Unsupervised and Adaptive Category Classification for a Vision-Based Mobile Robot

Masahiro Tsukada, Hirokazu Madokoro, and Kazuhito Sato

***Abstract*— This paper presents an unsupervised category classification method for time-series images that combines incremental learning of Adaptive Resonance Theory-2 (ART-2) and self-mapping characteristic of Counter Propagation Networks (CPNs). Our method comprises the following procedures: 1) generating visual words using Self-Organizing Maps (SOM) from 128-dimensional descriptors in each feature point of a Scale-Invariant Feature Transform (SIFT), 2) forming labels using unsupervised learning of ART-2, and 3) creating and classifying categories on a category map of CPNs for visualizing spatial relations between categories. We use a vision system on a mobile robot for taking time-series images. Experimental results show that our method can classify objects into categories according to their change of appearance during the movement of a robot.**

I. Introduction

Recently, robots having learning functions to adapt flexibly in various environments have been studied from various perspectives. In particular, although studies of autonomous behaviors that a robot chooses without human control have become active, many problems remain as obstacles to their practical use. Realization of advanced visual function of a robot is important because most information that humans use to determine behavior is visual information. One method to realize autonomous behavior for a robot is to obtain brain-like memory: a so-called World Image [1]. For creating a World Image, robots must classify objects into categories to understand the environment in terms of visual information. In robot vision studies, knowledge must be used together with vision to achieve a visual function resembling human sense [2]. Robots can obtain knowledge to classify visual information that is to be provided according to movements. After classification into such categories, the information can be saved as memories. In real environments for a robot, the number of categories is mostly unknown. The categories are also not known uniformly.

In this paper, we propose an unsupervised category classification method for discovering the number of categories. Additionally, we use Genetic Programming (GP) to generate intelligent behavior after acquiring various appearances in an environment. For target data of category classification, we use time-series images taken from a camera based on autonomous behavior patterns used in action trees generated by GP. We combined incremental learning of Adaptive Resonance Theory-2 (ART-2) [3] proposed by Grossberg et al. and self-mapping characteristics of Counter Propagation Networks (CPNs) [4] proposed by Nilsen. In actuality, ART-2 is a theoretical model of unsupervised neural networks of incremental learning that forms categories adaptively while maintaining stability and plasticity. Features of time-series images from the mobile robot's camera change with time. Using ART-2, which can learn time-series changes, our method enables an unsupervised category classification that did not need previous setting of the number of categories. A type of supervised neural networks—CPNs—actualize mapping and labeling together. Such networks comprise three layers: an input layer, a Kohonen layer, and a Grossberg layer. In addition, CPNs learn topological relations of input data for mapping weights between units of the input-Kohonen layers. The resultant category classifications are represented as a category map on the Kohonen layer.

Our method has the following three characteristics. First, our method can generate labels as a candidate of categories while maintaining stability and plasticity for time-series data. Second, automatic labeling of category maps can be realized using labels created by ART-2 as teaching signals for CPNs. Third, our method can present the diversity of appearance changes for visualizing spatial relations of each category on a two-dimensional CPN map. We evaluated our method using category classification experiments with time-series images taken by a camera on a robot moving with GP-generated behavior programs. As described herein, we emphasize the effectiveness of our method in category classification and acquisition of diverse appearances of an object to contribute to the improvement of accuracy of category classification.

II. Related work

In the field of computer vision, realization of generic object recognition to classify unknown objects in images into each category is anticipated as a technology to enable acquisition of intelligent systems [5]. Learning-based category classification methods in generic object recognition are roughly divisible into supervised category classification methods and unsupervised category classification methods. Using supervised category classification methods, classification categories are determined and training datasets including ground-truth labels for teaching signals are sometimes collected manually. Unsupervised category classification methods extract categories automatically for a problem of unknown classification categories and classify images into respective categories. Recently in the field of computer vision, studies of unsupervised category classification methods have been active and have attracted attention as

Masahiro Tsukada, Hirokazu Madokoro, and Kazuhito Sato are with the Faculty of Systems Science and Technology, Akita Prefectural University, Yurihonjo-shi, 015-0055, Japan (phone: +81 184 277081 email: m11a013@akita-pu.ac.jp).

