Recognition of Unconstrained Handwritten Numerals by Doubly Self-Organizing Neural Network*

Sung-Bae Cho

Dept. of Computer Science, Yonsei University
134 Shinchon-dong, Sudaemoon-ku, Seoul 120-749, Korea
Tel: +82-2-361-2720 Fax: +82-2-365-2579 E-mail: sbcho@csai.yonsei.ac.kr

Abstract

In this paper we present an efficient pattern recognizer based on a self-organizing neural network which can adapt its structure as well as its weights. The network, called Doubly Self-organizing Neural Network (DSNN), makes use of 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. In order to verify the superiority of the DSNN, experiments with the unconstrained handwritten numeral database of Concordia University in Canada were conducted. The proposed method has produced 96.05% of the recognition rate, which we show better than those of several previous methods reported in the literature on the same database.

1. Introduction

In the past several decades, a wide variety of methods have been proposed to realize the perfect recognizer of handwritten numerals by computer. Many systems have been developed but more work is still required to be able to match human performance [1]. Recently, on the other hand, the emerging technology of neural networks has largely exploited to implement a system towards a pattern recognizer of such level.

Among several models, the multilayer perceptron has been recognized as a powerful tool for pattern classification problems. Their strength is on the discriminative power and the capability to learn and represent implicit knowledge, but they also have faced to several difficulties in real-world problems. One of the shortcomings is how to determine the size and structure of the network. To overcome the difficulty, several approaches based on the structure adaptation of the networks have been recently proposed [2, 3, 4].

In this paper we propose a competent pattern recognizer based on a self-organizing neural network which can adapt its structure as well as its weights. The network, called Doubly Self-organizing Neural Network (DSNN), 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. In the following we shall show how the network is constructed properly in the problem of handwritten numerals.

*This work was supported in part from the Center for Artificial Intelligence Research (CAIR), the Engineering Research Center (ERC) of Excellence Program.

2. Preprocessing

2.1. Database Used

In this paper, we have used the handwritten numeral database of Concordia University of Canada, which consists of 6000 unconstrained numerals originally collected from dead letter envelopes by the U.S. Postal Services at different locations in the U.S. The numerals of this database were digitized in bilevel on a 64×224 grid of 0.153mm square elements, giving a resolution of approximately 166 PPI [5]. Among the data, 4000 numerals were used for training and 2000 numerals for testing. Figure 1 shows some representative samples taken from the database. We can see that many different writing styles are apparent, as well as numerals of different sizes and stroke widths.

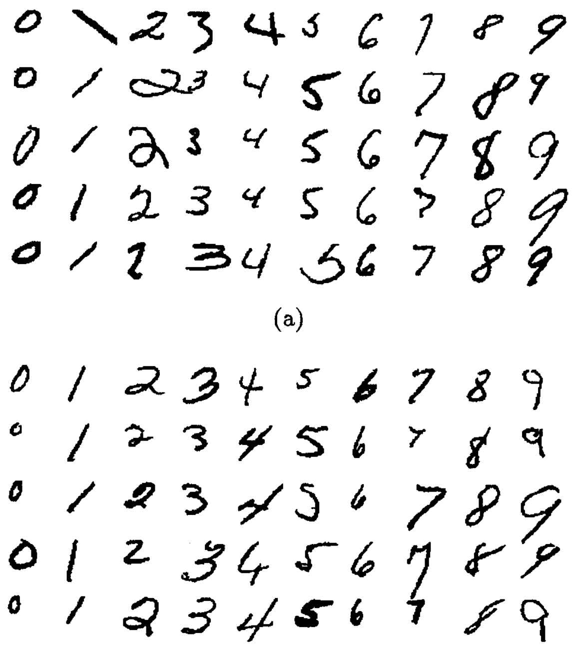
2.2. Feature Extraction

Numerals, whether handwritten or typed, are essentially line drawings, i.e., one-dimensional structures in a two-dimensional space. Thus, local detection of line segments seems to be an adequate feature extraction method. For each location in the image, information about the presence of a line segment of a given direction is stored in a feature map [6]. Especially, in this paper Kirsch masks have been used for extracting directional features, which were originally adopted in the [7].

