

-

Research on the Rejection Capabilities of the Signature-pressure-based Individual Recognition System for Counterfeit Signatures Using Optimized Neuro-template with Gaussian Function

Lina Mi

A dissertation submitted to
Kochi University of Technology
in partial fulfillment of the requirements
for the degree of

Doctor of Philosophy

Department of Engineering
Graduate School of Engineering
Kochi University of Technology
Kochi, Japan

March 2007

Abstract

In recent years, internet business has been developing rapidly along with the wide application of Internet and the security of users' information has become more and more important point for the development of internet business. Therefore developing an individual recognition system especially fit for internet application is becoming pressing task. Recently, biometrics information, such as finger print, iris, voice, face and signature, has been increasingly adopted in personal identification because of being unique and having high resistance to forgery.

In this research, a novel individual recognition system, in which biometrics information of signature pressure is exclusively employed to present personal feature, is developed for the application of internet business. The execution of the system includes two procedures, registration and recognition. First the user give three register signatures to register on the system (registration), after registration, the user can log on the system by giving one test signature (recognition) at anytime. In both procedures, the signature pressure data will be preprocessed firstly and then the data obtained from preprocessing of source pressure data are used either for registration or for recognition. Therefore two parts, preprocessing and neural network classifier (NN classifier), are included in the structure of the system.

In the preprocessing, the detected signature pressure data is firstly normalized, then equally dividing the normalized data into 300 sections and average value of each section is calculated as element of relay data. Second, validity check is executed on three relay data of register signatures. Last, the probability distributions of register relay data and inhibit relay data, which are artificially made by system, are analyzed and 50 elements are extracted as slab value from each relay data, which are used for NN learning or input to NN. In the preprocessing, the scale of source data is greatly reduced and personal feature of signature pressure are also extracted.

The neural network classifier of system is mainly studied in this paper and the uniqueness works in the research of this paper mainly include the following points.

- 1) Neuro-template Matching Method is introduced into the NN classifier of the system. According to this method, each registrant is assigned with a three-layer feed-forward neural network with uniform structure of $50 \times 35 \times 2$ and the NN classifier of the system is composed with the neuro-templates of all registrants. In case of registration, after learning with preprocessed source data of register signatures as

samples a new neuro-template is constructed for new registrant and then adopted as part of NN classifier by the system. In case of recognition, preprocessed pressure data of the test signature is matched with each existing neuro-template, and then outputs of all neuro-template are evaluated to decide the identity of signer. The performance of our system shows that Neuro-template Matching Method successfully simplified the registration procedure and removed the limitation on the number of registrants of the system. Furthermore, according to the study on mutual influence among neuro-templates, relearning of existent neuro-template caused by recruitment of new neuro-template is individual-dependent, and helpful for rejection capabilities of relearned neuro-templates for counterfeit signatures.

- 2) Gaussian ridge function is proposed as activation function of neurons in hidden layer and output layer of neuro-template to improve rejection capabilities of system for counterfeit signatures. Though the developed signature-pressure-based individual recognition system is effective in recognizing the authentic signatures, it suffers from poor rejection capabilities for counterfeit signatures. In order to improve rejection capabilities of the system on premise of ensuring the recognition capabilities satisfied, a kind of Gaussian function is proposed as activation function of neurons in hidden layer and output layer of each neuro-template, instead of originally employed sigmoid function. Different traditional Gaussian function of radial basis function (RBF), proposed Gaussian function is a ridge-like and semi-localized function, the corresponding sensitive field of function stretches out infinitely along the ridge and is restricted by the width parameter on the orthogonal direction of the ridge, and its sensitive field is controlled by the width parameter. The neuro-template with Gaussian ridge function presents the distribution of patterns of all known categories instead of partitioning the feature space as sigmoid function does. Hence the neuro-template with proposed Gaussian ridge function has more potential to improve rejection capabilities of the system with ensuring recognition capabilities of the system. The experiment results show that the employment of proposed Gaussian ridge function effectively improved the rejection capabilities of the system comparing with original system based on sigmoid function, at same time, however, it also led to slight decrease of the recognition capabilities of the system.
- 3) Width parameter σ of Gaussian ridge function is customized for each neuro-template using improved back propagation method (BP). In the pilot study. The width parameter of proposed Gaussian ridge function is selected manually and kept constant once proper value is decided. Though the neuro-template based on Gaussian ridge function with fixed σ is effective on improving the rejection capabilities of system for the counterfeit signatures, the improvement is not

significant enough. Moreover the recognition capabilities of system for genuine signatures become deteriorated a little by the employment of Gaussian ridge function with fixed sigma. To further improving the rejection performance of the system at same time ensuring recognition performance satisfied, the width parameter of Gaussian ridge function is optimized for each neuro-template. The experiment results showed that the customization of width parameter sigma not only effectively furthered the improvement of rejection capabilities of the system, but also partially recovered the deteriorated recognition capabilities of the system resulted by employment of Gaussian ridge function.

- 4) The uniformed characters are proposed as register characters of the system. Though the performance of our system has been improved by employment of proposed Gaussian ridge function with optimal width parameter sigma, the high discrepancy in the recognition capabilities of the system for different registrants has been seen and that indicates the insufficiency of performance stability of the system. To address that problem, uniformed characters is proposed as register characters of the system, for which personal signature is traditionally employed, in the last part of this research. To evaluate the feasibility and effectiveness of proposed method, four groups of characters are selected as uniformed register characters respectively and corresponding recognition capabilities of the system are tested and compared with that of original signature-based system. The experiment results show that the uniformity of register characters seems to be effective in reducing the fluctuation of the recognition capabilities of the system for different registrants and improving the performance robustness of the system without sacrificing the recognition performance itself. Furthermore the discrepancy in the performance stability of the systems with different uniformed register characters shows that the robustness of performance of the system is influenced by the complexity of the register characters, it suggests that uniformed register characters should be carefully selected to get better performance stability of the system and better performance itself. Last, from experiment results it also can be seen that the performance robustness is almost not affected by the employment of Gaussian ridge function and optimization of parameter sigma, which are proposed to improve the rejection capabilities of the system for counterfeit signatures.

Contents

Chapter 1	Introduction.....	1
Chapter 2	Structure of the System and Preprocessing	77
2.1	Structure of the hardware part of the sytem	8
2.2	Structure of the software part of the system	11
2.2.1	Main interface of the system	12
2.2.2	Registration section of the system	13
2.2.3	Recognition section of the system	15
2.3	Preprocessing	16
2.3.1	Inputting the signatures	17
2.3.2	Making relay data	17
2.3.3	Validity check	20
2.3.4	Making artificial data	23
2.3.5	Making information of valid points group	25
2.3.6	Making NN data	27

Chapter 3 Neuro-template classifier of the system28

3.1	Introduction	28
3.2	Overview of neural network.....	31
3.3	Original classifier of the sytem.....	36
3.3.1	Neuro-template matching method.....	37
3.3.2	Activation function of original classifier	40
3.4.	Proposed classifier of the system	44
3.4.1	Proposal of Gaussian activation function.....	44
3.4.2	Customization of parameter sigma for each neuro-template	49
3.5	Neuro-template learning.....	50
3.5.1	Error function.....	51
3.5.2.	Initialization of corresponding parameters	52
3.5.3	Improve back propagation (BP) algorithm	53
3.5.4	Customization of width parameter σ	60
3.6	Experiments.....	62
3.6.1	Experiment database and learning condition	63
3.6.2	Performance measurement and decision criteria	64
3.6.3	Performance of the systems	65
3.6.4	Influence of width parameter σ	69
3.6.5	Reciprocal influence of neuro-templates	73

Chapter 4 Analysis on robustness of the system and proposal of uniformed register characters 76

4.1	Introduction.....	76
4.2	Insufficient performance robustness of signature-based system.....	78
4.3	Proposal of uniformed register characters	79
4.4	Experiments	81
Chapter 5 Conclusions.....		90
REFERENCES.....		94
ACKNOWLEDGEMENT.....		99
PUBLICATIONS		100

Chaper 1

INTRODUCTION

In recent years, internet business has been becoming prosperous, along with the rapid development of network technology and wide application of internet, more and more business trades have been processing on internet, to guarantee the clients's rights and interests and sustainable development of internet business, it appears more and more important to protect internet security of clients' private information from illegale infringe. There are also other cases, such as electronic finance, remote log-on, appliance and data securely accessing, that make reliable internet individual authentication an essential requirement and give a new challenge to personal identity verification or identification techniques exsitent so far. The traditional security password and PIN are becoming unable to meet the request of internet identity authentication because of the ease with which they are shared, lost, stolen, or forgotten. On the other hand, biometric

features, such as fingerprint, palm print, iris, voice, face and signature, have been increasingly adopted in the individual identity system due to their many advantages including being unique and having high resistance to forgery in recent years. Though many biometric features such as fingerprint, palm print, iris and face, have advantage of high accuracy of verification, intricate and expensive equipments are necessary and the processings of these biometric features are very complicated, furthermore it is easy to cause the psychological resistance of user when employing these biometric features. All of these factors make these biometric features not fit for internet authentication. Different from other biometric features, signature is a time-honored means for identity confirmation and could be willingly accepted by people. Furthermore, signature can be easily detected with simple instrument. In view of internet utilization, we limit our concerns to signature verification.

Many efforts have been made on signature verification domain and numerous ideas, methods have been proposed in this area [1][2]. Signature verification systems available so far can be roughly grouped into two classes, one is based on static image of signature [3][4][5][6], in the other class, however, dynamic features involved in signature process is primarily concerned [7][8][9][10][11]. There are also some writer identification systems based on the combination of static characters and dynamics one [12][13][14]. Though many efforts have been devoted initially, static signature verification has always been considered more difficult by researchers because of complex signal preprocessing and feature extraction for static image [2], which often lead to obtained results not as good as dynamic approaches. Furthermore information contained in dynamic signal of signatures are invisible to users, this quality makes signature forgery virtually difficult. Therefore dynamic-feature-based signature verification system increasingly attracts the interest of researchers and many input devices have been specially developed for acquiring dynamic data from signature process[9][10].

In this research, we focus on dynamic signature identification. Extensive range of dynamic information like position, velocity, acceleration, and writing pressure has been reportedly applied, separately [9] [10] or collaboratively [7][11][14], into signature verification task. In our system, however, signature pressure is exclusively considered as individual information for individual identification. Automatic signature verification systems with pressure as the only feature information of personal identity have been

developed initially in literatures [10] [8] in which traditional algorithms including Dynamic Programming matching (DP matching), Discrete Fourier Analysis (DFT) and simple feature matching were used for data preprocessing or comparison between test signature and reference one. Different from two previous systems, in our signature-pressure-based identification system neural network technology is employed.

It has been proved that neural network is a kind of effective tools on realizing nonlinear mapping according to a set of given input-output training samples, and the successful applications of neural networks on many areas, such as pattern recognition, nonlinear adaptive system, communication and signal processing, have been reported. In the field of signature verification, there are also many reported studies of utilizing neural networks to classifying the signatures in recent years. But the neural networks used in these studies are very complicated [2]. In our system, neural network combined with multi-templates matching scheme is employed to classify the preprocessed signature pressure. The signature-pressure-based individual recognition system developed in previous study is very effective in recognizing the authentic signatures whose category have been learned, However it suffers poor rejection capabilities for forged signatures, whose category, in fact, is unknown for the system.

To improve the rejection capabilities of the system for counterfeit signatures on premise of guaranteeing the recognition capabilities for authentic signatures, a new signature-pressure-based individual recognition system using neuro-template are proposed in this research. In each neuro-template, a Gaussian function is employed as the activation function of each unit in hidden layer and output layer instead of traditionally employed Sigmoid function. Different with conventional Gaussian functions used as one of radial basis functions, the proposed Gaussian activation function is a ridge-like function. The characteristics of Gaussian ridge function contribute to effective improvement of rejection capabilities of the proposed system for counterfeit signatures.

