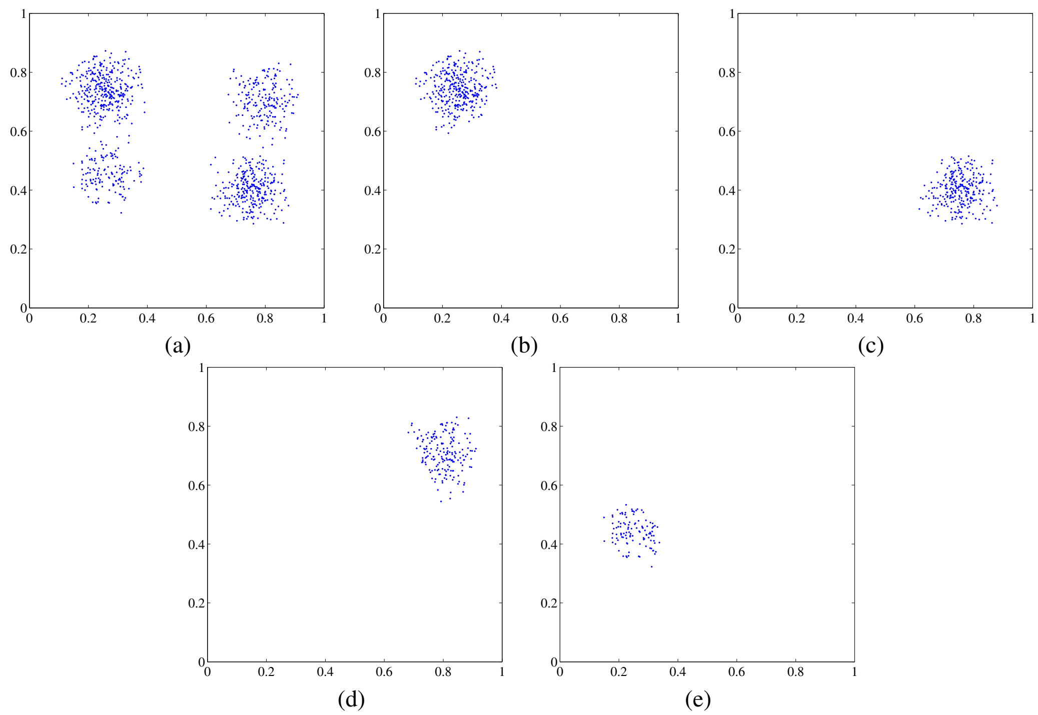

Fig. 8. Extraction results of cluster. (a) Clusters extracted by the conventional SOM. (b)−(e) Cluster extracted by from the largest community C_1, which the number of neurons belonging to is the largest, to the fourth community C_4, which the number of neurons belonging to is the smallest, by using CSOM, respectively.

On the other hand, Table I shows the community situation of CSOM after learning. The number of communities is the same as the number of clusters, we can understand the rough number of clusters by using simple method, which is to recognize the number of communities and the number of neurons belonging to the community. In this table, the number of members denotes the number of neurons belonging to each community, and we express the number of members as community size. In other words, each community C_1, C_2, C_3 and C_4 contains to 26, 22, 13 and 9 neurons, respectively. The community size depends on the number of the input data in the cluster. In other words, the size of the community, which is composed of the neurons self-organizing the cluster containing a lot of the input data, is large.

TABLE I
COMMUNITY SITUATION OF CSOM FOR 2-DIMENSIONAL INPUT DATA.

Number of community $n = 4$	
Community C_k	Number of members
C_1	26
C_2	22
C_3	13
C_4	9

4) Application to Data Extraction: The concept using CSOM is to recognize condition of the input data. We carry out the cluster extraction of cluster from the results of two algorithms as Fig. 5(b) and Fig. 6(h). The extraction method is a relatively simple as follows. In the conventional SOM, after learning, the input data, which is within a radius of R from all neurons on the map, are classified into the cluster. In CSOM, after learning, the input data, which is within a radius of R from all neurons belonging to each community on the map, are classified into the cluster.

The extraction result of the conventional SOM is shown in Fig. 8(a) and the extraction results of respective communities in CSOM are shown in Figs. 8(b)−(e), respectively ($R = 0.05$). In the conventional SOM, we can see that the result obtains almost same as the input data, in other words, the conventional SOM can not exact the cluster relationship. In CSOM, as all the neurons belonging to the each community self-organize the each cluster, the results as Figs. 8(b)−(e) obtain four clusters. Besides, the community size depends on the number of the input data in the cluster. Therefore, we can obtain the cluster containing the most input data in all the clusters by extracting the largest community C_1, as Fig. 8(b).

