Multilinguistic Handwritten Character Recognition by Bayesian Decision-Based Neural Networks

Hsin-Chia Fu, *Member, IEEE*, and Yeong Yuh Xu

***Abstract*— In this paper, we present a Bayesian decision-based neural network (BDNN) for multilinguistic handwritten character recognition. The proposed self-growing probabilistic decision-based neural network (SPDNN) adopts a hierarchical network structure with nonlinear basis functions and a competitive credit-assignment scheme. Our prototype system demonstrates a successful utilization of SPDNN to the handwriting of Chinese and alphanumeric character recognition on both public databases (CCL/HCCR1 for Chinese and CEDAR for the alphanumerics) and in-house database (NCTU/NNL). Regarding the *performance*, experiments on three different databases all demonstrated high recognition (86–94%) accuracy as well as low rejection/acceptance (6.7%) rates. As for the processing speed, the whole recognition process (including image preprocessing, feature extraction, and recognition) consumes approximately 0.27 s/character on a Pentium-100 based personal computer, without using a hardware accelerator or coprocessor.**

***Index Terms*— Bayesian decision-based neural networks, optical character recognition, self-growing probabilistic decision-based neural networks, supervised learning.**

I. Introduction

In RECENT years, there has been a significant increase in the electronic management of information by multimedia information systems. The handling of multimedia documents consists of the editing and display of texts, graphics, images, and handwriting. Currently, the keyboard and the mouse are still the dominant input devices for personal computer-based multimedia systems. However, in preparing a first draft and concentrating on content creation, pencil and paper are often superior to keyboard entry. By incorporating character recognition with a text-to-speech technology, converting handwriting directly to voice will be an interesting multimedia application.

A. Overview of Handwriting Recognition

The machine recognition of characters has been a topic of intense research since the 1960's [1]–[4], [11], [14]–[17]. After more than 30 years of rigorous attacks, studies in the field of handwriting recognition remain as active as ever. A comprehensive set of references to recent research and developments in *handwriting recognition* can be found in [4] and [14]. These two issues describe not only the state of the art but also, in many cases, ongoing research efforts. To search for the most updated activities, achievements, and demonstrations on handwriting recognition, the World Wide Web (WWW) seems to be an excellent medium for the researcher and, in fact, for anyone interested.

1) Current Status and Related Problems: In this paper, multilinguistic documents are considered as a mixture of two or more kinds of characters with different graphical structures, such as Chinese mixed with English. Documents containing several languages with the same or similar graphical structures, such as most of the western languages, are not the interest of this research because they can be considered as one type of character from the recognition point of view. However, Chinese (including Japanese Kanji) characters are unique and different from those of western languages in that they are nonalphabetic and have quite complicated stroke structures. In general, directly applying several monolanguage character recognition techniques to each individual type of character in a multilinguistic document can be quite difficult, owing to the following.

1) Separating mixed characters of different languages efficiently and correctly can sometimes be as hard as recognizing characters.
2) Implementing two or more different types of recognition modules in a system is not time and space (in both software and hardware) efficient.
3) Combining recognition results from two different types of recognition modules is somewhat unnecessary and nonproductive.

Therefore, it is desirable to design a uniform recognition architecture for multilinguistic character recognition. First, select a set of general features for characters of different languages in a document so that every character can be represented by a uniform feature vector. Compared with a large character set like Chinese,[1] alphanumerics can be considered to be a small subset of special characters to the larger character set. Then, a character recognition architecture for large character set can be adopted directly (or with minor modification) to multilinguistic character recognition. Thus, the uniform feature selection and the recognition architecture can be applied.

Recently, probabilistic decision-based neural networks (PDBNN's) that have the merits of both neural networks and statistical approaches were proposed to attack face and other biometrics recognition systems [13]. The modular structure in

[1] There are more than 40 000 modern Chinese characters, and 5401 characters are used frequently in daily life.

Manuscript received July 24, 1997; revised January 12, 1998. This work was supported in part by the National Science Council, R.O.C., under Grant NSC 85-2213-E009-125. The associate editor coordinating the review of this paper and approving it for publication was Prof. Yu-Hen Hu.

The authors are with the Department of Computer Science and Information Engineering, National Chiao Tung University, Hsinchu, Taiwan, R.O.C.

Publisher Item Identifier S 1053-587X(98)07060-3.

the PDBNN devotes one of its subnets to the representation of a particular person's face images. In each subnet, the discriminate function of a PDBNN is in the form of fixed number of mixtures of Gaussian distributions. This yields extremely low false acceptance and rejection rates for the face recognition systems.

However, the stroke complexity of multilinguistic characters varies from one stroke up to a few dozens of strokes. Hence, a fixed number of a mixture of Gaussian distributions is not suitable for the representation of character pixel distribution. In this paper, we propose a new PDBNN called the self-growing probabilistic decision-based neural network (SPDNN) to recognize multilinguistic characters. For different characters, the discriminate function of a SPDNN is in a form of a flexible number of mixture of Gaussian distributions.

This paper is organized as follows. Section II presents the mathematical background, the architecture, and the learning rules of the SPDNN. Then, the multilinguistic handwriting recognition system is presented in Section III. The proposed system consists of three modules, which are all implemented by the SPDNN. A personal adaptive module is recommended to further improve the recognition performance. Experimental results of these modules are provided and discussed in each section.

II. Self-Growing Probabilistic Decision-Based Neural Network

The self-growing probabilistic decision-based neural network (SPDNN) is a probabilistic variant of the decision-based modular neural network [10] for classification. One subnet of an SPDNN is designed to represent one object class. There are two properties of the SPDNN learning rules. The first one is the decision-based learning rules. Based on the teacher information that only tells the correctness of the classification for each training pattern, an SPDNN performs a distributed and localized updating rule. The updating rule applies *reinforced learning* to a subnet corresponding to the correct class and *antireinforced learning* to the (unduly) winning subnets.

The second property is the iteratively supervised learning and unsupervised growing (ISLUG). There are two learning phases in this scheme. After each subnet is initialized with one cluster (see Section II-B1) or is self-grown with a new cluster (Section II-B2), the system enters the supervised learning (SL) phase. In the SL phase, teacher information is used to reinforce or antireinforce the decision boundaries obtained during the initialization or self-growing stages. When the supervised training progress becomes very slow or is trapped in a paralysis state, yet the classification or recognition accuracy is not at a satisfied level, the training enters the unsupervised growing (UG) phase. In the UG phase, an SPDNN creates a new cluster in a subnet according to the proposed *self-growing* rule. Thereafter, the training enters the supervised learning phase again. The ISLUG learning procedure terminates when the training accuracy reaches a predefined satisfaction level. The detailed description of the SPDNN model is given in the following sections.

