NAIST-IS-DD0961022

Doctoral Dissertation

Unsupervised Category Formation and Its Applications to Robot Vision

Hirokazu Madokoro

September 24, 2010

Department of Information Systems
Graduate School of Information Science
Nara Institute of Science and Technology

A Doctoral Dissertation
submitted to Graduate School of Information Science,
Nara Institute of Science and Technology
in partial fulfillment of the requirements for the degree of
Doctor of ENGINEERING

Hirokazu Madokoro

Thesis Committee:

Professor Masatsugu Kidode	(Supervisor)
Professor Tsukasa Ogasawara	(Member)
Professor Toshikazu Wada	(Member, Wakayama University)
Associate Professor Norimichi Ukita	(Member)

Unsupervised Category Formation and Its Applications to Robot Vision*

Hirokazu Madokoro

Abstract

Recently, human-friendly robots such as pet robots, home robots, and human-symbiotic robots have emerged in our daily life. A new lifestyle—one of living with robots together—will be forthcoming in various environments at homes and offices. For these robots to be useful and valuable for the existence of humans, it is necessary to obtain the ability to perform various tasks autonomously and flexibly. To create this ability, robots must have systematic knowledge to adapt to various environments and to perform those various tasks using world image maps that are defined as knowledge to be memorized in the brain, and which can recall the real world in a memory.

This thesis presents an unsupervised category formation method using Self-Organizing Maps (SOMs) for creating category maps as world image maps. For signal-based automatic labeling, we introduce Counter Propagation Networks (CPNs) that are appended the Grossberg layer to SOMs which work for supervised learning. In the primary experiments, this thesis presents basic characteristics of CPNs to improve generalization capabilities of supervised neural networks based on topological data mapping. Using topological data mapping on CPNs, our method provides advantages to interpolate new data in sparse areas that exist among categories and to remove overlapping or conflicting data in original training data. Moreover, the proposed method can control the number of training data by changing the size of the category map according to a problem to be solved.

*Doctoral Dissertation, Department of Information Systems, Graduate School of Information Science, Nara Institute of Science and Technology, NAIST-IS-DD0961022, September 24, 2010.

For practical uses in an actual environment, this thesis presents two applications using a mobile robot. The first application is scene category formation for global position estimation of a mobile robot. Our method can extract changes in landscape revealed by viewing image sequences as concept patterns by SOMs. Effective position information is acquired by making hierarchical SOMs and using it to consolidate position estimation concept patterns. We evaluated the effect of shifts in position and direction while the robot was executing a trial journey on global position estimation. The extent of these shifts established beyond a doubt that our method was robust. The results of an on-site field test of a robot system in a hospital with a convalescence ward confirmed the effectiveness of our method for practical use.

The second application is unsupervised category formation for recognizing generic objects perceptually. For this application, we propose an unsupervised category formation method using Adaptive Resonance Theory-2 (ART-2) networks and CPNs. Using labels produced by ART-2 for teaching signals of CPNs, signal-based automatic labeling of units on the category map can be realized. Moreover, the combination of SOMs and ART-2 can represent spatiotemporal relations of input data. Using One Class-Support Vector Machines (OC-SVMs), our method enables feature representation that contributes to improved accuracy of classification for selecting feature points to concentrate characterized information of an image. Classification results of static images using a Caltech-256 object category dataset and dynamic images using time-series images according to movements respectively demonstrate that the proposed method can visualize spatial relations of categories while maintaining time-series characteristics. Moreover, we used Genetic Programming (GP) to create behavior sets for object classification to obtain diverse appearance changes of objects. Our method can represent diverse categories and recognize generic objects for actualizing advanced interaction between humans and robots.

Keywords:

Unsupervised learning, SOMs, CPNs, ART-2, robot vision, category formation, object recognition, position estimation.

Acknowledgements

First, I would like to express my deepest thanks to my supervisor, Professor Masatsugu Kidode. Without him, this thesis could never have been completed. He accepted me when I visited his laboratory in Feb. 2009. He thereafter provided me with boundless support and guidance in my study. He also provided me his thoughts on the philosophy related to academic studies at the time, not only at meetings and seminars in the laboratory, but also chatting with him at lunch breaks.

I would like to thank Professor Tsukasa Ogasawara and Professor Toshikazu Wada, committee members of my thesis, for their time reading and evaluating my thesis and for their invaluable comments and advice. I also thank Associate Professor Norimichi Ukita, a committee member evaluating my thesis. He provided me with special support and guidance in my study. He also gave me opportunities of social activities.

My appreciation also extends to Assistant Professor Hitoshi Habe, Assistant Professor Takamitsu Matsubara, and fellow students in the Advanced Intelligence Laboratory for their friendship and help, although I could not go to the laboratory so often. Thanks also to all staff at the Nara Institute of Science and Technology and the Graduate School of Information Science for their invaluable assistance.

Special gratitude is also extended to Associate Professor Kazuhito Sato. He is my supervisor at Akita Prefectural University (APU) and has been my mentor since I was a student at Akita University. Many thanks to my students at the Neuro Informatics Laboratory, Department of Machine Intelligence and Systems Engineering, Faculty of Systems Science and Technology, APU. Master course students Masahiro Tsukada and Yuya Utsumi, members of the robot intelligence group at our laboratory, put forth great effort to conduct experiments according to my requests. They collaborated and co-authored conference papers. I also thank my fellow engineers at SmartDesign Corp. They joined our seminar every week and provided various comments and advice from the viewpoint of business.

Finally, I must thank my parents, my sister, and my grandparents, who have all supported my life since the time I lived in my own hometown of Wakayama. I also thank my dearest wife, Haruka, whose love, understanding, and support are limitless.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1. Background and motivation

Numerous robots have been developed for specific purposes, especially in industrial uses, to augment or replace the labor of humans. These robots are operated automatically to repeat a simple task in an environment that is separated from people. In this century, the purposes of robots are expanding variously to include working, cooperating, and living with humans to provide support, enjoyment, and comfort. Recently, human-friendly robots such as pet robots, home robots, and human-symbiotic robots have emerged in our daily life [1]. A new lifestyle—one of living together with robots—will be forthcoming in various environments in homes and offices. For these robots to be useful and valuable for the existence of humans, it is necessary that they attain the ability to perform various tasks autonomously and flexibly. For creating this ability, robots must have systematic knowledge to adapt to various environments and to perform those various tasks.

As an ability for future robots, Nakano [2] defined world image maps as knowledge that is memorized in the brain, and which can recall the real world in a memory. World image maps are therefore similar to memories of the real world that are retained in the brain. Humans obtain various signals using the five senses (touch, taste, hearing, sight, and smell). We create world image maps using these signals. Moreover, we can behave intellectually using world image maps. Similarly, robots can move and behave autonomously if they can come to use world image maps shown in Fig. 1.1 as humans do. In the case of environments that

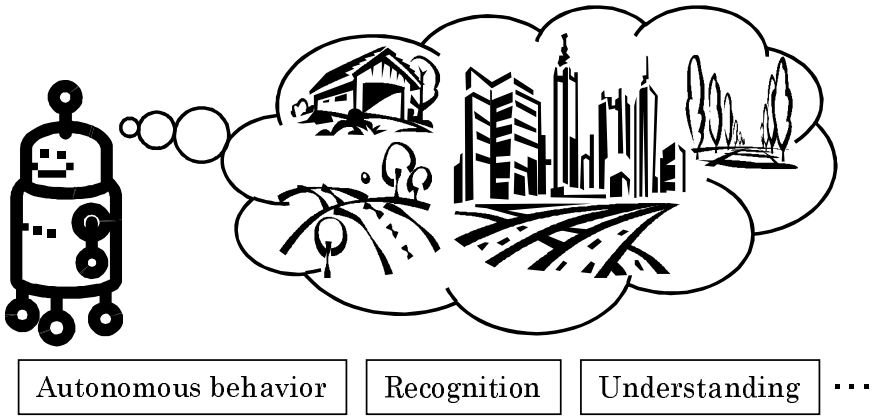


Figure 1.1. World image maps for a robot as a brain-like memory.

contain no map information, humans use memorized appearances of scenes and objects as landmarks for moving. This is one type of world image map to be memorized, with appearances of scenes and objects that are retained in memory as conceptual patterns. Robots could move in an environment without a map if they were able to create and use world image maps. For creating world image maps, robots must have the ability to recognize objects and scenes through sensor systems.

The visual cortex [3] occupies the largest part of the brain shown in Fig. 1.2. In robot sensor systems, visual information is a key channel for sensing an environment [4]. In Computer Vision (CV) technologies, numerous methods have been proposed to recognize objects and scenes in images as generic object recognition [5, 6, 7]. They classify objects and scenes into categories with identical or similar relations of characteristics. For the application of CV to Robot Vision (RV) systems, it is necessary to have calculation performance while retaining real-time processing in a limited resource environment [4, 8]. Recently, CV technologies are transferred to RV technologies with the advanced progress of calculation performance of computers, fast wireless communication technologies, and popularization of reconfigurable devices as hardware platforms to implement algorithms [9].

Object recognition methods in CV are divisible into two types: supervised object recognition and unsupervised object recognition. The majority is supervised

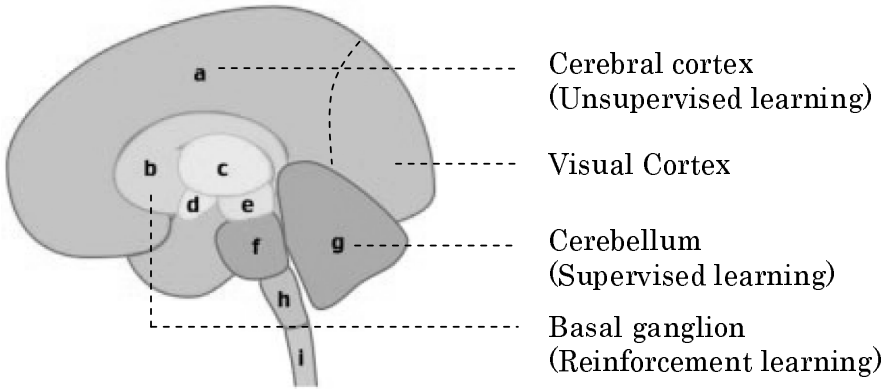


Figure 1.2. Cerebral regions and learning functions.

object classification aimed at high recognition rates in numerous categories for use in real-world applications [5]. Supervised methods necessitate the collection of images for training with teaching signals as ground truth datasets. Recently, unsupervised object recognition has become attractive as a method to discover categories automatically from numerous images [5]. For our ultimate goal of implementation of a robot, unsupervised object recognition is necessary to recognize and understand scenes and objects. For creating world image maps, all information collected by sensors must be used for obtaining knowledge of various types [2]. Therefore, we consider that it is important to discover and to extract hidden rules and knowledge based on unsupervised object recognition, not that used in supervised object recognition, which is forced to classify semantic categories.

Figure 1.3 depicts an example of a robot in an actual environment. Our objective is to move a robot in an actual environment. However, offices and homes are unsuitable environments in which to move a robot because these are arranged optimally for humans to work in or to live in. Therefore, it is necessary to train a robot in each environment. When using supervised learning-based methods, it is a heavy load to prepare training datasets with teaching signals. When using unsupervised learning-based methods, a user assigns semantic information to each category obtained by a robot. Therefore, advanced interactions between robots and humans can be realized using unsupervised learning-based methods.

For comparison of classification performances, the Caltech-256 object category dataset [10] is a famous open dataset used in generic object recognition. In

Figure 1.3. An example of a robot in an actual environment

the Caltech-256 dataset, one object consists of one image. No series information exists in this dataset, although objects of various types are included in one category. Regarding recognition systems used in a mobile robot, time-series images are useful for autonomous movement of a robot. Alternatively, robots can be controlled by an agent to collaborate with other robots.

In current generic object recognition studies, training from datasets is a valuable technology to obtain recognition rules for a target problem automatically. As in supervised object recognition, Support Vector Machines (SVMs) [11] and Boosting [12] are popularly used for powerful learning algorithms. These machine learning algorithms produce a good combination when used with part-based feature representations. As in unsupervised object recognition, Probabilistic Graphical Models (PGMs) [13] are widely used for text recognition. An advantage of PGMs is that they obtain high accuracy with selection of an interpretation that becomes the maximum of the posterior probability to calculate probabilities in each element and a joint probability between elements. However, the performance of PGMs depends strongly on the graphical structures to be created. Most models are created manually by an expert because automatic modeling tools have insufficient performance to create a structure from datasets. Moreover, the setting of the number of classifying categories is necessary before creating a model.

For selecting methods, we set two requirements used in unsupervised object recognition of time-series images obtained using a mobile robot. The first re-

quirement is an automatic mechanism to extract the number of categories. For problems that require unsupervised object recognition, the number of categories is unknown. The second requirement is an incremental mechanism that maintains stability and plasticity together. The environment in which a robot moves changes dynamically. In this situation, robots must be able to learn at any time for updating created categories.

The cerebral cortex, which sustains higher brain functions such as creativity, thinking, and memory, is shown to perform specialized unsupervised learning [14]. The Self-Organizing Maps (SOMs) proposed by T. Kohonen [15] act upon self-mapping high-dimensional input data into a low-dimensional space while maintaining topological data structures used in competitive and neighborhood learning functions. One key feature of SOMs is visualization of data distributions with topological preserving mapping [16]. Using this feature, SOMs have been applied to numerous applications of actual problems [17, 18].

Labeling of units on the mapping layer of the SOMs is the main step of determination of categories. The number of categories should be set before labeling. After learning of SOMs, it is necessary to assign labels to identify categories created on units of the mapping layer. The classification performance depends on labeling as a mapping result, although SOMs have a high mapping capability [19]. Labeling is a primitive step in unsupervised learning. Especially in the case of numerous units on the mapping layer, this work requires much loading. Moreover, no well-indexed methods exist for labeling. In many cases, experimenters or operators must label them manually. As methods to determine the number of categories, the handling of target problems differs in the cases of known and unknown quantities of categories. In numerous existing methods particularly addressing problems of unsupervised learning, the number of categories is assigned a priori because of the difficulty of evaluating the results, although the number of categories is unknown.

As incremental and unsupervised neural networks, Adaptive Resonance Theory (ART) proposed by Grossberg et al. [20] is a theoretical model used to learn and to memorize input data to long-term memory with stability and plasticity. The network structure of ART is designed based on the biological backgrounds to realize feature extraction, noise cutting, matching with stored memories, and

creation of new categories. Actually, ART has many variations: ART-1, ART-1.5, ART-2, ART-2A, ART-3, ARTMAP, Fuzzy ART, Fuzzy ARTMAP, etc. [20]. In [21], Kaylani et al. described that one limitation of ART is the category proliferation problem. They proposed Multiple Object-Genetic ART (MO-GART) using an optimization approach combined with Genetic Algorithms (GA) [22]. In their experiments, the performance of MO-GART approaches that of SVMs. However, this method is only applicable to supervised classification problems.

For this study, we use ART-2 [23], which can input continuous values. Although ART-2 creates categories for all input data, the number of categories increases according to the number of input data because ART-2 has no mechanisms to delete or to integrate redundant categories. Moreover, spatial relations among categories are not clear because categories are created sequentially. As described in this thesis, our method presents a visual image showing spatial relations in categories using SOMs for mapping to a two-dimensional space using labels generated by ART-2. Nielsen proposed Counter Propagation Networks (CPNs) [24] that are appended the Grossberg layer to SOMs which work for supervised learning. We used CPNs for automatic labeling of SOMs using labels created by ART-2.

Existing high-performance unsupervised clustering methods are k-means [25], probabilistic Latent Semantic Analysis (pLSA) [26], Latent Dirichlet Allocation (LDA) [27], Dirichlet Process Mixture (DPM) [28], and local Dirichlet Process (lDP) [29]. Using k-means, pLSA, and LDA requires setting of the number of categories in advance. As with SOMs, DPM requires no previous information of the number of clusters. However, DPM is unable to estimate the correct number of clusters if it has no previous knowledge of data that are used to create a probability distribution. In contrast, SOMs extract clusters using topological mapping in a low-dimensional space without prior knowledge of a probabilistic data distribution [30]. Therefore, we introduce SOMs for unsupervised clustering and category formation for a mobile robot without previous information of categories.

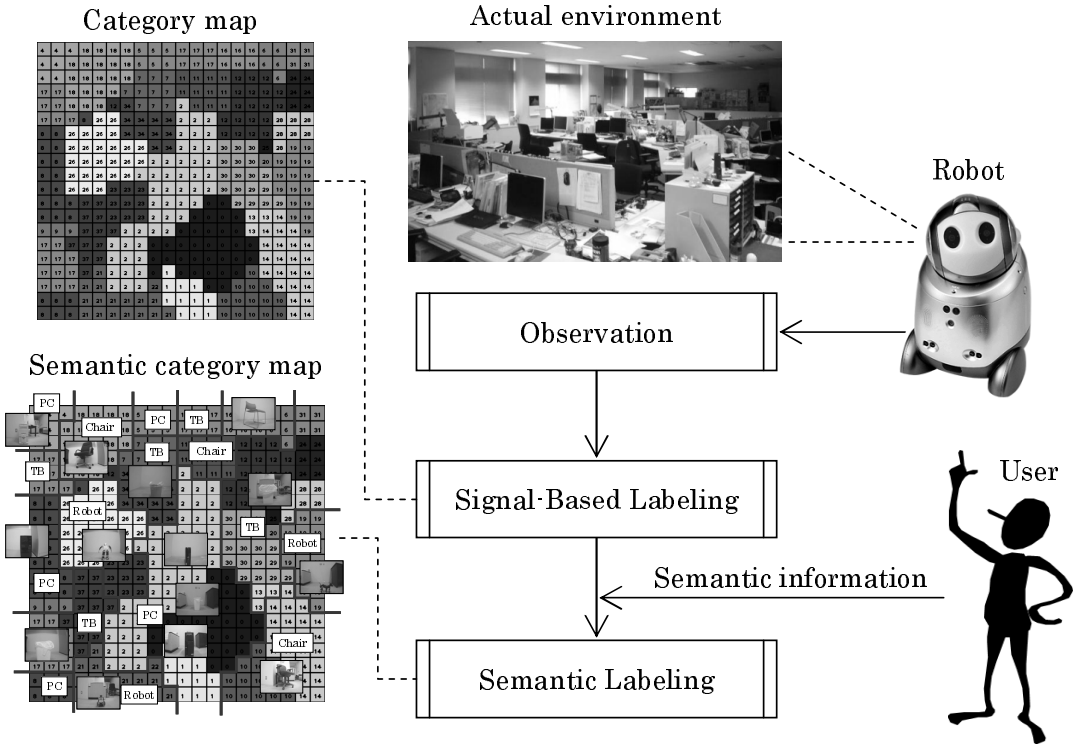


Figure 1.4. Use-case for creating semantic category maps.

1.2. Use-case and application examples

We present a practical example of category formation with a robot based on use-case analysis. We also present a scenario of semantic category creation for actualizing perceptual recognition in the same manner used by humans. Fig. 1.4 portrays a use-case example in the training step for a robot to obtain a semantic category map. A robot moves in an environment without any restrictions. For this movement, we consider that it is desirable for a robot to obtain not only various objects, but also various views of the objects. The robot learns sequentially using images obtained using a camera mounted on the robot. In the training step, our method requires no teaching signals prepared in advance as a ground-truth dataset for input images. Particularly, ART-2 networks generate labels only from input signals. Subsequently, CPNs create category maps using input datasets and labels generated using ART-2.

Using category maps, our method can recognize objects for test datasets as

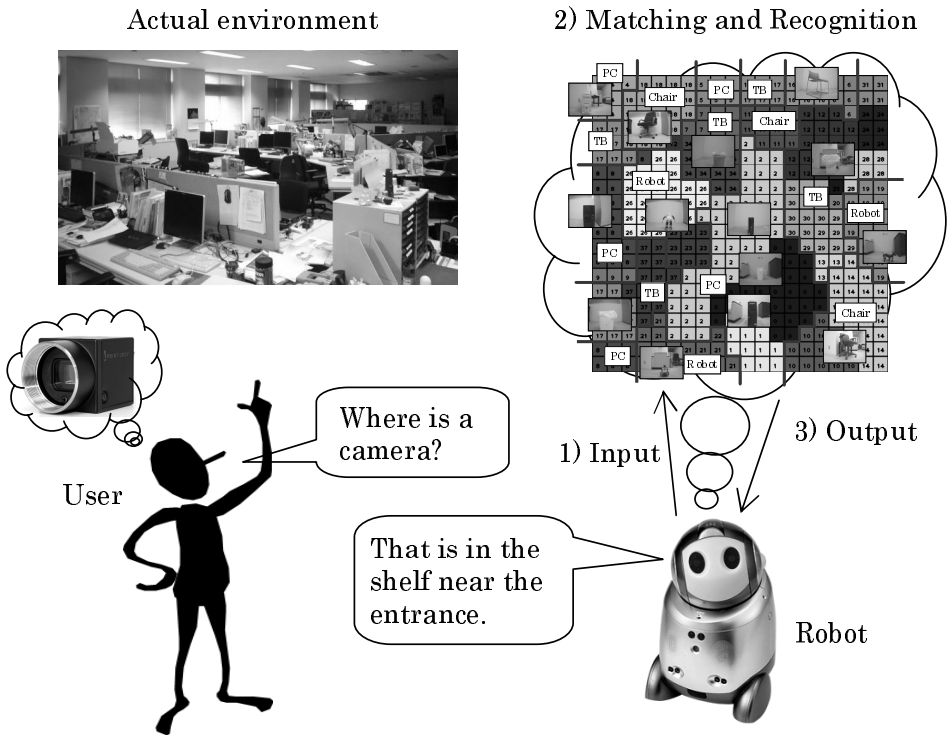


Figure 1.5. Example of perceptual recognition using semantic category maps.

signals of images. However, the recognition approach is mismatched to the perception of humans. In signal-based approaches, robots meet a perceptual gap with humans, although they both observe the same object.

Perceptual correspondence is an important factor to actualize advanced interaction between humans and robots. For example, we suppose that a user asks a robot about an object. We consider that users feel a communication gap if robots can only recognize an object as signals according to its shape, brightness distribution, color information, etc. In contrast, communication will be approved if robots can recognize an object perceptually.

For actualizing perceptual recognition, the assignment of semantic information is necessary for categories that are formed by signals. Herein, the assignment of semantic information is the naming of objects by a user manually. Therefore, this operation means matching with object names and labels, which are classified as signals. However, several labels correspond to one object name as a category because we set fine classification granularity of ART-2 networks. In the step of se-

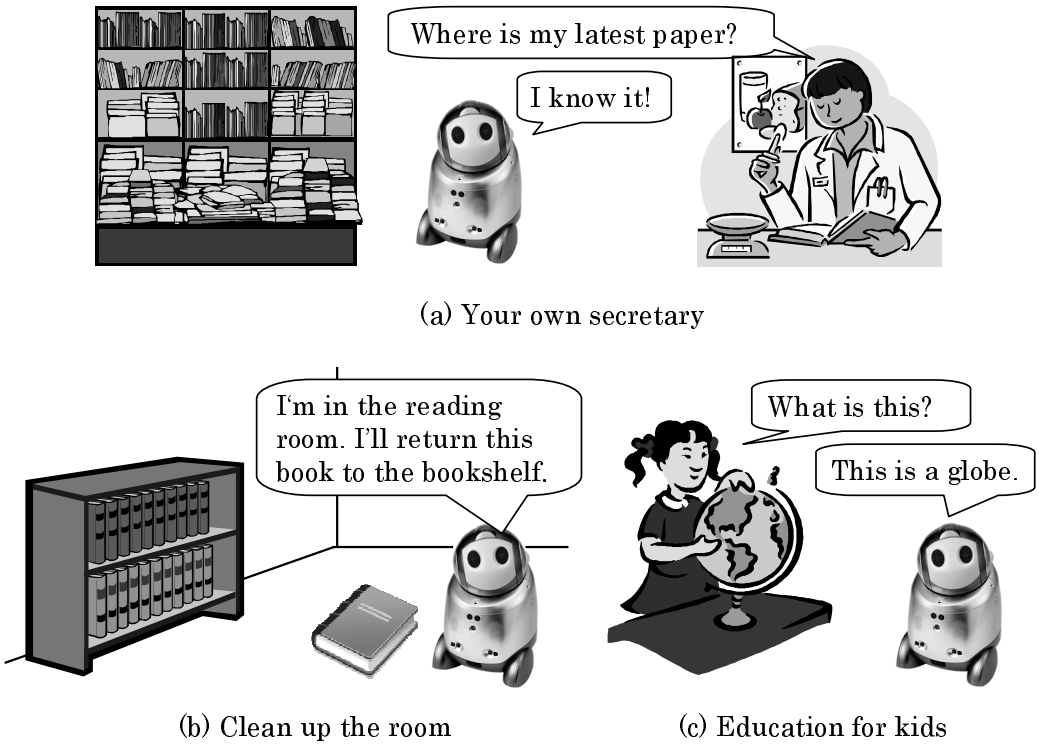


Figure 1.6. Application examples of this technology to recognize scenes and objects perceptually.

mantic information assignment, labels are integrated by a user on a category map. Several images are mapped to each label that contains a representative image. Based on a representative image, category names are assigned by a user in each label. For this study, we call this operation semantic labeling after unsupervised category formation.

Actually, unsupervised neural networks extract hidden rules in the datasets. Teaching signals are used for providing semantic information to classification results used for a classifier. In our method, it is sufficient to assign teaching signals with the number of labels created by ART-2. The load of semantic labeling to a category map is lower than that of supervised learning based methods because labels with similar input features are mapped to neighborhood regions. Moreover, our approach is oppositional to semi-supervised learning, by which training signals are partially assigned. In semi-supervised learning based methods, the recognition

performance is affected by the combination of training datasets that are assigned training signals. No rules exist to assign training signals to the set of training datasets. In our method, targets to assign teaching signals are represented as training results. Therefore, the load for teaching is lower than that of semi-supervised based methods. Robots can obtain perceptual recognition capability through minimum interaction with a user.

Next, we present a practical example of using semantic category maps for a robot. Fig. 1.5 portrays a use-case scenario for testing. In this case, a robot moves according to the testing mode. When the robot detects an object, the robot can input this image to the semantic category map. After matching the category map, the robot can recognize the object. This example presents that a person looks for a camera. In the case of signal-based recognition, the person must indicate the position with coordinate values and characters of the object numerically. In contrast, robots can communicate with humans if robots can recognize objects perceptually. Robots can provide user-friendly services using this technology. Previously, humans had to adapt to robots to compensate for the robot's shortage of functions. We consider that the actualization of perceptual recognition can provide new services to enable robots to adapt to humans.

Fig. 1.6 depicts three application examples of this technology. The first application is a secretary use. Numerous documents, goods, devices, etc. exist at an office or a laboratory. Using our technology, the robot organizes them perceptually. The robot works as a secretary from user's requests. We consider that we can have own secretary robot in the near future. The second application is a robot to cleanup objects in the room. In this example, the robot estimates its global position to understand the situation and context automatically. Subsequently, the robot finds the book and returns it to the bookshelf. The third application is a educational use for kids. Kids can learn object names in an environment. Moreover, it can use to learn forging words. Perceptual object recognition is a necessary technology for these applications.

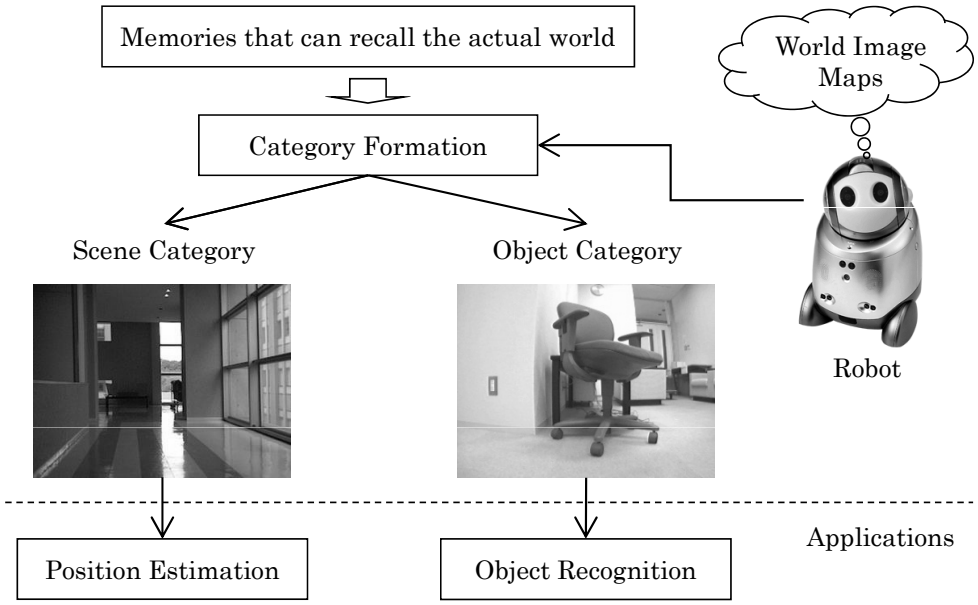


Figure 1.7. Category formation and its applications.

1.3. Thesis aims

This thesis presents an unsupervised category formation method using SOMs for creating category maps as world image maps. For signal-based automatic labeling, we introduce CPNs that are appended the Grossberg layer to SOMs which work for supervised learning. In the primary experiments, this thesis presents basic characteristics of CPNs to improve generalization capabilities of supervised neural networks based on topological data mapping. Using topological data mapping on CPNs, our method provides advantages to interpolate new data in sparse areas that exist among categories and to remove overlapping or conflicting data in original training data. Moreover, the proposed method can control the number of training data by changing the size of the category map according to a problem to be solved.

For practical uses in an actual environment, this thesis presents two applications shown in Fig. 1.7 using a mobile robot. The first application is scene category formation for global position estimation of a mobile robot. Our method can extract changes in landscape revealed by viewing image sequences as concept patterns by SOMs. Effective position information is acquired by making hier-

archical SOMs and using it to consolidate position estimation concept patterns. We evaluate the effect of shifts in position and direction while the robot was executing a trial journey on global position estimation.

The second application is unsupervised category formation for recognizing generic objects perceptually. For this application, we propose an unsupervised category formation method using ART-2 networks and CPNs. Using labels produced by ART-2 for teaching signals of CPNs, signal-based automatic labeling of units on the category map can be realized. Moreover, the combination of SOMs and ART-2 can represent spatiotemporal relations of input data. We evaluate feature representation that contributes to improved accuracy of classification for selecting feature points to concentrate characterized information of an image. Moreover, we evaluate visualization of spatial relations on labels and integrate redundant and similar labels generated by ART-2 as a category map using self-mapping characteristics and neighborhood learning of CPNs.

1.4. Thesis outline

This thesis is consisted of six chapters. The relation of each chapter is shown in Fig. 1.8. Chapter 2 presents a method using CPNs to improve generalization capabilities of supervised neural networks based on topological data mapping. Using topological data mapping on CPNs, our method presented provides advantages to interpolate new data in sparse areas that exist among categories and to remove overlapping or conflicting data in original training data. Moreover, our method can control the number of training data by changing the size of the category map according to a problem to be solved. As a type of supervised neural networks combined with our method, we select SVMs, which are attractive as learning algorithms having high generalization capabilities to be mapped to a high-dimensional space using kernel functions. We applied our method to classification problems of two-dimensional datasets for evaluation of basic characteristics of our method. Topological data mapping based compression of original training data induces resolution of conflict among data and reducing the number of Support Vectors (SVs) that are absorbed as soft margins. The classification results show that decision boundaries are changed and that generalization capabilities

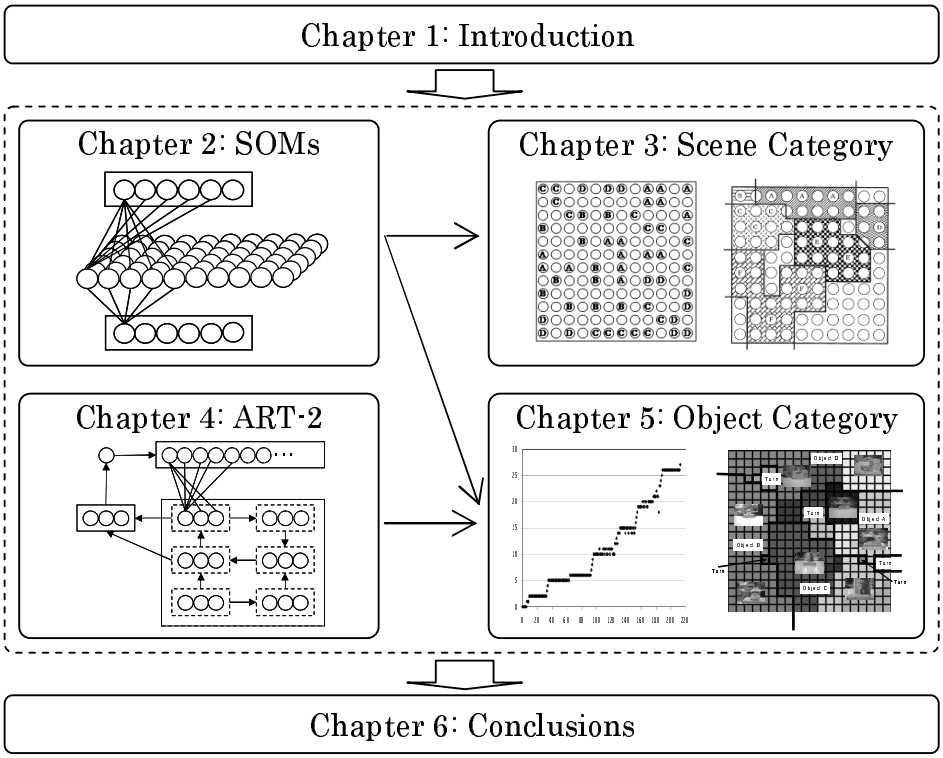


Figure 1.8. Relation of chapters of this thesis.

are improved using our method. Moreover, we applied our method to face recognition under various illumination conditions using the Yale Face Database B [31]. The results indicate that our method provides not only improved generalization capabilities, but also visualizes spatial distributions of SVs on a category map.

Chapter 3 presents a method of scene classification for robotic position estimation that can process position information without identifying special landmarks. The method, which combines viewpoint shifts with visual information about the environment, makes it possible for a robot to move both autonomously and purposefully. In the procedures, the robot surveys the landscape of an environment from multiple directions, obtaining self-localization from a viewing image sequence. For providing a statistical summary of the spatial layout properties of a scene, we use downsampling images for input features [32]. The robot is made aware of changes in the landscape via SOMs, which generate concept patterns. By making the SOM hierarchical, these concept patterns can be con-

solidated. This allows the robot to move both autonomously and purposefully in the environment toward a position by using previously collected information. By performing a travel experiment with the robot in an indoor environment, in which characteristic concept patterns recording topologies had been previously generated at various positions during a learning period, we confirmed that a correct self-localization estimate can be generated from landscape changes detected via viewpoint shifts.

Chapter 4 presents a method using ART networks, which are unsupervised and self-organizing neural networks that contain a stability-plasticity trade off, for representation of facial expression changes using orientation selectivity of Gabor wavelets. The classification ability of ART is controlled by a parameter called the attentional vigilance parameter. However, the networks often produce inclusions or redundant categories. Our method produces suitable vigilance parameters according to classification granularity using orientation selectivity. We evaluated our method using a facial expression dataset that represents the appearance and disappearance of facial expression changes to detect dynamic, local, and topological feature changes of facial expressions.

Chapter 5 presents an unsupervised learning-based method for selection of feature points and object category formation without previous setting of the number of categories. For unsupervised object category formation, this method has the following features: detection of feature points and description of features using a Scale-Invariant Feature Transform (SIFT) [33], selection of target feature points using One Class-SVMs (OC-SVMs) [34], generation of visual words using SOMs, formation of labels using ART-2, and creation and classification of categories on a category map of CPNs for visualizing spatial relations between categories. Classification results of static images using a Caltech-256 object category dataset and dynamic images using time-series images obtained using a robot according to movements respectively demonstrate that our method can visualize spatial relations of categories while maintaining time-series characteristics. Moreover, we emphasize the effectiveness of our method for category formation of appearance changes of objects.

And finally, Chapter 6 presents conclusions and future work of this study.

Chapter 2

Topological Data Mapping for Improving Generalization Capabilities

2.1. Introduction

Neural networks (NNs) are widely applied to many problems that show difficulty of formulation or reformulation because of dynamic, high-dimensional, or non-linear data distributions. Actually, NNs can create mapping relations to extract rules automatically through learning from given datasets. Especially, NNs express a profound impact for the problems that contain variations in input data because NNs can change the processing structures flexibly according to a target problem with incremental learning or re-learning. Especially in computer or robot vision studies that use the required algorithms in each target, NNs can create a classifier only from obtained data. Moreover, NNs are applicable to various applications according to the progress of processing performances of computers. As expanding applications of NNs, advanced and flexible recognition capabilities are necessary for use in various complex environments. In this situation, generalization capabilities are expected to be useful.

The NNs learn one time according to the target problem or data variation if NNs can acquire high generalization capabilities. Especially, high generalization capabilities are necessary in an environment that poses difficulty to the steady

collection of training data. In contrast, data can be too numerous because unknown data equal all data expected of training data. From the viewpoint of training data and learning algorithms, Kita [35] set the following two preconditions dealing with generalization capabilities: 1) NNs can extract some hidden rules constrained by training data; and 2) NNs have a mechanism not only to store or to recall training data, but also to discover rules to constrain the training data.

As described in this chapter, we specifically examine precondition 1) related in training data. This chapter presents a method to control the number of training data for improving the quality of training data using topological data mapping of Counter Propagation Networks (CPNs) [24]. Actually, CPNs are supervised NNs based on Self-Organizing Maps (SOMs) [15] for self-mapping input data to a low-dimensional space of usually one or two dimensions, with teaching signals to be assigned for labels as a category map. Using self-mapping characteristics of competitive learning and neighborhood learning of CPNs, our method can expand and compress training data while retaining the topological structures of original training data. Moreover, our method can change the number of training data concomitantly with changing of the number of units on the mapping layer. Using category maps of CPNs, new training data are interpolated in sparse regions and overlapping data are removed from original training data.

As the precondition 2) related to training algorithms, we use Support Vector Machines (SVMs) [11], which are remarkable NNs with excellent learning and mapping capabilities. Actually, SVMs are known to be able to obtain high performance of recognition and generalization capabilities to convert input data to a higher-dimensional space using kernel functions. At the training step, representative points called Support Vectors (SVs) are selected to gain decision boundaries with maximize margins among categories. The SVMs use training data selected for SVs, not all training data. This mechanism improved the training data quality. In our method, the combination of SVMs and CPNs can realize high generalization capabilities because new training data without overlapping or contradiction are selected from quantity expanded training data using topological mapping characteristics of CPNs.

This chapter consists of the following sections. We review related work in 2.2

for setting the position of our study. In 2.3, we explain important details of our method. In 2.4, we present basic characteristics of our method. We apply our method to a real problem of various illumination conditions using a large-scale database in 2.5. We discuss relations between category maps and SVs in 2.6. Finally, we conclude in 2.7.

2.2. Related studies

Various methods based on training data have been proposed especially in expansion of whole training data quantitatively [36, 37, 38, 39, 40, 41, 42]. Holmstrom et al. proposed a method to expand training data to add Gaussian-type white noise [36]. Karystinos et al. proposed a method to expand training data randomly based on probability density functions [37]. Although these methods can expand training data easily, they might not fulfill Kita's preconditions because noise or random expanded data have no hidden rules. In contrast, Tanaka et al. proposed a method to expand training data according to the distance from the center of categories [41]. Although this method is superior to methods used in random noise, they are only used for the situation in the distribution of sparse data among categories with readily apparent decision boundaries of categories.

Existing methods based on learning algorithms are proposed variously: a method for division into subnets by Chakraborty et al. [43], a method used in double-back propagation by Drucker et al. [44], a method to delete redundant units on the hidden layer by Matsunaga et al. [45], methods to tune weights [46, 47, 48, 49], active learning based methods [50, 51] etc. Tsuda [52] described that excellent learning algorithms have three features: high recognition rates for experiments, a theoretical basis, and easy realization. The SVMs combine high-recognition performance, especially in recognition programs, a theoretical basis based on the framework of Probably Approximately Correct (PAC) learning, and a calculation method leading to a quadratic programming problem. Therefore, we used SVMs as a classifier for advanced improvement of generalization capabilities.

