Communicated by Helge Ritter

Self-Organizing Map with Dynamical Node Splitting: Application to Handwritten Digit Recognition*

Sung-Bae Cho
Department of Computer Science, Yonsei University, Seoul, 120–749 Korea

This article presents a simple yet elegant pattern recognizer based on a dynamic node-splitting scheme for the self-organizing map that can adapt its structure as well as its weights. The scheme makes use of a structure adaptation capability to place the nodes of prototype vectors into the pattern space accurately so as to make the decision boundaries as close to the class boundaries as possible. In order to show the performance of the proposed scheme, experiments with the unconstrained handwritten digit database of Concordia University in Canada were conducted. The proposed method for an incremental formation of feature maps is 96.05 percent of the recognition rate. In view of the elegant simplicity of the approach, the reported performance is remarkable and can stand up to one of the best results reported in the literature with the same database.

1 Introduction

A wide variety of methods have been proposed to realize the perfect recognizer of handwritten digits by computer. Many systems have been developed, but more work is required to be able to match human performance (Suen, Nadal, Legault, Mai, & Lam, 1992). Recently the emerging technology of neural networks has been used to design a pattern recognizer that matches human ability.

Among several models, the multilayer Perceptron has been recognized as a powerful tool for pattern classification problems. Its strength lies in the discriminative power and capability of learning and representing implicit knowledge, but there are several difficulties in solving real-world problems. One of the shortcomings is how to determine the size and structure of the network. To overcome this difficulty, several approaches based on the structure adaptation of the networks have been proposed (Koikkalainen & Oja, 1990; Sanger, 1991; Fritzke, 1994; Li, Tang, & Fang, 1995).[1]

* An abstracted version of this article was presented at the 13^{th} International Conference on Pattern Recognition, Vienna, 1996.

[1] In addition, many papers related to this topic have been presented at several international conferences. Representative of them are Fritzke (1994), for a general approach to structure-adaptation of neural networks; Koikkalainen and Oja (1990) and Sanger (1991) for a typical tree-structured neural network; and Li et al. (1995) for another tree-structured neural network applied to a pattern recognition problem.

In this article we propose an efficient pattern recognizer based on a dynamic node-splitting scheme for the self-organizing map (SOM) that can adjust its structure as well as its weights during learning. The network utilizes the structure-adaptation capability to place the nodes of prototype vectors into the pattern space accurately so as to make the decision boundaries as close to the class boundaries as possible. This approach shapes the general structure-adaptive SOM into a pattern recognizer by splitting a node representing more than one class into a submap (composed of four nodes). It also allows an incremental formation of feature maps in the real-world problem of digit recognition. We will show how a network is properly constructed in order to solve the problem of handwritten digit recognition.

2 Preprocessing

2.1 The Database. The handwritten digit database of Concordia University of Canada consists of 6000 unconstrained digits originally collected from dead letter envelopes by the U.S. Postal Service at different locations in the United States. The digits of this database were digitized in bilevel on a 64×224 grid of 0.153 mm square elements, giving a resolution of approximately 166 pixels per inch (Suen, Nadel, Mai, Legualt, & Lam, 1990). Among the data, 4000 digits were used for training and 2000 digits for testing. The representative samples in Figure 1 taken from the database show that many different writing styles are apparent, as well as digits of different sizes and stroke widths.

2.2 Feature Extraction. Digits, whether handwritten or typed, are essentially line drawings—one-dimensional structures in a two-dimensional space. Thus, local detection of line segments might be an adequate method for extracting features. For each location in the image, information about the presence of a line segment at a given direction is stored in a feature map (Knerr, Personnaz, & Dreyfus, 1992). In this article Kirsch masks have been used for extracting directional features (Kim & Lee, 1994).

