Title	Off-line Chinese handwriting recognition using multi-stage neural network architecture
Author(s)	Jin, Lianwen; Chan, Kwokping; Xu, Bingzheng
Citation	Ieee International Conference On Neural Networks - Conference Proceedings, 1995, v. 6, p. 3083-3088
Issued Date	1995
URL	http://hdl.handle.net/10722/45565
Rights	©1995 IEEE. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE.

Off-line Chinese Handwriting Recognition using Multi-stage Neural Network Architecture

Lianwen JIN[†‡], Kwokping CHAN[‡], and Bingzheng XU[†]

†(The Institute of Radio Engg. & Auto., South China University of Technology, GuangZhou, 510641, China)
‡(Department of Computer Science, The University of Hong Kong, Pokfulam Road, Hong Kong)
{lwjin, kpchan}@csd.hku.hk, ecbzxu@scut.edu.cn

Abstract

In this paper, we propose a Multi-stage Neural Network Architecture (MNNA) which integrates several neural networks and various feature extraction approaches into a unique pattern recognition system. General mechanism for designing the MNNA is presented. A three-stage fully connected feedforward neural networks system is designed for Handwritten Chinese Character Recognition (HCCR). Different feature extraction methods are employed at each stage. Experiments show that the three-stage neural network HCCR system has achieved impressive performance and the preliminary results are very encouraging.

1 Introduction

Current interest in Artificial Neural Networks (ANN) is largely a result of their ability of learning, self-organization, and mimicking human intelligence. Neural computing techniques are expanding rapidly and have been used in a broad range of application, including pattern recognition, image processing, computer vision, functional approximation and automatic control. There are a lot of research and application activities in many paradigms. As a result, a large number of tools of neural networks which can be applied to Optical Character Recognition (OCR) are obtained. Indeed, applications to OCR were among the first to demonstrate the capabilities of neural networks [1-5,11,12].

Handwriting recognition has been a problem of considerable academic and industrial interests for many years[14, 15]. The main difficulty of handwriting recognition is that there are great variabilities for different writing styles. Good progress have been made in recognition of handwritten numerals and English letters[6, 7, 11]. However, for Handwritten Chinese Character Recognition (HCCR), due to a large vocabulary (more than 3755 Chinese characters are daily used) and many mutual similarities among different categories of characters, it is regarded as a very challenging problem. Although a lot of HCCR systems achieving high recognition rate have been reported, the performance of these systems would highly depend on the quality and degree of variations of the database used[17]. It may be safer to say that no single system so far developed is likely to achieve high recognition rate. This situation has led to recent efforts of designing more sophisticated system. As each single recognition scheme has its own merit and weakness, it is expected that performance can be improved significantly by combining several schemes, resulting in the so-called multi-experts system [7]. In fact, there have already been several papers that reported on the efficiency of multi-experts system in character recognition [6, 9, 10].

In this paper, from the merits of multi-experts system and neural computing techniques, we propose a multi-stage neural network architecture (MNNA) for Chinese handwriting recognition. A three-stage neural network system is built. Each stage consists of a feature extractor, which is employed to concentrate the input patterns into a few concise and highly representative features, and a neural networks classifier, which is employed to make an intelligent recognition decision. As the multilayer feedforward networks trained by back-propagation is good for classification and has been shown in various papers that it can approximate decision surface of arbitrary complexity[18, 19], we will use three topological different multilayer feedforward networks in our experimental system. Three feature extraction methods are designed. 10,000 handwritten characters from 100 categories are used to train each of the three neural networks. Another 5,000 characters are used to test the performance of the system. While each single-stage neural network produces the recognition rate of 90.92%, 89.58% and 88.06% respectively, the com-

bined system achieve the overall recognition rate of 99.34%, showing that the multi-stage architecture do help to improve the performance of the recognition system efficiently.