B. *For 2-dimensional input data (2)*

Next, we consider another 2-dimensional input data shown in Fig. 9(a). The input data is generated artificially as follows. The total number of the input data N is 1000, and the input data contain two clusters. 200 data are distributed within range from 0.4 to 0.6 horizontally and from 0.4 to 0.6 vertically. The remaining 800 data are distributed around the central data. We repeat the learning 15 times for all the input data, namely $T = 15000$. The learning conditions are the same as those used in Subsection IV-A.

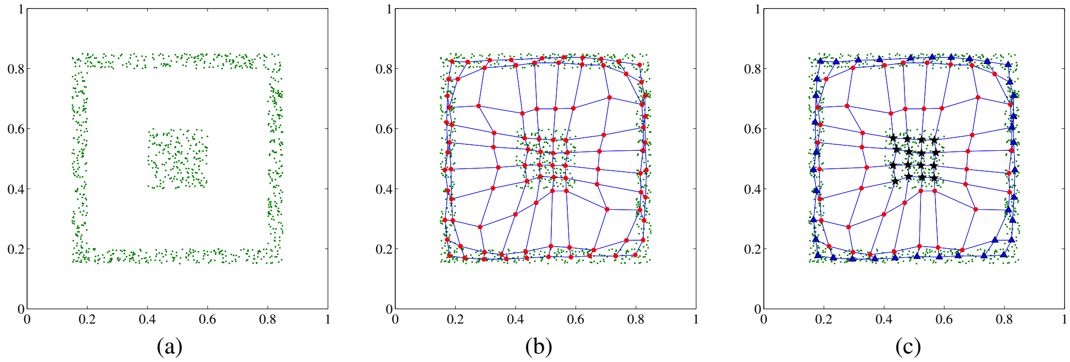

Fig. 9. Clustering for 2-dimensional input data. ▲ and ★ denote the largest community C_1 and the second largest community C_2, respectively. (a) Input data. (b) Learning result of the conventional SOM. (c) Learning result of CSOM.

The learning results of the conventional SOM and CSOM are shown in Figs. 9(b) and (c), respectively. We can see that CSOM can classify outside cluster into only one community. Furthermore, some neurons, which self-organize the central area where the input data are concentrated, create one community. Therefore, the number of communities is the same as the number of clusters.

Furthermore, Table II shows the community situation of CSOM for 2-dimensional input data. Each community C_1 and C_2 contains 39 and 16 neurons. The neurons belonging to C_1 self-organize the outside cluster, and the neurons belonging to C_2 self-organize the central cluster. The neurons gather on the cluster containing a lot of the input data, therefore, the community size becomes large depending on it.

TABLE II
COMMUNITY SITUATION OF CSOM FOR 2-DIMENSIONAL INPUT DATA.

Number of community $n = 2$	
Community C_k	Number of members
C_1	39
C_2	16

1) Application to Data Extraction: We carry out the cluster extraction of clusters by using the learning result of CSOM as Fig. 9(c). The extraction results of respective communities in CSOM are shown in Figs. 10(a) and (b) ($R = 0.05$). Because the inactive neurons belong to no community, only the neurons self-organizing the clusters belong to the communities. Therefore, the extraction results of CSOM obtain exactly two clusters, which are the center cluster and the out-side cluster.

C. For Hepta data

Next, we consider a 3-dimensional input data called Hepta data set [14] shown in Fig. 11(a) which has a clustering problem of different densities in clusters. The total number of the input data N is 212, and the input data has seven clusters. We repeated the learning 70 times for all the input data, namely $T = 14840$. The learning conditions are the same as those used in Subsection IV-A.

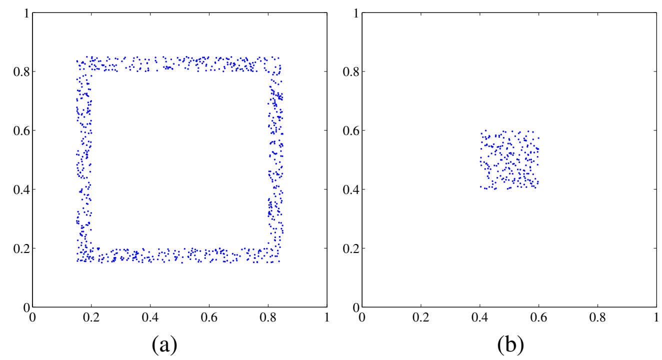

Fig. 10. Extraction results of cluster by using CSOM. (a) Cluster extracted by the largest community C_1. (b) Cluster extracted by the second community C_2.

