Off-Line Recognition of Totally Unconstrained Handwritten Numerals Using Multilayer Cluster Neural Network

Seong-Whan Lee

Abstract—In this paper, we propose a new scheme for off-line recognition of totally unconstrained handwritten numerals using a simple multilayer cluster neural network trained with the back propagation algorithm and show that the use of genetic algorithms avoids the problem of finding local minima in training the multilayer cluster neural network with gradient descent technique, and improves the recognition rates. In the proposed scheme, Kirsch masks are adopted for extracting feature vectors and a three-layer cluster neural network with five independent subnetworks is developed for classifying similar numerals efficiently. In order to verify the performance of the proposed multilayer cluster neural network, experiments with handwritten numeral database of Concordia University of Canada, that of Electro-Technical Laboratory of Japan, and that of Electronics and Telecommunications Research Institute of Korea were performed. For the case of determining the initial weights using a genetic algorithm, 97.10%, 99.12%, and 99.40% correct recognition rates were obtained, respectively.

Index Terms—Totally unconstrained handwritten numeral recognition, multilayer cluster neural network, genetic algorithm.

————————————————— ✦ —————————————————

1 INTRODUCTION

HANDWRITTEN character recognition systems have been proposed and implemented in a number of different ways. However, for the recognition of totally unconstrained handwritten numerals, it is probably safe to say that no simple scheme is likely to achieve high recognition and reliability rates. Thus recent efforts have been directed toward more sophisticated systems [1], [2]. In order to maximize the performance of unconstrained handwritten numeral recognition, the following two approaches could be considered; one is to design a feature extractor which does not miss important features while minimizing the number of meaningless pixels, and the other is to design a classifier which has good generalization power and minimum substitution error.

Our work has been motivated by these two approaches. The proposed scheme consists of two stages: a feature extraction stage for extracting four-directional local feature vectors with Kirsch masks and one global feature vector linearly compressed from a normalized input image, and a classification stage for recognizing numerals with a simple multilayer cluster neural network trained with the back propagation algorithm. In the case of using a neural network trained with the back propagation algorithm [3] as a classifier, initial weights of the neural network may cause the problem of finding local minima in training with the gradient descent technique because gradient descent techniques proceed by locally searching the immediate neighborhood of a current solution. However, the genetic algorithm has excellent global sampling abilities because it ensures broad coverage over the entire search domain. This suggests that using the genetic algorithm to provide

---

- S.-W. Lee is with the Department of Computer Science and Engineering, Korea University, 1, 5-ka, Anam-dong, Seongbuk-ku, Seoul 136-701, Korea.
  E-mail: swlee@human.korea.ac.kr.

good "seeds" from which the back propagation algorithm then continues to search will be effective [4]. Therefore, we combined a genetic algorithm with the back propagation algorithm to avoid the problem of finding local minima in training with the gradient descent technique.

In order to verify the performance of the proposed multilayer cluster neural network, experiments with handwritten numeral database of Concordia University of Canada, that of Electro-Technical Laboratory(ETL) of Japan, and that of Electronics and Telecommunications Research Institute(ETRI) of Korea were performed. For the case of determining the initial weights using a genetic algorithm, 97.10%, 99.12%, and 99.40% correct recognition rates were obtained, respectively.

2 REVIEW OF EARLIER WORK

In this section we review some methods for the recognition of totally unconstrained handwritten numerals and a multiple-expert system which incorporates different unconstrained handwritten numeral recognition methods. Then, we also review some representative recognition systems which have been implemented to classify unconstrained handwritten numerals with neural networks.

In the method of Mai and Suen [6], statistical data on the frequency of occurrence of features during training are stored in a database. The database is used to deduce the identification of an unknown sample.

To achieve even higher reliability by taking advantage of the inherent complementarity of the different unconstrained handwritten numeral recognition methods, Suen et al. [8] incorporated these methods into a multiple-expert system.

Very good results were recently reported using neural networks. In the method of Krzyzak et al. [9], features are first extracted from the contours of numerals: 15 complex Fourier descriptors from the outer contours and simple topological features from the inner contours. These features are then presented as input to a three-layer modified back propagation network. The modification aims at avoiding the problem of finding local minima in the gradient descent technique used to adjust the weights and results in a convergence rate which is twice as fast.