2 Design of Multi-stage Neural Network Architecture

The multi-stage neural network system is designed to seek an optimal combination of several complementary neural network based recognition schemes. A general block diagram of multi-stage neural networks architecture is depicted in figure 1.
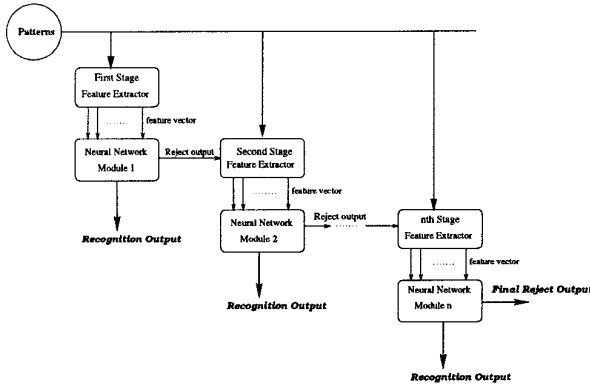

Figure 1: *Multi-stages Neural Networks Architecture*

In figure 1, input patterns are first put into the first stage module, which consists of a feature extractor and a NN classifier. The patterns will be represented by a set of feature vectors after the processing of the feature extractor. These feature vectors constitute the basis for making a decision by the neural network classifier—either acceptance for recognition or rejection. It could be assumed that the features extracted at the first stage will be most suitable for a subset of input patterns, which will be successfully recognized at this stage. The remaining patterns, which are rejected by the first-stage NN classifier, are thus put into the second-stage for further discrimination. Generally, at the nth-stage, the input is a subset of the patterns specified by the $(n-1)$th-stage NN classifier. Decision made at this stage will be based on the feature vectors extracted at the same stage, and the decision will define a subset of rejected patterns to be put into the next stage for further recognition. Recognition results will be declared at any stage for those patterns which are accepted to make a decision. The final stage contains the final recognition outputs and the overall rejection outputs.

From the procedure described above, it can be observed that the key of the MNNA lies in the following two aspects:

- Proper design of neural network classifier which are capable of rejecting spurious patterns at each stage. This is the fundamental of the multi-stage NN architecture. To achieve a good overall performance, neural network classifier at each stage should be able to reject all ambiguous patterns, to make the substitution rate as low as possible.

- Proper and efficient extraction of features at each stage. It is desirable that the selection of features at each stage would be as dependent as possible. Therefore, their complementary characteristics would be used in the combination phase.

Suppose the recognition rate, substitution rate and rejection rate at the ith stage are r_i, s_i, u_i respectively, if the feature extractors and NN classifiers at each stage are independent form one another, then the overall recognition rate for a n-stage system will be:

$$
\begin{aligned}
R_n &= r_1 + r_2(1 - r_1 - s_1) + r_3(1 - r_1 - s1)(1 - r_2 - s_2) + \cdots \\
&\quad + r_n(1 - r_1 - s_1)(1 - r_2 - s_2) \cdots (1 - r_{n-1} - s_{n-1})
\end{aligned}
$$

$$
\begin{aligned}
&= 1 - (1 - r_1 - s_1)(1 - r_2 - s_2) \cdots (1 - r_{n-1} - s_{n-1}) \\
&\quad - s_1 - s_2 u_1 - s_3 u_1 u_2 - \cdots - s_n u_1 u_2 \cdots u_{n-1}
\end{aligned}
$$

$$\tag{1}$$

It is interesting to note that in order to achieve good overall performance, it is essential that the substitution rate s_i should be low enough at each stage, but it is not necessary that the recognition rate r_i at each single stage are critically high. This property is very useful, since it is usually easier to design a recognition system with higher rejection rate and lower error rate rather than design a system with high recognition rate.

As a special and ideal case, considering if $s_i = 0$ for all $i = 0, 1, \cdots, n$, we will have:

$$R_n = 1 - (1 - r_1)(1 - r_2) \cdots (1 - r_{n-1}) \tag{2}$$

In this case, it can be easily proved that:

$$\lim_{n \to \infty} R_n = 1 \tag{3}$$

This means that under the situations when the substitution rate is zero at each stage, and when all feature extractors and NN classifiers are independent from one another, we can achieve almost 100% recognition rate by adding the number of modules.

One obvious advantage of the multistage system is that the recognition rate of each single neural network classifier does not need to be very high, but the overall recognition performance could still

be good enough. As an example, consider a three-stage system with the recognition rate, substitution rate and rejection rate of 0.8, 0, 0.2 for each single stage classifier respectively, the overall recognition rate will be:

$$R_3 = 1 - (1-0.8)(1-0.8)(1-0.8) = 0.992 = 99.2\%$$

Once again, we emphasize that to achieve good recognition performance for a multi-stage neural networks system, it is essential that the substitution rate of each individual neural classifier should be low enough, or more practically, the classifiers at the front stages should have low substitution rates.

