International Journal of Innovative Computing, Information and Control
Volume **5**, Number **4**, April **2009**

ICIC International ©2008 ISSN 1349-4198
pp. 1–ICONIP07-04

TASK SEGMENTATION IN A MOBILE ROBOT BY MNSOM AND CLUSTERING WITH SPATIO-TEMPORAL CONTIGUITY

Muhammad Aziz Muslim[1], Masumi Ishikawa[2], Tetsuo Furukawa[2]

[1]Dept. of Electrical Engineering
University of Brawijaya
Jl. MT Haryono 167, Malang, 65145, Indonesia
muh_aziz@brawijaya.ac.id

[2]Dept. of Brain Science and Engineering
Kyushu Institute of Technology
2-4 Hibikino, Wakamatsu, Kitakyushu 808-0196, Japan
{ishikawa, furukawa} @brain.kyutech.ac.jp

Received January 2008; revised June 2008

Abstract. *In our previous study, task segmentation was done by mnSOM, using prior information that winner modules corresponding to subsequences in the same class share the same label. Since this prior informatiom is not available in real situation, segmentation thus obtained should be regarded as the upper bound for the performance, not as a candidate for performance comparison. Present paper proposes to do task segmentation by applying various clustering methods to the resulting mnSOM, without using the above prior information. Firstly, we use the conventional hierarchical clustering. It assumes that the distances between any pair of modules are provided with precision, but this is not the case in mnSOM. Secondly, we used a clustering method based on only the distance between spatially adjacent modules with modification by their temporal contiguity. In the robotic field 1, the segmentation performance by the hierarchical clustering is very close to the upper bound for novel data. In the robotic field 2, the segmentation performance by clustering with the spatio-temporal contiguity is very close to the upper bound for novel data. Therefore, the proposed methods demonstrated their effectiveness in segmentation.*
Keywords: mnSOM, Task segmentation, Clustering, Mobile robot, Temporal contiguity, Spatio-temporal contiguity.

1. **Introduction.** Task segmentation in navigation of a mobile robot based on sensory signals is important for realizing efficient navigation, hence attracted wide attention. Tani and Nolfi [10] proposed 2-level hierarchical mixture of recurrent experts (MRE), which is an extension of the network architecture proposed by Jacobs et al.[3]. Wolpert and Kawato [12] proposed MOSAIC architecture for motor control with a responsibility signal to each module provided by the soft-max function.

In the conventional competitive learning, only a winner module or unit is highlighted, accordingly the degree of similarity between modules or units and interpolation among them are not taken into account. There are two types of "interpolation:" one is creating an output by interpolating outputs of multiple modules, and the other is creating a module by interpolating multiple modules. Let the former be called "output interpolation" and the latter be called "module interpolation." Our study here focuses on the latter.

The soft-max function [12] is an improvement over the conventional competitive learning in that the output interpolation is possible based on the responsibility signals. Similarity between modules, however, is not explicitly represented. Furthermore, the soft-max function and segmentation do not generally coexist; only when the soft-max function is asymptotically equivalent to winner-take-all, segmentation is possible at the sacrifice of interpolation.

Self Organizing Maps (SOM)[4] is a popular method for classification and visualization of data, and is capable of topology preservation. The resulting topological maps demonstrate the unit interpolation among neighboring units on a competitive layer of SOM. In contrast to SOM with a vector unit as its element, a modular network SOM (mnSOM) uses a function module as its element to increase its representation and learning capability [2]. Owing to competitive learning among function modules, mnSOM is capable of segmentation. Owing to topographic mapping of function modules on a competitive layer, neighboring function modules tend to have similar characteristics. Hence, interpolation among function modules becomes possible. Simultaneous realization of segmentation and interpolation is unique and unparalleled characteristics of mnSOM. mnSOM has also an advantage of computational stability in contrast to competitive learning due to careful assignment of learning rates to modules and classes.

We proposed to use mnSOM for task segmentation in navigation of a mobile robot [6][7]. In case of a mobile robot, however, the standard mnSOM is not applicable as it is, because it is based on the assumption that class labels are known a priori. In a mobile robot, only an unlabeled sequence of data is available. Hence, we proposed to decompose it into many subsequences, supposing that a class label does not change within a subsequence. Accordingly, training of mnSOM is done for each subsequence in contrast to that for each class in the standard mnSOM.