Kirsch defined a nonlinear edge enhancement algorithm as follows (Pratt, 1978):

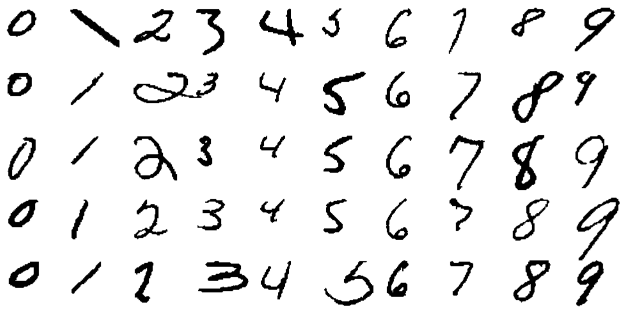
$$G(i, j) = \max\left\{1, \max_{k=0}^{7}[|5S_k - 3T_k|]\right\}$$

where

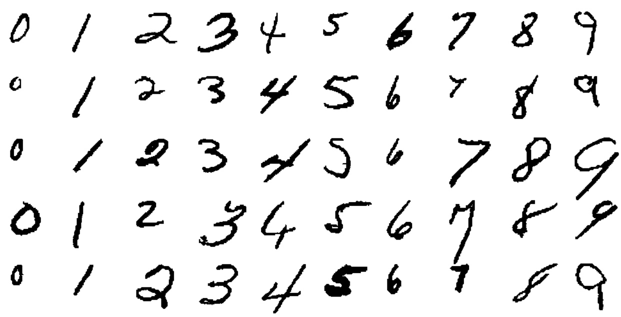
$$S_k = A_k + A_{k+1} + A_{k+2}$$

$$T_k = A_{k+3} + A_{k+4} + A_{k+5} + A_{k+6} + A_{k+7}.$$

Here, $G(i, j)$ is the gradient of pixel (i, j), the subscripts of A are evaluated modulo 8, and A_k $(k = 0, 1, \ldots, 7)$ is eight neighbors of pixel (i, j), defined as shown in Figure 2.



(a)



(b)

Figure 1: Sample data. (a) Training. (b) Test.

A_0	A_1	A_2
A_7	(i, j)	A_3
A_6	A_5	A_4

Figure 2: Definition of eight neighbors A_k ($k = 0, 1, \ldots, 7$) of pixel (i, j).

Since the data set was prepared by thorough preprocessing, each digit is scaled to fit in a 16×16 bounding box such that the aspect ratio of the image is preserved. Then feature vectors for vertical, horizontal, left-diagonal, and right-diagonal directions are obtained from the scaled image (Kim & Lee, 1994):

$$G_V(i, j) = \max(|5S_2 - 3T_2|, |5S_6 - 3T_6|)$$

$$G_H(i, j) = \max(|5S_0 - 3T_0|, |5S_4 - 3T_4|)$$

$$G_L(i, j) = \max(|5S_3 - 3T_3|, |5S_7 - 3T_7|)$$

$$G_R(i, j) = \max(|5S_1 - 3T_1|, |5S_5 - 3T_5|)$$

The final step in extracting the features compresses each 16×16 directional vector into a 4×4 one with averaging operator, which produces a value for 2×2 pixels with dividing by four the value obtained by summing the four values. In addition to the directional features, 4×4 compressed images have been used as global features. As a result, final features consist of five 4×4 features: four 4×4 local and one 4×4 global features.

3 The Method

In this section, we present the dynamic node-splitting scheme that can simultaneously determine a suitable number of nodes and the connection weights between input and output nodes in a SOM. The basic idea is simple:

1. Start with a basic neural network (in our case, a 4×4 map in which each node is fully connected to all input nodes).
2. Train the network with Kohonen's algorithm (Kohonen, 1990).
3. Calibrate the network using known input-output patterns to determine which node should be replaced with a submap of several nodes (in our case, a 2×2 map) and which node should be deleted.
4. Unless every node represents a unique class, go back to step 2.

Note that step 3 adjusts the structure of the map so that each node represents a unique label for the classification. In our scheme, the weights of a new node are initialized by interpolating the weights of neighboring nodes.

3.1 Basic Structure. The structure of the network is quite similar to Kohonen's SOM except for the irregular connectivity in the map. Figure 3 shows an instance of the network where each node represents a unique class. Each node is connected to all the input nodes with corresponding weights. (Actually, this is the final network structure obtained for recognizing the handwritten digits in the simulation.) The initial map of the network consists of 4×4 nodes. The weight vector of node i shall be denoted by $w_i \in R^n$.

