Visualizing Support Vectors and Topological Data Mapping for Improved Generalization Capabilities

Hirokazu Madokoro, *Member, IEEE* and Kazuhito Sato, *Member, IEEE*

***Abstract*— This paper presents a method to improve generalization capabilities of supervised neural networks based on topological data mapping used in Counter Propagation Networks (CPNs). Using topological data mapping on CPNs the method presented herein 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 Support Vector Machines (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 are improved using our method. Moreover, we applied our method to face recognition under various illumination conditions using the Yale Face Database B. The results indicate that our method provides not only improved generalization capabilities, but also visualizes spatial distributions of SVs on a category map.**

I. INTRODUCTION

Neural networks (NNs) are widely applied to many problems that show difficulty of formulation or reformulation because of dynamic, high-dimensional, or nonlinear 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 [1] 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 paper, we specifically examine precondition 1) related in training data. This paper 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) [2]. Actually, CPNs are supervised NNs based on Self-Organizing Maps (SOMs) [3] 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) [4], 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.

II. RELATED WORK

Various methods based on training data have been proposed especially in expansion of whole training data quantitatively. Holmstrom et al. proposed a method to expand training data to add Gaussian-type white noise [5]. Karystinos

Hirokazu Madokoro and Kazuhito Sato are with the Faculty of Systems Science and Technology, Akita Prefectural University, Yurihonjo-shi, 015-0055, Japan (phone: +81 184 272081; email: madokoro@akita-pu.ac.jp).

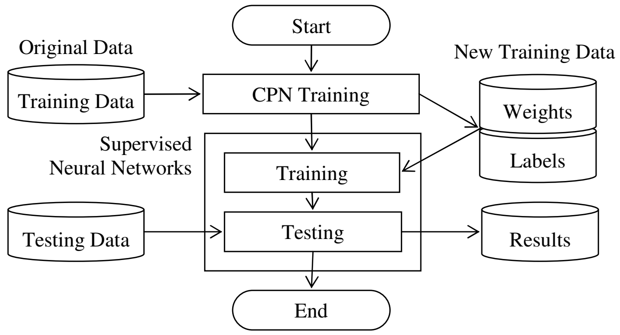


Fig. 1. Procedure of our method. Weights and labels of CPN are used as training data of SVM.

et al. proposed a method to expand training data randomly based on probability density functions [6]. 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 [7]. 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. [8], a method used in double-back propagation by Drucker et al. [9], a method to delete redundant units on the hidden layer by Matsunaga et al. [10], etc. Tsuda [11] 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.

III. 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 paper 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.

A. Whole architecture of our method

Fig. 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 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-Take-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) [17], etc.

B. 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. 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

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

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

$$c = argmin(d_{n,m}). \quad (2)$$

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

$$u^i_{n,m}(t+1) = u^i_{n,m}(t) + \alpha(t)(x_i(t) - u^i_{n,m}(t)). \quad (3)$$

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

$$v^j_{n,m}(t+1) = v^j_{n,m}(t) + \beta(t)(t_j(t) - v^j_{n,m}(t)). \quad (4)$$

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.

C. 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 $\vec{w}$ of the discriminant function f as N sets of input data $\vec{x_i}(i = 1, ..., N)$ defined as the following:

$$f(\vec{x}) = sign(\vec{w^T}\vec{x} - b), \tag{5}$$

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 $\vec{x_i}$, then the hyperplane of the margin is maximum in two classes calculated using the minimization problem as

$$L(\vec{w}, \vec{\epsilon}) = \frac{1}{2}||w||^2 + \sum_{i=1}^{N} \epsilon_i, \tag{6}$$

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

$$L_d(\vec{\alpha}) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{N} \alpha_i \alpha_j t_i t_j \vec{x^T} \vec{x_i}. \tag{7}$$

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

$$b = \vec{w^T}\vec{x_i} \pm 1. \tag{8}$$

To introduce a nonlinear mapping function Φ to a high-dimensional feature space, Eq. (7) is calculated as

$$L_d(\vec{\alpha}) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{N} \alpha_i \alpha_j t_i t_j \Phi(\vec{x^T}) \Phi(\vec{x_i}), \tag{9}$$