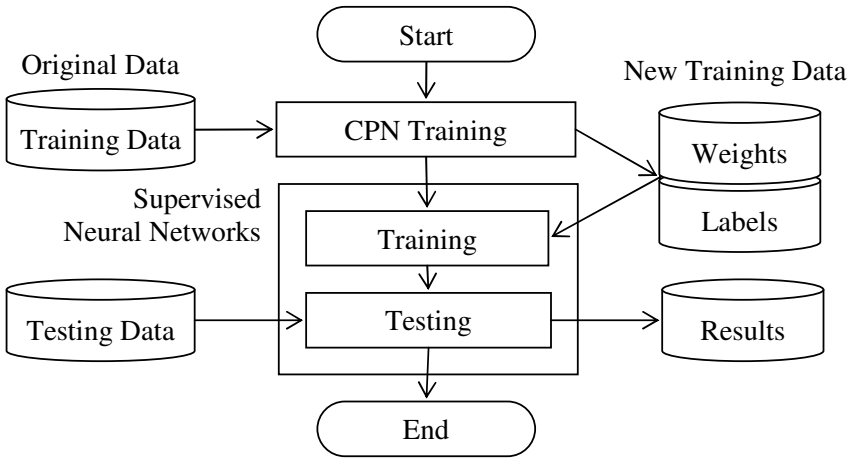


Figure 2.1. Procedure of our method. Weights and labels of CPN are used as training data of SVM.

2.3. Proposed method

To move from quantity control to quality improvement necessitates creation of data that are interpolated from sparse data and deletion of redundant, overlapping, and conflicting data. We specifically examine the topological mapping characteristic on CPNs. This chapter presents a method to improve generalization capabilities in aspects of quality improvement of training data using weights and labels created with CPNs. The following describes the overall architecture of our method and the respective learning algorithms used with CPNs and SVMs.

2.3.1 Whole architecture of our method

Fig. 2.1 depicts the procedures used for our method. First, CPNs are trained using original training data. All units of the mapping layer on the CPN are labeled automatically using teaching signals. The labeled units are called category maps. After learning of CPNs, new training data are created: the weights between the input layer and the mapping layer are used for new training data; the labels on the category map are used for new teaching signals. New training data are created while retaining topological structures of original data. Our method can control the number of new training data arbitrarily by changing the number of

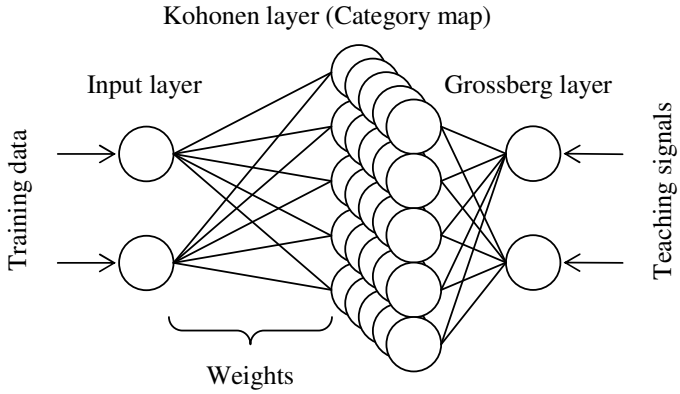


Figure 2.2. Network architecture of the CPN.

units on the mapping layer.

For the feature of our method, supervised NNs as a classifier are naive to the original training data. The NNs are trained using topological expanded or compressed data with CPNs. The CPNs map input data into a topological space as a category map with neighborhood training and Winner-Takes-All (WTA) competition. New data are interpolated with neighborhood learning and overlapping data are deleted through the WTA. The reason CPNs are not used as a classifier is that the CPN's inventor Nielsen described that the classification performance of CPNs is insufficient as a classifier in comparison to supervised NNs such as SVMs, Back-Propagation Networks (BPNs) [58], etc.

2.3.2 Counter Propagation Networks

The CPNs are supervised and self-organizing neural networks that combine Kohonen's competitive learning algorithm and Grossberg's outstar learning algorithm. The network comprises three layers: an input layer, a Kohonen layer, and a Grossberg layer. Fig. 2.2 shows the network architecture of CPNs. The input layer propagates training data. The Kohonen layer performs topological mapping through the WTA competition. The Grossberg layer propagates teaching signals and assigns labels to all units of the Kohonen layer. The labeled units are called category maps. In our method, the Kohonen layer contains two-dimensional units; the Input layer and the Grossberg layer contain one-dimensional units.

The CPN training algorithm is the following. Let $u^i_{n,m}(t)$ be the weight from the input unit i to the Kohonen unit (n, m) at time t. Let $v^j_{n,m}(t)$ be the weight from the Grossberg unit j to the Kohonen unit (n, m) at time t. These weights are initialized using random numbers. Let $x_i(t)$ be the input data to the input unit i at time t. The Euclidean distance $d_{n,m}$ between $x_i(t)$ and $u^i_{n,m}(t)$ is calculated as

$$d_{n,m} = \sqrt{\sum_{i=1}^{I}(x_i(t) - u^i_{n,m}(t))^2}. \tag{2.1}$$

The win unit c is defined, for which $d_{n,m}$ becomes a minimum by

$$c = argmin(d_{n,m}). \tag{2.2}$$

Let $N_c(t)$ be the units of the neighborhood of the unit c. The weight $u^i_{n,m}(t)$ inside $N_c(t)$ is updated using the Kohonen training algorithm as

$$u^i_{n,m}(t+1) = u^i_{n,m}(t) + \alpha(t)(x_i(t) - u^i_{n,m}(t)). \tag{2.3}$$

The weight $v^j_{n,m}(t)$ inside $N_c(t)$ is updated using the Grossberg outstar training algorithm as

$$v^j_{n,m}(t+1) = v^j_{n,m}(t) + \beta(t)(t_j(t) - v^j_{n,m}(t)). \tag{2.4}$$

Therein, $t_j(t)$ is the teaching signal to be supplied from the Grossberg layer, $\alpha(t)$ and $\beta(t)$ are the training coefficients that decrease with time. Training is finished when its iterations reach the maximum number. In our method, $\alpha(t)$ and $\beta(t)$ are set respectively as 0.5 and 0.9. The maximum number of training iterations is set as 1,000 steps.

2.3.3 Back Propagation Networks

The training algorithm of BPNs is as follows. Let $w_{ij}(t)$ be the weight between input unit i and hidden unit j. When input data $x_i(t)$ are supplied to the input units, the output $h_j(t)$ of hidden units are

$$h_j(t) = f(\sum_{i=1}^{I} x_i(t)w_{ij}(t)), \tag{2.5}$$

where f is a sigmoid function defined as

$$f(x) = \frac{1}{1 + e^{-x}}. \tag{2.6}$$

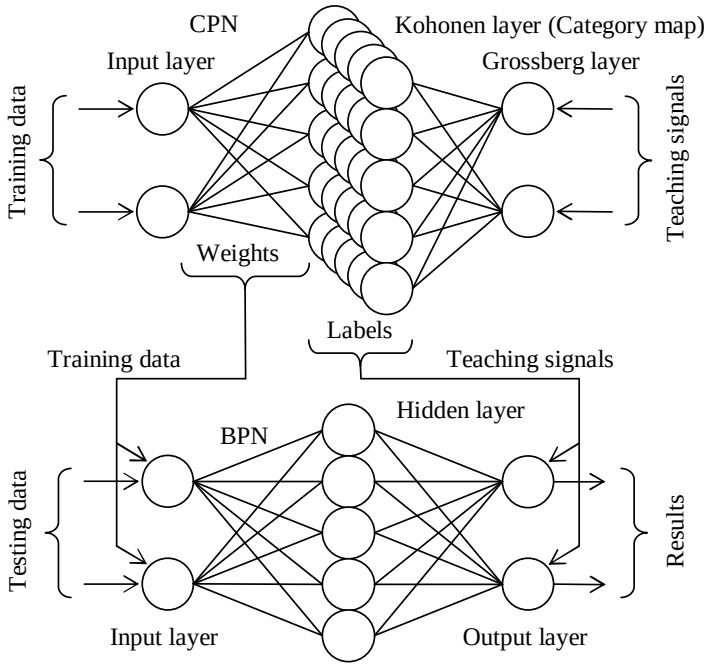


Figure 2.3. .

Let $w_{jk}(t)$ be the weight between the hidden unit j and the output unit k. The output $o_k(t)$ of the output layer is

$$o_k(t) = f(\sum_{j=1}^{J} h_j(t) w_{jk}(t)). \tag{2.7}$$

Let E be the mean square error defined as

$$E = \frac{1}{2} \sum_{K=0}^{K} (t_k(t) - o_k(t))^2, \tag{2.8}$$

where $t_k(t)$ is the teach signal supplied to the output layer. The aim of the BPN training is to reduce E with updating weights. Let $\Delta w_{kj}(t)$ and $\Delta w_{ji}(t)$ be the updating value of $w_{jk}(t)$ and $w_{ij}(t)$ as

$$\Delta w_{kj}(t) = \eta \delta_k h_j(t) + \alpha \Delta w_{kj}(t-1), \tag{2.9}$$

$$\begin{aligned} \Delta w_{ji}(t) &= \eta h_j(t)(1 - h_j(t)) \\ &\quad (\sum_{k=1}^{K} w_{kj}(t)\delta_k) + \alpha \Delta w_{kj}(t-1), \end{aligned} \tag{2.10}$$

$$\delta_k = (t_k(t) - o_k(t))o_k(t)(1 - o_k(t)), \tag{2.11}$$

where η and α are the learning coefficients related to convergence and stability. We set $\eta = 0.1$ and $\alpha = 0.3$. The weights $w_{jk}(t)$ and $w_{ij}(t)$ are updated as follows:

$$w_{jk}(t+1) = w_{jk}(t) + \Delta w_{kj}(t), \tag{2.12}$$

$$w_{ij}(t+1) = w_{ij}(t) + \Delta w_{ij}(t). \tag{2.13}$$

The learning finished when its iterations reached 1,000,000 steps or when E became less than 0.001.

2.3.4 Support Vector Machines

Actually, SVMs are linear classifiers based on a two-class classification using kernel functions. Since discovery of a calculation method using kernel tricks with kernel functions for replacement from a nonlinear space to a linear space of high dimensions, SVMs have come to be used popularly for numerous applications because of their high classification and generalization capabilities.

The learning of SVM is to calculate the bias b, weights $\boldsymbol{w}$ of the discriminant function f as N sets of input data $\boldsymbol{x_i}(i = 1, ..., N)$ defined as the following:

$$f(\boldsymbol{x}) = sign(\boldsymbol{w^T x} - b), \tag{2.14}$$

where $sign(u)$ is a step function to output 1 at $u > 0$ and -1 at $u \leq 0$. Presuming that the teaching signal is $t_i(i = 1..., N)$ with respect to $\boldsymbol{x_i}$, then the hyperplane of the margin is maximum in two classes calculated using the minimization problem as

$$L(\boldsymbol{w}, \boldsymbol{\epsilon}) = \frac{1}{2}||w||^2 + \sum_{i=1}^{N} \epsilon_i, \tag{2.15}$$

The second term of $\epsilon_i(\geq 0, i = 1..., N)$ is a parameter permitting incorrect classifications for the input data that are difficult to classify linearly. This mechanism is called the soft margin method. The minimization problem of L is solvable using Lagrange undetermined multipliers. When Lagrange multiplier α is introduced, then L is calculated as

$$L(\boldsymbol{w}, b, \boldsymbol{\alpha}) \quad = \quad \frac{1}{2}||w||^2 + \sum_{i=1}^{N} \epsilon_i$$

$$-\sum_{i=1}^{N}\alpha\{t_i(\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x_i}-b)-(1-\epsilon_i)\}$$
$$-\sum_{i=1}^{N}\epsilon_i, \tag{2.16}$$

subject to partial differential at $\boldsymbol{w}$

$$\boldsymbol{w}=\sum_{i=1}^{N}\alpha_i t_i \boldsymbol{x_i}, \tag{2.17}$$

subject to partial differential at b

$$\sum_{i=1}^{N}\alpha_i t_i = 0. \tag{2.18}$$

Substituting them, the objective function is

$$L_d(\boldsymbol{\alpha})=\sum_{i=1}^{N}\alpha_i-\frac{1}{2}\sum_{i,j=1}^{N}\alpha_i\alpha_j t_i t_j \boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{x_i}. \tag{2.19}$$

Actually, $\alpha_i(\geq 0)$ is calculated to solve the quadratic programming optimization problem subject to these constraining conditions. In addition, $\boldsymbol{x_i}$ subject to $\alpha_i > 0$ is selected to SVs on the hyperplane $\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x_i}-b=\pm 1$. In fact, b is calculated based on the definition of the hyperplane as

$$b=\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x_i}\pm 1. \tag{2.20}$$

To introduce a nonlinear mapping function Φ to a high-dimensional feature space, Eq. (2.19) is calculated as

$$L_d(\boldsymbol{\alpha})=\sum_{i=1}^{N}\alpha_i-\frac{1}{2}\sum_{i,j=1}^{N}\alpha_i\alpha_j t_i t_j \Phi(\boldsymbol{x}^{\boldsymbol{T}})\Phi(\boldsymbol{x_i}), \tag{2.21}$$

where the inner product $\Phi(\boldsymbol{x})^T\Phi(\boldsymbol{x_i})$ is calculable using the following trick by the kernel function K on the Hilbert space as

$$\Phi(\boldsymbol{x})^T\Phi(\boldsymbol{x_i})=K(\boldsymbol{x},\boldsymbol{x_i}). \tag{2.22}$$

Kernel function K uses the polynomial kernel, the Radial Basis Function (RBF), and the Sigmoid kernel, etc. In this study, we used RBF defined as

$$K(\boldsymbol{x},\boldsymbol{x_i})=exp(-\frac{||\boldsymbol{x}-\boldsymbol{x_i}||^2}{\lambda}), \tag{2.23}$$

where λ is the variance of RBF. Because the property of the Kernel differs in the setting of λ, we evaluate our method using results to change in a certain range.

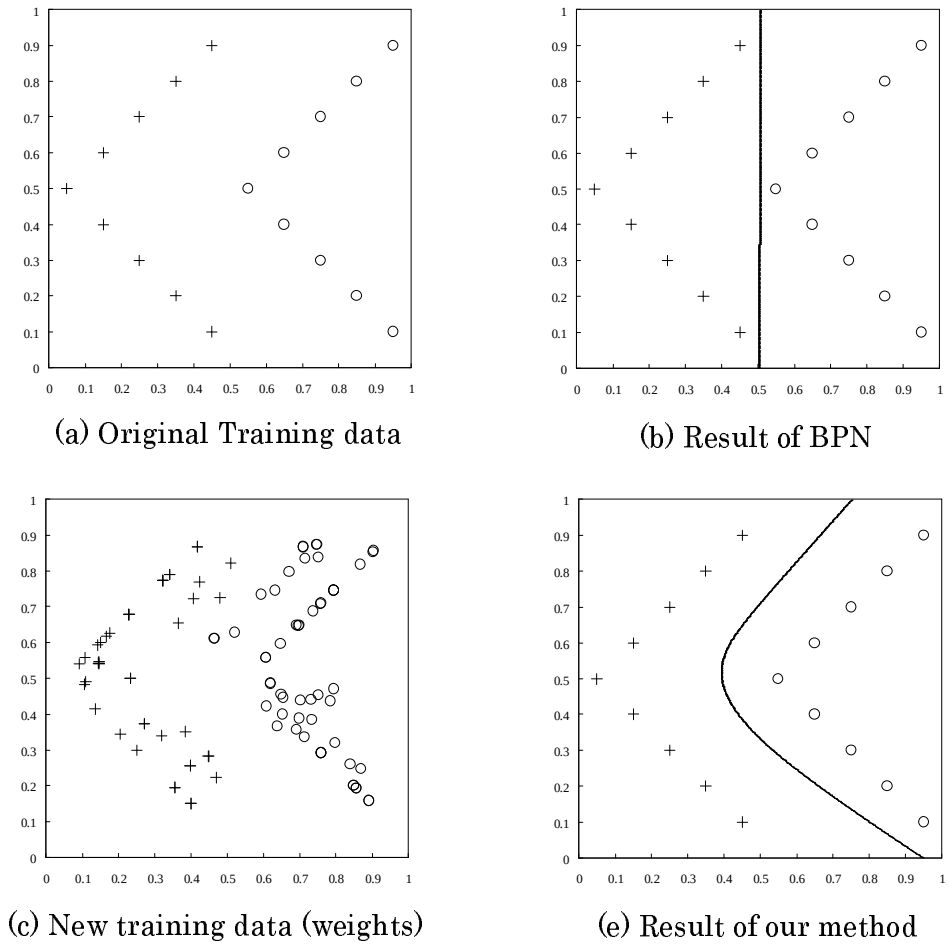


Figure 2.4. Classification of Arrows dataset.

2.4. Classification

We verify basic generalization capabilities of our method for classification benchmarks that can easily yield distributions of input data and classification results in a two-dimensional space. In this experiment, we evaluated our method using open datasets of two types: the Normal Mixtures dataset [53] and the Cone-Torus dataset [54], which are widely used for evaluation of generalization capabilities.

2.4.1 Arrows

We originally made a dataset for verification of the interpolating function. The dataset shown in Fig. 2.4(a) consists of two classes with 18 points. We named the dataset the Arrows dataset. Both classes are distributed in the shape of arrows; a wide space separates them.

For interpolating training data, we established the category map as 10 units $\times$ 10 units, as shown in Fig. 2.4(b). Fig. 2.4(c) shows the new training data obtained from the category map. Their labels correspond to the labels shown in the category map. The space between clusters was interpolated using the new training data. Fig. 2.4(d) shows the decision boundary that was obtained using our method. The curved decision boundary indicates that our method reflects the actual data distribution.

We compared our method with a normal BPN. Fig. 2.4(e) shows the decision boundary of the normal BPN. Although the normal BPN correctly classifies both clusters, the linear decision boundary indicates that the normal BPN did not reflect the data distribution.

2.4.2 Squares

Fig. 2.5(a) depicts classification of two clusters that are distributed like squares. We call it square clustering. The solid diamond points show the distribution of the large square cluster. The solid circle points denote the distribution of the small square cluster surrounded by the large one. Fig. 2.5(b) shows the result of the standard BPN trained by the original data points. The decision boundaries divided the large square cluster into two independent regions. Fig. 2.5(c) shows the result of our method. The decision boundary exists between the two clusters. A cluster of solid diamond points is arranged in a single region outside of the decision boundary.

2.4.3 Normal mixtures dataset

The Normal Mixtures dataset [53] created by Ripley et al. comprised two classes of 250-point training data and two classes of 1000-point testing data. In this dataset, some data points are overlapped around boundaries between clusters.

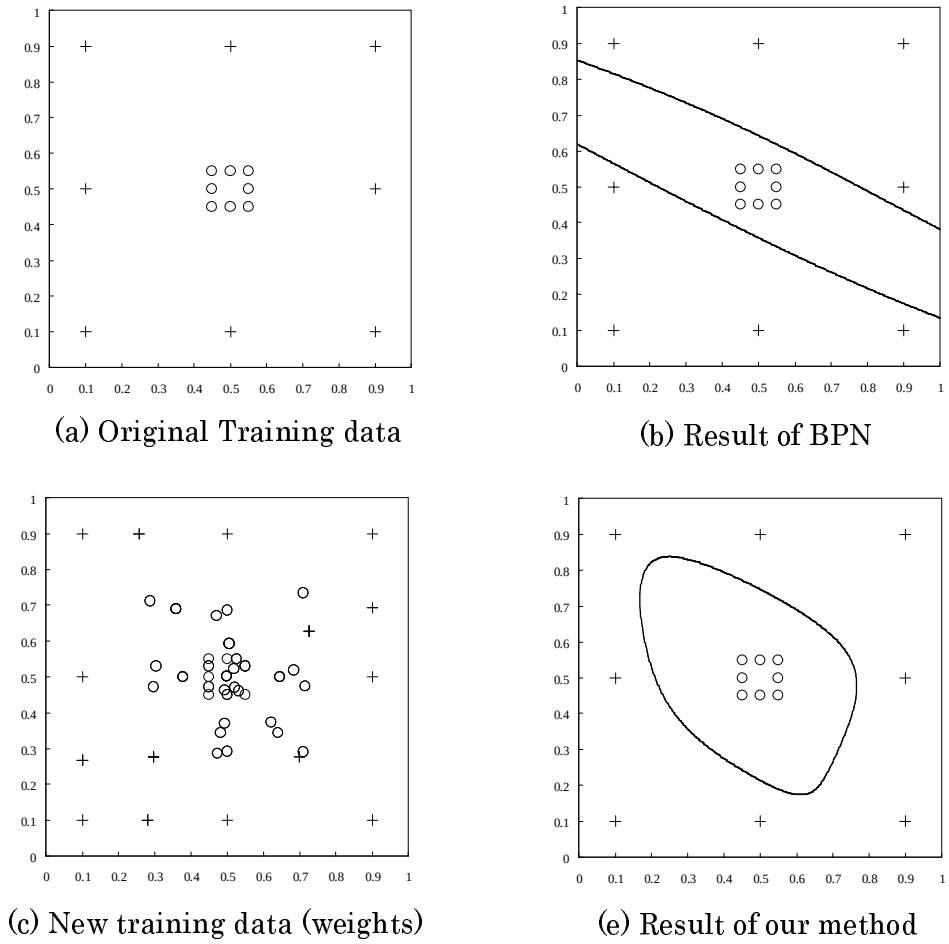


Figure 2.5. Classification of squares dataset.

Fig. 2.6 shows comparison results of error rates of the conventional SVM trained with original data and our method. We used category maps of three sizes: 10×10 units, 15×15 units, and 20×20 units. We changed λ, which shows the variance of RBF of Eq. 2.23 from 0.01 to 1.00 step by 0.01 and shown the results in this figure. Comparison results reflect that the error rates of our method are greatly decreased compared with results obtained using the conventional SVM. Especially, the results of 10×10 units indicate the minimum error rate.

Fig. 2.7(a) portrays classification results obtained using conventional SVM with original training data. The data points surrounded by circles represent training data selected as SVs. In the case of original data, many SVs that are

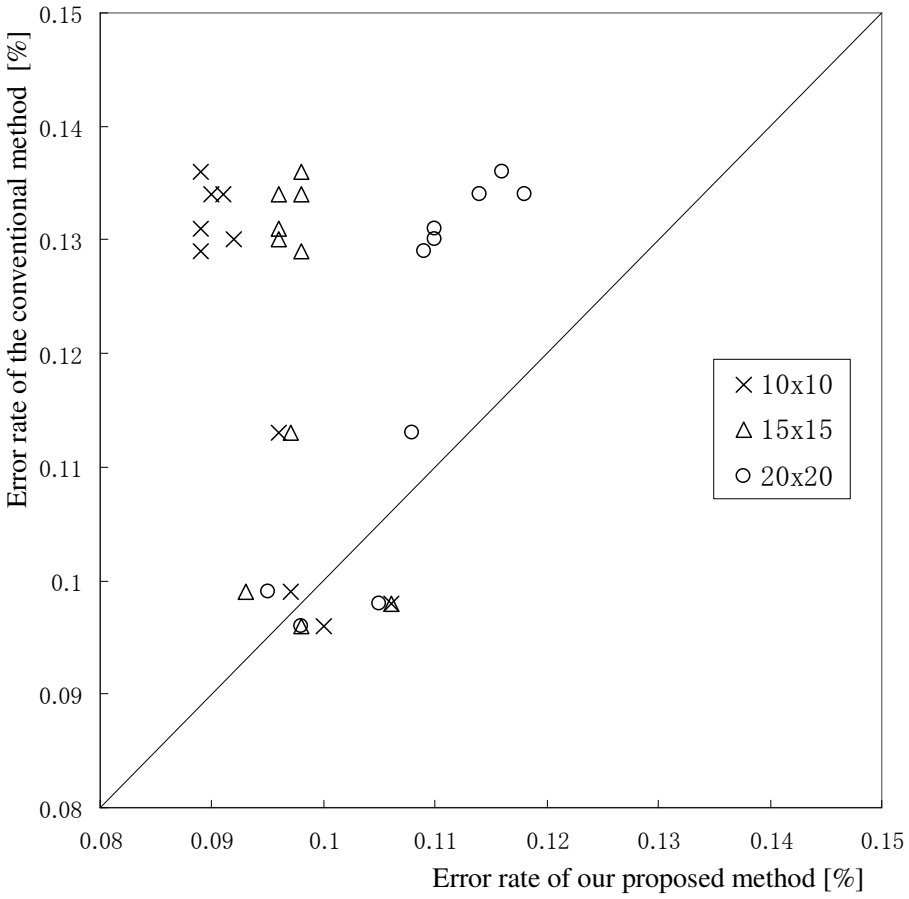


Figure 2.6. Comparison results of error rates of the Normal Mixtures dataset with change of λ and the size of category maps.

merged as a soft margin are visible. Fig. 2.7(b) portrays the classification results obtained using our method. We set the category map 10×10 units based on the comparison result presented above. The original training data are 250 points. In this case, the training data are compressed to 40 percent. We consider that compression was valid because original data exist sufficiently compared with the complexity of the data distribution. The SVs that are merged as a soft margin are reduced because overlapping data are removed with mapping characteristics of CPNs. The minimum error rate for the test dataset is 8.80 percent. Compared with the minimum error rate of 9.50 percent the conventional SVM, the error

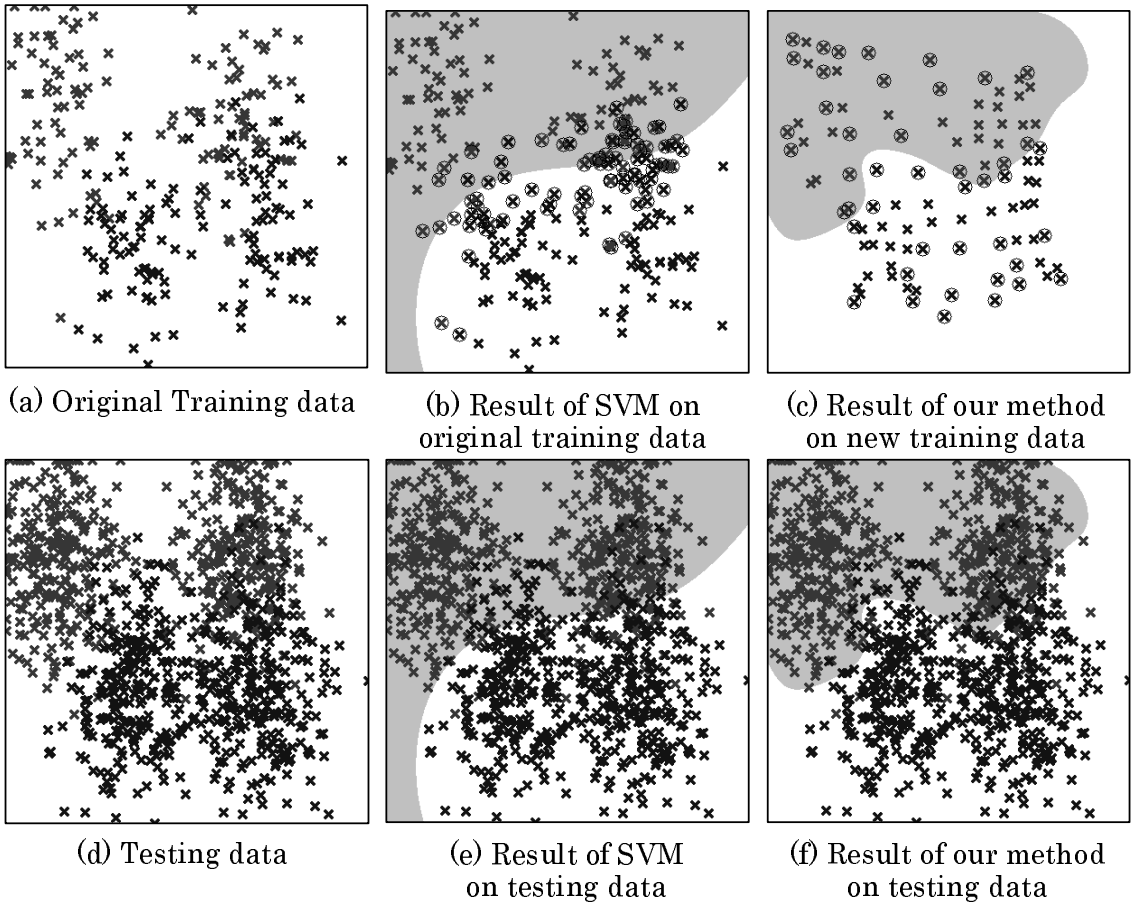


Figure 2.7. Classification results of the Normal Mixtures dataset.

rate is reduced 0.70 percent.

2.4.4 Cone-torus dataset

The Cone-Torus dataset [54], created by Kuncheva et al., includes three classes of 400-point training data and three classes of 400-point testing data. The data are distributed in a cone shape, a torus shape, and a Gaussian shape that is overlapped between them.

Fig. 2.8 shows error rates of the conventional SVM trained by original data and our method in the case of 10×10 units, 20×20 units, and 30×30 of the category map. In the comparison results shown in this figure, the category map

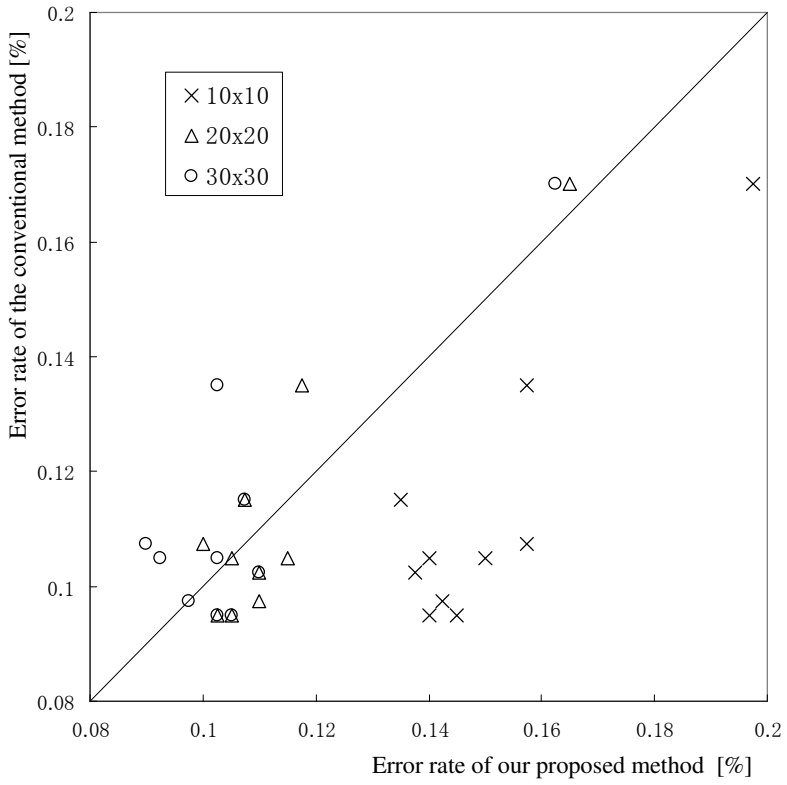


Figure 2.8. Comparison results of error rates of the Cone-Torus dataset with changing of λ and the size of category maps.

of 30×30 units is the minimum of the error rate. In this case, the training data are expanded to 225 percent.

Fig. 2.9 portrays the decision boundary and SVs obtained using the conventional SVM and using our method. We consider that the category map with a larger number of units that can create more numerous new training data is valid because this dataset contains overlapping data and complex boundaries in the data distribution. The minimum error rates for the test dataset are, respectively, 9.00 and 8.50 percent using the SVM trained using original data and our method. Therefore, the generalization capability is improved 0.50 percent using our method. In [43], the error rate using the same dataset with the method presented by Chakraborty et al. is 14.75 percent. Compared with the results, the error rate is improved 6.25 percent using our method.

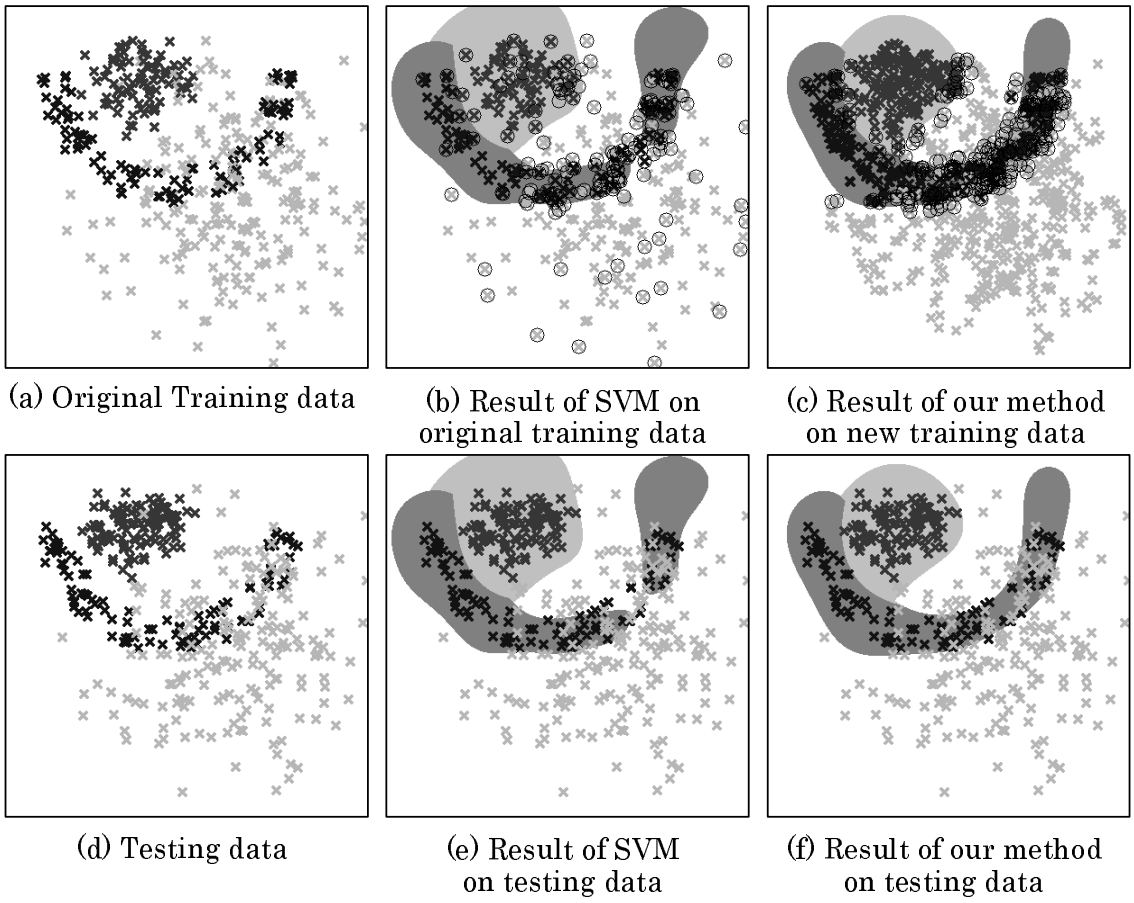


Figure 2.9. Classification results obtained using the Cone-Torus dataset.

2.5. Face recognition under various illumination conditions

In problems of high-dimensional input data such as image recognition, showing the existence of a hidden rule or not is a challenging task. Therefore, most problems are set to the evaluation target of generalization capabilities for the stability of outputs of NNs to the datasets to insert variations in the range that can recognize visually. In contrast, to know a priori that the target problem exists inside or outside using generalization capabilities over the Kita's precondition described above is unknown. Therefore, we consider that using a database with

Figure 2.10. Sample images of the Yale Face Database B in each subset.

which a hidden rule can be evaluated step-by-step is necessary. We use the Yale Face Database B [31], which is an open dataset, to treat various illumination conditions step-by-step.

2.5.1 The Yale face database B

This database consisted of facial images of 10 subjects with 64 illumination conditions of different azimuths and elevations. The database is separated to five subsets by azimuths and elevations of the lighting source. In appearance-based facial recognition processing, the feature difference of illumination conditions is greater than the difference among subjects.

In this experiment, we used Subset 1 for training and Subsets 2–4 for testing. In [55], Okabe et al. described that Subset 5 used for evaluation is invalid because the error rate reached 90 percent in the experimental result with their method using illumination cones. This rate is the same as the result for recognition at random. Therefore, we use no Subset 5.

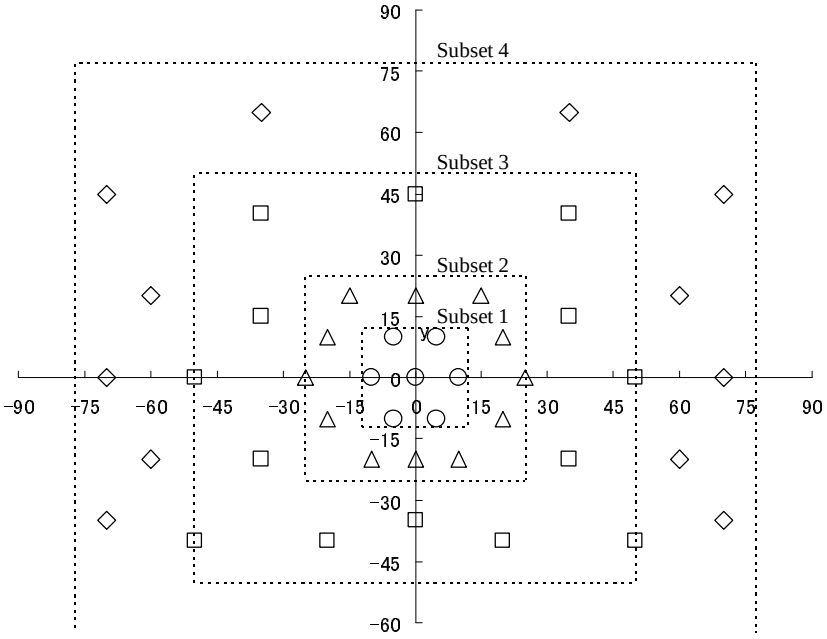


Figure 2.11. Positions of strobes corresponding to the images of each illumination subset.

2.5.2 Preprocessing

The original images are 256-gray-level images. The resolution is 640×480 pixels. We used only frontal images that are assigned two-dimensional coordinate points of the eyes and mouse. Using the coordinate points, the face region can be extracted easily. Lee et al. released the Extended Yale Face Database B [56] of 28 subjects to be the extracted face region of 168×192 pixels. For this experiment, we used this database after preprocessing of the histogram equalization and median filtering. Although the image quality of the low-contrast parts is improved with the histogram equalization, noise pixels were apparently affected by the histogram extension. We use a median filter for removing the noise. Subsequently, we conducted downsampling to 320×240 pixels for reducing the effect of head movements. Moreover, we used Principal Component Analysis (PCA) to reduce the number of dimensions of the input feature vectors [57]. We extracted up to the 50th feature value and used it as input data for the CPN. The accumulated contribution rate until the 50th component is 99.95 percent. Regarding the

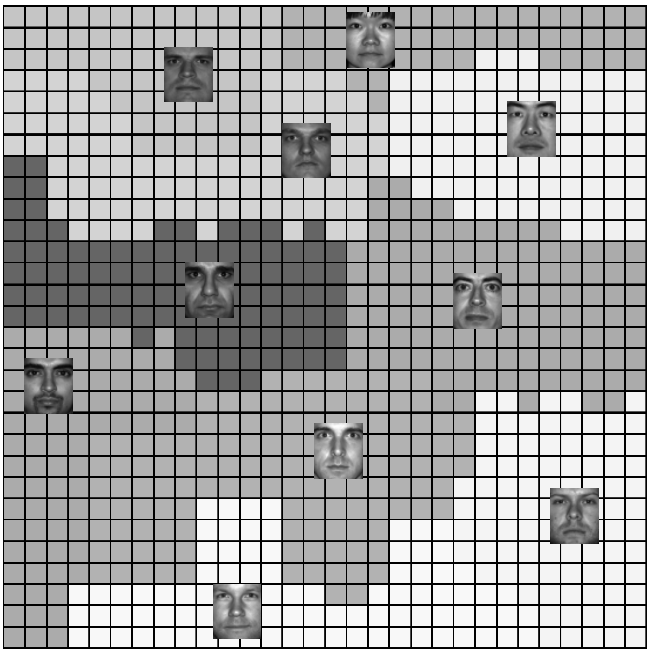


Figure 2.12. Category map (Face images without illumination changes show a person of each category).

robustness against illumination conditions, Okabe et al. obtained a good result with their method to use illumination cones [55]. We specifically examine simplicity of implementation to evaluate generalization capabilities in this experiment. Therefore, we do not use illumination cones.