The operation of our system includes two procedures, registration and recognition. First the user continuously give three signatures to register on the system (registration), and then log on the system by giving one test signature (recognition). In both procedures, the signatures (register signatures or test signature) are preprocessed first, and slab value representing feature of signature are obtained. In the registration, the

obtained slab value is used to train a new neuro-template, which will be described later, for the new registrant, then the trained new neuro-template is accepted as a part of neuro-template classifier and the procedure of new registration is completed. In the procedure of recognition, the slab value obtained from the preprocessed test signature is matched with all existing neuro-templates each by each, then the outputs of all neuro-templates are evaluated according the predetermined criteria and the identity of signer is recognized based on evaluation results. Therefore, the system is basically constructed by two parts, preprocessing and neuro-template classifier. The scheme of our system is expressed as Figure 1.1.

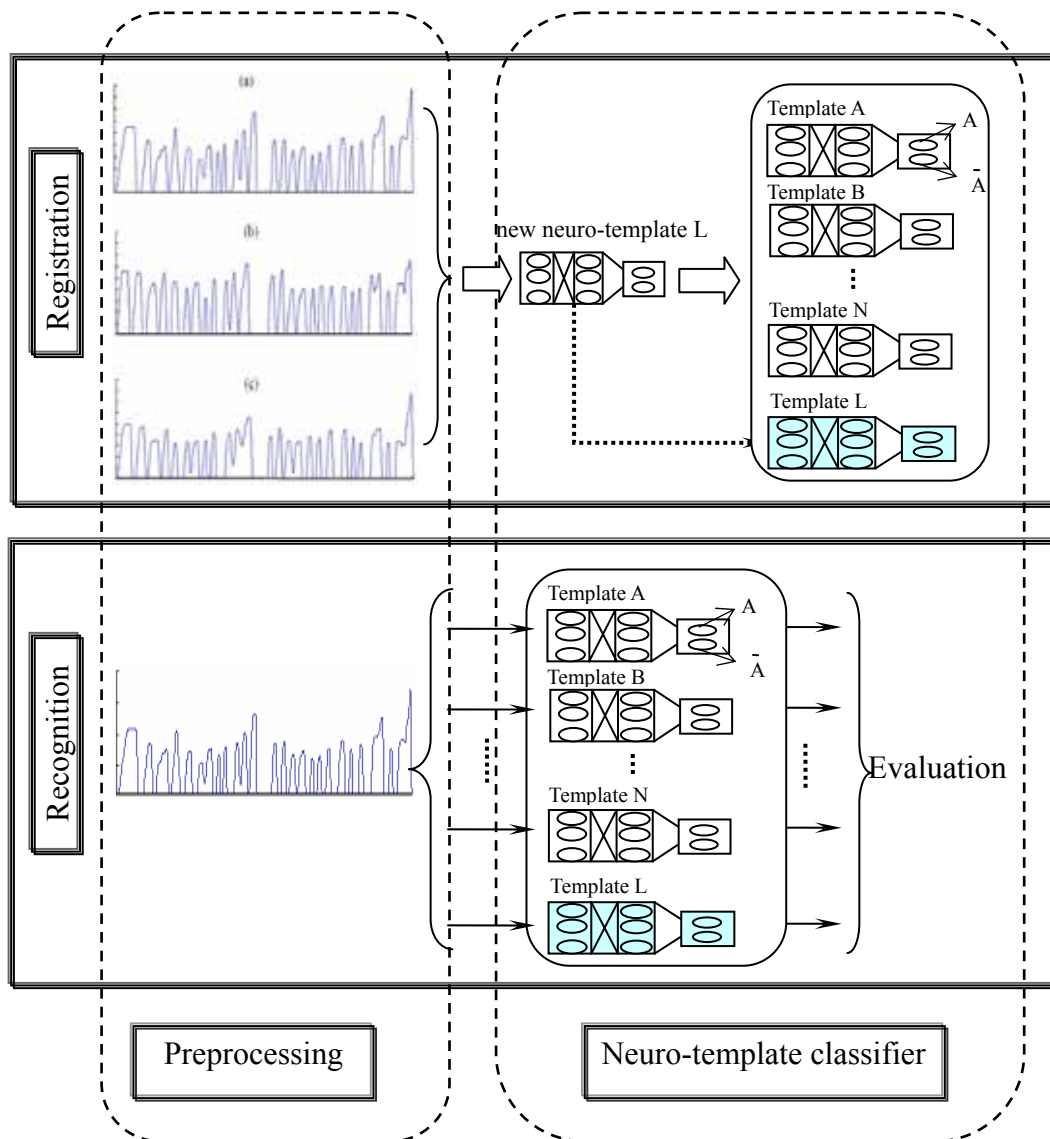


Figure 1.1 the Scheme of Individual Recognition System

The preprocessing of the detected signature pressure data includes deleting the redundant data, separating the stroke data and the interval data between neighbor strokes, normalizing the data of strokes, reducing the scale of data, the validity checking and extracting the feature vector. The neuro-template classifier is composed of the neuro-templates of all registrants, neuro-template is the integration of neural networks (NN) with template matching scheme, it is a three-layer feed-forward neural network (FNN) with uniform structure of $50 \times 35 \times 2$, namely 50 units in input layer of neuro-template, 35 unit in hidden layer and 2 units in output layer. Each neuro-template is in charge of identity authentication of corresponding registrant, namely deciding if the signer is the registrant corresponding to this neuro-template or not, with cooperation of all neuro-templates, the identity of signer is able to be recognized. With proposed Gaussian ridge function as activation function of units in hidden layer and output layer in each neuro-template, the rejection capabilities of each neuro-template for counterfeit signatures has a favorite improvement. Furthermore with customization of parameter sigma of Gaussian ridge function, the rejection capabilities are improved greatly and the recognition capabilities of neuro-template for authentic signatures, which deteriorated a little with employment of Gaussian ridge function, are also recovered.

In order to improve the performance robustness of our system for different registrants, the influence of register characters on the performance stability of the system is studied and the strategy of uniformed register characters is proposed in the second part of this research instead of originally employed signature. With uniformed register characters, disturbance of the difference of register signature characters is removed and the neuro-template is able to focus on individual feature of writing habit in terms of pressure and the performance stability of the system is improved.

To evaluate the effectiveness of the proposed Gaussian ridge function with customized parameter sigma and the uniformed register characters, experiments are designed and conducted respectively and corresponding experiment results are analyzed. Last the favorite conclusions are obtained according to experiments results.

This paper is arranged as follows. First the whole structure of individual recognition system and the preprocessing are depicted in chapter 2. Chapter 3 described the neuro-template classifier of the system, the neuro-template matching algorithm is also presented. Then the problem of poor rejection capabilities for expert counterfeit

signatures of the system is analyzed and the new recognition system based on Gaussian ridge function is proposed. To further the improvement of the rejection capabilities and guarantee recognition capabilities for authentic signatures, parameter sigma of Gaussian ridge function is customized for neuro-template. The experiment results demonstrate the effectiveness of the proposed system based on customized neuro-template with Gaussian ridge function. In chapter 4, the uniformed-register-characters-based individual recognition system is proposed to improve the performance robustness of system for different registrants and corresponding experiment results are analyzed. Last, the paper ends with conclusion about the developed individual identification system in chapter 6.

Chapter 2

STRUCTURE OF THE SYSTEM AND PREPROCESSING

Signature verification is a natural and intuitive method for identity verification, it has a long history of application, such as bank checks, document subscribing, and is already accepted as a common method of identity verification by people. Though there will be slight variations in a person's handwritten signature, the consistency formed by nature motion and practice over time creates a recognizable pattern that makes the handwritten signature a natural for biometric identification. With the development of computer technology, automatic signature individual identification has been studied for a long time and many systems have been developed [1][2]. The information used in automatic signature verification can be grouped into two classes, static information and dynamic information. The static information based signature identification, which is

traditionally signature verification, focus on what the handwritten looks like, namely it is based on static picture of handwritten. The dynamic information based signature identification takes into account how the signature is made and uses the behavioral biometrics of a hand written signature, including position, speed, stroke, pen pressure and timing information during the act of signing, to confirm the identity of a computer user. Distinct from the static information, which is easy duplicated by pasted bitmap, copy machine or an expert forger, it is virtually difficult to duplicate the dynamic information in the procedure of signing. In previous of our research, a dynamic signature individual recognition system has been developed especially for application of internet business. Different other dynamic signature verification system, in which several dynamic information are employed cooperatively, pen pressure of signature is exclusively used as personal feature information. Because only pen pressure of signature is used, an electronic pen with sensitive pressure sensor is used as the input device for our system. Our system can be divided into two parts, hardware part and software part, the structure of each part will be described in the following sections.

2.1 Structure of the hardware part of the system

The hardware of our system is composed of PC, electronic pen and data collection



(a) Electronic pen



(b) Data collection box



(c) Whole input subsystem

Figure 2.1 Input subsystem of individual identification system

box, which is not necessary in USB version of our system. The whole hardware of system is shown in Figure 2.1.

It can be seen from Figure 2.1 that our input system is comparatively simple and convenient. Pressure data are detected by the electronic pen, and then loaded to PC via the data collection box. In the USB mode, the detected pressure data are directly transported from the electronic pen to the PC.

The selected electronic pen has superior attributes of high sensitivity (0.1g resolution for pressure) and rapid speed (4ms time resolution), which ensured the extraction of the pressure data with high quality. At same time, the shape of electronic pen is very similar to an ordinary pen, it facilitates easy writing action for users. When a user registers or logs on the system, signatures are written on normal paper with this electronic pen.

Figure 2.2 shows the text of one of three register signatures and corresponding three pen pressure curves which is detected by the electronic pen from a legitimate registrant.

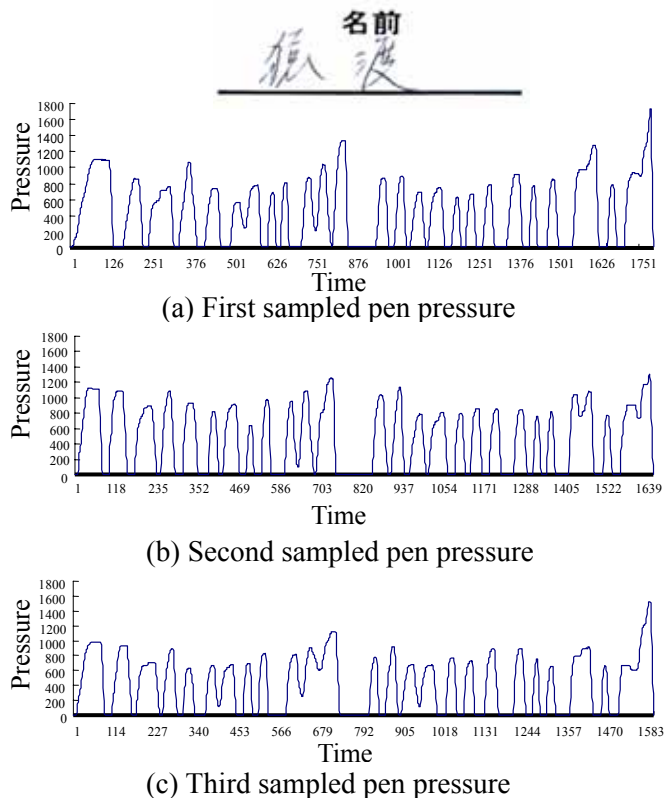
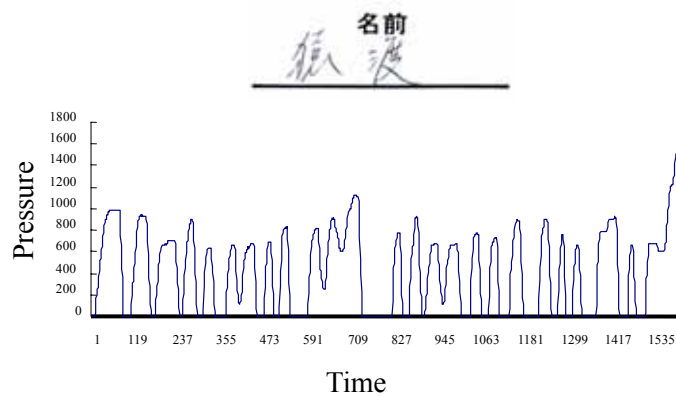


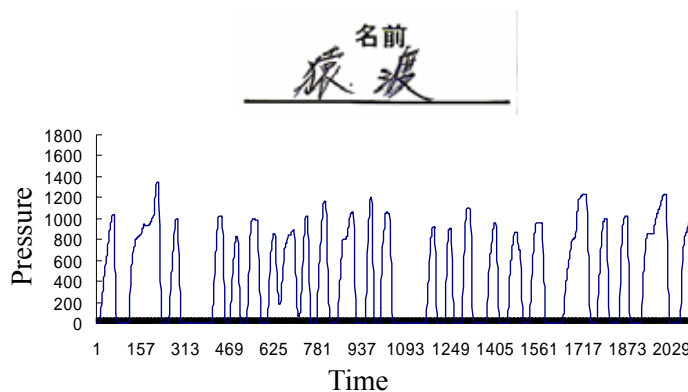
Figure 2.2 the text of one genuine signature and three pen pressure curves from genuine signatures

From Figure 2.2, it can be seen that though some variations shows up, consistent characters, which forms an unique individual pattern in the signature pressure, can be easily noticed in these curves. It indicates that with proper preprocessing and recognition algorithm, the individual characters in the signature pressure can be exclusively utilized to recognize the identity of the signer.