The simplest analytical measure is the Euclidean distance between x and w_i. The minimum distance defines the winner w_c. If we define a neighborhood set N_c around node c, at each learning step all the nodes within N_c are updated, whereas nodes outside N_c are left intact. This neighborhood is centered around that node for which the best match with input x is found as:

$$||x - w_c|| = \min_i\{||x - w_i||\}.$$

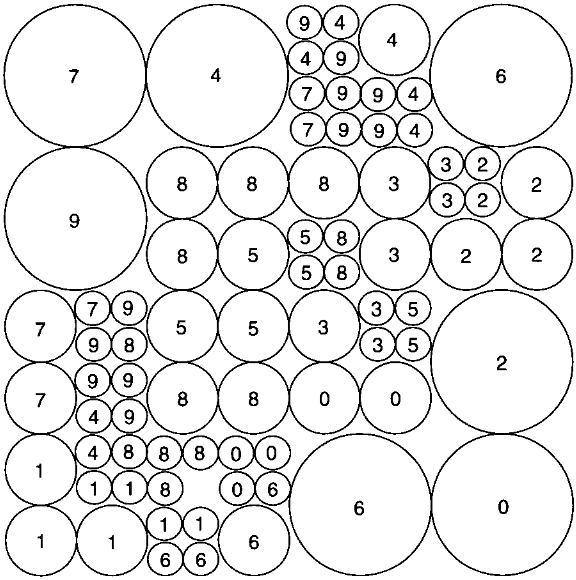


Figure 3: The proposed neural network. The number in each circle is the class that the corresponding node represents.

The width or radius of N_c can be time variable. For a good global ordering, it is advantageous to let N_c be very wide in the beginning and shrink monotonically with time (Kohonen, 1990).

The updating process may read

$$w_i(t+1) = \begin{cases} w_i(t) + \alpha(t)[x(t) - w_i(t)] & \text{if } i \in N_c(t), \\ w_i(t) & \text{if } i \notin N_c(t), \end{cases}$$

where $\alpha(t)$ is a learning rate, $0 < \alpha(t) < 1$.

3.2 Splitting Nodes. After a constant number of adaptation steps, a node representing more than one class is replaced with several nodes. (We used a submap of 2×2 nodes.) Obviously this node lies in a region of the input vector space where many misclassifications occur. If input patterns from different classes are covered by the same local node and activate this node to about the same degree, it might be the case where their vectors of local node activations are nearly identical.

Figure 4 shows how the network structure changes as some nodes representing duplicated classes are replaced by several nodes having finer resolution.