A. Discriminant Functions of SPDNN

One of the major differences between PDBNN [13] and SPDNN is that SPDNN extends the fixed number of clusters in a subnet of PDBNN to flexible number of clusters in a subnet of SPDNN. That is, the subnet discriminant functions of SPDNN are designed to model the log-likelihood functions of different complexed pixel distribution of handwritten characters. Thus, *reinforced* or *antireinforced* learning is applied to *all* the subnets of the global winner and the supposed (i.e., the correct) winner, with a weighting distribution proportional to the degree of possible involvement (measured by the likelihood) by each subnet. Given a set of iid patterns $\mathbf{X}^+ = \{x(t);\ t = 1, 2, \ldots, N\}$, we assume that the likelihood function $p(\mathbf{x}(t) \mid \omega_i)$ for class ω_i (i.e., a character class) is a mixture of Gaussian distributions.

Define $p(x(t) \mid \omega_i, \Theta_{r_i})$ to be one of the Gaussian distributions that comprise $p(x(t) \mid \omega_i)$, where Θ_{r_i} represents the parameter set $\{\mu_{r_i}, \mathbf{\Sigma}_{r_i}\}$ for a cluster r_i in subnet i.

$$p(\mathbf{x}(t) \mid \omega_i) = \sum_{r_i=1}^{R_i} P(\Theta_{r_i} \mid \omega_i) p\big(\mathbf{x}(t) \mid \omega_i, \Theta_{r_i}\big)$$

where $P(\Theta_{r_i} \mid \omega_i)$ denotes the prior probability of the cluster r_i. By definition, $\sum_{r_i=1}^{R_i} P(\Theta_{r_i} \mid \omega_i) = 1$, where R_i is the number of clusters in ω_i.

The discriminate function of the multiclass SPDNN models the log-likelihood function

$$\begin{aligned}\phi(\mathbf{x}(t), \mathbf{w}_i) &= \log p(\mathbf{x}(t) \mid \omega_i)\\ &= \log\left[\sum_{r_i=1}^{R_i} P\big(\Theta_{r_i} \mid \omega_i\big) p\big(\mathbf{x}(t) \mid \omega_i, \Theta_{r_i}\big)\right]\end{aligned} \quad (1)$$

where $\mathbf{w}_i = \{\mu_{r_i}, \mathbf{\Sigma}_{r_i}, P(\Theta_{r_i} \mid \omega_i), T_i\}$. T_i is the output threshold of the subnet i.

In most general formulations, the basis function of a cluster should be able to approximate the Gaussian distribution with full-rank covariance matrix, i.e., $\phi(\mathbf{x}, \omega_i) = -\frac{1}{2}\mathbf{x}^T \mathbf{\Sigma}_{r_i}^{-1}\mathbf{x}$, where $\mathbf{\Sigma}_{r_i}$ is the covariance matrix. However, for those applications that deal with high-dimension data but a finite number of training patterns, the training performance and storage space discourage such matrix modeling. A natural simplifying assumption is to assume uncorrelated features of unequal importance. That is, suppose that $p(\mathbf{x}(t) \mid \omega_i, \Theta_{r_i})$ is a D-dimensional Gaussian distribution with uncorrelated features

$$\begin{aligned}p\big(\mathbf{x}(t) \mid \omega_i, \Theta_{r_i}\big) = &\frac{1}{(2\pi)^{\frac{D}{2}} \left|\mathbf{\Sigma}_{r_i}\right|^{\frac{1}{2}}}\\ &\cdot \exp\left[-\frac{1}{2}\frac{\big(\mathbf{x}(t) - \mu_{r_i}\big)^T \big(\mathbf{x}(t) - \mu_{r_i}\big)}{\mathbf{\Sigma}_{r_i}}\right]\end{aligned} \quad (2)$$

where $\mathbf{x}(t) = [\mathbf{x}_1(t), \mathbf{x}_2(t), \ldots, \mathbf{x}_D(t)]^T$ is the input pattern, $\mu_{r_i} = [\mu_{r_i 1}, \mu_{r_i 2}, \ldots, \mu_{r_i D}]^T$ is the mean vector, and diagonal matrix $\mathbf{\Sigma}_{r_i} = \mathrm{diag}[\sigma_{r_i 1}^2, \sigma_{r_i 2}^2, \ldots, \sigma_{r_i D}^2]$ is the covariance matrix. As shown in Fig. 1, an SPDNN contains K subnets that are used to represent a K-category classification problem.

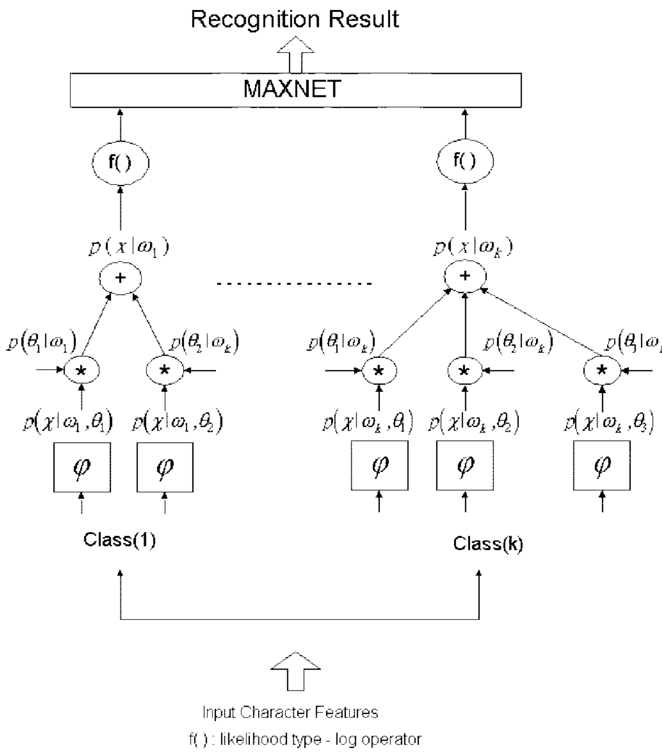


Fig. 1. Schematic diagram of a k-*class* SPDNN character recognizer.

Inside each subnet, an elliptic basis function (EBF) is used to serve as the basis function for each cluster r_i

$$\psi\big(\mathbf{x}(t), \omega_i, \Theta_{r_i}\big) = -\frac{1}{2}\sum_{d=1}^{D} \alpha_{r_i d}\big(x_d(t) - \mu_{r_i d}\big)^2 + \theta_{r_i} \quad (3)$$

where $\theta_{r_i} = -\frac{D}{2}\ln 2\pi + \frac{1}{2}\sum_{d=1}^{D}\ln \alpha_{r_i d}$. After passing an exponential activation function, $\exp\{\psi(\mathbf{x}(t), \omega_i, \Theta_{r_i})\}$ can be viewed as a Gaussian distribution, as described in (2), except for a minor notational change as in $\frac{1}{\alpha_{r_i d}} = \sigma^2_{r_i d}$.