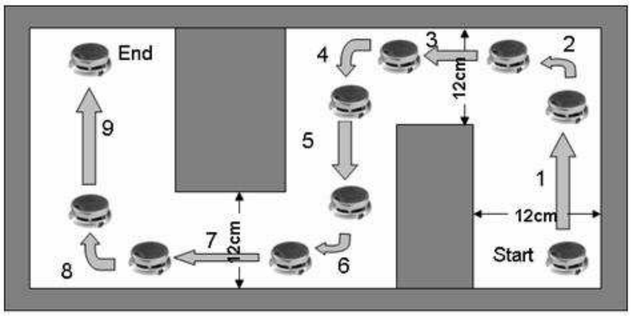
Although the conventional competitive learning can handle unlabeled sequence of data, mnSOM cannot handle them as will be shown later. Therefore, our previous studies [6][7] assumed availability of prior information that winner modules corresponding to subsequences in the same class share the same label. Since the above prior information is not available in real situation and is unrealistic, we propose to do task segmentation by clustering the resulting modules in mnSOM without using the above unrealistic prior information. Firstly, the conventional hierarchical clustering is applied. It assumes that the distances between any pair of modules are provided with precision. However, since mnSOM training adopts neighborhood learning as in SOM, the distance between a pair of far apart modules tends to be meaningless. Secondly, we use a clustering method based on the distance between only the spatially adjacent modules with modification by their temporal contiguity. This is what we call a clustering with spatio-temporal contiguity.

2. **Task Segmentation and Clustering.**

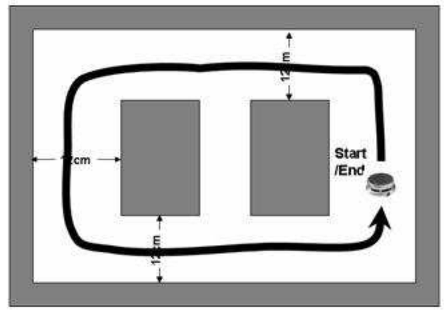
2.1. **Task Segmentation using mnSOM.**

2.1.1. *Data Segmentation.* Task segmentation, here, is to partition the entire movement of a robot from a start position to an end position into a sequence of primitive movements such as a forward movement or a right turn movement. Experiments are carried out using a Khepera II mobile robot. It has 8 infra-red (IR) proximity sensors for acquisition of information on the environment, and 2 separately controlled DC motors.

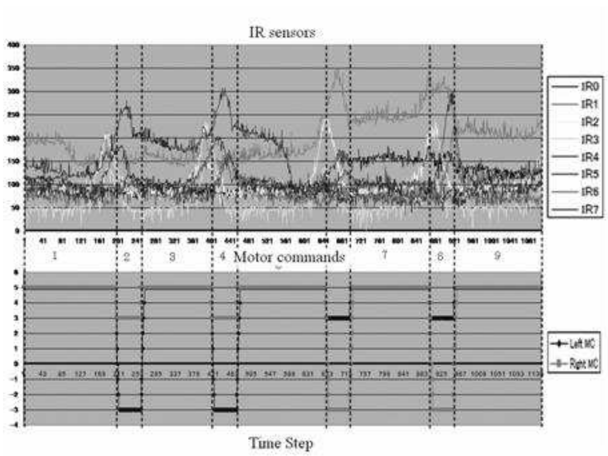
Robot movement is determined by wall following. In case of the robotic field in Figure 1.(a), task segmentation and environmental segmentation are similar with the exception of locally zigzag movement. Generally speaking, forward movement corresponds to straight corridor in a robotic field, right turn movement corresponds to L-shaped corner, and so forth. In contrast to this, in the robotic field in Figure 1.(b), task segmentation and environmental segmentation are different, particularly at T-junctions.



(a)



(b)



(c)

Sequence	Data Numbers	Subsequence Numbers	Labels
1	1-131	1,2,3,4,5, 6,7	F,F,F,F,F, F,L/F
2	132-187	7,8,9,10	L/F,L,L,L/F
3	188-312	10,11,12,13, 14,15,16	L/F,F,F,F,F, F,L/F
4	313-368	16,17,18,19	L/F,L,L,L/F
5	369-496	19,20,21,22, 23,24,25	L/F,F,F,F,F, F,R/F
6	497-549	25,26,27,28	R/F,R,R,R/F
7	550-666	28,29,30,31, 32,33,34	R/F,F,F,F,F, F,R/F
8	667-719	34,35,36	R/F,R,R/F
9	720-843	36,37,38,39, 40,41,42	R/F,F,F,F,F, F,F

(d)

FIGURE 1. (a) Robotic Field 1 (b) Robotic Field 2 (c) Data from the Robotic Field 1 (d) Example of data division in Robotic Field 1

Figure 1.(c) illustrates an example sequence of sensory-motor signals during movement in the robotic field 1. For later evaluation of training and test results, the whole dataset are manually segmented into 9 sequences based on motor commands as in Figure 1.(c). Sequences 1, 3, 5, 7 and 9 correspond to a class of forward movements, sequences 2 and 4 correspond to a class of left turns, and sequences 6 and 8 correspond to a class of right turns.

The whole dataset is split into subsequences with the uniform length of 20 as in Figure 1.(d) [6][7]. Figure 1.(d) and Figure 1.(a)provide approximate spatial segmentation of robotic field 1. Each subsequence has its own label. As a consequence of uniform splitting, some subsequences stretch over two consecutive sequences (e.g., a forward movement sequence followed by a left turn sequence). They are called "transition" subsequences, and constitute virtual classes.

