Multiple Recognizers System Using Two-Stage Combination

Jonghyun Paik, Sung-bae Cho, Kwanyong Lee, Yillbyung Lee

Dept. of Computer Science, Yonsei University, Seoul, 120-749, KOREA
TEL : +82-2-361-2713 FAX : +82-2-365-2579
e-mail : { baegjh, sbcho, kylee, yblee }@csai.yonsei.ac.kr

Abstract

Most of the multiple recognizers system use a single combination method, therfore recognition performance depends on the characteristics of selected combination method. In order to solve this dependency problem and to increase recognition performance, we propose a new combination architecture of multiple recognizer that has two combination stages. The proposed system consists of three stages - the recognition stage including 5 recognizers, the first combination stage including 3 combinators which belong to different level, and final combination stage including a simple combinator. We verify the performance of proposed system using two standard handwritten digit database, CEDAR and CENPARMI, and recognition performance is better than other single combiantor systems.

1. Introduction

As diverse as the research in the field of recognition may be, especially because the concept of electronic document has become ubiquitous in recent years, the most essential issue in recognition would have to be in maximizing the efficiency of recognizers. There are two main stems to which researchers in this field adhere to; one is to build a recognizer by utilizing multiple features simultaneously[5], and the other is to simply use multiple recognizers[3]. Such methods of utilizing a number of either features or recognizers have already been proven to enhance recognition performance through worldwide competition of recognizers[4, 6].

Generally, the multiple recognizer system using the method which obtains final results by combining multiple recognizers, is attracting a good deal of interest due to the fact that performance can easily be improved without toiling with the recognizer itself. However, in the case of multiple recognizer system using single combination algorithms, the performance depends on the characteristic of combination method. Therefore, we propose new architecture which has two stage combination, in order to prohibit this kinds of problem and to increase the recognition performance of the system.

The proposed system consists of three stages, recognition, first combination and final combination. 5 recognizers which use different input features are included in recognition stage, and three combination methods which belong to different level are used in first stage combination. Simple combinators, like voting or bayesian combines the results obtained by first stage combinators in final combination stage. Using this system, we can decrease bias due to the characteristic of single combinator and increase the recognition performance.

2 Recognizers Description

All recognizers have different feature vectors and use same classifying algorithm, multilayer perceptron. By using this classifying method, we not only keep consistency of each recognizer's output values, but also earn high recognition performance through learning ability of neural network.

2.1 D-Recognizer

General mesh which has equal-sized sub-regions doesn't consider the density of black pixels in input image, therefore researches about calculating dynamic mesh which keep equal density of black pixel in horizontal and vertical regions is done. Using dynamic mesh, even if the input image has distortions or variations, consistent mesh can be obtained i,e it makes an effect of nonlinear normalization. In order to get dynamic mesh feature, input image is divided into 8x8 sub-regions. The number of black pixels in 64 sub-regions are inputted to 64-35-10 structured neural net classifier.

2.2 K-Recognizer

Kirsch mask is an effective tool for four directional edge detection. Therfore K-Recognizer used directional features

1015-4651/96 $5.00 © 1996 IEEE
Proceedings of ICPR '96

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

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

Horizontal Direction

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

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

Vertical Direction

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

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

Left Diagonal Direction

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

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

Right Diagonal Direction

Figure 1. 4 direction kirsch masks

which are extracted from input image by using kirsch mask, as an input vector. We extracted four directional features - horizontal, vertical, right and left diagonal directions - from 16x16 size normalized image by applying kirsch masks in Figure 1 and following formula.

$$G(i,j)_H = max(|5S_0 - 3T_0|, |5S_4 - 3T_4|)$$
$$G(i,j)_V = max(|5S_2 - 3T_2|, |5S_6 - 3T_6|)$$
$$G(i,j)_R = max(|5S_1 - 3T_1|, |5S_5 - 3T_5|)$$
$$G(i,j)_L = max(|5S_3 - 3T_3|, |5S_7 - 3T_7|)$$

After extracting four directional features, we compressed the each 16x16 feature map into 4x4 sized feature map and also compressed size normalized image into 4x4[2]. 80 features obtained by this method are input to the 80-41-10 structured neural network classifier.