To demonstrate the individual feature in the signature pressure is larruping and hard to forge, the pen pressure curve and corresponding text of an authentic signature are compared with that of an expert counterfeit signature in Figure 2.3. The counterfeit signatures are the signatures written by intruder who wants to log on system illegally. Here, the counterfeit signatures are obtained from tracing the provided authentic



(a) the text and pen pressure curve of an authentic signature



(b) the text and pen pressure curve of an expert counterfeit signature

Figure 2.3 the pen pressure curve and text of an authentic signature and that of an expert counterfeit signature

signature by experiment subjects, therefore the counterfeit signatures used in our system are expert counterfeit signatures.

From Figure 2.3, it can be seen that though the texts of the authentic signature and the expert counterfeit signature are very similar, the significant disparity in the individual features of two pen pressure curves can be easily perceived. It shows that the individual features in the pen pressure of signature are very hard to forge even for skillful forger, it attributes to invisibility of the pen pressure signal to any forger, and the advantage of being a behavioral dynamic biometrics of pen pressure of the signature also contributes to the high difficulty of being forged. The pen pressure data of signature detected by the input devices of the system is loaded into PC and then processed by the software part of the system.

2.2 Structure of the software part of the system

When our individual recognition system is activated, the initialization is firstly conducted on the system, such as initializing the various dynamic link libraries, loading the setting file, reading the information file of existing registrants and activating the connection of Data collection box with PC (serial communication version) or USB port (USB version). If the system is successfully initialized, the main interface of the system is shown up with various menus, shortcut function keys and information of existing registrants. If an unknown person wants to utilize our system, he (or she) has to register on the system with three register signatures first. After being a legitimate registrant, he (or she) can log on the system with one test signature at any time. Therefore the software part of the system is basically composed by two sections, registration section and recognition section, which are relatively independent each other. The main structure of our system is shown as Figure 2.4.

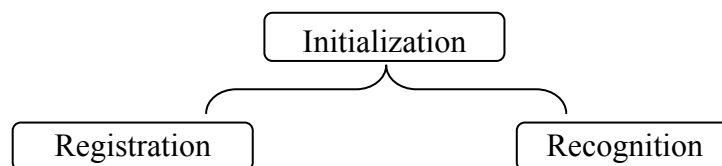


Figure 2.4 the main structure of the software part of the system

2.2.1 Main interface of the system

The main interface of our system is shown in Figure 2.5

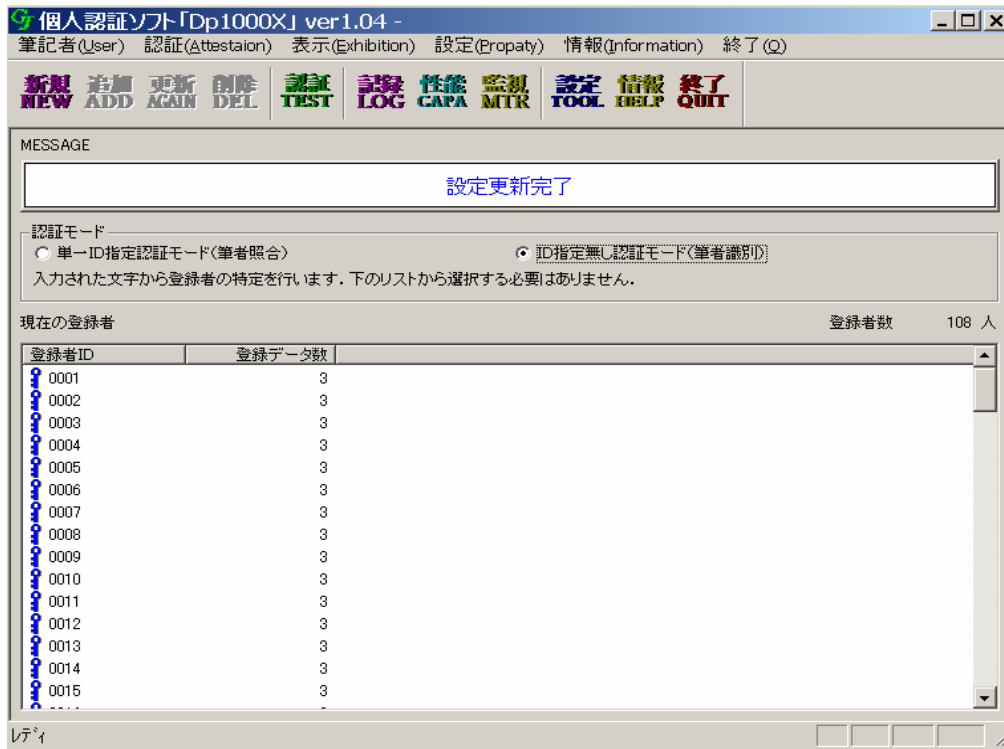


Figure 2.5 the main interface of the system

In the main interface, functions of shortcut keys “NEW” and “TEST” correspond to registration and recognition respectively, which are two elementary functions of our system. Besides previous two functions, others are auxiliary functions to the system. Shortcut key “ADD”, “AGAIN” and “DEL” are functions of adding additional register signatures for a certain registrant, renewing the register signatures and deleting the selected registrant from the system respectively. These functions are activated only when a certain registrant is selected. Shortcut key “LOG” keeps the record of all of the tests, including the time, source data file, test mode and test result. “CAPA” is used to check the capability of the electronic pen, including sampling speed, early sensitivity and operation state by sampling the pen pressure of one signature. “MTR” has the function of monitoring the value of pen pressure, the value of measured pen pressure

and measure frequency, and adjusting the early sensitivity of electronic pen. “TOOL” is used to set the properties of the system, such as the operation mode of the system which includes normal mode and maintenance mode, the data volume in the waiting time before the end of signature is detected. It should be noticed that many performance can not be set in the normal mode. The last two shortcut keys “HELP” and “QUIT” are used for providing the version information of the system and quitting the system respectively. The functions of all shortcut keys can be found in the corresponding menus.

It should be noticed that in the main interface, there is a section for selecting the mode of “TEXT”, one option is the mode of test with providing the ID information of single registrant (pen pressure authentication), the other option is the mode of test without providing the ID information of any registrant (pen pressure recognition). In the authentication mode, the system decides if the user logging on the system is the registrant with provided ID or not, that means in order to authenticate the identity of the user to test, it is enough to compare the signature characters of the user to test only with that of the indicated registrant and check the agreement of two characters, while other existing registrants are not necessary to be considered. In the recognition mode, however, the system have to compare the signature characters of the user to test with that of all of existing registrants and then decide the identity of the user to test based on the degree of similarity of each group of characters because of no information of the user is available in this mode. It can be seen that procedure of authentication can be considered as a part of recognition, therefore the recognition mode is generally regarded more complicated and difficult than the authentication mode. In this research, all the discussion and experiment results are based on the recognition mode. In order to become being able to be recognized by the system with one test signature, the user must register on the system with three register signatures.

2.2.2 Registration section of the system

During the registration, the sampled pen pressure data of three register signatures are

firstly preprocessed and then the feature vectors are extracted. The system learns the new registrant's individual feature from the obtained feature vectors and saves the feature information as reference data of the new registrant in the database for recognition. The flowchart of the registration section is shown in Figure 2.6.

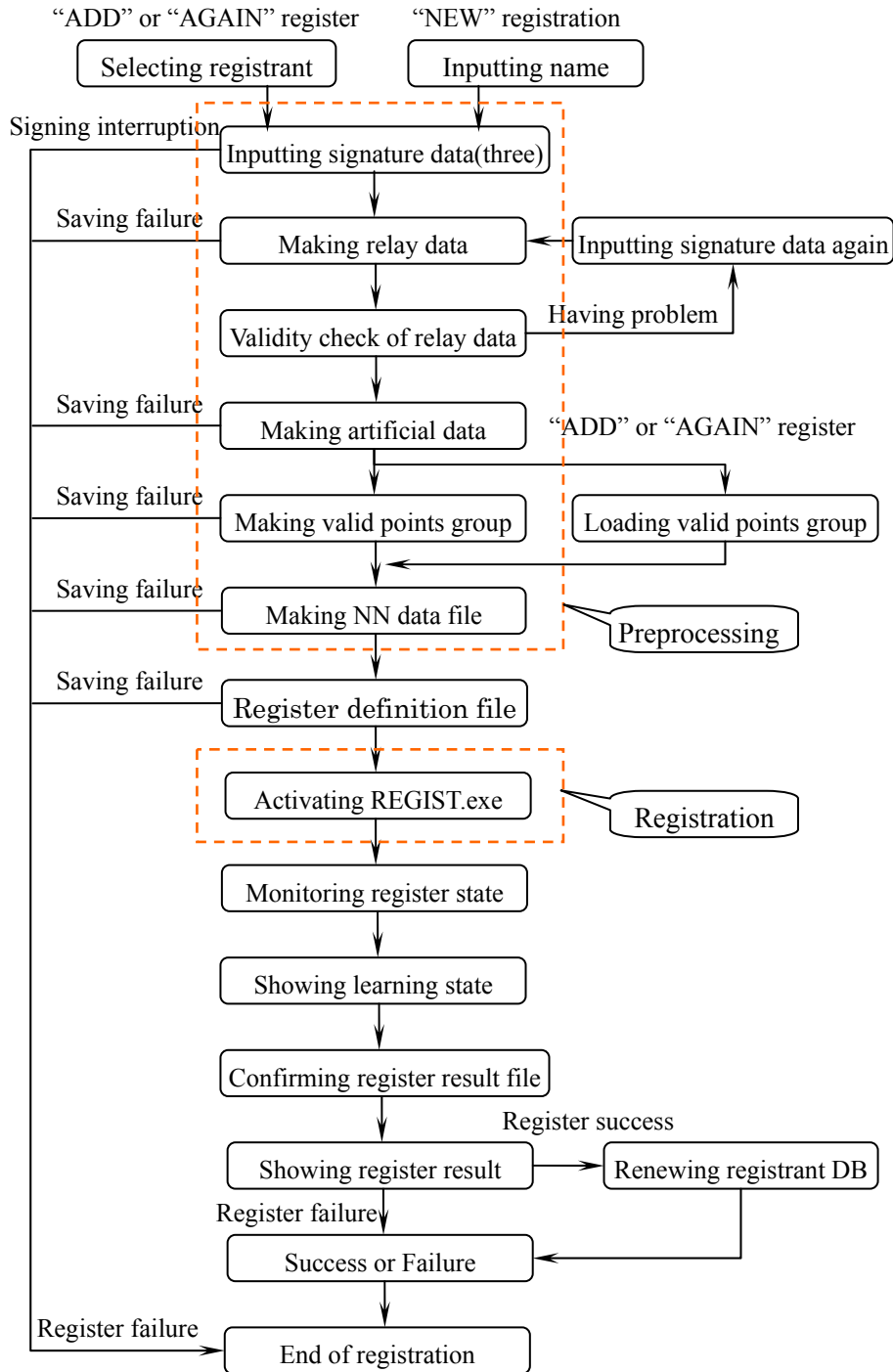


Figure 2.6 Flowchart of registration section

From Figure 2.6, it can be seen that besides auxiliary functions, such as saving file, showing result and monitoring progress state, preprocessing and registration are two main parts of registration section.

