An Adaptive Modular Neural Network With Application to Unconstrained Character Recognition

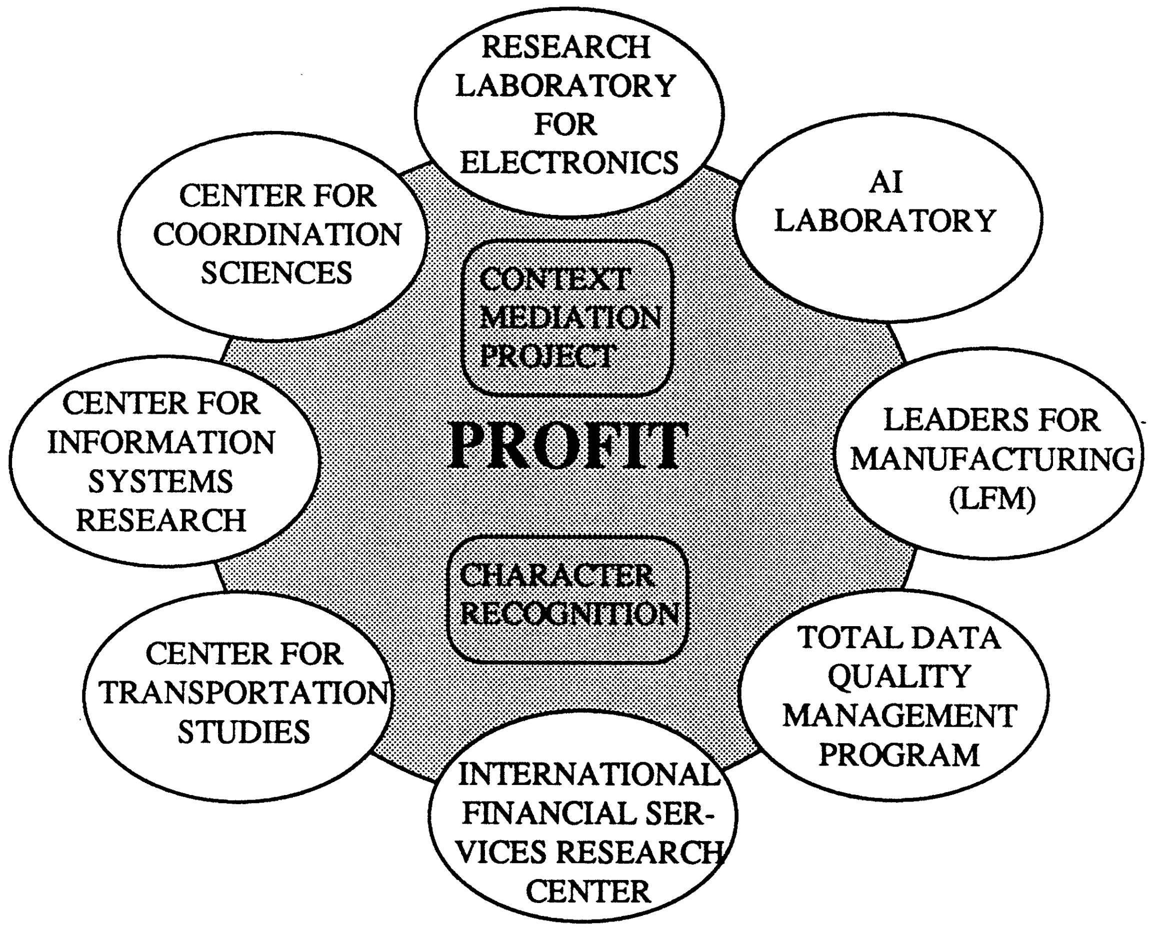
Lik Mui
Arun Agarwal
Amar Gupta
P.S.P. Wang

WP #3771 August 1993
PROFIT# 93-10

Productivity From Information Technology
"PROFIT" Research Initiative
Sloan School of Management
Massachusetts Institute of Technology
Cambridge, MA 02139 USA
(617)253-8584
Fax: (617)258-7579

Productivity From Information Technology (PROFIT)

The Productivity From Information Technology (PROFIT) Initiative was established on October 23, 1992 by MIT President Charles Vest and Provost Mark Wrighton "to study the use of information technology in both the private and public sectors and to enhance productivity in areas ranging from finance to transportation, and from manufacturing to telecommunications." At the time of its inception, PROFIT took over the Composite Information Systems Laboratory and Handwritten Character Recognition Laboratory. These two laboratories are now involved in research related to context mediation and imaging respectively.



In addition, PROFIT has undertaken joint efforts with a number of research centers, laboratories, and programs at MIT, and the results of these efforts are documented in Discussion Papers published by PROFIT and/or the collaborating MIT entity.

Correspondence can be addressed to:

The "PROFIT" Initiative
Room E53-310, MIT
50 Memorial Drive
Cambridge, MA 02142-1247
Tel: (617) 253-8584
Fax: (617) 258-7579
E-Mail: profit@mit.edu

AN ADAPTIVE MODULAR NEURAL NETWORK WITH APPLICATION TO UNCONSTRAINED CHARACTER RECOGNITION

Lik Mui, Arun Agarwal, Amar Gupta and P. S. P. Wang

Sloan School of Management

August, 1993

Management Overview

Paper documents continue to be the most common medium for information transmission in today's society. The task of automated document understanding is a challenging one and requires the development and integration of techniques from diverse areas, including vision, artificial intelligence, knowledge representation. language understanding and control theory. Automation of these processes in the context of handwritten documents involves creation of methods that can recognize characters with high accuracy and low substitution frequency. This paper describes research on unconstrained handwritten numeral recognition. A new adaptive modular network, which offers high noise tolerance, short adaption time, and high classifying capacity is described in this paper.

An Adaptive Modular Neural Network with Application to Unconstrained Character Recognition

Lik Mui
Department of EE and CS
MIT,
Cambridge, MA 02139

Arun Agarwal, Amar Gupta
Sloan School of Management
MIT,
Cambridge, MA 02139

Patrick Shen-Pei Wang
College of Computer Science
Northeastern University
Boston, MA 02115

Abstract

The topology and the capacity of a traditional multilayer neural system, as measured by the number of connections in the network, has suprisingly little impact on its generalization ability. This paper presents a new adaptive modular network that offers superior generalization capability. The new network provides significant fault tolerance, quick adaption to novel inputs, and high recognition accuracy. We utilize this paradigm for recognition of unconstrained handwritten characters.

1 Introduction

Paper documents continue to be the most common medium for information transmission in today's society. The task of automated document understanding is a challenging one and requires the development and integration of techniques from diverse areas, including vision, artificial intelligence, knowledge representation, language understanding and control theory. Automated document processing opens up new avenues for check verification, postal zip code reading, office automation, and a large number of other applications [1, 2, 3].