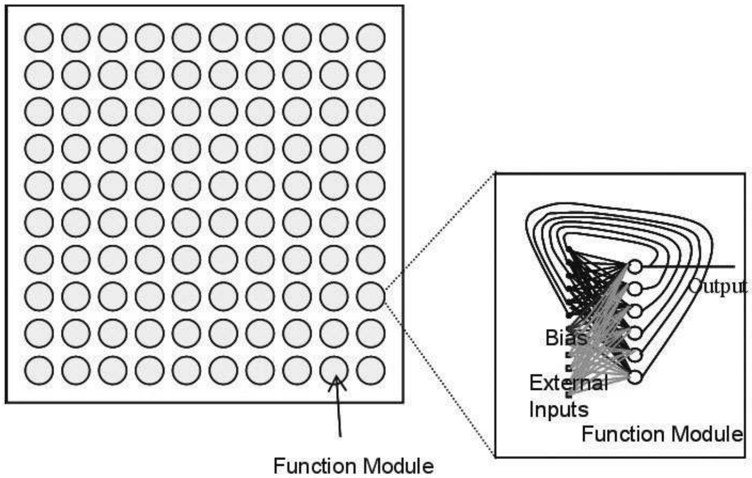


FIGURE 2. Array of modules in mnSOM and the function module as its element. The function module here is a fully connected RNN.

2.1.2. *The modified mnSOM.* The standard mnSOM deals with labeled data. In mobile robots, only a sequence of unlabeled data is available. Hence, we proposed to decompose it into subsequences [6][7]. Accordingly, training of mnSOM is done for each subsequence in contrast to that for each class in the standard mnSOM.

To deal with dynamical systems, recurrent neural networks (RNN) are employed as function modules in mnSOM [2]. Figure 2. illustrates the architecture of mnSOM and the function module as its element. Each mnSOM module is trained using backpropagation through time (BPTT) [11]. Accordingly, connection weights of module k, $\mathbf{w}^{(k)}$, are modified by [2],

$$\Delta w^{(k)} = \sum_{i=1}^{M} \Psi_i^{(k)}(t) \left(-\eta \frac{\partial E_i^{(k)}}{\partial w^{(k)}} \right) \tag{1}$$

where M is the number of subsequences, t is the iteration number in mnSOM learning, $E_i^{(k)}$ is the output error of the k-th module for the i-th subsequence, and $\Psi_i^{(k)}(t)$ is the learning rate of the k-th module for the i-th subsequence. The learning rates are carefully determined by the following normalized neighborhood function:

$$\Psi_i^{(k)}(t) = \frac{\phi\left(r\left(k, v_i^*\right); t\right)}{\sum_{i'=1}^{M} \phi\left(r\left(k, v_i^*\right); t\right)} \tag{2}$$

Neighborhood size decreases as time increases as follows,

$$\phi(r;t) = \exp\left[-\frac{r^2}{2\sigma^2(t)}\right] \tag{3}$$

$$\sigma(t) = \sigma_{min} + (\sigma_{max} - \sigma_{min}) e^{\frac{-t}{\lambda}} \tag{4}$$

where $r(k, v_i^*)$ stands for the distance between module k and the winner module v_i^*, ϕ is a neighborhood function, σ_{min} is the minimum neighborhood size, σ_{max} is the maximum neighborhood size, and λ is a neighborhood decay rate. These learning rates contribute to improvement of computational stability. mnSOM terminates when connection weights converge and the resulting mnSOM becomes stable.

2.2. **Difficulty in the standard mnSOM.** Figure 3.(a) illustrates the resulting map by the standard mnSOM. No information is available on the relation between modules with different colors. It indicates that segmentation based on Figure 3.(a) generates 23 classes, which is meaningless. Because of this, our previous studies [6][7] assumed availability of prior information that winner modules corresponding to subsequences in the same class share the same label. However, the prior class information is unavailable in real situation and is unrealistic.

