A Hand Image Instruction Learning System Using PL-G-SOM

T. Kuremoto[1], T. Otani[1], L.-B. Feng[2], K. Kobayashi[1, 3], and M. Obayashi[1]
[1]Gradual School of Science and Engineering, Yamaguchi University, Ube, Yamaguchi, Japan
[2]Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences, Shenzhen, Guangdong, China
[3]School of Information Science and Technology, Aichi Prefectural University, Nagakute, Aichi, Japan

Abstract - *Conventionally, human instructions to a robot are often given by previously designed signals such as voices and images. In this study, one's own "shapes of a hand" is suggested to be instructions in the human-machine interaction system. We proposed a Self-Organizing Map (SOM) with a memory layer named Transient-SOM (T-SOM) and adopted it to a hand image instruction learning system. In this study, instead of T-SOM, an improved SOM, Parameter-Less Growing Self-Organizing Map (PL-G-SOM) is used to improve the hand image instruction learning system. In order to verify the performance of the proposed system, comparison experiments were executed and the results showed the priorities of the new system.*

Keywords: *Human-machine interaction, Self-Organizing Map (SOM), Parameter-Less Growing SOM, Hand image instruction, Pattern recognition*

1 Introduction

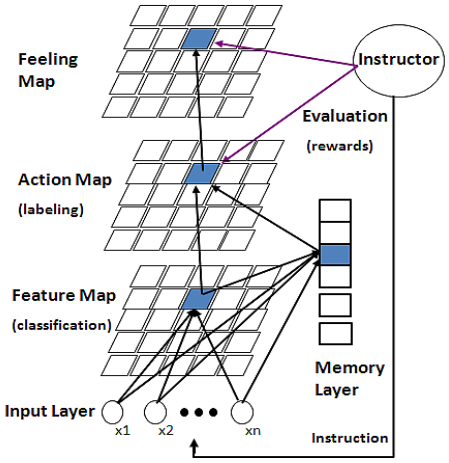
Hand gesture has been introduced to human-machine interaction since the end of last century (Pavlovic et al., 1997). Static images of hand shapes and dynamical videos of hand gestures are recognized by different mathematical models such as Multi-Layer Perceptron (MLP) (Rumelhart et al., 1986), Self-Organizing Map (SOM) (Kohonen, 1982, 1995, 1998), Hidden Markov Model (HMM) (Baum & Petrie, 1966) and so on.

Kohonen's SOM, as a well-known pattern recognition neural network, is a powerful tool to categorize high dimension data to one or two dimension space. In our previous studies, we proposed two kinds of improved SOMs, i.e., Transient SOM (T-SOM) (Kuremoto et al., 2006) which introduced a memory layer to reserve the matured "best much unit" (BMU), and Parameter-Less Growing SOM (PL-G-SOM) which combined the concepts of Berglund & Sitte's Parameter-Less SOM (PLSOM) (Berglund & Sitte, 2006) and Growing SOM (GSOM) (Bauer & Villmann, 1997; Villmann & Bauer, 1998; Dittenbach et al., 2000) to overcome the limitation of the size of map, the collapse of map' topology due to the unlearned data, and reduced the load of computation.

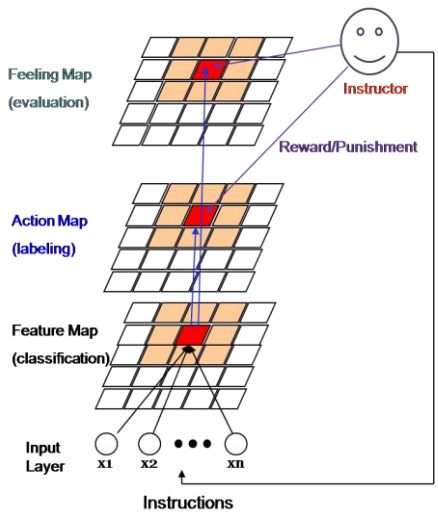
T-SOM has been applied to a hand image instruction learning system for a pet robot "AIBO" (Sony Ltd. Product 2003) successfully (Kuremoto et al., 2006; Hano et al., 2007) and PL-G-SOM succeeded as a voice instruction learning system for the robot (Kuremoto et al., 2010, 2011). In this research, we adopt PL-G-SOM into the hand image instruction learning system instead of T-SOM.