Most of the recent work on character recognition has focused on statistical, syntactic and structural approaches [4]. Following the growing popularity of the Backpropagation (BP) training method for multi-layered neural networks developed by the PDP group in UCSD [5], this new paradigm for designing handwriting recognition systems has grown to dominate the field. Two of the essential characteristics of a multi-layer neural networks are as follows:

1. **Learning :** Computer-based neural networks can be designed to modify their behavior in response to their environment. When provided a set of input-output vectors, some of these networks can self-adjust to produce consistent responses. In neural networks, learning is accomplished in one of two ways: (i) by altering structure of interconnections between the nodes; or (ii) by changing the signal strengths of these interconnections.

2. **Generalization :** Neural networks can be designed to generalize automatically as a result of their structure and not by using human in-

telligence embedded in the form of adhoc computer programs. It is this characteristic which makes the response of the system insensitive to minor variations in the input; such a system can thereby deal with the imperfect world in which we live.

Input patterns to a network are rarely evenly distributed across the feature space. In order to improve the generalization ability of a neural network classifier, researchers have applied various methodologies. One approach involves the use of networks that adopt local learning algorithms, such as k-nearest neighbor or radial basis function (RBF) [6]. This approach improves the robustness of a network if the right control parameters, such as the size of the neighborhood or the value of the regularization parameters, are used for local learning. These parameters are usually determined through a series of repetitive experiments. However, the local networks are limited either by their huge processing time requirement or by their restricted ability to generalize to a large set of pattern classes.

Another approach involves the reduction of the variability within a given input class by using digital signal processing (DSP) preprocessing routines such as smoothing, thinning, or image enhancement algorithms, to narrow the variability in the input space. Routines, such as those described in [7], can improve the generalization ability up to a certain point. But, if the number of input patterns classes is very large or if the patterns are spread very unevenly in the input space, the generalization ability breaks down. A third approach, exemplified by [8], involves changing either the size or the global architecture of the network, such as by altering the number of hidden layers, or by applying schemes of weight sharing or pruning. Although each of these approaches has been shown to be effective in specific environments, designing the optimal architecture requires substantial experimentation. Further, Martin and Pittman [9] have shown that the topology and the capacity of the multilayer neural system, as measured by the number of connections in the network, have surprisingly little impact on the generalization ability of the network.

Recent studies into modular network architecture have yielded positive results. In [10, 11, 12, 13], the investigators demonstrate that such a type of network can offer better generalization ability, classification accuracy, and superior tolerance to noisy inputs as compared to schemes based on single large network outlined in the previous paragraph.

Our research group focuses on the use of neural networks to read the amount field on bank checks [14, 15]. For this domain, alphanumeric bitmap strings (primarily numeric, along with special characters like dollar sign, period, and comma) need to be classified with high accuracy and low substitution frequency. As part of our research on unconstrained handwritten numeral recognition, we have developed a new adaptive modular network, which offers a high noise tolerance, short adaptation time, and high classifying capacity.

The next section of this paper describes the architecture of our character recognition system. The proposed network architecture and the training dynamics are discussed in Section 3. Comparative results are presented in Section 4. The potential advantages of the new network architecture are examined in the concluding section of the paper.

2 System Architecture for Unconstrained Character Recognition

Although the broad area of Optical Character Recognition (OCR) has witnessed intensive research over the past twenty years, automated reading of handwritten amounts on bank checks has generally been a difficult and a challenging area. This is because of the wide variations in handwriting styles, the absence of constraints on the total number of digits (unlike postal zip-code reading), and the need for extremely high accuracies with (almost) zero substitution errors. In order to deal automatically with bank checks, such as the one shown in Figure 1, we have developed the prototype whose overall architecture is shown in Figure 2, and is discussed in subsequent paragraphs.

Dynamic Thresholding: Completed checks in the U.S. are characterized by wide differences in background textures, writing devices, and colors and shades of the written material. In the dynamic thresholding stage, the histogram of the input image is used to remove noise and to enhance the clarity of characters.

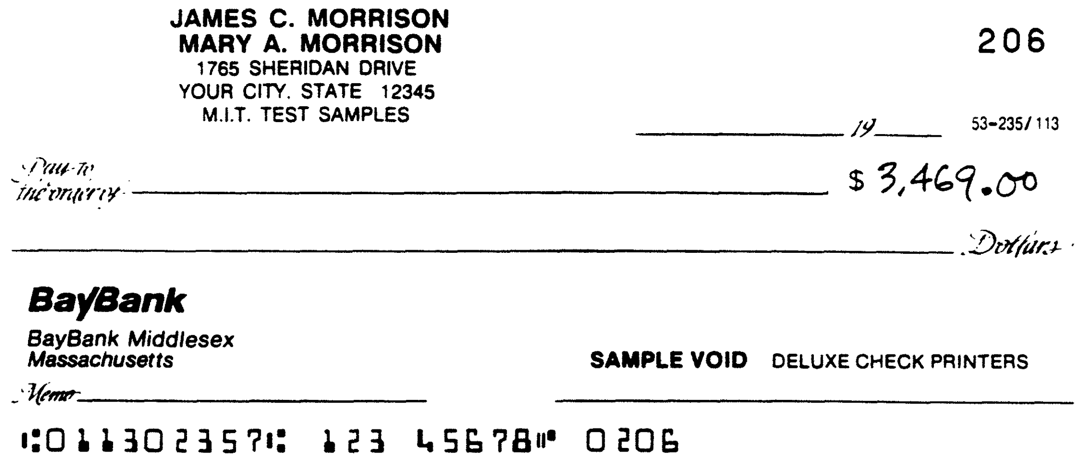


Figure 1: A Typical Bank Check

Segmentation Module: The bitmap of the courtesy amount is passed to the segmentation module which breaks the bitmap into distinct and meaningful pieces in three stages: extraction of connected components, splitting of upper and lower contours, and utilization of the hit and deflect strategy (HDS) for drawing segment cut lines. The output is a linked list of probable primitives in the form of connected components. This segmentation procedure is detailed fully in [16].

Syntax Checker and Processing: The problem of recognition of complex structures for denoting the cents portion on checks has been addressed by grammar checking function that enhances the accuracy of the system and also serves to verify whether the output of the system is reasonable.

Preprocessing: This stage reduces the variability in the slantness and thickness of the various characters. Detailed description can be found in [17] [18], [19] and [20].