TABLE 1
COMPILATION OF SOME OF THE BEST RESULTS FOUND IN LITERATURE

Methods	Recognized	Substituted	Rejected	Reliability	Training	Testing	PPI
[5]*	93.10	2.95	3.95	96.98	4000	2000	166
[6]*	92.95	2.15	4.90	97.74	4000	2000	166
[7]*	93.90	1.60	4.50	98.32	4000	2000	166
[8]*	93.05	0.00	6.95	100.00	4000	2000	166
[9]*	86.40	1.00	12.60	98.85	4000	2000	166
[9]*	94.85	5.15	0.00	94.85	4000	2000	166
[10]	90.00	1.00	9.00	98.90	7291	2007	300
[10]	92.00	2.00	6.00	97.80	7291	2007	300
[11]	90.30	1.00	8.70	98.90	7200	1800	
[12]	87.85	4.90	7.25	94.72	5000	3540	166
[13]	90.87	2.92	6.21	96.89	15000	10000	
[14]	95.54	1.99	2.47	97.96		2711	300
[14]	97.10	0.96	1.94	99.02		1762	300
[15]	99.50	0.50	0.00	99.50	5000	5000	
[16]	96.35	1.00	2.65	98.97		6000	166
[16]	98.20	0.77	1.03	99.23		2219	300
[17]	93.30	2.50	4.20	97.39	1820	7100	300
[18]	97.97	2.03	0.0	97.97	8783	7394	
[19]	87.95	1.04	11.01	98.83		2103	
[20]	92.60	4.60	2.80	95.30	19377	19377	

Le Cun et al. [10] achieved excellent results with a back propagation network using size-normalized images as direct input. Their solution consists of a network architecture which is highly constrained and specifically designed for the task. There are four internal layers, two layers made of independent groups of feature extractors and two layers which perform averaging/sub-sampling. The last internal layer is fully connected to the ten-element output, but all other connections are local and use shared weights. In total, there are 4,635 units and 98,442 connections but only 2,578 independent parameters.

Recently, Knerr et al. [11] reported that a neural network classifier with single-layer training could be applied efficiently to the recognition of handwritten numerals. In this method, the STEPNET procedure decomposes the problem into simpler sub-problems which can be solved by linear separators.

Table 1 shows the performance of some of the most reliable handwritten numeral recognition systems found in the literature. In the Table 1, '*' denotes the case in which the handwritten numeral database of Concordia University of Canada has been used. The fifth column of the table is the reliability of the algorithm, which is computed as shown in the following equation:

$$Reliability = \frac{Correct\ Recognition\ Rate}{Correct\ Recognition\ Rate + Substitution\ Error\ Rate}$$

(1)

For further information on earlier work, refer to [1] and [21].

3 FEATURE EXTRACTION

In the case of handwritten numerals, it seems natural to use some a priori knowledge about the recognition task in order to transform the low level information of the pixel images into a data representation of higher level. Two possibilities arise: either a priori knowledge is used to design preprocessing operations which are carried out explicitly before classification, or it is used to constrain the general architecture of the classifier, which results in a classifier specialized in the recognition task at hand. We have chosen the first possibility. Numerals, whether handwritten or typed, are essentially line drawings, i.e., one-dimensional structures in a two-dimensional space. Therefore, 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.

However, besides mere a priori knowledge, there are other factors which may influence the performance of a recognition system, such as the need for fast computation or hardware limitations. The first-order differential edge detectors are adequate for local detection of a line segment and fast computation. The Frei-Chen edge detector, Kirsch edge detector, Prewitt edge detector, Sobel edge detector, and so on are the representative edge detectors of first differential edge detectors. However, among these edge 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.

Kirsch defined a nonlinear edge enhancement algorithm as follows [22]:

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

(2)

where

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

(3)

and

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

(4)

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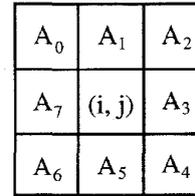

Fig. 1. Definition of eight neighbors of pixel (i, j).

In this paper, we calculate directional feature vectors for horizontal(H), vertical(V), right-diagonal(R), and left-diagonal(L) directions as follows:

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

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

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

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

(5)

As a final feature extracting step, each 16×16 feature map is compressed to 4×4 feature map by accumulating pixels of each 4×4 subregion. Fig. 2 shows the Kirsch masks used for calculating directional feature vectors.