3 Feature Extraction

It is evident that feature extraction and selection plays a central role in most pattern recognition system. In fact, the selection of appropriate sets of features which take into account different handwriting styles is also the heart of an OCR system. In our experiments, three sets of statistical features are used:

- Weighted Stroke Cross Counting Feature (WSCCF).
- Weighted Peripheral Feature (WPF).
- Pixel Probability Distribution Feature (PPDF).

All handwritten Chinese characters have been normalized to 64×64 image patterns before the feature extraction are taken.

3.1 Weighted Stroke Cross Counting

Let $f(i,j)$ denote the pixel value at position (i,j) of a 64×64 character image.

$$f(i,j) = \begin{cases} 1, & \text{if } (i,j) \text{ is a black pixel} \\ 0, & \text{if } (i,j) \text{ is a white pixel} \end{cases} \quad (4)$$

$$1 \leq i \leq 64, 1 \leq j \leq 64.$$

Imagine there are several raster scanning of the characters taken in the four direction of vertical, horizontal and two orthogonal diagonals, as shown in figure 2. The cross counting along a raster is defined as the number of times of the raster crosses the character strokes.

In figure 2, totally 48 fixed rasters are used. Thus 48 cross counting features are extracted along these rasters. However, as there are many variations among different handwritings, the features extracted along the same fixed rasters may not be invariant. Therefore, while keeping the 48 rasters as main rasters, we also take the neighbours
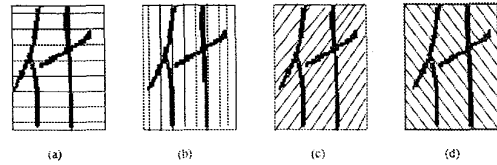

Figure 2: *Stroke cross counting features taken in four directions. In (a) and (b), eight equally spaced and parallel rasters are used to cover the character for horizontal and vertical direction. In (c) and (d), sixteen rasters are used to cover the character for the two diagonal direction.*

of each raster into consideration. A weight vector $W[9] = (0.2, 0.4, 0.6, 0.8, 1.0, 0.8, 0.6, 0.4, 0.2)$ is used for the purpose of including contributions of the eight neighbours of each main raster, in the hope that these additional weighted contributions will make the features extracted represent more invariant information of a handwritten characters. In fact, we have found by experiments that the weighted features are better than non-weighted features when using them in recognition.

The weighted stroke cross counting features along a raster I_i at the horizontal direction is defined as:

$$WCCH_i = \frac{1}{2} \sum_{k=-4}^{4} W[k+4] \sum_{j=1}^{N} f(I_i + k, j)$$
$$\overline{f(I_i + k, j + 1)} \quad (5)$$

Here $N = 64$, $I_i = 8i+4$ for $i = 0, 1, 2, ..., 7$, $\overline{f(I_i, j)}$ is the negative function of $f(I_i, j)$.

Similarly, the weighted stroke cross counting features taken in the direction of vertical and the two diagonal are given by:

- Vertical direction:

$$WCCV_j = \frac{1}{2} \sum_{k=-4}^{4} W[k+4] \sum_{i=1}^{N} f(i, J_j + k)$$
$$\cdot \overline{f(i+1, J_j + k)} \quad (6)$$
$$J_j = 8j + 4, \quad j = 0, 1, \cdots, 7$$

- Right slant diagonal direction:

$$WCCL_i = \frac{1}{2} \sum_{k=-4}^{4} W[k+4] \sum_{j=1}^{N-8i-4} f(I_i + k + j, j)$$
$$\cdot \overline{f(I_i + k + j, j + 1)} \quad (7)$$

$$WCCL_{j+8} = \frac{1}{2} \sum_{k=-4}^{4} W[k+4] \sum_{i=1}^{N-(8j+4)} f(i+1, i + J_j + k)$$
$$\cdot \overline{f(i, i + J_j + k)}, \quad (8)$$

$$I_i = 8i + 4, \quad i = 0, 1, \cdots, 7$$
$$J_j = 8j + 4, \quad j = 0, 1, \cdots, 7$$

- Stroke cross counting features taken in the left slant diagonal direction can be defined in the same way.

