Improved Generalization Abilities by Topological Data Mapping

Hirokazu Madokoro†, Kazuhito Sato‡, and Masaki Ishii†
† Akita Prefecture Industrial Technology Center
4-11 Sanuki, Araya-machi, Akita 010-1623, Japan
{madokoro,ishii}@akita-iri.pref.akita.jp
‡ Akita Research Institute of Advanced Technology
4-21 Sanuki, Araya-machi, Akita 010-1623, Japan
ksato@ait.pref.akita.jp

Abstract - *This report presents an optimizing method for training data on Back Propagation Networks (BPNs) to improve its generalization abilities. The method is based on topological data mapping used in Counter Propagation Networks (CPNs). The CPN saves the input data to the category map while holding topological data structures. We used weights and labels of the category map for the training data. The method has three features; 1) training data is changed by the size of category map, 2) interpolation training data is made under the topological space, and 3) overlapping training data is avoided by a Winner-Take-All competition. Experimental results show that the expanded training data improved the generalization ability.*

Key words - topological mapping, counter propagation networks, generalization abilities

1 Introduction

Back Propagation Networks (BPNs) [8] have been widely used for many systems such as image processing, robot control, market planning, and so on. Although the BPN is a popular artificial neural network [7], it is difficult to correct mapping of untrained data, which is known as a generalization problem.

Existing methods for improving generalization abilities are classifiable as two approaches. One approach is based on the network parameter. Matsunaga et al. removed redundant hidden units by competition based on a goodness factor, which is an influence value to the next unit [1]. Ishii et al. modified evaluation functions for weight representation by linear constraints [2]. Chakraborty et al. divided subnets whether a data point belong to a certain class or not [3]. Setting for the best combination from many parameters is a hard task for designers in each problem to be solved. Another approach is based on the training data. Tishby et al. presented a cross validation based method [4]. Using the cross validation, a sufficient number of training data is needed for dividing two type of data sets; training data sets and validation data sets. Karystions et al. randomly expanded training data [5]. Sarkar et al. added random noise to the training data [6]. The approach of generation or expansion training data should be considered the original data structure.

This report presents an optimizing method for training data on BPNs to improve its generalization abilities. The method is based on the topological data mapping on the category map of Counter Propagation Networks (CPNs) [7]. The CPN saves the input data to the category map while holding topological data structures. We used weights and labels of the category map for the training data. Experimental results show that the expanded training data improved the generalization ability.

2 An Optimizing Method for Training Data

This section presents the proposed method using the CPN category map. Then, we describe the training algorithm of the CPN.

2.1 Proposal Method

Both the CPN and the BPN are famous neural networks that use supervised training. Comparison of both mapping abilities shows that the BPN is superior to the CPN for many applications [7]. However, the generalization ability of the BPN often decreases in these cases: 1) lack of training data, 2) sparse training data between clusters, and 3) overlapping training data. The generalization ability is strongly affected by the training data.

Using neighborhood training and the Winner-Take-All (WTA) competition, the CPN maps the input data into the topological space as a category map. We used weights and labels of the category map for the training data. Using the category map, we directly address BPNs vulnerabilities: 1) training data is changed by the size of category map, 2) interpolation training data is made under the topological space, and 3) overlapping training data is avoided by the WTA competition. Our method uses the category map for improving generalization abilities.

Fig. 1 shows the network architecture and the procedure of our method. In the first step of training, the CPN trains from the original training data. We set a large neighborhood region for the interpolation characteristic. After training, the CPN produces a category map. The weights between the input layer and Kohonen layer and the labels of the category map become new training data for the BPN. In the next step, the BPN learns from this data. Using weights and labels, our method controls the amount of training data while retaining the topological structure of the original data.

2.2 Counter Propagation Networks

The CPN is a supervised and self organizing neural network that integrates Kohonen's training algorithm and Grossberg's outstar training algorithm. The network comprises three layers: an input layer, a Kohonen layer, and a Grossberg layer. The input layer receives the training data. The Kohonen layer performs topological mapping through WTA competition. The labeled units of this layer are called the category map. The Grossberg layer receives teaching signals and labels all units of the Kohonen layer. In our method, the Kohonen layer contains two-dimensional units; the Input layer and the Grossberg layer contains one-dimensional units.