(a)   (b)
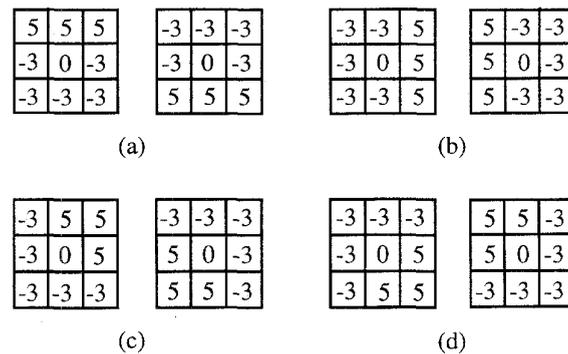
(c)   (d)

Fig. 2. Kirsch masks, (a) horizontal, (b) vertical, (c) right-diagonal, (d) left-diagonal.

In the case of using only these local features in totally unconstrained handwritten numeral recognition, global characteristics of input images may not be considered. Therefore, in this paper, we compressed the 16×16 normalized input image into 4×4 image and used this compressed image as a global feature. As a result, final feature vectors consist of $5 \times 4 \times 4$ feature vectors; $4 \times 4 \times 4$ local feature vectors and $1 \times 4 \times 4$ global feature vector. Fig. 3 shows the overview of the proposed feature extraction step.

4 MULTILAYER CLUSTER NEURAL NETWORK CLASSIFIER

A popular objective function in neural networks is the mean squared error(MSE) between the actual output of the network and a desired output. Minimizing the objective function is performed by a gradient descent procedure which requires the computation of the gradient of the objective function with respect to connection weights. Back propagation [3] is just an efficient way to compute this gradient. However, a naive back propagation network for identifying handwritten numerals is a large, fully-connected network, where all the units in a layer are connected to all the units in the following layer(s). This approach has severe deficiencies. If the number of weights is kept reasonably small, the network cannot even learn the training set accurately. On the other hand, if it is made big enough to learn the

training set, the excessive number of parameters causes overfitting [10]. In this paper, we propose a simple three-layer cluster neural network with five independent subnetworks.
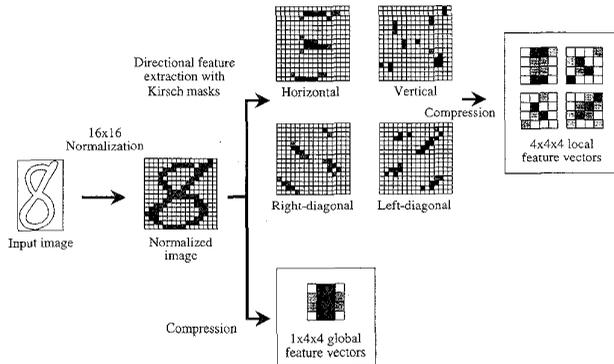

Fig. 3. Overview of feature extraction.

4.1 Network Architecture

The cluster neural network is that the units in a layer are clustered and each cluster is fully connected to a corresponding cluster in following layer, independently. The network architecture is presented in Fig. 4. The input layer consists of five 4×4 maps because the network inputs are $4 \times 4 \times 4$ local feature maps for four-directions and $1 \times 4 \times 4$ global feature map. Each feature map is fully connected to a corresponding cluster in the hidden layer, independently. However, the output layer is fully connected to all hidden units. The output is composed of 10 units: one per class. When a pattern belonging to class i is presented, the desired output is +1 for the ith output, and zero for the other output units. Therefore, the three-layer cluster neural network has 170 units and 2,080 independent parameters.
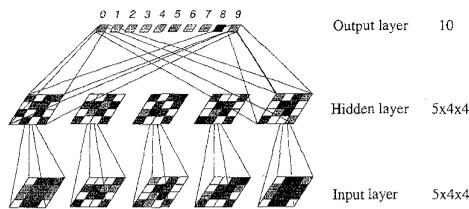

Fig. 4. Proposed simple cluster neural network architecture.

Besides the simplicity, one intuitive advantage of our three-layer cluster network is that while one subnetwork may be confused by a given input, others may not because each subnetwork has started from a different initial state and learned with different feature maps. One subnetwork, for example, may not be able to correctly classify specific sets of numerals because its weight pattern conflicts with those necessary to classify other numerals. This is to be expected since the totally unconstrained handwritten numerals are clearly not orthogonal. In other words, the sequence and types of strokes needed to form, for example, the digit '8' is very close to those required to form the digit '9.' Thus, it is more likely for a network to erroneously classify a '9' as an '8' than as a '5.'