B. Learning Rules for SPDNN

Recall that the training scheme for a multiclass SPDNN follows the ISLUG principle. The ISLUG training scheme contains the following two phases: *supervised learning* and *unsupervised growing*.

1) Supervised Learning in Each Subnet: Since the number of cluster in a subnet of an SPDNN can be adjusted in the unsupervised growing phase, each subnet is initialized with one cluster. The values of the parameters (mean and covariance) of a cluster in each subnet are initialized at the beginning of the first supervise learning phase. Suppose that $\mathbf{X}_i^+ = \{\mathbf{x}_i(1), \ldots, \mathbf{x}_i(M_i)\}$ is a set of given training characters that correspond to one of the L classes $\{\omega_i, i = 1, \ldots, L\}$; the mean μ_i and covariance $\mathbf{\Sigma}_i$ of the initial cluster in subnet i can be calculated as

$$\mu_i = \frac{1}{M_i}\sum_{m=1}^{M_i} \mathbf{x}_i(m) \quad (4)$$

$$\mathbf{\Sigma}_i = \frac{1}{M_i - 1}\sum_{m=1}^{M_i} (\mathbf{x}_i(m) - \mu_i)(\mathbf{x}_i(m) - \mu_i)^T. \quad (5)$$

During the *supervised learning* phase, training data is then used to fine tune the decision boundaries of each classes. The data adaptive scheme of the supervised learning for the multiclass SPDNN is the extension of the GS learning in [13]. Each class is modeled by a subnet with discriminant functions $\phi(\mathbf{x}(t), \mathbf{w}_i), i = 1, 2, \ldots, L$. At the beginning of each supervised learning phase, use the still-under-training SPDNN to classify all the training characters $\mathbf{X}_i^+ = \{\mathbf{x}_i(1), \mathbf{x}_i(2), \ldots, \mathbf{x}_i(M_i)\}$ for $i = 1, \ldots, L$. $\mathbf{x}_i(m)$ is classified to class ω_i if $\phi(\mathbf{x}_i(m), \mathbf{w}_i) > \phi(\mathbf{x}_i(m), \mathbf{w}_k)$, $\forall k \neq i$, and $\phi(\mathbf{x}_i(m), \mathbf{w}_i) \geq T_i$, where T_i is the output threshold for subnet i. According to the classification results, the training characters for each class i can be divided into three subsets.

- $D_1^i = \{\mathbf{x}_i(m); \mathbf{x}_i(m) \in \omega_i, \mathbf{x}_i(m)$ is classified to ω_i (correctly classified set)$\}$.
- $D_2^i = \{\mathbf{x}_i(m); \mathbf{x}_i(m) \in \omega_i, \mathbf{x}_i(m)$ is misclassified to other class ω_j (false rejection set)$\}$.
- $D_3^i = \{\mathbf{x}_i(m); \mathbf{x}_i(m) \notin \omega_i, \mathbf{x}_i(m)$ is misclassified to class ω_i (false acceptance set)$\}$.

The following reinforced and antireinforced learning rules [10] are applied to the corresponding subnets.

Reinforced Learning:

$$\mathbf{w}_i^{(m+1)} = \mathbf{w}_i^{(m)} + \eta\nabla\phi(\mathbf{x}_i(m), \mathbf{w}_i) \quad (6)$$

Antireinforced Learning:

$$\mathbf{w}_j^{(m+1)} = \mathbf{w}_j^{(m)} - \eta\nabla\phi(\mathbf{x}_i(m), \mathbf{w}_j) \quad (7)$$

In (6) and (7), η is a user defined learning rate $0 < \eta \leq 1$, and the gradient vectors $\nabla\phi$ can be computed in a similar manner, as proposed in [13].

For the data set D_2^i, reinforced and antireinforced learning will be applied to class ω_i and ω_j, respectively. As for the false acceptance set D_3^i, antireinforced learning will be applied to class ω_i, and reinforced learning will be applied to the class ω_j, where $x_i(m)$ belongs.

Threshold Updating: The threshold value $\mathbf{T}_i$ of a subnet i in the SPDNN recognizer can also be learned by reinforced or antireinforced learning rules.

2) Unsupervised Growing of a New Cluster: The network enters the unsupervised growing phase when the supervised learning reaches a saturated (learning state) but unsatisfied (classification accuracy) situation. There are three main aspects of the self-growing rules.

- I1) When should a new cluster be created?
- I2) Which cluster should be partitioned to create a new cluster?
- I3) How do we initialize the center and the covariance of the new cluster?

As for Issue I1, when the whole training set has been presented for a few times, the train status (especially the recognition accuracy) remains unchanged or unimproved. An extra cluster is suggested to improve the representation power of the SPDNN.

As for Issue I2, when an extra cluster is needed, a new cluster to be created is suggested from the subnet that caused the most misclassification during the recent supervised learning processes.