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

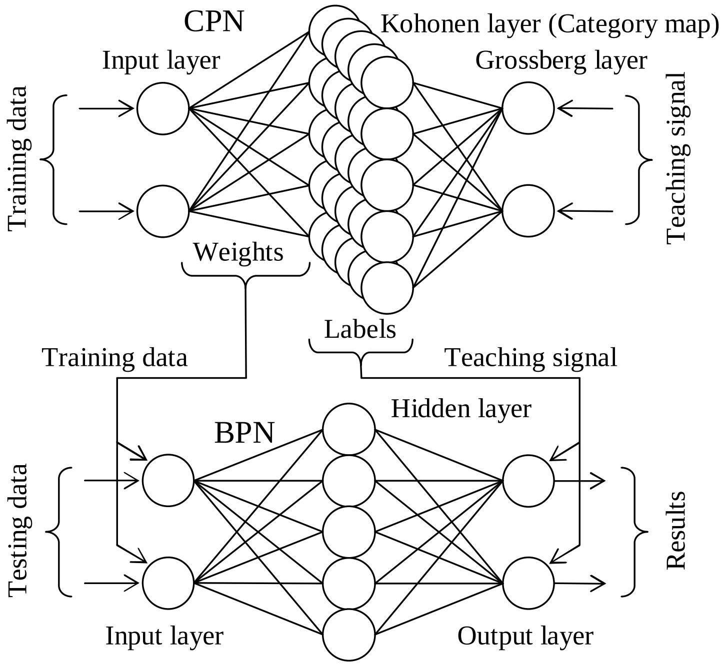


Figure 1: Network architecture and procedure of our method. The CPN trains itself using the original training data. The BPN trains by the weights and the labels of the CPN. The BPN does not know the original data.

$d_{n,m}$ between $x_i(t)$ and $w^i_{n,m}(t)$ is calculated as follows:

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

The winner unit c which $d_{n,m}$ becomes a minimum by:

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

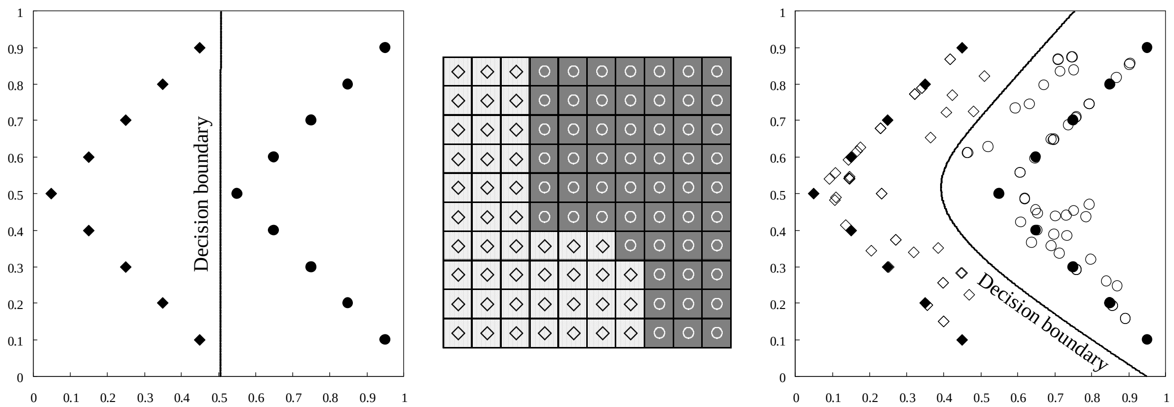
Let $N_c(t)$ be the units of neighborhood of the unit c. The weight $w^i_{n,m}(t)$ inside $N_c(t)$ is updated by the Kohonen training algorithm as follow:

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

The weight $u^j_{n,m}(t)$ inside $N_c(t)$ is updated by the Grossberg outstar training algorithm as follow:

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

where $t_j(t)$ is the teach 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 at 0.5 and 0.9, respectively. The maximum number of training iterations is set at 1,000 *steps*.



