수정된 카오스 신경망을 이용한 무제약 서체 숫자 인식

Recognition of Unconstrained Handwritten Numerals using Modified Chaotic Neural Networks

최한고, 김상희, 이상재
Han-Go Choi, Sang-Hee Kim, Sang-Jae Lee

요 약

본 논문은 수정된 카오틱 신경망(MCNN)을 이용하여 완전 무제약 서체 숫자 인식을 다루고 있다. 카오틱 신경망(CNN)의 동적 특성과 학습과정을 강화함으로써 복잡한 패턴인식 문제를 해결할 수 있는 유용한 신경망으로 수정하였다. MCNN은 신경망 구조와 뉴런 자체가 높은 차수의 비선형 동적특성을 갖고 있으므로 복잡한 서체 숫자를 분류할 수 있는 적합한 신경망이다. 숫자 확인은 원래의 숫자 이미지로부터 특징을 추출하고 MCNN에 근거한 분류기를 이용하여 숫자를 인식한다. MCNN 분류기의 성능은 Canada, Montreal의 Concordia 대학의 숫자 데이터 베이스로 평가하였다. 인식성능의 상대적인 비교를 위해 MCNN 분류기는 리커런트 신경망(RNN) 분류기와 비교하였다. 실험결과에 의하면 인식율은 98.0%이었으며, 이는 MCNN 분류기가 같은 데이터 베이스에 대해 발표되었던 다른 분류기와 RNN 분류기보다 성능이 우수함을 나타낸다.

Abstract

This paper describes an off-line method for recognizing totally unconstrained handwritten digits using modified chaotic neural networks(MCNN). The chaotic neural networks(CNN) is modified to be a useful network for solving complex pattern problems by enforcing dynamic characteristics and learning process. Since the MCNN has the characteristics of highly nonlinear dynamics in structure and neuron itself, it can be an appropriate network for the robust classification of complex handwritten digits. Digit identification starts with extraction of features from the raw digit images and then recognizes digits using the MCNN based classifier. The performance of the MCNN classifier is evaluated on the numeral database of Concordia University, Montreal, Canada. For the relative comparison of recognition performance, the MCNN classifier is compared with the recurrent neural networks(RNN) classifier. Experimental results show that the classification rate is 98.0%. It indicates that the MCNN classifier outperforms the RNN classifier as well as other classifiers that have been reported on the same database.

Keywords : Handwritten digits, chaotic neural networks, recurrent networks, pattern classifier

I. Introduction

Automatic recognition of handwritten characters or numerals is a typical field of pattern classification methods. A number of different methods have been studied and applied to realize the higher rate recognizer of handwritten documents such as bank checks and automatic mail sorting[1]. However, recognition of handwritten numerals that still lacks a satisfactory solution is technically difficult because of the high variability of the scanned image. This is caused by the peculiar writing style of different persons, the context of the digit, diverse of writing devices and media. These lead to scanned digits of different size and slant, and strokes that vary in width and shape.

In the past several decades, syntactic and statistical methods are applied to solve handwritten pattern recognition problems[2]. Over the last ten years or so, neural network based approaches have been more popular methods and steadily gaining better performance. Neural networks are promising models to discriminate complex patterns since they consist of distributed neurons to process nonlinearity and have the ability to learn from its environment. For the classification of more complex patterns, dynamic neural networks such as time-delay neural networks and recurrent neural networks are adequate models. They use feedback loops or delayed elements as memories in order to process temporal information and can perform well the recognition of complex patterns.

This paper presents an off-line method for recognizing totally unconstrained and isolated handwritten numerals using modified chaotic neural networks(MCNN). Chaotic neurons were modeled with biological neuron models that have highly nonlinear dynamic characteristics. Since the previously proposed CNN was trained using Hebbian learning rule inside layers and backpropagation learning rule between layers, it was not adequate learning method for succeeding data patterns. However, it has characteristics of superior nonlinearity, large data storage, and low probability of falling into the local minimum. Therefore, the CNN model is modified to be a useful network for solving dynamic pattern problems, and applied to the classification of unconstrained digits. Since a single network classifier cannot provide a satisfactory decision for complex pattern recognition, multiple network classifiers are generally used to achieve higher classification accuracy. This is because group decisions may reduce errors drastically and achieve a higher performance, thus they are better than any individual's. Since the proposed MCNN, however, has superior dynamic characteristics than other networks, a single MCNN classifier is used.