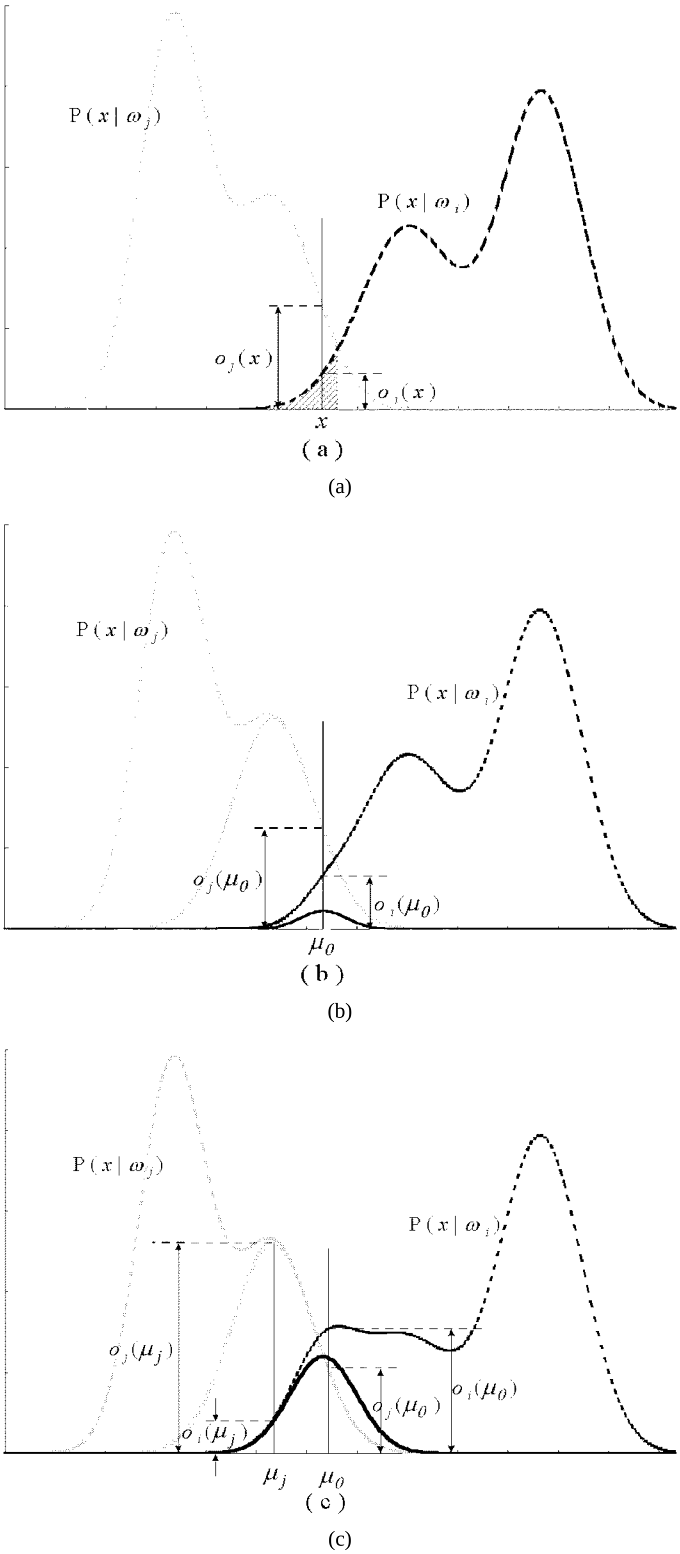


Fig. 2. Example of creating a new cluster in a mixture of Gaussian distributions. (a) For $x(t) \in \omega_i$, the $x_i(t)$ is not correctly classified since $o_i(x(t))$ is smaller than $o_j(x(t))$. A new cluster Θ_i' is needed in ω_i. (b) The new cluster Θ_i' is *overwhelmed* by the cluster Θ_j, i.e., $o_i(\mu_0)$ is still smaller than $o_j(\mu_0)$. (c) By having initialized with proper μ_0, σ_0, and $P(\Theta_0 \mid \omega_i)$, the new cluster Θ_i' can contribute enough to support class ω_i. For example, $o_i(\mu_0)$ is larger than $o_j(\mu_0)$, and $o_i(\mu_j)$ is smaller than $o_j(\mu_j)$.

As for Issue I3, when a new cluster is needed, its initial values of the *center* and *covariance* need to be properly determined; otherwise, a poor classification situation may still exist.

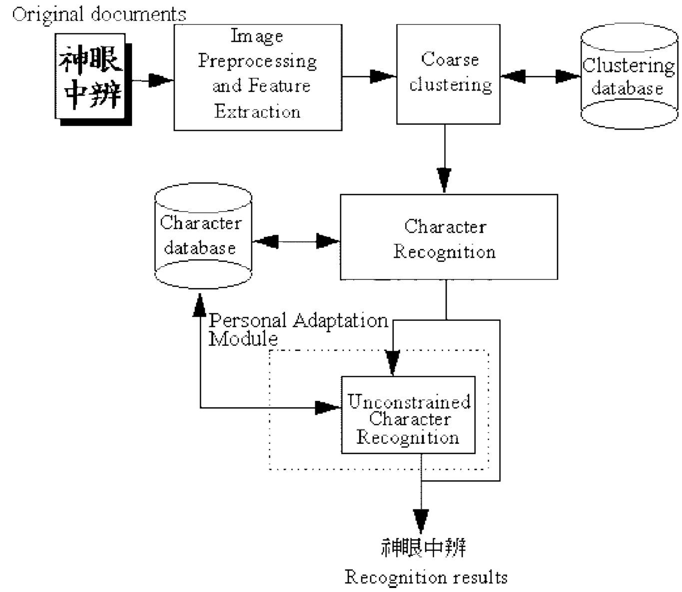


Fig. 3. System configuration of the multistage character recognition system. The character recognition system acquires images from a scanner. The coarse classifier determines an input character image to be one of the predefined subclass. The character recognizer matches the input character with a reference character. The personal adaptation module learns the user's own written style to enhance the recognition accuracy.

Assume that a training character $\mathbf{x}$ corresponding to class ω_i is presented to an SPDNN classifier. The cluster Θ_i is in class ω_i, and the cluster (say, Θ_j) is in the class ω_j, which corresponds to the largest response among the classes other than ω_i. Let o_i and o_j be the output of $\mathbf{x}$ from class ω_i and class ω_j, respectively. According to the retrieving scheme of the proposed SPDNN, if o_j is larger than or equal to o_i, the retrieving result for the training character $\mathbf{x}$ must be wrong. As shown in Fig. 2(a), the best position for the center of the new cluster should clearly be located at $\mathbf{x}$, i.e., $\mu_0 = \mathbf{x}$, so that the class with new cluster Θ_i' will generate the maximal output o_i for the training character $\mathbf{x}$. To determine the covariance matrix $\mathbf{\Sigma}_0$, first let $\mathbf{\Sigma}_0 = \sigma_0 \mathbf{I}$ and σ_0 be a positive constant (to be determined). As shown in Fig. 2(b), if the σ of the new cluster Θ_i' is not properly determined, the class with the new cluster will have its largest possible output $o_i(\mathbf{x})$ be smaller than the output $o_j(\mathbf{x})$ of class ω_j. In other words, the cluster Θ_i' is *overwhelmed* by cluster Θ_j, where μ_j and $\mathbf{\Sigma}_j$ are the center and covariance of cluster Θ_j. To prevent the *overwhelming* problem, Fig. 2(c) presents a properly initiated new cluster Θ_i'. Two constraints are suggested for a proper initial value of σ:

$$
\begin{aligned}
o_j(\mathbf{x}) &= \frac{P(\Theta_j \mid \omega_j)}{(2\pi)^{\frac{D}{2}} |\Sigma_j|^{\frac{1}{2}}} \exp\left[-\frac{1}{2} \frac{(\mathbf{x}-\mu_j)^T(\mathbf{x}-\mu_j)}{\mathbf{\Sigma}_j}\right] + \epsilon_j \\
&< o_i(\mathbf{x}) = \frac{P(\Theta_0 \mid \omega_i)}{(2\pi\sigma)^{\frac{D}{2}}} + \epsilon_i
\end{aligned} \tag{8}
$$

$$
\begin{aligned}
o_i(\mu_j) &= \frac{P(\Theta_0 \mid \omega_i)}{(2\pi\sigma)^{\frac{D}{2}}} \exp\left[-\frac{1}{2} \frac{(\mathbf{x}-\mu_j)^T(\mathbf{x}-\mu_j)}{\mathbf{\Sigma}_i}\right] + \epsilon_i' \\
&< o_j(\mu_j) = \frac{P(\Theta_j \mid \omega_j)}{(2\pi)^{\frac{D}{2}} |\Sigma_j|^{\frac{1}{2}}} + \epsilon_j'
\end{aligned} \tag{9}
$$

where $P(\Theta_0 \mid \omega_i)$ and $P(\Theta_j \mid \omega_j)$ are the prior probability of cluster Θ_i' and Θ_j, respectively. ϵ_i and ϵ_j represent the partial