2.2.3 Recognition section of the system

In the recognition, the pen pressure data of one test signature sampled from the user are firstly preprocessed and then matched with all reference data in the database to decide the identity of the user. Figure 2.7 shows the flowchart of the recognition section.

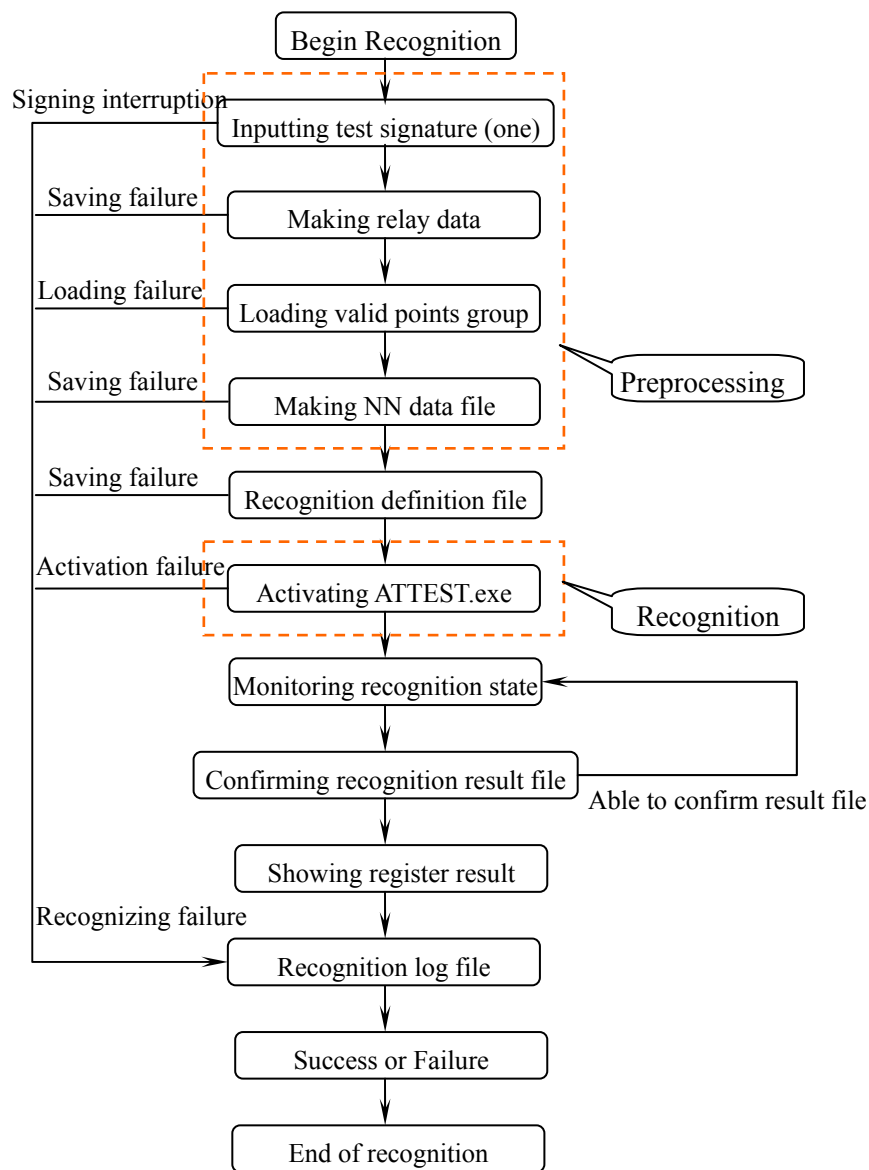


Figure 2.7 Flowchart of recognition section

Just like the registration section, besides of auxiliary parts, preprocessing and recognition are two main parts in the recognition section.

According Figure 2.6 and 2.7, it can be seen that preprocessing is important part of our system and it has crucial influence on the performance of the system.

2.3 Preprocessing

From the flowchart of the registration section and recognition section, it also can be seen that the preprocessing of registration section and that of recognition section are almost same, both of them include sampling signature, making relay data, obtaining valid points group (making or loading), making NN data. However it is obvious that the preprocessing part of recognition section is simpler and can be viewed as a part of that of registration section. Therefore the preprocessing of pen pressure data will be described with the preprocessing part of recognition section as example.

The flowchart of preprocessing is shown is Figure 2.8

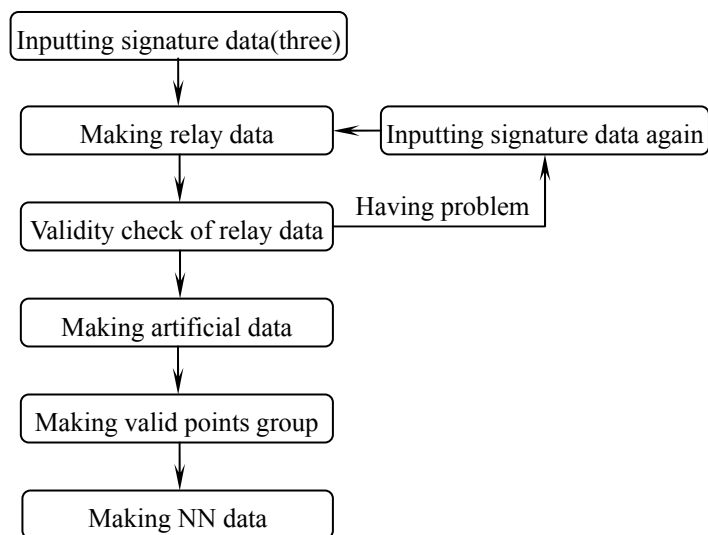


Figure 2.8 Flowchart of the preprocessing

2.3.1 Inputting the signatures

The procedure of sampling pen pressure data of signature is described as Figure 2.9

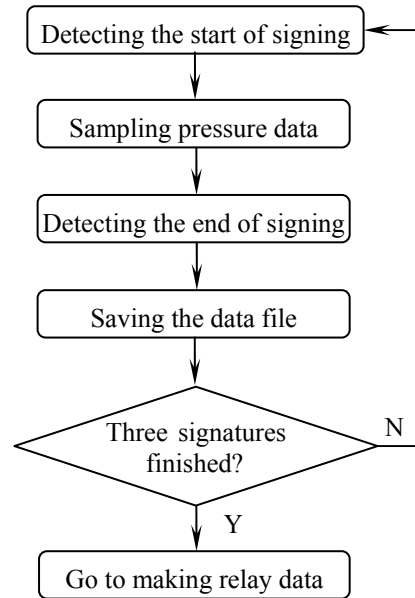


Figure 2.9 Sampling pen pressure data of signatures

In this procedure, the start and end of signing are automatically detected by the system. From the start of detecting signing, the system continuously collects the pressure value by the electronic pen and the dispersal of ten pressure data last collected is checked after each time of inputting a new pressure value, if the dispersal degree is over the predetermined threshold, the signing action is considered to start. During inputting the pressure data, the value of each collected pressure data is checked. If the input pressure value is continuously less than the threshold value for a certain waiting time, one time of signing is thought to end. All of the collected pressure data of a signature are saved as one source data file. After three register signatures are completed, the preprocessing on the collected source data file moves to next step, making relay data.

2.3.2 Making Relay data

In this stage of preprocessing, the collected source pressure data of signature are

firstly regularized, in which deleting the redundant data in head and tail of source data, separating the stroke data and the interval data located between neighbor strokes, and normalizing source data with moving average method are involved. Then the regularized source data are equally divided into 300 sections and the average of the data in each section is calculated, the resulted data are named relay data because that they are connection between source data and final form of preprocessed data. The corresponding procedure is demonstrated in Figure 2.10.

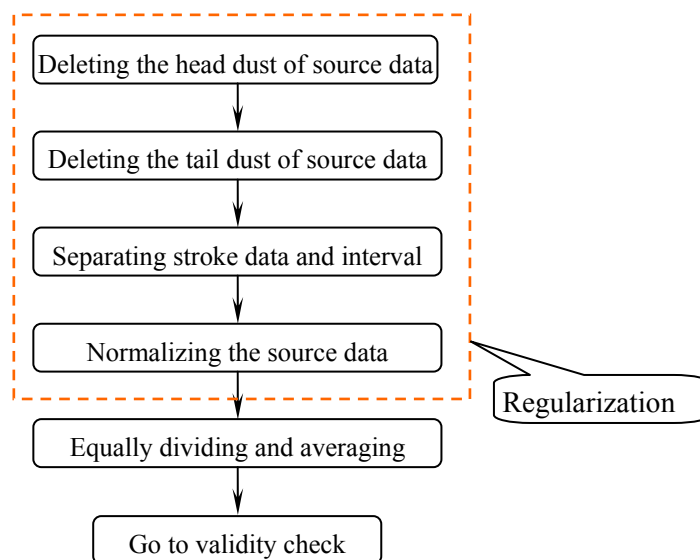


Figure 2.10 Making relay data

During collecting the source pressure data of signature, the unwanted data are inevitably included in the head and tail of the source data before the beginning and the end of one signature are detected. To eliminate the unwanted data, each value of source data is checked to fix the position of beginning and end of the source data, then the data located before the beginning and the end of source data are viewed as redundant data and removed.

After the redundant data in the head and tail of the source pressure data are deleted, the stroke data and the interval data between neighbor strokes are detected and separated. The information of each stroke and interval, including the position of the beginning point and the end point of each stroke and the size of each interval data, are

saved for later utilization.

Then the data of each stroke are consequently normalized and smoothed with moving average method aiming to remove the random noise data, which are included during collecting pressure data, and smooth the fluctuation of the source pressure data. Moving average method is one of simplest way to smooth fluctuating data. In this procedure a group of points in the data set to process with fix number is selected and the values of these points are added together, then dividing the sum of these points with the number of points to obtain the average value at the center of the group. Next, the point at the head end of the group of points is dropped, the point immediately following the selected group of points is added to the tail end of the group and the process is repeated and the points of the data set is processed each after each. How the moving average method worked is explained with an example. Supposing the data set to process has 30 points, a group of points with e_i ($i = 0,1,\Lambda 29$) as center and parameter $s_a = 4$ as span (the number of the points in the group is $2 * s_a + 1 = 9$) is selected, each point of the data set e_i is processed as following.

$$\begin{aligned}
 e'_0 &= e_0 \\
 e'_1 &= (e_0 + e_1 + e_2 + e_3 + e_4 + e_5) / 9 \\
 &\vdots \\
 e'_i &= \frac{\sum_{j=\max\{0, (i-4)\}}^{\min\{29, (i+4)\}} e_j}{2 \times 4 + 1} \quad \text{and } 1 \leq i \leq 28 \\
 &\vdots \\
 e'_{28} &= (e_{24} + e_{25} + e_{26} + e_{27} + e_{28} + e_{29}) / 9 \\
 e'_{29} &= e_{29}
 \end{aligned}$$

The previous procedure can also be explained as the following figure.

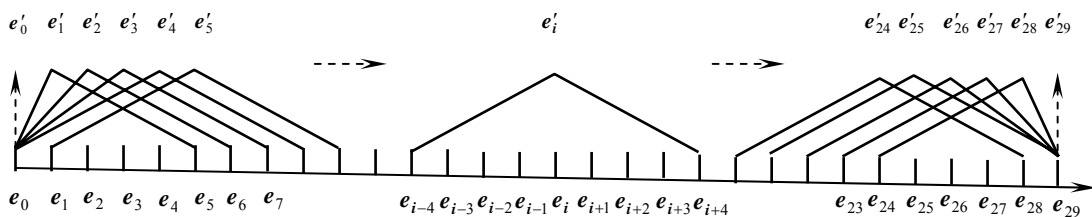


Figure 2.11 Moving average method

The common equation of the moving average method is shown as following

$$e'_i = \begin{cases} \frac{\sum_{j=\max[0, (i-s_a)]}^{\min[n_e-1, (i+s_a)]} e_j}{2 * s_a + 1} ; & 0 < i < n_e - 1 \\ e_i ; & i = 0 \text{ or } n_e - 1 \end{cases} \quad \text{--- (2.1)}$$

Where e_i, e_j are the i th, j th points in data set of one stroke respectively. e'_i is the i th point of data set obtained after being processed by moving average method. n_e is the number of points in one stroke. In our system, the span parameter of s_a is selected as 4 just as previous example.