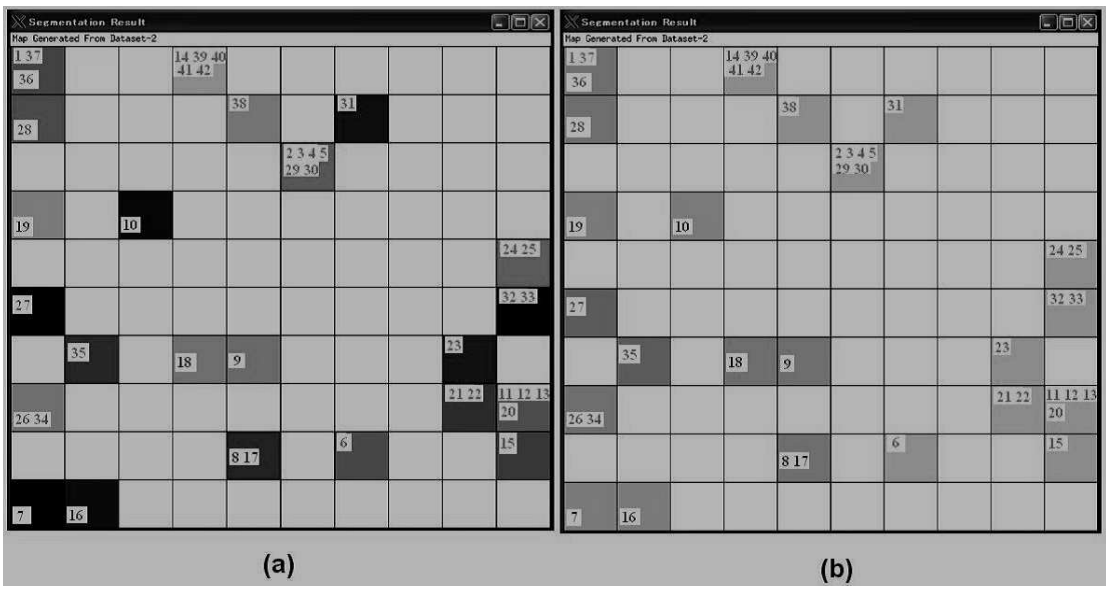


FIGURE 3. The resulting mnSOM with each color representing a data class. (a)Segmentation by the standard mnSOM. The resulting number of classes is 23, and is too large. (b) Segmentation by mnSOM with the assumption that winner modules corresponding to subsequences in the same class share the same label.

2.3. **Clustering.** In this paper, we propose to do segmentation using clustering methods based on the distance between modules in the resulting mnSOM. In contrast to SOM, the definition of the distance between modules is problematic, because the distance depends on input to these modules.

2.3.1. *Hierarchical Clustering.* A procedure of hierarchical clustering [1] is the following.

1. Let each module form a separate cluster.
2. Merge two clusters with the minimum distance.
3. Recalculate the distance between clusters.
4. Repeat steps 2 and 3 until the minimum distance between clusters exceeds a given threshold or the number of clusters reaches a given number of clusters.

An essential issue in clustering is how we define the distance between modules. Suppose

$$m_i = \arg\min_k MSE(k,i) \tag{5}$$

where $MSE(k,i)$ stands for the mean square error of module i given input subsequence k. We propose to define the distance between modules i and j by:

$$d_{ij} = \sqrt{(MSE(m_i,j) - MSE(m_i,i))^2 + (MSE(m_j,i) - MSE(m_j,j))^2)} \tag{6}$$

The inclusion of only the subsequences m and n in the definition is to prevent the distance from being blurred by many less relevant subsequences.

We then define the distance between clusters I and J. Suppose that the cluster I is composed of modules, $M_{I1}...M_{IR_I}$, and the cluster J is composed of modules, $M_{J1}...M_{JR_J}$. The distance between these two clusters is defined by,

$$D_{IJ} = \frac{1}{R_I R_J} \sum_{i=1}^{R_I} \sum_{j=1}^{R_J} d_{ij} \tag{7}$$

where d_{ij} is the distance between two individual modules i and j as in Eq.(7).

2.3.2. *Clustering with spatial contiguity.* In mnSOM the neighboring area shrinks as learning proceeds. This suggests that the distance between modules are meaningful only within neighboring modules. On the other hand, hierarchical clustering assumes that the distance between any pair of modules is meaningfully given. Considering this issue, we propose the following clustering method with spatial contiguity.

1. Calculate the distance between any pair of adjacent modules. For module (i,j), adjacent modules are (i,j-1), (i,j+1), (i-1,j) and (i+1,j).
2. Rank order distances between adjacent modules in increasing order.
3. Merge a pair of adjacent modules with the minimum distance.
4. Calculate the number of clusters formed by the merger.
5. Repeat steps 3 and 4 until the predefined number of clusters is obtained.

2.3.3. *Clustering with spatio-temporal contiguity.* In mobile robot data, temporally contiguous subsequences tend to have the same label. Accordingly, winner modules corresponding to temporally contiguous subsequences tend to have the same label. To take the temporal contiguity into account, we propose to modify Eq.(6) as follows,

$$d_{ij} = \sqrt{(MSE(m_i,j) - MSE(m_i,i))^2 + (MSE(m_j,i) - MSE(m_j,j))^2} * \left(1 - \exp\left(-\frac{|m_i - m_j|}{\tau}\right)\right) \tag{8}$$

where τ is a time constant for temporal contiguity, and m_i and m_j are subsequence numbers. In contrast to Eq. (6), the second term on the right-hand side of Eq. (8) reduces the distance between modules by taking into consideration the temporal contiguity of the corresponding subsequences. This modified definition of the distance is expected to provide the same label to temporally contiguous subsequences.

3. Experimental Results.

3.1. Task Segmentation. mnSOM modules learn internal models of nonlinear dynamics of robot-environment interaction by minimizing mean prediction error of sensory-motor signals at the next time step, given the past sensory-motor signals. After training, the resulting mnSOM provides a label to each module by a procedure in [6][7] taking advantage of prior information, that winner modules corresponding to subsequences in the same class share the same label. Given a subsequence, either experienced or novel, one of the modules becomes a winner. The label of the winner module provides task segmentation for each subsequence.