Like most of pattern recognition systems, the recognition process is divided into two steps; preprocessing stage to extract useful features and subsequent classification stage to identify correct patterns. In this paper, we preprocess the original numeric images using the Kirsch mask for extracting features, and then the MCNN using these features as inputs performs the recognition of numerals. In order to verify the recognizing behavior of the MCNN based classifier, we perform experiments with the totally unconstrained handwritten numeric database of Concordia University, Montreal, Canada as test images. For the relative evaluation of numeral recognition, the MCNN classifier is compared with the Elman type RNN classifier. It also compares with the results of other methods that have been reported using the same database. Experimental results show that the MCNN classifier improves numeral discrimination, and outperforms the RNN and other classifiers with respect to the correct recognition rate.

II. Feature Extraction

All handwritten or printed numerals can be considered as collection of short-segment lines. Thus, local detection of line segments seems to be an adequate feature extraction method. The first-order differential edge detector could be adequate for local detection of a line segment. Among several detectors, the Kirsch edge detector has been known to detect four directional edges more accurately than other edge detectors because the Kirsch edge detector considers all eight neighbors and the frequency difference between numeric itself and background image is small. Kirsch defined a nonlinear edge enhancement algorithm as follows[6].

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

$$S_k = A_k + A_{k+1} + A_{k+2} \quad (2)$$

$$T_k = A_{k+3} + A_{k+4} + A_{k+5} + A_{k+6} + A_{k+7} \quad (3)$$

In above equations, $G(i,j)$ is the gradient of a pixel (i,j), the subscripts of A are evaluated modulo 8, and $A_k (k=0,1,\ldots,7)$ is eight neighbors of the pixel (i,j) as shown Fig. 1.

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

Fig. 1. Eight neighbors of a pixel (i,j)

For each location in the image, information about the presence of a line segment for a given direction is stored in a feature map. In this paper, the original digit image has been size-normalized to 256x256 image and then 16x16 image. In order to train the spatial dependencies in the digit image, directional feature vectors for horizontal(H), vertical(V), right-diagonal(R), and left-diagonal(L) directions from the size-normalized image are calculated as follows:

$$G(i,j)_H = \max(|5S_0 - 3T_0)|, |5S_4 - 3T_4|) \quad (4)$$

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

$$G(i,j)_R = \max(|5S_1 - 3T_1)|, |5S_5 - 3T_5|) \quad (6)$$

$$G(i,j)_L = \max(|5S_3 - 3T_3)|, |5S_7 - 3T_7|) \quad (7)$$

Fig. 2 shows the Kirsch masks used for calculating directional feature vectors. The scale factors such as 5 and 3 in the figure emphasize the directional feature to be extracted and diminish the unwanted directional feature. Each 16x16 directional feature map computed using above equations is compressed to 4x4 feature map by accumulating pixels of each 4x4 subregion, and used this compressed image as a local feature. Furthermore, in order to consider the global characteristics of image, we simply compressed the 16x16 normalized image into a 4x4 image, and used this compressed image as a global feature. Therefore, final features consist of 5x4x4 feature vectors for each digit image; 4x4x4 local directional features, and 1x4x4 global feature. The gray level of these features is normalized in amplitude with dividing each pixel by the maximum gray level of the given feature vector. These amplitude-normalized features are used as input patterns to neural networks. Fig. 3 shows the overview of extracting image features.

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

(a) Horizontal

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

(b) Vertical

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

(c) Right-diagonal

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

(d) Left-diagonal

Fig. 2. Scale factors of Kirsch masks

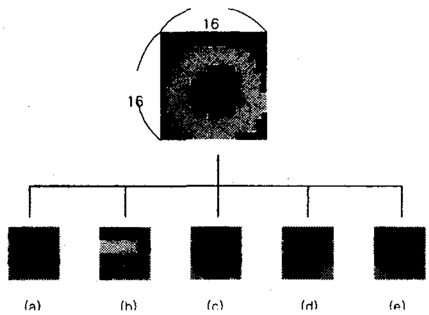


Fig. 3. Overview of extracting image features
(a)-(d) Local features using Kirsch mask,
(e) Global feature using simple compression