After normalization, the regularization of source data is completed. Next, the regularized source data are approximately equally divided into 300 sections, and the average of the data in each section is calculated as Equation (2.2).

$$r_i = \sum x_j / l_i ; \quad \text{int}(j/(m/n))=i \quad \text{and} \quad i=0,1,\Lambda n-1, j \in (0,1,\Lambda m-1) \quad \text{--- (2.2)}$$

Where m is the size of regularized source data, n is the size of relay data ($n=300$), r_i is the i th member of relay data, x_j is the j th element of regularized source data. For certain i , if the x_j belongs to i th section of source data or not is decided by whether the rounding of $j/(m/n)$ equals to i , l_i is the number of elements x_j located in the i th section of regularized source data. The obtained average values of 300 sections are named relay data and the procedure of making relay data is finished.

2.3.3 Validity Check

The three relay data resulted from the register signatures by previous processing are denoted as register relay data RR^1, RR^2, RR^3 . The validity check is conducted on these three register relay data to guarantee the quality of collected source data of three register signatures. Because the source data of signatures collected during registration will be used as reference data representing the signature characters of the corresponding registrant, the invalid signature pressure data may lead to unreliable reference database, consequently result in failure of recognition or authentication and bad performance of

the system. Therefore, it is important to check the quality of the collected register source pressure data. If the pressure data of a register signature deviate from other two signatures too much, this signature is viewed as invalid and then one more signature pressure data will be detected from the registrant. In our adaptive enrollment procedures, two factors, statistical distance D and correlation coefficient R , are considered for validity check.

Statistical distance (Euclidean distance) D between $Data1$ (presented as $d1$) and $Data2$ (presented as $d2$) is calculated as following equation.

$$D = \sqrt{\sum_{s=1}^S (d1(s) - d2(s))^2} \quad \text{--- (2.3)}$$

Statistical distance D of $d1$ and $d2$ represents the Euclidean distance of two data in the high dimension space, the less D is, the closer two data set are. In our application, small D of two relay data means that the feature vectors of the corresponding source data are close in feature space and they can be classified into same category of signature style.

Correlation coefficient R between $d1$ and $d2$ is calculated as following.

$$R = \frac{\sum_{s=1}^S (d1(s) * d2(s)) - S * Ave(d1) * Ave(d2)}{\sqrt{\sum_{s=1}^S (d1(s))^2 - S * Ave(d1)^2} \sqrt{\sum_{s=1}^S (d2(s))^2 - S * Ave(d2)^2}} \quad \text{--- (2.4)}$$

Where $Ave(d1)$ and $Ave(d2)$ are average of $d1$ set and $d2$ set respectively, assuming that s elements are included in each data set. The value of R varies between -1 and 1. The nearer 1 R is, the more closely the two data correlate.

Comparing each pair of three pressure data sets (1-2, 1-3 and 2-3), three groups of $D(I)$ $R(I)$ ($I=1 \sim 3$) are obtained as shown in Figure 2.12. The deviation of each pair of register signatures is denoted as $Flag(I)$ ($I=1 \sim 3$) and decided by the Equation 2.5.

$$Flag(I) = \begin{cases} 1 & (R(I) > THR \text{ AND } D(I) < THD) \\ 0 & (R(I) < THR \text{ OR } D(I) > THD) \end{cases} \quad \text{--- (2.5)}$$

where THR and THD are predetermined threshold for correlation coefficient and statistical distance respectively.

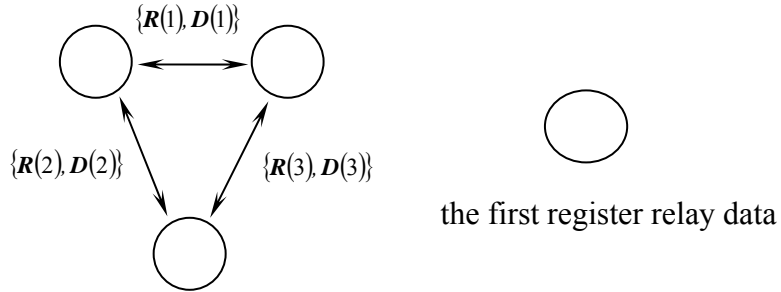


Figure 2.12 Correlation coefficient and statistical distance

Based on the state of three deviations, $Flag(1), Flag(2), Flag(3)$, the validity of three register signatures is determined according the Table 2.1.

Table 2.1 Validity check of register relay data

Flag(1)	Flag(2)	Flag(3)	Data(1)	Data(2)	Data(3)	Check
0	0	0	N	N	N	000
1	0	0	Y	Y	Y	100
0	1	0	Y	N	Y	010
0	0	1	N	Y	Y	001
1	1	0	N	Y	N	110
1	0	1	Y	N	N	101
0	1	1	N	N	Y	011
1	1	1	Y	Y	Y	111

Y: valid N: invalid

Only the result of check is 111, the collected three source data could be considered valid and the procedure of validity check is finished, otherwise the invalid signature will be abandoned as abnormal and registrant will be asked for more signatures. The validity check could effectively guarantee the quality of reference data and free users from having to re-sign many times.

2.3.4 Making artificial data

In our system, neural network combined with template matching method is employed as classifier in the later part of the registration and recognition. In the neuro-template classifier of each registrant, two output classes, objective class and non-objective class, are assigned in the output layer, namely two neurons are included in the output layer of each neuro-template. Objective class corresponds to the registrant to be verified or identified. Non-objective class is related to the signature feature of any person except for the target registrant, including all other legitimate registrants and any forger. Therefore during registration, two kinds of samples are needed for training the objective class and the non-objective one of the neuro-template respectively. The preprocessed authentic signatures detected from target registrant during registration procedure can be used as the sample of objective class and named enforce data. In case of the non-objective, in the original intention, forged signatures obtained by simulating the authentic signatures of the target registrant should be used as samples to training the non-objective class aiming to high rejection capability of the system for forged signatures, however, the forged signatures from the people other than target registrant are not practically available during registration of target registrant. Therefore the data used as samples of non-objective class, which are named inhibit data, have to be artificially made by the system based on the three authentic register signatures. The inhibit data are made as following.

- 1) Matching the counterparts of three regularized register source data based on least standard deviation. The concept is illustrated as Figure 2.13.

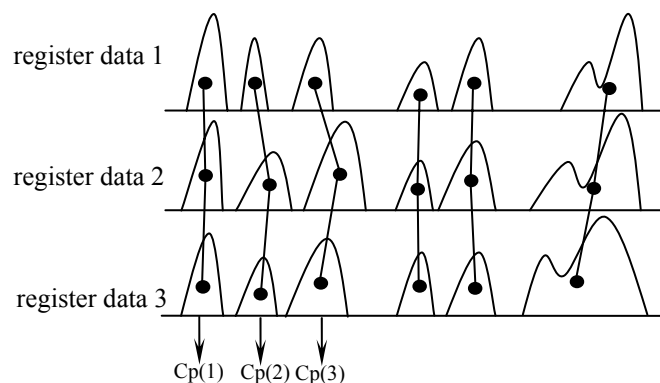


Figure 2.13 Matching the counterpart of register signatures

Then, constructing a new source data by assembling the counterparts of three register signatures. Assembly of new source data is demonstrated as following.

R ₁ (1)	R ₂ (2)	R ₃ (3)		R _{mod(K, 3)+1(k)}
R ₂ (1)	R ₃ (2)	R ₁ (3)		R _{mod(K+1, 3)+1(k)}
R ₃ (1)	R ₁ (2)	R ₂ (3)		R _{mod(K+2, 3)+1(k)}

2) Linearly enlarging or contracting the volume of each stroke data.

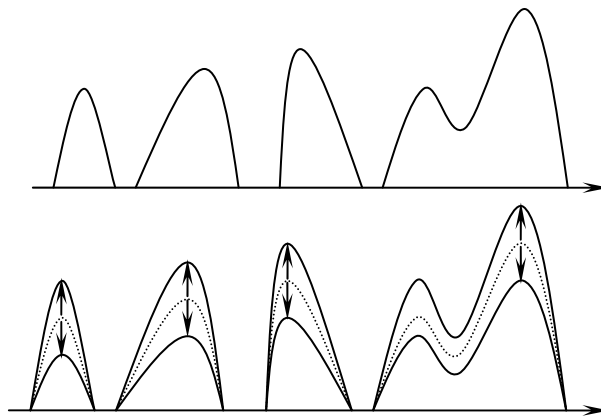


Figure 2.14 linearly converting the source pressure data

3) Deforming the data of each stroke with one of the transformation functions.

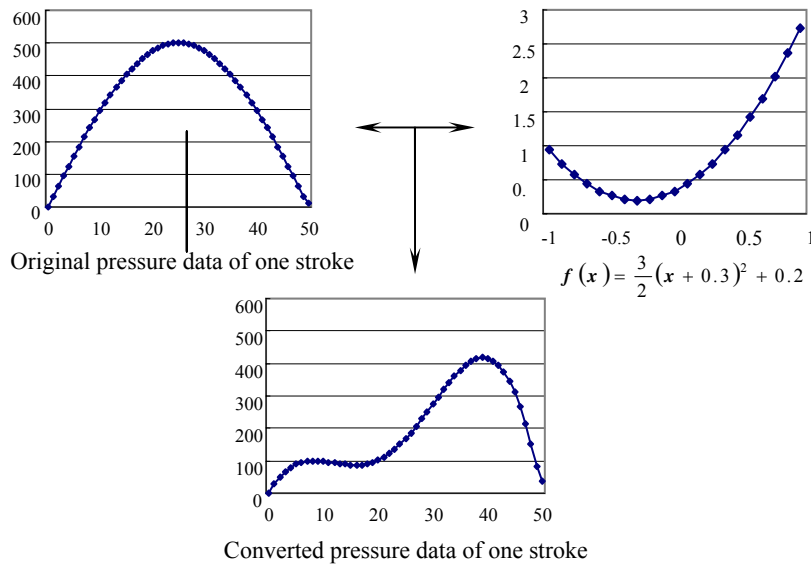


Figure 2.15 Deformation of the source pressure data

After previous three steps of processing, 10 source data are artificially constructed and named inhibit source data and the corresponding inhibit relay data, denoted as $RI^1, RI^2, \Lambda RI^a$ ($a=10$) are made with the way as same as that used for making register relay data.

2.3.5 Making information of valid points group

In this stage, the probability distribution of register relay data and inhibit relay data are statistically analyzed and the position information of valid points in both kinds of relay data is obtained based on the analyzation results. Then the feature vectors are extracted respectively by filtering the corresponding relay data with the valid points group.

For the three register relay data RR^1, RR^2, RR^3 , each data group of rr_i^1, rr_i^2, rr_i^3 ($i=0,1,\Lambda 299$), which are corresponding i th elements of RR^1, RR^2, RR^3 , average value μr_i and standard deviation σr_i are calculated. Thus probability distribution of rr_i^1, rr_i^2, rr_i^3 is represented by the normal distribution with a mean of μr_i and a standard deviation of σr_i . Similarly, the average value μi_i and standard deviation σi_i are calculated for each data group of $ri_i^1, ri_i^2, \Lambda ri_i^a$ ($i=0,1,\Lambda 299$, $a=10$), which are corresponding i th elements of inhibit relay data $RI^1, RI^2 \Lambda RI^a$, and the probability distribution of $ri_i^1, ri_i^2, \Lambda ri_i^a$ is depicted by the normal distribution with a mean of μi_i and a standard deviation of σi_i . The analyzation of probability distribution on the relay data (including register relay data and inhibit relay data) is shown in Figure 2.16.