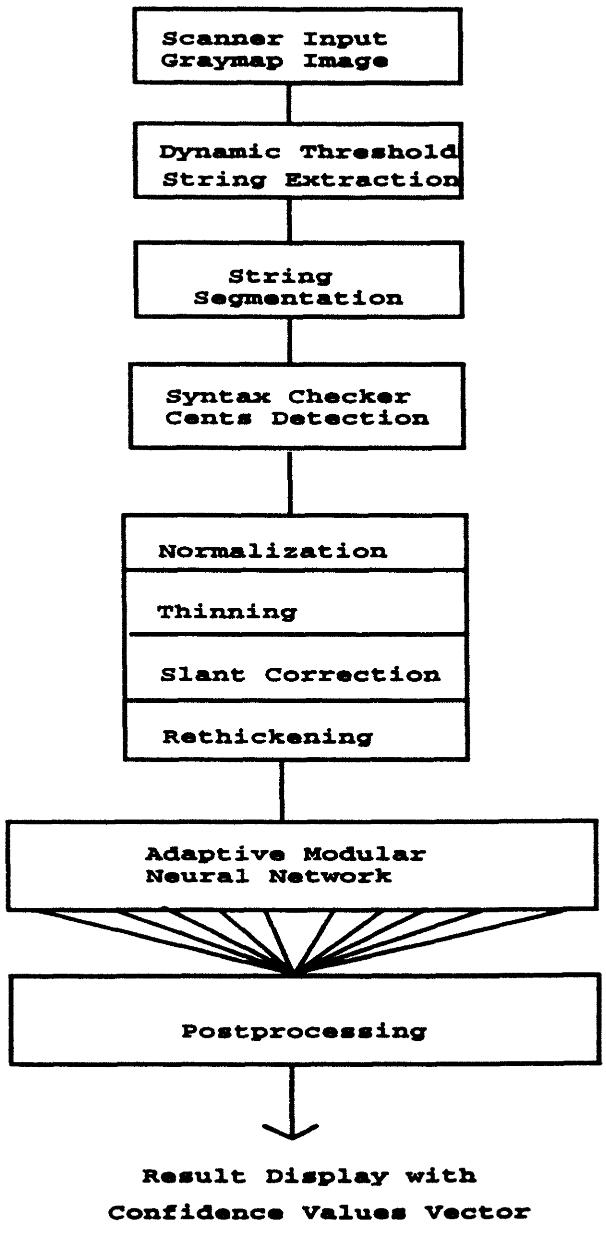


Figure 2: Block Diagram of a Check Processing System

Neural Classifier: This stage classifies the bitmap into one of the ten classes and is discussed in detail in Section 3.

Postprocessing: When the neural network detects a character that cannot be classified with adequate confidence, the particular digit is passed to a postprocessor, which utilizes traditional pattern recognition techniques to complement the connectionist approach. These techniques include structural analysis described in [21] and Freeman Chain Encoding Scheme detailed in [22].

3 Adaptive Modular Network

3.1 Network Architecture

The proposed network architecture is illustrated in Figure 3. The preprocessed numerals are arranged in 16×16 bitmap format and serve as input to the modular network. The network consists of three stages as described below:

1. **Locally Connected Layer:** The input bitmap is connected locally to a hidden layer of 81 hidden nodes. The connection scheme between the input and the first hidden layer of this net is local with a window size of 4×4 and with a moving increment of 2 pixels. Local connections are used to absorb some shift invariance in the input pattern.

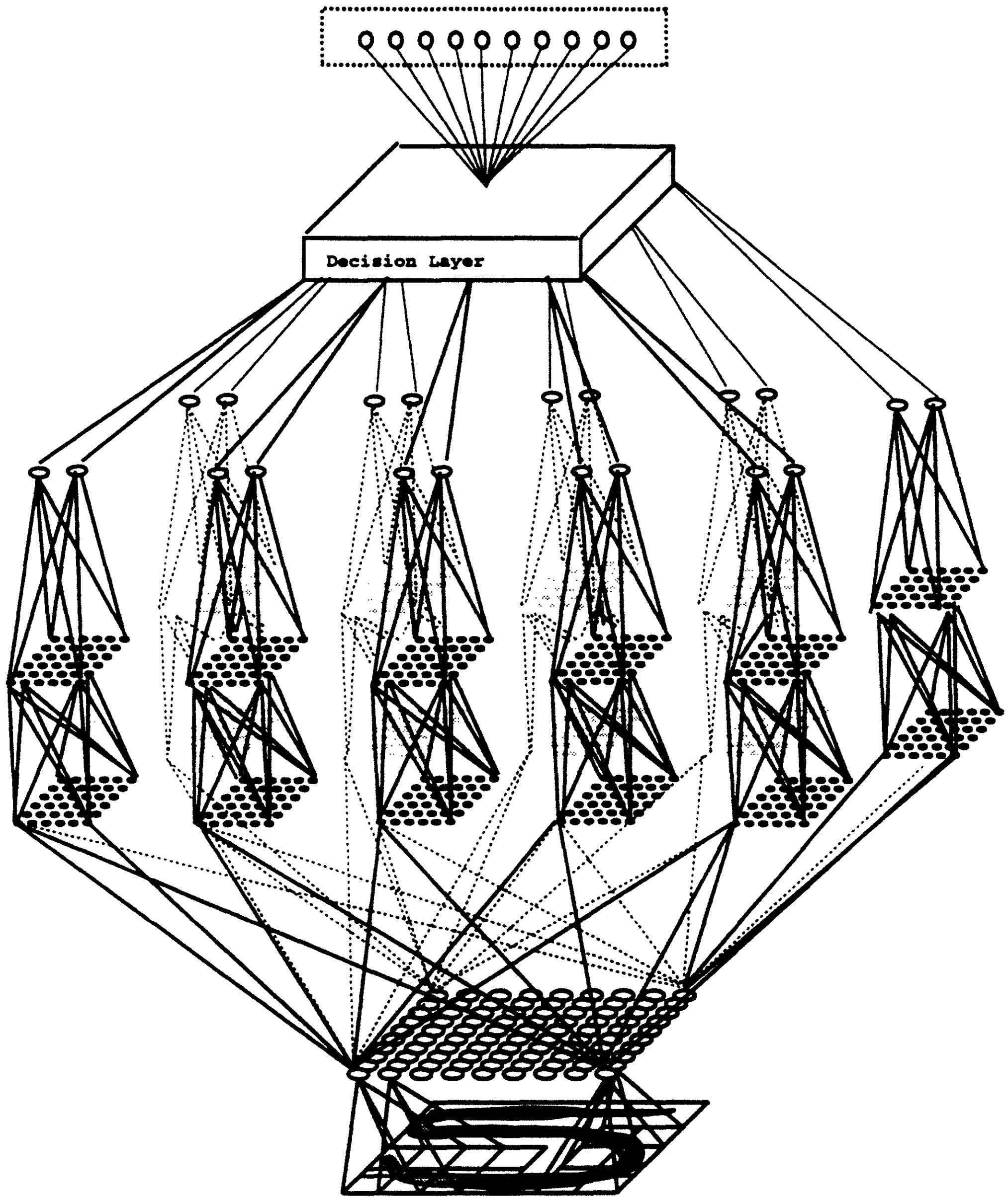


Figure 3: Proposed Network Architecture

2. **Subnets:** For recognizing numerals, there are ten small independent subnets, each of which is responsible for a particular digit. Each of the subnet has 2 hidden layers and 1 output layer.