4.2 Learning

Each subnetwork between the input layer and the hidden layer starts with random initial weights and learns with different feature maps, respectively. All of the connections in the network are adaptive, and are trained with back propagation algorithm. The transfer function at each unit is the familiar sigmoid function as follows:

$$O = \frac{1}{1 + e^{-\left(\sum wX + bias\right)}} \tag{6}$$

where, O is the output of nonlinear transfer function, w is weight, and X is input to the unit. w and $bias$ are randomly selected in start state and modified during training. The modification of weights is performed as follows:

$$\Delta w(n) = -\epsilon \partial E / \partial w + \alpha \Delta w(n - 1) \tag{7}$$

where, n is the training index, α is momentum, and E is MSE. After learning, each subnetwork will generally have different internal representations.

4.3 Determination of Optimized Initial Weights Using Genetic Algorithm

In order to determine the optimized initial weights of the multilayer cluster neural network, we combined a genetic algorithm with a back propagation algorithm. The genetic algorithm is an effective sampling technique because it ensures broad coverage over the entire search domain via stochastic search [4]. The genetic algorithm works by collecting information from the early and virtually uniform sampling of early generations, and then using this information to guide subsequent sampling towards particularly promising regions. The selection of a new element of the domain to evaluate therefore exploits global information from across the domain. Therefore, combining the genetic algorithm's global sampling with back propagation's local searching seems a extremely natural.

Genetic algorithms are iterative procedures which maintain a "population" of candidate solutions to the objective function $f(x)$:

$$P(t) = <x_1(t), x_2(t), ..., x_N(t)> \tag{8}$$

where, each x_i in population P is a string of length L. In this paper, each x_i represents a vector of initial weights of the multilayer cluster neural network. During each iteration step, called a "generation," the current population is evaluated, and, on the basis of that evaluation, a new population of candidate solutions is formed. A general sketch of the procedure appears in Fig. 5. The initial population $P(0)$ is usually chosen at random.

```
procedure GA+BP
begin
    t = 0;
    initialize P(0);
    learning with BP;
    evaluate P(0);
    while  termination condition not satisfied  do
    begin
        t = t+1;
        select P(t) from P(t-1);
        reproduce P(t);
        learning with BP;
        evaluate P(t);
    end
end
```

Fig. 5. Genetic algorithm combined with a back propagation algorithm.

5 EXPERIMENTAL RESULTS AND ANALYSIS

In this paper, we used three different databases; handwritten numeral database of Concordia University of Canada, that of ETL of Japan, and that of ETRI of Korea.
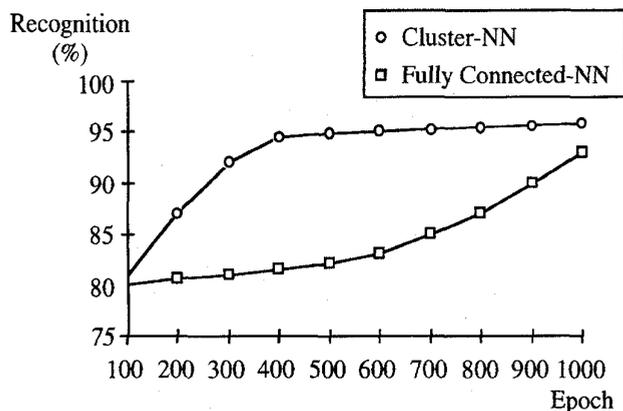
Fig. 6 Recognition rates per epochs using a fully connected multilayer neural network and a cluster neural network.

Four-thousand numerals were used for training and 2,000 numerals were used for testing from the numeral database of Concordia University of Canada, 6,000 numerals were used for training and 4,000 numerals were used for testing from the ETL-1 databases, and 4,000 numerals were used for training and 2,000 numerals were used for testing from the numeral database of Electronics and Telecommunications Research Institute, respectively.

The genetic algorithm has been used to create a population of initial weight vectors for the multilayer cluster neural network and then the back propagation algorithm has been used to optimize each of these. The MSE of each result was used for evaluating the initial weight vectors. Basic parameters of the genetic algorithm are presented in Table 2.