(a) Original training points (b) Category map of CPN (c) Result of our method

Figure 2: *Arrow* clustering. (a) Solid diamond points and solid circle points indicate the original training points; the decision boundary was generated from this data. (c) Blank diamond points and blank circle points show expanded training points by the CPN; the decision boundary was generated by our method. (b) Each unit corresponds to the expanded training data set.

3 Experimental Results

We next present two experimental results: classification of a two-dimensional space and human skin color extraction from general scene images. These experiments are intended to determine the effectiveness of topological mapping and its generalization abilities in real applications.

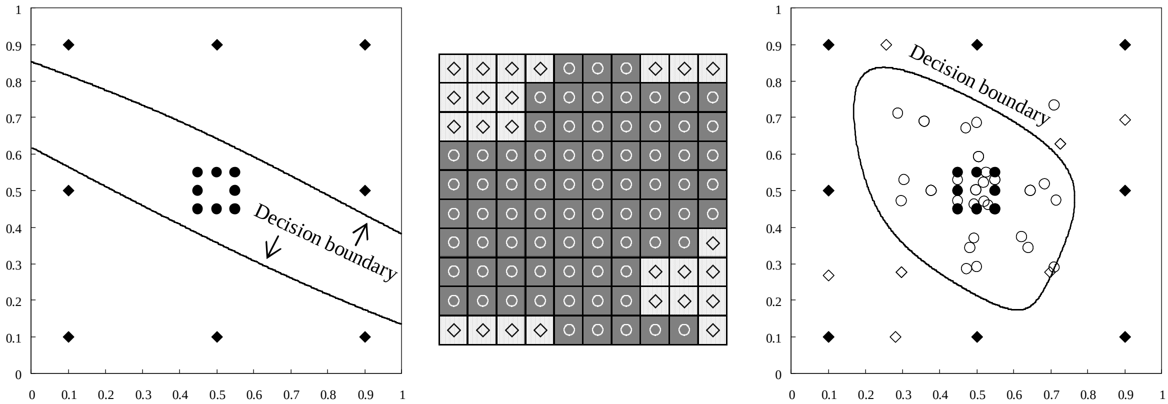
3.1 Clustering

Fig. 2(a) shows the result of the *Arrow* clustering. The solid diamond points and the solid circle points indicate the original training points. A wide space separated the clusters; consequently, the decision boundary was linear, as shown in Fig. 2(a). This result shows the BPN was unable to reflect the distribution of clusters [9]. Fig. 2(c) shows the result of our method. The blank diamond points and the blank circle points indicate new training data generated by the CPN. These points correspond to all units of the category map shown in Fig. 2(b). The curved decision boundary indicated that our method was reflected in the distribution of both clusters.

Fig. 3(a) shows the result of the *Square* clustering. The solid diamond points distributed the large *Square* cluster. The solid circle points distributed the small *Square* cluster surrounded by the large one. The decision boundaries shown in Fig. 3(a) divided the large *Square* cluster into two independent regions. Fig. 3(c) shows the results of our method by the category map shown in Fig. 3(b). The decision boundary exists between the two clusters. The cluster of solid diamond points is arranged in a single region outside of the decision boundary.

3.2 Skin-Color Extraction from Scene Images

Fig. 4 shows the system architecture of our method for extracting a human skin color region from a scene image. In [10], $L^*u^*v^*$ and $L^*a^*b^*$ are suitable color spaces for expressing a human skin. The component of L^* is a luminance value, and the components of u^*, v^*, a^*,



(a) Original training points (b) Category map of CPN (c) Result of our method

Figure 3: *Square* clustering. (a) Solid diamond points and the solid circle points show the original training points; the decision boundary was generated from this data. (c) Blank diamond points and blank circle points show expanded training points by the CPN; the decision boundary was generated by our method. (b) Each unit corresponds to the expanded training data set.

and b^* are chrominance values. We use both color spaces, except L^* to reduce the influence of brightness.