2 The structure of hand image instruction learning systems

The hand image instruction learning systems for partner robots, which are intelligent robots with the abilities of human-machine interaction, have a similar architecture as shown as in Figure 1. Hand images are preprocessed to yield their feature vectors as the input of systems. Three kinds of maps which are Feature Map, Action Map and Feeling Map are combined to realize input data classification, action selection and success rate expression respectively. Feature Map is a layer of T-SOM or PL-G-SOM, it clustering high dimension input data on 2-dimension spaces of maps. The input data of hand image instruction are feature vectors of hand shapes obtained by preprocessed images.



(a) T-SOM



(b) PL-G-SOM

Figure 1: Structures of instruction learning systems: (a) T-SOM (Kuremoto et al., 2006, Hano et al., 2007) for hand image instructions; (b) PL-G-SOM for voice instructions (Kuremoto et al., 2010, 2011), and used in this study for hand image instructions.

2.1 Feature Vector Space of Hand Image Data

2.1.1 Image Processing for hand Extraction

The hand image instruction system is designed for a partner robot learning to select different designed actions according to the different shape of a hand of its instructor. Hand area, i.e., skin area in the image captured by a CCD camera needs to be extracted and regularized at first. For a frame of image in RGB format, it is transformed to HSV format at first, then, using the threshold values of Hue (H), and Saturation (S) (Sherrah & Gong, 2001) and Red (R) threshold in RGB, skin area is extracted as a binary image. Noise elimination and holes filling are also effective to segment a hand area from the binary image. The thresholds for skin of a yellow race people in the room of fluorescent lights (around 500lx) are given as follows as we investigated:

1) When H, $S \in [0, 360]$ degree,
If $10 \le S < 15$, then $H > 350$;
If $15 \le S < 20$, then $H > 330$;
If $20 \le S < 30$, then $H > 300$ or $H < 40$;
If $30 \le S < 50$, then $H > 250$ or $H < 30$;
If $50 \le S < 70$, then $H > 230$ or $H < 30$;
If $70 \le S < 150$, then $H > 220$ or $H < 40$;
If $S < 10$ or $150 \le S \le 360$, then $H > 300$ or $H < 40$;
2) When R, G, $B \in [0, 255]$,
$30 < R < 250$.

2.1.2 Feature Space of Hand Shape

The instructions given by the instructor of robot are supposed as the different shapes of a hand. To distinguish the type of a hand shape, feature space definition is important to result high rate of pattern recognition. We discussed the methods of feature space construction in our previous works (Kuremoto et al., 2006; Hano et al., 2009) and proposed a useful feature vector space of hand shapes. The input images are analyzed by an 80-dimension vector space (See Fig. 2). From the origin of the space to the end of the hand area, the lengths in axes each 1.8-degree increased (80 axes) are the values of the feature vector, i.e., (x_1, x_2, …, x_{80}).

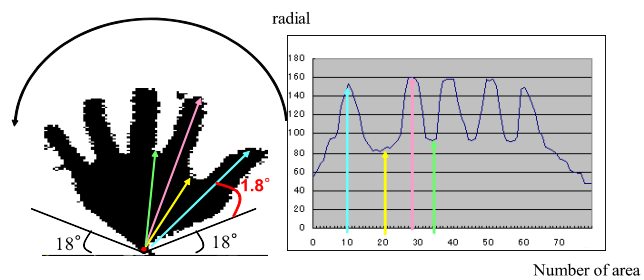


Figure 2: Input space of T-SOM and PL-G-SOM, constructed by an 80-dimension feature vector space of a hand instruction as a binary image. Left: a regularized image; Right: feature (input) space for Feature Map where horizontal axis is the dimension of vectors vertical axis is the value of vectors.