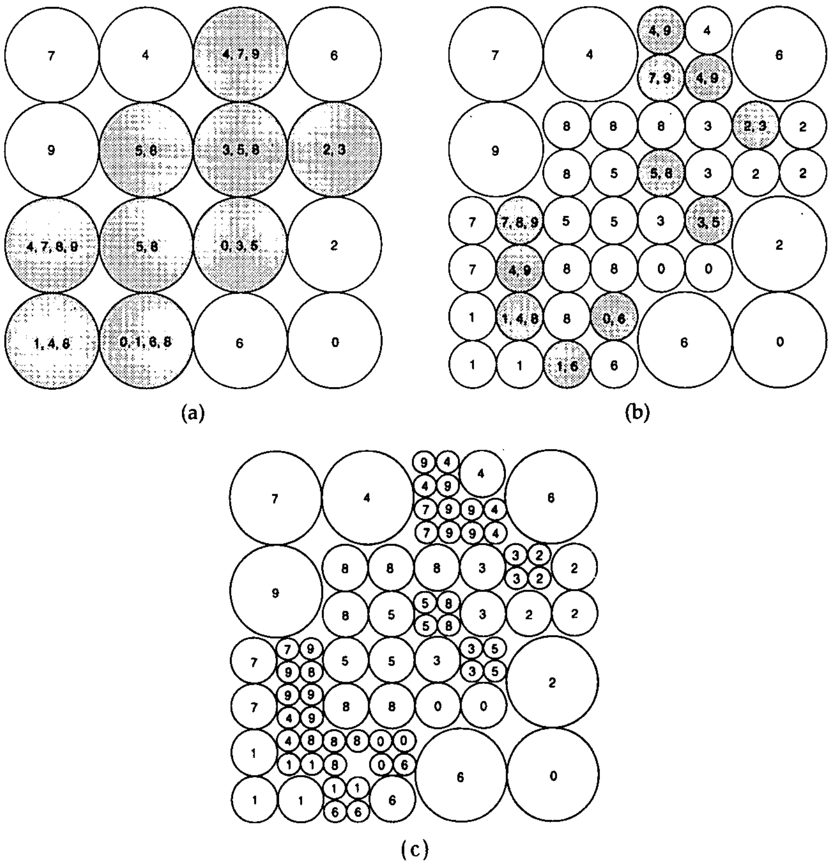


Figure 4: Map configurations changed through learning. (a) Initial status. (b) Intermediate status. (c) Final status.

3.3 Removing Nodes. The previous section shows how to extend the network structure. A necessary consequence is that all the nodes are connected directly or indirectly to each other. However, a problem may occur if the pattern space we try to discriminate has some disconnected regions. A solution can be found by introducing the deletion of nodes from the structure. An obvious criterion for a node to be deleted would be that it has a position in an area of the R^n where the probability density is zero (Fritzke, 1994). For this purpose, nodes that have been inactive longer than a specified length of time can be deleted. In the previous example, only one node is deleted at the final map (see Figure 4c).

Table 1: Comparisons of the Proposed Method with Alternative Methods (%)

Methods	Correct	Error	Reject	Training	Testing	Pixels per Inch
Lam & Suen, 1988	93.10	2.95	3.95	4000	2000	166
Nadal & Suen, 1988	86.05	2.25	11.70	4000	2000	166
Legault & Suen, 1989	93.90	1.60	4.50	4000	2000	166
Krzyzak, Dai, & Suen, 1990	86.40	1.00	12.60	4000	2000	166
Krzyzak, Dai, & Suen, 1990	94.85	5.15	0.00	4000	2000	166
Mai & Suen, 1990	92.95	2.15	4.90	4000	2000	166
Suen et al., 1990	93.05	0.00	6.95	4000	2000	166
Le Cun et al., 1990	96.40	3.40	0.20	7291	2007	
Martin & Pittman, 1990	96.00	4.00	0.00	35200	4000	
Cohen, Hull, & Srihari, 1991	95.54	4.46	0.00		2711	300
Cohen et al., 1991	97.10	2.90	0.00		1762	300
Knerr et al., 1992	90.30	9.70	0.00	7200	1800	
Lemarie, 1993	97.97	2.03	0.00	8783	7394	
Kim & Lee, 1994	95.40	4.60	0.00	4000	2000	166
Kim & Lee, 1994	95.85	4.15	0.00	4000	2000	166
Proposed method	96.05	3.95	0.00	4000	2000	166

4 Experimental Results

After training the proposed network with 4000 handwritten digits of the database, the recognition rate on the 2000 test data has been investigated. Table 1 shows the performances of the proposed method together with the results reported by some alternative methods. It also provides information about the size of the data sets used for training and testing, along with the scanning resolution in pixels per inch. Although some of the previous methods have achieved better recognition results, they used a highly tuned architecture or a much larger training data set than used here. The error rate of the proposed method is 3.95 percent, and in view of the elegant simplicity of the approach, this performance is remarkable and can stand comparison with one of the best results reported in the literature.

Previous work (Le Cun et al., 1990) showed that good generalization can be obtained only by designing a network architecture that contains a certain amount of a priori knowledge about the problem. The basic design principle is to minimize the number of free parameters that must be determined by the learning algorithm without overly reducing the computational power of the network. This principle increases the probability of correct generalization because it results in a specialized network architecture that has a reduced entropy. The proposed method of dynamic node splitting provides the automatic determination of the architecture to fit the problem at hand.

A thorough analysis on the recognition results reveals that most of the confusion makes sense. For example, class 0 has three instances of misclassification (2, 6 and 8), all of which are neighbors to the correct node in the map produced by the proposed scheme obtained in the simulation (see Figures 4c and 5). Figure 5 shows in detail how the topological orderings in pattern classes are preserved in the final map. As can be seen, the nodes for the same or similar classes form a sort of cluster topologically close to each other. For example, the class 9 consists of two clusters: one close to the class 4 and the other close to the class 7.