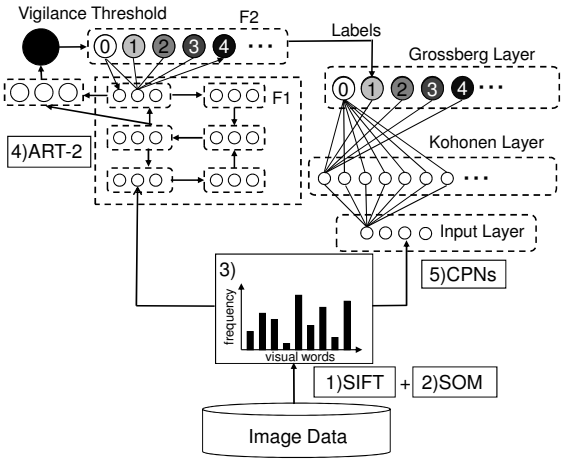

Fig. 1. Network architecture of our method.

technologies to express vision information. Sivic et al. proposed an unsupervised category classification method using probabilistic Latent Semantic Analysis (pLSA) and Latent Dirichlet Allocation (LDA), which are generative models from the statistical text literature [6]. They modeled an image containing instances of several categories as a mixture of topics and attempted to discover topics as object categories from numerous images. Zhu et al. introduced Probabilistic Grammar – Markov Models (PGMM) of generative models that combined Probabilistic Context-Free Grammars (PCFG) and Markov Random Fields (MRF) [7]. They used this method to create an object category model for object detection and unsupervised category classification. Todorovic et al. proposed an unsupervised identification method using optical, geometric, and topological characteristics of multi-scale regions consisting of two-dimensional objects [8]. They represented each image as a tree structure by division of multi-scale images. Moreover, Nakamura et al. proposed an unsupervised category classification method using multimodal information of vision, hearing, and touch [9]. They achieved category classification of objects that resemble human senses using embodied interactions of a robot. However, these methods include the restriction of prior settings of the number of classification categories. Therefore, those methods are applied only slightly to classification problems in a real environment for which the number of categories is unknown.

III. Proposed Method

Fig. 1 depicts the network architecture of our method. The network performs three tasks: calculating Bag-of-Features, generating labels for classifying sequential changes of appearances, and creating a category map for visualizing spatial relations between categories. The procedures are the following:

1) Extracting feature points and calculating descriptors using Scale-Invariant Feature Transform (SIFT),
2) Creating visual words of all SIFT descriptors using Self-Organizing Maps (SOM),
3) Calculating histograms of SIFT descriptors matched with visual words,
4) Generating labels using ART-2,
5) Creating a category map using CPNs.

Procedures 1) through 3), which correspond to preprocessing, are based on the representation of Bag-of-Features. The SIFT processing consists of two steps: detection of feature points and description of features. Generally, SIFT is used as a descriptive method of local features in generic object recognition. For producing visual words, we use SOM, which can learn neighborhood regions while updating cluster structure, whereas k-means must decide data of the center of a cluster. Because we use SOM, our method can represent visual words that lower the minimum level of false classification. Furthermore, the combination of ART-2 and CPNs enables unsupervised category classification that labels a large quantity of images in each category automatically. The detailed algorithms are the following.

A. Creating visual words using SOM

In our method, we apply SOM, not k-means, which is generally used in Bag-of-Features, for creating visual words. In the learning step, SOM updates weights while maintaining topological structures of input data. Actually, SOM creates neighborhood unit regions around the burst unit that demands a response of the input data. Therefore, SOM can classify various data whose distribution resembles that of the training data. In addition, Terashima et al. reported that SOM is superior to k-means as an unsupervised classification method that is useful to minimize misrecognition [10]. The learning algorithm of SOM [11] is the same as the algorithm used between the input-Kohonen layers of CPNs.

B. Generating of labels using ART-2

Various ART types exist[12]. Our method uses ART-2, into which it is possible to input continuous values [3]. The learning algorithm of ART-2 is the following.