2.2 Action Map

The instructor presents his/her instructions with the different shapes of his/her hand to a robot, and in the view of the robot, hand shapes which are observed mean a state of the environment s_t, the robot intends to select a valuable action a_t (i) adapting to the state, $i = 1, 2, \ldots, A$, according to a stochastic action policy π, which is according to Gibbs distribution (Boltzmann distribution) as shown by Eq. (1).

$$\pi_t(a_t(i) \mid s_t) = \frac{e^{\frac{Q_t(s_t, a_t(i))}{T}}}{\sum_{j=1}^{A} e^{\frac{Q_t(s_t, a_t(j))}{T}}}, \qquad (1)$$

where T is a parameter named "temperature" which comes from the physical state description of a system (higher temperature lower possibility), t is the iteration time of learning, A is the number of available actions. When an action is selected according to Eq. (1) and performed by the robot, its

instructor evaluates the action by giving a reward/punishment r to robot. The reward is accepted and used to modify the value of Q_t by Eq. (2), where Q_t is called "state-action value function" in reinforcement learning (RL) (Sutton & Barto, 1998).

$$Q_{t+1}(s_{t+1}, a_{t+1}(i)) = Q_t(s_t, a_t(i)) + r \quad . (2)$$

Where s means the state of environment observed by the robot, a is the action selected by the learner, r is the reward (scalar) given by the instructor.

2.3 Feeling Map

To express the degree of how an instruction is learned by robot, a Feeling Map which has the same number of units with Action Map is designed as shown in Figure 1. Feeling Map expresses instruction recognition rate, i.e., the feeling of robot: more successful, happier it is. Feelings of partner robots, such as pet robots, entertainment robots, caring robots and so on, are important for human-machine interaction (HMI) when they are able to express vividly by their face expressions or the gestures (Kuremoto et al., 2007). The distance between input pattern and units on Feature Map and the reward from instructor are used to calculate feeling values which is normalized in [-1.0, 1.0] where high positive value means happiness and 0.0 is the initial value of each unit here. The calculation of Feeling Map is given by Eq. (3).

$$F_{t+1}(i) = F_t(i) \pm aC - bD_i, \qquad (3)$$

where $F(i)$ notes the feeling value of unit i on the Feeling Map (zero initially), C notes the continue times of reward or punishment, D_i is the Euclidean distance (squared error) between the unit i on Feature Map and the input data, a, b are constants, and $0 < a < 1,\ 0 < b << 1$.

3 T-SOM

The algorithm of T-SOM (Kuremoto et al., 2006, Hano et al., 2007) and whole system processing is shown as follows:

–Step 1: Initialization. Choose random values (0.0, 1.0) for unit m_i of a *2-dimension* map corresponding to an *n-dimension* input space. $(i = 1,2,\ldots,N \times M)$

– Step 2: Input data. Present a training sample $\mathbf{x}(x_1, x_2 \ldots, x_n)$ to the Input Layer.

– Step 3: Find *BMU*, i.e., the best match unit of Memory Layer or Feature Map. A *BMU c* is decided by using minimum Euclidean distance criterion Eq. (4).

$$\boldsymbol{c} = \arg\min_i (\| \mathbf{x} - \mathbf{m}_i \|), \qquad (4)$$

where $\mathbf{x}$ is input feature vector $(x_1, x_2, \ldots, x_n)$. If *BMU c* is found from Memory Layer, then Feature Map (Step 4) is skipped.

– Step 4: Competitive learning. Using a learning rule given by Eq. (5) to update the value of $\mathbf{m}_i$.

$$\triangle \mathbf{m}_i = \alpha h_{ci} (\mathbf{x} - \mathbf{m}_i), \qquad (5)$$

where α is a learning rate and h_{ci} is a neighborhood function given by Eq. (6).

$$h_{ci} = \exp\left(-\frac{\| \mathbf{c}_i - \mathbf{c} \|^2}{2\sigma^2}\right) . \qquad (6)$$

Here, $\boldsymbol{c_i}, \boldsymbol{c}$ denotes the positions of an arbitrary unit on the output map and *BMU*, respectively, $i = 1, 2, \ldots, k \le N \times M$, σ is a constant. Obviously, $h_{ci}(x) \ge 0$, $h_{ci}(0) = 1$, $h_{ci}(\infty) = 0$.