TABLE 2
GENETIC ALGORITHM PARAMETERS

Total trials	100
Population size	20
Mutation rate(%)	0.007
Crossover rate(%)	0.0007
Generation gap	1.000
Epochs for one generation	10

TABLE 3
RESULTS ON THE TESTING SET

Method \ Database	Concordia	ETL-1	ETRI
Krzyzak et al.[9]	94.85%	–	–
BP	96.69%	98.51%	99.20%
BP+GA	97.10%	99.12%	99.40%

Fig. 7 shows the examples from the handwritten numeral database which were misclassified. Table 4 shows the confusion matrix for the method BP+GA using the handwritten numeral database of Concordia University. Fig. 8 shows atypical data [10]. The proposed multilayer cluster neural network classifies these correctly, even though they are quite unlike anything in the training set.

The proposed scheme has been also implemented on wavefront array processor architecture. For the detailed description of the parallel hardware implementation, refer to [23].

Fig. 7. Examples of misclassified numerals.

TABLE 4
CONFUSION MATRIX FOR THE METHOD BP+GA

Class	0	1	2	3	4	5	6	7	8	9	Rejected	Substituted	Recognized
0	195					1	1		3		0%	2.5%	97.5%
1		196	2			1		1			0%	2%	98%
2			195	2				1	2		0%	2.5%	97.5%
3			3	193					2	2	0%	3.5%	96.5%
4		2	1		194		2			1	0%	3%	97%
5	1	1		3		195					0%	2.5%	97.5%
6	2	1				1	196				0%	2%	98%
7			1	1	2		1	193	1	1	0%	3.5%	96.5%
8	1	3			2				193	1	0%	3.5%	96.5%
9	1		1	1			1	2	2	192	0%	4%	96%
									Average		0%	2.9%	97.1%

Fig. 8. Atypical data classified correctly by the proposed cluster neural network.

6 CONCLUDING REMARKS

In this paper, we proposed a new scheme for off-line recognition of totally unconstrained handwritten numerals using a simple multilayer cluster neural network trained with the back propagation algorithm.

The proposed multilayer cluster neural network has five independent subnetworks between the input layer and the hidden layer. Each subnetwork starts with the optimized initial weights determined by using genetic algorithm combined with the back propagation algorithm and learns with different feature maps using the back propagation algorithm, respectively. After learning,

each subnetwork has different internal representations. Consequently, the proposed multilayer cluster neural network could classify similar numerals efficiently.

In this paper, we used a simple multilayer cluster neural network which has 10 output units: one per class. However, considering multiple models for the class which has wide variations, it is expected that the performance of proposed scheme will be improved.

Further investigation should be made, however, to design a locally constrained cluster network architecture which has good generalization and involves multiple models and to develop a technique in which segmentation and recognition are integrated for the recognition of unconstrained handwritten, connected numerals.

ACKNOWLEDGMENT

REFERENCES

[1] C.Y. Suen, C. Nadal, R. Legault, T.A. Mai, and L. Lam, "Computer Recognition of Unconstrained Handwritten Numerals," *Proc. IEEE*, vol. 80, no. 7, pp. 1,162-1,180, July 1992.

[2] J. Rocha and T. Pavlidis, "A Shape Analysis Model with Applications to a Character Recognition System," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 16, no. 4, pp. 393-404, Apr. 1994.

[3] D.E. Rumelhart, G.E. Hinton, and R.J. Williams, "Learning Internal Representations by Error Propagation," *Parallel Distributed Processing*, vol. 1. Cambridge, Mass.: MIT Press, 1986.

[4] R.K. Belew, J. McInerney, and N.N. Schraudolph, "Evolving Networks: Using the Genetic Algorithm with Connectionist Learning," *Artificial Life II*, C.G. Laugton, C. Taylor, J.D. Farmer, and S. Rasmussen, eds. Addison-Wesley, 1991, pp. 511-547.

[5] L. Lam and C.Y. Suen, "Structural Classification and Relaxation Matching of Totally Unconstrained Handwritten ZIP-Code Numbers," *Pattern Recognition*, vol. 21, no. 1, pp. 19-31, 1988.

[6] T. Mai and C.Y. Suen, "A Generalized Knowledge-Based System for the Recognition of Unconstrained Handwritten Numerals," *IEEE Trans. Systems, Man and Cybernetics*, vol. 20, no. 4, pp. 835-848, 1990.