3. **Decision Layer:** Here, decisions are made about the correct output for the entire network on a winner takes all basis.

The output of the locally connected layer is connected fully to the first hidden layer of the subnet which consists of 40 nodes. Input values are summed as followed :

$$net_i = \sum_{j=1}^{81} w_{ij} o_j$$

where w_{ij} is the weight values from the ith node in the upper layer to the jth in the lower layer and o_j is the output of node j of the locally connected layer. These values are mapped to activation values of the hidden layer using the standard sigmoidal function:

$$o_i = \frac{1}{(1 + e^{-net_i})}$$

Each node in the first hidden layer of the subnet is fully connected to the second hidden layer; each of these layers consists of 20 nodes. The full connection approach was preferred over local or shared weight connection scheme for the last two hidden layers because experiments with the latter approach did not affect the overall system accuracy by more than half a percent. The latter approach also doubles the overall system response time.

For the purpose of high volume transaction, we decided to use the simpler implementation.

The second hidden layer of the subnet is fully connected to the third (ie. the output) layer which consists only of two nodes. The first node is the positive + node, whose activation represents the recognition of the corresponding class of the subnet. The other node is the complement node $+'$, whose activation represents the recognition of a class that does not belong to the subnet. The ten different subnets yield a set of 20 output nodes which provide the output vector used for classification of the input bitmap.

Among the different options for implementing the decision layer, we experimented with two approaches. First, we used a feedforward Grossberg layer [23]. However, the results did not meet our expectation. The overall system accuracy is about 5 to 10 percent worse than that of the implemented system, which is described in the following paragraph. Possible reasons for such degraded performance is under current investigation.

Our second approach was based on experimental observation. We observed that in most cases, only one of the subnet's positive node was turned on; further, except for the this subnet's complement node, all other complement nodes were also turned on. Based on this observation, a winner takes-all scheme was implemented; this scheme involves checking all the positive nodes, and selecting the most activated one as the winner if two conditions are satisfied:

1. The selected node must possess an activation value higher than a designated threshold value (Δ_h); and
2. The next most activated node must possess an activation value less than the most activated node by (Δ_l).

If either of the two conditions fails, the decision is passed on to the complement nodes, and the least activated one is deemed to be the winner. The idea of restricting each subnet to classify only two types of inputs, either within or outside the particular class that the subnet is responsible for, reduces the classifying complexity of each subnet. As each subnet is responsible only for identifying the corresponding pattern class in the pattern space, each subnet needs only to demarcate the region that is associated with its class. Figure 4 illustrates this principle. Instead of using a single large network to partition the pattern space into individual sections that correspond to individual pattern class, our network is only concerned with a single class at a time. From this point of view, a much less complex network is required to differentiate what is inside a particular class from what is not.

3.2 Training Stage with Improved Backpropagation

A supervised training algorithm based on the modified backpropagation developed by Rumelhart and Hinton [5] has been used for training the network. An example of an input bitmap from the training file is shown in Figure 5.

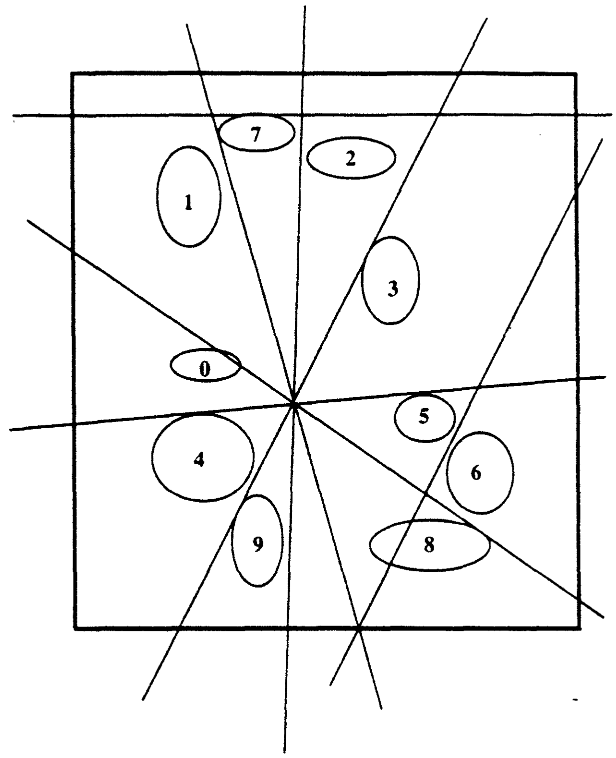
As shown in this figure, each input bitmap is tagged by its target value

Figure 4. Schematic Illustration of ideas discussed in paper concerning the generalizability of the proposed network

Right right side is an attempt to illustrate what traditional neural network attempts to accomplish in the learning stage, namely, dividing the pattern space into distinct regions belonging to each pattern class

The lower left figure tries to illustration the role of the positive node in classification. The positive node only distinguishes the class of pattern that belong to a given pattern class

The lower right figure depicts the role of the complement node in enhancing classification by demarcating the regions corresponding to the 'does-not belong' class.