1) Top-down weights Z_{ji}, bottom-up weights Z_{ij}, and outputs p_i, q_i, and u_i on the F1 of sublayers are initialized as

$$Z_{ji}(0) = 0, \qquad Z_{ij}(0) = \frac{1}{(1-d)\sqrt{M}}, \tag{1}$$

$$p_i(0) = q_i(0) = u_i(0) = v_i(0) = w_i(0) = x_i(0) = 0.0. \tag{2}$$

2) The input data I_i are presented to the F1; the sublayers are propagated as

$$w_i(t) = I_i(t) + au_i(t-1), \tag{3}$$

$$x_i(t) = \frac{w_i(t)}{e + \|w\|}, \tag{4}$$

$$v_i(t) = f(x_i(t)) + bf(q_i(t-1)), \tag{5}$$

$$u_i(t) = \frac{v_i(t)}{e + \|v\|}, \tag{6}$$

$$p_i(t) = \begin{cases} u_i(t) & \text{(inactive)} \\ u_i(t) + dZ_{Ji}(t) & \text{(active)}, \end{cases} \tag{7}$$

$$q_i(t) = \frac{p_i(t)}{e + \|p\|}, \tag{8}$$

$$f(x) = \begin{cases} 0 & if \quad 0 \le x < \theta \\ x & if \quad x \ge \theta. \end{cases} \tag{9}$$

3) Search for the maximum active unit T_j as

$$T_J(t) = max(\sum_j p_i(t) Z_{ij}(t)). \tag{10}$$

4) Top-down weights Z_{ji} and bottom-up weights Z_{ij} are updated as

$$\frac{d}{dt} Z_{Ji}(t) = d[p_i(t) - Z_{Ji}(t)], \tag{11}$$

$$\frac{d}{dt} Z_{iJ}(t) = d[p_i(t) - Z_{iJ}(t)]. \tag{12}$$

5) The vigilance threshold ρ judges whether input data belong to a category.

$$\frac{\rho}{e + \|r\|} > 1, r_i(t) = \frac{u_i(t) + cp_i(t)}{e + \|u\| + \|cp\|}. \tag{13}$$

When (13) is true, the active units reset and go back (3) to search again. Repeat (2) and (4) until the change rate of F1 is sufficiently small if (13) is not true.

In addition, a and b are coefficients on feedback loops from u to w and from q to v. Here, c is a propagation coefficient from p to r, and d is a learning rate coefficient. Furthermore, $cd/(1-d) \le 1$ is the constraint between them, and θ is a parameter to control a noise detection level in v. We set θ to 0.1 and ρ to 0.850 in our method.

C. Creating category maps using CPNs

The CPNs perform pattern mapping [4], i.e. CPNs map one pattern onto another pattern in all sets of patterns. When a pattern is presented, the learned network classifies patterns into specific categories using weights. Our method can automate labeling with generation of labels as teaching signals to the units of the Grossberg layer on CPNs. The CPN learning algorithm is the following.

1) $u^i_{n,m}(t)$ are weights from an input layer unit $i(i = 1, ..., I)$ to a Kohonen layer unit $(n, m)(n = 1, ..., N, m = 1, ..., M)$ at time t. Therein, $v^j_{n,m}(t)$ are weights from a Grossberg layer unit j to a Kohonen layer unit (n, m) at time t. These weights are initialized randomly. The training data $x_i(t)$ show input layer units i at time t. The Euclidean distance $d_{n,m}$ separating $x_i(t)$ and $u^i_{n,m}(t)$ is calculated as

$$d_{n,m} = \sqrt{\sum_{i=1}^{I} (x_i(t) - u^i_{n,m}(t))^2}. \tag{14}$$

2) The unit for which $d_{n,m}$ is smallest is defined as the winner unit c as

$$c = argmin(d_{n,m}). \tag{15}$$

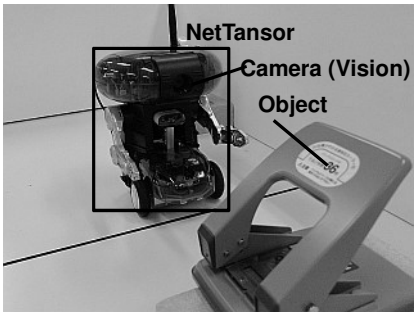

Fig. 2. Robot used for experiments (NetTansor by Bandai Co. Ltd.).