– Step 5: Vector quantization (labeling). After sufficient iterations of Step 3 and Step 4, i.e., if the distance from a *BMU* to the input is less than a threshold value, the input pattern is classified to be a unit of Action Map. This process is as same as LVQ-I (Kohonen, 1995), but labeling those units of Action Map is executed by a reinforcement learning algorithm described in Step 6.

– Step 6 Action learning. Using a reinforcement learning algorithm described in next subsection, robot select "correct actions" according to the reward or punishment from its instructor. The details are described in section 2.2.

– Step 7: Feeling formation. Units on Feeling Map are the levels of learning for actions on Action Map. The details are described in section 2.3.

– Step 8: Additional learning. For new instruction learning, T-SOM stores the succeeded unit weights into Memory Layer, and reset the units of Feature Map into random value. Additional learning or refresh learning then is able to repeat from Step 1.

T-SOM overcomes the limitation of a size fixed map of classic SOM and showed its efficiency as a real robot internal model to learn hand image instruction (Kuremoto et al., 2006; Hano et al., 2009), however, after a matured "BMU" is stored in the Memory Layer and the unit on Feature Map is refreshed with random weights of connections to the input, the distance between units of trained map and untrained input pattern (new data) shows disordered as indicated by Berglund & Sitte (Berglund & Sitte, 2006).

4 PL-G-SOM

Combine the idea of Growing SOMs (GSOM) (Bauer & Villmann, 1997, Dittenbach, Merkl & Rauber 2000) and the Parameter-Less SOM (Berglund and Sitte, 2006) together, we proposed a novel SOM named PL-G-SOM (Kuremoto et al., 2010, 2011) to realize additional learning, optimal

neighborhood preservation, and automatic tuning of parameters and applied it to a voice instruction learning system.

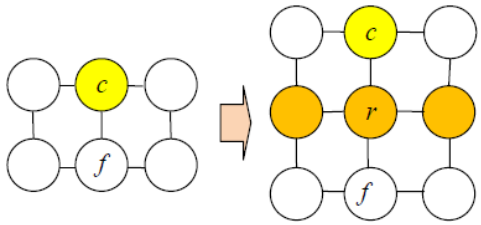


Figure 3: Insert a row/column into the feature map. Unit *c* is a *BMU* and *f* is the farthest unit among the neighbors of *c*, *r* the inserted row/column (Kuremoto et al., 2010).

Instead of the Memory Layer in T-SOM, Parameter-Less Growing Self-Organizing Map (PL-G-SOM) sets a small size of the feature map initially, and when a new input is not able to find a *BMU* from the initial map, that is, the distance between the input and the *BMU* ($\| \mathbf{x} - \mathbf{m}_i \|$) is larger than a threshold, a new row/column is inserted in to enlarge the feature map. For example, in Fig. 3, a new node *r* in the new row/column is inserted into the middle of node *c* and node *f*, where *c* is the nearest node to the new input and *f* is the neighbor of *c*. The weight of connection between the input and the new node has an average value of *c* and *f*,

$$\mathrm{m_r} = 0.5(\mathrm{m_c} + \mathrm{m_f}) \ , \qquad (7)$$

for nodes which are *r*'s neighbors in the new row/column:

$$\mathrm{m_{r+l}} = 0.5(\mathrm{m_{c+l}} + \mathrm{m_{f+l}}) \ , \qquad (8)$$

where *l=1, 2, …, N* or *M*. Unit *f* is chosen which has a largest Euclidean distance from the *BMU*-like *c* among the neighbors of *c*, and after this process, the map size changes to $N\times(M+1)$, or $(N+1)\times M$.