There is strong evidence that the recognizer made by the proposed neural network preserves the topological ordering of the input patterns of the handwritten digits. This can be of relative superiority to the conventional tree-structured approach to the node-splitting scheme. In order to improve the performance, we are attempting to incorporate the concept of k-nearest neighbor rule into the decision of the final class. Unfortunately the results are not exceptionally reliable. Work in progress seeks to improve reliability by introducing the rejection criteria to the decision process.

5 Concluding Remarks

This article proposes an elegant self-organizing neural network that might solve the complex classification problem. The key advantage of the network is that it automatically finds a network structure and size suitable for the classification of complex patterns through the ability of structure adaptation. Experimental results with handwritten digits have revealed that the proposed network efficiently and accurately classifies real patterns with high variations.

Handwritten digit recognition is not a new subject, and many good approaches have been published. The overwhelming majority of these approaches, however, use classical pattern recognition methods or backpropagation networks. This article demonstrates that very competitive recognition results can be obtained with the self-organizing map if it is suitably combined with a dynamic node-splitting scheme. Research is continuing with more difficult tasks such as handwritten Hangul (Korean script) recognition.

Acknowledgments

This work was supported in part by the offline character recognition consortium at KAIST (Korea Advanced Institute of Science and Technology). I thank Ari Milgram for his work on drafts of this article.

References

Cohen, E., Hull, J. J., & Srihari, S. N. (1991). Understanding handwritten text in a structured environment: Determining ZIP codes from addresses. *Int. J. Pattern Recognition and Artificial Intelligence* 5 (1&2): 221–264.

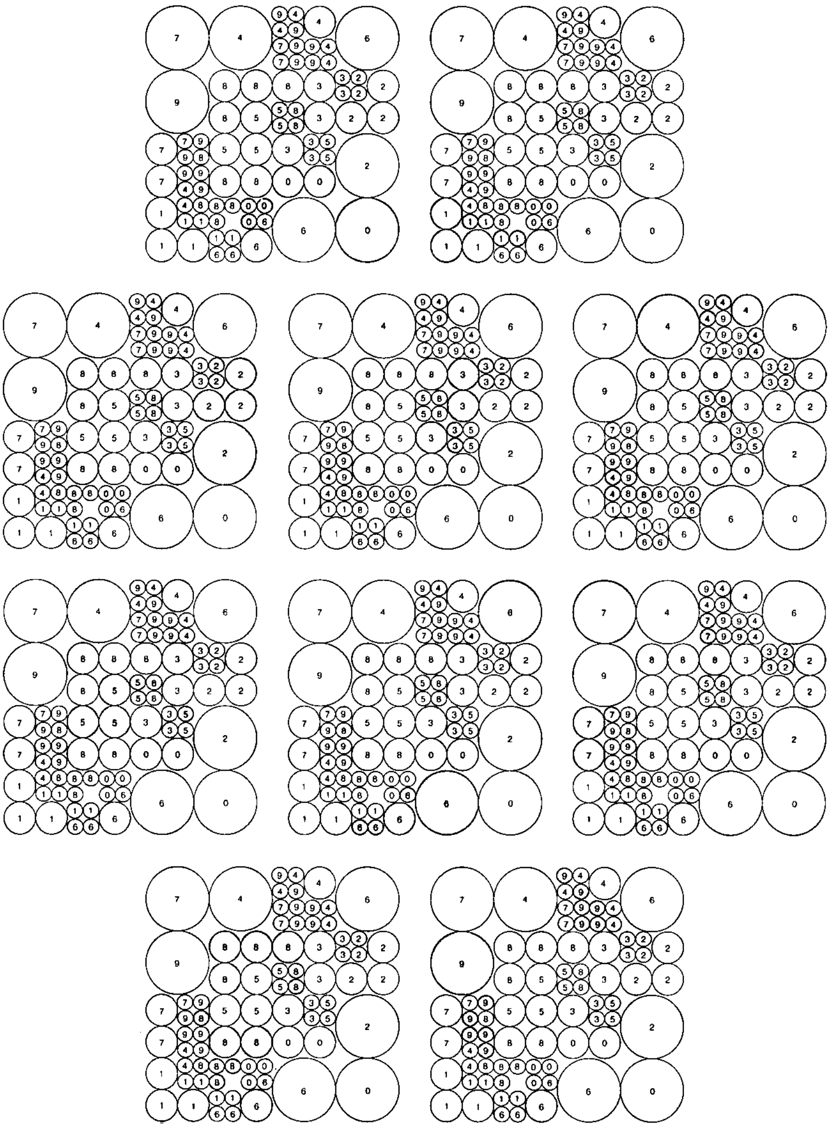


Figure 5: Maps preserving the topological orderings in the pattern classes.