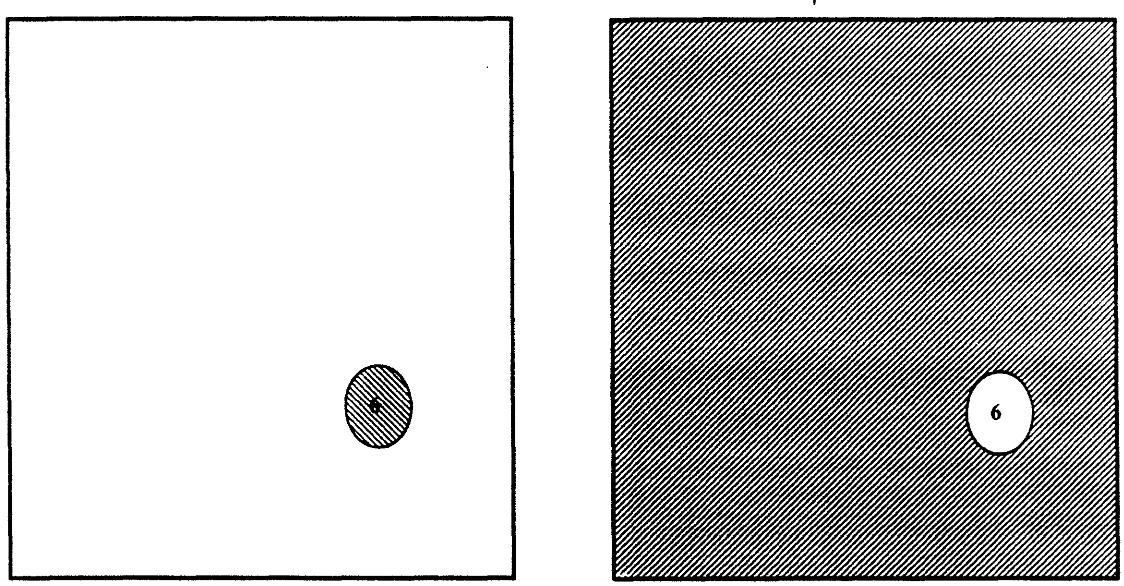


Figure 4:

```
0000000000000000
0011011111111100
0111111111111110
0110000000011100
0000000000011000
0000000000011000
0000000000011000
0000000000011000
0000000000011000
0000000000111000
0000000000110000
0000000000110000
0000000000110000
0000000000110000
0000000001110000
0000000000000000
```

Figure 5: A bitmap of a numeral

and read by the training program. Given t as the target value for subnet s, the output error for the output positive and complement nodes are computed as follows:

For positive node:

$$t \in s \quad E^s = 1 - OUT^s$$

$$t \notin s \quad E^s = -OUT^s$$

For complement node:

$$t \in s \quad E^s = -OUT^s$$

$$t \notin s \quad E^s = 1 - OUT^s$$

To compute the weight adjustments, consider the training process for a single weight from neuron p in the hidden layer j to neuron q in the output layer k. The output of a neuron in a layer k is subtracted from its target value to produce an error signal. This is multiplied by the derivative of the standard sigmoidal function:

$$\delta_{q,k} = OUT_{q,k}(1 - OUT_{q,k})(Target_q - OUT_{q,k})$$

The Widrow-Hoff or Delta rule [24] is applied to compute the incremental weight change:

$$\Delta w_{pq,k} = \eta_j \delta_{q,k} OUT_{p,j}$$

In the proposed training method, graded or fixed η_j are suggested as initial seed values. The η_j value for each weight layer can be adjusted independently to improve the time required for convergence of the network. The empirical formula used for updating η_j for epoch number $n+1$ and the layer j was as follows:

$$\eta_j[n+1] = \alpha_o \eta_j[n] \sqrt{\sum_q (Target_q - OUT_{q,k})^2} + \epsilon_o$$

where α_o is some empirical constant and $\epsilon_o > 0$ is a residual learning rate.

The dependence of the learning rate η_j on the mean square error (MSE) of the output nodes provides some control on the rate of learning. A high learning rate at the outset of the training stage reduces the probability for settling around a local minima. As the learning MSE approaches zero, a reduced learning rate can improve the time needed for convergence. The network trained with different initial η_j seed values is called multi variable-learning rate method and network trained with the same initial η_j seed values is called fixed variable-learning rate method.

Now the weight change is given by:

$$w_{pq,k}[n+1] = w_{pq,k}[n] + \Delta w_{pq,k} + momentum(\Delta w_{pq,k})$$

Since the hidden layers contain no target vector, the training process described above cannot be applied. The $\delta's$ calculated in the output layer are used to adjust the weights of the first hidden layer. In turn the $\delta's$ of this layer are used to adjust the weights of the subsequent hidden layers. The weights of the hidden layers are adjusted as follows:

$$\delta_{p,j} = OUT_{p,j}(1 - OUT_{p,j})(\sum_q \delta_{q,k} w_{pq,k})$$

After the error is calculated, the changes in the weights are calculated as in the case of the output layer.

4 Results

4.1 Training Speedup

For training the adaptive modular network, we utilized two different methods. One involved the variable-learning rate procedure discussed earlier in this paper, and the other involved using the generalized delta rule of McClelland

and Rumelhart [5]. When the MSE is near 0.05, the training process is interrupted repeatedly to test the network on a set of new digit pattern and to ensure that the network does not overtrain on the training set. Once the recognition rate on the new test set starts to decrease, the training cycle is stopped. In our case, the final MSE was about 0.02.

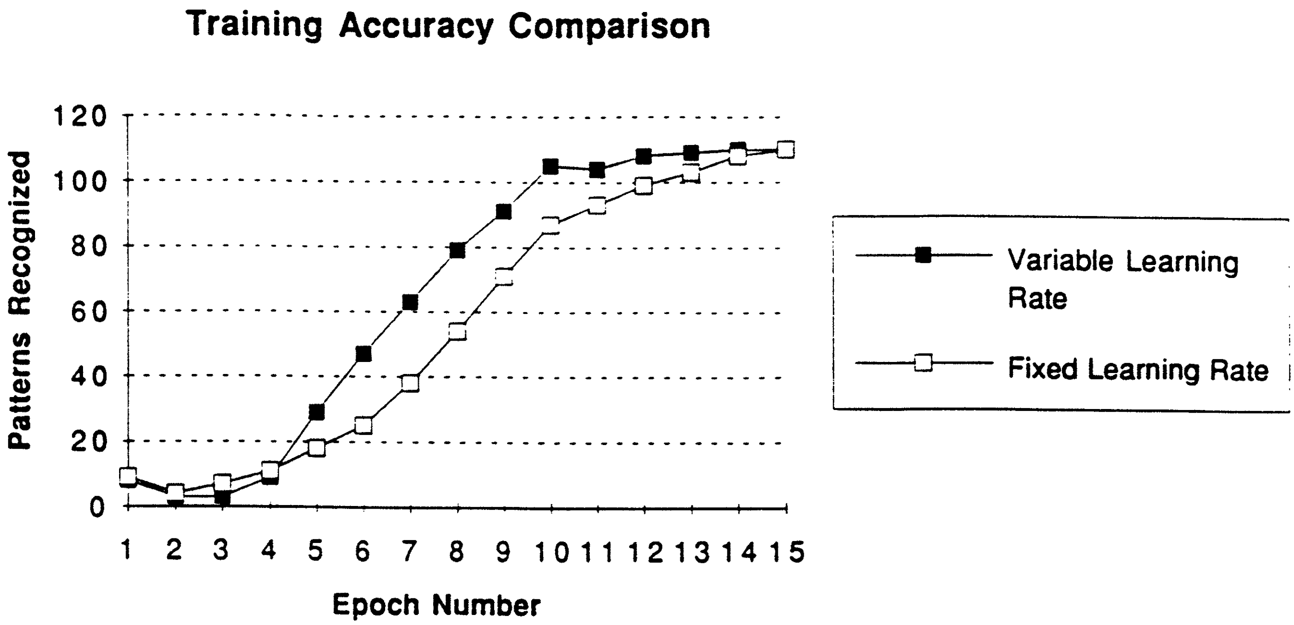
The results are summarized in Figure 6 and Figure 7, where $\alpha_o = 1.0$ and $\epsilon_o = 0.1$ are assumed. Figure 6a shows the recognition accuracy and Figure 6b shows Output Mean Square Error during the training cycles. In both the figures the learning rate η_j for the variable-learning procedure and for generalized delta rule was assumed to be 0.2.