III. Chaotic Neural Networks

The primitive chaotic model was the Hodgkin-Huxley equation. Caianiello[3] and Nagumo-Sato[4] modified this model for making chaotic neural networks. Aihara et al. made a discrete time model with continuous output, and applied this model to chaotic neural networks[5]. The chaotic response of neuron model generally gives adverse effect on optimization problems, but the transient chaos of neuron model could be beneficial to overcome the local minimum problem. Ishi et al modified for information processing[8]. Even though some modifications on chaotic neuron models have been made, those previously proposed chaotic neuron models are still complicate to apply in neural network and need more dynamic characteristic in neuron itself and learning algorithm. In spite of these problems, the chaotic neural network has many possibilities in the applications on optimization and classification. In this section, the traditional chaotic neuron model is studied for analyzing the characteristics, and a novel modified chaotic neuron model is presented for simplifying model and enforcing dynamic characteristics[9].

3.1 Chaotic neuron model and network

The conventional chaotic neuron model, suggested by Nagumo and Sato[4], has two different types of inputs simultaneously; input from same layer and extraneous input, and also has a refractory term, which is a self-feedback. The refractory term performs effective dynamic characteristics through repeated signal controlling as one of three terms, which affect the output of the chaotic neuron. The neuron model is shown in Fig. 4.

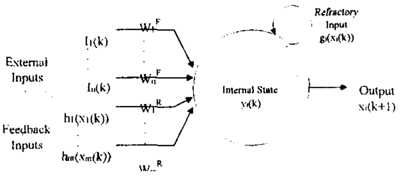


Fig. 4. Chaotic neuron unit

Generally, the dynamics of the ith chaotic neuron in networks at discrete time $k+1$ is

$$x_i(k+1)=f_N\Big[\sum_{j=1}^{n} w_{ij}^F \sum_{r=0}^{k} K_s^r I_j(k-r)+\sum_{j=1}^{m} w_{ij}^R \sum_{r=0}^{k} K_m^r h_j(x_i(k-r))-\alpha\cdot\sum_{r=0}^{k} K_r^r g_i(x_i(k-r))-\theta_i\Big] \quad (8)$$

where $f_N(\cdot)$ is a sigmoid function, w_{ij}^F and w_{ij}^R are coupling coefficients from the jth external neuron and the jth feedback neuron to the ith neuron, respectively. $I_j(k-r)$ is the strength of the jth externally applied input at time $k-r$, $h_j(x_i(k-r))$ is a transfer function of the axon connected on the jth chaotic neuron, and $g_i(x_i(k-r))$ is a refractory function of the ith chaotic neuron, usually an identity function. Both n and m are the numbers of external and feedback inputs applied to the chaotic neuron. The decay parameters, K_s^r, K_m^r, and K_r^r are the damping factors of the external, feedback, and refractoriness, respectively. Those decay parameters were assumed the same values as, K, for simplifying neuron model. The constant α is a positive parameter, and the θ_i is the threshold of the ith chaotic neuron.

Aihara deals with the ith chaotic neuron equation in a reduced form by dividing feedback, external, and refractory terms[7]. Each term is expressed as

$$\xi_i(k+1)=K\cdot\xi_i(k)+\sum_{j=1}^{n} w_{ij}^F I_j(k) \quad (9)$$

$$\eta_i(k+1)=K\cdot\eta_i(k)+\sum_{j=1}^{m} w_{ij}^R h_j(f(y_i(k))) \quad (10)$$

$$\zeta_i(k+1)=K\cdot\zeta_i(k)-\alpha g_i(f_i(y_i(k)))-\theta_i(1-k) \quad (11)$$

If the inertial state of a chaotic neuron at time $t+1$ is expressed as follows,

$$y_i(k+1)=\xi_i(k+1)+\eta_i(k+1)+\zeta_i(k+1). \quad (12)$$

Eq. (12) can be written as follows.

$$y_i(k+1)=K(\xi_i(k)+\eta_i(k)+\zeta_i(k))+\sum_{j=1}^{n} w_{ij}^F I_j(k)+\sum_{j=1}^{m} w_{ij}^R h_j(f_N(y_i(k)))-\alpha g_i(f_N(y_i(k)))-\theta_i(1-k) \quad (13)$$

Since $y_i(t)=\xi_i(t)+\eta_i(t)+\zeta_i(t)$, eq. (13) can be expressed as

$$y_i(k+1)=Ky_i(k)+\sum_{j=1}^{n} w_{ij}^F I_j(k)+\sum_{j=1}^{m} w_{ij}^R h_j(f_N(y_i(k)))-\alpha g_i(f_N(y_i(k)))-\theta_i(1-k). \quad (14)$$

The chaotic neural networks proposed by Aihara have complicate structure as Fig. 5.