Training and testing images are shown in Fig. 5(a) and (b), respectively. Training data were obtained from three regions: the skin color region, the head-hair region, and the background region. Each region was set in five points. Training data were acquired from the second neighborhood region ($5pixels \times 5pixels$) from each points. Therefore, the CPN was trained by 375 points ($5pixels \times 5pixels \times 5points \times 5regions$).

Fig. 5(c) shows the result of skin region trained by the original data. Fig. 5(d)$\sim$(h) presents the extraction results of our method. For each extraction result image, upper right rates are shown: the skin color extraction rate and the false extraction rate. These results correspond to the category maps shown in Fig. 6. Comparison with the result of original data shows that the skin color extraction rate was increased by our method. The false extraction rate was decreased more than $20units \times 20units$ of the category maps. The region of high brightness near the cheek and low brightness near the upper neck were extracted using our method.

4 Discussion

We next discuss generalization abilities from the perspective of the category map. The category map displays the number of clusters from the original data. For example, the category map shown in Fig. 2(b) consists of two clusters. However, the category map shown in Fig. 3(b) comprises five clusters. The circle cluster is only one cluster, but the diamond cluster contains four separate clusters. The category map indicates that it is better to recognize the separated clusters as individual clusters. Therefore, our method can optimize the number of output units. We infer that this optimization increases the expression ability of the BPN. In Fig. 6, the cluster of the skin color region is independent. However, the skin color region has some specific units. For example, regarding the category map shown in Fig. 6(c), two units are labeled head-hair inside of the skin color region. We consider that if we exchange the head-hair labels for the skin color labels, the extraction results will be

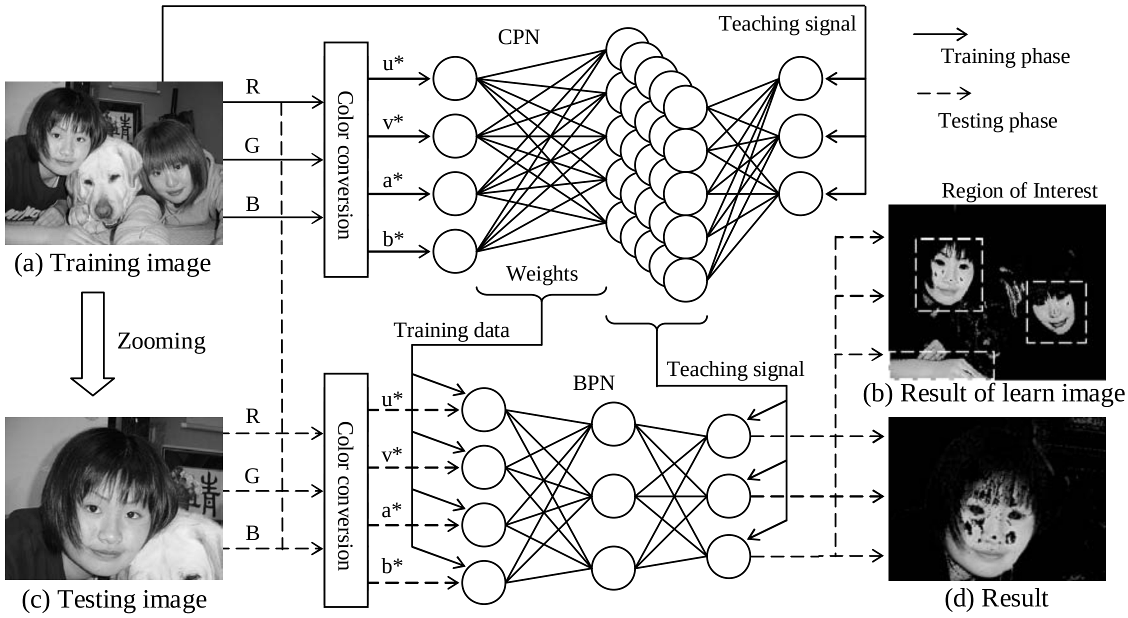


Figure 4: System architecture of our method to extract a human skin color region from scene images. (a) Training data are extracted from this wide range image. (b) The region of interest of the human skin color region is detected from the training image result. (c) Testing image after zooming the camera to the region of interest. (d) Result of human skin color region from testing image.

improved. We infer that the category map indicates suitable number of the output units of the BPN. Clusters of the head-hair region are divided into two clusters. We also consider than if we assign two output units for the heard-hair region, the extraction results will be improved.