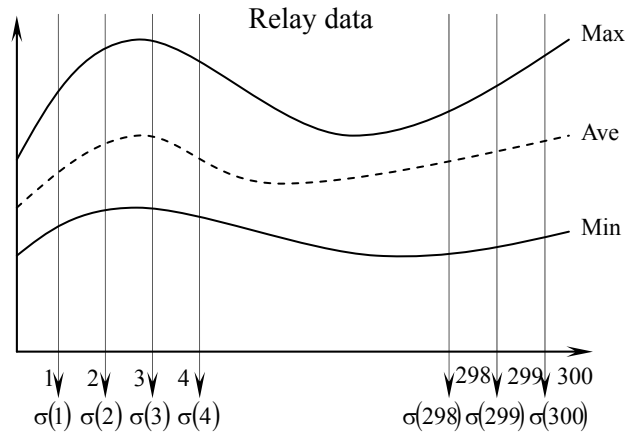


Figure 2.16 analysis of probability distribution of each group of corresponding elements of relay data

With the probability distributions of $rr_i^1, rr_i^2, \wedge rr_i^a$ and corresponding that of rr_i^1, rr_i^2, rr_i^3 are compared and the overlap between two probability distributions is calculated as p_i ($0 \leq p_i \leq 1$). The Comparison of probability distribution is shown in Figure 2.17.

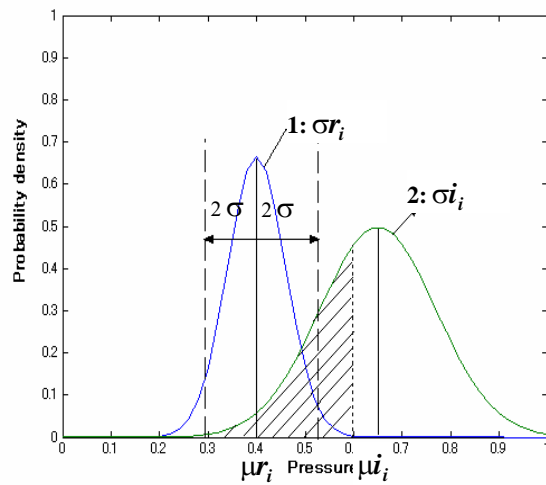


Figure 2.17 Comparison between probability distribution of rr_i^1, rr_i^2, rr_i^3 and that of $rr_i^1, rr_i^2, \wedge rr_i^a$

Then the elements of each register relay data are sorted with sort ascending of p_i , and the 50 elements in the head of queue are marked with 1 as valid points in the corresponding relay data and other elements are marked with 0. Namely for the register relay data $RR^i i \in [1, 2, 3]$, the element rr_j^i , whose corresponding probability distribution

of rr_j^1, rr_j^2, rr_j^3 has no overlap or less overlap with the probability distribution of $ri_j^1, ri_j^2, \dots, ri_j^a$, is considered as the point being able to represent the signature feature of target registrant. In the same way, the position information of valid elements in each inhibit relay data $RI^i, i \in [1, 2, \dots, a]$ are obtained. The procedure of making valid points group is illustrated as Figure 2.18.

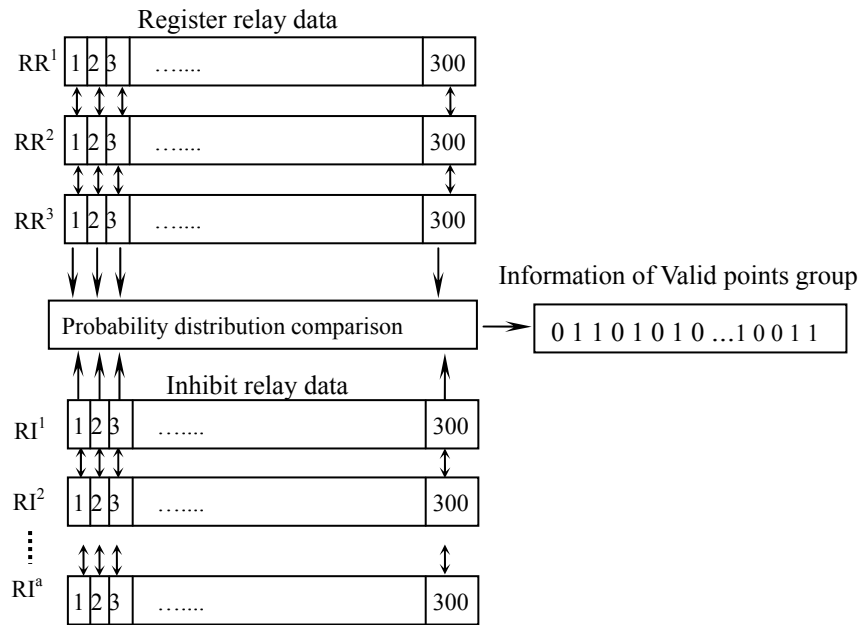


Figure 2.18 Making information of valid points group

The information of valid points, a series composed of zero and one, will be used for extracting feature vector from the relay data (including register relay data and inhibit relay data).

2.3.6 Making NN data file

With obtained position information of valid point, the relay data are filtered and the elements marked with 1 are extracted as feature vector in form of slab value, which is named NN data in the sense that it will be input directly into Neuro-template during training and recognition. The NN data extracted from register relay data are called enforce NN data because they are used as sample of objective class of the neuro-template, the NN data made from inhibit relay data are named inhibit NN data.

The resulted NN data (enforce NN data and inhibit NN data) are saved in a file for later utilization. The process of making NN data file from relay data is shown in Figure 2.19. In this stage, data scale is further reduced from 300 to 50 and personal features embedded in pressure data of signatures are also extracted.

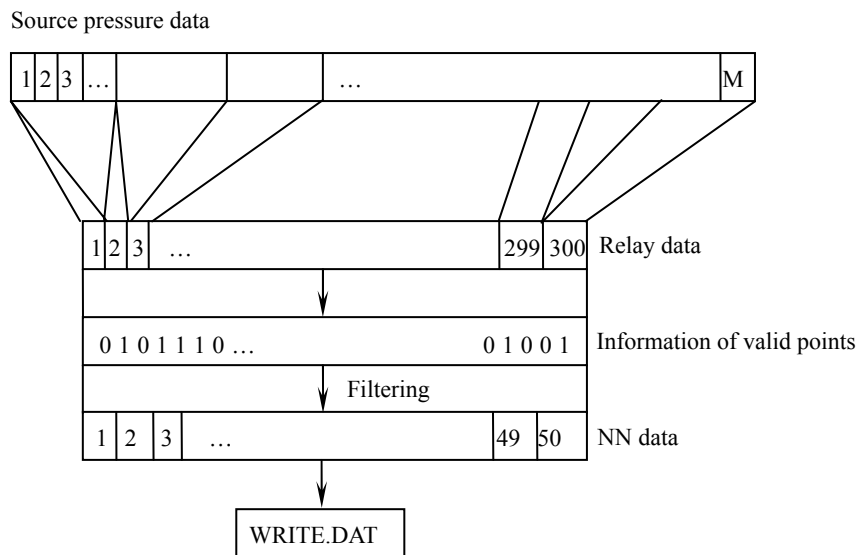


Figure 2.19 Making NN data file

Up to now, the preprocessing on the detected source pressure data of signature is finished.

Chapter 3

NEURO-TEMPLATE CLASSIFIER OF THE SYSTEM

3.1 Introduction

With feature vectors issued from test signature obtained by preprocessing, the identity of signer will be decided by feature comparison. Because signature identification problem, in which the feature vector to test is compared with the feature reference set of all writers, can be reduced to verification issue relating to each user, in which the feature vector to test is compared only with the reference data set of target user, moreover signature verification can ideally be represented as two-class partition problem, the part of system performing the feature comparison task can be called two-class classifier: one is objective-class which consists of genuine signatures of target user, the other

non-objective class including all forgeries and genuine signatures of the people other than target user. Of course, prior to performing comparison, reference data set and classifier should be established with the feature vectors extracted from register signatures. In our signature pen pressure identification system, just as mentioned before, the feature vector is used either as reference data set to generate classifier for the target registrant (registration case) or as the input of each existent classifier to verify the claimed identity of registrant corresponding to the classifier (recognition case). Because only dynamic pressure signal of signature is considered in our application task, the classifier for dynamic-signal-based signature recognition or verification can be divided two categories as following.

One is the conventional symbolic method directly based on probability distribution and statistics performance of reference data set consisted of feature vectors, many comparison techniques of this category, such as maximum likelihood, regional correlation, elastic matching and tree matching, have been proposed and widely used in signature verification and many other recognition fields such as image recognition, voice recognition, electronic signals recognition. However, in the signature verification or identification task, as good forgery samples are not generally available, or not available in sufficient number, the non-objective class is never known completely, the statistical theory based techniques are not very appropriate and reliable. Furthermore the issue of feature extraction of both static and dynamic signature verification has not been satisfactorily solved, in our dynamic pressure of signature based individual recognition system, the global other than local feature is considered, though the preprocessing described in Chapter 2 has shown good effectiveness and the performance of our system is very promising, it is not able to ensure that the feature vectors extracted from source pressure by the preprocessing expatiated in previous chapter are optimal, so the conventional comparison techniques directly based on feature vectors are not good choice for our system. The other problem is that those conventional classifier are often complex and frequently and frequently involve a great deal of repetitive calculation. This often requires very considerable processing time and restricts the application of the signature verification system, in which one of these conventional classifiers is

employed.

The other category of classifier is neural network based comparison techniques. One of great advances in signature verification since 1989 article is the increasingly frequent use of neural networks^[2]. The significant superiority over other techniques made of neural networks to find its way into identity verification systems and be applied in signature segmentation, static signature verification, and dynamic signature verification now. In our system, neuro-template which is the combination of neural networks technique with template matching scheme is employed as classifier of the system.

In pilot study of our system, sigmoid function, which is common choice for the activation function of multilayer feed-forward neural network, is employ as activation function of neurons in hidden layer and output layer of neuro-template. Though the recognition capability of the resulted system for genuine signatures is very promising, the system suffers from poor rejection capability for the skilled counterfeit signatures. Then a novel neuro-template with semi-localized Gaussian ridge function is proposed to improve the unsatisfactory rejection capability of original system for the skilled counterfeit signatures. This chapter is arranged as follows: first, the neural network technique is reviewed. In this part, the basic knowledge of neural network, such as the origination and development, and typical structures of neural networks are introduced. Next, the conventional classifier of our system involving many aspects, such as error function, learning algorithm, parameter decision etc, are described and discussed. In the following section, the problems of original system is presented and the source that leads the problem is analyzed in terms of classifier, then the new classifier with proposed activation function and corresponding superiority is detailed. In later study of the system, the wide parameter sigma of proposed Gaussian ridge function are customized for the classifier of each registrant for further improvement of performance of the system. In the last part of this chapter, the performances of three systems developed in different stage of study are investigated and compared and the effectiveness of proposed method is shown with the favorite experiment results.

3.2 Overview of Neural Networks

In recent twenty years, along with the blossom of artificial intelligence (AI) techniques, the pattern recognition techniques are developed rapidly. The AI techniques, such as fuzzy techniques and neural networks, are widely applied in the classifiers of the recognition system and the improvement on the performance of the corresponding system has been apparent and recognizable. Especially the neural networks technique has increasingly become a practical and effective technology and successfully applied in many fields, the major one of these applications is concerned with pattern recognition problems. Here the neural network is the artificial neural network (ANN), which is a new technique prosperous in recent years. It is inspired by the neurobiological analogy and engineers often look to neurobiology to get new ideas for neuro-computing architectures to solve problems more complex than those easily addressable using conventional techniques. The schematic diagram of a “genetic neuron” with its main components labeled as basic model of neurobiological unit is shown as Figure 3.1

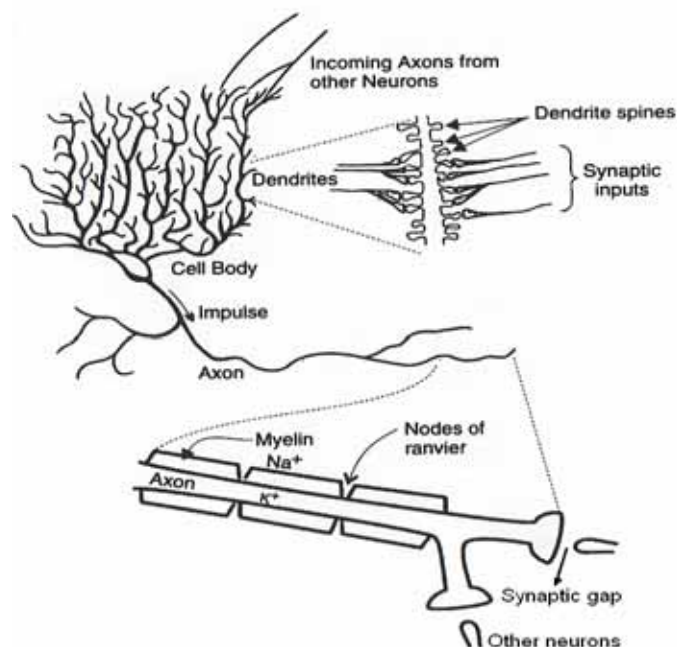


Figure 3.1 A neuron with its components

In Figure 3.1, the characteristic Na^+ , K^+ , Cl^- are the ions prevalent inside and outside the cell membrane.