Kirsch defined a nonlinear edge enhancement algorithm as follows [8]:

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

Proceedings of ICPR '96



(a)

(b)

Figure 1: Sample data for: (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).

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.

In this paper, input pattern is size-normalized by 16×16 and then directional feature vectors for horizontal(H), vertical(V), right-diagonal(R), and left-diagonal(L) directions are calculated from the size-normalized image with the Kirsch masks defined by Figure 3. Then, each 16×16 directional feature vector is compressed to 4×4 feature vector. In addition to the directional features, 4×4 compressed image has been used as global features.

5	5	5
-3	0	-3
-3	-3	-3

-3	-3	-3
-3	0	-3
5	5	5

(a)

-3	-3	5
-3	0	5
-3	-3	5

5	-3	-3
5	0	-3
5	-3	-3

(b)

-3	5	5
-3	0	5
-3	-3	-3

-3	-3	-3
5	0	-3
5	5	-3

(c)

-3	-3	-3
-3	0	5
-3	5	5

5	5	-3
5	0	-3
-3	-3	-3

(d)

Figure 3: Kirsch masks used for extracting four directional features; (a) horizontal direction; (b) vertical direction; (c) right-diagonal direction; (d) left-diagonal direction.

3. Doubly Self-organizing Neural Network

In this section we present a structure-adaptive self-organizing neural network which is able to simultaneously determine a suitable number of nodes and the connection weights between input and output nodes. The basic idea is very simple:

1. Start with a basic neural network (in our case, 4×4 map of which each node is fully connected to all input nodes).
2. Train the current network with the Kohonen's algorithm [9].
3. Calibrate the network using known I/O patterns to determine;
   (a) which node should be replaced with a submap of several nodes (in our case, 2×2 map), and
   (b) which node should be deleted.
4. Unless every node represents a unique class, goto 2.

Note that the step 3 positions the node in regions where the current network does not produce a unique label for the classification. In our model, the weights of new nodes are interpolated from those of neighboring nodes.

3.1. Structure and Adaptation of Network

The structure of the network is very similar to Kohonen's Self-Organizing Map (SOM) except the irregular connectivity in the map. Figure 4 shows an instance of the network where each node represents a unique class. Every node is connected to all the input nodes with corresponding weights. (Actually, this is the final network structure obtained for recognizing the handwritten numerals in our 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 for the match of x with the w_i may be the inner product $x^T w_i$, which is based on the *Euclidean distances* 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:

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

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 [9].

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, $0 < \alpha(t) < 1$.

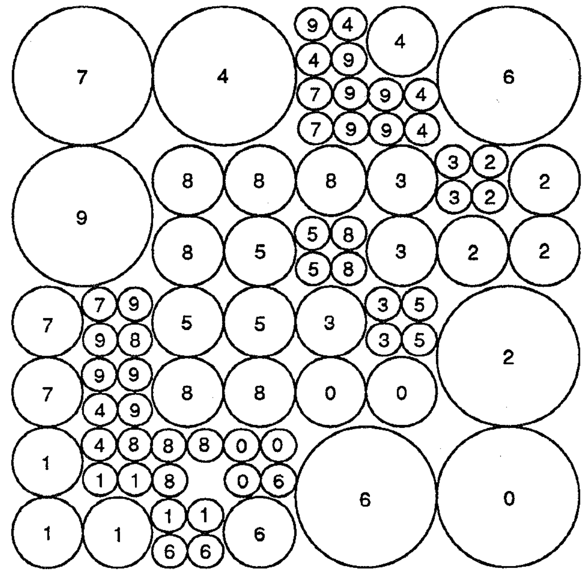


Figure 4: Doubly self-organizing neural network. The figure in each circle means the class that the corresponding node represents.

3.2. Insertion of New Nodes

After a constant number of adaptation steps, a node representing more than one class is replaced with several nodes. (In our case, we have 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 5 shows how the network structure changes as some nodes representing duplicated classes are replaced by several nodes having finer resolution.