2.3 G-Recognizer

It is well known from vector analysis that the gradient vector points in the direction of maximum rate of change at image (x, y)[1]. The gradient vector can represent two important quantities in edge detection, the one is magnitude and the other is direction of the vector. G-Recognizer use direction of gradient vector as an input feature. $\alpha(x, y)$ represnt the direction angle of the vector at (x, y). Then from vector analysis,

$\alpha(x,y) = tan^{-1}\frac{G_x}{G_y}$

where the angle is measured with respect to the X axis. G_x and G_y can be easily obtained by convolution of two sobel operator masks in Figure 2. After extracting direction of gradient vector, we quantized gradient directions into 12 units, and also divided input image into 4x4 equal regions. From this process, we can generate 192 input vectors and these feature vector input to the 192-77-10 structure neural network classifier.

2.4 H-Recognizer

Histogram of input image is widely used in image processing and character recognition. H-Recognizer also uses two kinds of histogram as an input feature vector. The one is hole histogram which is obtained by calculating the number of white pixels surrounded by black pixels. And the other is bottom, up, right and left projection histogram. The size of input feature vector is 70, and the structure of neural net classifier is 70-41-10.

Z_1	Z_2	Z_3
Z_4	Z_5	Z_6
Z_7	Z_8	Z_9

3x3 Window

-1	-2	-1
0	0	0
1	2	1

Gx mask

-1	0	1
-2	0	2
-1	0	1

Gy mask

Figure 2. two sobel masks

2.5 C-Recognizer

The contour chain-code is one of the most effective features for representing boundary information of a pattern. Generally, two types of chain-code - 4 direction and 8 direction - is widely used for recognizing handwritten characters[1]. In this research, we applied 8 direction chain-code. In order to obtain chain-code, we extract contour from size normalized image first, and then contour chain-code is acquired. After obtaining contour chain-code, size normalized image is divided into 4x4 sub regions and the chain-code histogram is acquired from each sub regions. For classifying chain-code histogram features, 128-71-10 structured neural network is used.

3 Combinator Description

3.1 Definition

Recognition of pattern is, in a problem made up of M number of classes ($A = \{1, ..., M\}$), deciding on the class to which the unknown input pattern x belongs ($v \in A \cup \{M+1\}$ s.t $M+1$ is reject class) ; and in the combinator, the output result e_k of K number of recognizers is used to come to the final decision $v = F_i(e_k(x))$. At this time, combination function F_i can be divided into the abstract(F_A), Rank(F_R), Measurement(F_M) levels according to the informations supplied by $e_k(x)$ [8]. Each level can be represented by following formula.

$v = F_A(e_k(x))$ s.t $e_k(x) = j_k$ and $j_k \in A \cup \{M+1\}$.
$v = F_R(e_k(x))$ s.t $e_k(x) = \{r_k^i(x) | (1 \le i \le M)\}$.
$v = F_M(e_k(x))$ s.t $e_k(x) = \{m_k^i(x) | (1 \le i \le M)\}$.

3.2 First-stage Combinators

The Bayesian, weighted borda function and neural network method are selected as first stage combinators, because they produce the best performance within each level.

Bayesian method

In the bayesian method, the possibility of error of each recognizer influences the final result. The possibility of error of each recognizer may be illustrated in a confusion matrix and using this, one can educe a conditional probability $P(x \in C_i)|e_k(x) = j)$. Assuming that each recognizer is independent, the bayesian method can be defined following formula[7].

$$F_A(e_k(x)) = \begin{cases} j & \text{if } BEL(j)=\max_{i=0}^{9} BEL(i) \geq \alpha \\ Rej & \text{otherwise} \end{cases}$$

$$BEL(i) = \eta \Pi_{k=1}^{K} P(x \in C_i | e_k(x) = j_k), \text{ for } i = 1,..,\text{M}$$

Borda Function

In this method, "$M - r_k^i$" is allocated according to the output rank r_k^i by recognizer e_k for class i, and the sum of this figure and all recognizers is to be the Borda score for that class; the rank is decided in the order of the score. In this method there is non-weighted Borda function which combines all recognizers with the same weight and the weighted Borda function which combines with different weight. The Borda function can be defined as follows.

$$F_R(e_k(x)) = max_{i \in A}(B_i(e_k(x)))$$

$$B_i(e_k(x)) = \begin{cases} \sum_{k=1}^{K}(M - r_k^i(x)) & \text{Non weighted} \\ \sum_{k=1}^{K} W_k * (M - r_k^i(x)) & \text{weighted} \end{cases}$$