The soma contains the cell nucleus and is responsible for providing the necessary support functions to the entire neuron. It acts as an information processor by summing the electrical potentials from many dendrites. Dendrites are the receptors of electrical signals from other cells. Axons are charge of carrying the signals of neuron from its head to tail. The tail of axons together with the dendrites or somas of other neurons construct the synapse unite. Synapses are elementary structure and functional units severing as medium of interactions between the neurons.

Artificial neural network technique is originated from the conception of using physical practicable system to imitate the structures and functions of neurons of human brains. It began with the pioneering work of McCulloch and Pitts and their classical studies of neurons described the logical calculus of neural network. The model proposed by McCullon-Pitts could demonstrate substantial computing potential. Then Wiener described important concepts of control, communication, and signal processing based on his perception of similarities between computer and brain which spurred interest in developing the science of cybernetics. Von Neumann used the idealized switch-delay elements derived from the neuron models of McCullonch and Pitts to construct the EDVAC computer and suggested the research in using “brain language” to design brain-like processing machines. The next great major development is proposal of learning scheme for updating the synaptic strengths between the neurons by Hebb. Hebb’s famous postulate of learning, which we now refer to as Hebbian learning rule, stated that information can be stored in synaptic connections and the strength of a synapse would increase by the repeated activation of one neuron by the other one across that synapse. Rochester et al made a important supplement to the Hebb’s theory of learning in the brain, they demonstrated that that it was essential to add inhibition for the theory to actually work for a neuronal assembly. In the 1950s, Frank Rosenblatt proposed a neuron-like element called perceptron. This perceptron was trainable machine and it learned to classify certain patterns by modifying the synaptic strengths.

The perceptron architecture generated a great excitement in the early data of pattern recognition. Rosenblatt provided the learning procedure algorithm for adjusting the free parameters in the network composed by the perceptrons with a number of neurons and connections. At same period, Widrow and Hoff proposed a called Widrow-Hoff learning rule of learning mechanism where the summed square error in the network output was minimized. During the 1960s, Minsky and Pappert demonstrated the computational limitations of single-layer perceptrons with elegant mathematics. Their theorems were widely interpreted as discrediting the utility of all perceptron-like devices as learning machines. However, the discrediting of perceptrons was said to be an overreaction in the later retrospect. Nilsson had shown in his book called *Learning Machines* that multilayer perceptrons can be used to separate patterns nonlinearly in a hyperspace. But a mechanism for learning in multilayer perceptrons was not clear at that time. During the 1970's a handful accomplished the pioneering work in neural networks. Several neural network models were developed during this period which contributed towards gaining a better understanding of mechanisms and circuitry involved in various functions carried out by the brain. Grossberg introduced a number o architectures and theories including an adaptive model of a neuron and showed its use as a short-term memory. In the 1980's the era of renaissance started with several publications that furthered the potential of artificial neural networks. Hopfield's paper introduced fully connected network of neurons and addressed their potential as associative memories.

A principled definition of ANN is quoted from Hecht-Nielsen^[23] like following:

A neural network is a parallel, distributed information processing structure consisting of processing elements (which can possess a local memory and can carry out localized information processing operations) interconnected via unidirectional signal channels called *connections*. Each processing element has a single output connection that branches (“fans out”) into as many collateral connections as desired; each carries the same signal — *the processing element output signal*. The processing element output signal can be of any mathematical type desired. The information processing that goes on within each processing element can be defined arbitrarily with the restriction that it

must be completely local; that is it must depend only on the current values of the input signals arriving at the processing element via impinging connections and on values stored in the processing element's local memory.

A general model of a neuron with connection and the simple processing unit which is capable of performing nonlinear transformations is shown in Figure 3.2.

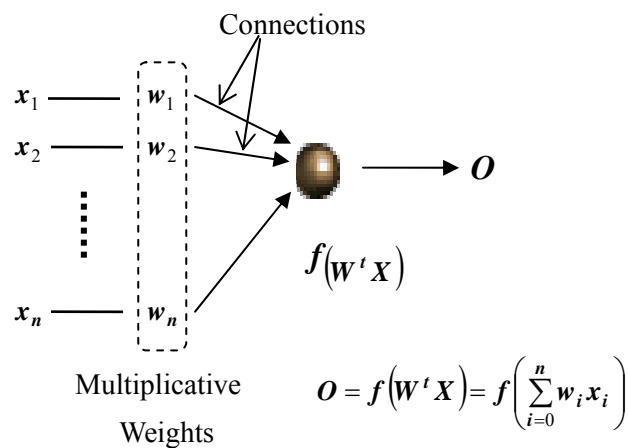


Figure 3.2 the general model of a neuron

In the Figure 3.1, the node is called processing elements (*neurons*), the links in the figure is named *connection*. Each connection acts as instantaneous unidirectional signal-conduction path. Each neuron can accept any number of incoming connections (called *input connections*) and have any number of outgoing connections, but the signals in all of outgoing connections must be same, in effect, each neuron has a single *output connection* that branch or fan out into copies to form multiple output connections each of which carries identical signal (*output signal*). Each neuron can have local memory, and possesses a *transfer function* which can use and alter local memory, use input signals and produce the processing neuron's output signal.

The characters of ANN playing an important role in a wide variety of practical applications are shown in following.

Adaptiveness. Powerful learning algorithms and self-organizing rules allow it to self-adapt as per the requirements in a continually changing environment. Powerful learning algorithms make it possible that the value of connections between each of processing element is flexible and variable with the variation of input information. Self-organizing is that the connections between each of neurons are also variable as the variation of input information.

Distributed memory and redundancy. The information in ANN is distributed and memorized in large number of neurons. It means the network is redundancy, which leads to the superior fault tolerance that the memory of the network is of.

Parallel processing. Architectures with a large number of processing neurons enhanced by extensive interconnectivity provide for concurrent processing as well as parallel distributed information storage.

Nonlinear processing. Ability to perform tasks involving nonlinear relationships and noise-immunity make it a good candidate for classification and prediction. The feature of noise-immunity means the robustness of neural networks and some errors and noises cannot deteriorate the performance of neural networks because of strong interconnections between neurons

Up to now, many kinds of ANN with different structure and learning algorithm have been proposed to solve the problems in different applications, including multi-layer perceptron, radial basic function network, Hopfield network, self-organizing network, etc. The topological structure of a neural network can be basically grouped into two classes: Feed-forward network, and Feedback network. Each neuron only accepts input from previous layer and output to next layer in the feed-forward network. There is no feedback. The feed-forward network generally is composed of several layers. Inputs of the i th layer are just connected with outputs of the $(i-1)$ th layer. In the feedback network all neurons are calculating neuron that can accept inputs from and output to environmental neurones simultaneously. The connections between them are

bi-directional.

It is an essential characteristic of ANN to modify the memory information by adjusting its parameters according to environments. Generally, three kinds of learning mode, including supervised learning, unsupervised learning, and reinforcement learning, have been utilized to parameter regulation.

Supervised learning is the kind of learning studied in most current research in machine learning, statistical pattern recognition, and artificial neural networks. It is learning from examples provided by a knowledgeable external supervisor. The learning system regulates its parameters according to the error between expected outputs and actual outputs. The training data consist of pairs of input objects and desired outputs. The output of the function can be a continuous, or can predict a class label of the input. The task of the supervised learner is to predict the value of the function for any valid input after having seen only a small number of training examples.

Unsupervised learning is a method distinguished from supervised learning by the fact that there is not a supervisor. In unsupervised learning, a data set of input objects is gathered, then is typically treated as a set of random variables. A joint density model is then built for the data set. Unsupervised learning can be used in conjunction with Bayesian inference to produce conditional probabilities for any of the random variables given the others, and be also useful for data compression.

Reinforcement learning is different from any of supervised or unsupervised learning. It is learning what actions can lead to maximize a reward. The learner is not told which actions to take but instead must discover which actions yield the most reward by trying them. Reinforcement learning is a computational approach to understanding and automating goal-directed learning and decision-making.

3.3 Original classifier of the system

In our individual recognition system, neural network technique is employed to process the feature vectors obtained from preprocessing procedure and decide the identity of the signer from whom the feature vector comes. In various candidate types of neural network, multilayer feed-forward neural network(MFNN) is selected as classifier of the system. Different from conventional pattern recognition system in which only one neural network is used to identify all of the classes to recognize, neuro-template matching method which is the combination of MFNN with template matching strategy is introduced in the classifier of the system considering the practical condition of our application.

3.3.1 Neuro-template Matching Method

For the traditional MFNN with many neurons in output layer employed in many pattern recognition systems, when the system recruits new class the structure of MFNN have to be rebuild and the learning of whole network have to be executed to adjusting all the weight parameters with samples of the new class. That would inevitably lead to expensive computation cost including time and required hardware. Furthermore, with an increase of classes in the output layer, the structure of network will become too complicated to effectively work. Though some methods had been proposed to solve this problem, the effectiveness is not satisfying. If the number of the classes to recognize in the system is fixed or can be predicted and not very large, this problem is not serious problem which influences badly the performance of the system. However, for the pattern recognition system in which the number of classes to recognize is variable and can not be predicted or the number is very large, just like the signature identification system, it is seemed impossible for only one traditional MFNN to identify all classes. To address the previous problem, neuro-template matching method, the strategy of which is shown as Figure 3.2, is introduced into our system.

In this method, each user is assigned with one three layer feed-forward neural network with uniform structure of $50 \times 35 \times 2$, namely there are 50 neurons in input layer, 35 neurons in hidden layer and 2 neurons in output layer. Since each user's MFNN has

fixed structure just like template, it is called neuro-template. The structure of each neuro-template is shown in Figure 3.3. For each neuro-template, the corresponding registrant to whom this neuro-template is assigned called target user or target registrant of this neuro-template for convenience (only registrant can be assigned with neuro-template in the system). In the output layer of each neuro-template, one neuron corresponds to the objective class which represents the signature pattern of the target registrant and denoted as o_1 , the other neuron corresponds to non-objective class which characterizes the signature patterns of all other people except the target registrant and labeled o_2 . Each neuro-template is in charge of verification of identity of its corresponding user, that is to say, each neuro-template functions as a sub-classifier which only determine the test user is the target registrant or not. The recognition on the identity of test user is accomplished by matching the feature vector of signature pressure obtained from the test user with all registrant's neuro-templates and evaluating the outputs of all neuro-templates.