Figure 7 assumes the learning rate for the variable-learning procedure to be $\eta = [\eta_1 = 0.5, \eta_2 = 0.3, \eta_3 = 0.1]$ and for generalized delta rule to be $\eta = 0.2$. For a multi variable-learning rate procedure there is 40 percent reduction in training time for achieving 90 percent accuracy and a similar speedup is observed for reducing MSE to 0.2.

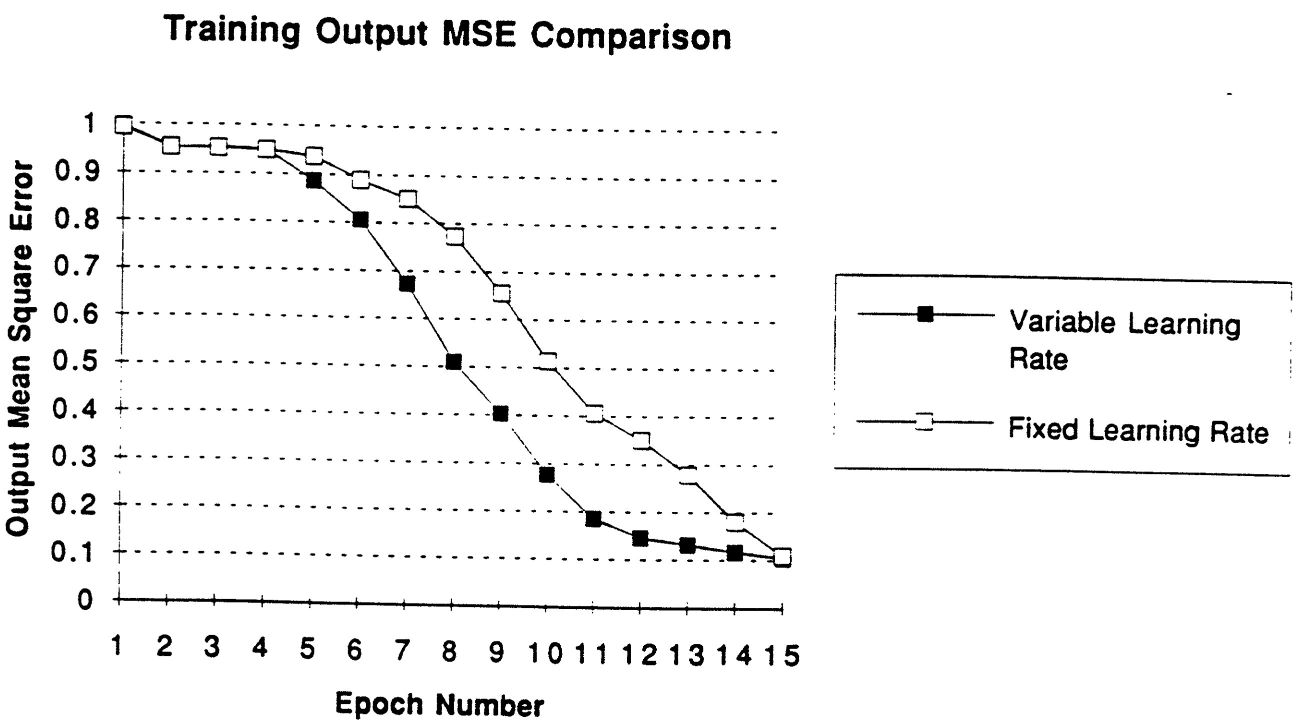
It is seen that the variable-learning rate method has quicker convergence as opposed to generalized delta rule method. Interestingly, the multi-variable-learning rate method offers superior convergence as compared to the fixed variable-learning rate method.

4.2 Recognition Performance

For a training set of 5993 digits, the training phase was terminated after a mean square error of 0.25 was reached, with an accuracy of over 99.5 percent

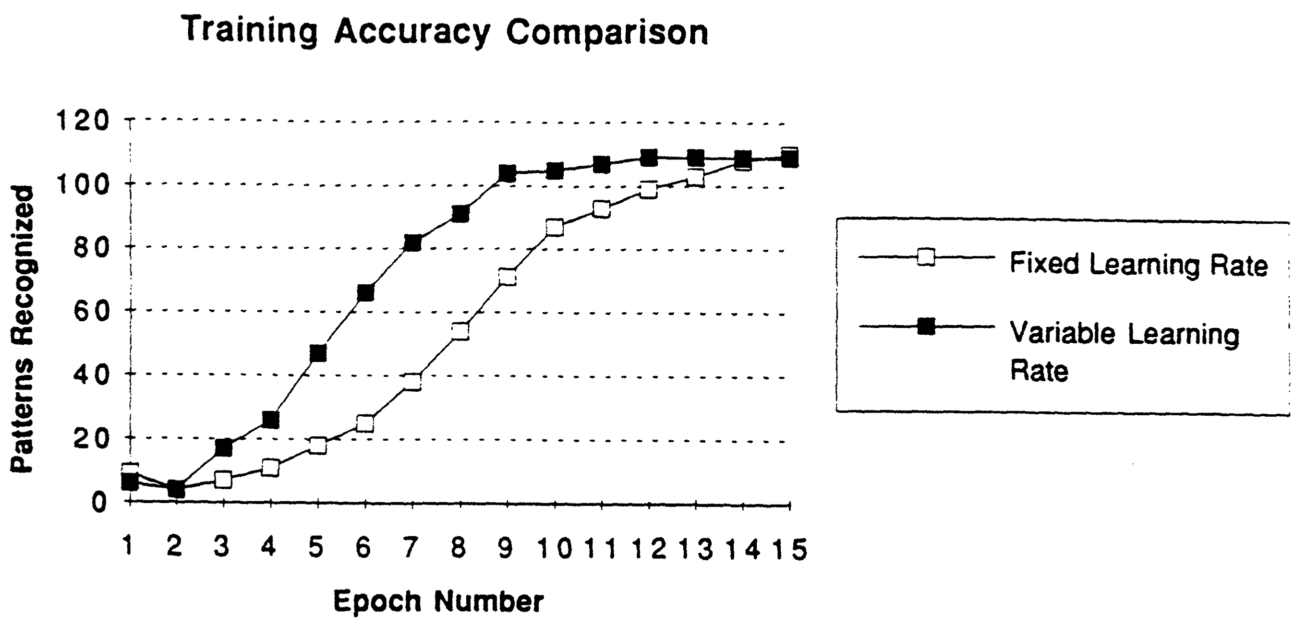


(a)