Neural Network

When K number of recognizers output the vector of confidence value for M number of classes, a multilayer perceptron can be composed whose input node number KxM and the output node numbers M. By constructing the neural networks in such a way, the recognizing disposition and characteristic of recognizer are automatically revealed. The method of using neural network can be illustrated as follows.

$$F_M(e_k(x)) = \max_{i \in A} NN_i(e_k(x))$$
$$\text{s.t } NN_i(e_k(x)) = \frac{1}{1+e^{-net_i(e_k(x))}}$$
$$\text{s.t } net_j(e_k(x)) = \sum_{k=1}^{K} \sum_{i=1}^{M} W_{ijk} m_k^i(x) = \theta_j$$

3.3 Final stage combinators

Majority voting and bayesian method are used for final combinators. These method are simple method for combining top choice results from first stage combinators. When the first stage combinator is represented like, $F_i(e_k(x))$ where k=1,..,5 and i= $\{ A, R, M \}$.
the final stage combinator S_V and S_B can be represented following formula.

$$S_V(F_i(e_k(x)) = \begin{cases} j & \text{if } C_F(j)= \max_{i \in A} C_F(i) > 1 \\ Rej & \text{otherwise} \end{cases}$$

$$S_B(F_i(e_k(x)) = \begin{cases} j & \text{if } BEL(j)= \max_{m=0}^{9} BEL(m) \geq \alpha \\ Rej & \text{otherwise} \end{cases}$$

The complete configuration of proposed system is shown in Figure 3.

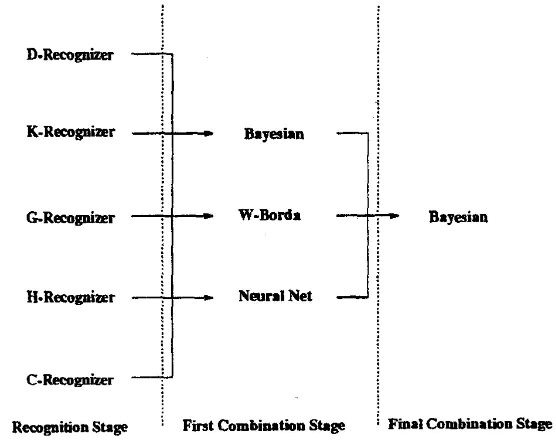

Figure 3. Block diagram of complete system

4 Experimental Results

4.1 Experimental Environments

The proposed system is implemented on Pentium PC and Linux O/S. Two standard handwritten digit databases, CEDAR and CENPARMI DB, are used for verifying the recognition performance of proposed system. In the case of CEDAR DB, 13923 digits in the br directory are used for training recognizers and the remainder are used for cross validation and combinator training and 2,561 digits is the bs directory are used as test data. There are three groups which contains 2,000 digits respectively in the CENPARMI DB. A group is used for training recognizers, B group is used for verifying and combinator training and C group is used as test data..

4.2 Analysis of Experimental Results

The recognition performance of each recognizer is shown in Table 1. As shown in the result table, while G-Recognizer

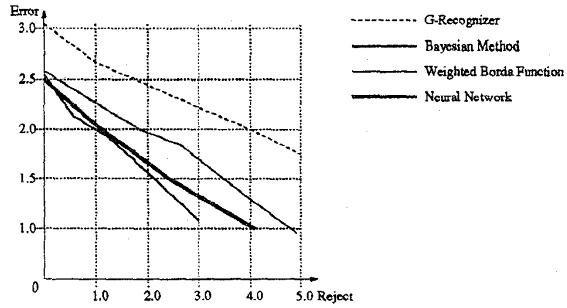

Figure 4. E-R graph for CENPARMI DB

shows the best recognition rate, D-Recognizer shows the worst performance.