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

$$\Phi(\vec{x})^T \Phi(\vec{x_i}) = K(\vec{x}, \vec{x_i}). \tag{10}$$

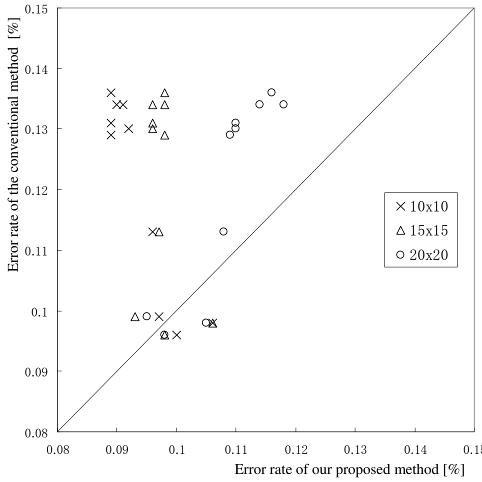
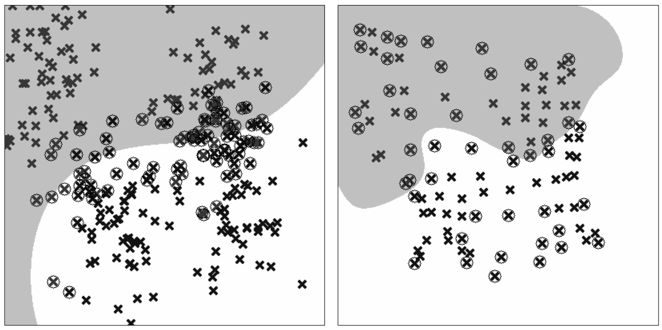


Fig. 2. Comparison results of error rates of the Normal Mixtures dataset with change of λ and the size of category maps.



(a) Classification result of SVM trained by original data

(b) Classification result of our method with new training data

Fig. 3. Classification results of the Normal Mixtures dataset.

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

$$K(\vec{x}, \vec{x_i}) = exp(-\frac{||\vec{x} - \vec{x_i}||^2}{\lambda}), \tag{11}$$

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.

IV. 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 [12] and the Cone-Torus dataset [13], which are widely used for evaluation of generalization capabilities.

A. Normal Mixtures Dataset

The Normal Mixtures dataset [12] 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.

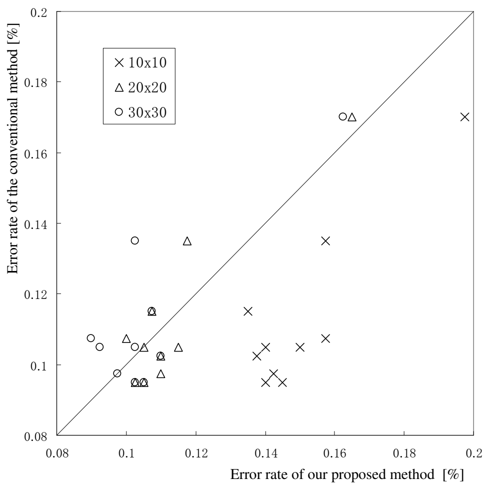
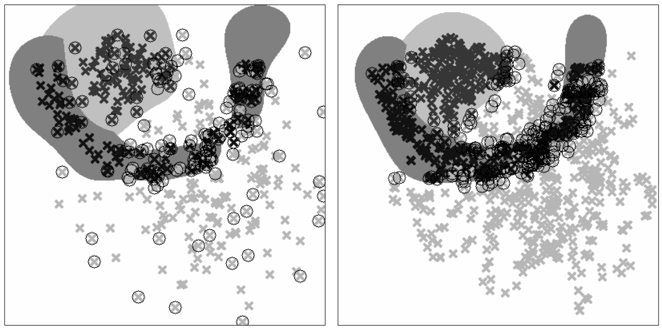


Fig. 4. Comparison results of error rates of the Cone-Torus dataset with changing of λ and the size of category maps.



(a) Classification result of SVM trained by original data

(b) Classification result of our method with new training data

Fig. 5. Classification results obtained using the Cone-Torus dataset.

Fig. 2 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. 11 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. 3(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 merged as a soft margin are visible. Fig. 3(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 rate is reduced 0.70 percent.

B. Cone-Torus Dataset

The Cone-Torus dataset [13], 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. 4 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 of 30×30 units is the minimum of the error rate. In this case, the training data are expanded to 225 percent.

Fig. 5 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 [8], 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.

V. 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 which a hidden rule can be evaluated step-by-step is necessary. We use the Yale Face Database B [14], which is an open dataset, to treat various illumination conditions step-by-step.

A. 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 [15], Okabe et al. described that

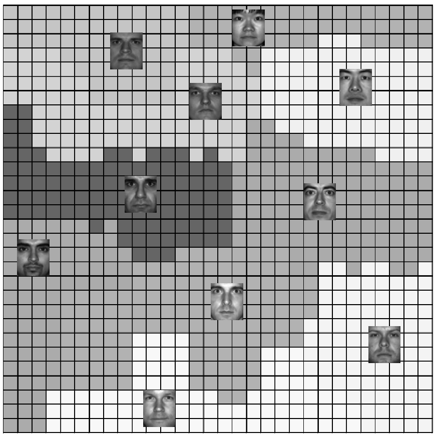


Fig. 6. Category map (Face images without illumination changes show a person of each category).