[7] R. Legault and C.Y. Suen, "Contour Tracing and Parametric Approximations for Digitized Patterns," *Computer Vision and Shape Recognition*, A. Krzyzak, T. Kasvand, and C. Y. Suen, (eds.). Singapore: World Scientific Publishing, 1989, pp. 225-240.

[8] C.Y. Suen, C. Nadal, T.A. Mai, R. Legault, and L. Lam, "Recognition of Handwritten Numerals Based on the Concept of Multiple Experts," *Proc. of First Intl. Workshop on Frontiers in Handwriting Recognition*, Montreal, Canada, pp. 131-144, Apr. 1990.

[9] A. Krzyzak, W. Dai, and C.Y. Suen, "Unconstrained Handwritten Character Classification Using Modified Backpropagation Model," *Proc. of First Int. Workshop Frontiers in Handwriting Recognition*, Montreal, Canada, pp. 155-166, Apr. 1990.

[10] Y. Le Cun, et al., "Constrained Neural Network for Unconstrained Handwritten Digit Recognition," *Proc. of First Int. Workshop on Frontiers in Handwriting Recognition*, Montreal, Canada, pp. 145-154, Apr. 1990.

[11] S. Knerr, L. Personnaz, and G. Dreyfus, "Handwritten Digit Recognition by Neural Networks with Single-Layer Training," *IEEE Trans. Neural Networks*, vol. 3, no. 6, pp. 962-968, Nov. 1992.

[12] P. Ahmed and C.Y. Suen, "Computer Recognition of Totally Unconstrained Handwritten ZIP Codes," *Int'l. J. Pattern Recognition and Artificial Intelligence*, vol. 1, no. 1, pp. 1-15, 1987.

[13] M. Beun, "A Flexible Method for Automatic Reading of Handwritten Numerals," *Philips Technical Review*, vol. 33, pp. 89-101 and pp. 130-137, 1973.

[14] E. Cohen, J.J. Hull, and S.N. Srihari, "Understanding Handwritten Text in A Structured Environment: Determining ZIP Codes from Addresses," *Int'l. J. Pattern Recognition and Artificial Intelligence*, vol. 5, nos. 1 and 2, pp. 221-264, 1991.

[15] B. Duerr, W. Haettich, H. Tropf, and G. Winkler, "A Combination of Statistical and Syntactical Pattern Recognition Applied to Classification of Unconstrained Handwritten Numerals," *Pattern Recognition*, vol. 12, pp. 189-199, 1980.

[16] P.D. Gader, D. Hepp, B. Forester, T. Peurach, and B.T. Mitchell, "Pipelined Systems for Recognition of Handwritten Digit in USPS ZIP Codes," *Proc. U.S. Postal Service Advanced Technology Conf.*, pp. 539-548, Nov. 1990.

[17] C.L. Kuan and S.N. Srihari, "A Stroke-Based Approach to Handwritten Numeral Recognition," *Proc. U.S. Postal Service Advanced Technology Conf.*, pp. 1033-1041, 1988.

[18] B. Lemarie, "Practical Implementation of a Radial Basis Function Network for Handwritten Digit Recognition," *Proc. Second Int. Conf. on Document Analysis and Recognition*, Tsukuba, Japan, pp. 412-415, Oct. 1993.

[19] B.T. Mitchell and A.M. Gillies, "A Model-Based Computer Vision System for Recognizing Handwritten ZIP Codes," *Machine Vision and Applications*, vol. 2, pp. 231-243, 1989.

[20] L. Stringa, "A New Set of Constraint-Free Character Recognition Grammars," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 12, pp. 1210-1217, Dec. 1990.

[21] D.-S. Lee and S.N. Srihari, "Handprinted Digit Recognition: A Comparison of Algorithms," *Proc. Third Intl. Workshop Frontiers in Handwriting Recognition*, Buffalo, USA, pp. 153-162, May 1993.

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

[23] Y.-J. Kim, S.-W. Lee, and M.-W. Kim, "Parallel Hardware Implementation of Handwritten Character Recognition System on Wavefront Array Processor Architecture," *Proc. Third Intl. Conf. Document Analysis and Recognition*, Montreal, Canada, pp. 715-718, Aug. 1995.