The learning results of the conventional SOM and CSOM are shown in Figs. 11(b) and (c), respectively. CSOM can classify the neurons on seven clusters into respective seven communities. Furthermore, Table III shows the community situation of CSOM for Hepta data. Each community from C_1 to C_7 contains from 7 to 4 neurons. There is no significant difference in the number of members between respective communities. This is because that the number of the input data in each cluster is almost the same, however, the density distribution of central cluster and other six clusters is different. Therefore, the each community size is almost the same when both the number of the input data and the density distribution in each cluster are almost the same. In other words, we can consider that the subtle difference of the community size is the influence of the initial state and order of the input data.

D. For 3-dimensional data

Finally, we consider another 3-dimensional input data shown in Fig. 12(a). The input data is generated artificially as follows. The total number of the input data N is 1000, and the input data contain two clusters. Centrally-located 200 data are distributed within cube 0.2 on a side. The remaining 800 data are distributed around centrally-located data. We repeat the learning 15 times for all the input data, namely

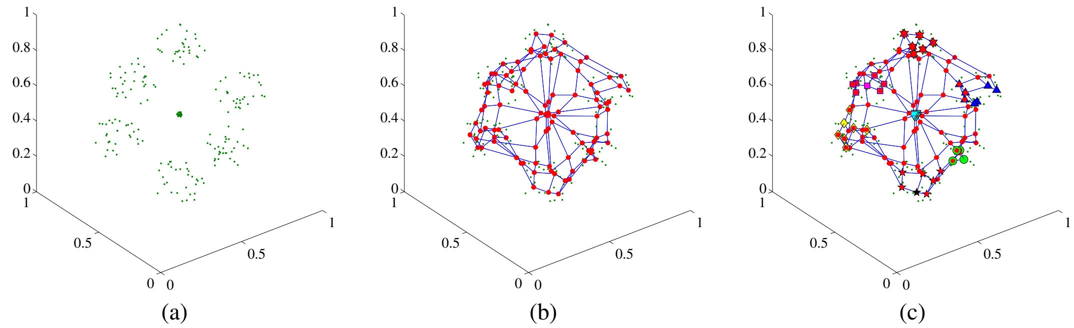

Fig. 11. Clustering for Hepta data. ▲, ★, ■, ♦, ▼, ✶ and ● denote from the largest community C_1 to the seventh largest community C_7, respectively. (a) Input data. (b) Learning result of the conventional SOM. (c) Learning result of CSOM.

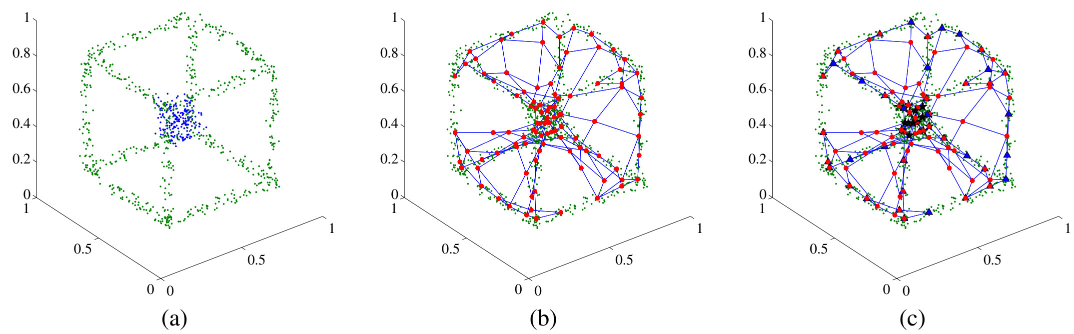

Fig. 12. Clustering for 3-dimensional input data. ▲ and ★ denote the largest community C_1 and the second largest community C_2, respectively. (a) Input data. (b) Learning result of the conventional SOM. (c) Learning result of CSOM.

TABLE III
COMMUNITY SITUATION OF CSOM FOR HEPTA DATA.

Number of community	$n = 7$
Community C_k	Number of members
C_1	7
C_2	7
C_3	7
C_4	7
C_5	6
C_6	6
C_7	4

$T = 15000$. The learning conditions are the same as those used in Subsection IV-A.

The learning results of the conventional SOM and CSOM are shown in Figs. 12(b) and (c), respectively. CSOM can classify the input data in the outside cluster having a complicated shape into only one community. Furthermore, some neurons, which self-organize the central area where the input data are concentrated, create one community, and a lot of the inactive neurons belong to no community. In other words, the number of communities is the same as the number of clusters. Therefore, from Table IV, we can know that the number of clusters is two by the number of communities.

TABLE IV
COMMUNITY SITUATION OF CSOM FOR 3-DIMENSIONAL INPUT DATA.