As another feature, the category map controls the number of the training data while holding the topological structure of the original data. For the experiment on clustering, we used a fixed size of the category map ($10units \times 10units$). In this case, the training data were expanded to more than five times their original number. The generalization ability of the BPN was generated by interpolated training data, shown as the blank diamonds and blank circles, in the sparse region of the cluster. In the experiment on extraction of the human skin color region, we changed the size of the category map from $10units \times 10units$ to $50units \times 50units$ as shown in Fig. 6. The number of the original data points was 375. In the case of $10units \times 10units$, the training data were compressed to 1/4. In the case of $20units \times 20units$, the training data became nearly equal to the original data. In the case of over $30units \times 30units$, the training data were more numerous than the original data. The scene image is too complex, but we set only five points for each region. This result demonstrates that the BPN acquires efficient generalization ability from few training points.

5 Conclusions

This report proposed an optimizing method for training data on BPNs to improve its generalization abilities. The following results were obtained.

- Using the weights and labels of the category map, our method controls the number of

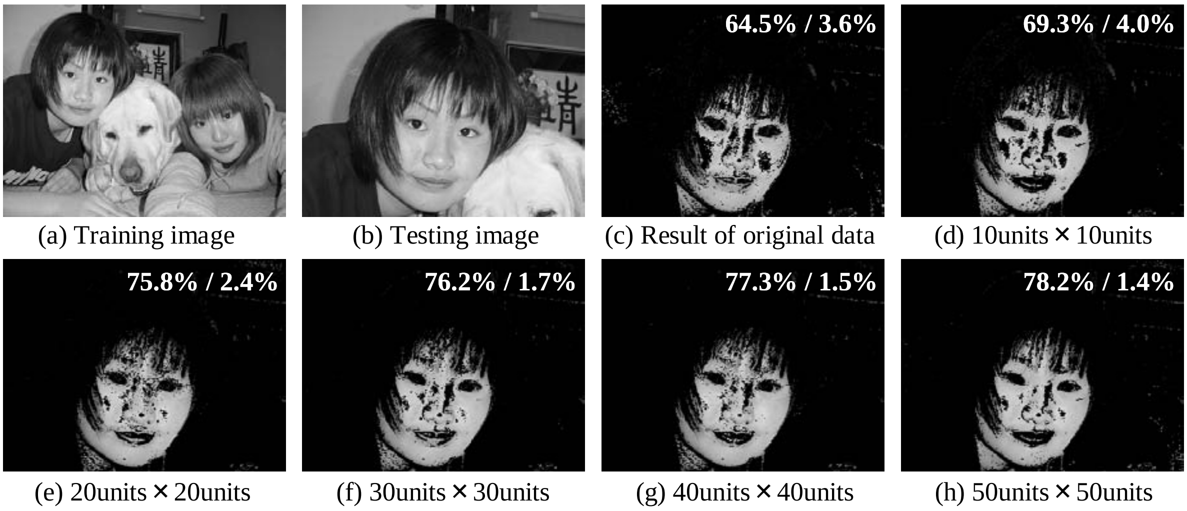


Figure 5: Extraction results of the human skin color region. (a) Wide range scene image for extracting the training data. (b) Zoom image for testing. (c) Result of BPN from original data. (d)∼(h) Result of our method. For each extraction result image, upper right rates are shown: the skin color extraction rate and the false extraction rate.

training data while maintaining the topological structure of the original data.

- The BPN generated a new decision boundary that reflects the cluster distribution.
- Our method improves the generalization ability from few training points.
- The cluster distribution on the category map indicated the appropriate number of output units of the BPN.
- These results for human skin color extraction verify that our method is effective for the practical use in real life applications.

Future studies must define rules for modifying the category map when numerous points appear inside a cluster or a divided cluster.