2.5.3 Classification results

Fig. 2.12 portrays a category map that was generated by CPNs as a learning result. The set of weights and labels corresponding to each unit on the category

Table 2.1. Comparison of the minimum error rates.

Method	Subset 2	Subset 3	Subset 4	All
Conventional SVM	0.00%	26.43%	53.33%	26.58%
Proposed	0.00%	7.14%	40.83%	15.53%

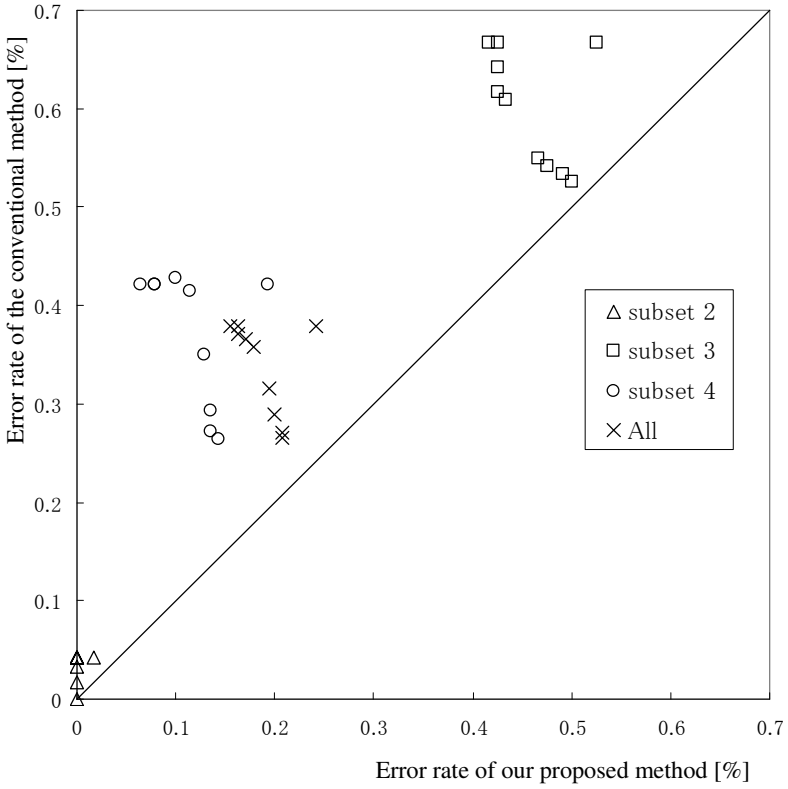


Figure 2.13. Comparison of results of error rates with changing of λ and the size of category maps.

map is used for the new training data. Using the category map, spatial relations of input data can be visualized. The categories contain no bias or discrete regions. Independent categories are created in each subject with similar features.

Fig. 2.13 portrays results of a comparison of error rates in each subset using the original data and our method. We changed λ from 0.1 to 1.0 step by 0.1 repeated 10 times. In all results of our method, the error rates are lower than those obtained using the conventional SVM trained by original data. Table 2.1 shows the minimum error rates in each subset. Especially in Subset 3, the error rate is dominantly decreased to 19.29 percent. The maximum recognition rate is 11.05 percent; the minimum error rates of the conventional SVM and our method are, respectively, 26.58 and 15.53 percent.

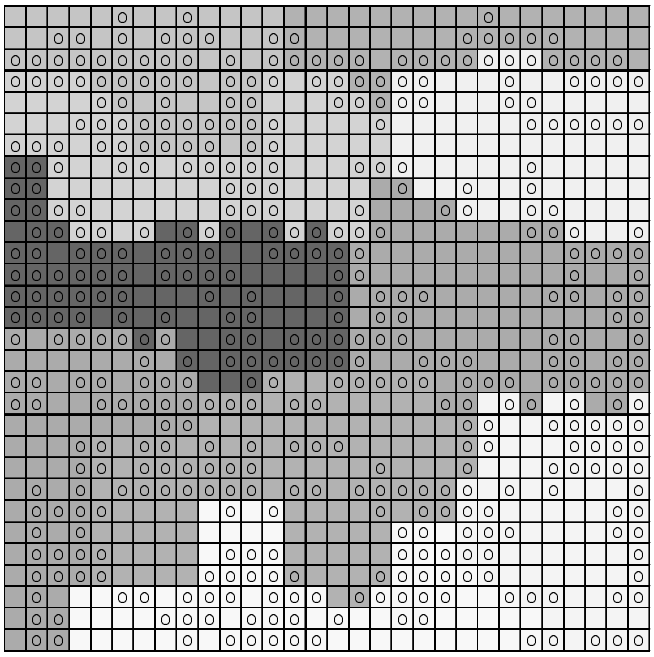


Figure 2.14. SV units on the category map.

2.6. Discussion

First, we verify the combination of CPNs with other supervised NNs except for SVMs. Our method based on training data can combine any supervised NNs. As popularly used NNs, BPNs [58] are used in various applications. We combined with BPNs and conducted the experiment with the same conditions. The minimum error rate with CPNs and BPNs is 21.83 percent. Similarly, the minimum error rate with BPNs is 26.05 percent. The improvement of generalization capabilities is only 4.21 percent. We consider that the effect for topological mapping of learning data with this combination is insufficient because BPNs learn using all training data for getting a mapping relation. In contrast, we consider that the combination with SVMs enhance both characteristic features because training data are examined as SVs with SVMs in case of expanding of training data with a category map.

Along with changing the size of the category map, our method can change the total number of training data arbitrarily. This means that our method can expand or compress the number of training data according to a target problem. From

the experimental results, the effect of improving generalization capabilities of expansion is greater than that of compression. This result supports the knowledge of quantitative retention of data, which improves quality. Feature points around decision boundaries are selected as SVs. In contrast, new training datasets are created based on the whole distribution of feature points with our method using topological mapping characteristics of CPNs. The convergence of error rates of BPNs is decreased using these datasets. We consider that this is the reason for peaking of the improvement of generalization capability with BPNs. In SVMs, data points except for decision boundaries are not selected as SVs. We consider that this is the reason to improve the error rate than BPNs. The SVs are selected only from feature points of the original data. In contrast, our method can create new feature points expect of the original feature points based on topological structures. Therefore, these SVs contribute to improvement of generalization capabilities.

Subsequently, data points that contribute to creation of decision boundaries as SVs can be visualized as a category map using our method. Units that are selected as SVs are depicted as circles on Fig. 2.14 in the category map presented in Fig. 2.12. Unlike the clustering problems on a two-dimensional space, it is difficult to see the distributions of SVs to be selected for deciding the classification accuracy and decision boundary when the dimensions of input features are numerous. Fig. 2.14 portrays that selected units as SVs are distributed around the boundaries. Our method can visualize the spatial distribution of SVs that create hyperplanes from a category to map any high-dimensional input data. In addition, a similarity and neighboring relation among SVs can be elucidated using category maps. Moreover, we consider that SVs that are absorbed as soft margins can be visualized, although such SVs are not apparent in this experiment.

2.7. Conclusion

This chpter presents a method to improve generalization capabilities using expansion or compression of training data while retaining topological structures using topological mapping characteristics of CPNs. We applied our method to classification problems of two types: Normal Mixtures dataset and a Cone-Torus

dataset. Compared with classification results, our method is superior to the conventional SVM using original training data. Moreover, we applied our method to the face recognition problem under various illumination conditions using the Yale Face Dataset B. The error rate is decreased by 11.05 percent compared with the conventional SVM and the generalization capability is improved using our method. Additionally, we visualized the distribution of data points to be selected as SVs on the category map using our method. We ascertained that SVs are distributed around the boundaries on the category map.

In our method, we selected the best size of category maps. The suitable training data are different in each problem to be solved. Automatic setting of the size of category maps is the subject of our future work. Moreover, we will apply our method to large-scale problems.

Chapter 3

Scene Category Formation and Position Estimation

3.1. Introduction

It is anticipated that human-friendly robots capable of every types of autonomous movement will eventually be created for general-purpose use in environments such as office, home, sickrooms, etc. Regarding these situations, one important task that mobile robots must be able to perform is position estimation, which in turn requires a basic mechanism whereby the robot can recognize its own position in the environment. Ordinarily, sensors of distance traveled, e.g., ultrasonic or infrared sensors, are used for self-position of a robot. Because a sensor of travel distance accumulates errors due to interruption of the travel by slipping of wheels, vibration, etc, the total error increases with the distance traveled until the position information stored in the robot's interior becomes unreliable. Ultrasonic sensors imply interference and wraparound, and hence precise sensing of the environment becomes difficult due to the high degree of directionality required. Yamada et al. used infrared sensors for environmental recognition [59]. However, because the range of an infrared sensor is short, motion along a wall becomes necessary, which to some extent prevents the robot from classifying the room in which it is moving.

Recently, vision-based mechanisms have attracted the notice of researchers as potential robot sensors. The methods proposed in [60, 61, 62] for position

estimation used vision sensors, as did the landscape-based methods proposed in [63, 64, 65, 66, 67, 68, 69]. In the earlier references, it was found that the need for prior establishment of landmarks in order to create models that extract the distinctive features in the environment limits the range of motion. Furthermore, when obstacles are present in the environment, mistakes are easily made in the landmark extraction process, with possibly fatal consequences for the robot. In the later references, the authors note that changes can easily be made in a model based on visual information obtained, directly or otherwise, from a camera. Because this information is simple to store in memory, methods based on visual information are found to be robust even in a complicated environment.

An unusual number of method have been proposed for landscape-based self-position estimation. For example, Nishimura et at. used non-monotonic continuous Dynamic Programming (DP) to make spotting position estimations that are independent of the robot's direction of travel [66]. Maeda et al. used a parametric eigenspace method to reduce the volume of memory needed to store images. For cases where several positions generate similar landscapes, the eigenspaces are also similar, which alerts the robot to possible problems. This leads to efficient position estimations [67]. In the work of Georg et al., environment maps were generated in a self-organizing way from distinctive feature vectors of a scene, resulting in closely-spaced position estimations within the environment [68]. However, in all of these familiar methods, even after their respective algorithms are used to extract distinctive features and decrease the amount of memory volume, the number of images that can be stored in bulk memory is still not enough to use for position estimation by matching with images obtained at test travel time. Accordingly, it is a challenging task to respond in a flexible way to changes in the scene caused by differences in brightness, movement of local objects, etc. for each environment used. On the other hand, SOMs proposed by Kohonen [15] are able to store topology of a scene expressed in images. Because these distinctive features are represented by activated states of the neurons, it becomes possible to decrease the memory capacity required for a position estimate map over the full range of image sequences. Moreover, since SOMs require no explicit teaching data, it is expected that personal world image maps reflecting global changes of scene in the environment can be generated whose quality approaches that of

human memories.

Human beings depend on the images they take in form the outside world in order to generate world image maps in the brain, which they then use to verify their personal positions. In our research we have allowed this idea to guide us in developing a technique a robot can use to make self-position estimates. Our method uses the changes in an SOM caused by changes in the landscape to generate world image maps. A special feature of our technique is the concept pattern defined as the encoding of topological information shifts. By superimposing these concept patterns, the robot acquires a characteristic world image maps relative to landscape position, which it stores in its interior. We anticipate that by using concept patterns and world image maps, which are modeled after human memory procedures, we can achieve a degree of robustness in self-position that is unrealizable by more orthodox methods. The results we processes the topological features of a scene to becomes hierarchical, we can consolidate the various portions of a concept pattern, leading to correct self-position.

In Section 3.2 we describe how our self-position method uses the hierarchical SOMs, in Section 3.3 we assess a viewing image sequence parameters, in Section 3.4 we present the results of position estimate experiments, and in Section 3.5 we evaluate our method. Finally, in Section 3.6 we describe the results of a field test we performed in a hospital with a view to making the method application-friendly, which tested teh effectiveness of our method.

3.2. A method for position estimation using hierarchical SOMs

3.2.1 Obtaining the viewing image sequence

In [69], the authors proposed omnidirectional sensors as a system for robot vision, because they can take an entire environment all at once. However, because of the special imaging equipment needed to obtain omnidirectional images, in our research we used ordinary images obtained from a CMOS or CCD camera built into the robot as visual sensors. Because the images are obtained in one direction only under these conditions, they can only record minor changes in the local

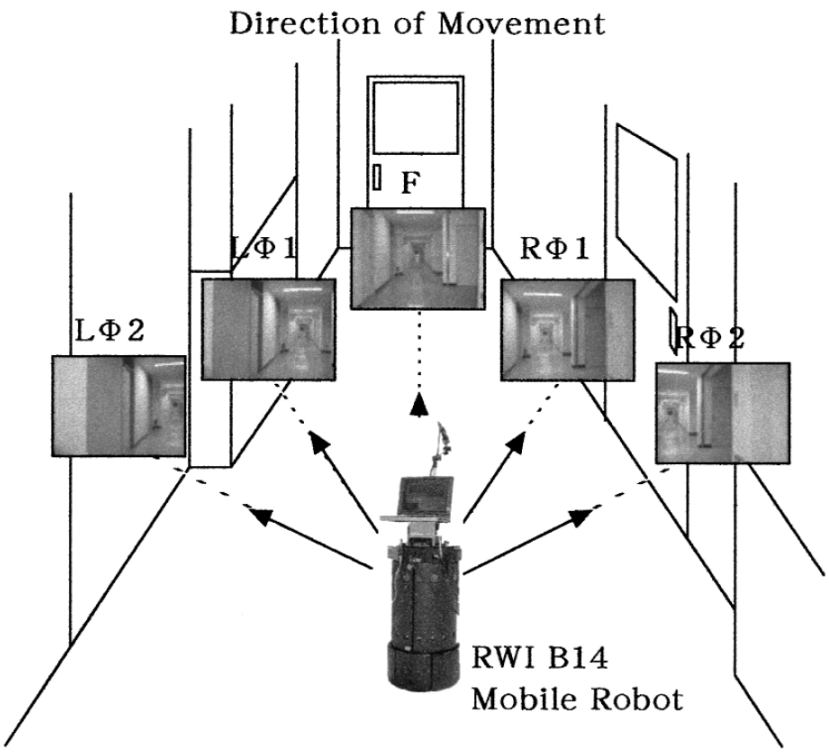


Figure 3.1. Method to take sequential view images from the robot.

environment. To address this deficiency, we equipped the robot with a pivoting mechanism, as shown in Fig. 3.1, which allows it to shift its viewpoint to the right or left of its direction of travel, which is taken as a standard. The set of images resulting from taking the view in these directions forms a series which we will refer to as a viewing image sequence.

The robot used in this research (a B14 mobile robot manufacture by the RWI Company) was equipped with a CCD camera having a 256 step gray scale and a resolution of 320 $\times$ 240 pixels. In addition, after careful consideration of the limitations imposed by the hardware build into the robot and by the processing time required, we used image compression to reduce the size of the original images obtained from the CCD camera.

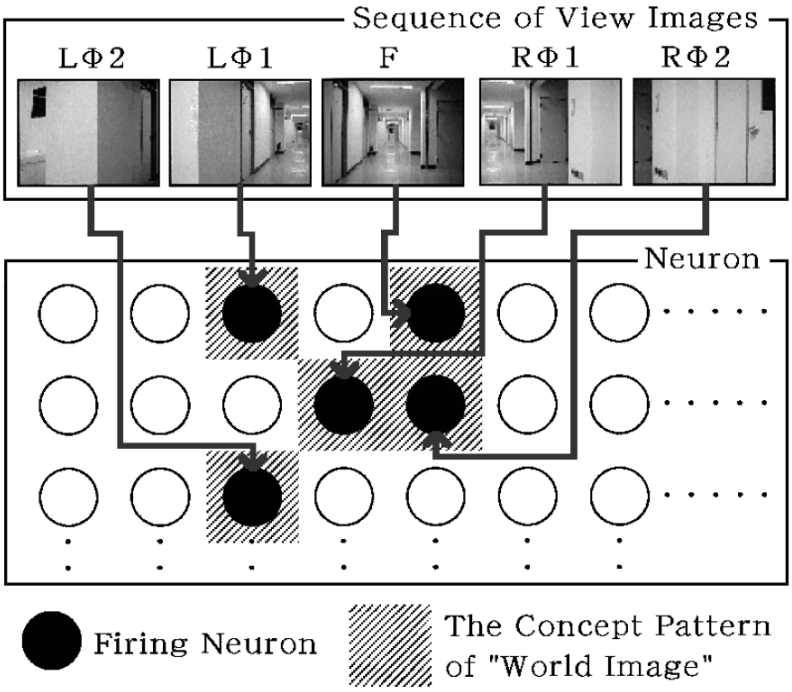


Figure 3.2. Concept patterns of world image maps.

3.2.2 Concept pattern for the landscape

Although the human brain possesses no knowledge of any kind when it is first created, it acquires various forms of knowledge while undergoing the experiences of the growth process. This knowledge is generated by recalling real-world experiences from archetypes in memory, which make up what we refer to as world image maps [2, 70]. Human use world image maps to progress intellectually in various ways. In the same way, if a robot could be equipped with such world image maps, it could acquire knowledge in a self-organizing way by sensing its surroundings while moving in a purposeful and at the same time functional manner. We will discuss this in what follows.

In our scheme, the viewing image sequence shows landscape changes, which causes the world image maps created within the robot to depend on position information. This allows a self-position estimation to be made. As shown in Fig. 3.2, in our method the viewing image sequence reflects different neuron firings that correspond to the topological characteristics of a scene. Each of these firing

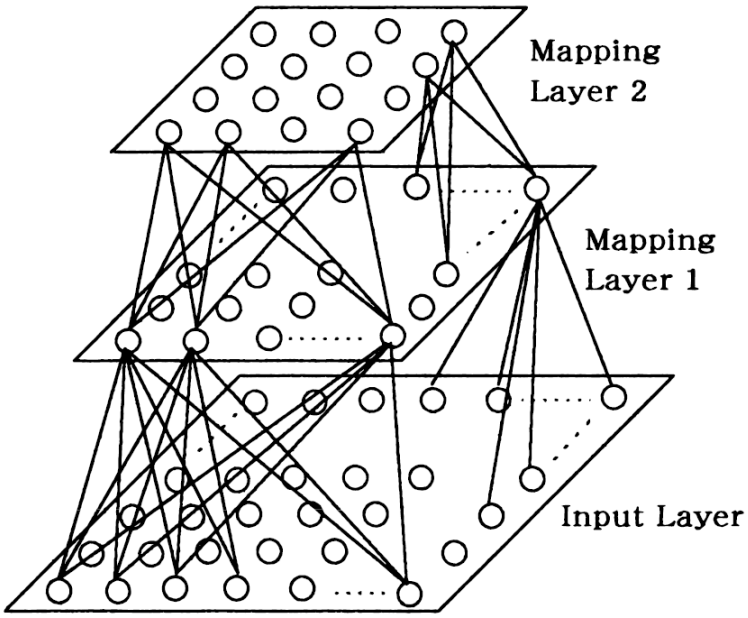


Figure 3.3. Hierarchical Self-Organizing Maps (HSOMs).

neurons generates a concept of the landscape at some position, and the firing distribution of neurons as a whole defines the concept patterns [70].

3.2.3 Hierarchical structure of SOMs

There are distinctive features in the way the topological mapping that makes up SOMs store the topology of the input data. Accordingly, we can map the position information by using the landscape changes that the viewing image sequence encodes as input to SOMs. Since learning by SOMs require no teaching signals, the robot can acquire positional information in a self-organizing way as it moves along its path.

Ordinarily, the network of SOMs comprises two layers, the input layer and the mapping layer. In our method, concept patterns related to the viewing image sequence are generated by units in the SOM mapping layer. In order to consolidate these concept patters, the SOM is made hierarchical. We made our hierarchical SOM [71] by using an input layer and two mapping layers, similar to the network structure shown in Fig. 3.3.

Starting with the layer at the lowest position in the figure, a previously com-

pressed image at one layer becomes the input for the next layer up. Thus, the middle layer (which we will refer to as the first mapping layer in what follows) becomes the input with respect to the mapping layer. In the middle layer, concept patterns are generated that related to the input images. In the top layer (which we refer to as the second mapping layer in what follows), connections are made among the concept patterns generated in the first mapping layer so that only one neuron will fire for each concept pattern generated.

During the learning period and prior to the trial journey, any of the units of the second mapping layer that fired were identified and labels were attached to them. In the course of the trail journey, position estimations are obtained by determining which positions correspond to the labels of the units that have fired, i.e., the labels attached during the learning period. During its trial journey, the robot is in the neighborhood of various labeled positions. A position estimation is defined to be successful if as the robot pauses at one of these labeled positions of the first neighborhood units with that position's label attached during learning fire. We define the position estimation rate as the number of successful position estimation cycles divided by the total number of position estimation cycles performed.

3.3. Parameters for the viewing image sequence

In order to determine parameters that can be used to match the viewing image sequence with the robot's viewpoint shifts, we performed the experiments described in the three parameters below.

3.3.1 Downsampling levels

Careful consideration of the real-time nature of the position estimate leads us to conclude that image downsampling is unavoidable. However, the size of the downsampling grid has a strong influence on how much the volume of data, the noise component, and the number of distinctive features, e.g., edge components, etc., can be reduced in the compressed image. Therefore, we carried out experiments to find the relation between the size of the downsampling grid and the position estimate rate.

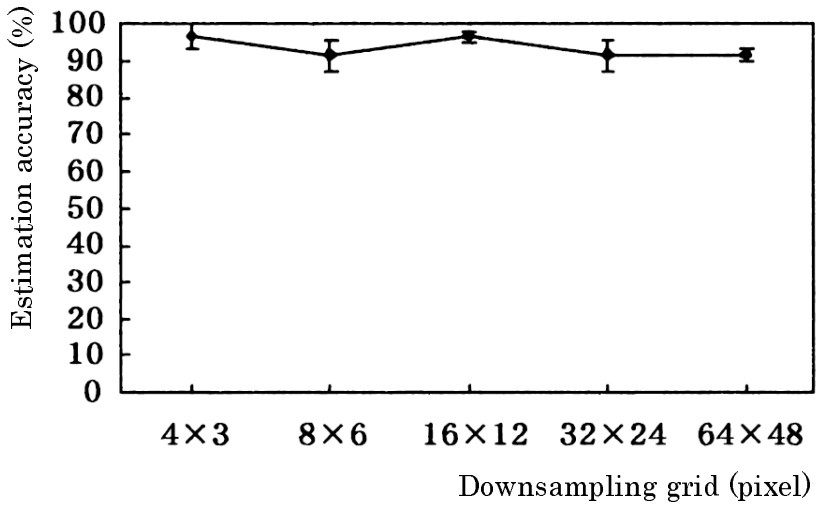


Figure 3.4. Relations between position estimation rates and downsampling levels.

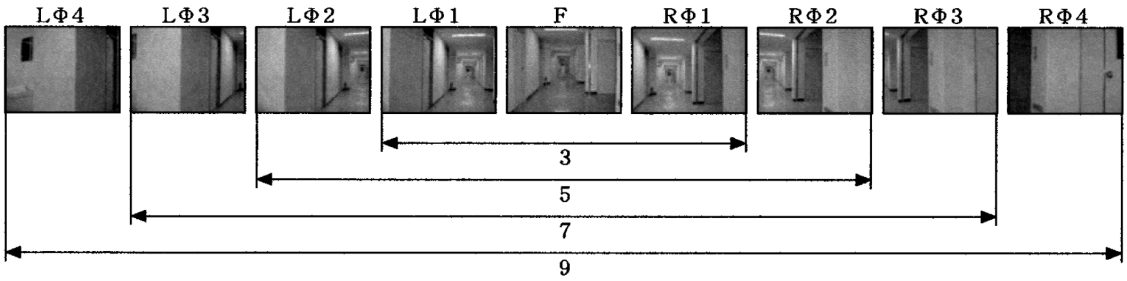


Figure 3.5. The number of viewpoint movements.

Figure 3.4 shows how the position estimate rate changes as the size of the downsampling grid is varied from 4 pixels vertical × 3 pixels horizontal (Level 1) to 64 pixels vertical × 48 pixels horizontal (Level 5). The figure indicates that position estimation rates of over 90 percent are obtained for all the downsampling levels, and that those with the highest rates are Levels 1 and 3. Accordingly, in this research, we used the parameters for the downsampling Level 3 in order to ensure the highest processing speed.

3.3.2 Number of view point shifts

Multiple landscape views of the environment are generated by using the robot's pivoting mechanism to shift its viewpoint. Increasing the number of viewpoint

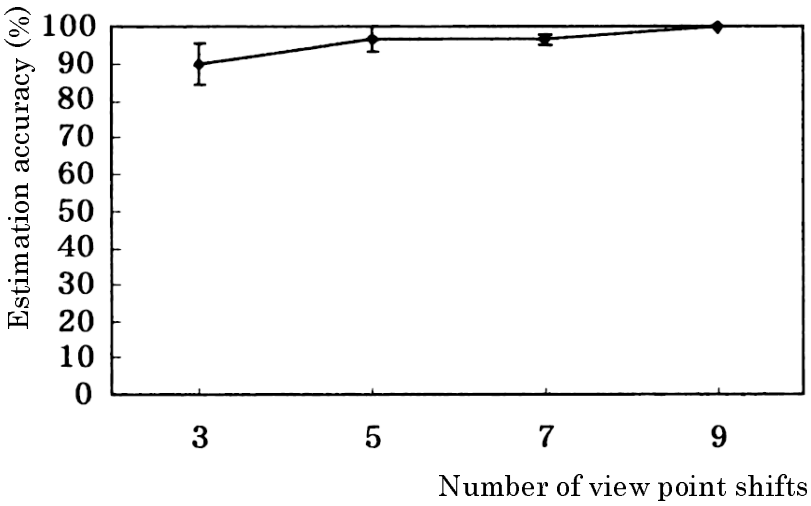


Figure 3.6. Relations between position estimation rates and the number of viewpoint movements.

shifts in the manner shown in Fig. 3.5 allows the robot to obtain more position-related information, because it is capable of taking in a broader ranging scene. However, various problems develop as the amount of data becomes large, among them increased time required to process the full range. In contrast, when the number of viewpoint shifts is small, the information needed for self-position estimation cannot be obtained. For this reason, we carried out experiments to determine the relation between number of viewpoint shifts and the position estimation rate.

Figure 3.6 shows a plot of the position estimation rate versus the number of viewpoint shifts, which was varies from 3 directions to 9 directions. According to the figure, an increase in the number of viewpoint shifts is accompanied by an improvement in the position estimation rate, which is easy to understand. Hence, after careful consideration of the real-time nature of the position estimate, we conclude that it is desirable to make position estimates based on a smaller number of viewpoint shifts. However, with 3 directions clearly the position estimation rate is low and the scatter is large. Conversely, above 5 directions the difference in position estimation rates is almost imperceptible. Thus, in this research the value we chose for the number of viewpoint shifts was 5 directions, for which the position estimation rate is high and at the same time the number of viewpoint

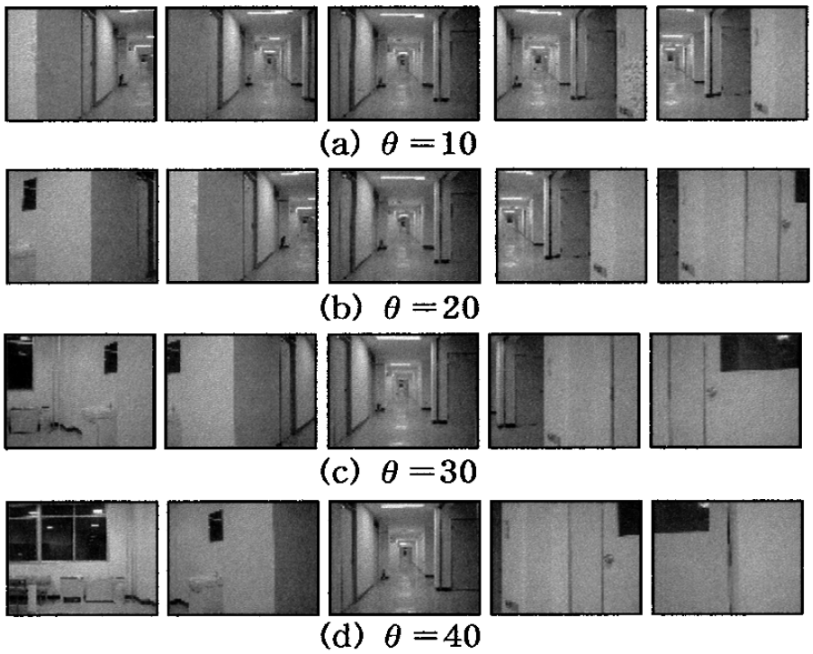


Figure 3.7. Angles of viewpoint movements.

shifts is also as small as possible.

3.3.3 Viewpoint shift angles

Just as it does with the number of viewpoint shifts, the landscape view of the environment varies with the robot's viewpoint shift angle. As shown in Fig. 3.7, when the viewpoint shift angle is taken to be small, and the robot surveys a narrow scene, the overlap between viewpoints becomes large. Likewise, when the viewpoint shift angle is taken to be large, and the scene surveyed is broad, the overlap between viewpoints becomes small. Hence, in order to guide us in what range of scene surveyed and overlap between viewpoints are necessary in estimating position, we carried out experiments to determine the relation between the viewpoint shift angle and the position estimation rate.

Figure 3.8 shows a plot of the position estimation rate versus the viewpoint shift angle, which was varied from 10 deg to 40 deg. According to the figure, although the position estimation rate does not appear to vary strongly with the viewpoint shift angle, when the angle = 10 deg, for which the most stable results

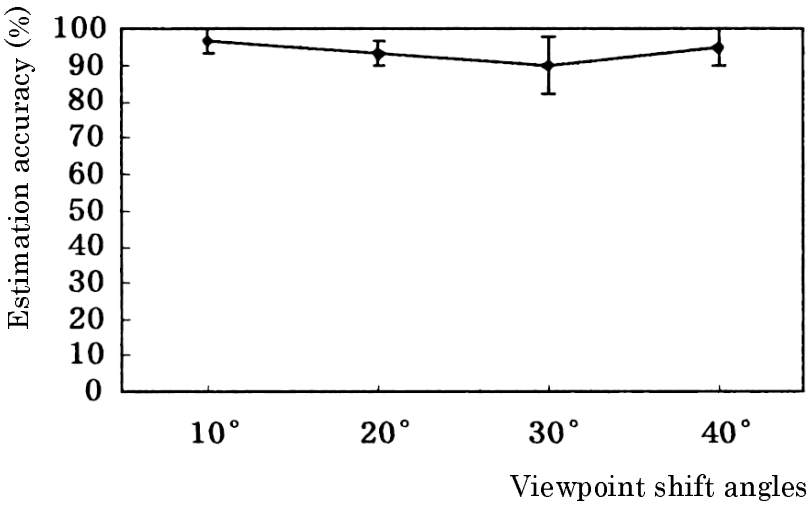


Figure 3.8. Relations between position estimation rates and the angles of viewpoint movements.

were obtained.

3.4. Position estimation experiments

In order to ascertain the effectiveness of our method, we conducted position estimation experiments in a corridor and a lobby. We first took 5 circuits of the surroundings, from which the robot could generate viewing image sequences in the learning phase and use them in the testing phase, respectively. The parameters

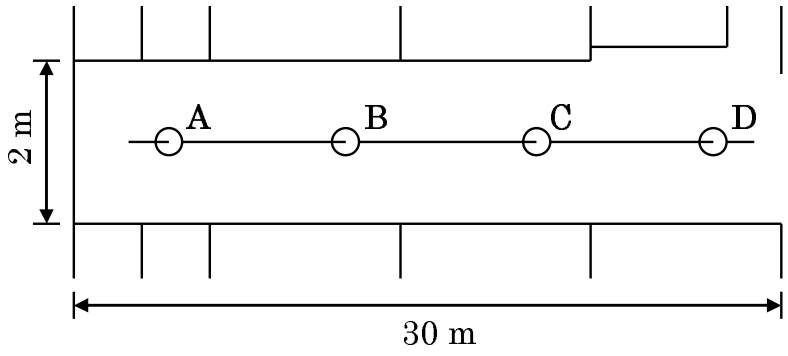


Figure 3.9. Experimental environment in corridor.

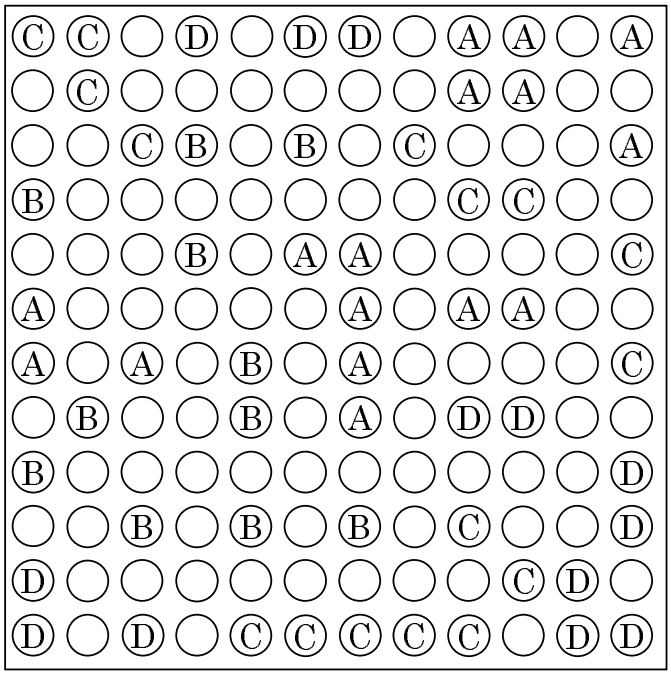


Figure 3.10. Concept patterns on the first mapping layer in corridor.

of the viewing image sequences were those specified in the preceding paragraphs: the value of the compression was Level 3 (16 pixels $\times$ 12 pixels); the number of viewpoint shift was 5 directions; and the viewpoint angle was10 deg.

3.4.1 Position estimation in a corridor

In this experiment, we set the robot travel down the center of the corridor shown in Fig. 3.9 from Position A to Position D. Viewing image sequences were taken and position estimates were made at each position. The circles in the figure indicate the position of the robot, while the lines from the centers of the circles indicate the robot's direction of progress.

Figure 3.10 shows the neuron elements in the first mapping layer of the hierarchical SOM that were caused to fire during the learning phase. The figure indicates that concept patterns, defined as neuron firing distributions, were generated at every position. There was no overlap between the various concept patterns, and the layer was partitioned into regions corresponding to the various

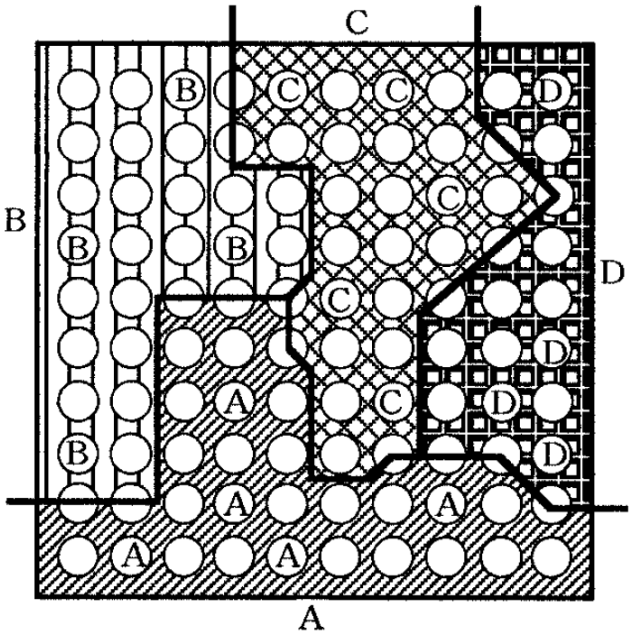


Figure 3.11. Mapping results on the second mapping layer in corridor.

positions.

Figure 3.11 shows the neuron elements in the second mapping layer of the hierarchical SOM that were caused to fire. In this figure we have labeled the neuronal units that fired during the learning phase with the position indicators from A to D. The first neighborhood neurons are grouped together and treated as same units, i.e., the same position label is attached to all eight units. These results verify that the mapping consolidates the concept patterns generated during the learning phase, so that a characteristic cluster of neurons forms for each position.

As position estimates were made during the trial, firings occurred in the neighborhood units with labels attached during the learning phase, and effective results were obtained for position estimation when these firings were grouped together. The position estimation accuracy in the corridor was 95 percent.

3.4.2 Position estimation in a lobby

In this experiment, we made the robot travel counterclockwise along the walls of the lobby shown in Fig. 3.12. Viewing image sequences were taken and position

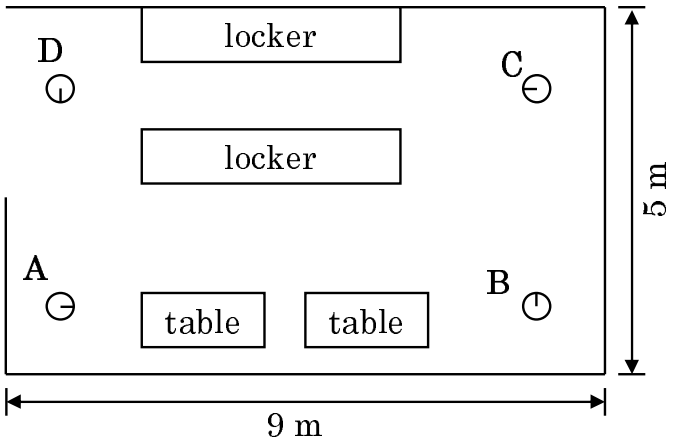


Figure 3.12. Experimental environment in lobby.

estimates were made at the four positions where cornering took place (Positions A–D). The resulting concept patterns for the landscapes generated in the learning phase are shown in Fig. 3.13, while Fig. 3.14 shows the results of consolidating the concept patterns. In this experiment, as in the experiment in the corridor, a concept pattern was formed at each position during the learning phase. The respective concept patterns did not overlap, and the layer was partitioned into regions corresponding to the various positions. Then units with labels attached at the learning time were separated into consolidated groups corresponding to their respective positions, so that a characteristic cluster formed for each position.

As position estimations were made during the trial moving, firings occurred in the vicinity of units with labels attached during the learning phase, and effective results were obtained for position estimates when these firings were grouped together. The position estimation accuracy in the lobby was 100 percent., i.e., higher in the lobby than in the corridor. In our opinion, this was because the viewing image sequences encoded large changes in the landscape as the robot traveled from position to position in the lobby, so that the hierarchical SOM could precisely identify the distinctive features of a scene.

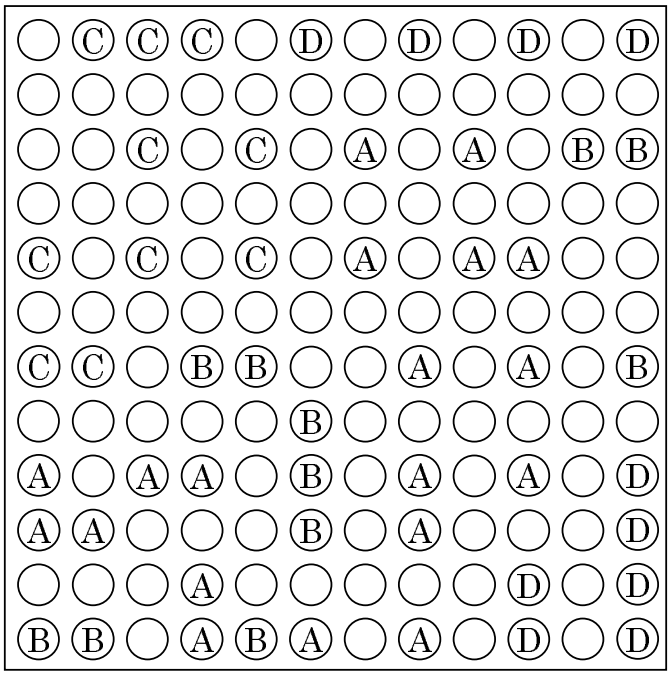


Figure 3.13. Concept patterns on the first mapping layer in lobby.