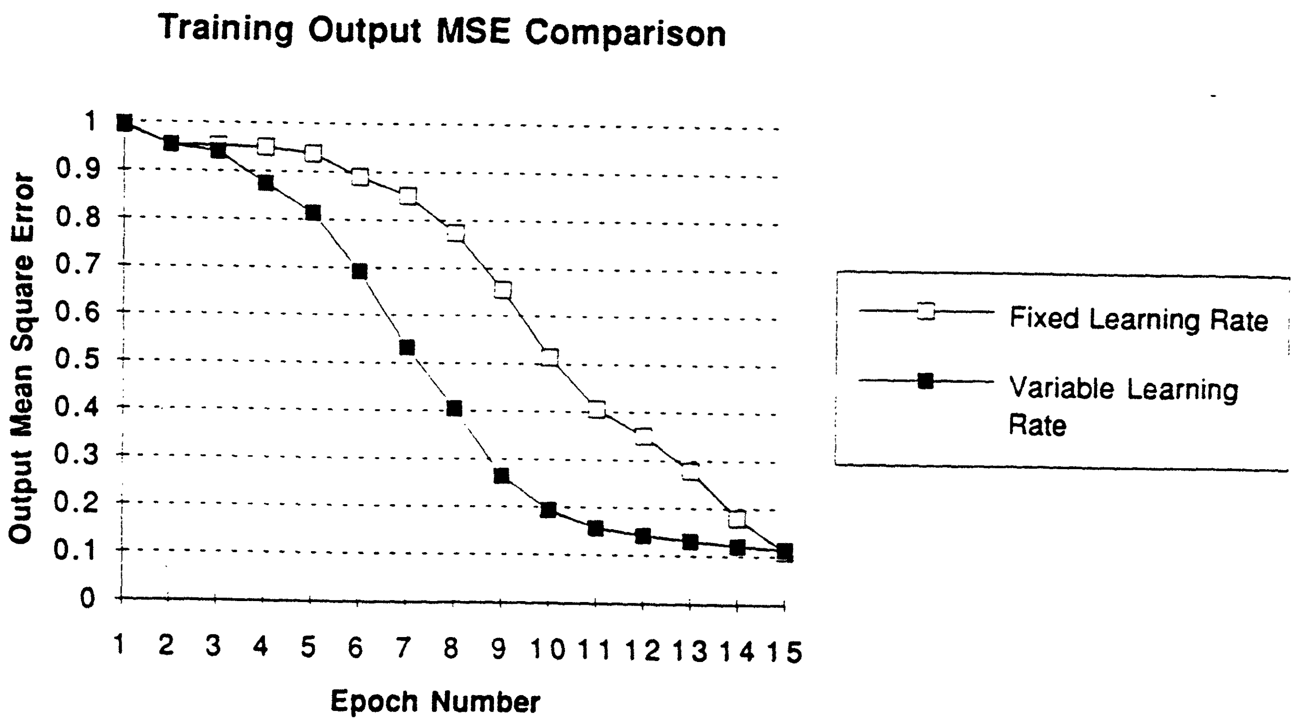


(b)

Figure 6: Comparison of training speedup



(a)



(b)

Figure 7: Comparison of training speedup

Classification Class	0	1	2	3	4	5	6	7	8	9
Number of Patterns	694	483	687	686	706	614	701	718	324	316
Number Rejected	17	11	46	41	30	53	52	29	40	3
Number Correct	674	454	623	641	675	544	656	685	258	300
Correct Percentage	98.4	94.5	94.1	96.9	97.3	95	97.1	97	88	95.5
Substitution	1.6	5.5	5.9	3.1	2.7	5	2.9	3	12	4.5

Table 1: Comparison of recognition performance of different classes

on the training set. The entire training process involved 87 epoches, with one epoch being defined as the set of all non-overlapping input patterns. The network was then tested using a new set of 5929 digits yielding accuracy and substitution rates for different digits as shown in Table 1. The overall accuracy was 96 percent. Interestingly, the digit '8' has significantly lower classification accuracy than the other classes. The reason could be that the number '8' shares some characteristic features of every other class.

4.3 Comparison with Other Networks

The network described in this study is similar in many ways to that proposed by Jacobs, and et al. (1991) in [12] on vowel discrimination task. In both of our studies, supervised learning is conducted by several subnets, each of which handles a subset of the input training classes. Both studies reveal high generalization ability of this approach. However, Jacobs, and et. al.'s work speeds up their network training by using a new error function, which they claim to have significant convergence improvement over the traditional error function. In our training speedup scheme, we adopt a simpler scheme of updating the learning rate based on the MSE of the output nodes that also enables significant reduction in training time. Further differences between our networks exist in terms of representation and integration of the output vector. Our approach of two complemented output nodes seems to work extremely well for the purpose of character recognition. In similar works on recognizing characters done by Hoffman and et al in [11] using a multi-networks, the system achieves up to 80% script character recognition accuracy. Using more traditional single multilayered perceptron approach to character recognition, other research groups (such as Le Cun and et.al. at AT&T Bell Labs [25] who use a modified single multi-layered perceptron network trained using backpropagation algorithm) have achieved 3.4% error with zero rejection on printed and handwritten zip code data, and 9% rejections for 1% substitution for handwritten zipcode data set. Due to the different applications, the training stages, and the testing sets of data, comparison between the performance of our proposed network and those of the others mentioned is difficult. However, based purely on the accuracy results, our network seems to out perform any of these networks.

The network that we have been using for our existing prototype is a three layered network trained using backpropagation. When we compared this network with the new one presented in this paper, a marked improvement in recognition accuracy was observed. Using the same system setup and dataset, the more traditional backpropagation neural network with a 256 input nodes, one hidden layer of 40 hidden nodes, and 10 output nodes was benchmarked against the new network using the same 5993 numerals that were used for testing the new network. A total of 120 epoches was required by this three layered network to provide an accuracy of 99.6% for the training set and an accuracy of 91.5% for the test set of 5929 numerals that were also used for testing the proposed network.