Database	Recognizer	Rec	Sub
CENPARMI	D-Recognizer	91.65	8.35
	K-Recognizer	95.85	4.15
	G-Recognizer	96.20	3.60
	H-Recognizer	94.10	5.90
	C-Recognizer	94.80	5.20
CEDAR	D-Recognizer	95.43	4.57
	K-Recognizer	97.50	2.50
	G-Recognizer	97.89	2.11
	H-Recognizer	97.07	2.93
	C-Recognizer	97.34	2.66

Table 1. Performance of each recognizer

Table 2 represents the performance after first stage combination. show similar performances. The performance of combined system is 1 - 1.5% higher than recognizers'. Figure 4 and Figure 5 are error/rejection graph which shows performance comparison between recognizer stage and first combination stage.

Database	Combinator	Rec	Sub
CENPARMI	Bayesian	97.55	2.45
	W-Borda	97.40	2.60
	Neural Net	97.60	2.40
CEDAR	Bayesian	98.36	1.64
	W-Borda	98.40	1.60
	Neural Net	98.40	1.60

Table 2. Performance of first stage combinators

The performances of complete system are shown in Table 3. Even Majority voting which is very simple combination method can increase the recognition performance and Bayesian method shows better performance than Majority

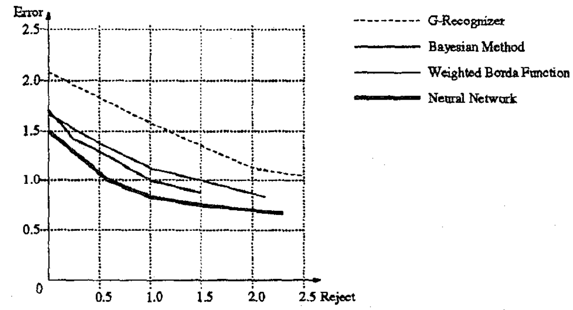

Figure 5. E-R graph for CEDAR DB

Database	Combinator	Rec	Sub
CENPARMI	Voting	97.65	2.35
	Bayesian	97.80	2.20
CEDAR	Voting	98.50	1.50
	Bayesian	98.62	2.38

Table 3. Performance of first stage combinators for CENPARMI test data

voting. Figure 6 represents the performance comparison graph among recognizer, first stage combinator and final stage combinator.

5 Conclusions

As studies on character recognition become more vigorous, a number of researches were carried out to improve recognizer's performance in various forms, among which the most important was combining numerous recognizers. In this paper, we proposed a new architecture of multiple recognizer system which has two combination stages. In

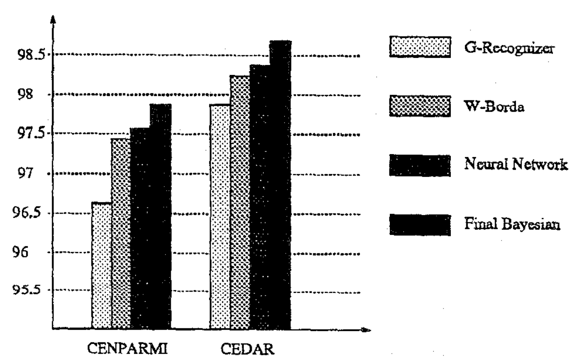

Figure 6. Performance comparison among three stages

the first stage, three combinators combine the results of five recognizers and the next stage final combinator decides final result of the system based on the results of three first stage combinators.

The performance of proposed system was tested by two standard numeral databases, CEDAR and CENPARMI DB. As a consequence of experiments, we verified that proposed system's performance is better than single combinator system's. However, increasing rate was very small and some combinators which were not referenced above experiment didn't show performance increasing. It shows that the behavior of recognizer and combinator is very important in multiple recognizer system. Therefore, for future works, we will minimize the dependency of each recognizer and implement recognizers that are complementary to each other.

References

[1] R. G. Gonzalez, R. E. Woods,"Digital Image Processing," Addison Wesley, 1992.

[2] Y. J. Kim, S-W Lee, "Off-line Recognition of Unconstrained Handwritten Numerals using Multilayer Cluster Neural Network Combined with Genetic Algorithm," The Journal of Korea Information Science Society, pp.1468-1479, 1994

[3] D-S. Lee. S. N. Srihari, "Handprinted Digit Recognition : A Comparison of Algorithms," Proceedings of 3rd IWFHR, pp.153-162, 1993.

[4] T. Matsui, T. Noumi, I. Yamashita, T. Wakahara, M. Yoshimuro, "State of the Art of Handwritten Numeral Recognition in Japan - The Results of the First IPTP Character Recognition Competition," Proceedings of the Second ICDAR, pp.391-396, 1993.