Acknowledgments

This work was supported by a Grant-in-Aid for Young Scientists (B) No.16700230 from the Ministry of Education, Culture, Sports, Science and Technology of Japan.

References

[1] 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.

[2] M. Ishii and I. Kumazawa, "Introduction of Linear Constraints on Weight Representation of Multilayer Networks for Generalization and Application for Character Recognition," IEICE D-II vol.J84-D-II, no.3, pp.541-548, Mar., 2001.

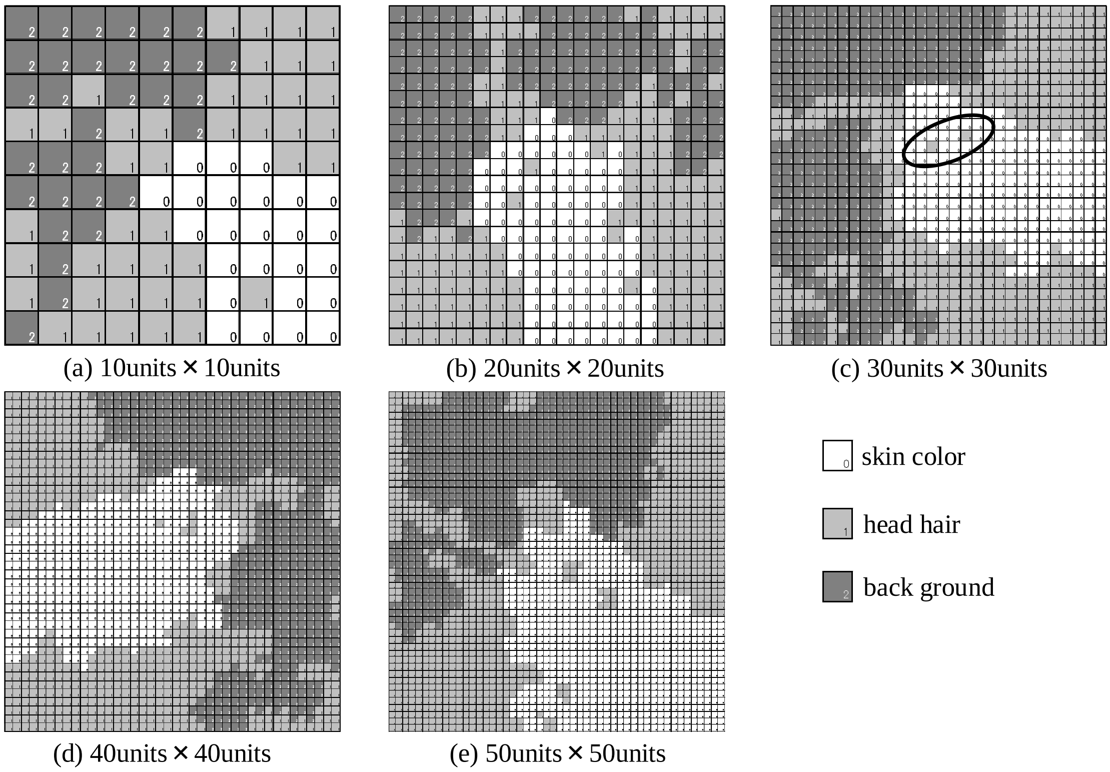


Figure 6: Category maps of the CPN. Each category map corresponds to the results of Fig. 5(d)∼(h).

[3] 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.

[4] N. Tishby and E. Levin, "Consistent Inference of Probabilities in Layered Networks: Predictions and generalization," International Joint Conference on Neural Networks, vol.2, pp.403-409, 1989.

[5] 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.

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

[7] R. H. Nielsen,"Counterpropagation Networks," Applied Optics, vol.26, No.23, pp.4979-4984, Dec., 1987.

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

[9] 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.

[10] K. Takahashi, M. Matsuura, T. Sugiyama and K. Abe, "Comparison of Color Coordinate Systems Based on Human-Made Color Classification and Image Segmentation," IEICE D-II vol.J84-D-II, no.7, pp.1378-1388, Jul., 2001.