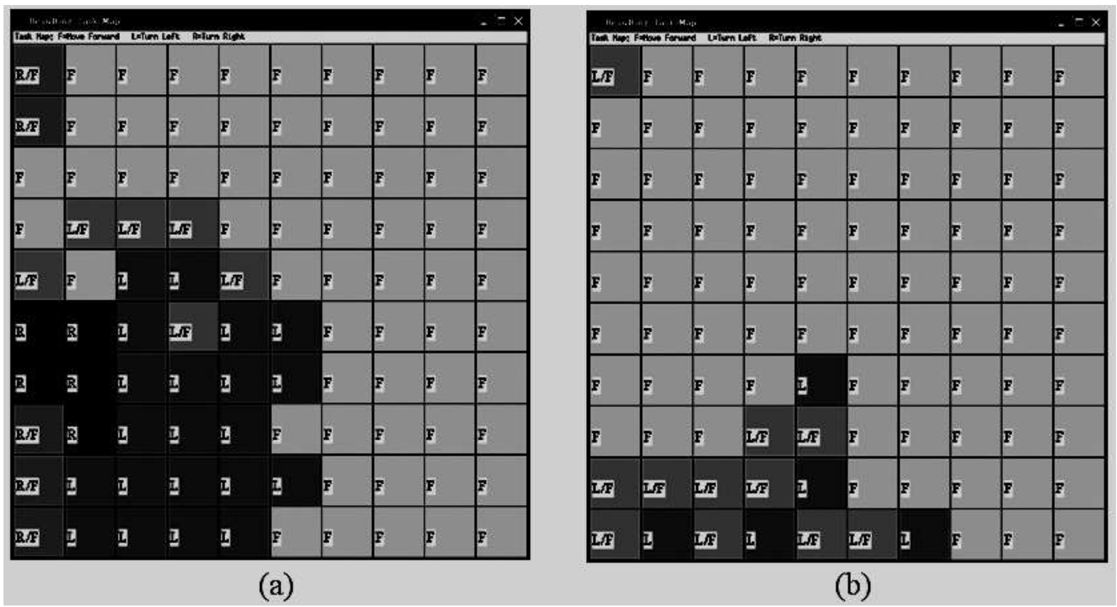


FIGURE 4. Resulting Task Map: (a) for robotic field 1, (b) for robotic field 2. Labels "F", "L", "R", "L/F", and "R/F"stand for forward movement, left turn, right turn, the transition between forward movement and left turn, and the transition between forward movement and right turn, respectively.

Figure 4. depicts the resulting task maps for robotic field 1 and robotic field 2. To evaluate the segmentation performance of the task map, training datasets as well as novel dataset are given to them.

Figure 5. illustrates the resulting labels for test subsequences for robotic field 1 and robotic field 2. The numbers written in the mnSOM module are subsequence numbers assigned by the corresponding mnSOM module. Figure 5.(a) and Figure 1.(d) gives approximate relationship between location of the robot and the corresponding winner module. It is to be noted that the result should not be regarded as a candidate for performance comparison, because it uses unrealistic prior information which is not available in real situation. The result, therefore, should be regarded as the upper bound for segmentation performance.

3.2. Clustering. We propose to do task segmentation by applying various clustering methods to the resulting mnSOM without using prior information that winner modules

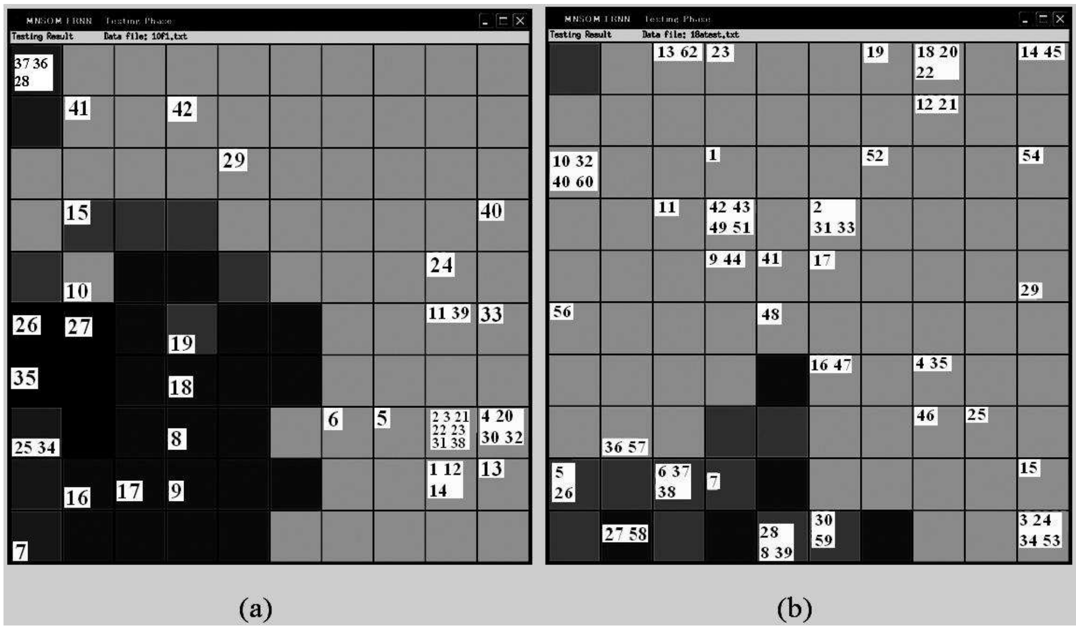


FIGURE 5. Resulting labels for novel subsequences based on mnSOM (a) for robotic field 1, (b) for robotic field 2.