Fig. 4. Image preprocessing on handwritten characters: (from top) original text image, smoothed text, linear normalized text, nonlinear normalized text and thinned text.

output of classes ω_i and ω_j from the clusters other than Θ_i and Θ_j at $\mathbf{x}$. ϵ_i' and ϵ_j' are the partial output at μ_j of the clusters other than Θ_i and Θ_j. The prior probability $P(\Theta_0 \mid \omega_i)$ of cluster Θ_i' can be initialized as $(\sigma/\bar{\sigma}_i)P(\Theta_i \mid \omega_i)$, where $\bar{\sigma}_i = (1/R_i)\sum_{r_i=1}^{R_i}\sigma_{r_i}$, in which σ_{r_i} is the covariance of cluster r_i in class ω_i. Since ϵ_i, ϵ_j, ϵ_i', and ϵ_j' are very small at $\mathbf{x}$ and μ_i, they are ignored in the following σ estimation.

These two constraints imply that cluster Θ_i' and cluster Θ_j will not overwhelm each other. To satisfy (8), σ is initialized to be less than $(\frac{P(\Theta_i|\omega_i)}{(2\pi)^{D/2}\bar{\sigma}_i o_j(\mathbf{x})})^{\frac{2}{D-2}}$. Then, σ can be iteratively decreased by a small value η $(0 \leq \eta \leq 1)$ until (9) is satisfied, and the final value of σ can be a proper initial value of σ_o for the new cluster Θ_i'.

III. Multistage Handwritten Character Recognition System

An SPDNN-based handwritten character recognition system is being developed in the Neural Networks Laboratory of National Chiao Tung University. The system configuration is depicted in Fig. 3. All the major processing modules, including preprocessing and feature extraction module, coarse classifier, character recognizer, and its personal adaptive module, are implemented on a Pentium-100 based personal computer.

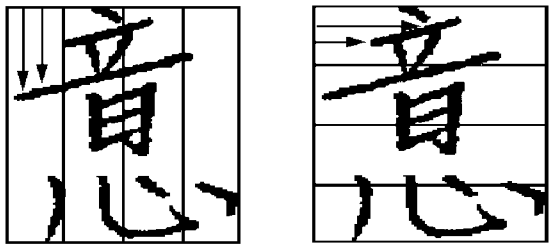
The system built on the proposed SPDNN has been demonstrated to be applicable under reasonable variations of character orientation, size, and stroke width. This system also has been shown to be very robust in recognizing characters written by various tools, such as pencils, ink pens, mark pens, and Chinese calligraphy brushes. The prototype system takes 270 ms on average to identify a character image out of a commonly used Chinese character set (e.g., the 5401 set in [19]) on a Pentium-100 based personal computer. For alphanumerical character recognition, the recognition is about three times faster.

A. Image Preprocessing and Feature Extraction

Image preprocessing of a multilinguistic character recognition is by no means of any different from the monolanguage character recognition. Character segmentation on free format handwritten character is a very difficult task; thus, it is usually an interactive task between segmentation and recognition. Since the multilinguistic character recognition is already a complicated recognition problem, and the interactive segmentation methods would slow down the processing speed, we must restrict our handwritten character domain to be free format on Chinese characters and handprinted characters on alphanumerics. Thus, an *interactive rule-based character segmentation* [5] is applied to slice and separate the whole page image into a sequence of character images. Basically, this method is based on some heuristic rules to combine several isolated connected components into a separated character.

The binary images of a handwritten character are then passed through a series of image processing stages, such as boundary smoothing, noise removing, space normalization, and stroke thinning operations. Fig. 4 depicts a series of preprocessing results of some Chinese and English characters.

Feature extraction. Using statistical features in pattern recognition has been very successful for a long time. A character can be well represented by a two-dimensional (2-D) image pattern; thus many statistical pattern recognition techniques have been applied in this type of character recog-



(a) (b)

Fig. 5. Extracting the CCT and BSPN features from a Chinese character.

nition. Among various statistical features [6], we selected the *crossing count* (CCT), *belt shape pixel number* (BSPN), and *stroke orientation feature* (STKO) as candidate features for the proposed character recognition system. As shown in Fig. 5, features such as CCT and BSPN represent the stroke complexity and the pixel density of a character image. In [9], Kimura *et al.* proposed the directional code histogram and used this feature for Chinese character recognition successfully. The architecture of a multistage SPDNN character recognizer is depicted in Fig. 6. As shown in Fig. 7, the STKO is a simplified version of Kimura's directional code histogram.

B. Multistage Character Recognition

Since there are as many as 5401 commonly used characters and 62 alphanumerics and symbols in a Chinese majored multilinguistic Chinese document, it is desirable to perform a coarse classification (or clustering) to reduce the number of candidate characters for the *character recognition*. With a smaller candidate set, not only can the overall recognition speed and recognition accuracy be greatly improved, but the training on the SPDNN character recognizer can be much easier and faster as well.

1) Coarse Classification: In order to achieve a balanced recognition performance in a multistage recognition system, the coarse classifier needs to maintain a very high accuracy, (e.g., $\geq$ 99.9%). Although this is a difficult task, we proposed to use the CCT feature and the SPDNN with *overlapped* boundaries to implement the coarse classifier. This design is to achieve low sensitivity in personal writing style and high classification rate among characters. By applying the two public databases suggested in Section III-B3 and the ISLUG principle, we have trained the proposed coarse classifier to achieve this goal. The training and testing results are listed in Table I. At the end of the retrieving phase of the coarse classifier, the number of candidate characters with respect to the input character is reduced to 516 characters in average.

2) Character Recognition: The design of the character recognizer is also based on the SPDNN model. For a K-character recognition problem, an SPDNN character recognizer consists of K subnets. A subnet i in the SPDNN recognizer estimates the distribution on the patterns of character i only and treats those patterns that do not belong to character i as the "non-i" patterns. The combined features such as CCT, BSPN, and STKO are used in the SPDNN character recognizer. The training of the character SPDNN was conducted with the ISLUG principle. During the retrieving phase, each of the subnets corresponding to candidate characters from coarse classifier produces a score according to its discriminate function $\phi(\mathbf{x}(t), \mathbf{w}_i)$. The subnet that produces the highest score is the winner, and its corresponding reference character is considered as the result of the character recognizer.