3.2 Weighted Peripheral Features

Peripheral feature is a kind of contour feature[16]. The first order peripheral feature is constructed by taking the distance from each points on four edges of a character image frame to a stroke boundary encountered first, as shown in figure 3(a). The second-order peripheral feature is defined as the distance from the edge point to a stroke boundary encountered second, as shown in figure 3(b).
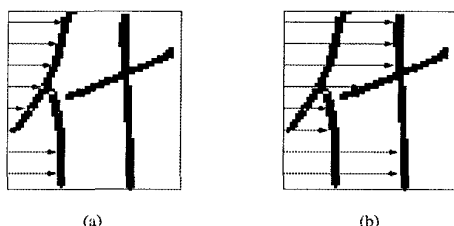

(a)   (b)

Figure 3: *Peripheral Features. In (a), eight first-order peripheral features is constructed from the left edge of the character. In (b), the second-order peripheral features are illuminated.*

Eight equally spaced edge points at each of the four edge of a character frame are specified. Again, the weighting technique stated in the previous section are employed here to include the weighted contributions of the neighbours of every specified edge point. 32 dimensional first-order weighted peripheral features and 32 dimensional second-order weighted peripheral features are finally extracted by this way, giving a total of 64 dimensions feature vector.

3.3 Pixel Probability Distribution Feature

The third kind of feature we used is similar to that described in [13]. The difference is that the re-grouped grids we used are 8×8, not 10×10 which was used in [13]. Therefore, a 256 dimensional feature vector is extracted. Comparing with the 400 dimensional feature vector used in [13], the feature space of our method is much smaller and more practicable.

However, the 256 dimensional feature space might also be too large for a neural-based system. Take a three-layers feedforward network as an example. Suppose the number of input layer neurons and output neurons is 256 and 100 respectively, if 400 neurons are used in the hidden layer,

then there would be totally 142,400 free parameters(links) for this networks. To train such network would be very slow. Therefore, we applied the K-L transform (KLT) to transform the 256 dimensional feature space into a smaller 128 dimension feature space, while still keeping 92.6% of the accuracy.

4 Experiments

4.1 Experimental Data

100 categories of handwritten Chinese characters are used in our experiment, as shown in figure 4. Each category contains 150 different variations (distorted patterns are included). 100 characters from each category are randomly picked out as training data, giving total 10,000 training characters. The remaining 5,000 characters are used as testing data.
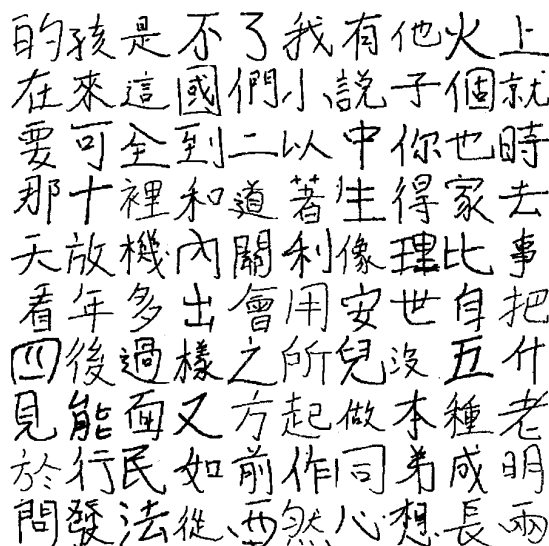

Figure 4: 100 categories of experimental datum.

4.2 Experimental System Architecture

A three-stages neural network architecture is designed in our experimental system, as shown in figure 4. Each stage consists of a three-layer feedforward neural network and a distinct feature extractor. Each neural network is trained by the standard backpropagation algorithm. The complexities of the three networks are described in table 1.
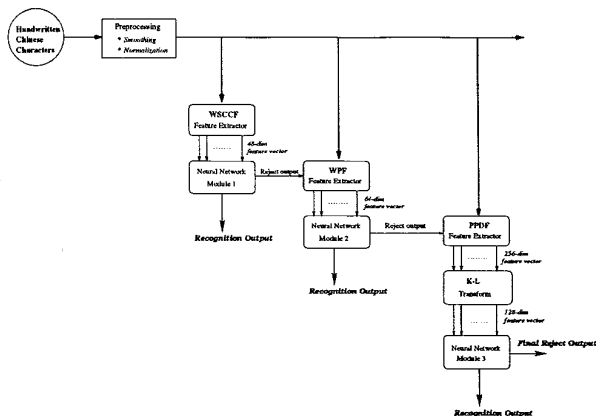
Figure 5: *Block diagram of a three-stage neural network system for handwritten Chinese character recognition*