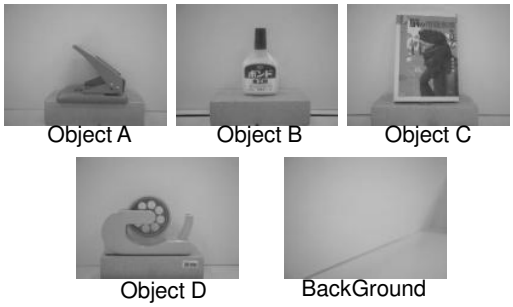

Fig. 3. Target objects and background.

3) Here, $N_c(t)$ is a neighborhood region around the winner unit c. Also, $u^i_{n,m}(t)$ of $N_c(t)$ is updated using Kohonen's learning algorithm, as

$$u^i_{n,m}(t+1) = u^i_{n,m}(t) + \alpha(t)(x_i(t) - u^i_{n,m}(t)). \tag{16}$$

4) In addition, $v^j_{n,m}(t)$ of $N_c(t)$ is updated using Grossberg's outstar learning algorithm, as

$$v^j_{n,m}(t+1) = v^j_{n,m}(t) + \beta(t)(t_j(t) - v^j_{n,m}(t)). \tag{17}$$

In that equation, $t_j(t)$ is the teaching signal to be supplied to the Grossberg layer. Furthermore, $\alpha(t)$ and $\beta(t)$ are the learning rate coefficients that decrease with the progress of learning. The learning of CPNs repeats until the learning iteration that was set previously. We set the learning iteration to 10,000 steps and $\alpha(t)$ and $\beta(t)$ to 0.5.

IV. Experimental results

In this experiment, we generate behavior programs using GP along two routes that we set in the same environment. We evaluate the effectiveness of our method for comparison of category classification results in the difference of time-series images taken using a camera acquired with movements of a robot.

A. Robot and experimental environment

Fig. 2 portrays a home robot (NetTansor; Bandai Co. Ltd.) used in this experiment. The robot is 190 mm high, 160 mm long, and 160 mm wide. The camera specifications are: imaging device, 1/4 inch CMOS; image format, JPEG; resolution, 320 $\times$ 240 pixels; and frame rate, 15 fps. The

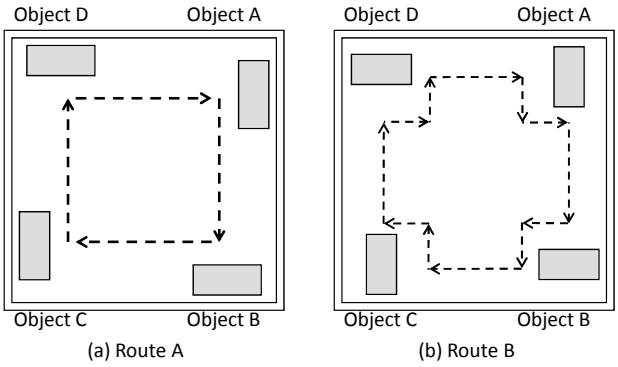

Fig. 4. Experimental environment and robot routes.

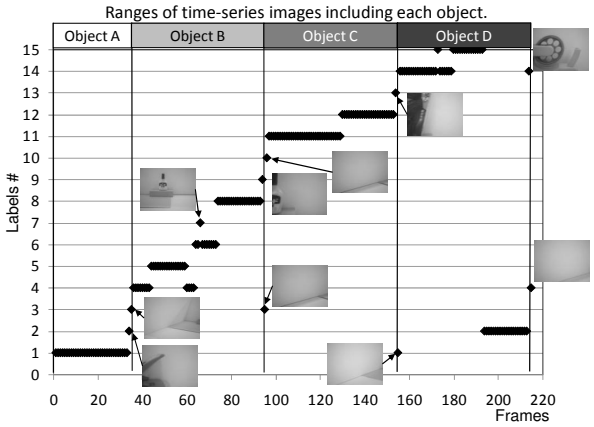

Fig. 5. Labeling result of ART-2 at Behavior A.