3) Experimental Results: In this section, two experimental results for handwriting recognition will be discussed. In the first part, we used the CCL/HCCR1 [19] handwritten characters database, which have been used by several handwritten character recognition research groups [5], [6], [12]. The second experiment explored the ability of SPDNN in solving the multilinguistic handwritten character recognition problem, which is seldom discussed in the character recognition literature.

a) Experiment 1—Handwritten Chinese recognition: We have conducted experiments on the CCL/HCCR1 database, which contains more than 200 samples of 5401 frequently used Chinese characters. The samples were collected from 2600 people, including junior high school and college students as well as employees of ERSO/ITRI. According to the most recent survey on handwriting recognition [4], [14], [18], most of the handwritten Chinese OCR's were done on small database, i.e., training and testing on very small size of character set, e.g., a few hundred characters. As far as studies conducted on the recognition on a complete set of commonly used Chinese characters, Xia [20] developed an experimental system on the 3755-character set and achieved an 80% recognition rate. Recently, Li and Yu [12] reported a 93.43% recognition accuracy on the CCL/HCCR1 database. Our SPDNN-based system achieved a slightly lower accuracy performance (90.11%). Table II summarizes the performance comparison of these two systems based on the CCL/HCCR1 database. We would like to comment on why the SPDNN has inferior performance. First, compared with the huge number of character feature sets (400–500) used by [12], the features used by SPDNN only consist of 96 sets. A more relevant comparison could be made if comparable number of training and testing features for these two systems were available. In fact, the SPDNN character recognizer is designed to use no more than 100 sets of features since more feature set means more memory storage and longer recognition time. Two things explain why SPDNN can live with fewer features and yet achieve a comparable performance. 1) The mixture of Gaussian-based discriminant function permits the SPDNN to learn the precise decision boundaries. 2) The self-growing rules allow for just enough Gaussian clusters to represent the character image distribution. Therefore, the proposed SPDNN system is very suitable to implement on a personal computer system. Li and Yu's system was implemented on a Sparc-2 workstation.

b) Experiment 2—Multilinguistic handwriting recognition: By searching on major conference proceedings, journals, as well as Web sites, we have yet to find any performance test report on this type of handwriting. We therefore conducted experiments on the combined databases of CCL/HCCR1 and CEDAR [8]. The training and testing data sets for Chinese characters are selected by the same method used in *Experiment 1*. The CEDAR database contains various style of handwritten

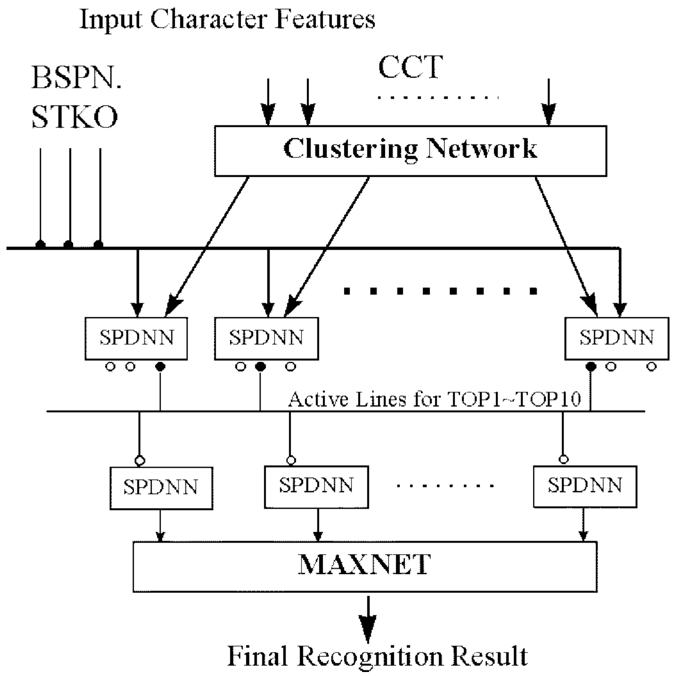


Fig. 6. Architecture of the three-stage recognition system.

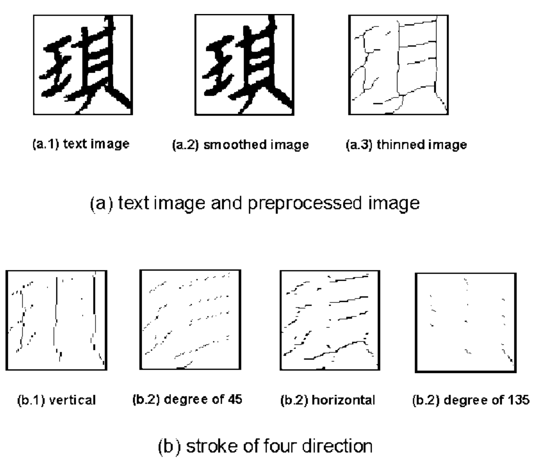


Fig. 7. Preprocessing and STKO feature extraction of a Chinese character.

TABLE I
TRAINING AND TESTING RESULTS OF COARSE CLASSIFICATION ON THE CCL/HCCR1 AND THE CEDAR DATABASES. HALF OF THE RANDOMLY SELECTED CHARACTERS IN EACH OF DATABASES ARE USED FOR TRAINING, AND THE OTHER HALF ARE USED FOR TESTING

Number of cluster	Ave. No. of characters in a cluster	Training Accuracy	Testing Accuracy
61	516	99.9 %	99.8%

alphanumerics, which were lifted from envelop address blocks from the United States. Among the data, 4000 alphanumerics were used for training and 2000 for testing. We also conducted experiments with rejection. Rejection criteria was implemented through the threshold value T_i, which can be learned by the reinforced and antireinforced learning rules. In general, when an input character is correctly recognized with certain confidence, its output of the discriminate function should maintain a certain gap larger than T_i with respect to the second largest output from other discriminate functions.

TABLE II
PERFORMANCE OF DIFFERENT HANDWRITTEN CHARACTER RECOGNIZERS ON THE CCL/HCCR1 DATABASE. THE *Li-yu1* SYSTEM WAS IMPLEMENTED BY THEIR MAIN FEATURES, AND THE *Li-yu2* SYSTEM WAS IMPLEMENTED BY THEIR FULL SET OF FEATURES [12]

Systems	Accuracy	Feature sets	Train/Test ratio
SPDNN	90.12%	96	50-50
Li-yu1	88.65%	400	50-1
Li-yu2	93.43%	500	50-1

TABLE III
PERFORMANCE OF SPDNN HANDWRITTEN CHARACTER RECOGNIZERS WITH AND WITHOUT REJECTION ON THE CCL/HCCR1 AND CEDAR DATABASES

Systems	Top 1 Accu.	Top 2 Accu.	Top 3 Accu.	Rej. %
SPDNN	90.12%	93.49%	94.75 %	0 %
SPDNN	94.11%	97.01%	97.67 %	6.7 %

The experimental results follow: For the sake of comparison, we adjusted the thresholds T_i so that the proposed system has a 0% false rejection rate during the training phase. The recognition accuracy with 0% and 6.7% false rejection rates at the testing phase are shown in Table III. Li and Yu's method can not provide rejection function in their Bayesian rule-based statistical recognition system [12]. However, SPDNN's rejection function is based on the reinforced and antireinforced learning rules; thus, each subnet, which represents a character in SPDNN, can have its own rejection criteria. We think this characteristic is beneficial for real-world applications.