Fritzke, B. (1994). Growing cell structures–a self-organizing network for unsupervised and supervised learning. *Neural Networks 7* (9):1441–1460.

Kim, Y. J., & Lee, S.W. (1994). Off-line recognition of unconstrained handwritten digits using multilayer backpropagation neural network combined with genetic algorithm. In *Proc. 6th Workshop on Image Processing and Understanding* (pp. 186–193). (In Korean.)

Knerr, S., Personnaz, L., & Dreyfus, G. (1992). Handwritten digit recognition by neural networks with single-layer training. *IEEE Trans. on Neural Networks 3* (6):962–968.

Kohonen, T. (1990). The self-organizing map. *Proceedings of the IEEE, 78* (9):1464–1480.

Koikkalainen, P., & Oja, E. (1990). Self-organizing hierarchical feature maps. In *Proc. Int. Joint Conf. Neural Networks* (pp. 279–284). San Diego, CA.

Krzyzak, A., Dai, W., & Suen, C. Y. (1990). Unconstrained handwritten character classification using modified backpropagation model. In *Proc. 1st Int. Workshop on Frontiers in Handwriting Recognition* (pp. 155–166). Montreal, Canada.

Lam, L., & Suen, C. Y. (1988). Structural classification and relaxation matching of totally unconstrained handwritten ZIP-code numbers. *Pattern Recognition 21* (1):19–31.

Le Cun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., & Jackel, L. D. (1990). Handwritten digit recognition with a back-propagation network. *Advances in Neural Information Processing Systems 2*:396–404.

Legault, R., & Suen, C. Y. (1989). Contour tracing and parametric approximations for digitized patterns. In A. Krzyzak, T. Kasvand & C. Y. Suen (Eds.), *Computer Vision and Shape Recognition* (pp. 225–240), Singapore: World Scientific Publishing.

Lemarie, B. (1993). Practical implementation of a radial basis function network for handwritten digit recognition. In *Proc. 2nd Int. Conf. Document Analysis and Recognition* (pp. 412–415). Tsukuba, Japan.

Li, T., Tang, Y. Y., & Fang, L. Y. (1995). A structure-parameter-adaptive (SPA) neural tree for the recognition of large character set. *Pattern Recognition 28* (3):315–329.

Mai, T., & Suen, C. Y. (1990). A generalized knowledge-based system for the recognition of unconstrained handwritten numerals. *IEEE Trans. on Systems, Man and Cybernetics 20* (4):835–848.

Martin, G. L. & Pittman, J. A. (1990). Recognizing hand-printed letters and digits. *Advances in Neural Information Processing Systems 2*:405–414.

Nadal, C. & Suen, C. Y. (1988). *Recognition of totally unconstrained handwritten digit by decomposition and vectorization.* (Tech. Rep.) Concordia University, Montreal, Canada.

Pratt, W. K. (1978). *Digital image processing,* New York: Wiley.

Sanger, T. D. (1991). A tree-structured adaptive network for function approximation in high-dimensional spaces. *IEEE Trans. on Neural Networks 2* (2):285–293.

Suen, C. Y., Nadal, C., Legault, R., Mai, T. A., & Lam, L. (1992). Computer recognition of unconstrained handwritten numerals. *Proceedings of the IEEE 80* (7):1162–1180.

Suen, C. Y., Nadal, C., Mai, T., Legault, R., & Lam, L. (1990). Recognition of handwritten numerals based on the concept of multiple experts. In *Proc. 1st Int. Workshop Frontiers in Handwriting Recognition* (pp. 131–144). Montreal, Canada.

Received March 13, 1996; accepted October 1, 1996.