Stage	Input neurons	Hidden neurons	Output neurons	Links	Features used
Stage one	48	108	100	15984	WSCCF
Stage two	64	96	100	15744	WPF
Stage three	128	128	100	29184	PPDF

Table 1: *The structures of the neural networks used at each stage*

4.3 Rejection strategies

As mentioned in section 2, to achieve good performance for a multi-stage neural network system, it is essential that neural networks at each stage should be capable of rejecting erroneous patterns to make the substitution rate as low as possible. Erroneous patterns include ambiguous patterns, which are close to the border between two classes, and outliers, which set far outside the range covered by the distribution of learning samples[4].

Let $\vec{X}$ be the input feature vector presented to the network, i_1 and i_2 be the indices of two most active output neurons, and f_1, f_2 be their output values. The following two simple rejection criteria will be used:

1) If $f_1 \leq \theta_1$, then reject $\vec{X}$. This criterion is designed to deal with outliers, in which case f_1 should be low.

2) If $|f_1 - f_2| \leq \theta_2$, then reject $\vec{X}$. This criterion is designed to deal with ambiguous patterns, in which case f_1 and f_2 should be similar.

Here θ_1 and θ_2 are some proper parameters predetermined.

4.4 Experimental result

Training each of the three multi-layer neural networks are independently performed on the 10,000 samples. Then the combined system is tested on the 5,000 testing characters. A test character is first input to the first-stage module. If the NN classifier at the first stage accepts it to make a decision, it will declare the recognition result on this character, and then the next test character will be input. Otherwise, this character is rejected and will be put into the second module for another approach of recognition. If the character is also rejected by the second stage, the third stage module will be activated and recognition decision will be finally made at this stage. In our experiment on 5000 handwritten characters, 4536 characters were corrected recognized at the first stage, 8 characters were mis-classified, and 456 characters were rejected. When the rejected 456 characters were input to the second stage, 6 characters were mis-classified and 130 characters were rejected. Finally, 1 character was mis-classified and 18 characters were rejected at the third stage. Therefor, the overall recognition rate of the proposed three-stage system is 1-(8+6+1+18)/5000 = 99.34%. Comparing with the recognition rate of 90.92%, 89.58% and 88.06% obtained by each single module, it is shown that the performance have been improved significantly by the multi-stage architecture. Detail experimental results are given in table 2.

Stage	Error rate	Rejection rate	Recognition rate
Stage one	0.16%	9.12%	90.72%
Stage two	0.22%	10.2%	89.58%
Stage three	0.38%	11.56%	88.06%
Overall	0.3%	0.36%	99.34%

Table 2: *Recognition results of each single stage neural network classifier and their combined performance.*

5 Conclusion and Discussion

We have presented a multi-stage neural network architecture for handwritten Chinese character recognition. It has been shown by experiments that the multi-stage architecture is very helpful to improve the performance of the recognition system.

Two important issues of designing a MNNA system are:

- The selection and extraction of independent and complementary features.

- The design of high discriminative neural networks which are capable of rejecting spurious patterns.

Three simple fully connected multi-layer feedforward networks with standard backpropagation training algorithm were applied to handwritten Chinese character recognition. Although our experiments were performed only on a small set of Chinese characters, the results obtained were very promising and encouraging. However, for a larger vocabulary, with the corresponding increase in the number of the neurons and links of the network, training using the BP algorithm will become rather slow. In such case, to construct alternative neural network structures with faster training algorithm, such as the Radial Basis Function (RBF), will be our further research interest.

References

[1] Josef Skrzypek, Jeffrey Hoffman, "Visual Recognition of Script Characters and Neural Network Architectures", Neural Networks: Advance and applications, E.Gelenbe (Editor), Elserier Sci. Pub., pp. 109-114, 1991.