Table 3.1. Relation between estimation accuracies and shifted direction.

	Original	Right–1	Right–2	Left–1	Left–2
Corridor (%)	95	92	83	92	83
Lobby (%)	100	92	92	100	100

3.5. Evaluation of position estimation

3.5.1 Shifts in the robot's direction of travel

Ordinarily it is a challenging task for a robot to maintain the same direction of travel when it is moving autonomously, due to slips, shakes, etc. that interfere with its progress. For this reason, we carried out the experiment shown in Fig. 3.15, whose purpose was to evaluate the effect of direction shifts caused by stopping for position estimation. In this experiment, the robot's direction of travel was shifted stepwise to the right or to the left by 10 deg (Right–1, Left–1) and

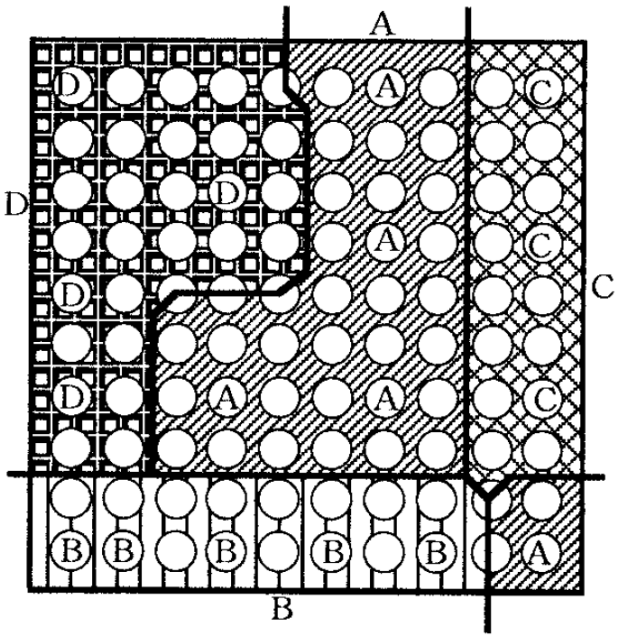


Figure 3.14. Mapping results on the second mapping layer in lobby.

by 20 deg (Right–2, Left–2).

Table 3.1 shows the relation between the position estimation accuracy and these direction shifts. According to the table, the position estimation accuracy did not appear to decrease in the corridor experiment, except when the direction shift was 20 deg for which the position estimation accuracy became 83 percent. In our view, there was no decrease in the topological characteristics features of the partial world image maps, so that when the viewing image sequence as a whole is created the resulting changes are also small. Hence, when the direction shifts cause changes in a part of a concept, the rest of the concepts will compensate for the changes, leave the concept pattern unaffected. From this we conclude that our method, which is based on the concept patterns, is capable of robust self-position estimation.

3.5.2 Shifts in the robot's point of departure

Like the robot's direction shifts, shifts in position caused by events that interrupt its progress cause problems whenever the robot stops and starts at fixed points

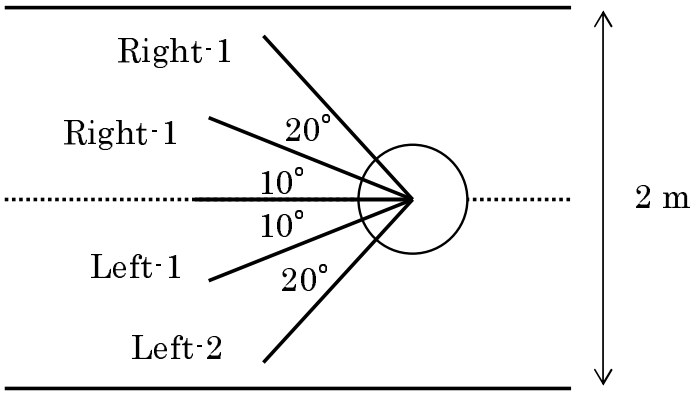


Figure 3.15. Robustness for direction.

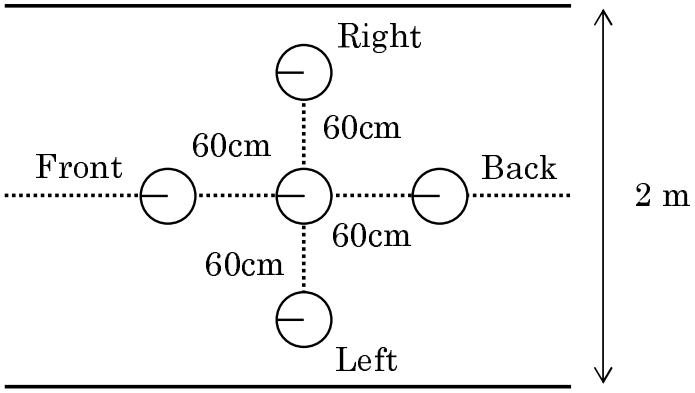


Figure 3.16. Robustness for position.

in its moving. To address this issue, we carried out the experiment shown in Fig. 3.16 to evaluate the effect of position shifts caused by stopping for position estimation. In this experiment, we shifted the robot's position forward, backward, left, and right by 60 cm from the position it occupied during the learning phase.

Table 3.2 shows the relation between the position estimation accuracy and these shifts in the departure point. According to the table, the position estimation accuracy falls below 50 percent in the corridor. We explain this as follows: because the changes in the corridor landscape recorded in the viewing image sequence were minor, and the distinctive feature identified at the various positions were similar in appearance, the resulting changes in the topological characteristics of the viewing image sequence as a whole caused by the position shifts were not

Table 3.2. Relation between estimation accuracies and shifted position.

	Original	Front	Back	Right	Left
Corridor (%)	95	42	50	25	42
Lobby (%)	100	100	92	92	83

Table 3.3. Hospitals with medical treatment sickbeds

Type	Ordinary beds	Medical treatment beds
Number of persons in room	–	4 or fewer
Sickroom area per person	At least $4.3m^2$	At least $6.4m^2$
Width of side corridor	At least $1.2m^2$ (interior)	At least $1.8m^2$ (interior)
Width of central corridor	At least $1.6m^2$ (interior)	At least $2.7m^2$ (interior)

sufficient for a correct position estimation. In contrast, there was no perceptible decrease in the overall position estimation accuracy in the lobby caused by the position shifts except for the small decrease of 83 percent in the position estimation accuracy for the left-hand shift. In this case, we claim that the changes in the corridor landscape recorded in the viewing image sequence were substantial, and the distinctive features identified at the various positions were different in appearance. This leads us to conclude that our method of self-position estimation is not robust against position shifts.

3.6. Evaluation testing in a clinical environment

3.6.1 Experimental environment

We had studying the use of a patrol robot system for service providing facilities under long term care insurance, so called convalescent wards in a general hospital. Among these facilities, we selected the clinical facilities of Sotoasahikawa General Hospital in Akita city as the experimental environment [72]. As shown in Table

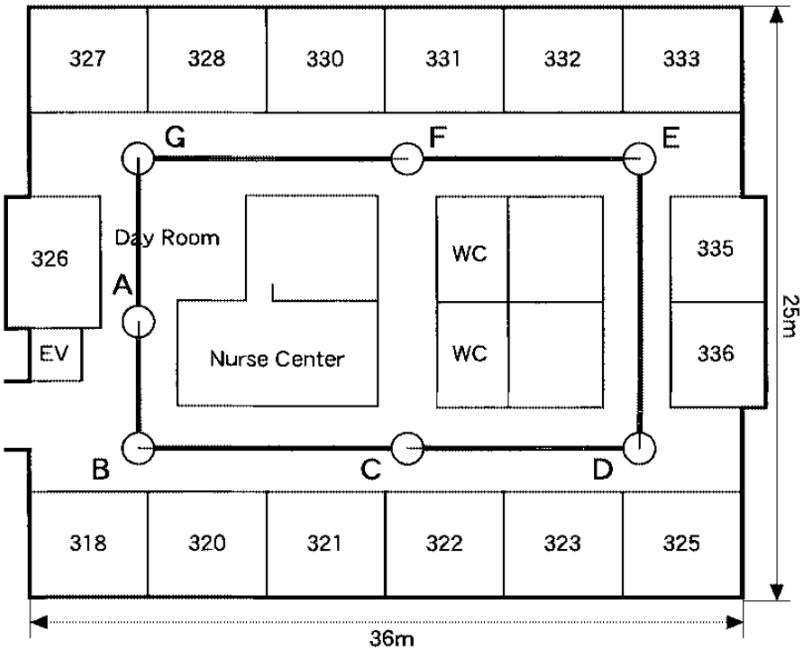


Figure 3.17. Experimental environment of global positions at hospital.

3.3, since these facilities the sickroom area per patient is greater than average (at least $4.3m^2$ for ordinary sickbeds and at least $6.4m^2$ for treatment sickbeds), and the corridor width is 1.5 times that of ordinary facilities, it is an environment suitable for autonomous running by the robot. In working out a practical implementation, the functional performance of the robot and adjustment of the environment and the facilities are very important. Medical treatment sickbeds are beds for patients who chiefly require long-term hospital care. The number of such beds has been increasing rapidly: it rose by about 85,000 in fiscal year 1999 alone. In addition, since an environment suitable for long-term medical care must be provided in such facilities, the environments of the corridors and the sickrooms are well laid out and orderly.

3.6.2 Position estimation results

In order for the robots to estimate their position as they make their rounds, they need global position information (positions along their routes where the corridors split) in order to select the proper route, and local position information

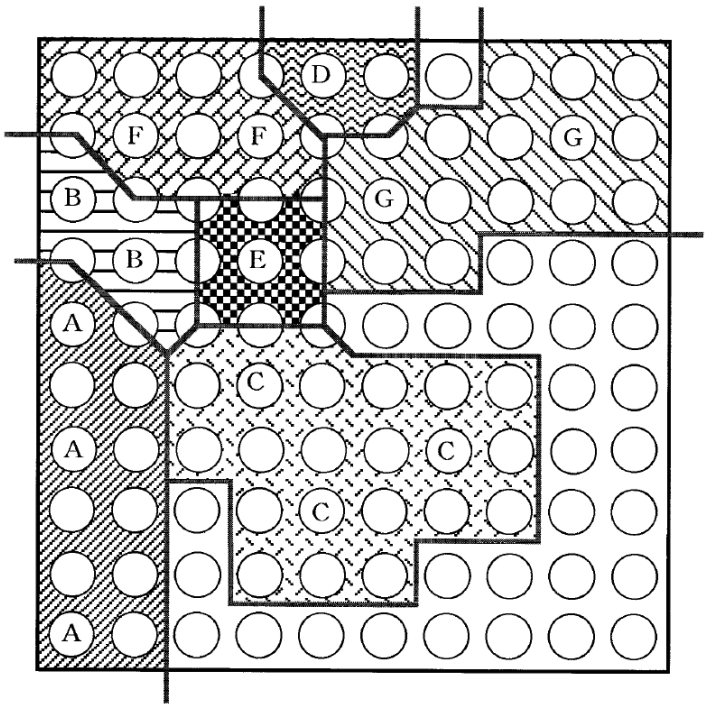


Figure 3.18. Mapping result of global positions.

(position corresponding to room numbers) in order to recognize the entrances and exits of each sick room. In this work, we equipped the robots with separate and independent hierarchical SOMs for global position estimation and local position estimation, because of large differences in the information collected, i.e., the topological distinctive features revealed by landscapes of the viewing image sequences, for the two types of estimates. In the experiment, world image maps were constructed in the learning phase using position information from viewing image sequences gathered by making the robots conduct three rounds on the facilities floor. After these rounds a trial journey was executed.

Seven global positions were specified, as shown in Fig. 3.17: Position A just before the nurse's station, Positions B, D, E, G at each of the floor's corners, and at the forks (Positions C, F). Likewise, a total of six local positions were specified, as shown in Fig. 3.19, from Position A in front of room number 333 to Position F in front of room number 327. Figs. 3.18 and 3.20 show the results of various position estimation experiments. In the position estimations during

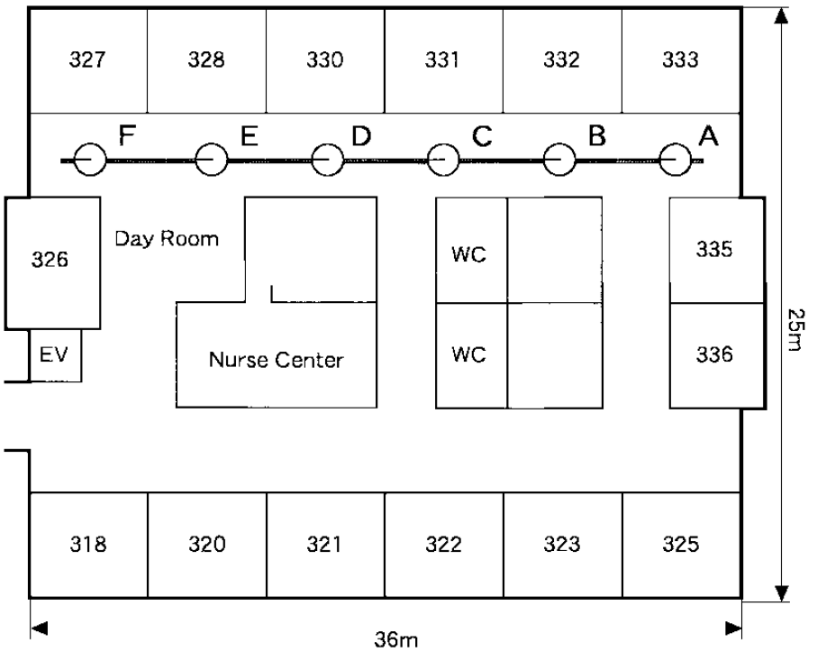


Figure 3.19. Experimental environment of local positions at hospital.

the trial journey, firings occurred among neurons in clusters with common labels attached in the course of constructing the world image maps during the learning phase, and correct position estimates were made, regardless of which position was in question.

The position estimation results shown in Figs. 3.18 and 3.20 reveal that the world image maps acquired during learning exhibits clustering, and that the clusters differ in size from one position to the next. We argue that those positions that gave rise to large extended clusters (Positions A, C, G in Fig. 3.18, Position F in Fig. 3.20) were positions where there was frequent traffic by nurses and caregivers coming and going, due to the large amount of daily service they were providing. This produced large contrasts with the viewing image sequence acquired during the learning phase. In contras, at the positions where small compact clusters formed (Positions D, E in Fig 3.18, Positions B, D in Fig. 3.20) the traffic was small, leading to only slight departures from the static viewing image sequence acquired during the learning phase. That is, the concept patterns generated by our method from the topological distinctive features of a scene are capable of

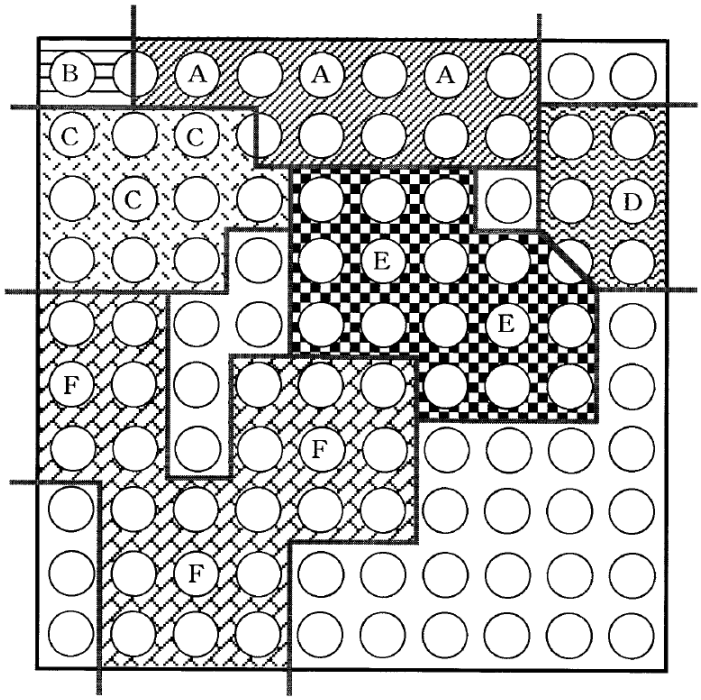


Figure 3.20. Mapping result of global positions.

discriminating whether people come and go rarely or often, indicating a further advantage of the position-specific world image maps that are created.

Thus, our method can learn dynamic changes in real surroundings (encoded as scatter in the scene's position distinctive features) in a self-organizing way. Because the method acquires position-specific world image maps, during trial journeys the robot can implement robust position estimation based on changes detected in the viewing image sequence.

Therefore, because dynamic changes (scatter of the phase characteristics of the scene) in an actual environment can be learned in a self-organizing manner and world image maps that are position-specific can be acquired, the proposed technique allows position estimation that is robust to variation of the sequence of view images during the test run.

3.7. Conclusion

In this chapter we developed a method for a robot to estimate its position from changes in landscape that accompany shifts in viewpoint. The results listed below were obtained.

- We found that changes in landscape revealed by viewing image sequences could be extracted as concept patterns by a SOM. Effective position information is acquired by making this hierarchical SOM and using it to consolidate position estimation concept patterns.
- We identified the following parameter for effectively characterizing the viewing image sequence from the standpoint of position estimation: the compression level, the number of viewpoint shifts, and the viewpoint shift angles.
- We evaluated the effect of shifts in position and direction while the robot was executing a trial journey on position estimation. The extent of these shifts established beyond a doubt that our method was robust.
- The results of an on-site field test of a robot system in a hospital with a convalescence ward confirmed the effectiveness of our method for practical use.

In the future, we plan to do experiments that will evaluate the relation between he number of neurons in the position estimation mapping layers and spatial resolution accuracy (position estimation ability) in order to extend the range of usefulness of this method.

Chapter 4

Representation of Orientation Selectivity on ART2

4.1. Introduction

People with rich facial expressions are robust to uncertain situations or adverse circumstances. The roles of facial expressions in communication among people are important and various. Especially in a close relationship, we can mutually understand the feeling and intensity from the information of facial expressions. In the field of human communication, computer recognition of facial expressions has been studied for realizing a natural and flexible Man-Machine Interface (MMI) that can interpret the feeling or intensity of users [73].

Ekman defined six basic expressions (anger, sadness, disgust, happiness, surprise, and fear) based on six kinds of basic feelings [76]. However, the number of categories to express is unknown because facial expressions exist that are invalid or which reflect several mixed feelings. In this chapter, we introduce Adaptive Resonance Theory (ART) networks [23] as a method to represent detection of dynamic, local, and topological changes of facial expressions. The ART, which was proposed by Grossberg et al., is a theoretical model of an unsupervised and self-organizing neural network to form a category adaptively in real time while maintaining stability and plasticity. Using incremental learning of ART, the method can classify facial expressions without presetting of the number of categories. In addition, facial expressions that are controlled by feelings change over

time through aging. We consider that ART, which can learn over time, is useful to deal with time-series movements of facial expressions.

However, setting the parameters of ART networks is very difficult and complex; furthermore, classification results depend strongly on settings and combinations of parameters. Especially, a parameter called the attentional vigilance parameter strongly influences classification granularity. In addition, ART networks generate inclusions or redundant categories, even though the setting of vigilance parameters is the same. In this chapter, we specifically describe orientation selectivity of Gabor wavelets for analyzing classification granularity of ART networks. The method can detect dynamic, local, and topological changes of facial expressions for category changes of ART networks. Moreover, the method can prevent redundant categories through the use of orientation selectivity.

4.2. Related studies

Akamatsu described two types of facial diversity [74]. Facial components such as eyes, eyebrows, and the mouth are different for each person. Facial features of those facial components' position, size, location, etc. are also different. This is called static diversity. On the other hand, we move facial muscles to express internal emotions unconsciously or express emotions as a message. Facial expressions are produced by the facial components and their transition from a normal facial expression. This is called dynamic diversity. Regarding facial recognition in the field of facial image processing, only the use of static diversity is sufficient to obtain good results. For facial expression recognition, it requires not only static diversity but also dynamic diversity as a time-series to cope with facial pattern transitions.

Nishiyama et al. [75] proposed facial scores, a method to describe facial expression rhythms. They pointed out that facial expressions that are describable with a Facial Action Coding System (FACS) by Ekman [76] are only static features. Therefore, they did not use Action Units (AUs) of FACS. They originally used setting feature points because FACS can not describe time-series transitions of facial expressions. On the other hand, humans can recognize facial expressions to detect movements of local facial components from entire structures of faces.

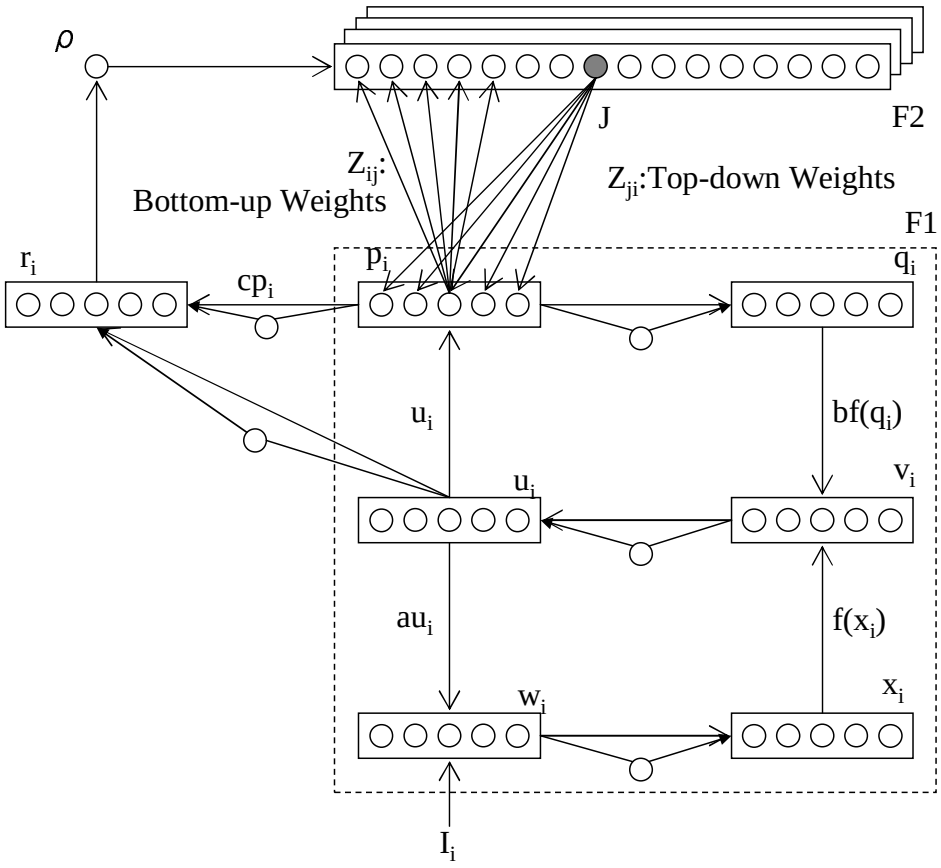


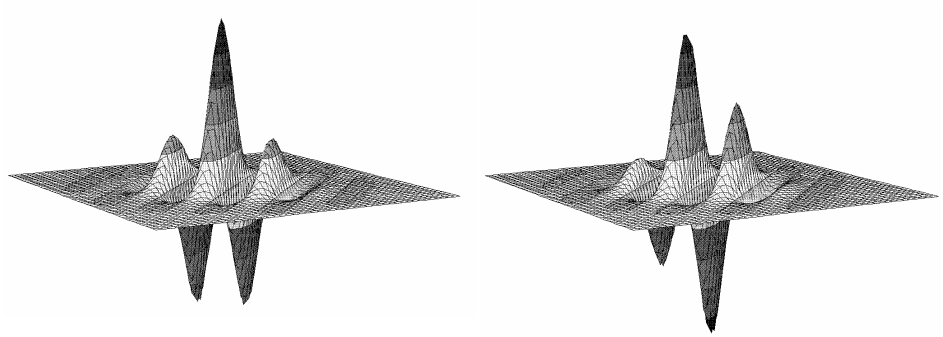
Figure 4.1. Architecture of an ART2 network.

We do not need to detect facial elements as movements of characteristic points. We can automatically detect dynamic, local, and topological feature changes of facial expressions from whole facial changes.

4.3. Adaptive Resonance Theory 2

Actually, ART has many variations: ART1, ART1.5, ART2, ART2-A, ART3, ARTMAP, Fuzzy ART, Fuzzy ARTMAP, etc. [20]. We use ART2 [23], into which analog values can be input.

Figure 4.1 shows the ART network architecture. The network consists of two fields: Field 1 (F1) for feature representation and Field 2 (F2) for category



(a) Real Part (b) Imaginary Part

Figure 4.2. Three-dimensional representations of Gabor wavelet filters.

representation. The F1 consist of six sub-layers: p_i, q_i, u_i, v_i, w_i, and x_i. These sub-layers realize Short Term Memory (STM), which enhances features of input data and detects noise for a filter. The F2 realizes Long Term Memory (LTM) based on finer or coarser recognition categories. The algorithm of ART2 is the following.

1. The top-down weights Z_{ji} and bottom-up weights Z_{ij} are initialized as

$$Z_{ji}(0) = 0, \quad Z_{ij}(0) = \frac{1}{(1-d)\sqrt{M}}. \tag{4.1}$$

2. The sub-layers of F1 are initialized as

$$p_i(t) = q_i(t) = u_i(t) = v_i(t) = w_i(t) = x_i(t) = 0. \tag{4.2}$$

3. The input data I_i are presented to the F1. The sub-layers are propagated as

$$w_i(t) = I_i(t) + au_i(t-1), \tag{4.3}$$

$$x_i(t) = \frac{w_i(t)}{e + ||w||}, \tag{4.4}$$

$$v_i(t) = f(x_i(t)) + bf(q_i(t-1)), \tag{4.5}$$

$$u_i(t) = \frac{v_i(t)}{e + ||v||}, \tag{4.6}$$

$$q_i(t) = \frac{p_i(t)}{e + ||p||}, \tag{4.7}$$

$$p_i(t) = \begin{cases} u_i(t) & (inactive) \\ u_i(t) + dZ_{Ji}(t) & (active), \end{cases} \tag{4.8}$$

where

$$f(x) = \begin{cases} 0 & if \quad 0 \leq x < \theta, \\ x & if \quad x \geq \theta. \end{cases} \tag{4.9}$$

4. Search for the maximum active unit T_J as

$$T_j(t) = \sum_j p_i(t) Z_{ij}(t), \tag{4.10}$$

$$T_J(t) = max(T_j(t)). \tag{4.11}$$

5. The weights Z_{ji} and Z_{ij} are updated as follows.

$$\frac{d}{dt} Z_{Ji}(t) = d[p_i(t) - Z_{Ji}(t)] \tag{4.12}$$

$$\frac{d}{dt} Z_{iJ}(t) = d[p_i(t) - Z_{iJ}(t)] \tag{4.13}$$

6. The output value of $r_i(t)$ is calculated as

$$r_i(t) = \frac{u_i(t) + cp_i(t)}{e + ||u|| + ||cp||}. \tag{4.14}$$

The reset is defined as

$$\frac{\rho}{e + ||r||} > 1. \tag{4.15}$$

7. If eq. (4.15) is true, the active unit is reset; go back to 4) to search again. If no active unit exists, a new category is created; return to 3). If eq. (4.15) is not true, repeat 3) and 5) until the changing of F1 is sufficiently small, then return to 2).

Parameters are the following: a and b are coefficients on feedback loops from u_i to w_i and from q_i to v_i; c is a coefficient from p_i to r_i; d is a learning rate; $cd/(1-d) \leq 1$ is the constraint between them; and θ is a parameter to control a noise-detection level in layer v.

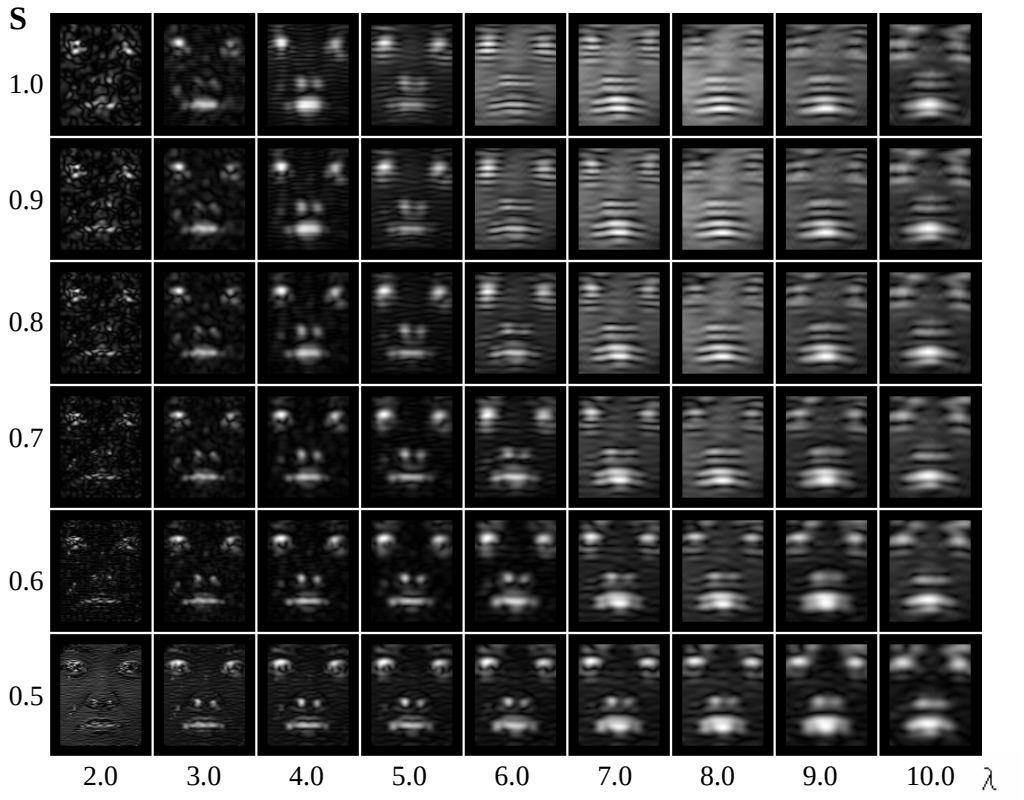


Figure 4.3. Gabor wavelet output images of the combination of λ and S.

4.4. Gabor wavelets

Visual information captured by the retina is conveyed to Visual area 1 (V1) in the occipital lobe via the Lateral Geniculate Nucleus (LGN). The V1 consists of two visual cells: simple cells and complex cells. The LGN and simple cells have receptive fields. Receptive fields respond to a particular stimulus of figures such as the size, length, direction, movement direction, color, and frequency. This is called response selectivity. Since the time Hubel and Wiesel [78] discovered orientation selectivity on receptive fields from their electrophysiological experiment using anesthetized cats, orientation selectivity has become the most well known among response selectivity.

Various methods based on visual cortex information processing models have been proposed to develop image processing or computer vision systems [74, 77,

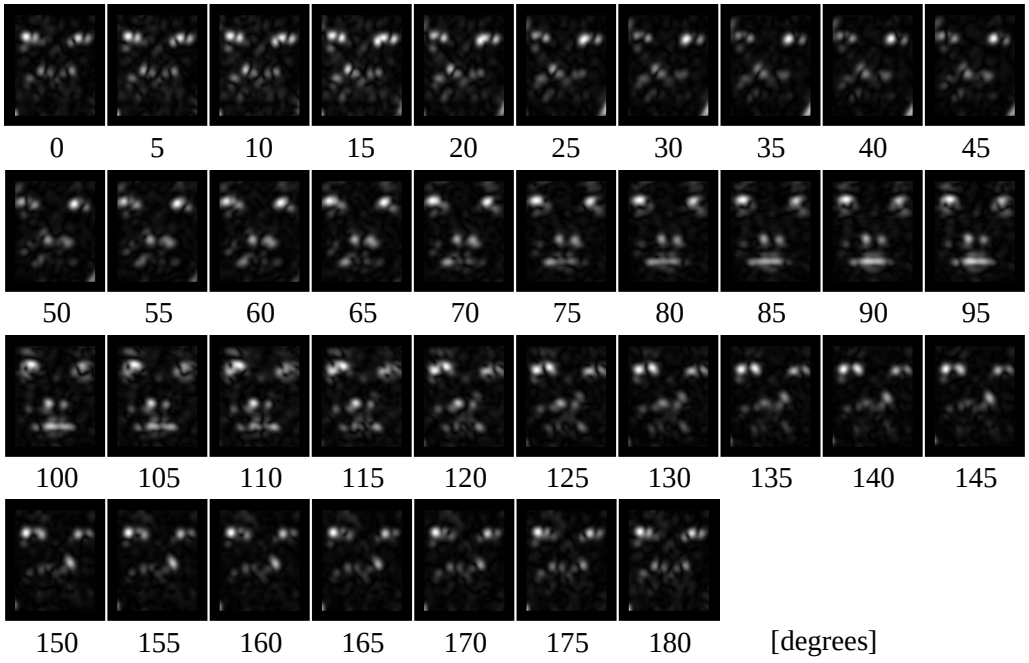


Figure 4.4. Gabor wavelet output images of θ ($0 \leq \theta \leq 180$, $\lambda = 4.0$ and $S = 0.7$)

79]. The representation of Gabor wavelets, which can emphasize an arbitrary characteristic with inner parameters, is closed to receptive fields. Therefore, Gabor wavelets are applied to various fields such as character recognition, texture classification, and facial image processing [84, 85]. Gabor wavelets are functions that are combined with a plane wave propagating to one direction and a Gaussian wave. A three-dimensional (3D) representation of Gabor wavelets is shown in Fig. 4.2.

Let λ be a wavelength, and let σ_x and σ_y respectively denote widths of horizontal and vertical directions of Gaussian windows, where θ is the angle between the direction of a plane wave and the horizontal axis. The output of Gabor wavelets $G(x, y)$ is given as

$$G(x,y) = \exp\{-\frac{1}{2}(\frac{R_x^2}{\sigma_x^2} + \frac{R_y^2}{\sigma_y^2})\} \exp(i\frac{2\pi R_x}{\lambda}), \tag{4.16}$$

where

$$\begin{bmatrix} R_x \\ R_y \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}. \tag{4.17}$$

Table 4.1. Target frames that portray facial expressions.

Facial expressions	1st	2nd	3rd
Anger	18-30	50-57	76-82
Sadness	11-24	40-48	65-78
Disgust	15-32	52-65	90-100
Happiness	21-47	64-78	-
Surprise	16-26	52-60	81-92
Fear	16-34	56-63	85-98

When Euler's formula,

$$\exp(i\theta) = \cos\theta + i\sin\theta, \tag{4.18}$$

is applied, the formula (4.16) is changed as:

$$G(x,y) = R_m(x,y) + iI_m(x,y), \tag{4.19}$$

$$R_m(x,y) = \exp\{-\frac{1}{2}(\frac{R_x^2}{\sigma_x^2} + \frac{R_y^2}{\sigma_y^2})\}cos(\frac{2\pi R_x}{\lambda}), \tag{4.20}$$

$$I_m(x,y) = \exp\{-\frac{1}{2}(\frac{R_x^2}{\sigma_x^2} + \frac{R_y^2}{\sigma_y^2})\}sin(\frac{2\pi R_x}{\lambda}). \tag{4.21}$$

The final output is

$$G(x,y) = \sqrt{Rm^2(x,y) + Im^2(x,y)}. \tag{4.22}$$

The suitable values of σ_x, σ_y are reported as a function of [81], so that

$$\begin{bmatrix}\sigma_x \\ \sigma_y\end{bmatrix} = \lambda \begin{bmatrix}S_x \\ S_y\end{bmatrix}, \tag{4.23}$$

where S_x and S_y are coefficients.

4.5. Experimental results

The classification and recognition of facial expressions that are associated with dynamic variety are required to detect local and topological changes of facial

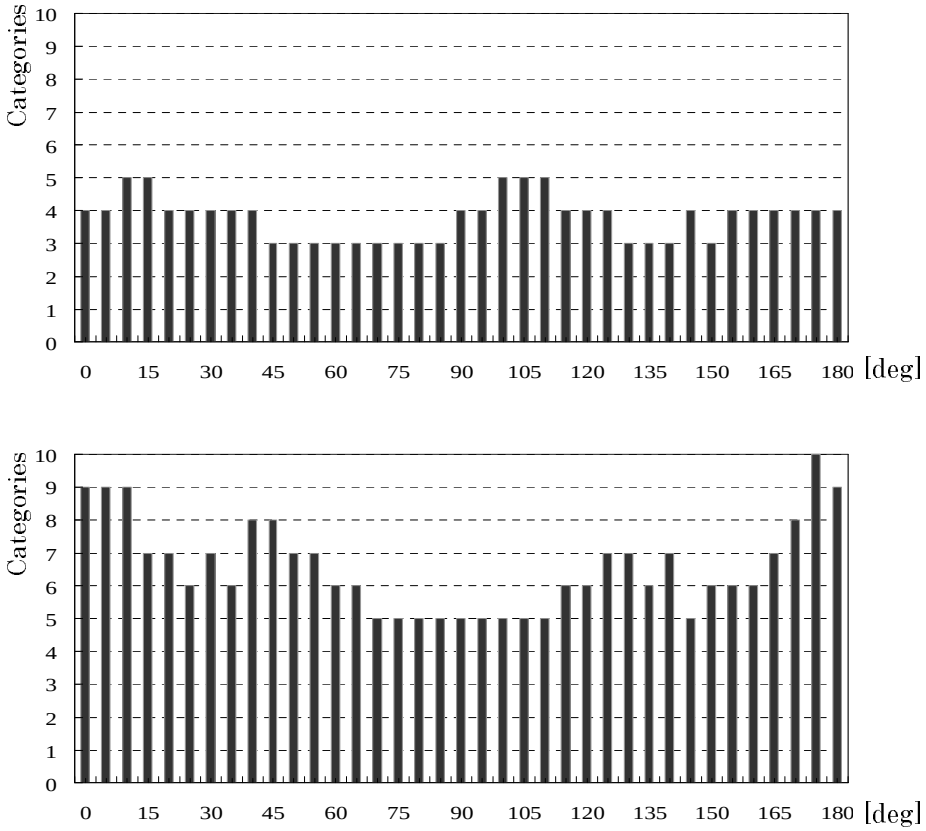


Figure 4.5. The number of categories in anger (upper) and sadness (lower) through 0 – 180 deg by 5 deg steps ($\rho = 0.970$).

components such as eyes, eyebrows, and the mouth, from global changes of overall facial patterns. The purpose of this experiment is to detect facial expression changes for the category changes of ART networks from datasets that include both expressive and normal faces. Moreover, we evaluate orientation selectivity from categorical changes of ART.

4.5.1 Target images

For this experiment, our evaluation targets are six basic facial expressions (anger, sadness, disgust, happiness, surprise, and fear) as defined by Ekman. We took 600 facial images from each person. The frame rate was 10 frames per second.