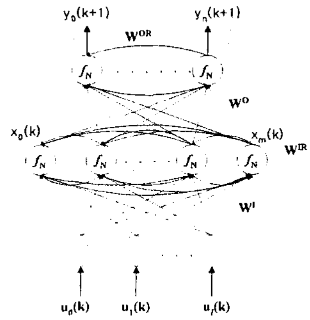


Fig. 5. Chaotic neural networks model

This network adopted two kinds of learning rules, backpropagation learning rule between layers and Hebbian learning rule for interlayer. In order to apply the continuous Hopfield neural network structure to the recurrent inputs, Aihara et al. define the symmetric structure of recurrent weights as $w_{ij}^R = w_{ji}^R$, $w_{ii}^R = 0$. This neural network uses two kinds of learning rules in same network. Since the structure decreases the efficiency of learning and the dynamic characteristics of network, this model is not appropriate for modeling dynamic systems.

3.2 Modified Chaotic Neural Networks

Although the chaotic neuron model inherently has robust dynamic characteristics, the traditional chaotic neural networks, decrease the dynamic characteristics in the structure and the learning rules. They used the backpropagation learning rule for the forward inputs between layers, and the Hebbian learning rule (the continuous Hopfield learning algorithm) for the recurrent inputs in interlayer. These learning rules may appropriate for the static patterns but not for the dynamic system applications such as forecasting, identifications, signal processing, and dynamic system control. In this paper, the structure of the CNN is modified, and the new learning rule is proposed for enhancing the dynamic characteristics.

Modified chaotic neural network is a globally coupled neural network. Each chaotic neuron unit uses the chaotic neurons that are globally coupled with present and past outputs of chaotic neuron units. Modified chaotic neural networks in Fig. 6 have two different coupling coefficients (weights) for both directions among the neurons of interlayer, and forward direction between layers. The connecting weights between layers are one directional. This connection weights in interlayer is defined as nonsymmetric form, $w_{ij}^{OR} \neq w_{ji}^{OR}$, $w_{ii}^{OR} \neq 0$ and $w_{ij}^{IR} \neq w_{ji}^{IR}$, $w_{ii}^{IR} \neq 0$. This structure is similar with fully recurrent neural networks.

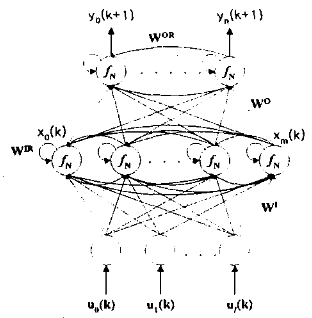


Fig. 6. Structure of modified CNN

Consider Fig. 6, where for each discrete time k. $u_i(k)$ is the ith input, $S_j^H(k)$ is the weighted sum of inputs and refractory input to the jth neuron in the hidden layer, $x_j(k)$ is the output of the jth neuron in the hidden layer, K is the refractory parameter of a chaotic neuron, and $f_N(\cdot)$ is a nonlinear sigmoid function. Both w^I and w^{IR} represent the weight vector between input and hidden layers, and inter-connecting weight vector in the hidden layer.

The weighted sum of the jth neurons in the hidden layer is as follows;

$$S_j^H(k) = \sum_{i=1}^{l} w_{ij}^I u_i(k) + \sum_{q=1}^{m} w_{qj}^{IR} x_q(k-1) + K S_j^H(k-1) \qquad (15)$$

The jth neuron's output of the hidden layer is as follow;

$$x_j(k) = f_N[S_j^H(k)] \qquad (16)$$

Consider Fig. 6, where for each discrete time k, $y_p(k)$ is the pth output of output neuron, $S_p^O(k)$ is the weighted sum of inputs and refractory input to the pth output neuron in the output layer, $x_j(k)$ is the output of the jth neuron in the hidden layer, K is the refractory parameter of a chaotic neuron and $f_N(\cdot)$ is a nonlinear sigmoid function. Both w^O and w^{OR} represent the weight vector between hidden and output layers, and inter-connecting weight vector in the output layer.