[2] Y.Lecun, B.Boser, J.S. Denker et al., "BP applied to a handwritten Zip code recognition", Neural Computation 1, pp. 541-551,1989.

[3] I. Guyon, "Application of Neural Networks to Character Recognition", International Journal of Pattern Recognition and Artificial Intelligence, Vol.5 , pp. 353 - 382, 1991.

[4] Francoise Fogelman Soulie, E. Viennet, B. Lamy, "Multi-modular neural network architectures: Applications in Optical character and human face recognition", International Journal of Pattern Recognition and Artificial Intelligence, Vol. 7, No. 4, pp. 721 - 755, 1993.

[5] C.J. Burges, J.I. Ben, J.S. Denker, Y.Lecun, C.R. Nohl, "Off line recognition of handwritten postal words using neural networks", International Journal of Pattern Recognition and Artificial Intelligence, Vol. 7, No. 4, pp. 689-704, 1993

[6] Toshihiro Matsui, Ikuo Yamashita and Toru Wakahara, "The results of the First IPTP Character Recognition competition and studies on Multi-expert recognition for handwritten numerals", IEICE Trans. on Info. Syst. Vol. E77-D, No.7, pp. 801-808, July 1994.

[7] C.Y. Suen, R. Legault, C.Nadal, M.Cheriet and L. Lam, "Building a new generation of handwriting recognition system", Pattern Recognition Letters, Vol.14, No. 4, pp. 303 - 315, 1993.

[8] Marek W. Kurzynski, "On the Multistage Bayes Classifier", Pattern Recognition Vol. 21, No. 4, pp. 355-35, 1988.

[9] F. Kimura and M. Shridhar, "Handwritten Numerical Recognition Based on Mulitiple Algorithms", Pattern Recognition, Vol. 24, No. 10, pp. 969-983, 1991.

[10] Y.S. Huang and C.Y. Suen, "Combination of Multiple Classifiers with Measurement Values", in Proceedings of the Third Internal Conference of Document Analysis and Recognition, Japan, pp. 598 - 601, 1993.

[11] Amar Gupta, M.V. Nagendraprasad, A. Liu, P.S.P. Wang, S.Ayyadirao "An integrated architecture for recognition of totally unconstrained handwritten numerals", International Journal of Pattern Recognition and Artificial Intelligence, Vol. 7, No. 4, as above, pp. 757-773, 1993.

[12] Stefan Knerr, Leon Personnaz, and G. Dreyfus, "Handwritten Digit Recognition by Neural Networks with Single-Layer Training", IEEE Trans on Neural Networks, Vol. 3, No. 6, Nov, 1992.

[13] Tze Fen Li, and Shiaw Shian Yu, "Handprinted Chinese Character Recognition Using The Probability Distribution Feature", International Journal of Pattern Recognition and Artificial Intelligence, Vol. 8, No.5, pp. 1241-1258, Oct. 1994.

[14] Shunji Mori, Kazuhiko Yamamoto, and M. Yasuda, "Research on Machine recognition of Handprinted characters", IEEE Trans. on Pattern Analysis, Machine Intelligence, Vol. 6, No. 4, pp. 386-405, 1984

[15] Shunji Mori, C. Y. Suen and Kazuhiko Yamamoto, "Historical Review of OCR Research and development", Proceedings of the IEEE, Vol. 80, No.7, July, 1992.

[16] Mitsuru Ohkura, Yasuhiro Shimada, Mitsuru Shiono and R. Hashimoto "On Discrimination of Handwritten Similar Kanji Characters", in Proceedings of the third International Conf. on Document Anal. and Recognition, pp. 589-592, 1993.

[17] Jun S. Huang, "Optical handwritten Chinese character recognition", Handbook for Patter recognition and Computer Vision, Eds. C.H.Chen, L.F.Pau, and P.S.Wang, World Sci Pub. Co., pp. 595-624, 1993.

[18] Donr Hush and Bill G. Horne, "Progress in Supervised Neural Networks", IEEE Signal Processing Magazine, vol. 10, no. 1, pp. 8-39, 1993.

[19] K. Hornik, "Multilayer feedforward networks are universal approximators", Neural Networks, vol. 2, pp. 359-366, 1989.