Following adaptive learning rate and neighborhood function are used in PL-G-SOM:

$$\Delta\mathbf{m}_i(t) = \varepsilon(t) h_{ci}^{\varepsilon}(t)(\mathbf{x}(t) - \mathbf{m}_i(t)) \ , \qquad (9)$$

$$\varepsilon(t) = \frac{\|\mathbf{x}(t) - \mathbf{m_c}(t)\|^2}{r(t)} \ , \qquad (10)$$

$$r(t) = \max(\|\mathbf{x}(t) - \mathbf{m_c}(t)\|^2, r(t-1)) \ , \qquad (11)$$

$$r(0) = \|\mathbf{x}(0) - \mathbf{m_c}(0)\|^2 \ , \qquad (12)$$

$$h_{ci}^{\varepsilon}(t) = \exp(-\frac{\|\mathrm{c}_i - \mathrm{c}\|^2}{(\sigma(\varepsilon(t)))^2}) \ , \qquad (13)$$

$$\sigma(\varepsilon(t)) = \sigma_{\max} \cdot \varepsilon(t), \ \sigma(\varepsilon(t)) \geq \sigma_{\min} \ . \qquad (14)$$

Here, $\varepsilon(t)$ is an adaptive learning rate, and $h_{ci}^{\varepsilon}(t)$ is a neighborhood function, $\sigma(t)$ is the neighborhood size. All of them are calculated by the distance between input and the BMU. $\sigma_{\max}, \sigma_{\min}$ are positive parameters, for example, the value may be the size of the map and 1.0, respectively. $\boldsymbol{c_i}, \boldsymbol{c}$ denotes the positions of an arbitrary unit on the output map and *BMU*, respectively, *i=1, 2, …, k* $\leq$ $N\times M$. Obviously, $h_{ci}(x) \geq 0$, $h_{ci}(0) = 1$, $h_{ci}(\infty) = 0$.

Table 1: Parameters used in the experiments of SOM, T-SOM and PL-G-SOM.

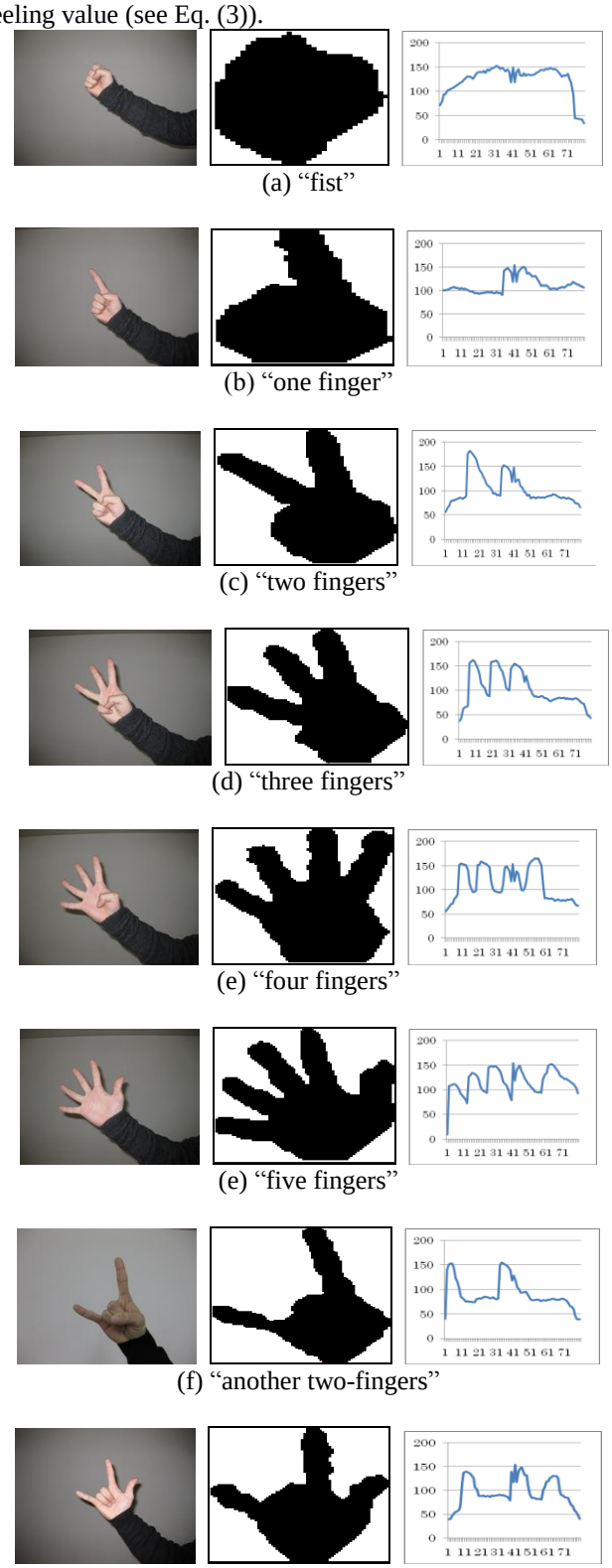
Description	Symbol	Quantity
Size of image	**Width** $\times$ **Height**	208 x156
Size of initial T-SOM and PL-G-SOM	$N \times M$	5x5
Iteration times	t	800
Temperature	T	1.0
Number of instructions (actions)	$a(i)$	8
Maximum/Minimum neighborhood in PL-G-SOM	$\sigma_{\max}$, $\sigma_{\min}$	$N \times M / 2$, 0.7
Reward for one action selected	r	10.0
Parameters of Feeling Map	$\boldsymbol{a}, \boldsymbol{b}$	0.1, 0.0001