The weighted sum of the pth neurons in the output layer is as follows;

$$S_p^O(k) = \sum_{j=1}^{m} w_{jp}^O x_j(k) + \sum_{r=1}^{n} w_{rp}^{OR} y_r(k-1) + KS_p^O(k-1) \quad (17)$$

$$y_p(k) = f_N[S_p^O(k)] \quad (18)$$

Using eqs. (15) and (16), the weighted sum of a neuron in the output layer (eq. 17) can be defined as follows;

$$\begin{aligned} S_p^O(k) = & \sum_{j=1}^{m} w_{jp}^O f_N[\sum_{i=1}^{l} w_{ij}^I u_i(k) + \sum_{q=1}^{m} w_{qj}^{IR} x_q(k-1) \\ & + KS_j^H(k-1)] + \sum_{r=1}^{n} w_{rp}^{OR} y_r(k-1) + KS_p^O(k-1) \\ = & \sum_{j=1}^{m} w_{jp}^O f_N[\sum_{i=1}^{l} w_{ij}^I u_i(k) + \sum_{q=1}^{m} w_{qj}^{IR} f_N[\sum_{i'=1}^{l} u_{i'}(k-1) \\ & + \sum_{j=1}^{m} w_{jq}^{IR} x_j(k-2) + KS_q^H(k-2)]] \\ & + \sum_{r=1}^{n} w_{rp}^{OR} f_N[\sum_{j'=1}^{m} w_{j'r}^O x_{j'}(k-1) + \sum_{r'=1}^{n} w_{r'r}^{OR} y_{r'}(k-2) \\ & + kS_r^O(k-2)] + KS_p^O(k-1) \end{aligned} \quad (19)$$

$$O_p(k) = NF(u(l), x(l), y(l); l \le k) \quad (20)$$

where $O_p(k)$ is the pth output of the chaotic neural network, $NF(\cdot)$ is a nonlinear function and it represents a nonlinear dynamic mapping chaotic neural networks.

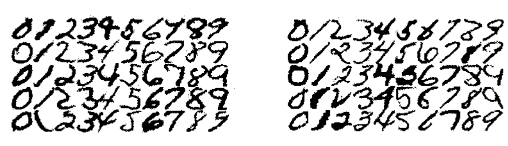
This neural network model is a globally coupled with present and past inputs and outputs of all neurons. Therefore, this network could model any complex nonlinear dynamic system with globally coupled. In chaotic neural network, connecting coefficients between layers and inside layer were updated by error backpropagation algorithm. In this application, the slope of a sigmoid function is chosen as 1 to reduce the chaotic response of a neuron.

The Elman type RNN is used to compare the recognition performance of the proposed network. It consists of three layers. In the input layer, it has external inputs of image features and additional inputs that are fully fed from all neurons of the hidden layer. All the units in a layer are connected to all the units in the following layers. It differs from the Jordan's RNN in which additional inputs are fed from neurons of the output layer. The sigmoid function and error backpropagation algorithm are used as the activation function of the node and learning algorithm.

IV. Experimental Results

4.1 Database

We used the handwritten numeric database of Concordia University, Montreal, Canada[10] as test numerals. The database consists of isolated and unconstrained numeral images originally collected from dead letter envelopes by the U.S. Postal Services at different locations in the United States. The numerals of this database were digitized in bilevel on a 64x224 grid of 0.153[mm] square elements, giving a resolution of approximately 166 PPI. The number of learning and test data is 4000 and 2000 numerals, respectively and some of the learning data are duplicated. The sample of the learning data is not included in the test data set. Fig. 7 shows some representative samples taken from the numeric database. Many different numeral sizes and stroke widths as well as writing styles are apparent. Each numeric image has been preprocessed to obtain the local and global features, explained in section 2, and then these features are used as inputs of the neural networks.