corresponding to subsequences in the same class share the same label. Figure 6. illustrates the resulting segmentation of novel dataset by hierarchical clustering for robotic field 1 and robotic field 2. The task maps in Figure 6 are similar to those by mnSOM in Figure 4 to some extent.

Figures 7 and 8 indicate that proper value of τ shifts some winner modules corresponding to adjacent subsequences (e.g. subsequence 16 and 17 in Figure 8) into the same cluster, and changes cluster boundary.

Table 1 gives summary of segmentation performance by various clustering methods in addition to the upper bound for the segmentation performance. It is the correct segmentation rate by mnSOM using prior information. Since this prior information is unavailable in real situation, this should be regarded as the upper bound for the segmentation performance, not as a candidate for performance comparison.

In clustering with spatio-temporal contiguity, the performance of clustering depends on the time constant parameter, τ, in Eq. (9). τ=0 corresponds to clustering with spatial contiguity and positive values of τ correspond to clustering with spatio-temporal contiguity. Table 1 indicates that the performance is the best at τ=7 for the robotic field 1, while the performance is the best at τ=19 for robotic field 2. In robotic field 1, the performance of the hierarchical clustering is superior to that of clustering with spatio-temporal contiguity. In robotic field 2, the performance of clustering with spatio-temporal contiguity is superior to that of the hierarchical clustering. The reason for this is left for future study.

4. **Conclusions and Discussions.** In this paper we proposed to apply various clustering methods to the resulting mnSOM for task segmentation. This is to get rid of the unrealistic prior information that winner modules corresponding to subsequences in the same class

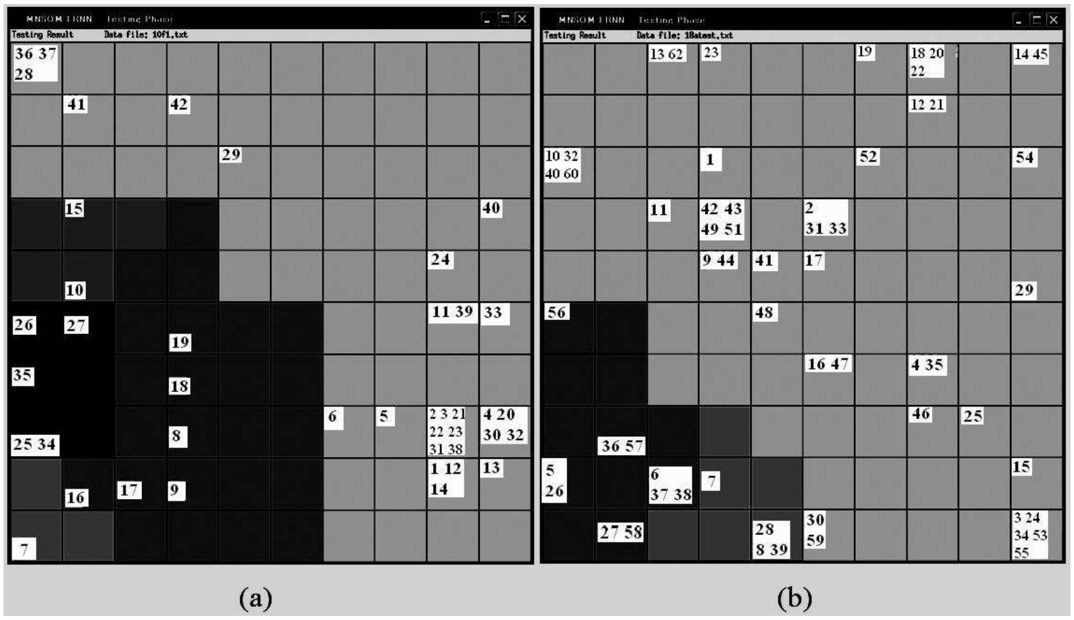


FIGURE 6. The Resulting Segmentation for novel subsequences by Hierarchical Clustering: (a) robotic field 1, (b) robotic field 2.

TABLE 1. Correct Segmentation rate (%) by various Clustering Methods. "upper bound" stands for the correct segmentation rate by mnSOM with prior information. "Tr1", "Tr2", "Tr3", "Tr4" stand for training dataset 1, 2, 3 and 4, respectively. "Ave" stands for the average over 4 datasets. "Novel" stands for novel dataset.