5 The Experiment and Results

5.1 Hand Image Instruction Learning Systems Using T-SOM and PL-G-SOM

Eight kinds of hand shapes meaning 8 instructions of actions available for a robot are shown in Fig. 4 (a)-(g). Regularized binary images of hand shape showed the effectiveness of the preprocessing of images, meanwhile, feature data showed the availability of pattern recognition for their distinguished apparentness. The parameters used in T-SOM and PL-G-SOM are listed in Table 1. The number of units of SOM, T-SOM and PL-G-SOM was 5x5=25 initially. 800 iterations were executed during the learning process because the convergence of Squared Error (SE: distance between input and weights of connections, see Eq. (4)) and

Feeling value (see Eq. (3)).



(a) “fist”

(b) “one finger”

(c) “two fingers”

(d) “three fingers”

(e) “four fingers”

(e) “five fingers”

(f) “another two-fingers”

(g) “another three-fingers”

Figure 4: Hand image instructions used in the experiments: (left) Original images of 8 kinds of gestures; (center) hand shapes obtained by the image processing; (c) Features in the input space (80-dimension on horizontal axis).

5.2 The Results and Analyses

To confirm the efficiency of learning methods, learning curves which are depicted by the graph of training time versus errors or performance of the models are commonly used. We investigated the change of distance between input feature data and units weights and the change of Feeling value according to the training time, and show them by Fig. (5) and Fig. (6). PL-G-SOM showed the furthest convergence in both evaluation figures. The value of Feelings of T-SOM and PL-G-SOM reached their highest altitude 1.0, which means that 100% success rate was achieved as hand image instruction recognition/execution. Classic SOM showed unstable performance for the learning process had not realized convergent result. The reason may be considered that the training time for SOM was not enough, and suitable parameters such as Temperature, learning rate, and neighborhood function were not used, usually they are decided by empirical values.

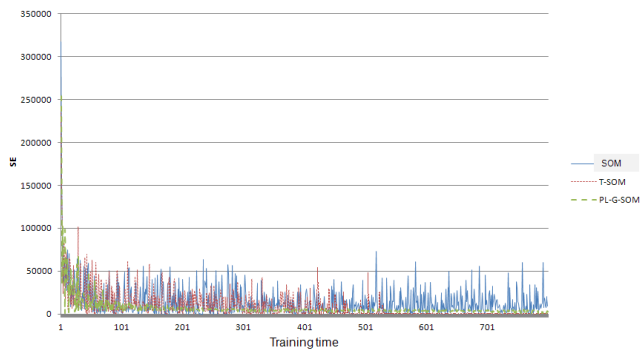


Figure 5: Comparison of different SOMs: the distances (or squared errors: SE) between *BMU* and input vectors (totally 8 kinds of gestures) decreased by training time, PL-G-SOM showed the best performance of learning convergence.