3.3. Deletion of Nodes

The previous section gives us the way how to extend the network structure. A necessary consequence thereof 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. For this purpose, we delete some nodes that do not activate for a long while. In our example, only one node is deleted at the final map. (See

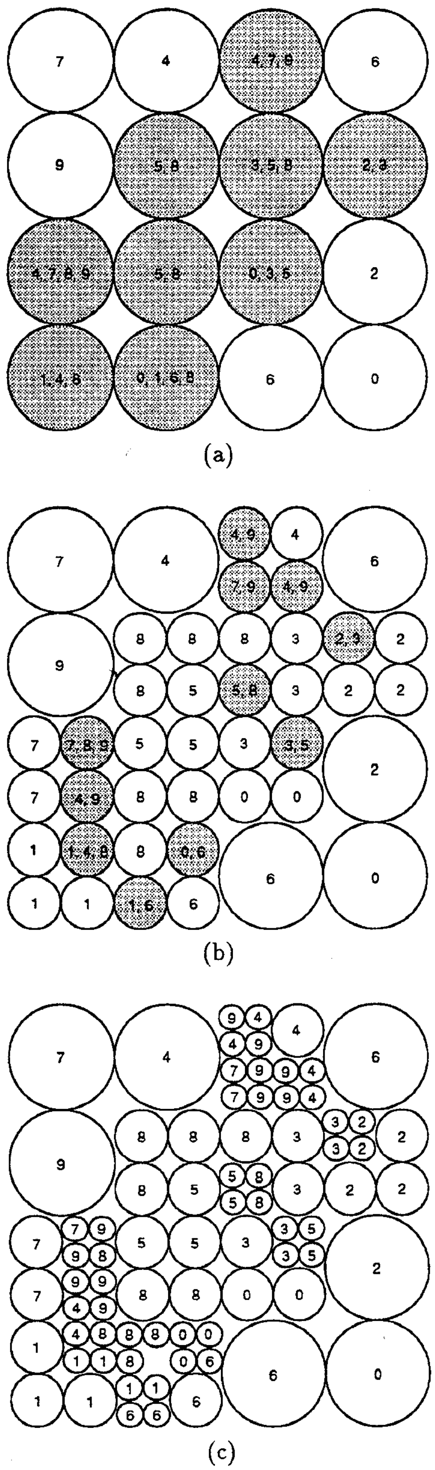


Figure 5: Map configurations changed through learning: (a) Initial status; (b) Intermediate status; (c) Final status.

Table 1: Comparisons of the presented method with the related (%).

Methods	Recog	Error	Reject	Reliability
Lam [10]	93.10	2.95	3.95	96.98
Nadal [11]	86.05	2.25	11.70	97.45
Legault [12]	93.90	1.60	4.50	98.32
Krzyzak [13]	86.40	1.00	12.60	98.85
Krzyzak [13]	94.85	5.15	0.00	94.85
Mai [14]	92.95	2.15	4.90	97.74
Suen [1]	93.05	0.00	6.95	100.00
DSNN	96.05	3.95	0.00	96.05

Figure 5(c).)

4. Experimental Results

After training the DSNN with 4000 handwritten numerals of the database, the recognition rate on the 2000 test data has been investigated. Table 1 shows the performance of the presented method along with the results produced by some previous methods reported on the same database. The reliability in the table is computed as the following equation:

$$\text{Reliability} = \frac{\text{Recog Rate}}{\text{Recog Rate} + \text{Error Rate}},$$

where the Substitution Error Rate is the portion of patterns which are classified incorrectly by the method. The error rate of the DSNN is 3.95%, which is a big improvement compared with those of the previous methods, but in terms of the reliability this result cannot be said as excellent. Further work is in progress towards emphasizing this aspect by introducing the reject criteria to the decision process.

Table 2 reports the confusion matrix for the proposed method with respect to the data set. It can be seen from this table that most of the confusion makes sense: For example, 0 has three instances of misclassification, 2, 6 and 8, all of which are just neighbors to the correct node in the map produced by the DSNN obtained in the simulation. (See Figure 5(c).)