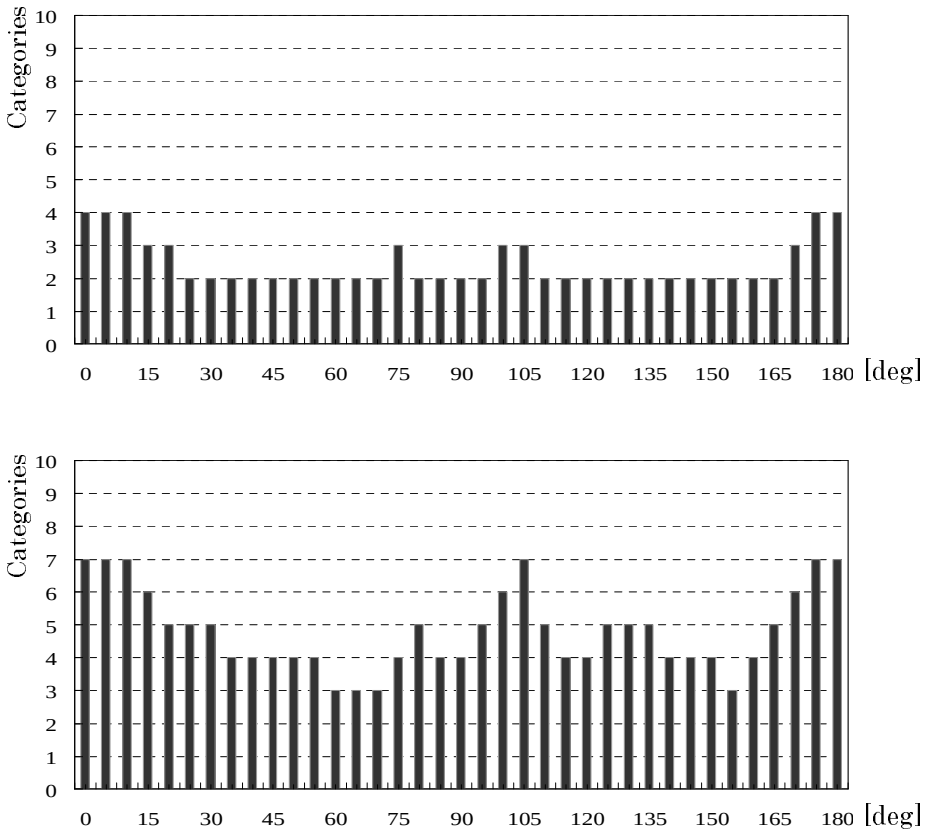


Figure 4.6. The number of categories in disgust (upper) and happiness (lower) through 0 – 180 deg by 5 deg steps ($\rho = 0.970$).

Each facial expression comprises 100 images. The images at W320 × H240 pixels resolution were taken using a CCD camera in front of the face. We manually clipped the facial region of W92 × H110 pixels from the images. For automatic facial detection, we plan to use a method using Haar-like features by Papageorgiou et al. [82].

The targeted person is a woman in her 20s, a university graduate student. She repeated one expressed face and a normal face in each facial expression. Therefore, this image dataset consists of two face types in each facial expression: one type of expression face and a normal face. The facial expressions are intentional. However, the timing of expression is idiosyncratic: the targeted person decided that timing. After the dataset acquisition, we specified appearance

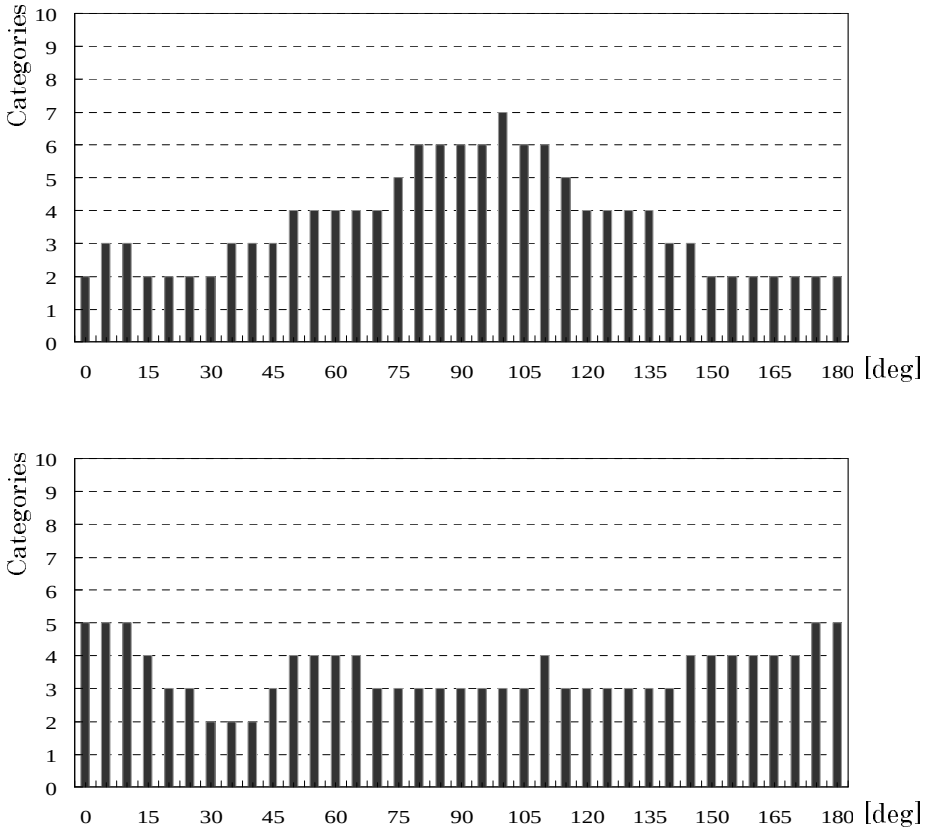


Figure 4.7. The number of categories in surprise (upper) and fear (lower) through 0 – 180 deg by 5 deg steps ($\rho = 0.970$).

and disappearance points of all facial expression datasets. The appearance and disappearance points are shown in Table 4.1.

4.5.2 Parameters

We evaluated parameters of Gabor wavelets and ART2 networks. Figure 4.3 shows the relationship between λ and S in the case of $\theta = 0$. The respective ranges of λ and S are $2.0 \leq \lambda \leq 10.0$ and $0.5 \leq S \leq 1.0$. We selected $\lambda = 4.0$ and $S = 0.7$ $(\sigma = 2.8)$ because these representations are sparse features. In this case, we set the parameters subjectively. Optimizing parameters using automatic and objective setting methods is a subject for our future work [83].

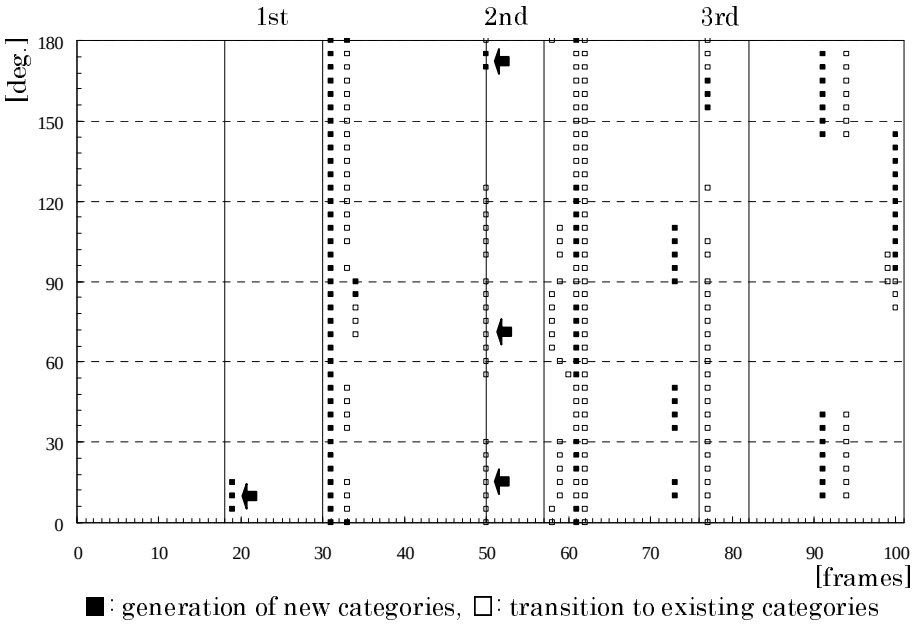


Figure 4.8. Categorical changes in anger ($\rho = 0.97$).

The electrophysiological knowledge indicates that the visual range of receptive fields is 1–5 degrees to yield a response to an input stimulation, the parameter θ is set in each case to five degrees. Figure 4.4 shows a two-dimensional representation of Gabor wavelets from 0 to 180 degrees by 5 degree steps.

We set the parameters of ART2 networks, $\theta = 0.01$, $a = b = 10$, $c = 0.225$, $d = 0.8$, $e = 0.0001$, based on our experience and the Grossberg's original paper [23].

4.5.3 Results and discussion

Figures 4.5 — 4.7 shows the number of categories in each direction from 0 to 180 degrees step by 5 degree steps. The vigilance parameter ρ is set to 0.970. The directions with a large number of categories mean that facial feature changes are large in the input images. The number of categories of surprise, which features the open level of a mouth at 90 degrees and nearby, is larger than for the other directions. This result means that the category changes of these directions are remarkable. The number of categories of sadness, which features wrinkles between

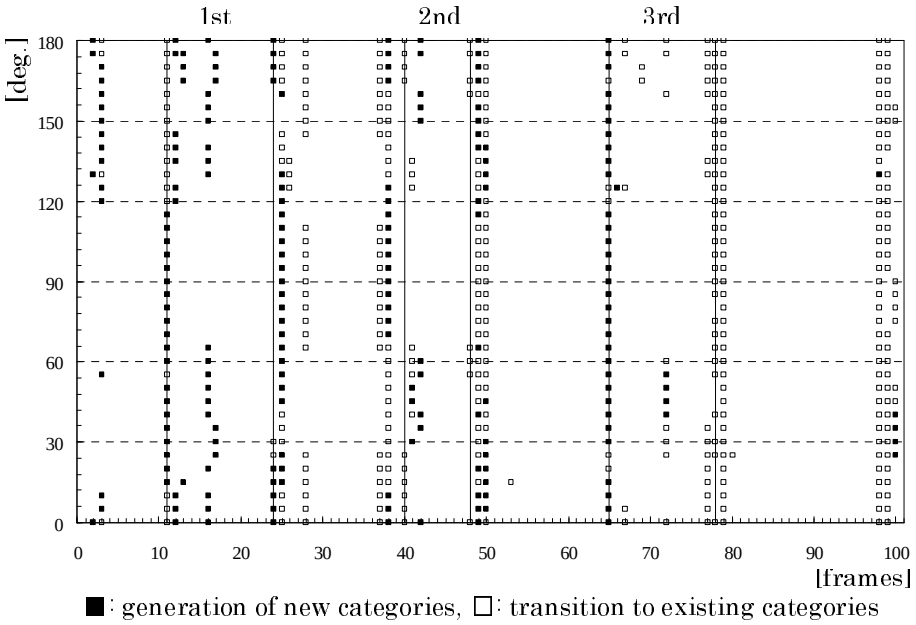


Figure 4.9. Categorical changes in sadness ($\rho = 0.97$).

the eyebrows at 0 and 180 degrees and nearby, is larger than for the other directions. The result of sadness is a large number of categories compared with other facial expressions. This result means that category changes of these directions are also remarkable.

Figures 4.8 — 4.13 show category changes in the case of $\rho = 0.970$. This figure shows the generation of new categories as filled rectangles and transitions to existing categories as empty rectangles. The vertical lines in each graph are appearance or disappearance of facial expressions as specified in the section 4.5.1. These lines that correspond to Table 4.1 show the changing frames of appearance and disappearance between the normal expression and each facial expression. Category changes are not required on these lines because the appearance and disappearance continue a few frames before and after specified frames.

The appearance of anger is represented for categorical changes of ART networks in all three times (Fig. 4.8). The directions at the first appearance are only three: 5, 10, and 15 degrees. The directions at the second and third appearance include wide ranges of directions, meaning that the ranges of orientation selectiv-

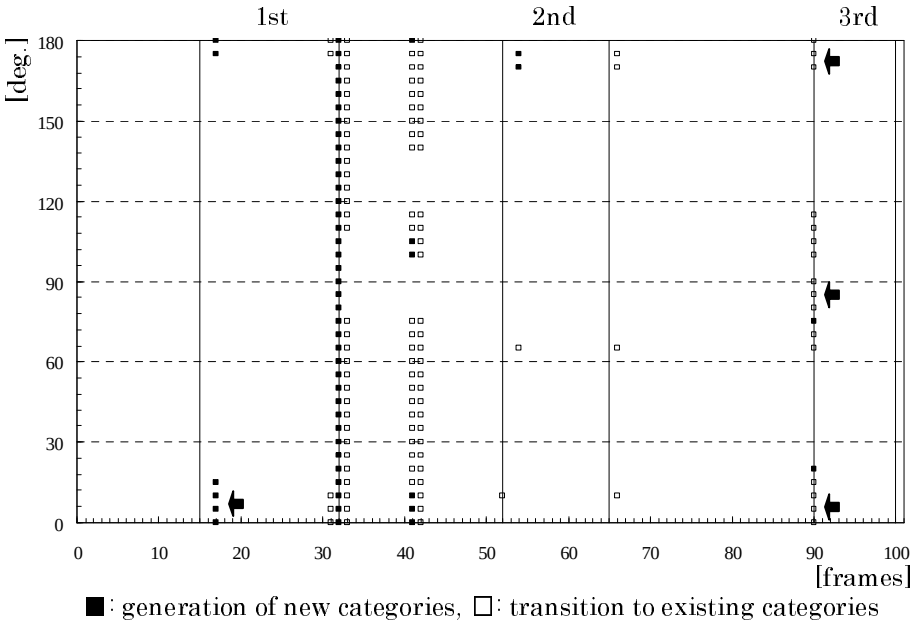


Figure 4.10. Categorical changes in disgust ($\rho = 0.97$).

ity are narrow at the first point and wide at the second and third points. The first and second disappearance of anger can be detected, but the third one can not be detected. The appearance and disappearance of sadness are represented (Fig. 4.9). However, the setting value of ρ is high because many categories occurred, except at the transition points. The expression of disgust shows a weak response (Fig. 4.10). This response means the classification granularity is low, although slight orientation selectivity is apparent. The categorical changes of happiness are redundant (Fig. 4.11). The classification granularity seems to be short because the second expression can only detect two directions: 100 and 105 degrees. In this case, the setting value of ρ cannot be increased. The open level of the mouth differs before and after the 34th frame of the first appearance. That difference is detectable for the category changes. The open level of the mouth at surprise is characteristic (Fig. 4.12). The categorical changes are noticeable around 90 degrees. However, in expectation of facial appearance points, the result is strongly reflective of eye blinking. The categorical changes of fear are not detected (Fig. 4.13). This result indicates that the vigilance parameter, $\rho = 0.970$, is small.

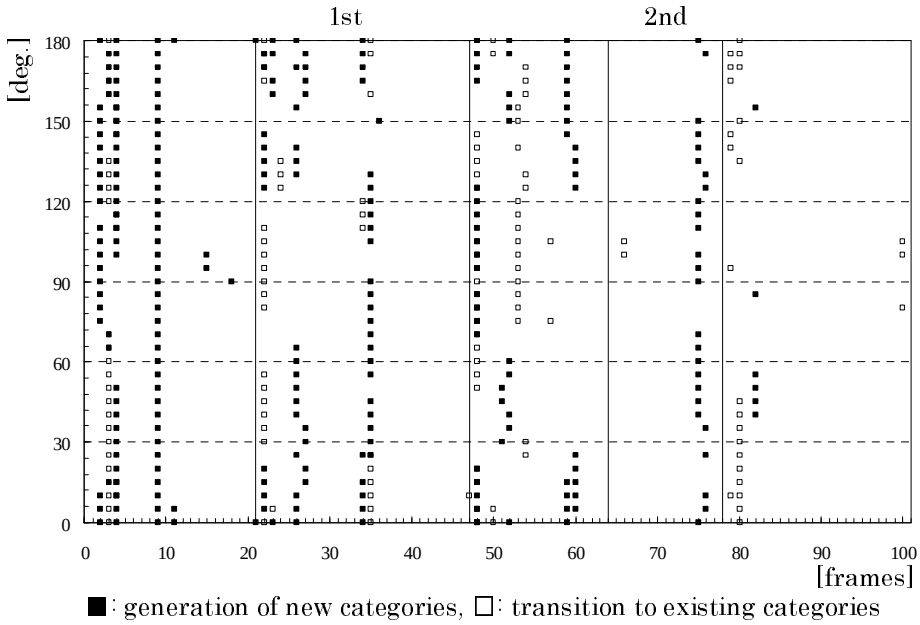


Figure 4.11. Categorical changes in happiness ($\rho = 0.97$).

Next, Figures 4.14–4.17 show the results of sadness, fear, and anger with changing vigilance parameters. The vigilance parameter of sadness were turned down step by 0.010 because the category is redundant in Fig. 4.9. The vigilance parameter of fear and anger were turned up to 0.980 because the classification granularity is short in Figs. 4.10 and 4.13. The redundant categories were decreased at the result of sadness in case of $\rho = 0.960$ (Fig. 4.14). The category changes are seen at the first appearance. Moreover, in the case of $\rho = 0.950$, the category changes are more apparent at the first and second appearance (Fig. 4.15). The category changes appeared all directions at the 98th and 99th frames. The cause is eye blinking, which occurred in the frames. We consider that the features of eye blinking are easy to divide into other features because eye blinking occurred in almost all directions. The appearance and disappearance of disgust appeared in the case of $\rho = 0.980$, especially in the second one (Fig. 4.16). In the case of $\rho = 0.980$ of fear, all appearances were detected (Fig. 4.17). Especially, it was detected in a wide range at the second appearance.

The method can detect facial appearance points, even in the lower setting of

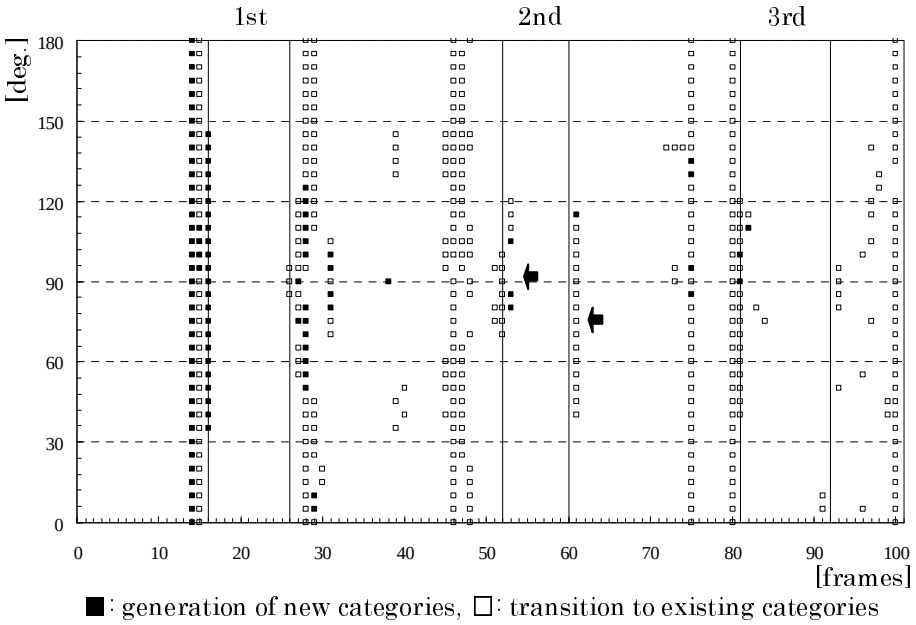


Figure 4.12. Categorical changes in surprise ($\rho = 0.97$).

ρ using orientation selectivity. In other words, the method can reduce redundant categories with the lower setting of ρ. Moreover, the method can detect facial expression changes with the range of avoiding redundant categories to increase the setting of ρ within orientation selectivity if the classification granularity is insufficient for a problem to be solved. We consider that the method can realize an advanced type of facial expression recognition for the next step of facial expression classification using the patterns of category changes with orientation selectivity.

4.6. Conclusion

This chapter presents a method for representation of facial expression changes using orientation selectivity of Gabor wavelets on ART networks. The method produced suitable vigilance parameters according to classification granularity using orientation selectivity. Moreover, the method represented the appearance and disappearance of facial expression changes to detect dynamic, local, and topological feature changes from whole facial images.

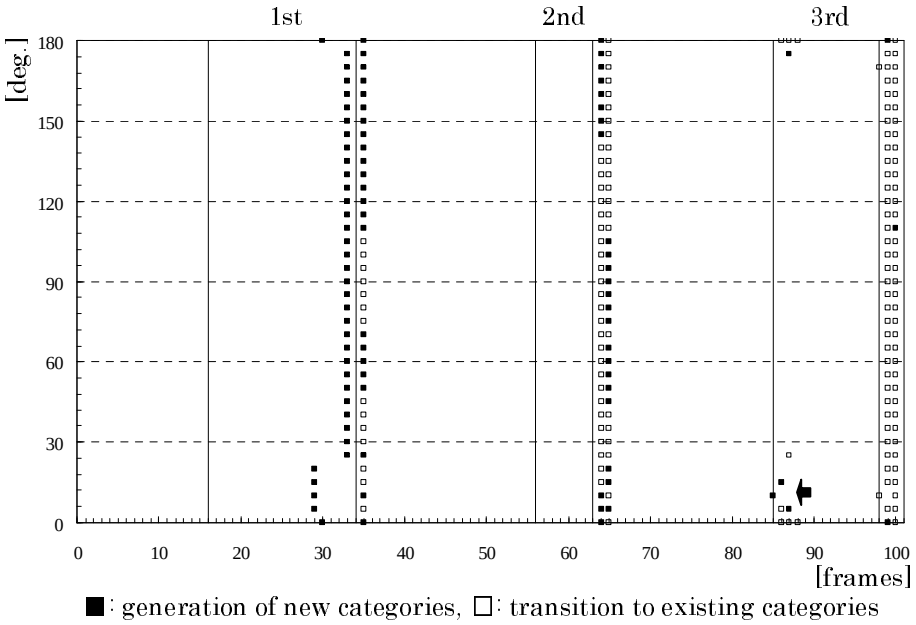


Figure 4.13. Categorical changes in fear ($\rho = 0.97$).

Future studies must evaluate other response selectivity, such as wavelength, amplitude, frequency and direction of motion. In addition, we are going to take examinations about the formation of categories for long-term facial changes, implementation of oblivion mechanisms, fusion with context information, etc. to realize a natural and flexible MMI.

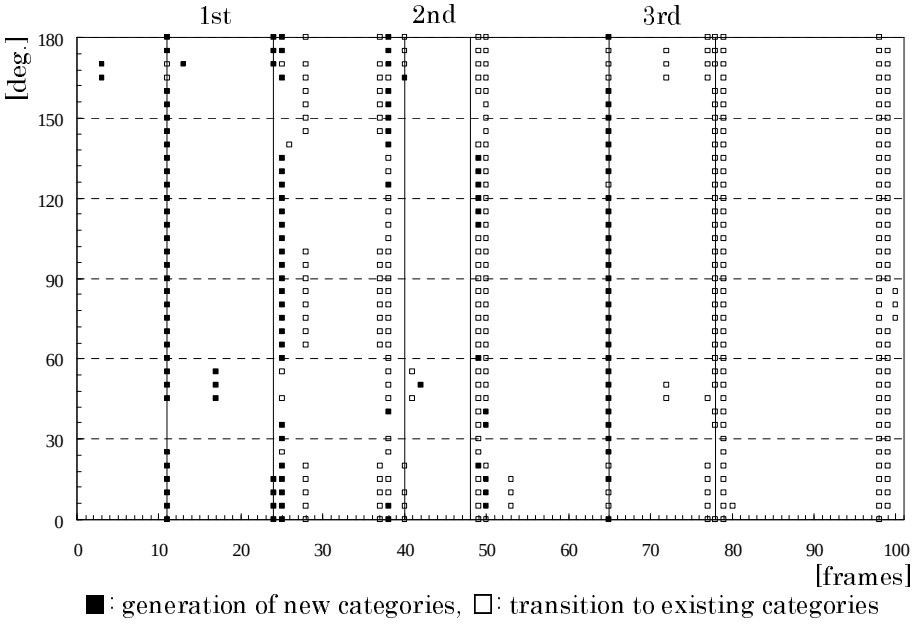


Figure 4.14. Categorical changes in sadness ($\rho = 0.96$).

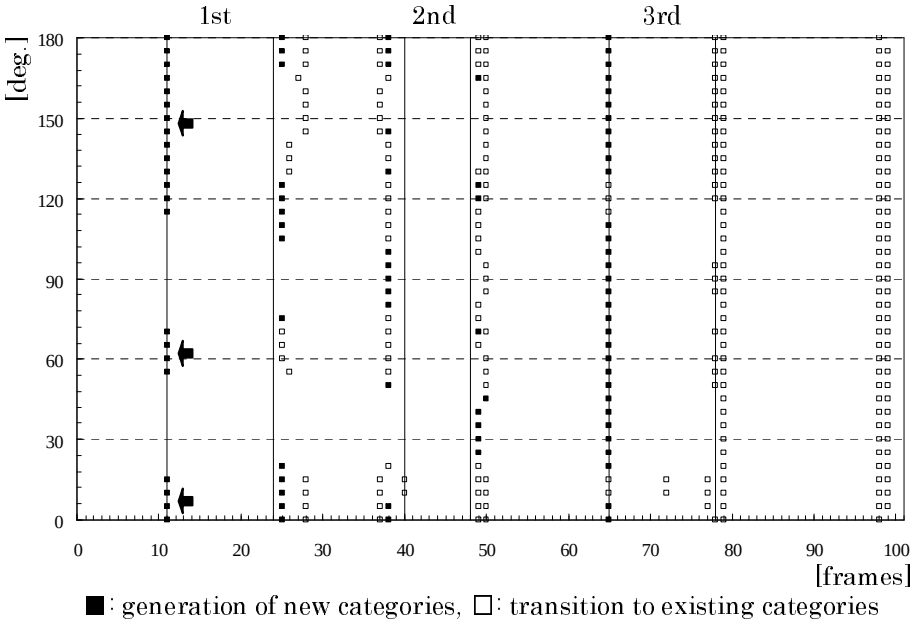


Figure 4.15. Categorical changes in sadness ($\rho = 0.95$).

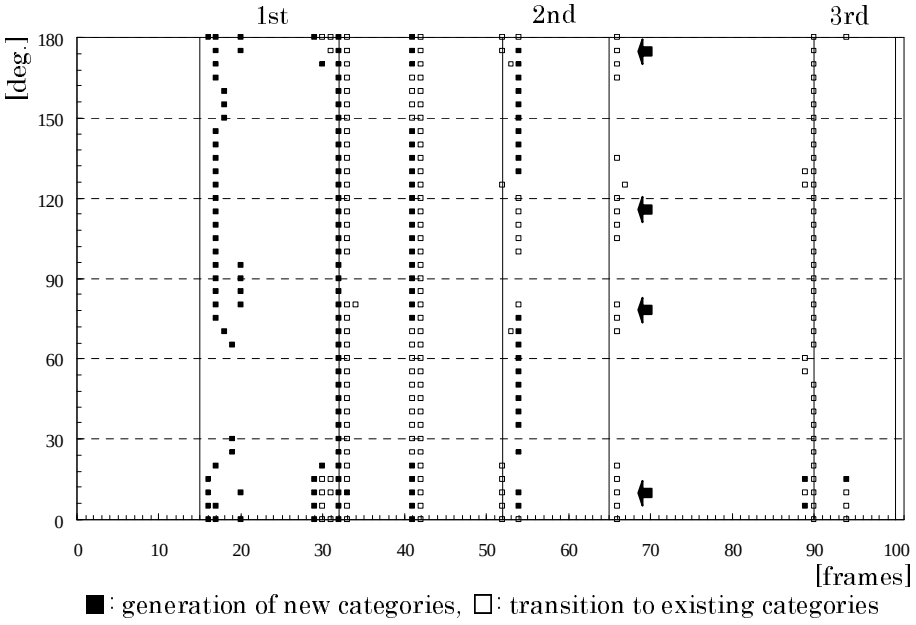


Figure 4.16. Categorical changes in disgust ($\rho = 0.98$).

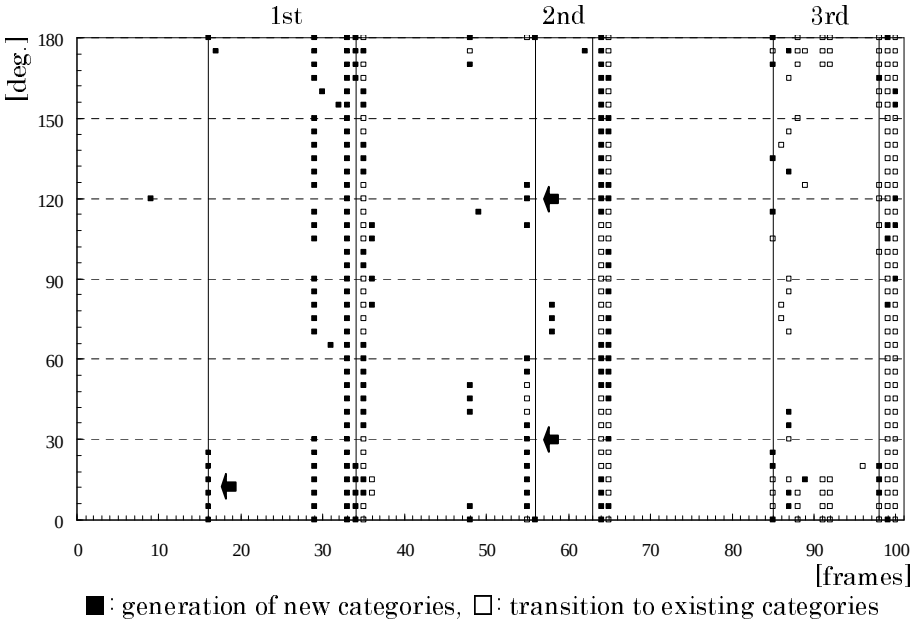


Figure 4.17. Categorical changes in fear ($\rho = 0.98$).

Chapter 5

Object Category Formation and Recognition

5.1. Introduction

Because of the advanced progress of computer technologies and machine learning algorithms, generic object recognition has been studied actively in the field of computer vision [5]. Generic object recognition is defined as a capability by which a computer can recognize objects or scenes to their general names in real images with no restrictions, i.e., recognition of category names from objects or scenes in images. In the study of robotics, one method to realize a robot having learning functions to adapt flexibly in various environments is to obtain brain-like memory: so-called world image maps [2]. For creating world image maps, robots must classify objects and scenes in time-series images into categories and memorize them as Long-Term Memory (LTM). Additionally, in actual environments for a robot, the number of categories is mostly unknown. Moreover, the categories are not known uniformly. Therefore, a robot must classify while generating additional categories.

This chapter presents unsupervised feature selection and category formation for application to robot vision. Our method has the following four capabilities. First, our method can localize target feature points using One Class-Support Vector Machines (OC-SVMs) [34] without previous setting of boundary information. Second, our method can generate labels as a candidate of categories for input

images while maintaining stability and plasticity together. Third, automatic labeling of category maps can be realized using labels created using Adaptive Resonance Theory-2 (ART-2) as teaching signals for Counter Propagation Networks (CPNs). Fourth, our method can present the diversity of appearance changes for visualizing spatial relations of each category on a two-dimensional map of CPNs. Through object classification experiments, we evaluate our method using the Caltech-256 object category dataset [10], which is the *defacto* standard benchmark dataset for comparing the performance of algorithms in generic object recognition, and time-series images taken by a camera on a mobile robot.

This chapter presents the following. First, we describe related work in Section 5.2. Next, we present the number of classification targets of categories in an actual environment based on a questionnaire investigation in Section 5.3 Subsequently, we explain detailed specifications of our image representation method, our category formation method, and the whole architecture of our method in Sections 5.4, 5.5, and 5.6, respectively. We present experimental results in Sections 5.7 5.8, and 5.10. Finally, we respectively present related discussion and salient conclusions in Sections 5.11 and 5.12.

5.2. Related studies

The problem of Simultaneous Localization and Mapping (SLAM) has attracted immense attention in mobile robotics studies [86]. The objective of SLAM is to build a map and update it while simultaneously estimating locations for a robot. Cummins et al. proposed Fast Appearance Based Mapping (FAB-MAP) [87] as a probabilistic approach to recognizing places based on their appearance. The objective of FAB-MAP is similar to SLAM: to build a map of routes using appearance changes of scene images obtained using a camera on a mobile robot. Our objective is to classify images obtained using a camera on a mobile robot in categories for recognizing objects.

Learning-based object classification methods are roughly divisible into supervised object classification methods and unsupervised object classification methods. Supervised object classification methods require training datasets including teaching signals extracted from ground-truth labels. However, unsupervised ob-

ject classification methods require no teaching signals with which categories are automatically extracted to a problem of unknown classification categories for classifying images into respective categories. Recently, studies of unsupervised object classification methods have been active. The subject has attracted attention because it might provide technologies to classify visual information flexibly in various environments.

In recent studies of object classification, various methods have been proposed to combine the process of detecting regions or positions of an object as a target of classification and recognition [93, 94, 95, 96, 97, 98, 99, 100, 101, 102]. Barnard et al. proposed a word–image translation model as a method based on regions [89]. They automatically annotated segmentation images using images that assigned some keywords previously. Lampert et al. proposed an Efficient Subwindow Search (ESS) that can quickly detect a position of an object using branch and bound methods and integration images [90]. Using ESS, they realized first partial generic object detection to calculate previously output values of Support Vector Machines (SVMs) in each feature point and to localize a search range gradually. Moreover, Suzuki [91] et al. proposed a local feature selection method used in Bag-of-Features (BoF) [92] with SVMs. This method classifies local features into background features and target features used for BoF. However, these methods require previously acquired training samples with teaching signals. Therefore, these methods are inapplicable to an actual environment for which a target region and a background region can not be decided uniformly.

As unsupervised object classification methods, Sivic et al. proposed an unsupervised object classification method using pLSA and LDA, which are generative models from the statistical text literature [93]. 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 (PGMMs) of generative models that combined Probabilistic Context-Free Grammars (PCFGs) and Markov Random Fields (MRFs) [94]. They used this method to create an object category model for object detection and unsupervised object classification. Moreover, they proposed Probabilistic Object Models (POMs) that improved their method and enabled classification, segmentation, and recognition of objects [95]. Todorovic

Figure 5.1. Photos in the target environment for the questionnaire investigation.

Table 5.1. Results of questionnaires administered to 10 subjects.

	A	B	C	D	E	F	G	H	I	J	Max.	Min.	Ave.
Rough	22	11	17	6	4	7	12	14	11	8	22	4	11
Fine	37	24	41	17	17	14	20	36	35	34	41	14	28

et al. proposed an unsupervised identification method using optical, geometric, and topological characteristics of multiscale regions consisting of two-dimensional objects [96]. They represented each image as a tree structure by division of multiscale images. Moreover, Nakamura et al. proposed an unsupervised object classification method using multimodal information of vision, hearing, and touch [97]. They achieved object 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, these methods are applied only slightly to classification problems in an actual environment for which the number of categories is unknown.

5.3. Categories in an actual environment

Numerous categories exist in an actual environment. Humans can recognize several tens of thousands of categories [88]. We consider that it is possible for a robot to classify categories in an actual environment to specify them clearly. In this chapter, we used a questionnaire investigation to find the number of classification targets of categories used for an actual environment. The target environment is

Table 5.2. Categories from which more than two subjects were extracted as a rough classification from the questioner investigation.

Number	Extracted Categories
7	computers, chairs, whiteboard.
5	human, books, desks, refrigerator, sink.
4	shelves, partitions, rockers.
3	plant, bookshelves, doors.
2	goods, electrical appliances, stationery, boxes, windows, sundry goods, file shelves, printers, microwave oven, trash boxes, shared shelves, walls, dinning table.

our research room at the Neuro Informatics Laboratory, Akita Prefectural University. Fig. 5.1 depicts photographs taken in the room. The floor space is about 90 square meters. Ten university students participated as subjects. They walked around the room a few minutes for observation. Subsequently, they wrote categories that they found and recognized as a categories on the questionnaire sheet. The questionnaire sheets consisted of two classification types: rough classification and fine classification.

Table 5.1 presents results of the number of categories to be extracted with this investigation. In the rough classification, 11 categories were extracted, consisting of 4 minimum categories and 22 maximum categories. In the fine classification, 28 categories were extracted, consisting of 14 minimum categories and 44 maximum categories. Table 5.2 categories from which more than two subjects were extracted. In the rough classification, chairs, desks, computers, etc., which are numerous in the research room, are extracted. Moreover, large objects such as a whiteboard and a refrigerator, for which the number of the category is one instance in the room, are extracted. In the fine classification presented in Table 5.3, small items such as cups and umbrellas are extracted, although categories that are the same in the rough classification are extracted. Extracted objects such as PaPeRo (a communication robot produced by NEC), Mindstorms (a self-assembled robot by LEGO), and NetTansors (a web-camera embedded robot by

Table 5.3. Categories from which more than two subjects were extracted as fine classification from the questioner investigation.

Number	Extracted Categories
9	trash boxes
8	books
6	chairs, rockers.
5	desks, shelves, whiteboard, microwave oven, printers, MindStorms, refrigerator.
4	doors, table taps, bookshelves, cups, computers, keyboards, tissue boxes, umbrellas.
3	bags, TV, mice, cameras, radio control cars, blind, NetTansors.
2	pens, tool boxes, windows, shoes, sundry shelves, displays, PC desks, pot, clear boxes, terrestrial globe, clock, plant, coffee machine, rocking chair, dishes, sink, robots on desks, walls, miniature garden, teacher's area, hardware, laptop PC, staplers, hollow punch, network cameras, USB memory units, people.

Bandai) are extracted in each category that can be extracted to one category as a robot.

5.4. Image representation

In fact, BoF [92], which represents features for histograms of visual words with local features as typical patterns extracted from numerous images, is widely used to emphasize the effectiveness in image representation methods of generic object recognition. In BoF of our method depicted in Fig. 5.2, we applied OC-SVMs for selecting Scale-Invariant Feature Transform (SIFT) [33] feature points as target regions in an image. Furthermore, we applied SOMs for creating visual words and histograms in each image from selected features.

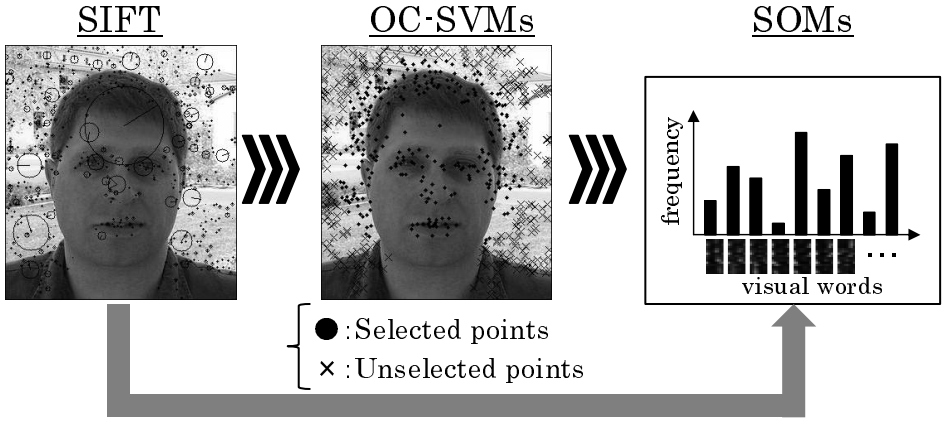


Figure 5.2. Procedures of our image representation method based on BoF.

Our target is SIFT feature points on an object for recognition. Therefore, target regions and target feature points respectively mean object regions and feature points on an object. The OC-SVMs are unsupervised-learning-based binary classifiers that enable density estimation without estimating a density function. Therefore, OC-SVMs can apply to real-world images without boundary information. Detailed algorithms of SIFT, OC-SVMs, and SOMs are the following.

5.4.1 Description of features using SIFT

Generally, SIFT is used as a descriptive method of local features in generic object recognition. Mikolajczyk et al. [104] compared descriptors of varias types such as shape context [105], steerable filters [106], PCA-SIFT [107], differential invariants [108], spin images [109], SIFT [33], complex filters [110], and moment invariants [111]. They showed that the SIFT-based descripters performs the best [104]. The SIFT processing consists of two steps: detection of feature points and description of features [112]. The procedures are the following.