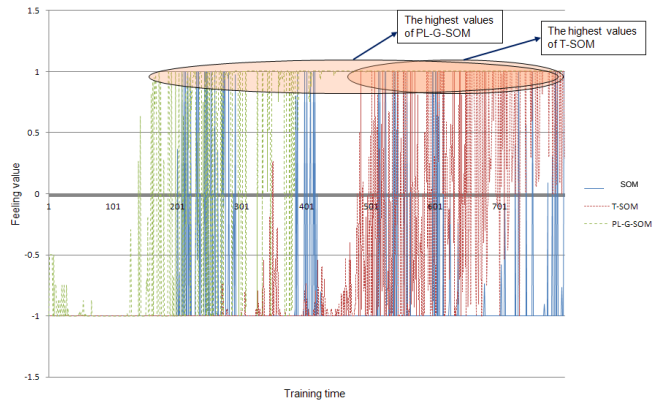


Figure 6: Comparison of different SOMs: the feeling value (totally 8 kinds of gestures) increased according to training time, PL-G-SOM showed the highest performance of learning convergence.

5	0	0	2	4	1	1
4	0	2	2	2	2	2
2	0	6	3	4	7	0
0	3	3	3	3	1	5
0	2	1	7	3	3	3
2	2	3	0	0	0	0

Figure 7: Action Map obtained by PL-G-SOM after training. 42 (6x7) units corresponding to 8 kinds of hand image instructions (0-7) are showed.

Fig. 7 shows the state of Action Map of PL-G-SOM after training. The number of units had grown from 5x5=25 to 6x7=42. Later input training data such as 4, 5, 6, 7 occurred fewer units meanwhile earlier data had more units for the more input times during the training. A method to avoid this situation is that training each instruction for certain times (until it converges) previously, then input all of data to find the global solution.

Fig. 8 shows the growth of the number of units on Feature Map (and Memory Layer) of T-SOM and PL-G-SOM. The speed and the quantity of PL-G-SOM were larger than T-SOM as the same hand image instruction learning systems.

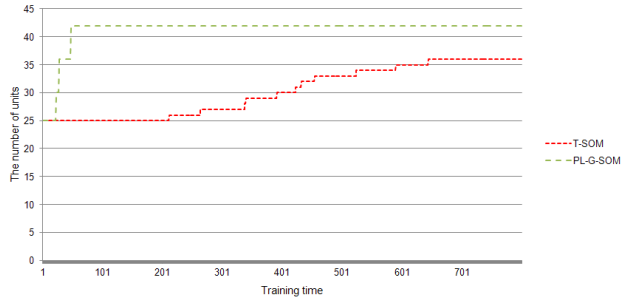


Figure 8: The number of units (neurons) on different maps grew differently during the learning process. The initial sizes of T-SOM and PL-G-SOM were same as 25, and the final sizes were 36 for T-SOM and 42 for PL-G-SOM respectively.

Furthermore, we used untrained samples with different changed variations of the eight shapes of hand, and confirmed the robustness of the trained system using T-SOM and PL-G-SOM. The limitation degree of tilt was about 30, and 45 degree for pan rotation.

From the comparisons of learning performance, it is able to conclude that PL-G-SOM proposed in this study showed more effective than the conventional T-SOM as the hand image instruction learning system.

6 Conclusions

A hand image instruction learning system for partner robot using PL-G-SOM was proposed. Comparing with the conventional system using T-SOM, PL-G-SOM showed better learning performance than T-SOM. Experiments using real robot are expected to confirm the online learning ability of the proposed system in the future.