C. SPDNN for Personal Adaptive Recognition

In order to further enhance the recognition accuracy, we proposed personal adaptation on the SPDNN character recognizer.

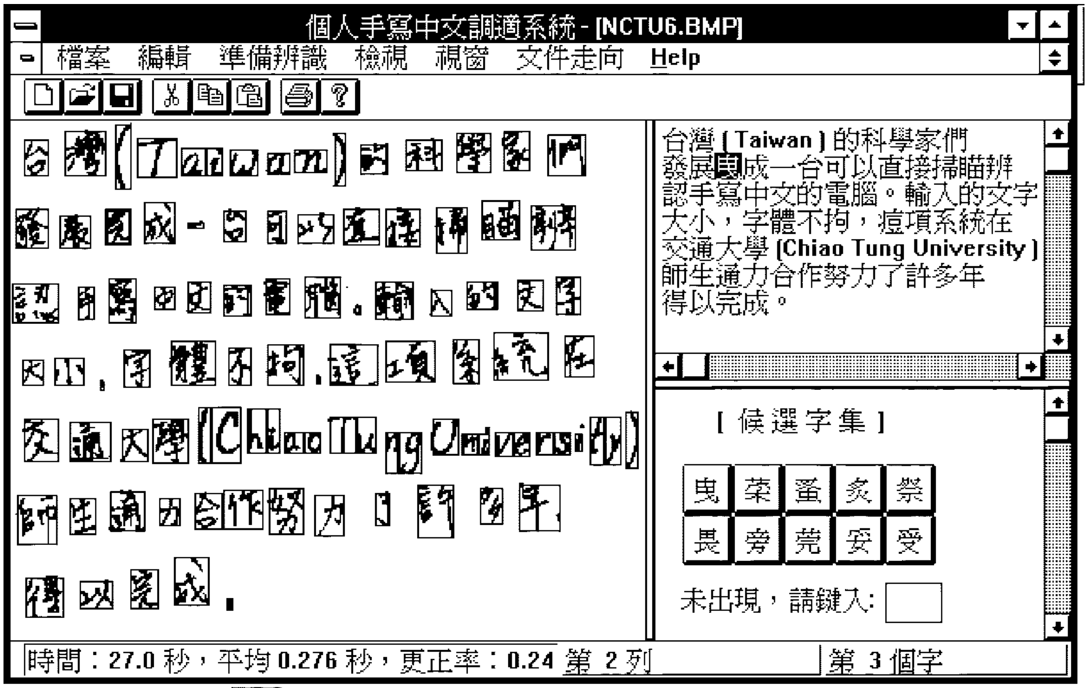


Fig. 8. User interface and a recognition snapshot of the proposed three-stage recognizer.

1) Unconstrained Freehand Written Character Recognition: Most of the recently announced handwritten character recognition systems claimed their benchmarking recognition performance to be higher than 90%. However, when they were tested on unconstrained freehand writing, most of their recognition accuracy fell between 40% and 50% [7]. Hence, we suggested an unconstrained freehand-writing recognition module to adaptively fine tune the parameters of the SPDNN character recognizer in order to learn the user's own writing style. When input characters were misclassified, the erroneous recognition results were manually corrected by a user. In the meantime, the parameters or the decision boundaries of the corresponding character SPDNN are modified and improved by performing the reinforced and antireinforced learning processes. In addition, when it is necessary, clusters in a character SPDNN may be created (self-growing rules) to better approximate the partition boundaries. In order to prevent the excessive learning of the designated character boundary, the adaptive learning process usually includes a verification process. Naturally, the reinforced and antireinforced learning processes are applied to the SPDNN's associated with the mismatched character and its similar characters (the Top 10 candidates). When increasingly more unconstrained freehand written characters are presented to the system, each character SPDNN will gradually learn the user's personal writing style.

2) Experimental Results and Performance Evaluation: In order to evaluate the performance of the unconstrained freehand writing recognition module for its adaptation and recognition capabilities, we prepared our in-house database (NCTU/NNL) in the following manner. We first selected the 300 most commonly used characters from the Chinese textbooks in elementary schools in Taiwan. Then, these 300 Chinese characters and the alphanumerics were written without any restriction on the writing style by several students in our university 10 times over several days. We intended to simulate a natural and general unconstrained freehand written database in this manner. The testing results for five user's adaptation processes are illustrated in Table IV. The recognition rates was raised from 44.09% to 82.2% during the first five learning cycles, and the performance may finally increase up to 90.03% in ten learning cycles. Fig. 8 depicts the user interface and a snapshot of recognition results of the prototype system.

TABLE IV
BY APPLYING 300 COMMONLY USED CHARACTERS WRITTEN WITH NO CONSTRAINTS BY FIVE STUDENTS, THE PROPOSED ADAPTIVE SYSTEM SHOWS SIGNIFICANT IMPROVEMENT ON THE RECOGNITION ACCURACY DURING THE TEN LEARNING CYCLES

Trial	user#1	user#2	user#3	user#4	user#5	avg.
1st	50.6%	33.6%	38.1%	52.3%	45.7%	44.0%
2nd	67.7%	69.0%	55.5%	56.6%	61.1%	62.0%
3rd	78.6%	80.0%	69.9%	71.5%	72.7%	74.5%
4th	84.3%	78.7%	69.3%	75.9%	86.0%	79.5%
5th	84.6%	87.6%	73.9%	79.9%	85.1%	82.2%
6th	81.9%	89.0%	76.2%	80.2%	84.0%	82.2%
7th	86.5%	89.6%	79.9%	78.5%	84.6%	83.8%
8th	89.5%	90.3%	79.5%	86.9%	89.3%	87.1%
9th	90.5%	90.6%	81.2%	87.7%	89.3%	87.9%
10th	93.6%	91.4%	84.6%	90.5%	90.1%	90.0%

IV. CONCLUDING REMARKS

In this paper, a neural network multilinguistic handwritten character recognition system is proposed and implemented on a Pentium-100-based personal computer. This system performs classification, character recognition, and unconstrained freehand writing recognition. The SPDNN, which is a Bayesian decision-based neural network, was applied to implement the major modules of this system. This modular neural network deploys one subnet to take care one object (character), and therefore, it is able to approximate the decision region of each class locally and precisely. This locality property is attractive especially for personal freehand writing or signature identification applications. A personal adaptation for the character recognition module is proposed and implemented to improve the recognition performance on unconstrained freehand writing. On the other hand, due to the enormous number of variations involved, handwriting recognition applications still require more work before they can reach comparable performance by a human. Therefore, document analysis and recognition become an interesting and fascinating research topic in the field of intelligent information processing.