Robotic Field	Data-set	upper bound	Hierar-chical	Spatio-temporal contiguity					
				$\tau \approx 0$	τ=2	τ=7	τ=11	τ=15	τ=19
1	Tr1	94.4	85.71	86.9	86.9	88.1	78.6	67.9	67.9
	Tr2	96.4	85.71	82.1	82.1	84.5	67.9	66.7	52.4
	Tr3	94.0	91.67	78.6	78.6	83.3	71.4	75.0	54.8
	Tr4	100	90.48	80.9	80.9	83.3	63.1	65.5	53.6
	Ave	96.2	88.4	82.1	82.1	84.8	70.3	68.8	57.1
	Novel	94.0	92.9	83.3	83.3	86.9	82.1	70.2	67.9
2	Tr1	97.6	88.7	86.3	86.3	94.4	91.1	91.1	93.6
	Tr2	96.0	88.7	83.1	83.1	86.3	86.3	86.3	91.1
	Tr3	99.2	85.5	91.1	91.1	92.3	92.7	92.7	90.3
	Tr4	98.4	91.1	87.1	87.1	89.5	89.5	89.5	89.5
	Ave	97.8	88.5	86.9	86.9	90.6	89.9	89.9	91.1
	Novel	95.2	92.7	80.6	80.6	87.9	87.9	87.9	93.6

share the same label. Since this prior information is not available in real situation, this should be regarded as the upper bound for segmentation performance, not as a candidate for performance comparison.

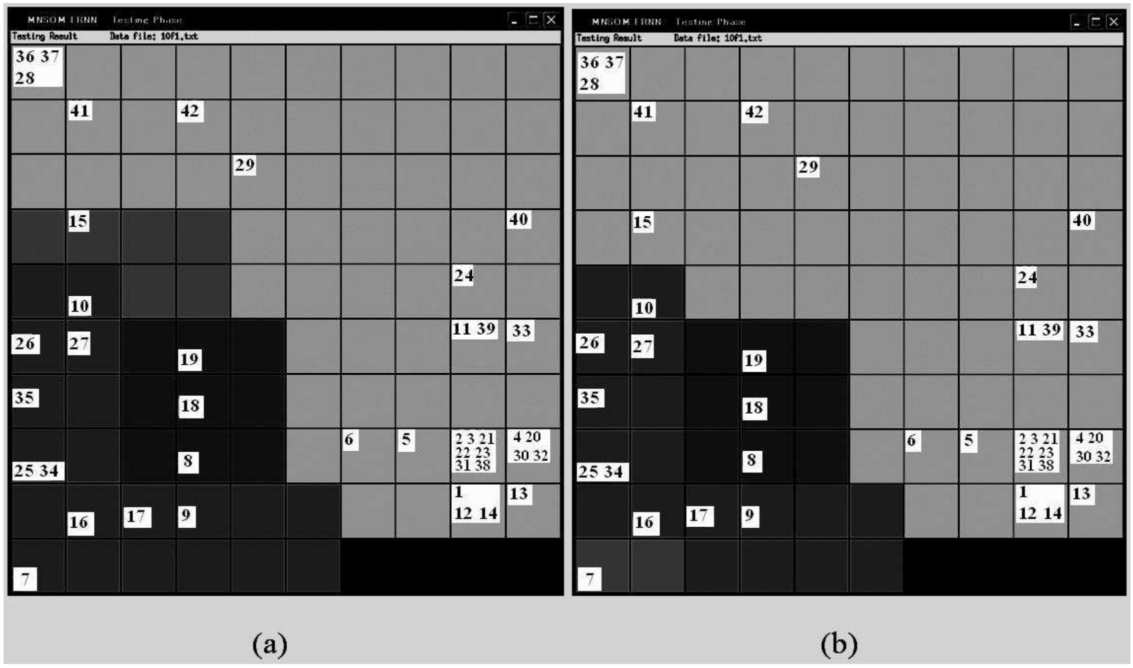


FIGURE 7. Resulting Segmentation by Clustering with Spatio-temporal Contiguity for Robotic Field 1, (a) τ=2, (b) τ=7