Detection of scale-space extrema

The first stage of keypoint detection is to identify locations and scales that can be repeatably assigned under differing views of the same object. Detecting locations that are invariant to scale change of the image can be accomplished by searching for stable features across all possible scales, using a continuous function of scale

known as scale space. The scale space of an image is defined as a function, $L(u, v, \sigma)$, that is produced from the convolution of a variable-scale Gaussian, $G(x, y, \sigma)$, with an input image, $I(u, v)$. The difference-of-Gaussian function convolved with the image, $D(u, v, \sigma)$, can be computed from the difference of two nearby scales separated by a constant multiplicative factor k as

$$D(u, v, \sigma) = (G(x, y, k\sigma) - G(x, y, \sigma)) * I(u, v). \tag{5.1}$$

where

$$L(u, v, \sigma) = G(x, y, \sigma) * I(u, v). \tag{5.2}$$

Therefre, $D(u, v, \sigma)$ is defined as

$$D(u, v, \sigma) = L(u, v, k\sigma) - L(u, v, \sigma). \tag{5.3}$$

Herein, it has been shown by Koenderink [113] and Lindeberg [114] that under a variety of reasonable assumptions the only possible scale-space kernel is the Gaussian function as

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} exp(-\frac{x^2 + y^2}{2\sigma^2}) \tag{5.4}$$

The relation between D and $\sigma^2\nabla^2$ can be understood from the heat diffusion equation as

$$\sigma^2\nabla^2 = \frac{\partial G}{\partial \sigma} \approx \frac{G(x, y, k\sigma) - G(x, y, \sigma)}{k\sigma - \sigma}. \tag{5.5}$$

Therefore,

$$G(x, y, k\sigma) - G(x, y, \sigma) \approx (k - 1)\sigma^2\nabla^2 \tag{5.6}$$

In order to detect the local maxima and minima of $D(x, y, \gamma)$, each sample point is compared to its eight neighbors in the current image and nine neighbors in the scale above and below. It is selected only if it is larger than all of these neighbors or smaller than all of them. The cost of this check is reasonably low due to the fact that most sample points will be eliminated following the first few checks. An important issue is to determine the frequency of sampling in the

image and scale domains that is needed to reliably detect the extrema. However, it turns out that there is no minimum spacing of samples that will detect all extrema, as the extrema can be arbitrarily close together. This can be seen by considering a white circle on a black background, which will have a single scale space maximum where the circular positive central region of the difference-of-Gaussian function matches the size and location of the circle. For a very elongated ellipse, there will be two maxima near each end of the ellipse. As the locations of maxima are a continuous function of the image, for some ellipse with intermediate elongation there will be a transition from a single maximum to two, with the maxima arbitrarily close to each other near the transition. Therefore, we must settle for a solution that trades off efficiency with completeness. In fact, as might be expected and is confirmed by our experiments, extrema that are close together are quite unstable to small perturbations of the image.

Eliminating edge responses

The difference-of-Gaussian function will have a strong response along edges, although the location along the edge is poorly determined and therefore unstable to small amounts of noise. A poorly defined peak in the difference-of-Gaussian function will have a large principal curvature across the edge but a small one in the perpendicular direction. The principal curvatures can be computed from a two-dimensional Hessian matrix, $\boldsymbol{H}$, computed at the location and scale of the keypoint as

$$\mathbf{H} = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy}. \end{bmatrix} \tag{5.7}$$

The eigenvalues of H are proportional to the principal curvatures of D. Borrowing from the approach used by Harris and Stephens [115], we can avoid explicitly computing the eigenvalues, as we are only concerned with their ratio. Let αbe the eigenvalue with the largest magnitude and β be the smaller one. Then, we can compute the sum of the eigenvalues from the trace of $\boldsymbol{H}$, Tr($\boldsymbol{H}$), and their product from the determinant,Det($\boldsymbol{H}$), as

$$\text{Tr}(\boldsymbol{H}) = D_{xx} + D_{yy} - (D_{xy})^2 = \alpha\beta, \tag{5.8}$$

$$\mathrm{Det}(\boldsymbol{H}) = D_{xx}D_{yy} = \alpha + \beta. \tag{5.9}$$

In the unlikely event that the determinant is negative, the curvatures have different signs so the point is discarded as not being an extremum. Let r be the ratio between the largest magnitude eigenvalue and the smaller one, so that $\alpha = \gamma\beta$. Then,

$$\frac{\mathrm{Tr}(\boldsymbol{H})^2}{\mathrm{Det}(\boldsymbol{H})} = \frac{(\alpha+\beta)^2}{\alpha\beta} = \frac{(\gamma\beta+\beta)^2}{\gamma\beta^2} = \frac{(\gamma+1)^2}{\gamma}. \tag{5.10}$$

which depends only on the ratio of the eigenvalues rather than their individual values. The quantity (r+1)2/r is at a minimum when the two eigenvalues are equal and it increases with r. Therefore, to check that the ratio of principal curvatures is below some threshold, r, as

$$\frac{\mathrm{Tr}(\boldsymbol{H})^2}{\mathrm{Det}(\boldsymbol{H})} < \frac{(\gamma_{th}+1)^2}{\gamma_{th}}. \tag{5.11}$$

This is very efficient to compute, with less than 20 floating point operations required to test each keypoint. The experiments in this paper use a value of $r = 10$, which eliminates keypoints that have a ratio between the principal curvatures greater than 10.

Eliminating low contrast keypoints

The next step is to perform a detailed fit to the nearby data for location, scale, and ratio of principal curvatures. This information allows points to be rejected that have low contrast and are therefore sensitive to noise or are poorly localized along an edge. The approach proposed by Brown et al. [116] uses the Taylor expansion up to the quadratic terms of the scale-space function, $D(x, y, \gamma)$, shifted so that the origin is at the sample point:

$$D(\mathrm{x}) = D + \frac{\partial D}{\partial \mathrm{x}}^T \mathrm{x} + \frac{1}{2}\mathrm{x}^T \frac{\partial^2 D}{\partial \mathrm{x}^2}\mathrm{x}. \tag{5.12}$$

The location of the extremum, $\hat{\mathrm{x}}$, is determined by taking the derivative of this function with respect to x and setting it to zero, giving

$$\hat{\mathrm{x}} = -\frac{\partial^2 D}{\partial \mathrm{x}^2}^{-1} \frac{\partial D}{\partial \mathrm{x}}. \tag{5.13}$$

The function value at the extremum, $D(\hat{\mathrm{x}})$, is useful for rejecting unstable extrema with low contrast. This can be obtained by substituting equation 5.13 into 5.12, giving

$$D(\hat{\mathrm{x}}) = D + \frac{1}{2}\frac{\partial D}{\partial \mathrm{x}}^T \hat{\mathrm{x}}. \tag{5.14}$$

The resulting 3×3 linear system can be solved with minimal cost. If the offset $\hat{x}$ is larger than 0.5 in any dimension, then it means that the extremum lies closer to a different sample point. In this case, the sample point is changed and the interpolation performed instead about that point. The final offset $\hat{x}$ is added to the location of its sample point to get the interpolated estimate for the location of the extremum.

Orientation assignment

The keypoint descriptor can be represented relative to this orientation and therefore achieve invariance to image rotation. The scale of the keypoint is used to select the Gaussian smoothed image, L, with the closest scale, so that all computations are performed in a scale-invariant manner. For each image sample, $L(x, y)$, at this scale, the gradient magnitude, $m(x, y)$, and orientation, $\theta(u, v)$, is precomputed using pixel differences as

$$m(u, v) = \sqrt{f_u(u, v)^2 + f_v(u, v)^2}, \tag{5.15}$$

$$\theta(u, v) = \tan^{-1}\frac{f_v(u, v)}{f_u(u, v)}. \tag{5.16}$$

$$\begin{cases} f_u(u, v) = L(u + 1, v) - L(u - 1, v) \\ f_v(u, v) = L(u, v + 1) - L(u, v - 1) \end{cases} \tag{5.17}$$

An orientation histogram is formed from the gradient orientations of sample points within a region around the keypoint. The orientation histogram has 36 bins covering the 360 degree range of orientations. Each sample added to the histogram is weighted by its gradient magnitude and by a Gaussian-weighted circular window with a γ that is 1.5 times that of the scale of the keypoint. Peaks in

the orientation histogram correspond to dominant directions of local gradients. The highest peak in the histogram is detected, and then any other local peak that is within 80 % of the highest peak is used to also create a keypoint with that orientation. Therefore, for locations with multiple peaks of similar magnitude, there will be multiple keypoints created at the same location and scale but different orientations. Only about 15 % of points are assigned multiple orientations, but these contribute significantly to the stability of matching. Finally, a parabola is fit to the 3 histogram values closest to each peak to interpolate the peak position for better accuracy.

Descriptor representation

The final step is to compute a descriptor for the local image region that is highly distinctive yet is as invariant as possible to remaining variations, such as change in illumination or 3D viewpoint.

The image gradient magnitudes and orientations are sampled around the keypoint location, using the scale of the keypoint to select the level of Gaussian blur for the image. In order to achieve orientation invariance, the coordinates of the descriptor and the gradient orientations are rotated relative to the keypoint orientation. For efficiency, the gradients are precomputed for all levels of the pyramid. A Gaussian weighting function with γ equal to one half the width of the descriptor window is used to assign a weight to the magnitude of each sample point. The purpose of this Gaussian window is to avoid sudden changes in the descriptor with small changes in the position of the window, and to give less emphasis to gradients that are far from the center of the descriptor, as these are most affected by misregistration errors. The keypoint allows for significant shift in gradient positions by creating orientation histograms over 4×4 sample regions. A gradient sample on the left can shift up to 4 sample positions while still contributing to the same histogram on the right, thereby achieving the objective of allowing for larger local positional shifts.

It is important to avoid all boundary affects in which the descriptor abruptly changes as a sample shifts smoothly from being within one histogram to another or from one orientation to another. Therefore, trilinear interpolation is used to distribute the value of each gradient sample into adjacent histogram bins. In

other words, each entry into a bin is multiplied by a weight of $1 - d$ d for each dimension, where d is the distance of the sample from the central value of the bin as measured in units of the histogram bin spacing. The experiments in this paper use a $4 \times 4 \times 8 = 128$ element feature vector for each keypoint.

Finally, the feature vector is modified to reduce the effects of illumination change. First, the vector is normalized to unit length. A change in image contrast in which each pixel value is multiplied by a constant will multiply gradients by the same constant, so this contrast change will be canceled by vector normalization. A brightness change in which a constant is added to each image pixel will not affect the gradient values, as they are computed from pixel differences. Therefore, the descriptor is invariant to affine changes in illumination. However, non-linear illumination changes can also occur due to camera saturation or due to illumination changes that affect 3D surfaces with differing orientations by different amounts. These effects can cause a large change in relative magnitudes for some gradients, but are less likely to affect the gradient orientations. Therefore, we reduce the influence of large gradient magnitudes by thresholding the values in the unit feature vector to each be no larger than 0.2, and then renormalizing to unit length. This means that matching the magnitudes for large gradients is no longer as important, and that the distribution of orientations has greater emphasis. The value of 0.2 was determined experimentally using images containing differing illuminations for the same 3D objects.

5.4.2 Selected feature points using OC-SVMs

As described earlier, the OC-SVMs are unsupervised learning classifiers that estimate the dense region without estimation of the density function [34]. The OC-SVMs set a hyperplane that separates data points near the original point and the other data points using the characteristic by which the outlier data points are mapped near the original point on a feature space with a kernel function. The discriminant function $f(\cdot)$ is calculated to divide input feature vectors x_{i} into two parts as shown in 5.3. The position of the hyperplane is changed according to parameter ν, which controls outliers of input data with change, and which has

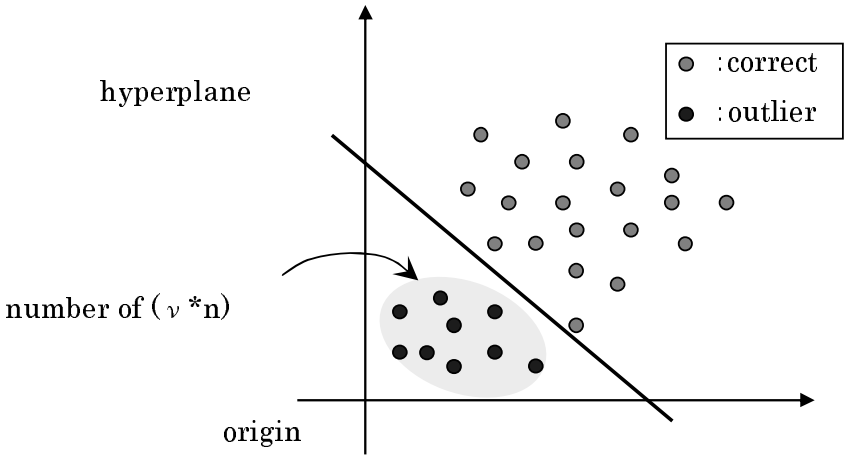


Figure 5.3. Distribution of correct and outlier data points and the hyperplane on a high-dimension feature space of OC-SVMs.

range of 0–1.

$$f(x) = sgn(\omega^\top \Phi(x) - \rho) \tag{5.18}$$

The restriction is set to the following.

$$\begin{aligned} \omega^\top z_i &\geq \rho - \zeta_i, i = 1, ..., l \\ \zeta_i &\geq 0, i = 1, ..., l, 0 < \nu \leq 1 \end{aligned} \tag{5.19}$$

The optimization problem is solved with the following restriction

$$\begin{aligned} \frac{1}{2}\|\omega\|^2 + \frac{1}{\nu l}\sum_{i=1}^{l} \quad & \zeta_i \quad -\rho \\ & \rightarrow \quad \min\, \omega, \zeta, \text{and } \rho \end{aligned} \tag{5.20}$$

Therein, z_i represents results of the mapping input vector x_i to the high-dimension feature space.

$$\Phi : x_i \mapsto z_i \tag{5.21}$$

In those expressions, ω and ρ are results of the optimization problem. The Lagrangian function of the optimization problem is calculated to solve the optimiza-

tion problem.

$$L(\omega, \zeta, \rho, \alpha, \beta) = \frac{1}{2}\|\omega\|^2 + \frac{1}{\nu l}\sum_{i=1}^{l}\zeta_i - \rho$$
$$-\sum_{i=1}^{l}\alpha_i((\omega^\top z_i) - \rho + \zeta_i) - \sum_{i=1}^{l}\beta_i\zeta_i \tag{5.22}$$

In those expressions, α and β of the Lagrangian function are maximized. Ω, ρ and ζ of the Lagrangian function are minimized. Lagrangian functions that are partially differentiated by ω, b, ρ and ζ are 0 for an optimized solution.

$$\frac{\partial}{\partial\omega}L = 0 \rightarrow \omega = \sum_{i=1}^{l}\alpha_i z_i \tag{5.23}$$

$$\frac{\partial}{\partial\zeta_i}L = 0 \rightarrow \alpha_i = \frac{1}{\nu l} - \beta_i \tag{5.24}$$

$$\frac{\partial}{\partial\rho}L = 0 \rightarrow \sum_{i=1}^{l}\alpha_i = 1 \tag{5.25}$$

$$\begin{cases} \alpha_i \cdot [\rho - \zeta_i - \omega^\top z_i] = 0, \quad i = 1, ..., l \\ \rho - \zeta_i - \omega^\top z_i \leq 0, \quad i = 1, ..., l \\ 0 \leq \alpha_i \leq \frac{1}{\nu l}, \quad i = 1...., l \\ \beta_i \cdot \zeta_i = 0, \quad -\zeta_i \leq 0, \quad \beta_i \geq 0, \quad i = 1, ..., l \end{cases} \tag{5.26}$$

Equations (5.23)–(5.26) are substituted to Lagrangian function. A binary optimization problem is developed if the inner product is transposed to the kernel.

$$\frac{1}{2}\sum_{i,j=1}^{l}\alpha_i\alpha_j k(z_i^\top z_j),$$
$$0 \leq \alpha_i \leq \frac{1}{\nu l}, i = 1, ..., l, \sum_{i=1}^{l}\alpha_i = 1 \tag{5.27}$$

Support vectors are learning data z_i fulfilling assumptions of (5.26) , $\alpha_\mathrm{i} > 0$ and ζ_i=0. The equation (5.23) is expanded. An equality is true if α_i and β_i are not 0 for an optimized solution and ρ is calculated as

$$f(z) = \sum_{i=1}^{l}\alpha_i k(x_i, z) - \rho, \tag{5.28}$$

where $\zeta_i = 0$. Points of $\Phi(x)$ are not apparent in the discriminant function that is a binary problem using a kernel trick. Therefore, huge calculation costs of the inner product can be avoided and the number of calculations can be reduced. Parameter ν of OC-SVMs is a high limit of unselected data and lower limit of support vectors if the solution of the optimization problem (5.20) fulfills $\rho \neq 0$.

5.4.3 Creating visual words using SOMs

For our method, we apply SOMs, not k-means, which is generally used in BoF, for creating visual words. In the learning step, SOMs update weights while maintaining topological structures of input data. Actually, SOMs create neighborhood regions around the burst unit, which demands a response of the input data. Therefore, SOMs can classify various data whose distribution resembles the training data. In addition, Terashima et al. reported that SOMs are superior to k-means as an unsupervised classification method that is useful to minimize misrecognition [117]. The learning algorithm of SOMs [15] is the same as the algorithm used between the input-Kohonen layers of CPNs. In this method, we used all SIFT features for creating visual words at the learning step of SOMs. We used SIFT features selected by OC-SVMs for generating histograms based on visual words. Based on our preliminary experiment, we set the learning iteration to 100,000 times. Additionally, we set the number of units of the Kohonen layer to 100 units. We created visual words to extract weights between Kohonen layer units and input layer units.

5.5. Unsupervised category formation

Figure 5.4 depicts the architecture of our unsupervised category formation method that combined incremental learning of ART-2 and self-mapping characteristics of CPNs. Actually, ART-2 is a theoretical model of unsupervised neural networks of incremental learning that forms categories adaptively while maintaining stability and plasticity together. Features of time-series images from the mobile robot change with time. Using ART-2, our method enables an unsupervised category formation that requires no setting of the number of categories.

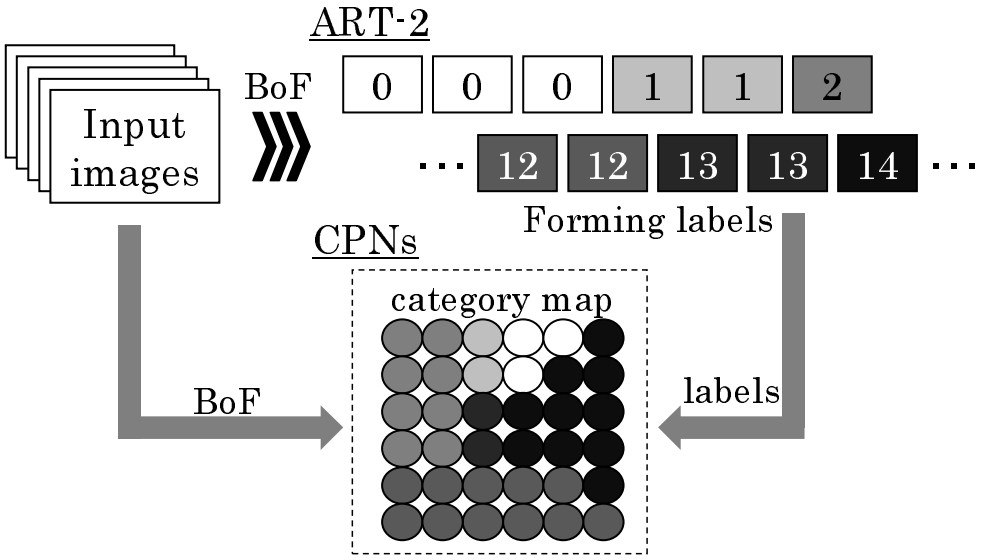


Figure 5.4. Architecture of our unsupervised category formation method.

A type of supervised neural network, CPN, actualizes 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 formations are represented as a category map on the Kohonen layer. Our method can reduce these labels using the Winner-Takes-All competition of CPNs. In addition, our method can visualize relations between categories on the category map of CPNs. Detailed algorithms of ART-2 and CPNs are the following.

5.6. Whole architecture of our method

In generic object recognition, it is a challenging task to develop a unified model to address all steps from feature representation to creation of classifiers. The aim of our study is the realization of category formation for generic object recognition to apply theories with different characteristics for each step. Fig. 5.5 depicts the network architecture of our method. The procedures are the following.

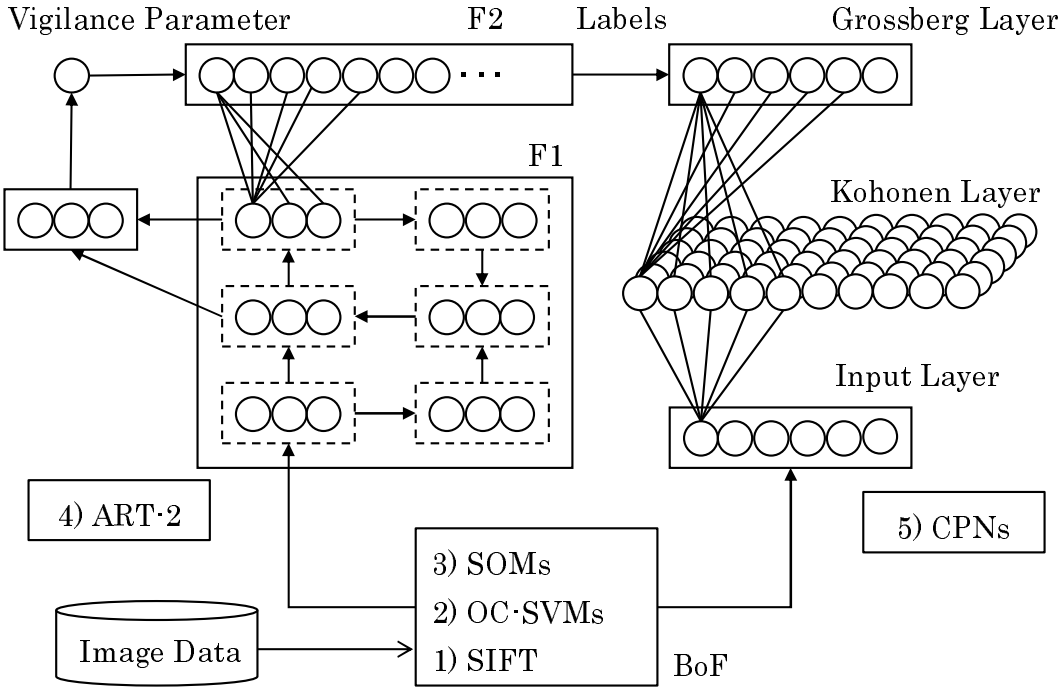


Figure 5.5. Whole architecture of our method.

1. Extracting feature points and calculating descriptors using SIFT
2. Selecting SIFT features using OC-SVMs
3. Creating visual words of all SIFT descriptors and calculating histograms of selected SIFT descriptors matched with visual words using SOM
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 BoF. We apply OC-SVMs to select SIFT feature points for localizing target regions in an image. For producing visual words, we use SOMs, which can learn neighborhood regions while updating the cluster structure, although k-means must decide data of the center of a cluster. Actually, SOMs can represent visual words that minimize misclassification [117]. Furthermore, the

Table 5.4. Setting values of parameters used in experiments.

Parameters		Setting values
OC-SVMs	ν	0.5
ART-2	θ	0.1
	ρ	0.920
CPNs	$\alpha(t)$	0.5
	$\beta(t)$	0.5
	learning iteration	10,000

combination of ART-2 and CPNs enables unsupervised category formation that labels a large quantity of images in each category automatically. Table 5.4 shows parameters of OC-SVMs, ART-2, and CPNs with each experiment.

5.7. Experimental results obtained using the Caltech-256 dataset

This section presents experimental results of image classification using Caltech-256 [10] to compare the performance of algorithms in generic object recognition. The target of this experiment is object classification of static images because Caltech-256 has no temporal factors in each category. We use the highest 20 categories with the number of images in 256 categories. The results of selection of SIFT features and recognition rates for classification of 5, 10, and 20 categories are the following.

5.7.1 Selection of feature points and generation of labels

Figures 5.6 and 5.7 depict results of selected feature points using OC-SVMs on five sample images of Caltech-256. Fig. 5.6 shows that our method can select feature points of target objects in images of the Leopards and Face categories. In addition, Fig. 5.7 shows that our method can select feature points around the wings that characterize airplanes for various images of the Airplane category.

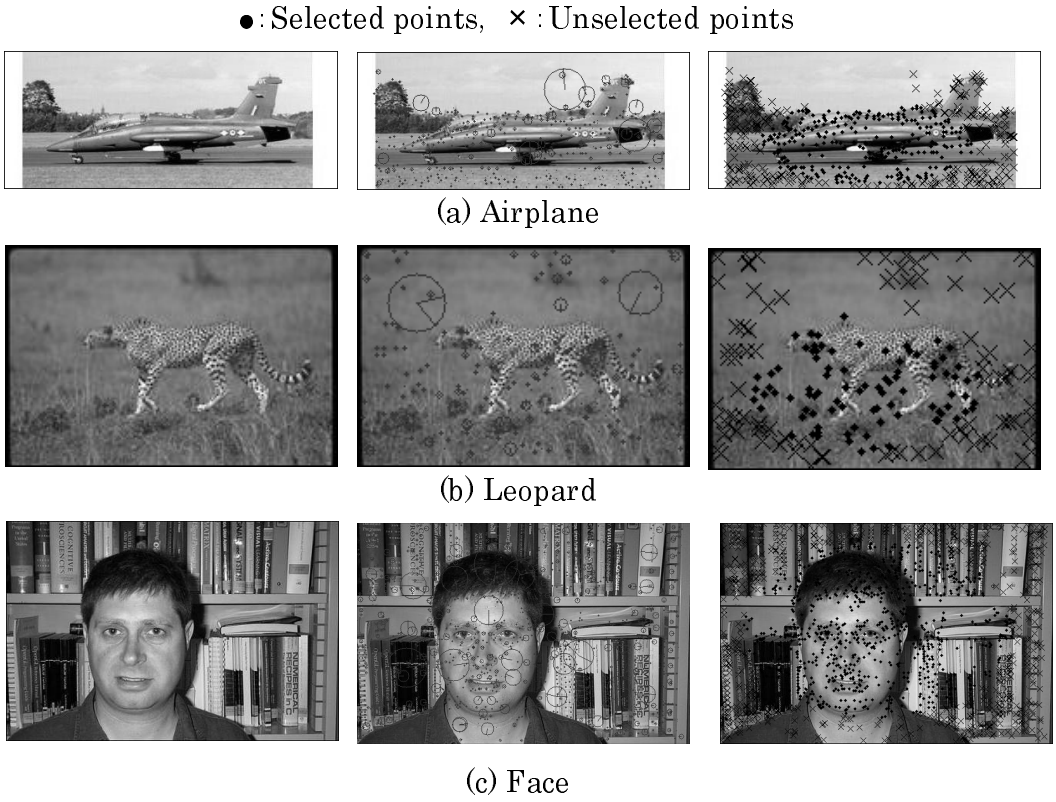


Figure 5.6. Results of selected SIFT feature points in different categories of Caltech-256.

Figure 5.8 depicts labels generated by ART-2. The vertical and horizontal axes respectively represent labels and images. The independent labels in each category without confusion are generated among different categories. Moreover, for the Airplane, Motorbike, and Face categories one label is generated; for the Car-side and Leopards categories several labels are generated. These results demonstrate that OC-SVMs can select SIFT features of target objects and show that ART-2 can generate independent labels to images for which backgrounds and appearances of objects differ in each category.

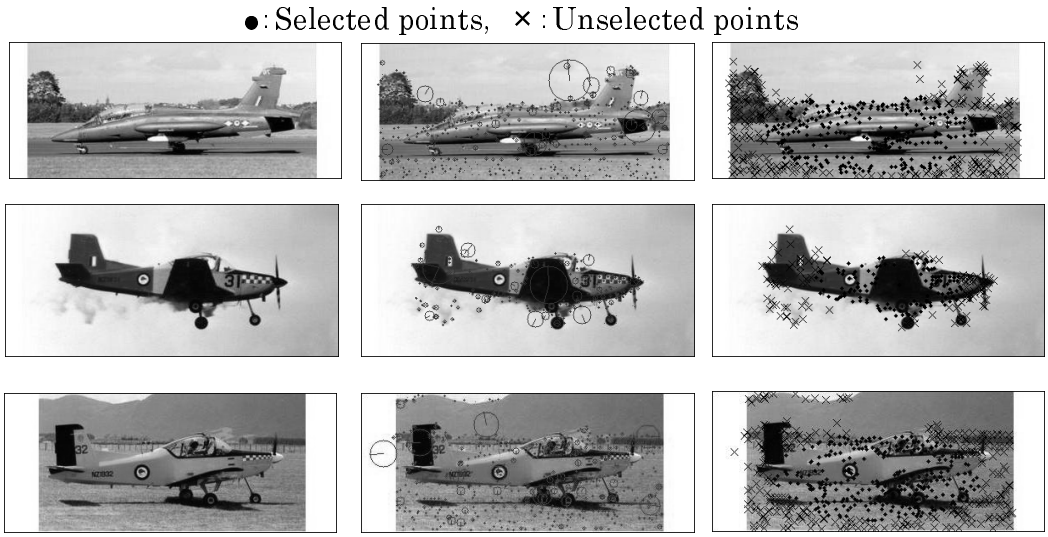


Figure 5.7. Results of selected SIFT feature points in same categories of Caltech-256.

5.7.2 Object classification

Figure 5.9 depicts a category map generated by CPNs for classifications of five categories: Airplane, Car-side, Motorbike, Face, and Leopards. We show images that mapped each unit and mapping regions in each category on the category map. Fig. 5.9 depicts that CPNs created categories for mapping to neighborhood units on the category map in each image with labels generated by ART-2. The Car-side and Leopards categories contain several labels. The Car-side category is mapped to neighborhood units. On the other hand, the Leopards category is divided into two regions.

Figure 5.10 depicts labels by ART-2 on 20-object classification. The bold line shows the number of images in 10 categories. The circles and squares portray images for which ART-2 confused labels on 10 and 20 categories, respectively. In the 10-object classification, ART-2 generated independent labels in all categories, although three images of two labels are confused. In the 20-object classification, independent labels of 19 categories are generated, except for the Zebra category that is confused of all images, although 16 images of five labels are confused.

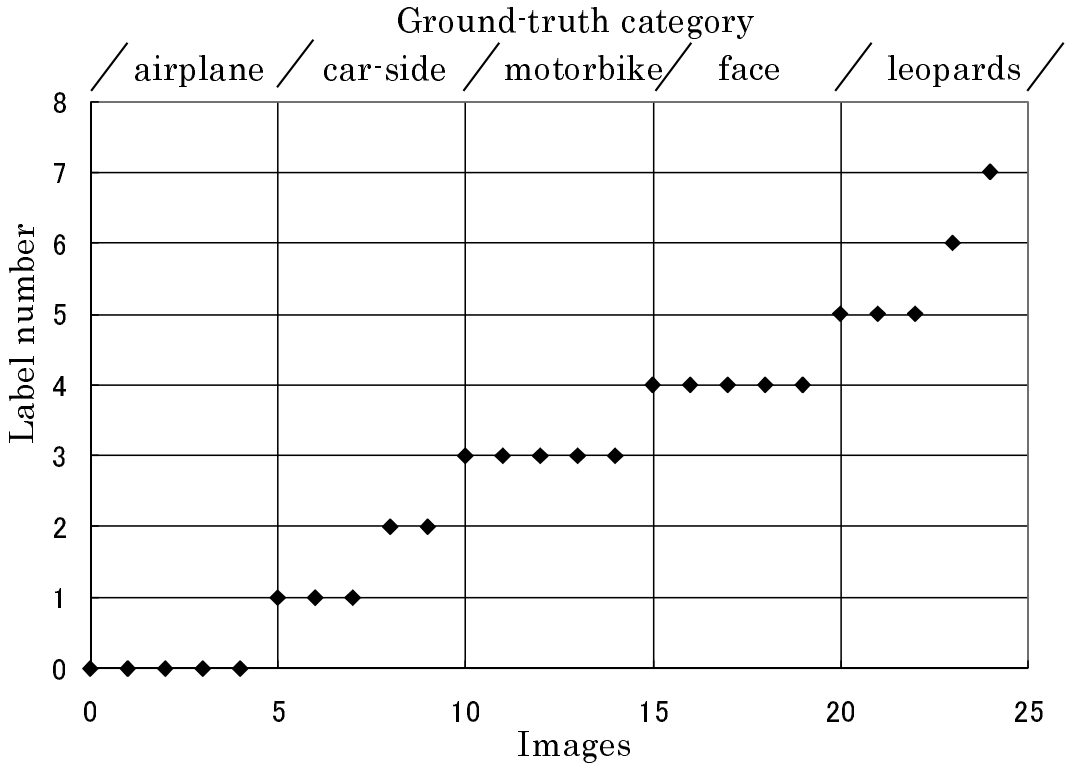


Figure 5.8. Results of formed labels using ART-2 at five categories.

Confusion of labels occurs often in images of Ketch, Hibiscus, and Guitar-pick categories. Although confused labels are restrained until 10-object classification, numerous confused labels are apparent in the 20-object classification.

Figure 5.11 depicts a category map generated by CPNs on 20-object classification. The names of categories and the number of images are shown on the category map. For all images in each category, 11 categories are mapped to neighborhood units. The CPNs created categories for mapping neighborhood units on the category map in images of each category by which ART-2 generated several labels. In addition, categories without their names are mapped images of different categories. Here, for quantitative evaluation of the classification performance of our method, we use the following recognition rate.

$$(RecognitionRate) = \frac{(CorrectData)}{(AllData)} \times 100. \tag{5.29}$$

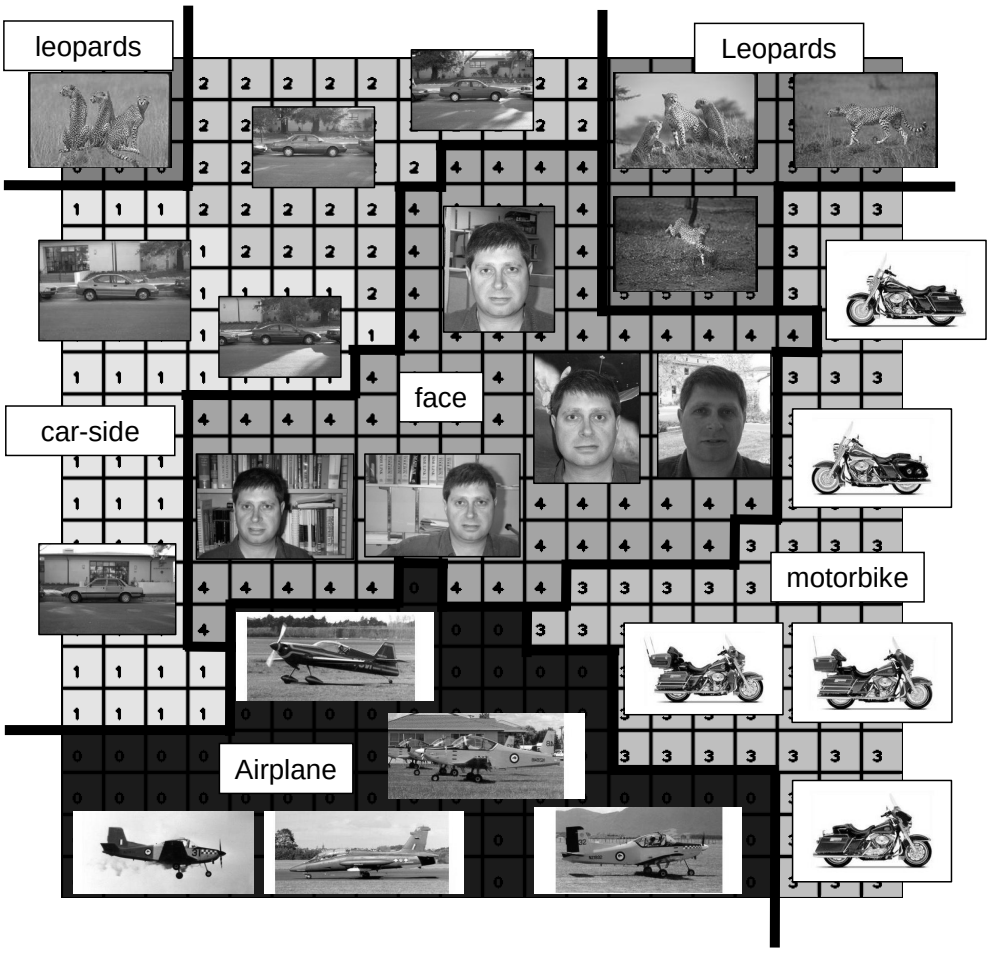


Figure 5.9. Result of category mapping using CPNs of five categories.

Table 5.5 portrays recognition rates in 5, 10, and 20 categories without OC-SVMs and with OC-SVMs for training and testing datasets. The recognition rates without OC-SVMs were, respectively, 84%, 70%, and 64% for training datasets and 76%, 30%, and 38% for testing datasets in 5, 10, and 20 categories. In our method, the recognition rates were, respectively, 96%, 94%, and 81% for training datasets and 76%, 42%, and 45% for testing datasets in 5, 10, and 20 categories. These results address the effectiveness to select SIFT feature points using OC-SVMs.

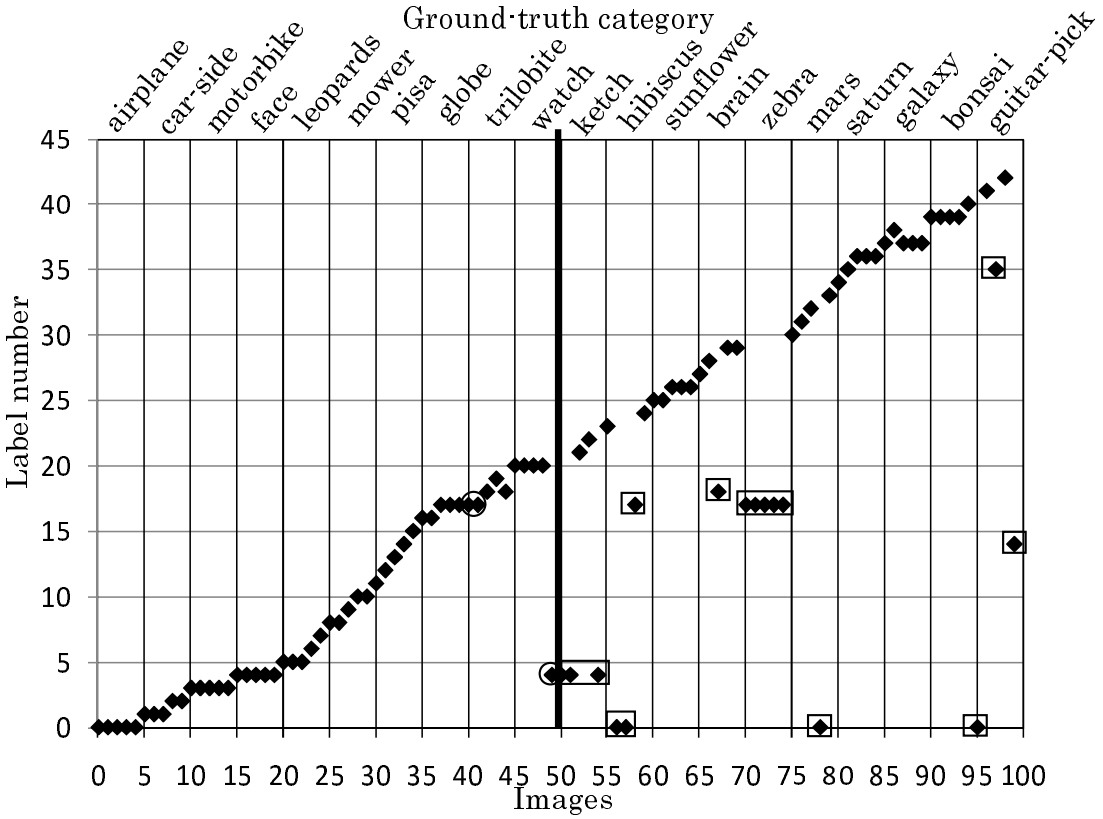


Figure 5.10. Results of formed labels using ART-2 at 10 and 20 categories.

5.8. Experimental results obtained using a mobile robot

In this section, we applied our method to object classification experiments using time-series images taken by a camera with movements of a robot. In this experiment, we evaluated our method for object classification of dynamic images because the target is time-series images according to the change of appearances. We built an original experimental environment to take images of datasets. This section presents the experimental environment and results of our method as the following.