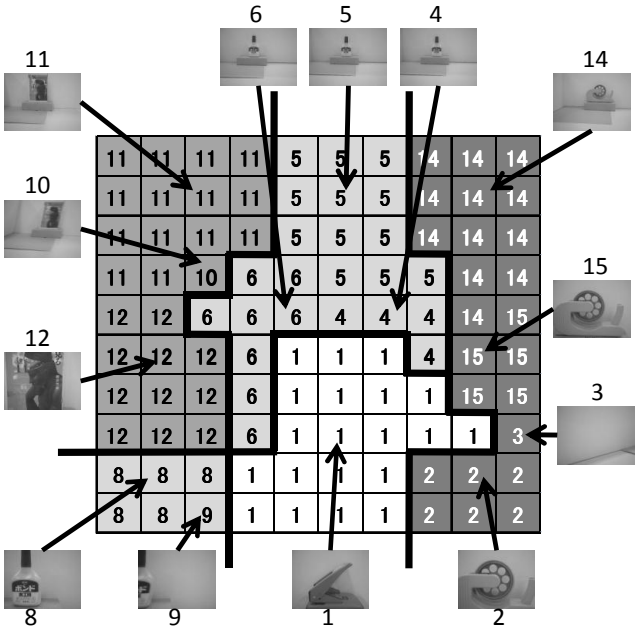

Fig. 6. Mapping result of CPNs at Behavior A. Numbers on the category map and the top and bottom parts of the images correspond to the labels created by ART-2.

moving environment is 1,150 $\times$ 1,150 mm surrounded by 300 mm high white walls. We set four objects in the environment. We assumed the environment for moving of this robot as a desk. In consideration of the robot height, we used office supplies with characteristic shapes. The target objects are a punch (Object A), a bond (Object B), a book (Object C), and cellophane tape (Object D) shown in Fig. 3. Fig. 4 shows the assignment of objects in the environment and the roughly determined goals of routes for the robot. We generated behavior programs using GP. We set landmarks on both routes. Fig. 4(a) portrays a simple route along with walls. Fig. 4(b) presents a route that acquires various appearances around each object. For this experiment, we created datasets consisting of time-series images in each behavior. Datasets comprise training datasets and testing datasets for which the robot runs two rounds in the environment. In the learning phase, we evaluate both results of labels generated by ART-2 and category maps generated by CPNs. In the testing phase, we evaluate results of category maps generated by CPNs.

B. Generation of robot behavior

Actually, GP expands the genotype of Genetic Algorithms (GA) to handling structural expressions such as trees or graphs. As a heuristic approach, GP is applied to generation of robot programs. Tree structures consist of non-terminal nodes (functions), terminal nodes (variables or constant values), and a root. For this study, we used GP for generating two behavior programs to run for routes A and B. Nodes used for GP were the following.

- Terminal nodes: *move*, *left*, *right*, *upleft*, and *upright*,
- Non-terminal nodes: *runif*, *progn2*, and *progn3*.

Terminal nodes cope with forward movement, 90 deg turns to the left and to the right, and 15 deg turns to the left and to the right. The non-terminal node *runif* is a condition judgment by which the first argument is executed if there is a landmark in front of the robot; the second argument is executed if no landmark exists. The non-terminal nodes *progn2* and *progn3* are functions that execute two arguments and three arguments sequentially. For the simulation, we used the map dividing the environment into 10 $\times$ 10 blocks. One block corresponds to 115 $\times$ 115 mm. The fitness value is increased when the robot finds a landmark and runs through it. We set the population size to 50 individuals and the generation to 100 steps. We used the best individuals as behavior programs. We respectively call Behavior A and Behavior B to be generated in routes A and B.