[5] T. Matsui, T. Tsutsumida, S. N. Srihari, "Combination of Stroke/Background Structure and Contour-direction Features in Handprinted Alphanumeric Recognition", Proceeding of 4th IWFHR, pp. 1994.

[6] T. Noumi, T. Matsui, I. Yamashita, "Result of Second IPTP Character Recognition Competition and Studies on Multi-Expert Handwritten Numeral Recognition," Proceedings of 4th IWFHR, pp.338-346, 1994

[7] J. Paik, Y. Lee, "Handwritten Numeral Recognition Using Multiple Combined Method," Proceedings of 1st Workshop on Character Recognition in Korea, pp.156-159, 1993.

[8] L. Xu, A. Krzyzak, C. Y. Suen, "Methods of Combining Multiple Classifiers and Their Applications to Handwriting Recognition," IEEE Trans. on Systems, Man and Cybernetics, vol. SMC-22, no. 3, pp.418-435, 1992.

An Improved Backpropagation Neural Network Learning*

Ivelin Peev **Stoyanov**
Bulgarian Academy of Science, Institute for Information Technologies
"Acad. G.Bonchev" str bl.29A 1113 Sofia, BULGARIA, E-mail : ips@iinf.bg

Abstract :
*The Backpropagation Neural Network (**BPNN**) is a well known and wide used mathematical model for pattern recognition, nonlinear function approximation, time series prediction, etc. There are many applications, which require large input and hidden layers. In such cases, the learning process takes a long time. Many authors propose different methods to reduce the learning time, through convergence improvement.*

*In the present report, a topological method is proposed to cope with this problem. The neurons whose weights tend toward constant values at the learning process are fixed and they are not learned till the end of learning time. The Neural Network learning stops either when the error rate achieves an appropriate minimum, or when the learning time overcomes a constant value. Experiments demonstrate that this method decreases the learning time with about **50%**.*

Keywords: Backpropagation Neural Network, Neural Network Learning, Pattern Recognition, Time Series Prediction.

I. INTRODUCTION

Significant amount of work has been done during the past ten years in the area of neural network modelling. These researches attempt to build a model of neural networks that can solve important problems, such as visual processing, natural language understanding, feature forecasting, etc.

Rumelhart and co., have developed a method *"error back-propagation"*, for learning association between input and output pattern using two or more layers. It is a procedure for learning optimal weights of neurons, which builds up the Long Term Memory. Back-propagation is a supervised learning technique, that compares the responses of the output units with the desired response, and readjusts the weights in the network so that the next time when the same input is presented to the network, the response of the network will be closer to the desired response.

The updating is given by the formula

$$\Delta \mathbf{W}ij = \eta.\delta i.\mathbf{X}j, \quad (1)$$

where $\delta i = \mathbf{f'(hi).(Ti\text{-}Oi)}$ for neurons belonging to the last layer,

or $\delta i = \mathbf{f'(hi).\sum wji.\delta j}$, for other layers.

Ti and **Oi** are target and computed output values for the input pattern.

The above algorithm is a steepest gradient descent, i.e. each learning step goes to the steepest direction toward a valley of the surface error.

Many extensions and modifications of the **BPNN** have been developed during the past years. Their basic goals are to make learning faster, to avoid the local minima and to improve the generalization ability. Most of them focus mainly on the updating rule, keeping a fixed feed-forward architecture, such as: alternative cost function [3] , addition of a momentum term[4], adaptive parameters[5,8]. What about the topology optimization, there are works that consider unimportant units' pruning[1,7] and non-useful connection removing, during training [6]. They both remove the units or weights, which outputs or values tend to zero.

Our intensive investigations revealed that during the learning time, some of the hidden neurons (over 50%) change their weights significantly slower than the other ones, i.e. their weights go toward constant value. Using that as a basis, we propose an extension of the learning process, that examines the updating weight values of each hidden neuron for single learning step and for a learning epoch (a couple of learning cycles). If during the learning they go toward zero, then a procedure fixes the corresponding nodes and interrupts their learning. This operation reduces the total learning time almost twice.

The method is explained in the next chapter, and an application of it is demonstrated in the 3-th one.

* This Paper is partially supported by National Science Foundation Grant I-411/94