TABLE I
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%

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.

B. 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 [16] 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. 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 robustness against illumination conditions, Okabe et al. obtained a good result with their method to use illumination cones [15]. We specifically examine simplicity of implementation to evaluate generalization capabilities in this experiment. Therefore, we do not use illumination cones.

C. Classification results

Fig. 6 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 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.

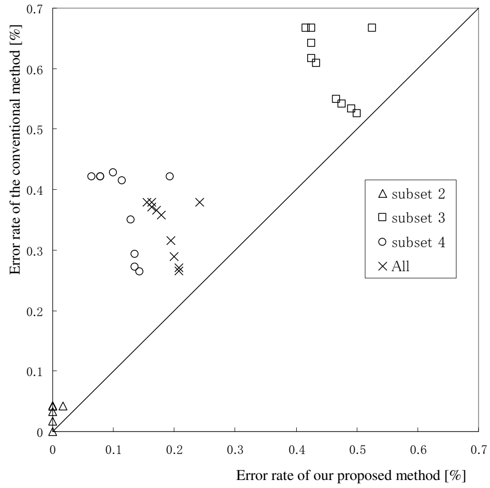


Fig. 7. Comparison of results of error rates with changing of λ and the size of category maps.

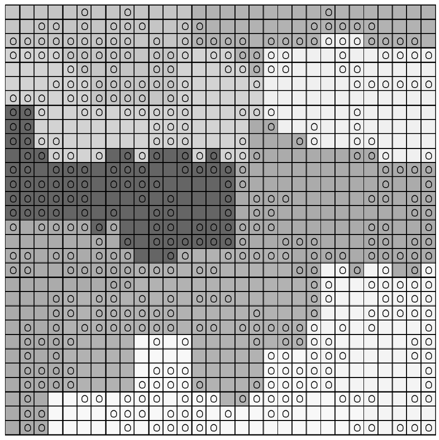


Fig. 8. SV units on the category map.

Fig. 7 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 I 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.

VI. 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 [17] 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 improved 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. 8 in the category map presented in Fig. 6. 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. 8 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.

VII. Conclusions

This paper 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.

Acknowledgment

This work was supported by a Grant-in-Aid for Young Scientists (B) No. 21700257 from the Ministry of Education, Culture, Sports, Science and Technology of Japan.

References

[1] H. Kita, ”Generalization of Neural Networks,” Systems, Control and Information, vol.36, no.10, pp.625–633, 1992.
[2] R. H. Nielsen, ”Counterpropagation Networks,” Applied Optics, vol.26, no.23, pp.4979–4984, Dec. 1987.
[3] T. Kohonen, Self-organizing maps, Springer Series in Information Sciences, 1995
[4] V. Vapnik, Statistical Learning Theory, Wiley, 1998.
[5] 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.
[6] 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.
[7] N. Tanaka, T. Koreyeda, T. Inoue and K. Kajitani, ”A Method of Learning BP Network by Expanding the Distribution of Category,” IEICE D-II vol.J81-D-II, no.2, pp.293-300, Feb. 1998.
[8] D. Chakraborty and N. R. Pal, ”A Novel Training Scheme for Multi-layered Perceptrons to Realize Proper Generalization and Incremental Learning,” IEEE Trans. Neural Networks, vol.14, no.1, pp.1–14, Jan. 2003.
[9] 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.
[10] Y. Matsunaga, K. Murase, O. Yamakawa and M. Tanifuji, ”A Modified Back-Propagation Algorithm that Automatically Removes Redundant Hidden Units by Competition,” IEICE D-II vol.J79-D-II, no.3, pp.403-412, Mar. 1996.
[11] 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.
[12] B. D. Ripley, Pattern Recognition and Neural Networks, Cambridge University Press, Jan. 1996.
[13] L. I. Kuncheva, ”Combining Pattern Classifiers: Methods and Algorithms,” Wiley, Jul. 2004.
[14] 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 Machine Intelligence, vol.23, No.6, pp.643–660, June 2001.
[15] T. Okabe and Y. Sato, ”Support Vector Machines for Object Recognition under Varying Illumination,” Information Processing Society of Japan (IPSJ), Vol.44, No.SIG_5(CVIM_6), pp.22–29, Apr. 2003.
[16] K.C. Lee, J. Ho, and D. Kriegman, ”Acquiring Linear Subspaces for Face Recognition under Variable Lighting,” IEEE Trans. Pattern Analysis Machine Intelligence, vol.27, No.5, pp.684–698, May 2001.
[17] D. E. Rumelhart, G. E. Hinton and R. J. Williams, ”Learning Representations by Back-Propagating Errors,” Nature, vol.323, pp.533–536, 1986.