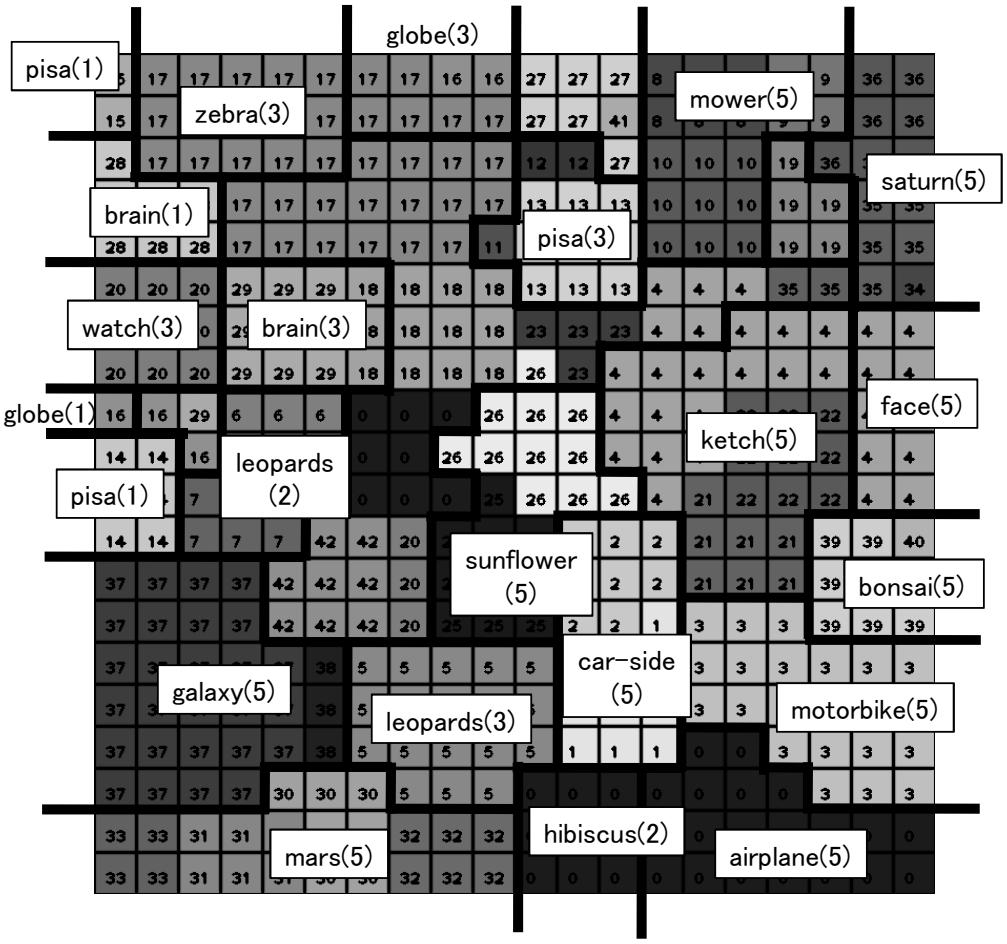


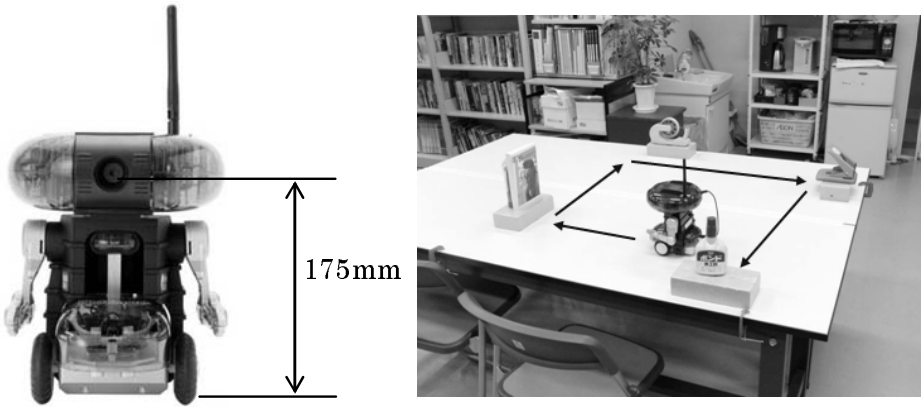
Figure 5.11. Result of a category map of 20 categories.

5.8.1 Specific object recognition obtained using a small mobile robot

Figure 5.12 portrays a home robot (NetTansor; Bandai Co. Ltd.) used in this experiment. Table 5.6 presents specificaions of the robot. The robot is 190 mm high, 160 mm long, and 160 mm wide. The camera specifications are the following: imaging device, 1/4 inch CMOS; image format, JPEG; resolution, 320 × 240 pixels; and frame rate, 15 fps. The moving environment is 1,150 × 1,150 mm surrounded by 300 mm high white walls. Fig. 5.13 shows the assignment of objects in the environment and the roughly determined goals of routes for

Table 5.5. Recognition rates of learning and testing datasets used in Caltech-256

	Without OC-SVMs		Our method	
	Learning	Testing	Learning	Testing
5 categories	84%	76%	96%	76%
10 categories	70%	30%	94%	44%
20 categories	64%	38%	81%	50%



(a) Robot (b) Experimental environment

Figure 5.12. Robot used for experiments (NetTansor; Bandai Co. Ltd.).

the robot. 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. Target objects were a hole punch (Object A), a plastic bottle of glue (Object B), a book (Object C), and a cellophane tape holder (Object D) shown in Fig. 5.13. For this experiment, we created datasets consisting of time-series images as shown in the behavior of Fig. 5.13 Datasets comprise RUN1 and RUN2 for which the robot runs twice around in the environment.

Figure 5.14 depicts results of selected feature points using OC-SVMs on four samples of time-series images taken by the robot. Our method can select feature points near objects against various appearance changes. In images of Object D, feature points of whole and a part of Object D are, respectively, selected distant

Table 5.6. Specifications of NetTansor

Body	Height	190 mm
	Width	160 mm
	Depth	160 mm
	Weight	960 g (include battery)
Camera	Imaging device	1/4 inch CMOS
	Resolution	320 × 240 pixels
	Frame Rate	15 fps
	Compression	JPEG

Table 5.7. Recognition rates of learning and testing datasets of time-series images

		Testing Datasets		
		RUN1	RUN2	Mean
Training	RUN1	98.1%	96.2%	
Datasets	RUN2	97.2%	98.8%	96.7%

from the object and near the object. In addition, feature points are selected not only of the object, but also around the object.

Figure 5.15 depicts labels generated by ART-2 on the experiment using time-series images of RUN1. The vertical and horizontal axes respectively represent labels of ART-2 and frames in images. The top parts portray ranges including objects and parts of the robot turned 90 deg as time-series images. In this result, 27 labels are generated from time-series images of 220 frames. In addition, the labels are more numerous than the target objects because labels are assigned to each image taken by the robot turned 90 deg from the four corners in the environment. Objects A, B, C, and D respectively generated 3, 2, 6, and 8 labels. Fig. 5.16 depicts a category map generated by CPNs. On the category map, we show mapping regions of images in each object. Each object classified with different labels with ART-2 is mapped to neighborhood units on the category map of CPNs shown in Fig. 5.16. In addition, images of turning of labels 3

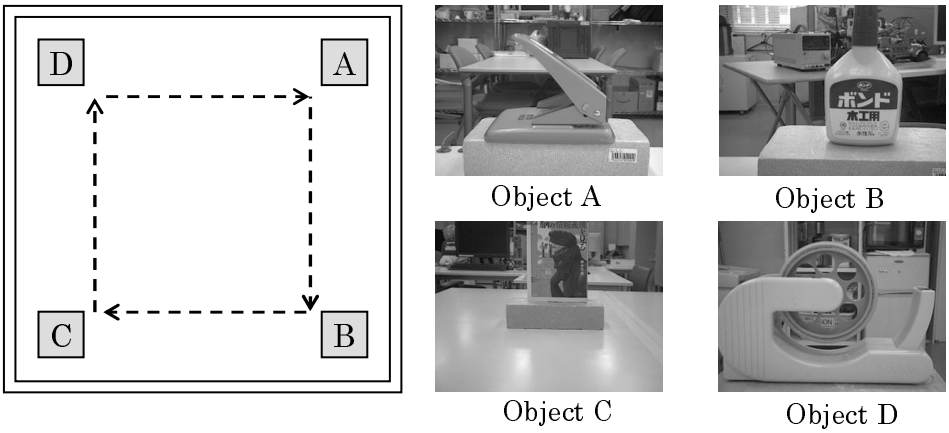


Figure 5.13. Four objects and the robot route used for our experiment.

and 4 are mapped around border units between categories. Table 5.7 portrays recognition rates for training and testing calculated using equation (5.29). This experiment evaluated recognition rates for all combinations of datasets of RUN1 and RUN2 for learning and testing. Underlined values are the recognition rates for training. In [118], 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 objects. Table 5.7 shows that recognition rates for training and testing datasets are more than 90%. Moreover, the mean recognition for testing datasets is 96.7%. In contrast, images of turning include misrecognitions and confused labels in each object.

5.8.2 Generic object recognition using an actual-size mobile robot

Based on the results of the questioner presented in Table 5.1, we evaluated our method as generic object recognition in an actual environment using an actual-size mobile robot. We used PaPeRo developed by NEC. This robot is a prototype for a personal robot used especially for child-care purposes [120]. Table 5.8 presents specifications of this robot related to its use for this experiment. The robot is 385 mm high, 282 mm long, and 251 mm wide. Comparison with NetTansor

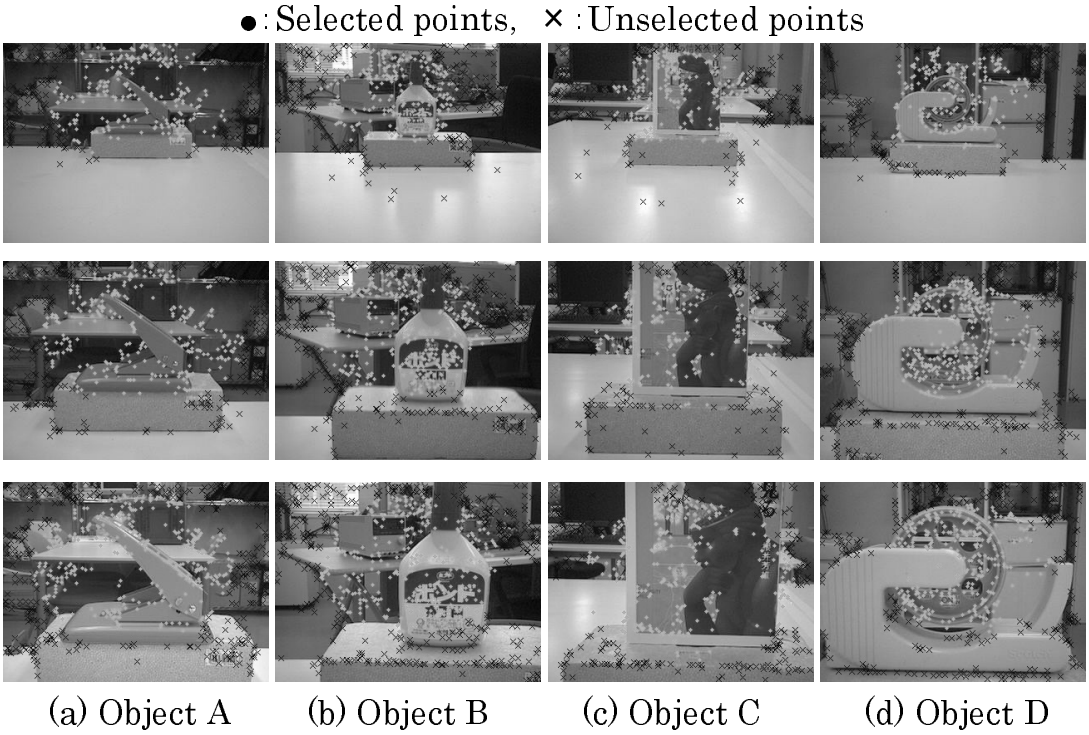


Figure 5.14. Results of selected SIFT feature points of time-series images.

shows that PaPeRo shows that it has sufficient capabilities to move on the floor. Moreover, servomotors are equipped for the drive system to control movements with high precision. We used one camera for monocular vision, but two cameras are mounted for stereo vision. The specifications of cameras are the following: imaging device, CCD; image format, JPEG; resolution, 320 × 240 pixels; and frame rate, 30 fps.

Figure 5.17 depicts the experimental environment. This room is a vacant room used as a professor's room. It contains a desk, a table, a sofa, and a cabinet. The floor is carpeted. In the room are a window and a blind. We closed the blind to avoid effects of sunlight while taking images through the experiment.

We selected target objects that can move portably. They were neither too large or too small compared with this robot, from the top group of the number of extracted categories by the questioner investigation presented in Table 5.1. Fig. 5.18 depicts target objects of four categories: Personal Computers (PCs),

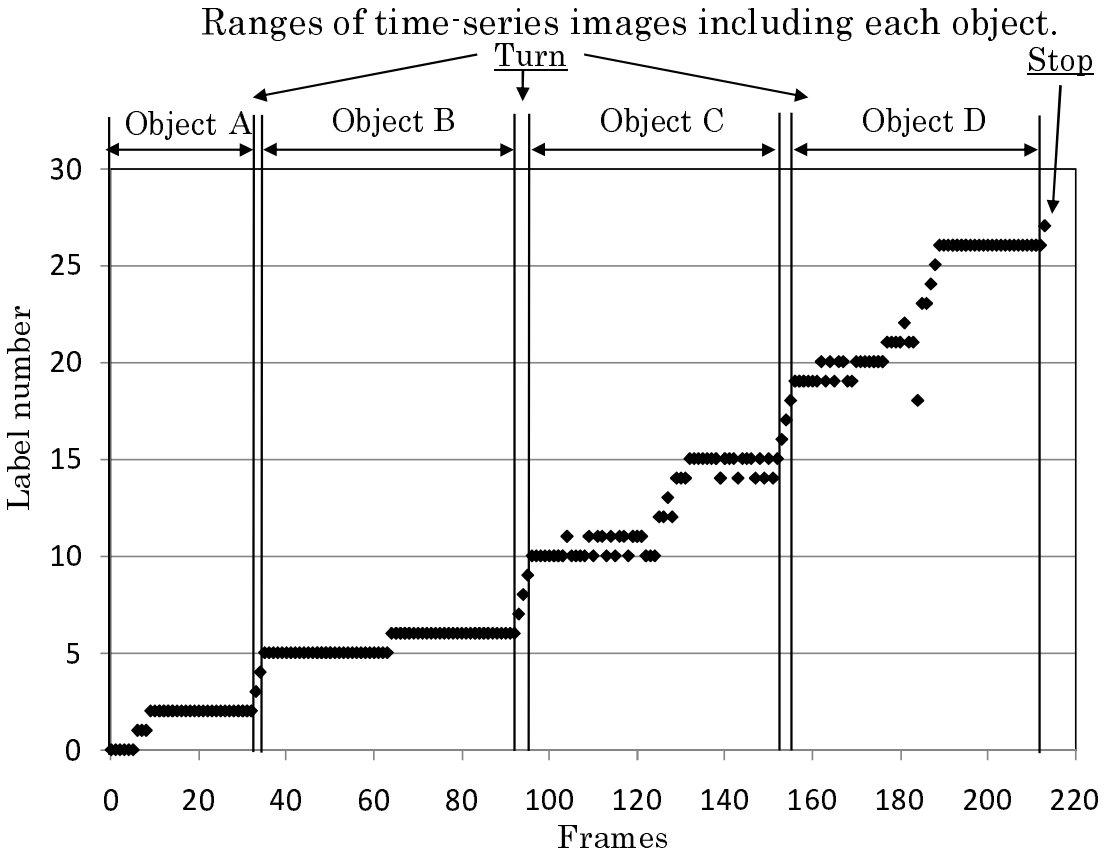


Figure 5.15. Results of labels created using ART-2 from time-series images.

Chairs, Robots, and Trash Bins (TBs). We selected medium-size desktop PCs to be placed under the desk. We used only OA chairs, although chairs of numerous types exist there. Comparison with other objects shows that robots are the smallest targets for this experiment. We selected TBs that have no patterns or labels on the surface. We used different objects in same category for testing.

Figure 5.19 portrays routes for the robot and assignments of objects for learning and testing. The robot moves the environment one round clockwise to use the behavior set consisting of forward movements and 90 deg turns. For learning, each object in the same category is assigned to extend lines of the routes. After one round, the robot movement is suspended to take images. Subsequently, we changed objects to the next category; the robot resumed movement to take im-

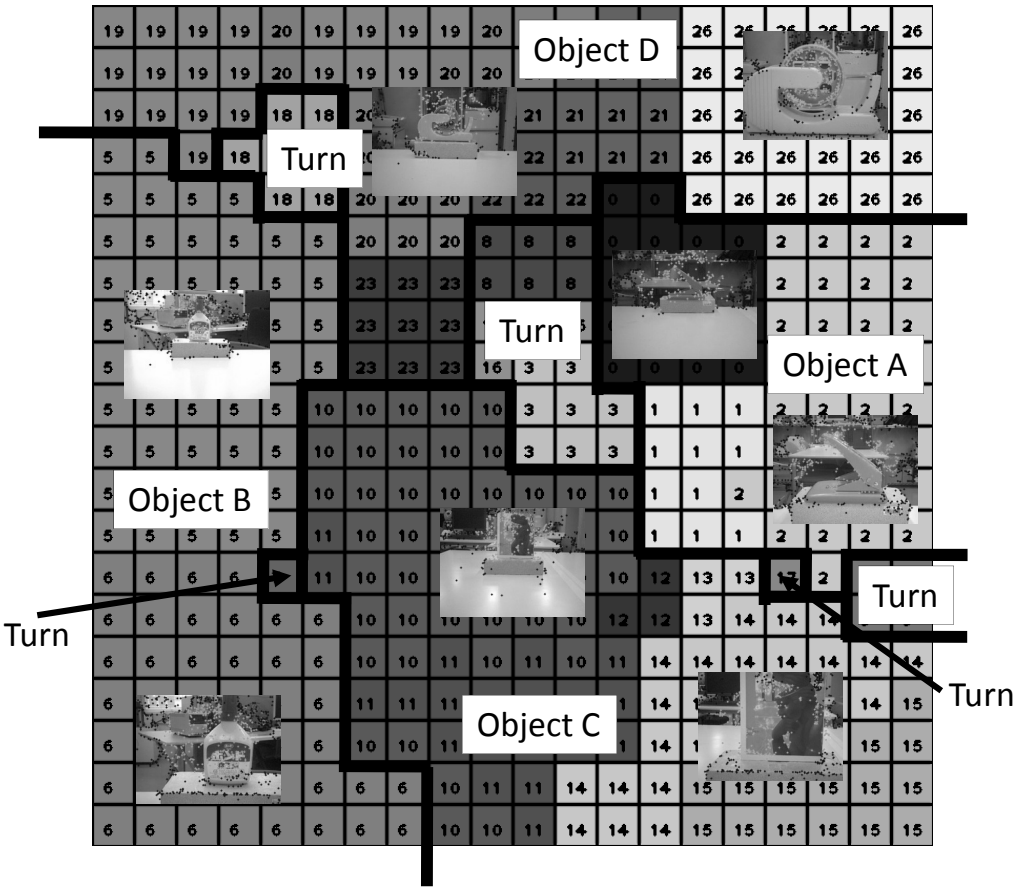
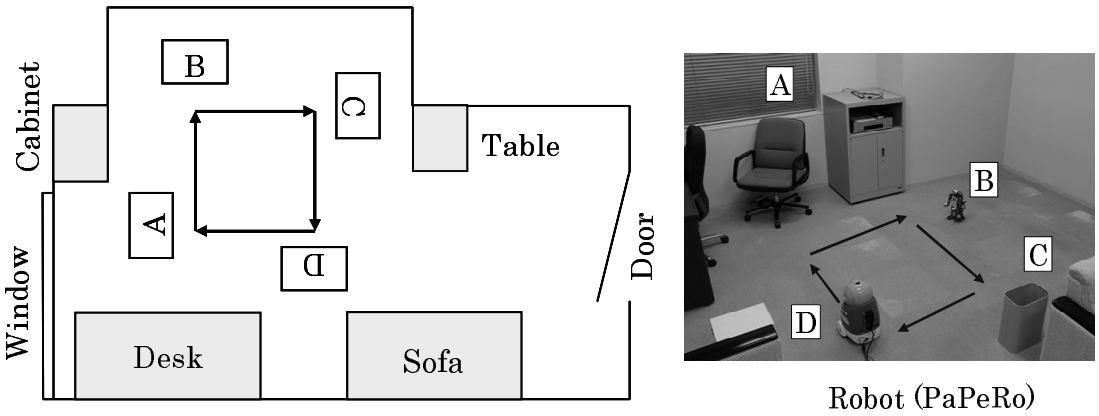


Figure 5.16. Mapping result of images on the category map of CPNs used in labels generated by ART-2.

ages. For testing, we assigned four different objects in each corner. The robot moved using the same behavior set. We took four datasets to change the positions of objects clockwise.

Figure 5.20 shows feature selection results of images with OC-SVMs. In this experiment, the range of moving for the robot is wide and the sizes of the target objects are various. Therefore, background feature points are selected. Moreover, classification target object robots are smaller than those of other objects. Feature points including background regions were extracted because the occupancy of background regions is larger than that of other images. The PC and TB feature points are few because shapes and components of these objects are simple.



(a) Layout of the room. (b) Landscape of experiment.

Figure 5.17. Experimental environment and an actual-size mobile robot for generic object recognition.

Therefore, feature points include background regions that were extracted because the occupancy of background regions is larger than those of other images.

Figure 5.21 portrays labels generated by ART-2. For this experiment, we set ρ to 0.5 to prevent redundant categories. Results show that ART-2 generated 38 labels from 320 frames of input images. We consider that the reason for generation of numerous labels is the diversity of appearance of objects, although we set a small value of ρ. Moreover, images of the robot turning are included in training datasets. In the last part of input frames, overlapping labels are apparent. In fact, ART-2 generated categories additionally from images to be changed objects. In this environment, four patterns of background regions are repeated. We consider that overlapping is caused by these background patterns to be memorized.

Figure 5.22 portrays the category map created by the labels. The category map size is 20 $\times$ 20 units. Categories are created for each independent region. However, these categories are separated into several regions. Using CPNs, 38 labels generated by ART-2 were integrated to 29 labels.

Table 5.9 presents test results for Datasets 1, 2, 3, and 4. Each dataset comprises 180 frames. The highest recognition accuracy is 53.3% in robots. In contrast, the lowest recognition accuracy is 21.9% in PCs. The recognition accuracy is decreased in Datasets 2 and 3. Especially, the recognition accuracy of

Table 5.8. Specifications of PaPeRo by NEC [121]

Body	Height	385 mm
	Width	282 mm
	Depth	251 mm
	Weight	6.5 kg (include battery)
Movement	Drive system	Servomotor × 2
	Speed	23 cm/s (maximum)
Camera	Imaging device	CCD Camera × 2
	Resolution	320 × 240 pixels
	Frame Rate	30 fps
	Compression	JPEG
Software Architecture	Operating system	Windows XP
	Development tool	RoboLabo

PCs is 0% in Dataset 2 and 4.4% in Dataset 3. Our method failed to recognize PCs and TBs. The recognition accuracy is decreased by this false recognition. The robot ran the same route from the start point under similar patterns of backgrounds, although objects were replaced in the test datasets. In this environment, the complexity of backgrounds at the routes of the forward movement after the start and the forward movement after the two sets of 90-deg turns is higher than that of the other two routes. In the latter route of complex backgrounds, images include the door near the entrance. Our method selected these SIFT features in the background region. Results for test datasets show that the same units on the category map are burst. This false recognition occurs in cases where the distance between the robot and objects is great. We consider that these burst patterns occur in response to patterns of background regions.

Our method selected foreground regions in an unsupervised manner using OC-SVMs. However, false recognition occurred in cases with small objects shown in an image with a background of high complexity. We consider that restriction of the distance between the robot and objects is necessary instead of using all frames for a target of training and recognition. The objective of our method is to

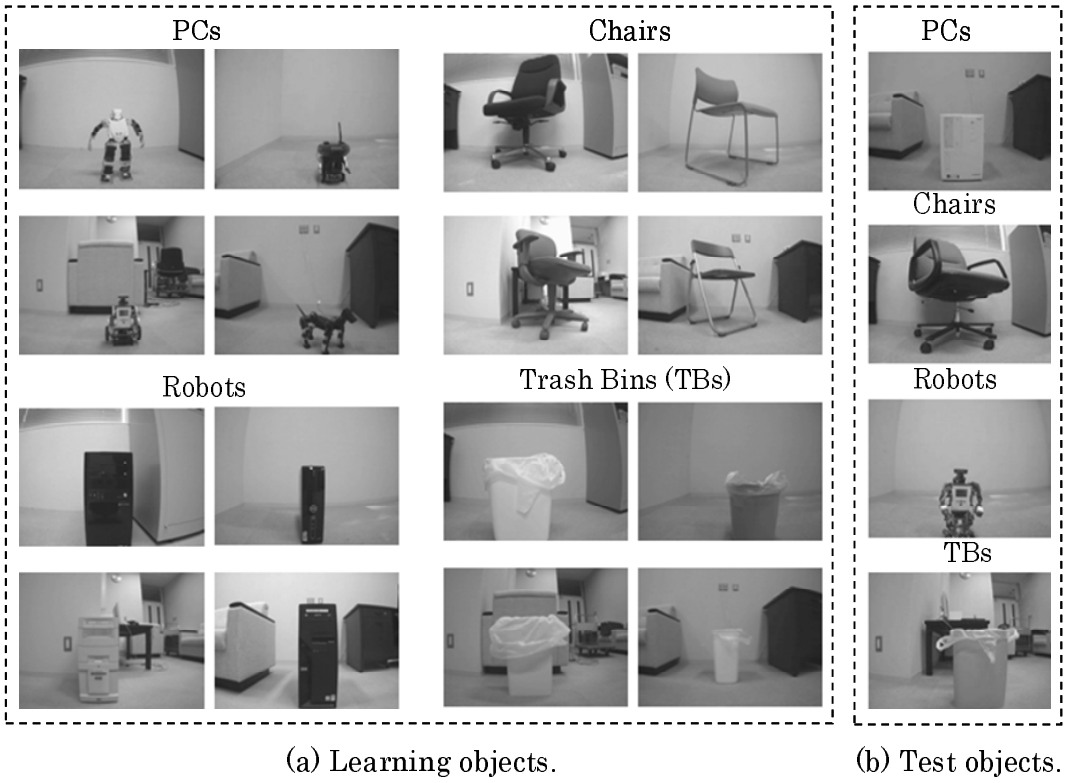


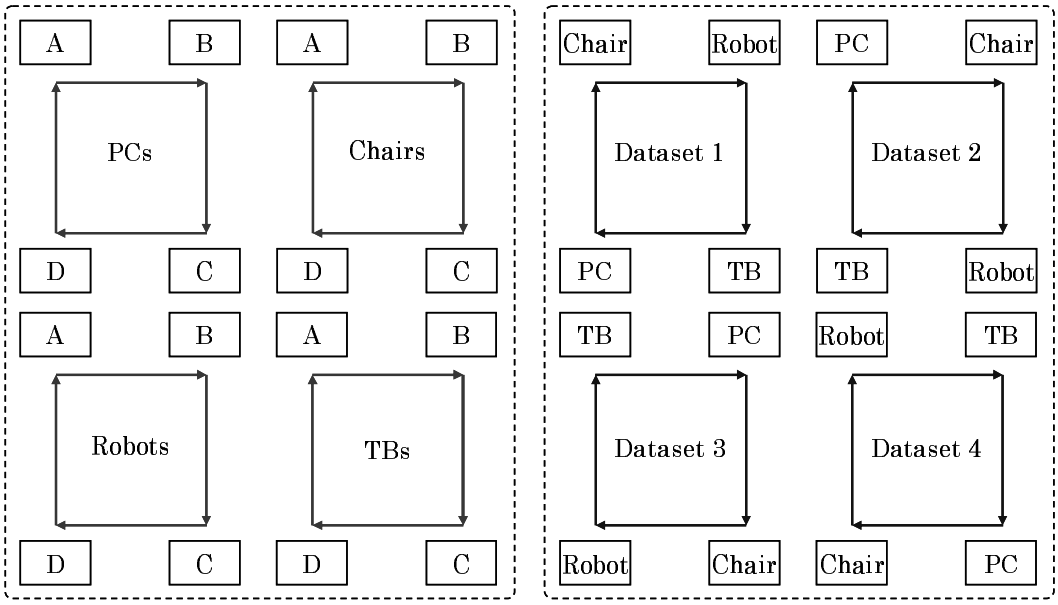
Figure 5.18. Classification target objects for learning and testing.

recognize one object in a scene image. We must extend our method to the target to classify multiple objects in one image.

5.9. Generation of robot behavior using GP

5.9.1 experimental environment

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.



(a) Learning routes and objects. (b) Test routes and objects.

Figure 5.19. Routes and assignments of objects for learning and testing.

- 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 × 10 blocks. One block corresponds to 115 × 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.

Figure 5.23 shows the assignment of objects in the environment and the roughly determined goals of routes for the robot. We generated behavior pro-

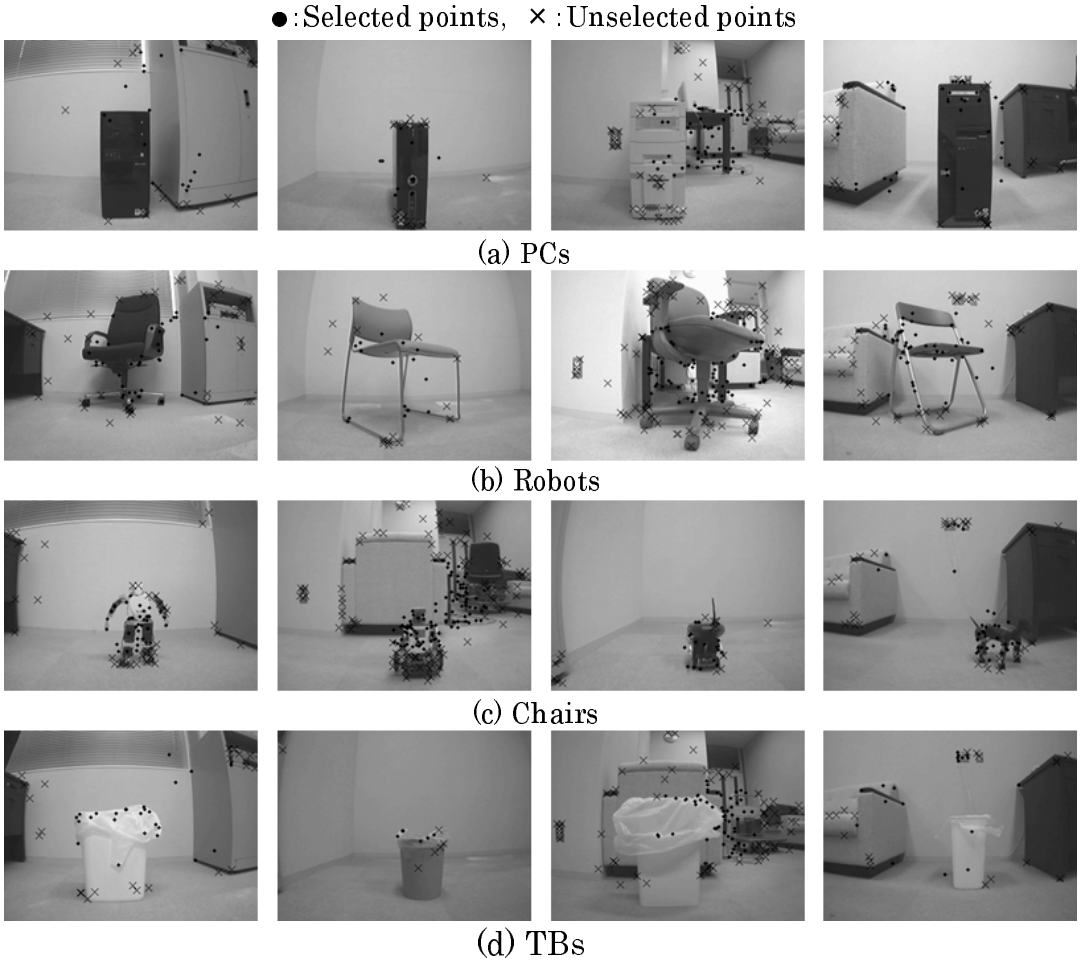


Figure 5.20. Selected feature points with OC-SVMs.

grams using GP. We set landmarks on both routes. Fig. 5.24 portrays a generated tree and its simulation result of the simple route along with walls shown in Fig. 5.23 (a). Fig. 5.24 presents a generated tree and its simulation result of the route that acquires various appearances around each object shown in Fig. 5.23 (b). 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

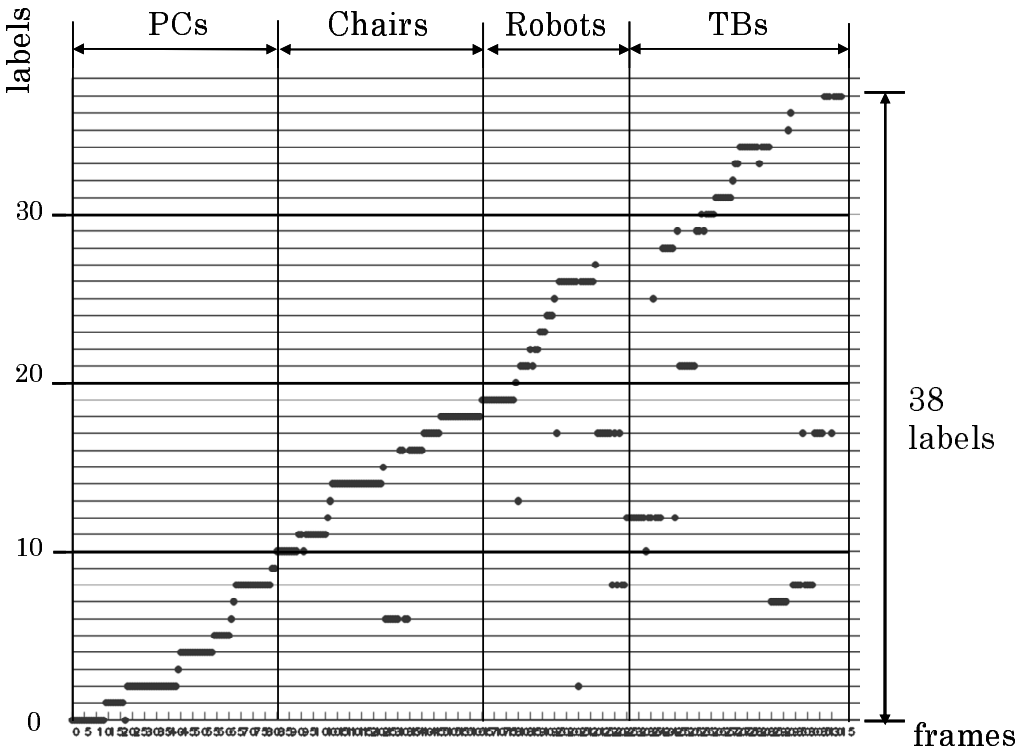


Figure 5.21. Results of labels created using ART-2.

by CPNs.

5.9.2 Classification results

In [118], 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 5.10 and 5.11 respectively present the target datasets and the recognition rate in each dataset for training and testing. The target datasets presented in Table 5.10 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

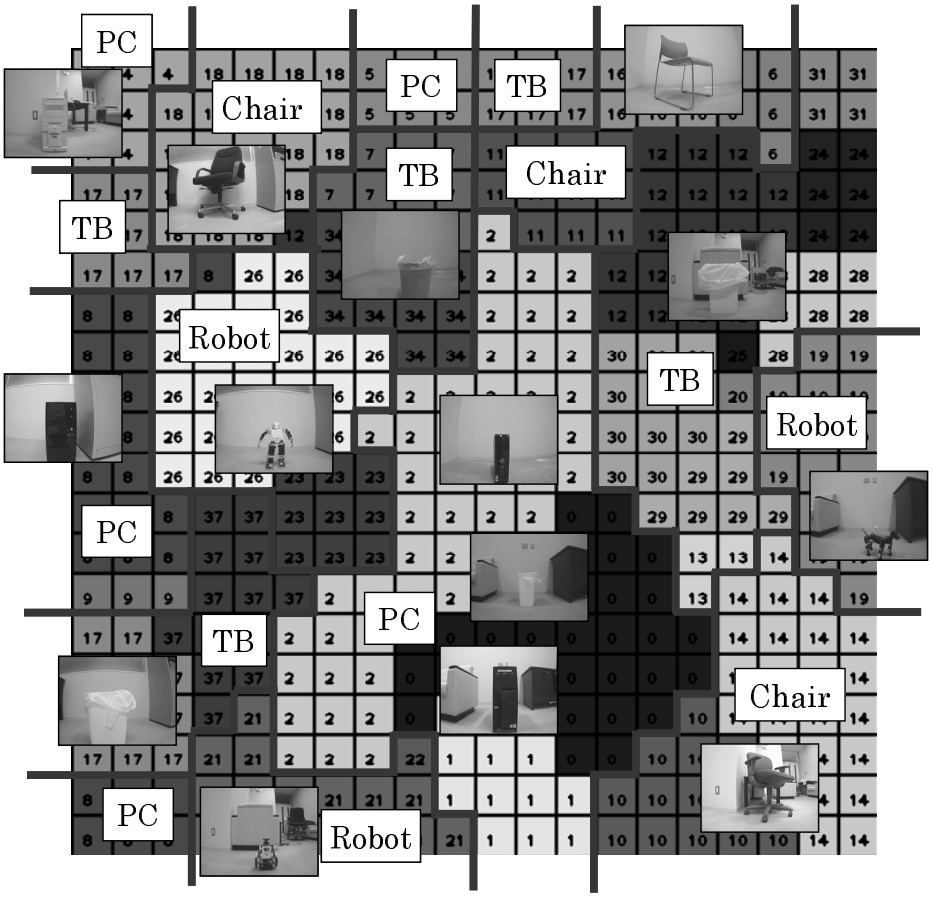


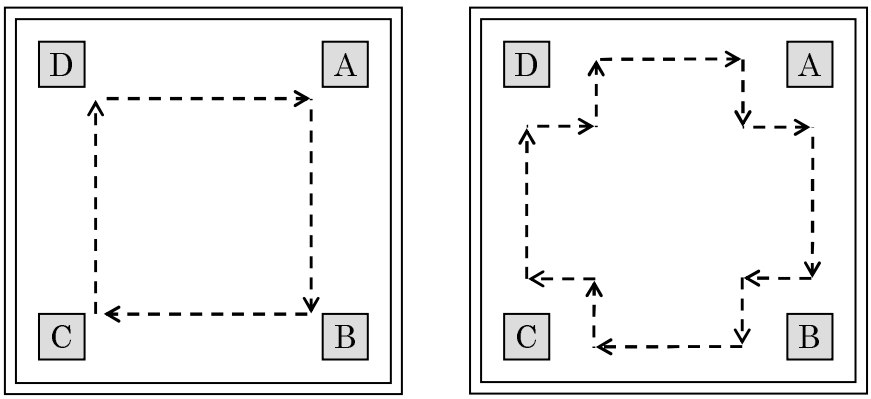
Figure 5.22. Mapping result of objects on the category map.

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%. 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.

Table 5.9. Recognition accuracy [%].

	Chair	Robot	TB	PC	Average
Dataset 1	63.6	60.7	24.1	40.0	48.2
Dataset 2	22.7	66.0	31.6	0	30.3
Dataset 3	31.8	20.5	90.2	4.4	33.9
Dataset 4	61.4	65.9	44.2	43.2	52.2
Average	44.9	53.3	47.5	21.9	41.2



(a) Route A (b) Route B

Figure 5.23. Experimental environment and robot routes.

5.10. Computational costs

The robot we used for this experiment has a wireless LAN system that enables it to communicate with a PC as an external computation environment. Therefore, we conducted calculations for learning and testing on a PC. Computational costs of our method are as follows.