7 References

[1] Bauer, H.-U., & Villmann Th. “Growing a hypercubical output space in a self-organizing feature map”. IEEE Transaction on Neural Networks, Vol. 8, No.2, 218-226, 1997.

[2] Baum, L.E., Petrie, T.“Statistical inference for finite state Markov chains”, The Annals of Mathematical Statistics, Vol. 37, No.6, 1554-1563, 1966.

[3] Berglund, E., & Sitte, J. “The parameter-less self-organizing map algorithm”. IEEE Transaction on Neural Networks, Vol. 17, No.2, 305-316, 2006.

[4] Cottrell, M., Fort, J.C., & Pages, G. “Theoretical aspects of the SOM algorithm”. Neurocomputing, Vol.21, 119-138, 1998.

[5] Dittenbach, M., Merkl, D., & Rauber, A. “The growing hierarchical self-organizing map”. In Proceedings of IEEE International Joint Conference on Neural Network (IJCNN ’00) , 15-19, 2000.

[6] Fritzke, B. “Growing grid – a self-organizing network with constant neighborhood range and adaption strength”. Neural Processing Letters, Vol. 2, No.5, 9-13, 1995.

[7] Hano, T., Kuremoto, T., Kobayashi, K., & Obayashi, M. “A hand image instruction learning system using Transient-SOM”. Transactions on SICE (Society of Instrument and Control Engineering),Vol.43, No.11, 1004-1006, 2007 (in Japanese)

[8] Kohonen, T. “Self-organized formation of topologically correct feature maps”. Biological Cybernetics, Vol.43, No.1), 59-69, 1982.

[9] Kohonen, T. “Self-Organizing Maps”, Berlin; Heidelberg; New-York: Springer, Series in Information Sciences, 1995.

[10] Kohonen, T. “The self-organizing map”. Neurocomputing, Vol. 21, 1-6, 1998.

[11] Kuremoto, T., Hano, T., Kobayashi, K. & Obayashi, M. “For partner robots: A hand instruction learning system using transient-SOM”. In Proceedings of The 2nd International Conference on Natural Computation and The 3rd International Conference on Fuzzy Systems and Knowledge Discovery (ICNC '06-FSKD'06), 403-414, 2006.

[12] Kuremoto, T., Hano, T., Kobayashi, K. & Obayashi, M. “Robot feeling formation based on image features”. In Proceedings of International Conference on Control, Automation and Systems (ICCAS 2007), 758-761, 2007.

[13] Kuremoto, T., Komoto, T., Kobayashi, K., & Obayashi, M. “Parameterless-Growing-SOM and its application to a voice instruction learning system”. Journal of Robotics, 9 pages (online), 2010.

[14] Kuremoto, T., Yamane, T., Feng, L.-B., Kobayashi, K., & Obayashi, M. “A human-machine interaction system: A voice command learning system using PL-G-SOM”. Proceedings of International Conference on Industrial Engineering and Management Special Session in MASS (IEEE-IEM 2011), IEEE Catalog Number: CFP1141H-CDR, 83-86, 2011.

[15] Pavlovic, V.I., Sharma, R., Huang, T.S. “Visual interpretation of hand gesture for human-computer interaction: a review”. IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol. 19, No. 7, 667–693, 1997.

[16] Rumelhart, D.E., McClelland, J.L., & The PDP Research Group. “Parallel distributed processing: Explorations in microstructure of cognition”, The MIT Press, Cambridge, 1986.

[17] Sherrah, S., Gong, S. (2001). “Skin Colour Analysis”, Available online URL:http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/GONG1/cvOnline-skinColourAnalysis.html

[18] Sutton, S.S., Barto, A.G. “Reinforcement learning: an instruction”, The MIT Press, London, 1998.

[19] Villmann, Th., Bauer, H. –U. “Applications of the Growing Self-Organizing Map”. Neurocomputing, Vol. 21, 91-100, 1998.

[20] Wilson, A.D., & Bobick, A.F. “Parametric hidden Markov models for gesture recognition”. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 21, No.9, 884-900, 1999.