5 Discussion and Conclusion

This paper presented a new adaptive modular network that offers superior generalization capability. The new network provided significant fault tolerance, quick adaption to novel inputs, and high recognition accuracy. We demonstrated this paradigm on recognition of unconstrained handwritten characters and observed the following:

Generalization of the Network: The proposed network outperforms traditional large feedforward network in terms of overall recognition accuracy. Also the new architecture simplifies the classification task by means of its divide and conquer strategy. By taking the outputs of all the small subnets, high accuracy was obtained.

Speedup of the Network: The speedup in the training time achieved by the modification to the backpropagation algorithm is significant. The dependence of the learning rate on the mean square error (MSE) of the output nodes provides some control on the rate of learning. A high learning rate at the outset of the training stage reduces the probability for settling around a local minima. As the learning MSE approaches zero, a reduced learning rate improves the time needed for convergence. Varying the learning rate as a function of the output mean square error makes the network learning process converge much faster than otherwise.

Extensibility of the Network: Since the dynamics for each individual subnet is entirely independent of the presence of other subnets, the proposed network can be extended to classify a large set of distinct patterns without compromising the accurary of the overall system. The only drawback is that the training and recognition times may be proportionately lengthened for a larger set of pattern classes.

Due to the different applications, the training stages, and the testing sets of data, comparison between the performance of our proposed network and those of the others mentioned is difficult. However, based purely on the accuracy results, our network seems to out perform existing networks.

6 Acknowledgement

We thank our fellow researchers in the Character Recognition Group, especially Karim Hussein, for their technical assistance, and the sponsors of the MIT International Financial Services Research Center for the financial support.

References

[1] Gupta, A., S. Hazarika, M. Kallel and P. Srivastava, *Optical Image Scanners and Character Recognition Devices: A Survey and New Taxonomy*, MIT IFSRC Report # 107-89, 1989.

[2] Stern, R., *From Intelligent Character Recognition to Intelligent Document Processing*, Proc. Int. Electronic Imaging Exposition and Conference, Prentice-Hall, pp 236-245, 1987.

[3] Tappert, C. C., C. Y. Suen and T. Wakahara, *On-line Handwriting Recognition–A Survey*, Proc. Int. Conf. on Pattern Recognition, Rome, pp 1123-1132, 1988.

[4] Wang, P. S. P. (ed), *Character and Handwriting Recognition - Expanding Frontiers*, World Scientific Publishing, 1991.

[5] Rumelhart, D.E. G. E. Hinton and R. Williams, *Learning Internal Representations by Error Propagation*, Parallel Distributed Processing:

Explorations in the Microstructure of Cognition, Volume 1, Bradford Books, Cambridge, MA, pp 318-362, 1986.

[6] Bottou, L., and V. Vapnik, *Local Learning Algorithms*, Neural Computation, 4, pp 888-900, 1992.

[7] Burel, G., P. Isabelle and Jean-Yves Catros, *Recognition of Handwritten Digits by Image Processing and Neural Network*, IEEE/INNS International Joint Conference on Neural Networks, Volume 3, pp 666-671, 1992.

[8] Nowlan, S. J., and G. E. Hinton, *Simplying Neural Networks by Soft Weight-Sharing*, Neural computation 3, pp 473-493, 1992.

[9] Martin, G.L. and J.A. Pittman, *Recognizing Hand-Printed Letters and Digits Using Backpropagation Learning*, Neural Computation 3, pp 258-267, 1991.

[10] Bebis, G. N., and M. Georgiopoulos, *Increasing Classification Accuracy Using Multiple Neural Network Schemes*, SPIE Vol. 1709 Applications of Artificial Neural Networks III, pp 221-231, 1992.

[11] Hoffman, J., J. Skrzypek, and J. J. Vidal, *Cluster Network for Recognition of Handwritten Cursive Script Characters*, Network Networks, Vol 6(1), pp 69-78, 1993.

[12] Jacobs, R. A., M. I. Jordan, S.J. Nowlan and G.E. Hinton, *Adaptive Mixture of Local Experts*, Neural Computation 3(1), pp 79-88, 1991.

[13] Baxt, W. G., *Improving the Accuracy of an Artificial Neural Network Using Multiple Differently Trained Networks*, Journal of Neural Computation, 4, pp 772-780, 1992.

[14] Nagendraprasad,. M. V., A. Liu and A. Gupta, *A System for Automatic Recognition of Totally Unconstrained Handwritten Numerals*, IFSRC No. 218-92, Sloan School of Management, MIT, 1992.

[15] Wang, P.S.P., M.V. Nagendraprasad and A. Gupta, *A Neural Net Based "Hybrid" Approach to Handwritten Numeral Recognition*, From Pixels to Features III: Frontiers in Handwriting Recognition, (eds) S Impedovo and J. C. Simon, Elsevier Science Publishers B. V., pp 145-154, 1992.

[16] Sparks, P., *A Hybrid Method for Segmenting Numeric Character*, SB Thesis, MIT, 1990.

[17] Lam, L., S. W. Lee and C. Y. Suen, *Thinning Methodologies - A Comprehensive Survey*, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 14, No. 9, pp 869-885, 1992.

[18] Suen, C.Y. and P. S. P. Wang (eds), *Advances in Thinning Algorithms*, World Scientific Publishing, 1993 [to appear].

[19] Nagendraprasad, M. V., P. S. P. Wang and A. Gupta, *Algorithms for Thinning and Rethickening Digital Patterns*, Journal of Digital Signal Processing, Academic Press, Vol. 3, No. 2, pp 97-102, 1993.

[20] Nagendraprasad, M. V., P. S. P. Wang and A. Gupta, *An Improved Algorithm for Thinning Binary Digital Patterns*, Proc. of Eleventh ICPR, The Hague, The Netherlands, August 30-September 3, pp 286-389, 1992.

[21] Duggan, M., *Enhancing Accuracy of Automated Numerical Recognition*, SB in Electrical Engineering, MIT, 1992.

[22] Wang, P. S. P., and A. Gupta, *An Improved Structural Approach for Automated Recognition of Handprinted Characters*, International Journal of Pattern Recognition and Artificial Intelligence, Vol 5(1 & 2), pp 97-121, 1991.

[23] Grossberg, S., *Nonlinear Neural Networks: Principles, Mechanisms, and Architectures*, Neural Networks, Vol 1(1), pp 17-61, 1988.

[24] Widrow, B., and M. E. Hoff, *Adaptive Switching Circuits*, IRE WESCON Conv. Record, Part 4, pp 96-104, 1960.

[25] Le Cun, Y., B. Boser, J.S. Denker, D. Henderson, R. E. Howard, W. Hubbard and L. D. Jackel, *Handwritten Digit Recognition with a Back-Propagation Network*, Advances in Neural Information Processing Systems 2, pp 396-404, 1990.