- SOMs: 7 min per 1,000 frames
- SIFT and OC-SVMs: 11 min per 1,000 frames
- Training for ART-2 and CPNs: 45 s per 1,000 frames

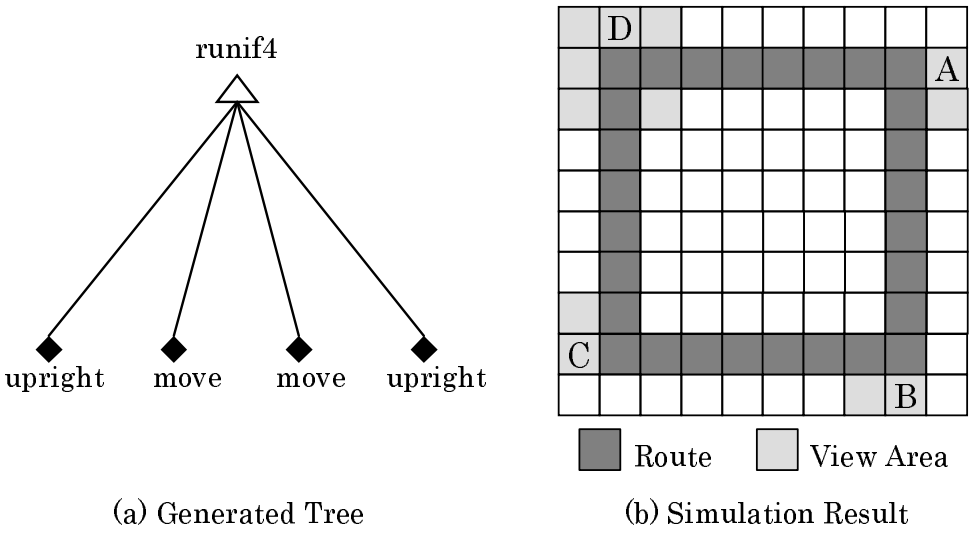


Figure 5.24. Generated tree and simulation result of Behavior A.

Table 5.10. Target datasets.

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

- Testing for CPNs: 0.15 s per frame

Some important parameters of our computational environment are Core 2 Duo 2.2 GHz CPU (Intel Corp.); 1.7 G bytes memory,; Vine Linux 4.2 OS,; and the Eclipse 3.4 development tool with OpenCV 1.0. The mean calculation cost for SIFT and OC-SVMs is 0.66 s per frame, although it depends on the number of feature points. The mean calculation cost for CPN testing is 0.15 s per frame, which enables calculation in real-time for the 30 fps input image.

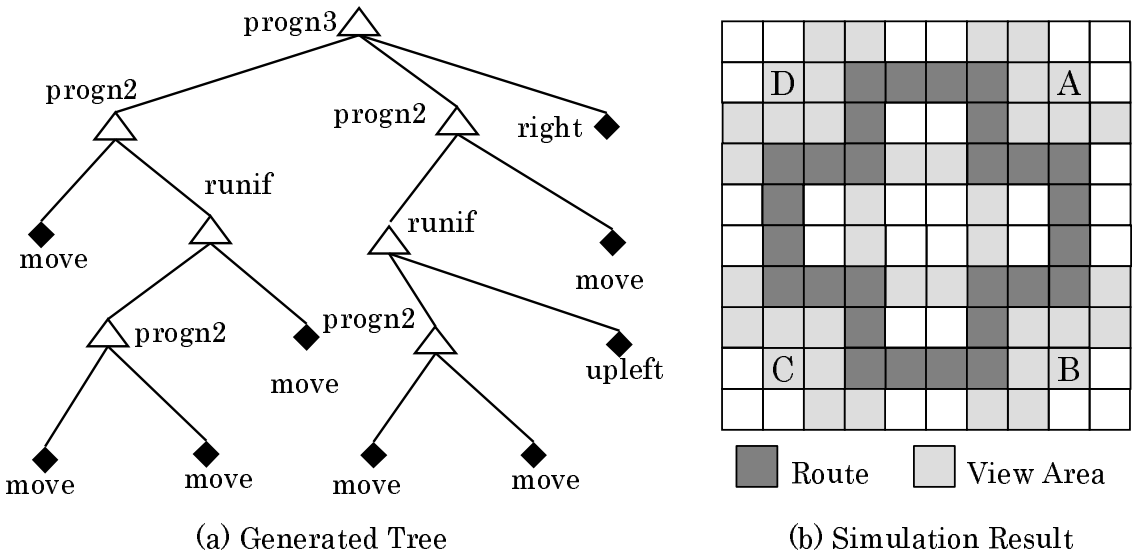


Figure 5.25. Generated tree and simulation result of Behavior B.

5.11. Discussion

Experimental results of Caltech-256 and time-series images of the robot show that OC-SVMs select feature points not only of the whole object, but also of the background and surrounding regions, and of partial objects. These results signify that OC-SVMs can select a region to concentrate specific information in an image, i.e. features that characterize an image, not feature points to be classified into the object and background.

Table 5.11. Recognition rates in each behavior [%].

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

Humans, when classifying objects, devote attention to a region that gathers information for characterizing an object, not the whole object. We consider that selection of SIFT features using OC-SVMs can describe features effectively for category formation to represent features and can thereby improve classification accuracy.

In the static object classification using Caltech-256, the accuracy of our method reached 81% for training and 50% for testing of 20-object classification. The unsupervised object classification method proposed by Chen et al. [95] showed respective performances of 76.9% for training and 67.4% for testing of 26-object classification for the Caltech dataset. The accuracy of our method is apparently inferior to that of the existing method. Nevertheless, our method can classify objects without previous setting of the number of categories. Therefore, our method is effective for application to problems that are known as challenging tasks of classification of categories whose ranges and types are unclear.

In this experiment, we observed 10 categories for which multiple labels are generated on ART-2. The images of Caltech-256 have no time-series factors, although ART-2 learns time-series changes of input data positively. Therefore, we inferred that ART-2 maintains no continuity of labels. For the relation of labels generated by ART-2 and a category map on CPNs, categories that maintained continued and non-continued labels are mapped respectively to neighborhood and separated units on the category map of CPNs.

In the dynamic object classification using time-series images of the robot, the accuracy shows high performance of better than 90% for training and testing datasets. This result means that our method can classify time-series images into categories used for characteristics of ART-2. Category formation for generic object recognition is necessary to classify categories for assigning one label to one category. However, category formation for robot vision is necessary to classify categories for assigning labels positively to changes in appearance with sensing in an environment.

We consider that ART-2 can learn changes in appearance positively for generation of labels. Nevertheless, the number of labels of ART-2 is greater because the appearance changes in the environment increase along with the behavior of turning 90 deg. The CPNs created categories in each object whose appearance

differs from that of neighboring units. 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. This result means that our method can represent the diversity of categories on category formation.

In this study, we are aiming at category formation to an actual environment for which the number of categories is mostly unknown. This experimental result demonstrates that our method can apply category formation such as that shown for this environment using ART-2 and CPNs for visualizing spatial relations of time-series images on the category map. We consider that this category formation method is 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.

5.12. Conclusion

This chapter presented an unsupervised method of SIFT feature points selection using OC-SVMs and category formation combined with incremental learning of ART-2 and self-mapping characteristic of CPNs. Our method enables feature representation that contributes to improved accuracy of classification for selecting feature points to concentrate characterized information of an image. Moreover, our method can visualize spatial relations of labels and integrate redundant and similar labels generated by ART-2 as a category map using self-mapping characteristics and neighborhood learning of CPNs. Therefore, our method can represent diverse categories.

Future studies must be conducted to develop methods to extract boundaries among clusters automatically and to determine a suitable number of categories from category maps of CPNs. Additionally, we will examine approaches that include generation of robot behavior for classification and recognition of objects.

Chapter 6

Conclusions and Future Studies

6.1. Conclusions

This thesis presented an unsupervised category formation method using SOMs to apply to robot vision. The primary experiment to evaluate basic characteristics of SOMs showed that generalization capabilities can be improved using expansion or compression of training data while retaining topological structures using topological mapping characteristics. We applied our method to classification problems of two types: Normal Mixtures dataset and Cone-Torus dataset. Compared with classification results, our method is superior to the conventional SVMs using original training data. Moreover, we applied our method to the face recognition problem under various illumination conditions using the Yale Face Dataset B. The error rate is decreased by 11.05 percent compared with the conventional SVMs and the generalization capability is improved using our method. Additionally, our method visualized the distribution of data points to be selected as SVs on the category map. We ascertained that SVs are distributed around the boundaries on the category map.

For practical uses in an actual environment, this thesis presented two applications using a mobile robot. The first application is scene category formation for position estimation and to create world image maps of a robot. We presented a method using hierarchical SOMs for a robot to estimate its location from changes in landscape that accompany shifts in viewpoint. We found that changes in landscape revealed by viewing image sequences could be extracted as concept patterns

by SOMs. Effective position information is acquired by making hierarchical SOMs and using it to consolidate position estimation concept patterns. We identified the following parameter for effectively characterizing the viewing image sequence from the standpoint of position estimation: the compression level, the number of viewpoint shifts, and the viewpoint shift angles. We evaluated the effect of shifts in position and direction while the robot was executing a trial journey on position estimation. The extent of these shifts established beyond a doubt that our method was robust. The results of an on-site field test of a robot system in a hospital with a convalescence ward confirmed the effectiveness of our method for practical use.

The second application is unsupervised category formation of generic objects to recognize and understand the environment where the robot moves. The primary experiment to evaluate basic characteristics of ART-2 showed that the proposed method selected suitable vigilance parameters according to classification granularity using orientation selectivity. Moreover, our method represented the appearance and disappearance of feature changes to detect dynamic, local, and topological changes of facial expression images. For object category formation, we presented an unsupervised method of SIFT feature points selection using OC-SVMs and category formation combined with incremental learning of ART-2 and self-mapping characteristic of CPNs. Our method enables feature representation that contributes to improved accuracy of classification for selecting feature points to concentrate characterized information of an image. Moreover, our method can visualize spatial relations of labels and integrate redundant and similar labels generated by ART-2 as a category map using self-mapping characteristics and neighborhood learning of CPNs. Therefore, our method can represent diverse categories of generic objects for actualizing advanced interaction between humans and robots.

6.2. Future Studies

Category formation for actual datasets obtained using a camera on a mobile robot yields promising results for application to generic object recognition and global position estimation using unsupervised neural networks of two types. However,

some important concerns remain. Herein, we discuss six topics for future studies that will improve work on this subject.

The first concern is automatic setting of the category map size. The suitable number of training data differs in each target problem. Actually, SOMs are a model of functional localization of the brain. For comparison with the number of neurons of the brain, the SOM units are quite few. Our method allocates categories to all units, although each category is formed partially. We will modify SOM algorithms to represent whole categories in a local part of the category map, similar to functional localization of the brain. As the neighborhood region of SOMs, we determined its size relative to the number of category map units. Regarding the relation to the model of the brain, it is desired to change the neighborhood region size adaptively for updating weights that are burst simultaneously, thereby reflecting the strength of the input stimulus. Moreover, we will examine hardware implementation of our method in the case of increasing calculation costs using large-scale category maps.

The second concern is evaluation of other response selectivity of features, such as wavelength, amplitude, frequency, and direction of motion. We consider that our method can represent these features similarly to human perception and present natural visual features. The expressive capability of ART-2 will be actualized using the response selectivity used in these features. For our evaluation experiments, we used short-term datasets, with 100 frames per subject. We will evaluate our method using long-term datasets that include variations with aging.

The third concern is to examine approaches that include generation of robot behavior. We used GP for generating behavior sets of wall-following, which is the basic behavior used to move in an unknown environment. However, it is a challenging task to obtain not only various objects, but also various view patterns. We must modify the fitness function to move closer to an object that is located distant from walls. The behavior generated by GP is a global behavior set. We will develop local behavior sets to obtain various view patterns to a specific object. For obtaining various view patterns, it is necessary to introduce approaches not only to feedback after movements, but also estimation of view patterns before movements. We will introduce Bayesian approaches that actualize high-probabilistic estimation using only a few datasets.

The fourth concern is comparison of accuracy with existing unsupervised methods for object recognition, e.g. pLSA, LDA, and DPM. In our method, we compared recognition performances with the state-of-the-art method of unsupervised learning in generic object recognition. We conducted no comparison experiment for object recognition and position estimation using a mobile robot. Open benchmark datasets obtained using robot vision systems are available for comparison with other methods. We will use these datasets to conduct comparative experiments.

The fifth concern is extension to multiple object detection and recognition. We will use co-occurrence mechanisms between foreground objects and background regions for image representation. In this study, our recognition target is a single object. For several objects with the same category in an image, it is possible to recognize feature representation of BoF. However, it is impossible for our method to recognize objects with different categories. We will extend OC-SVMs to multiple classifiers. For improvement of feature representation capability and recognition performance, we will use co-occurrence between foreground regions and background regions and between categories.

The last remaining concern is to extract boundaries among clusters automatically and to determine a suitable number of categories from category maps of CPNs. We will investigate methods to detect category boundaries and to determine the number of categories. With our method, users manually determine boundaries that correspond to semantic categories. As a result of CPNs, candidates of boundaries are extracted automatically from boundaries between labels. In contrast, actual categories are represented with several labels. Users must integrate labels corresponding to actual categories. We will develop a method to integrate labels as categories using the distribution of weights around neighborhood units. We will integrate labels to decide the number of categories that correspond to the number of perceptual categories.

References

[1] A. Zelinsky, Y. Matsumoto, J. Heinzmann, and R. Newman, "Towards Human Friendly Robots: Vision-based Interfaces and Safe Mechanisms", *Proc. of International Symposium on Experimental Robotics*, pp. 431–442, March 1999.

[2] K. Nakano, *Manufacturing of a Brain – Thinking about Biotechnology from a Making of a Robot –*, Kyoritsu Shuppan, Aug. 1995. (in Japanese)

[3] S. Amari and K. Toyama, *Encyclopedia of brain sciences*, Asakura Publishing, Apr.2000

[4] T. Kanade, "Computer Vision," *The Journal of the Institute of Electronics, Information, and Communication Engineers*, vol. 83, no. 1, pp. 32–37, Jan. 2000. (in Japanese)

[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), pp. 1–24, Nov. 2007. (in Japanese)

[6] A. Pinz, "Object Categorization," *Foundations and Trends in Computer Graphics and Vision*, vol. 1, no. 4, pp. 255–353, 2006.

[7] A. Bosch, X. Munoz, R. Marti, "A Review: Which is the Best Way to Organize/Classify Images by Content?," *Image and Vision Computing*, vol. 25, no. 6, pp. 778–791, June 2007.

[8] T. Kanade, "Robot Vision," *Journal of Information Processing Society of Japan*, vol.44, no.11, pp. 1130–1137, Dec. 2003. (in Japanese)

[9] H. Ishiguro, "Vision studies for robots supporting humans," *Journal of Robotics Society of Japan*, vol. 27, no. 6, pp. 592–595, July 2009. (in Japanese)

[10] G. Griffin, A. Holub, and P. Perona, "Caltech-256 Object Category Dataset," *California Institute of Technology Technical Report*, no. 7694, March 2007.

[11] V. N. Vapnik, *Statistical Learning Theory*, Wiley, 1998.

[12] R. E. Schapire, "The strength of weak learnability," *Machine Learning*, vol. 5, no. 2, pp. 197–227, 1990.

[13] D. Koller and N. Friedman, *Probabilistic Graphical Models*, The MIT Press, Aug. 2009.

[14] K. Doya, "What are the computations of the cerebellum, the basal ganglia, and the cerebral cortex," *Neural Networks*, vol. 12, pp. 961–974, 1999.

[15] T. Kohonen, "Self-organized formation of topologically correct feature maps," *Biological Cybernetics*, vol. 43, no. 1, pp. 59–69, 1982.

[16] K. Tasdemir and E. Merenyi, "Exploiting Data Topology in Visualization and Clustering of Self-Organizing Maps," *IEEE Trans. Neural Networks*, vol. 20, no. 4, pp. 549–562, Apr. 2009.

[17] C. H. Chang, P. Xu, R. Xiao, and T. Srikanthan, "New adaptive color quantization method based on self-organizing maps," *IEEE Trans. Neural Networks*, vol. 16, no. 1, pp. 237–249, Jan. 2005.

[18] X. Tan, S. Chen, Z.-H. Zhou, and F. Zhang, "Recognizing partially occluded, expression variant faces from single training image per person with SOM and soft k-NN ensemble," *IEEE Trans. Neural Networks*. vol. 16, no. 4, pp. 875–886, Jul. 2005.

[19] A. Rauber, "LabelSOM: On the Labeling of Self-Organizing Maps," *Proc. International Joint Conf. on Neural Networks*, pp. 1–6, 1999.

[20] S. Grossberg, "Adaptive pattern classification and universal recoding, II: Feedback, expectation, olfaction, and illusions," *Biological Cybernetics*, vol. 23, no. 4, pp. 187–202, Dec. 1976.

[21] A. Kaylani, M. Georgiopoulos, M. Mollaghasemi, G. C. Anagnostopoulos, C. Sentelle, and M. Zhong, "An Adaptive Multiobjective Approach to Evolving ART Architectures," *IEEE Trans. Neural Networks*. vol. 21, no. 4, pp. 529–550, Feb. 2010.

[22] J. Holland, *Adaptation in Natural and Artificial Systems*, The MIT Press, Apr. 1992.

[23] 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.

[24] R. H. Nielsen, "Counterpropagation Networks," *Applied Optics*, vol. 26, no. 23, pp. 4979–4984, Dec. 1987.

[25] J. McQueen, "Some methods for classification and analysis of multivariate observations," *Proc. Fifth Berkeley Symposium on Mathematical Statistics and Probability*, pp. 281–297, 1967.

[26] T. Hofmann, "Probabilistic latent semantic indexing," *Proc. 22nd Annual International ACM SIGIR Conf. Research and Development in Information Retrieval*, pp. 50–57, 1999.

[27] D. M. Blei, A. Y. Ng, and M. I. Jordan , "Latent Dirichlet Allocation," *Journal of Machine Learning Research 3*, pp. 993–1022, 2003

[28] T. S. Ferguson, "A Bayesian Analysis of Some Nonparametric Problems," *The Annals of Statistics*, vol. 1, no. 1, pp. 209–230, 1973.

[29] Y. Chung and D. B. Dunson, *The local Dirichlet process*, Springer (online), Jan. 2009.

[30] D. Brugger, M. Bogdan, and W. Rosenstiel, "Automatic Cluster Detection in Kohonen's SOM," *IEEE Trans. Neural Networks*, vol. 19, no. 3, pp. 442–459, Mar. 2005.

[31] A. S. Georghiades, P. N. Belhumeur, and D. J. Kriegman, "From Few to Many: Illumination Cone Models for Face Recognition under Variable Lighting and Pose," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 23, no. 6, pp. 643–660, June 2001.

[32] A. Oliva and A. Torralba, "Building the Gist of a Scene: The Role of Global Image Features in Recognition," *Progress in Brain Research: Visual perception*, vol. 155, pp. 23–36, 2006.

[33] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91-110, 2004.

[34] B. Scholkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, and R. C. Williamson, "Estimating the Support of a High Dimensional Distribution," *Neural Computation*, vol. 13, no. 7, pp. 1443–1471, 2001.

[35] H. Kita, "Generalization of Neural Networks," *Systems, Control and Information*, vol. 36, no. 10, pp. 625–633, 1992. (in Japanese)

[36] L. Holmstrom and P. Koistinen, "Using additive noise in back-propagation training," *IEEE Trans. Neural Networks*, vol. 3, no. 1, pp. 24–38, Jan. 1992.

[37] G. N. Karystions and D. A. Pados, "On Overfitting, Generalization and Randomly Expanded Training Set," *IEEE Trans. Neural Networks*, vol. 11, no. 5, pp. 1050–1057, Sep. 2000.

[38] D. Sarkar, "Randomness in Generalization Ability: A Source to Improve It," *IEEE Trans. Neural Networks*, vol. 7, no. 3, pp. 676–685, May 1996.

[39] C. Lee and L. D. A. Landgrebe, "Decision boundary feature extraction for neural networks," *IEEE Trans. Neural Networks*, vol. 8, no. 1, pp. 75–83, Jan. 1997.

[40] K. Hara and K. Nakayama, "Training Data Selection Method for Generalization by Multilayer Neural Networks," *IEICE Trans. Fundamentals of Electronics, Communications and Computer Sciences*, vol. E81-A, no. 3 pp. 374–381, Mar. 1998. (in Japanese)

[41] N. Tanaka, T. Koreyeda, T. Inoue and K. Kajitani, "A Method of Learning BP Network by Expanding the Distribution of Category," *Trans. Institute of Electronics, Information and Communication Engineers (D–II)*, vol. J81–D–II, no. 2, pp. 293–300, Feb. 1998. (in Japanese)

[42] M. Kayama and S. Abe, "Training Neural Net Classifier for Improving Generalization Capability," *Trans. Institute of Electronics, Information and Communication Engineers (D–II)*, vol. J76–D–II, no. 4, pp. 863–872, Apr. 2001. (in Japanese)

[43] D. Chakraborty and N. R. Pal, "A Novel Training Scheme for Multilayered Perceptrons to Realize Proper Generalization and Incremental Learning," *IEEE Trans. Neural Networks*, vol. 14, no. 1, pp. 1–14, Jan. 2003.

[44] H. Drucker and Y. L. Cun, "Improving generalization performance using double backpropagation," *IEEE Trans. Neural Networks*, vol. 3, no. 6, pp. 991–997, Nov. 1992.

[45] Y. Matsunaga, K. Murase, O. Yamakawa and M. Tanifuji, "A Modified Back-Propagation Algorithm that Automatically Removes Redundant Hidden Units by Competition," *Trans. Institute of Electronics, Information and Communication Engineers (D–II)*, vol. J79–D–II, no. 3, pp. 403–412, Mar. 1996. (in Japanese)

[46] M. Ishii and I. Kumazawa, "Introduction of Linear Constraints on Weight Representation of Multilayer Networks for Generalization and Application for Character Recognition," *Trans. Institute of Electronics, Information and Communication Engineers (D–II)*, vol. J84–D–II, no. 3, pp. 541–548, Mar. 2001. (in Japanese)

[47] M. Tonomura and K. Nakayama, "An Internal Information Optimum Algorithm for Multilayer Perceptrons and Its Generalization Analysis," *Trans. Institute of Electronics, Information and Communication Engineers (D–II)*, vol. J84–D–II, no. 5, pp. 830–842, May 2001. (in Japanese)

[48] S. Abe, M. Kayama and H. Takenaga, "Acceleration of Learning Improvement of Generalization Ability for Pattern Classification Networks," *Trans.*

Institute of Electronics, Information and Communication Engineers (D–II), vol. J76–D–II, no. 3, pp. 647–652, Mar. 1993. (in Japanese)

[49] J. S. N. Jean and J. Wan, "Weight smoothing to improve network generalization," *IEEE Trans. Neural Networks*, vol. 5, no. 5, pp. 752–763, Sep. 1994.

[50] H. Takizawa, T. Nakajima, H. Kobayashi and T. Nakamura, "An Active Learning Algorithm Based on Existing Training Data," *IEICE Trans. Information and Systems*, vol. E83–D, no. 1, pp. 90–99, Jan. 2000.

[51] S. Vijayakumar and H. Ogawa, "Improving Generalization Ability through Active Learning," *IEICE Trans. Information and Systems*, vol. E82–D, no. 2, pp. 480–487, Feb. 1999.

[52] K. Tsuda, "Overview of Support Vector Machine," *The Journal of the Institute of Electronics, Information, and Communication Engineers*, vol. 83, no. 6, pp. 460–466, Jul. 2000. (in Japanese)

[53] B. D. Ripley, *Pattern Recognition and Neural Networks*, Cambridge University Press, Jan. 1996.

[54] L. I. Kuncheva, ıCombining Pattern Classifiers: Methods and Algorithms, Wiley, Jul. 2004.

[55] T. Okabe and Y. Sato, "Support Vector Machines for Object Recognition under Varying Illumination," *Information Processing Society of Japan*, vol. 44, no. SIG_5(CVIM_6), pp.22–29, Apr. 2003. (in Japanese)

[56] K.C. Lee, J. Ho, and D. Kriegman, "Acquiring Linear Subspaces for Face Recognition under Variable Lighting," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 27, no. 5, pp. 684–698, May 2001.

[57] A. Nagata, T. Okazaki, S. Choi, and H. Harashima, "Basis Generation and Description of Facial Images Using Principal-Component Analysis," *Trans. Institute of Electronics, Information and Communication Engineers (D–II)*, vol. J79–D2, no. 7, pp. 1230–1235, Jul. 1996. (in Japanese)

[58] D. E. Rumelhart, G. E. Hinton and R. J. Williams, "Learning Representations by Back-Propagating Errors," *Nature*, vol. 323, pp. 533–536, 1986.

[59] S. Yamada and M. Murota, "Recognition of surroundings by mobiel robots from action sequences using self-organizing maps," *Journal of Robotics Society of Japan*, vol. 17, no. 6, pp. 855–864, 1999. (in Japanese)

[60] H. Takeda, C. Facchinetti, and J.C. Latombe, "Planning the motions of a mobile robot in a sensory uncertainty field," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 16, no. 10, pp. 1002–1017, 1994.

[61] T. Kanbara, J. Mirua, and Y. Shirai, "Selection of efficient landmarks for an autonomous vehicle," Proc. Intelligent Robots and Systems, vol. 2, pp. 1332–1338, July 1993.

[62] M. Kagami, M. Inaba, and H. Inoue, "World map generation using real-world landmarks and the effectiveness of this technique," *Robotics Society of Japan Arts and Sciences lectures, reprint collection*, no. 1, pp. 405–406, Nov. 1994. (in Japanese)

[63] S. Handa, K. Tsubonouchi, and S. Aburate, "A method for self-localization of mobile robots based on identification of color images," *Journal of Robotics Society of Japan*, vol. 8, no. 6, pp. 641–651, Dec. 1990. (in Japanese)

[64] T. Maeda, A. Ishiguro, and S. Tsuji, "Investigation of a prior unknown circumstances based on ominidirectional images," *Research Reports on Sensory Processing*, vol. 95, no. 5, pp. 73–80, 1995. (in Japanese)

[65] Y. Matsumoto, M. Inaba, and H. Inoue, "Viewing image sequences based on representing the road traveled for use in navigtion," *Journal of Robotics Society of Japan*, vol. 15, no. 2, pp. 236–242, March 1997. (in Japanese)

[66] T. Nishimura, S. Nozaki, and R. Oka, "Spotting-based global localization by a mobile robot based on non-monotonic continuous DP using a time series of images," *Trans. Institute of Electronics, Information and Communication Engineers (D–II)*, vol. J81–D–II, no. 8, pp. 1876–1884, Aug. 1998. (in Japanese)

[67] S. Maeda, G. Kuno, and Y. Shirai, "Global position verification of robot motion based on eigenspace method," *Trans. Institute of Electronics, Information and Communication Engineers (D–II)*, vol. 81, no. 8, pp. 1876–1884, Aug. 1998. (in Japanese)

[68] G. von Wichert, "Mobile robot localization using a self-organized visual environment representation," *Robotics and Autonomous Systems*, vol. 25, pp. 185–194, Nov. 1998.

[69] K. Syouya, Y. Yagi, and M. Yachida, "Using ominidirectional vision sensors for simultaneous estimation of layouts of the static landscape in a mobile environment and self-localization," *Journal of Robotics Society of Japan*, vol. 17, no. 3, pp. 432–468, March 1999. (in Japanese)

[70] K. Oikawa and T. Tsuchiya, "A method of acquiring world images and navigation for a behavior-based autonomous mobile robot," *Journal of Robotics Society of Japan*, vol. 16, no. 1, pp. 65–73, 1998. (in Japanese)

[71] J. Lampinen and E. Oja, "Clustering properties of hierarchical self-organizing maps," *Journal of Mathematical Imaging and Vision*, pp. 261–272, 1992.

[72] Medical Treatment Law, Article 1, Section 5. (in Japanese)

[73] M. Pantic and L. J. M. Rothkrantz, "Automatic Analysis of Facial Expressions: The State of the Art," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 22, no. 12, pp. 1424–1445, Dec. 2000.

[74] M. J. Lyons, J. Budynek, and S. Akamatsu, "Automatic Classification of Single Facial Images," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 21, no. 12, pp. 1357–1362, Dec. 1999.

[75] M. Nishiyama, H. Kawashima, T. Hirayama, and T. Matsuyama, "Facial Expression Representation based on Timing Structures in Faces," *IEEE Int'l. Workshop on Analysis and Modeling of Faces and Gestures*, pp. 140–154, 2005.

[76] P. Ekman and W. V. Friesen, *Unmasking the Face: A Guide to Recognizing Emotions from Facial Clues*, Malor Books, 2003.

[77] G. Donato, M. S. Bartlett, J. C. Hager, P. Ekman, and T. J. Sejnowski, "Classifying Facial Actions," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 21, no. 10, pp. 974–989, Oct. 1999.

[78] D. H. Hubel and T. N. Wiesel, "Functional Architecture of Macaque Monkey Visual Cortex," *Proc. Royal Soc. B*, vol. 198, pp. 1–59, 1978.

[79] C. Liu, "Gabor-Based Kernel PCA with Fractional Power Polynomial Models for Face Recognition," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 21, no. 10, pp. 974–989, Oct. 1999.

[80] T. S. Lee, "Image representation using 2D Gabor wavelets," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 18, no. 10, pp. 959–971, Oct. 1996.

[81] A. K. Jain and S.K. Bhattacharjee, "Address block location on envelopes using Gabor filters: supervised method," *Proc. Conf. Pattern Recognition*, vol. II, pp. 264–267, Sep. 1992.

[82] C. P. Papageorgiou, M. Oren, and T. Poggio, "A general framework for object detection," *Proc. Conf. Computer Vision*, pp. 555-562, 1998.

[83] T. Randen and J. H. Husoy, "Filtering for texture classification: a comparative study," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 21, no. 4, pp. 291–310, Apr. 1999.

[84] D. Shan and R. K. Ward, "Statistical Non-Uniform Sampling of Gabor Wavelet Coefficients for Face Recongnition," *Proc. Conf. Acoustics, Speech, and Signal Processing*, vol. 2, pp. 73–76, 2005.

[85] L. Chengjun and W. Harry, "A Gabor Feature Classifier for Face Recognition," *Proc. Conf. Computer Vision*, vol. 2, pp. 270–275, 2001.

[86] G. Dissanayake, P. Newman, S. Clark, H.F. Durrant-Whyte, and M. Csorba, "An experimental and theoretical investigation into simultaneous

localisation and map building (SLAM)," *Lecture Notes in Control and Information Sciences: Experimental Robotics VI*, Springer, 2000.

[87] M. Cummins and P. Newman, "FAB-MAP: Probabilistic Localization and Mapping in the Space of Appearance," *The International Journal of Robotics Research*, vol. 27, no. 6, pp. 647–665, June 2008.

[88] I. Biederman, "Human image understanding: recent research and a theory," *The second workshop on Human and Machine Vision II*, vol. 13, pp. 13–57, 1986.

[89] K. Barnard, P. Duygulu, N. D. Freitas, D. Forsyth, D. Blei, and M. Jordan, "Matching Words and Pictures," *Journal of Machine Learning Research*, vol. 3, pp. 1107–1135, 2003.

[90] C. H. Lempert, M. B. Blaschko, and T. Hofmann, "Beyond Sliding Windows: Object Localization by Efficient Subwindow Search," *Proc. Conf. Computer Vision and Pattern Recognition*, pp. 1–8, 2008.

[91] K. Suzuki, T. Matsukawa, and T. Kurita, "Bag-of-features car detection based on selected local features using Support Vector Machine," *Technical report of IEICE. PRMU*, vol. 108(484), pp. 7–12, 2009. (in Japanese)

[92] G. Csurka, C. R. Dance, L. Fan, J. Willamowski, and C. Bray, "Visual categorization with bags of keypoints," *Proc. European Conf. Computer Vision*, pp. 59–74, 2004.

[93] J. Sivic, B. C. Russell, A. A. Efros, A. Zisserman, and W. T. Freeman, "Discovering Objects and their Localization in Images," *Proc. Conf. Computer Vision*, pp. 370–377, 2005.

[94] L. Zhu, Y. Chen, and A. Yuille, "Unsupervised Learning of Probabilistic Grammar – Markov Models for Object Categories," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 31, no. 1, pp. 114–128, Jan. 2009.

[95] Y. Chen, L. Zhu, A. Yuille, and H. Zhang, "Unsupervised Learning of Probabilistic Object Models(POMs) for Object Classification, Segmenta-

tion, and Recognition Using Knowledge Propagation," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 31, no. 10, pp. 1747–1761, Oct. 2009.

[96] S. Todorovic and N. Ahuja, "Unsupervised Category, Modeling, Recognition, and Segmentation in Images," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 30, no. 12, pp. 2158–2174, Dec. 2008.

[97] 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. (in Japanese)

[98] R. Fergus, P. Perona, and A. Zisserman, "Object Class Recognition by Unsupervised Scale-Invariant Learning," *Proc. Conf. Computer Vision and Pattern Recognition*, pp. 264–271, 2003.

[99] M. Frits and B. Schiele, "Towards Unsupervised Discovery of Visual Categories," *Lecture Notes in Computer Science, Springer*, vol. 4147, pp. 232–241, 2006

[100] G. Kim, C. Faloutsos, and M. Hebert, "Unsupervised Modeling of Object Categories Using Link Analysis Techniques," *Proc. Conf. Computer Vision and Pattern Recognition*, pp. 1–8, June, 2008.

[101] B. Ommer and J. M. Buhmann, "Learning the Compositional Nature of Visual Object Categories for Recognition," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 32, no. 3, pp. 501–516, Mar. 2010.

[102] L. Fei-Fei, R. Fergus, and P. Perona,, "One-shot learning of object categories," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 28, no. 4, pp. 594–611, Apr. 2006.

[103] K. Grauman and T. Darrell, "Unsupervised learning of categories from sets of partially matching image features," *Proc. Conf. Computer Vision and Pattern Recognition*, pp. 19–25, June 2006.

[104] K. Mikolajczyk and C. Schmid, "A Performance Evaluation of Local Descriptors," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 27, no. 10, pp. 1615–1630, Oct. 2002.

[105] S. Belongie, J. Malik, and J. Puzicha, "Shape Matching and Object Recognition Using Shape Contexts," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 2, no. 4, pp. 509–522, Apr. 2002.

[106] W. Freeman and E. Adelson, "The Design and Use of Steerable Filters," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 13, no. 9, pp. 891–906, Sept. 1991.

[107] Y. Ke and R. Sukthankar, "PCA-SIFT: A More Distinctive Representation for Local Image Descriptors," *Proc. Conf. Computer Vision and Pattern Recognition*, pp. 511–517, 2004.

[108] J. Koenderink and A. van Doorn, "Representation of Local Geometry in the Visual System," *Biological Cybernetics*, vol. 55, pp. 367–375, 1987.

[109] S. Lazebnik, C. Schmid, and J. Ponce, "Sparse Texture Representation Using Affine-Invariant Neighborhoods," *Proc. Conf. Computer Vision and Pattern Recognition*, pp. 319–324, 2003.

[110] F. Schaffalitzky and A. Zisserman, "Multi-View Matching for Unordered Image Sets," *Proc. European Conf. Computer Vision*, pp. 414–431, 2002.

[111] L. V. Gool, T. Moons, and D. Ungureanu, "Affine/Photometric Invariants for Planar Intensity Patterns," *Proc. European Conf. Computer Vision*, pp. 642–651, 1996.

[112] H. Fujiyoshi, "Gradient-Based Features Extraction -SIFT and HOG-," *Information Processing Society of Japan Technical Report: Computer Vision and Image Media*, pp. 211-224, 2007. (in Japanese)

[113] J. J. Koenderink, "The structure of images," *Proc. of Biological Cybernetics*, vol. 50, no. 5, pp. 363–370, Aug. 1984

[114] T. Lindeberg, "Scale-space theory: A basic tool for analysing structures at different scales," *Journal of Applied Statistics*, vol. 21, no. 2, pp. 224–270, 1994.

[115] C. Harris and M. Stephens, "A combined corner and edge detector," *Proc. Conf. Alvey Vision*. pp. 147–151, 1988.

[116] M. Brown and D. G. Lowe, "Invariant features from interest point groups," *Proc. Conf. British Machine Vision*, pp.656–665, 2002.

[117] 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. (in Japanese)

[118] K. Mikolajczyk and C. Schmid, "A Performance Evaluation of Local Descriptors," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 27, no. 10, pp. 1615–1630, Oct. 2005.

[119] G. Liu, Z. Lin, Y. Yu, and X. Tang, "Unsupervised Object Segmentation with a Hybrid Graph Model (HGM)," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 32, no. 5, pp. 910–924, May 2010.

[120] Y. Fujita, "Personal Robot R100," *Journal of Robotics Society of Japan*, vol. 18, no. 2, pp. 40– 41, Mar. 2000. (in Japanese)

[121] Y. Fujita, "Overview of Personal Robot PaPeRo," *RSJ 2001*, 2001. (in Japanese)

List of publications

Journal Papers

1. H. Madokoro, K. Sato, and M. Ishii, "Acquisition of World Image and Self-Localization Using Sequential View Images," *Trans. Institute of Electronics, Information and Communication Engineers (D–II)*, vol. J83–D–II, no. 12, pp. 2587–2596, Dec. 2000. (in Japanese)

2. K. Sato, M. Ishii, and H. Madokoro, "Experiment and Evaluation on Patrol Robot System for Hospital," *Trans. Institute of Electronics, Information and Communication Engineers (D–I)*, vol. J84–D–I, no. 6, pp. 855–866, Jun. 2001. (in Japanese)

3. M. Tsukada, Y. Utsumi, H. Madokoro, and K. Sato, "Unsupervised Feature Selection and Category Classification for a Vision-Based Mobile Robot," *Trans. Institute of Electronics, Information and Communication Engineers Inf. & Syst.*, (accepted with conditions).

International Conferences

1. H. Madokoro, K. Sato, and M. Ishii, "Orientation Selectivity for Representation of Facial Expression Changes," *Proc. International Joint Conference on Neural Networks*, pp. 1210–1215, Aug. 2007.

2. H. Madokoro, and K. Sato, "Visualizing Support Vectors and Topological Data Mapping for Improved Generalization Capabilities," *Proc. International Joint Conference on Neural Networks*, pp. 4226–4232, July 2010.

3. H. Madokoro, K. Sato, and M. Ishii, "Improved Generalization Abilities by Topological Data Mapping," *Proc. Workshop on Self-Organizing Maps*, pp. 513–520, Aug. 2005.

4. H. Madokoro, K. Sato, and M. Ishii, "Training Data Modeling Using Counter Propagation Networks for Improved Generalization Abilities," *Proc. International Conference on Computational Intelligence for Modeling, Control and Automation*, pp. 999–1004, Nov. 2005.

5. M. Tsukada, H. Madokoro, and K. Sato, "Unsupervised and Adaptive Category Classification for a Vision-Based Mobile Robot," *Proc. International Joint Conference on Neural Networks*, pp. 4157–4162, July 2010.

6. Y. Utsumi, M. Tsukada, H. Madokoro, and K. Sato, "Selection of SIFT Feature Points for Scene Description in Robot Vision," *Proc. International Joint Conference Systems, Man and Cybernetics*, Oct. 2010, (accepted).

Technical Journals

1. H. Madokoro, K. Sato, M. Ishii, "Acquisition of World Images and Self-Localization Estimation using Viewing Image Sequences," *Systems and Computers in Japan, Wiley Periodicals, Inc.*, vol. 34, no. 1, pp. 68–78, 2003.

2. K. Sato, M. Ishii, H. Madokoro, "Testing and Evaluation of Patrol Robot System for Hospital," *Electronics and Communications in Japan (Part III: Fundamental Electronic Science), Wiley Periodicals, Inc.*, vol. 86, No. 12, 2003.

Patent

1. Patent No. 3731672, Pattern Extraction Devices, H. Madokoro, K. Sato, and M. Ishii, Japanese Patent, Oct. 2005. (in Japanese)