C. Classification results (Behavior A)

Figs. 5 and 6 respectively depict labels generated by ART-2 and a category map generated by CPNs. Time-series images in Behavior A are classified into 15 labels in Fig. 5. The labels are more numerous than the target objects because labels were assigned to each image taken by the robot turned 90 deg from the four corners in the environment. Moreover, different labels are assigned to images including the whole object and images as partial objects. Each object classified with different labels with ART-2 is mapped to neighborhood

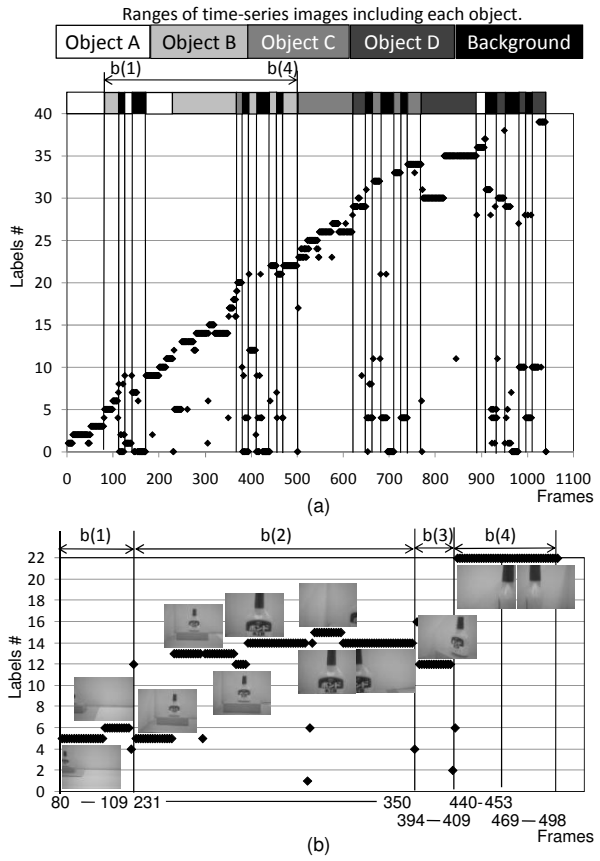

Fig. 7. Labeling result of ART-2 at Behavior B: (a) all frames, (b) frames of Object B.

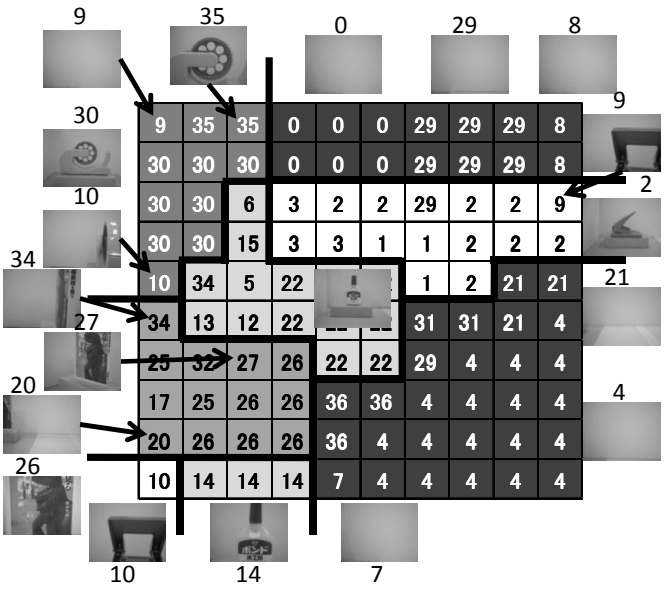

Fig. 8. Mapping result of CPNs at Behavior B. Numbers on the category map and the top and bottom parts of the images correspond to the labels created by ART-2.

units on the category map of CPNs in Fig. 6. Labels 7 and 13 that correspond to one image are not apparent to the category map on CPNs.