(a) Training data (b) Test data

Fig. 7. Sample numeric data

4.2 Structure and Parameters of Neural Networks

The structure of the RNN consists of 160 units in the input layer (80 external inputs of 64 local and 16 global features, and 80 feedbacks from the hidden layer). The number of units of the hidden and the output layers is 80 and 10 units, respectively. The MCNN consists of 80 neurons of only external inputs at the input layer, 80 and 1 neurons in the hidden and output layers. Total number of weights to be updated is 13600 in the RNN and 12881 in the MCNN. Both networks have similar number of weights. The parameters of two networks are as follows; learning rate=0.15, 0.3 and momentum rate=0.6, 0.6 in the RNN and the MCNN. The refractory rate of the MCNN is set to 0.6. The structure and parameters are determined experimentally in order to maximize the recognition rate. The neural networks are trained for 500 iterations since the error does not reduce rapidly and keep the almost same value after those iterations. The final errors are 0.00987 in the RNN and 0.00273 in the MCNN.

4.3. Recognition Results

Table 1 shows recognition rates and error rates for the RNN and the MCNN on the training and test data sets. Comparing with the RNN, the MCNN achieves 1.4% and 1.9% improved recognition performance on the training and test data sets respectively. The error rate of the MCNN is about two times less than that of the RNN, which means significant reduction in the recognition of unconstrained numerals. This result confirms that the proposed MCNN with smaller number of coupling coefficients performs better than the RNN. Table 2 and 3 present the confusion matrices of the RNN and the MCNN on the test data set. In these tables, we can identify the discrimination performance of each network. The MCNN classifier outperforms the RNN classifier in the recognition of all numerals, ranging from 1% improvement of recognition in the number of '5' to 3% improvement in the number of '3'.

Table 1. Recognition rates on the training and test data sets

	Recognition rate (Error rate)	
	Training set	Test set
RNN	97.90% (2.10%)	96.10% (3.90%)
MCNN	99.30% (0.70%)	98.00% (2.00%)

Table 2. Confusion matrix for the RNN

Digit	0	1	2	3	4	5	6	7	8	9	Substituted	Recognized
0	192			1			2	1	4		4.0	96.0
1	1	194	1		1			2		1	3.0	97.0
2		1	192	4		1			1	1	4.0	96.0
3	1		2	191			1		5		4.5	95.5
4	1	2		1	192	1		3			4.0	96.0
5	1			3		193	2			1	3.5	96.5
6	1	1				2	191		4	1	4.5	95.5
7		1	3		2			192		2	4.0	96.0
8	2	4		1					192	1	4.0	96.0
9	1	1		2				2	1	193	3.5	96.5
Average											3.9	96.1

Table 3. Confusion matrix for the proposed MCNN

Digit	0	1	2	3	4	5	6	7	8	9	Substituted	Recognized
0	195	1					2			2	2.5	97.5
1		198						1		1	1.0	99.0
2		1	196	2				1			2.0	98.0
3				197		1			2		1.5	98.5
4	1	2			195	1		1			2.5	97.5
5	1			1		195	2		1		2.5	97.5
6	1						196		2	1	2.0	98.0
7	1	2						195		2	2.5	97.5
8				2			1		197		1.5	98.5
9		1						2	1	196	2.0	98.0
Average											2.0	98.0

Table 4 compares the performance of the proposed method along with the results of other methods experimented on the same database[11]. All methods in the table use the same number of numerals for training and test data sets from the Concordia database. The reliability in the table is computed as the following equation.

$$Reliability = \frac{Recognition\ rate}{Recognition\ rate + Error\ rate} \times 100$$

where the error rate is the portion of patterns which are classified incorrectly by the method.

From the experimental results, it is verified that the proposed method outperforms previous other methods in terms of the recognition rate. The proposed MCNN classifier shows relatively lower error rate and higher reliability compared with those of the other methods. The reliability may be improved by introducing the reject criteria to the decision process in the system. We also find that the proposed network performs well using only one output unlike multiple outputs in most of the previous networks. Thus, the modified MCNN could be a more useful model for classifying the unconstrained handwritten numerals.

Table 4. Comparison between the proposed method and other methods

Methods	Recognized	Substituted	Rejected	Reliability
Kim	95.85	4.15	0.00	95.85
Krzyzak	94.85	5.15	0.00	94.85
Lam	93.10	2.95	3.95	96.98
Legault	93.90	1.60	4.50	98.32
Mai	92.95	2.15	4.90	97.74
Nadal	86.05	2.25	11.70	97.45
Suen	93.05	0.00	6.95	100.00
Cho	96.05	3.95	0.00	96.05
Lee	97.30	2.70	0.00	97.30
Proposed	98.00	2.00	0.00	98.00