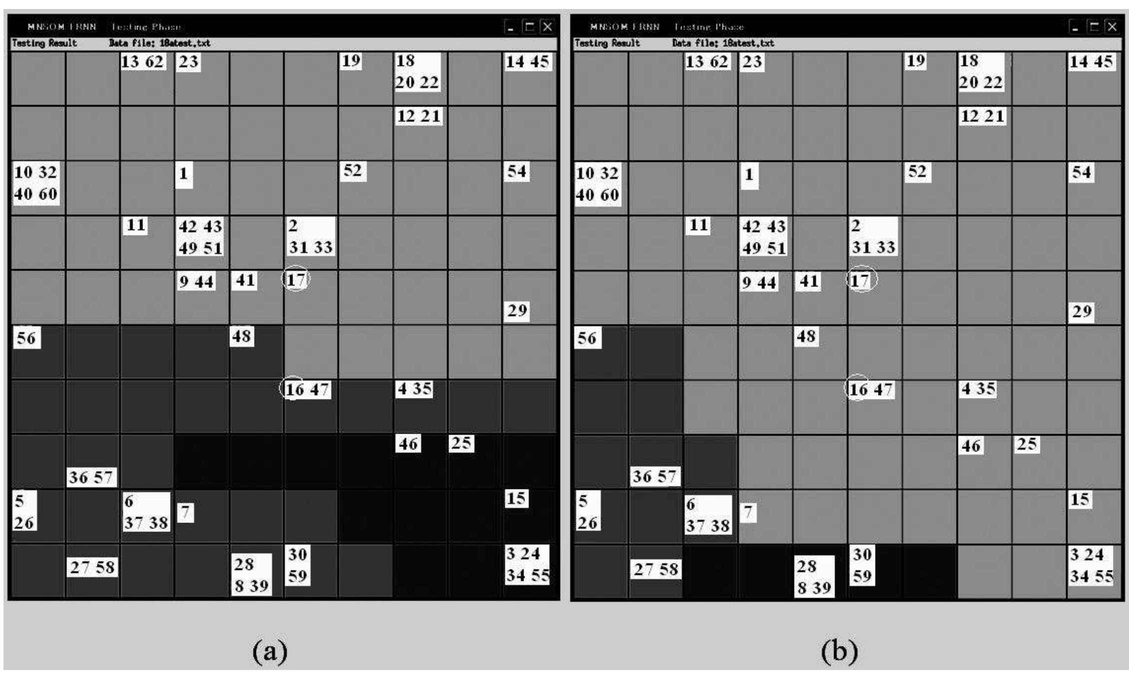


FIGURE 8. Resulting Segmentation by Clustering with Spatio-temporal Contiguity for Robotic Field 2, (a) τ=2, (b) τ=19. Subsequences 16 and 17 (circled) which are in different clusters in (a) move into the same cluster in (b)

Firstly, we proposed to use the conventional hierarchical clustering. This supposes that the distances between any pairs of modules are provided with precision, but this is not the case in mnSOM. Secondly, we proposed to use a clustering method based on the distance between only the spatially adjacent modules with modification by their temporal contiguity.

In the robotic field 1, the segmentation performance by the hierarchical clustering is very close to the upper bound for novel data. In the robotic field 2, the segmentation performance by clustering with the spatio-temporal contiguity is very close to the upper bound for novel data. Therefore, the proposed methods demonstrated their effectiveness of segmentation. However, segmentation performance for training data is significantly lower than the upper bound. The improvement of this is left for future study.

Acknowledgment. This research was partially supported by the 21st Century COE (Center of Excellence) Program and by Grant-in-Aid for Scientific Research(C)(18500175) both from the Ministry of Education, Culture, Sports, Science and Technology(MEXT), Japan.

REFERENCES

[1] Duda, R.O., P.E. Hart, and D.G. Stork, Pattern Classification, *Wiley-Interscience*, 2001

[2] Furukawa, T., K. Tokunaga, S. Kaneko, K. Kimotsuki, and S. Yasui, Generalized self-organizing maps (mnSOM) for dealing with dynamical systems, *Proc. of NOLTA2004*, Fukuoka, Japan, pp. 231-234, 2004

[3] Jacobs, R., M. Jordan, S. Nowlan, and G. Hinton, Adaptive Mixtures of Local Experts, *Neural Computation*, 3, pp. 79-87, 1991

[4] Kohonen, T., *Self-Organizing Maps*, Springer, 1995

[5] Kohonen,T., Self organized formation of topologically correct feature maps, *Biological Cybernetics*, 43, pp. 59-69, 1982

[6] Muslim, M.A., M. Ishikawa, and T. Furukawa, A New Approach to Task Segmentation in Mobile Robots by mnSOM, *Proc. of 2006 IEEE WCCI (IJCNN2006 Section)*, Vancouver, Canada, pp. 6542-6549, 2006

[7] Muslim, M.A., M. Ishikawa, and T. Furukawa, Task Segmentation in a Mobile Robot by mnSOM : A New Approach To Training Expert Modules, *Neural Computing and Application*, vol. 16, pp. 571-580, Springer, 2007

[8] Muslim, M.A., M. Ishikawa, and T. Furukawa, Task Segmentation in a Mobile Robot by mnSOM and Hierarchical Clustering, *International Workshop on Self-Organizing Maps (WSOM) 2007*, Bielefeld, Germany, 2007

[9] Pawelzik, K., K. Kohlmorgen, and K.R. Muller, Annealed Competition of Expert for a Segmentation and Classification of Switching Dynamic, *Neural Computation*, 8(2), pp. 340-356, 1999

[10] Tani, J., and S. Nolfi, Learning to perceive the world as articulated: an approach for hierarchical learning in sensory-motor systems, *Neural Networks*, 12, pp. 1131-1141, 1999

[11] Williams, R.J., and D. Zipser, Gradient-based learning algorithms for recurrent networks and their computational complexity, *Backpropagation: Theory, Architectures and Applications*, Erlbaum, pp. 433-486, 1992

[12] Wolpert, D.M., and M. Kawato, Multiple paired forward and inverse models for motor control, *Neural Networks*, Vol.11, pp. 1317-1329, 1998