D. Classification results (Behavior B)

Figs. 7 and 8 respectively depict labels generated by ART-2 and a category map generated by CPNs. Time-series images in Behavior B are classified using 40 labels in Fig. 7(a). The labels are more numerous in Behavior B because appearances in the environment became various with increasing behavior patterns. Around objects, generating labels tends to become complicated because Behavior B includes appearances of images that include only a wall or a small object located far from the robot. In the state of turning, labels of these images tend to increase. Here, Fig. 7(b) presents results of labels generated using ART-2 in images showing Object B on 80–498 frames in Fig. 7(a). In Fig. 7(b), we portray analyses of the relation of appearances and labels in each object. This result demonstrates that our method can generate labels based on appearance changes of objects, although the behavior of turning is increased in comparison with Behavior A. Fig. 8 shows that CPNs created categories for mapping to neighborhood units in the category map in each image for which ART-2 generated plural labels in each object. Based on the relations of categories and images in each label, our method can express different appearances of objects. In addition, although ART-2 created different labels from wall images, CPNs created the background category with mapping to neighborhood units on a category map. Furthermore, images of turning or partially defective objects are mapped around border units between categories. In Labels 9 and 10, CPNs can classify images that are confused labels by ART-2 for mapping to neighborhood units in each category.

E. Recognition results

For quantitative evaluation of the classification performance of our method, we use the following recognition rate.

$$(RecognitionRate) = \frac{(CorrectData)}{(AllData)} \times 100. \qquad (18)$$

In [13], the recall rate of SIFT is less than 50% when objects are occluded more than 30%. We annotated images including defective objects of more than 30% as being of the category of backgrounds and 'other'. Tables I and II respectively present the target datasets and the recognition rate in each dataset for training and testing. The target datasets presented in Table I consist of A-1 and A-2 for the first and second rounds, with Behaviors A and B-1 and B-2 for the first and second rounds with Behavior B. This experiment evaluated recognition rates for all combinations of four datasets for learning and testing.

The respective recognition rates for training datasets A-1, A-2, B-1, and B-2 are 99.1, 98.8, 90.8, and 96.8%. In Behavior A, the respective recognition rates for testing A-2 and A-1 after learning A-1 and A-2 are 98.8 and 93.5%. In addition, the respective recognition rates for testing B-1 and B-2 after learning A-1 and A-2 are 63.5, 64.3, 51.5, and 50.4%.

In Behavior B, the respective recognition rates for testing B-2 and B-1 after learning B-1 and B-2 are 86.8 and 87.2%.

TABLE I

TARGET DATASETS.

	First round	Second round
Behavior A	A-1	A-2
Behavior B	B-1	B-2

TABLE II

RECOGNITION RATES IN EACH BEHAVIOR [%].

		Testing Datasets				Mean rates for testing datasets	
		A-1	A-2	B-1	B-2		
Training Datasets	A-1	99.1	98.8	63.5	64.3	75.5	
	A-2	93.5	98.8	51.5	50.4	65.1	70.3
	B-1	83.8	77.1	90.8	86.8	82.6	
	B-2	94.0	95.8	87.2	96.8	92.3	87.5

In addition, the respective recognition rates for testing A-1 and A-2 after learning B-1 and B-2 are 83.8, 77.1, 94.0, and 95.8%. The respective mean recognition rates for testing datasets for Behavior A and for Behavior B are 70.3 and 87.5%. This result means that Behavior B is superior to Behavior A for learning.

V. DISCUSSION

Category classification for generic object recognition is necessary to classify categories for assigning one label to one category. However, category classification for robot vision is necessary to classify categories for assigning labels positively to appearance changes with sensing in an environment. We consider that ART-2 can learn appearance changes positively for generating labels. Nevertheless, the number of labels of ART-2 is greater because appearance changes in the environment are increased with complicating behavior patterns. The CPNs created categories in each object whose appearance differs from that of neighboring units. The CPNs integrated wall or robot-turning images to the category of background and 'other', although these images are caused by increasing labels on ART-2. In addition, with the topological mapping characteristic based on the neighborhood learning of CPNs, images that characterized each object and images for which the robot is turning are mapped respectively near the center in each category and near borders between categories. The mean recognition rate for learned Behavior B is 17.2% higher than that of Behavior A. We consider that the accuracy of the category classification is improved with acquisition of various appearances. Therefore, our method enables category classification of time-series images in a real environment, including appearance changes of objects. We consider this category classification method as effective not only for computer vision for generic object recognition, but also for robot vision for which the number of categories is unknown and for which appearances in an environment are various.