This is a strong evidence that the classifier made by the proposed neural network preserves the topological ordering of the input patterns, the handwritten numerals. In order to improve the performance in this point, we are attempting to incorporate the concept of k-nearest neighbor rule into the decision of the final class.

5. Concluding Remarks

In this paper, we have presented a novel self-organizing neural network that can adapt the structure as well as the weights. We have found that the

Table 2: Confusion matrix for the proposed method.

Class	0 / 5	1 / 6	2 / 7	3 / 8	4 / 9	Recognized
0	197	0	1	0	0	
	0	1	0	1	0	98.5%
1	0	192	4	1	0	
	0	2	0	1	0	96.0%
2	0	0	189	3	2	
	1	0	1	3	1	94.5%
3	0	2	2	190	0	
	0	0	1	3	2	95.0%
4	0	3	0	0	193	
	0	1	1	1	1	96.5%
5	1	0	0	1	0	
	193	2	1	2	0	96.5%
6	2	1	0	0	0	
	2	191	0	4	0	95.5%
7	0	6	0	0	0	
	0	0	193	1	0	96.5%
8	1	0	2	2	0	
	2	5	0	188	0	94.0%
9	1	0	1	0	0	
	1	1	1	0	195	97.5%
Average						96.05%

proposed neural network 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 numerals have revealed that the proposed network is very effective for the classification of the real patterns with high variations. The further works are under going with the more difficult task such as handwritten Hangul (Korean script) recognition.

Acknowledgments

The author would like to thank Ms. Kyoungmi Lee for her support to the implementation of the algorithms and the simulation performed for this research.

References

[1] C.Y. Suen, C. Nadal, R. Legault, T.A. Mai and L. Lam, "Computer recognition of unconstrained handwritten numerals," *Proceedings of the IEEE*, vol. 80, no. 7, pp. 1162–1180, 1992.

[2] T.D. Sanger, "A tree-structured adaptive network for function approximation in high-dimensional spaces," *IEEE Trans. on Neural Networks*, vol. 2, no. 2, pp. 285–293, March 1991.

[3] B. Frizke, "Growing cell structures–a self-organizing network for unsupervised and supervised learning," *Neural Networks*, vol. 7, no. 9, pp. 1441–1460, 1994.

[4] T. Li, Y.Y. Tang and L.Y. Fang, "A structure-parameter-adaptive(SPA) neural tree for the recognition of large character set," *Pattern Recognition*, vol. 28, no. 3, pp. 315–329, 1995.

[5] C.Y. Suen, C. Nadal, T. Mai, R. Legault and L. Lam, "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, 1990.

[6] S. Knerr, L. Personnaz and G. Dreyfus, "Handwritten digit recognition by neural networks with single-layer training," *IEEE Trans. on Neural Networks*, vol. 3, no. 6, pp. 962–968, 1992.

[7] S.W. Lee, "Multilayer cluster neural network for totally unconstrained handwritten numeral recognition," *Neural Networks*, vol. 8, no. 5, pp. 783–792, 1995.

[8] W.K. Pratt, *Digital Image Processing*, New York: Wiley, 1978.

[9] T. Kohonen, "The self-organizing map," *Proceedings of the IEEE*, vol. 78, no. 9, pp. 1464–1480, 1990.

[10] L. Lam and C.Y. Suen, "Structural classification and relaxation matching of totally unconstrained handwritten ZIP-code numbers," *Pattern Recognition*, vol. 21, no. 1, p. 19–31, 1988.

[11] C. Nadal and C.Y. Suen, "Recognition of totally unconstrained handwritten digit by decomposition and vectorization," *Technical Report*, Concordia University, Montreal, Canada, 1988.

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

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

[14] T. Mai and C.Y. Suen, "A generalized knowledge-based system for the recognition of unconstrained handwritten numerals," *IEEE Trans. on Systems, Man and Cybernetics*, vol. 20, no. 4, pp. 835–848, 1990.