V. Conclusions

This paper presents the recognition of totally unconstrained handwritten numerals using the modified chaotic neural networks. The CNN has inherently highly nonlinear dynamics and chaotic properties. The network consists of neurons with self-feedbacks and mutual connections with other neurons in a layer. Thus, it has superior dynamic characteristics comparing with other networks since it processes both past values in neurons themselves and mutual couplings among neurons as iteration proceeds. In this paper, the CNN has been modified to be a useful network for solving dynamic pattern problems and successfully applied to enhance the recognition rate of unconstrained handwritten numerals. Compared with other classifiers such as recurrent neural networks and several previous methods reported in the literature, the proposed MCNN classifier shows the best performance in recognition rates when evaluated on the same database. Thus, the proposed MCNN might be a more appropriate pattern classifier than other multilayered dynamic networks for the discrimination of handwritten numerals. Although our work focuses on recognition of handwritten numerals, we expect that the method can be easily expanded to cope with more complex classification tasks such as the recognition of handwritten characters.

접수일자 : 2000. 11. 15 수정완료 : 2000. 12. 21

본 연구는 1999년도 금오공과대학교 학술연구비 지원에 의하여 연구된 논문임.

References

[1] Y. S. Huang and C. Y. Suen, "A method of combining multiple experts for the recognition of unconstrained handwritten numerals", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 17, no. 1, pp. 90-94, 1995.

[2] C. Y. Suen, C. Nadal, R. Legault, T. A. Mai, and L. Lam, "Computer recognition of unconstrained handwritten numerals", *Proc. IEEE*, vol. 80, pp. 1162-1180, 1992.

[3] E. R. Caianiello and A. DeLuca, "Decision equation for binary system: Application to neuronal behavior", Kybernetik, vol. 3, pp. 33-40, 1966.

[4] J. Nagumo and S. Sato, "On response characteristics of a mathematical neuron model", Kybernetik, vol. 10, pp. 155-164, 1972.

[5] L. Chen and K. Aihara, "Chaotic simulated annealing by a neural network model with transient chaos", Neural Networks, vol. 8, no. 6, pp. 915-930, 1995.

[6] W. K. Pratt, *Digital Image Processing*,

New York, Wiley, 1978.
[7] K. Aihara, T. Takabe, and M. Toyoda, "Chaotic neural networks", Physics Letters A, 144, pp. 333-340, 1990.
[8] S. Ishi, K. Fukumizu, and S. Watanabe, "A network of chaotic elements for information processing", Neural Networks, vol. 9, no. 1, pp. 25-40, 1996.
[9] S. H. Kim, H. G. Choi, and W. W. Park, "Trajectory control of robotic manipulators using chaotic neural networks", ICNN, vol. 3, pp. 1685-1688, 1997.
[10] C. Y. Suen, C. Nadal, T. Mai, R. Legault, and L. Lam, "Recognition of handwritten numerals based on the concept of multiple experts", *Proc. 1st Int. Wkshp. Frontiers Handwriting Recognition*, Montreal, Canada, pp. 131-133, Apr. 1990.
[11] S. B. Cho, "Neural-network classifiers for recognizing totally unconstrained handwritten numerals", *IEEE Trans. on Neural Networks*, vol. 8, no. 1, pp. 43-53, Jan. 1997.

최한고(Han-Go Choi)
正會員
1979년 경북대학교
전자공학과(공학사)
1988년 Univ. of Florida
전기공학과(공학석사)
1992년 Univ. of Florida
전기공학과(공학박사)
1979년-1986년 국방과학연구소 근무
1993년-현재 금오공과대학교 전자공학부 부교수
관심분야: 디지털 신호처리, 신경회로망, 의용전자

김상희(Sang-Hee Kim)
正會員
1983년 홍익대학교
전기공학과(공학사)
1985년 연세대학교
전기공학과(공학석사)
1992년 Texas A&M University
전기공학과(공학박사)
1993년-현재 금오공과대학교 전자공학부 부교수
관심분야 : 제어계측, 신경회로망, 카오스

이상재(Sang-Jae Lee)
正會員
1980년 연세대학교
전기공학과(공학사)
1982년 연세대학교
전기공학과(공학석사)
1990년 연세대학교
전기공학과(공학박사)
1991~92년 Stanford University Post-Doc.
1983~현재 금오공과대학교 전자공학부 교수
관심분야 : 적응제어, 신경망 및 카오스 이론