VI. CONCLUSION

This paper presented our proposition of an unsupervised category classification method combined with incremental learning of ART-2 and self-mapping characteristic of CPNs. We applied our method to an unsupervised category classification based on appearance changes. We created behavior programs using GP for category classification experiments of time-series images to show the characteristics and effectiveness of our method. Experiments have demonstrated that our method represents diverse appearance changes of objects as labels using incremental learning of ART-2. Moreover, our method can visualize spatial relations of labels and integrate redundant or similar labels generated by ART-2 as a category map using self-mapping characteristics and neighborhood learning of CPNs. Our method can represent diverse categories and improve classification performance as well as acquire various appearances. The recognition rate using Behavior B for providing various appearances is superior to the recognition rate using Behavior A. These results demonstrate the necessity of generating behavior patterns and acquiring additional various appearances using GP.

Future studies must be done to develop methods to extract borders among clusters automatically and to determine a suitable number of categories from category maps of CPNs. We will apply category maps of our method to non-terminal nodes of GP and automatically generate behavior patterns acquiring various appearances.

REFERENCES

[1] K. Nakaya, "Making of a Brain – Thinking about Biotechnology from a Making of a Robot –," Kyoritsu Shuppan Co., Aug. 1995.

[2] T. Kanade, "Computer Vision and AI – Their Relation and Non-Relation –," The Journal of Artificial Intelligence, vol. 18, no. 3, May 2003.

[3] G.A. Carpenter and S. Grossberg, "ART 2: Stable Self-Organization of Pattern Recognition Codes for Analog Input Patterns," Applied Optics, vol. 26, pp. 4919-4930, 1987.

[4] R. Hetch-Nielsen, "Counterpropagation networks," Proc. of IEEE First Int'l. Conference on Neural Networks, 1987.

[5] K. Yanai, "The Current State and Future Directions on Generic Object Recognition," Journal of Information Processing: The Computer Vision and Image Media, vol. 48 no. SIG16 (CVIM 19), Nov. 2007.

[6] J. Sivic, B.C. Russell, A.A. Efros, A. Zisserman, and W.T. Freeman, "Discovering Objects and their Localization in Images," IEEE International Conference on Computer Vision, pp. 370-377, 2005.

[7] L. Zhu, Y. Chen, and A. Yuille, "Unsupervised Learning of Probabilistic Grammar – Markov Models for Object Categories," IEEE Trans. PAMI vol. 31, no. 1, Jan. 2009.

[8] S. Todorovic and N. Ahuja, "Unsupervised Category, Modeling, Recognition, and Segmentation in Images," IEEE Trans. PAMI, vol. 30, no. 12, Dec. 2008.

[9] T. Nakamura, T. Nagai, and N. Iwahashi, "Multimodal Object Categorization by a Robot," Journal of the Institute of Electronics, Information, and Communication Engineers, D vol. J91-D, no. 10, pp. 2507-2518, 2008.

[10] M. Terashima, F. Shiratani, and K. Yamamoto, "Unsupervised Cluster Segmentation Method Using Data Density Histogram on Self-Organizing Feature Map," Journal of the Institute of Electronics, Information, and Communication Engineers, D-II vol. J79-D-II, no. 7, pp. 1280–1290, Jul. 1996.

[11] T. Kohonen, "Self-Organizing Maps," Springer Series in Information Sciences, 1995.

[12] G.A. Carpenter and S. Grossberg, "Pattern Recognition by Self-Organizing Neural Networks," The MIT Press, 1991.

[13] K. Mikolajczyk and C. Schmid, "A Performance Evaluation of Local Descriptors," IEEE Trans. PAMI, vol. 27, no. 10, Oct. 2005.