Number of community	$n = 2$
Community C_k	Number of members
C_1	41
C_2	17

1) Application to Data Extraction: We carry out the cluster extraction of cluster by using the learning result of CSOM as Fig. 12(c). The extraction results of respective communities in CSOM are shown in Figs. 13(a) and (b) ($R = 0.05$). Because the inactive neurons belong to no community, only the neurons self-organizing the clusters belong to the communities. Therefore, the extraction results of CSOM obtain exactly two clusters, which are the center cluster and the out-side cluster.

V. CONCLUSION

In this study, we proposed a new SOM algorithm which is called Community Self-Organizing Map (CSOM). The important feature of CSOM is that neurons create some neuron communities according to their winning frequency We have applied CSOM to the clustering and the data extraction for various input data for the clustering and the data extraction.

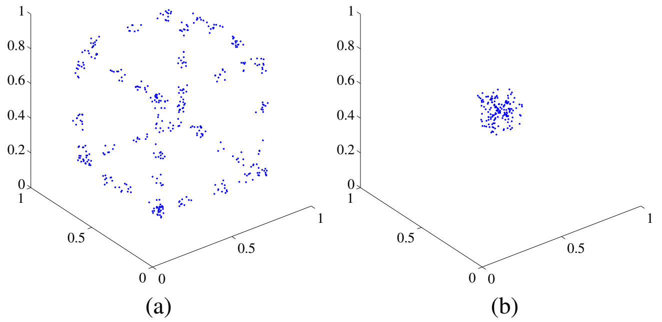

Fig. 13. Extraction results of cluster by using CSOM. (a) Cluster extracted by the largest community C_1. (b) Cluster extracted by the second largest community C_2.

We have confirmed that the number of communities created by CSOM is the same as the number of clusters, and the each community size depends on the number of the input data in the cluster and the shape of the cluster. We have confirmed the effectiveness of CSOM in the application to the cluster extraction.

REFERENCES

[1] T. Kohonen, *Self-Organizing Maps*, Berlin, Springer, vol. 30, 1995.

[2] Y. Cheng, "Clustering with Competing Self-Organizing Maps," *Proc. of IJCNN'92*, vol. IV, pp. 785-790, 1992.

[3] W. Wan and D. Fraser, "M2dSOMAP: Clustering and Classification of Remotely Sensed Imagery by Combining Multiple Kohonen Self-Organizing Maps and Associative Memory," *Proc. of IJCNN'93*, vol. III, pp. 2464-2467, 1993.

[4] J. Vesanto and E. Alhoniemi, "Clustering of the Self-Organizing Map," *IEEE Trans. Neural Networks*, vol. 11, no. 3, pp. 586–600, 2002.

[5] C. Derek and E. Adrian, "Finding Curvilinear Features in Spatial Point Patterns: Principal Curve Clustering with Noise," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 6, pp. 601-609, 2000.

[6] P. Doucette, P. Agouris and A. Stefanidis, "Self-Organized Clustering for Road Extraction in Classified Imagery," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 55, Issues 5-6, pp. 347-358, 2001.

[7] H. Matsushita and Y. Nishio, "Tentacled Self-Organizing Map for Effective Data Extraction," *Proc. International Neural Network Conference on Neural Networks*, pp. 1929-1936, 2006.

[8] M. Tomita, H. Matsushita and Y. Nishio, "Behavior of Fatigable SOM and its Application to Clustering," *Proc. International Neural Network Conference on Neural Networks*, pp. 6558-6563, 2006.

[9] A. Ultsch and H. P. Siemon, "Kohonen's Self Organizing Feature Maps for Exploratory Data Analysis," *Proc. International Neural Network Conference*, pp. 305-308, 1990.

[10] X. Zhang and Y. Li, "Self-organizing map as a new method for clustering and data analysis," *Proc. International Neural Network Conference on Neural Networks*, pp. 2448-2451, 1993.

[11] M. A. Kraaijveld, J. Mao, and A. K. Jain, "A nonlinear projection method based on Kohonen's topology preserving maps," *IEEE Trans. Neural Netws.*, vol. 6, no. 3, pp. 548-559, 1995.

[12] N. R. Pal and V. K. Eluri, "Two efficient connectionist schemes for structure preserving dimensionality reduction," *IEEE Trans. Neural Netws.*, vol. 9, no. 6, pp. 1142-1154, 1998.

[13] M. C. Su and H. T. Chang, "A new model of self-organizing neural networks and its application in data projection," *IEEE Trans. Neural Netws.*, vol. 12, no. 1, pp. 153-158, 2001.

[14] A. Ultsch, "Clustering with SOM: U*C," Proc. Workshop on Self-Organizing Maps, pp.75-82, 2005.