ACKNOWLEDGMENT

The authors acknowledge Prof. S.-Y. Kung and Dr. S.-H. Lin for their helpful suggestions regarding the probabilistic DBNN and statistical pattern recognition methods.

REFERENCES

[1] H. S. Baird, "Feature identification for hybrid structural/statistical pattern classification," *Comput. Vision, Graphics, Image Process.*, vol. 42, pp. 318–333, 1988.
[2] R. M. Bozinovic and S. N. Srihari, "Off-line cursive word recognition," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 11, pp. 68–83, Jan. 1989.
[3] D. J. Burr, "Designing a handwriting reader," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-5, pp. 554–559, 1983.
[4] F. H. Cheng and W. H. Hsu, "Research on Chinese OCR in Taiwan," *Int. J. Pattern Recogn. Artif. Intell.*, vol. 5, nos. 1 and 2, pp. 139–164, 1991.
[5] C.-C. Chiang, T. Cheng, and S.-S. Yu, "An iterative rule-based character segmentation method for Chinese documents," in *Proc. Int. Conf. Chinese Comput.*, Singapore, June 4–7, 1996.
[6] H. C. Fu and K. P. Chiang, "Recognition of handwritten Chinese characters by multi-stage neural network classifiers," in *Proc. 1995 IEEE Int. Conf. Neural Networks*, Perth, Australia.
[7] H. C. Fu, S. C. Chuang, Y. Y. Xu, W. H. Su, and K. T. Sun, "A personal adaptive module for unconstrained handwritten Chinese characters recognition," in *Proc. Int. Symp. Multi-Technol. Inform. Process.*, Hsinchu, Taiwan, R.O.C, Dec. 1996.
[8] J. J. Hull, "A database for handwritten text recognition research," in *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 16, pp. 550–554, June 1994.
[9] F. Kimura, K. Takashina, S. Tsuruoka, and Y. Miyake, "Modified quadratic discrimininant functions and application to Chinese character recognition," in *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-9, pp. 149–153, Jan. 1987.
[10] S. Y. Kung and J. S. Taur, "Decision-based hierarchical neural networks with signal/image classification applications," *IEEE Trans. Neural Networks*, vol. 6, pp. 170–181, Jan. 1995.
[11] Y. L. Cun *et al.*, "Handwritten zip code recognition with multi-layer networks," in *Proc. 10th Int. Conf. Pattern Recogn.*, 1990, pp. 35–40.
[12] T.-F. Li and S.-S. Yu, "Hand-printed Chinese character recognition using the probability distribution feature," *Int. J. Pattern Recogn. Artif. Intell.*, vol. 8, no. 5, pp. 1241–1258, 1994.
[13] S.-H. Lin, S. Y. Kung, and L. J. Lin, "Face recognition/detection by probabilistic decision-based neural networks," in *IEEE Trans. Neural Networks* (Special Issue on Artificial Neural Network and Pattern Recognition), vol. 8, pp. 114–132, 1997.
[14] S. Mori, C. Y. Suen, and K. Yamamoto, "Historical review of OCR research and development," *Proc. IEEE*, vol. 80, pp. 1029–1058, July 1992.
[15] G. Nagy, "Optical character recognition," in *Handbook of Statistics*, P. R. Krishnaiah and L. N. Kanal, Eds. Amsterdam, the Netherlands: North Holland, 1982, pp. 621–649.
[16] G. Nagy, "Chinese character recognition: A twenty-five year retrospective," in *Proc. 12th Int. Conf. Pattern Recogn.*, 1988, pp. 163–167.
[17] C. Y. Suen, M. Berthod, and S. Mori, "Automatic recognition of hand-printed character—The state of the art," *Proc. IEEE*, vol. 68, pp. 469–487, Apr. 1980.
[18] J.-W. Tai, "Some research achievements on Chinese character recognition in China," *Int. J. Pattern Recogn. Artif. Intell.*, vol. 5, nos. 1 and 2, pp. 199–206, 1991.
[19] L. T. Tu *et al.*, "Recognition of hand-printed Chinese characters by feature matching," in *Proc. 1991 First Nat. Workshop Character Recogn.*, Taipei, Taiwan, R.O.C, 1991, pp. 166–175.
[20] Y. Xia, "Research report on interactive self-learning system of handwritten Chinese characters," Dept. Comput. Sci., Tsing-Hua Univ., Hsinchu, Taiwan, R.O.C., Nov. 1989.

Hsin-Chia Fu (M'78) received the B.S. degree from National Chiao-Tung University, Taipei, Taiwan, R.O.C., in electrical and communication engineering in 1972 and the M.S. and Ph.D. degrees from New Mexico State University, Las Cruces, both in electrical and computer engineering in 1975 and 1981, respectively.

From 1981 to 1983, he was a Member of the Technical Staff at Bell Laboratories, Murray Hill, NJ. Since 1983, he has been on the Faculty of the Department of Computer Science and Information Engineering, National Chiao-Tung University. From 1987 to 1988, he served as the Director of the Department of Information Management, Research Development and Evaluation Commission of the Executive Yuan, Taiwan. From 1988 to 1989, he was a visiting scholar at Princeton University, Princeton, NJ. From 1989 to 1991, he served as Chairman of the Department of Computer Science and Information Engineering. From September to December of 1994, he was a Visiting Scientist at Fraunhofer Institute for Production Systems and Design Technology (IPK), Berlin, Germany. His research interests include digital signal/image processing, VLSI array processors, and neural networks.

Dr. Fu was the corecipient of the 1992 and 1993 Long-Term Best Thesis Award, with K. T. Sun and C. C. Chiang, and the recipient of the 1996 Xerox OA Paper Award. He has served as a founding member, Program Co-Chair (1993), and General Co-Chair (1995) of the International Symposium on Artificial Neural Networks. He is presently serving on the Technical Committee on Neural Networks for Signal Processing of the IEEE Signal Processing Society. He has authored more than 100 technical papers and two textbooks, entitled *PC/XT BIOS Analysis* (Taiwan: Sun-Kung) and *Introduction to Neural Networks* (New York: Third Wave). He is a member of the IEEE Signal Processing and Computer Societies, Phi Tau Phi, and the Eta Kappa Nu Electrical Engineering Honor Society.

Yeong Yuh Xu was born in Hsinchu, Taiwan, R.O.C., in 1973. He received the B.S. degree in electrical engineering in 1995 from National Sun Yat-Sen University, Kaohsiung, Taiwan, and the M.S. degree in computer science and information engineering in 1997 from National Chiao-Tung University, Hsinchu, Taiwan. He is now a Ph.D. candidate in the Computer Science and Information Engineering Department at the National Chiao-Tung University, Taipei, Taiwan.

His research interests include pattern recognition, neural networks, and content-based image/video retrival.