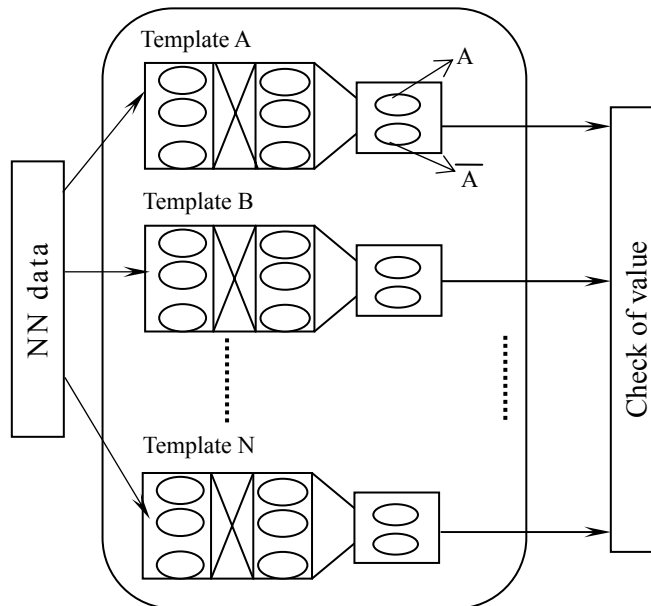


Figure 3.3 Neuro-template matching method

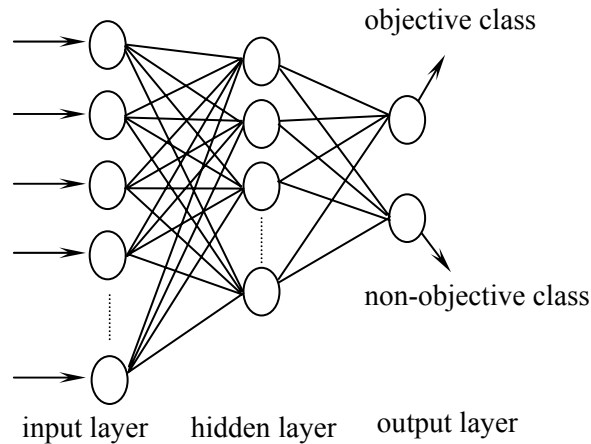


Figure 3.4 Structure of neuro-template

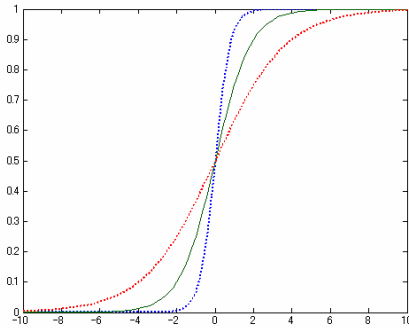
During the enrollment process, the register data from new legitimate registrant are first fed to each existent neuro-template and are evaluated consequently. For each existent neuro-template, if the new registrant is misrecognized as its target registrant, namely false classification caused by new register signatures, then relearning on this neuro-template is activated with new register data as specimen of non-objective class of this neuro-template. Otherwise this template is left untouched. After relearning on all of the existent neuro-templates in which misclassification occurred, a new neuro-template is constructed for the new registrant. That is to say instead of all templates like traditional MFNN in which whole network has to be rebuilt during recruitment of new class, only the learning of new template and the relearning of the existing templates in which the false classification is caused are necessary in the new registration procedure. Therefore registration procedure is significantly simplified and the limitation imposed on the number of registrants is also removed because of the relative independence of neuro-templates. Depending on if prior knowledge (always ID number) is available or not, signature-based individual identification systems can be classified into two groups, one is signature verification in which test signature is compared with the reference signature set indicated by ID number. The other is signature identification in which test signature has to be compared with all reference signatures set. The latter is general

viewed as more difficult. Both modes are included in our system for flexible application. Another advantage of neuro-template is that it makes easy shift between two modes possible.

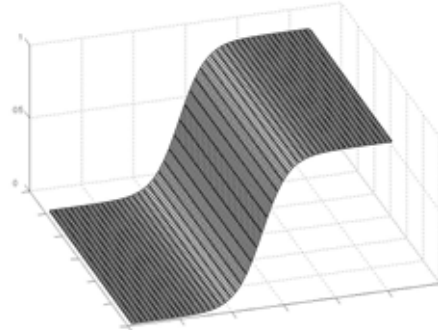
3.3.2 Activation function of original classifier

In the initial stage of studying, sigmoid function, which is conventional choice for the activation function of MFNN, is employed as the activation function of neurons in hidden layer and output layer of each neuro-template to realize the verification of corresponding target registrant, the performance of resulted system is very promising on aspect of recognizing the user whose pattern of signature pressure has been learned, namely the constructed system is able to effectively recognize the identity of the registrants via the pressure of their genuine signature. However, rejection capabilities of this system for skill counterfeit signatures are not satisfying. Counterfeit signatures are the signatures made by simulating the registrants' genuine signatures by intruder and corresponding patterns which should be classified to non-objective class of each neuro-template are, in fact, unknown pattern for each neuro-template because no samples of this class available to train the corresponding neuro-template. The rejection capability for counterfeit signatures is an important performance index for the signature verification or identification system. Therefore it is essential important to improve the rejection capabilities of originally developed system for counterfeit signatures at same time keep the recognition capabilities of the system for genuine signatures satisfying.

In each neuro-template, input-output mapping of each unit in the hidden and output layers is shown as Equation (3.1) and (3.2).



(a) 2-dimensional



(b) 3-dimensional

Figure 3.5 Sigmoid function

$$s = \sum_{i=1}^p w_i x_i - \theta = W^T X - \theta \quad (3.1)$$

$$y = f(s) = \frac{1}{1 + \exp(-s)} \quad (3.2)$$

where $x_i \in \mathbf{R}$, $X \in (x_1, x_2, \dots, x_p)$ is inputs of the neuron. $w_i \in \mathbf{R}$, $W \in (w_1, w_2, \dots, w_p)$ is the corresponding weights of the inputs. $\theta \in \mathbf{R}$ is a threshold. y is the output of the neuron. So the direct mapping relation between input and output of neuron $f(s)$ is sigmoid activation function of given in the following equation

$$y = \frac{1}{1 + \exp(-(W^T X - \theta))} \quad (3.3)$$

where the orientation of the sigmoid is determined by the direction of W , its location is determined by the bias θ , the steepness of sigmoid slope is determined by $\|W\|$. Its 2-dimensional and 3-dimensional shapes are illustrated in Figure 3.5.

Figure 3.5(a) shows the curves of the sigmoid function with different steepness in 2-dimensional space. Figure 3.5(b) illustrates the sigmoid function in 3-dimensional space. As can be seen from the Figure 3.5, the sigmoid function is universal function,

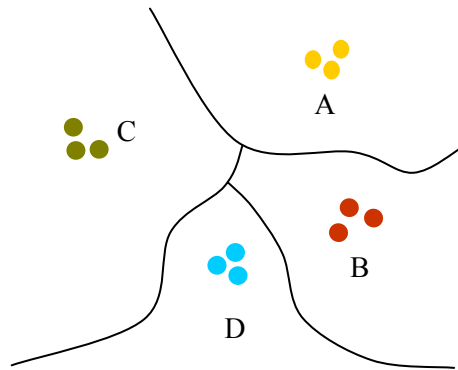


Figure 3.6 classification space of a network with sigmoid nonlinearity

and its activation region is infinite, i.e. they are non-zero in infinite domain. The decision regions for classification are formed by partitioning the input space with the hyperplanes. A multi-layer perceptron with sigmoid activation function can separate the classes by using hidden neurons, which form hyperplanes in the input space as indicated in Figure 3.6. According to the results of many studies, the three-layer neural network with sigmoidal nonlinearities and two layers weights is capable of approximating any decision boundary to arbitrary accuracy^[25] (of course high accuracy is only achieved when sufficient learning and proper parameters of weights and bias). Therefore the neural networks with sigmoid function may have high accuracy in classifying the patterns of the categories whose patterns have been, sufficiently if not completely, learned. However when a specimen of categories which have no samples available for the NN learning, i.e. it is unknown categories for networks, is evaluated by the network, network tends to classify those patterns to its known categories. That is ascribed to the limitation of sigmoid function which is universal function. The network with sigmoid activation function accomplishes the classification only by partitioning the feature space according to samples of a set of known categories instead of representing the probability density of each category and that leads to involvement of the decision regions of only known categories in input space as shown in Figure 3.6 and over extensive decision regions for the known categories. This disadvantage becomes more apparent especially when the classification space is sparse, just like our application case. In our system two

categories (objective category and non-objective one) are included in each neuro-template as described before. For genuine signatures they belong to the known category because samples are available for the learning of the neuro-template, owing to the characters of multi-layer perceptron with sigmoid nonlinearity, they could be effectively classified by corresponding neuro-template. That leads to excellent recognition capabilities for genuine signatures of the system developed in our early research. However for practical counterfeit signatures, Due to too extensive variety of signature patterns included in non-objective category (all kinds of signature features except for that of target registrant), it is impossible to represent non-objective category of the neuro-template substantially by the inhibit data artificially produced in our system. This incomplete representation for non-objective category makes most practical forgeries unknown categories for corresponding neuro-template. Therefore because of the limitation of multi-layer perceptron with sigmoid function as mentioned above, the practical forgeries are very likely to be misclassified as objective class due to high similarity with genuine signatures. Furthermore insufficient learning caused by the shortage of samples not only for objective category but also non-objective category of each neuro-template in our application has also important contribution to the poor rejection capabilities of original system. Figure 3.7 gives an explicit illustration on previous explanation.

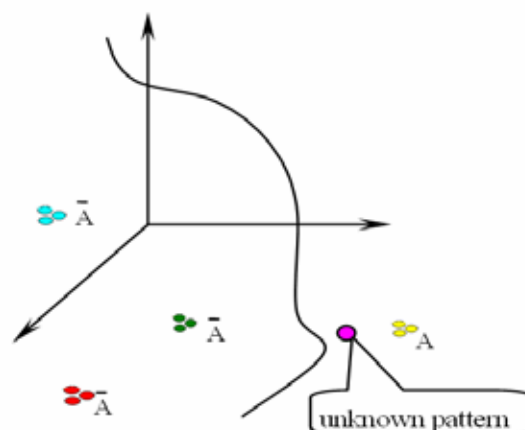


Figure 3.7 classification space of neuro-template with sigmoid function

To improve the insufficient rejection capabilities of the system for skilled forgeries and at same time keep the recognition capabilities for genuine signatures not decrease abruptly, it is essential to construct a more effective neural network based classifier.

3.4 Proposed classifier of the system

As mentioned in previous section, it is necessary to find a more effective classifier which makes the studied individual recognition system have not only excellent recognition capabilities for authentic signatures but also satisfying rejection capabilities for counterfeit signatures.

3.4.1 Proposal of Gaussian activation function

In order to improve the rejection capabilities of the original system for counterfeit signatures without sacrificing the promising recognition capabilities for genuine signatures, a group of neuro-templates corresponding to all registrants with each neuro-template having structure of three-layer FNN used as the classifier, in which Gaussian function is employed as the activation function of each neuron in hidden and output layers, instead of the conventional sigmoid function. The formula of Gaussian function is given as following

$$f(x) = \exp\left(-\frac{x^2}{2\sigma^2}\right) \quad (3.4)$$

where σ is width parameter of the Gaussian function. The 2-dimensional shapes of Gaussian functions with different widths and altitudes are shown in Figure 3.8.

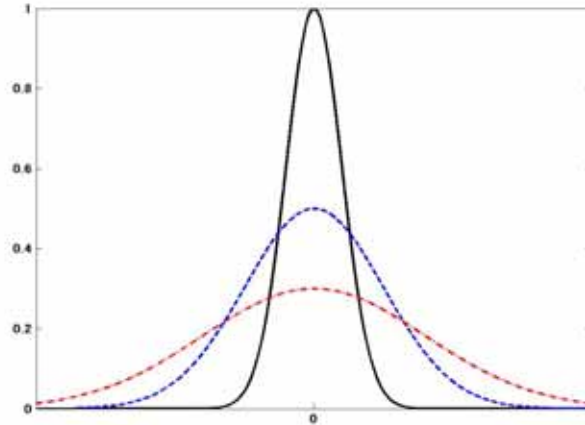


Figure 3.8 Gaussian function

As can be seen from Figure 3.8, the function value reaches zero as the width approaches infinity. Distinct from sigmoid function, Gaussian function is a kind of typical localized function and the active region of the Gaussian function is decided by the width parameter. It presents the distributions of patterns of all known categories by forming the sensitive fields for each known category during NN learning procedure instead of partitioning recognition space as sigmoid function does. As mentioned before, the flexible boundary can be obtained by MFNN with sigmoid function. However, this flexible boundary can not present the distribution of patterns as accurately as the sensitive fields formed by MFNN with Gaussian function for each known category in sparse feature space. Furthermore, in the feature space constructed by MFNN with Gaussian function, the whole area out of the sensitive fields of all known categories corresponds to unknown category, and this area is very extensive in the sparse feature space which is just our application case. As shown in Figure 3.9, when a pattern is out of the sensitive fields of known categories which are controlled by width parameter of Gaussian ridge function and threshold, though similar to a certain known category, it will likely be considered as unknown category. Therefore forged signatures, though near the sensitive field of the category of genuine signatures, may have high probability of being rejected by proper selected width parameter and threshold.

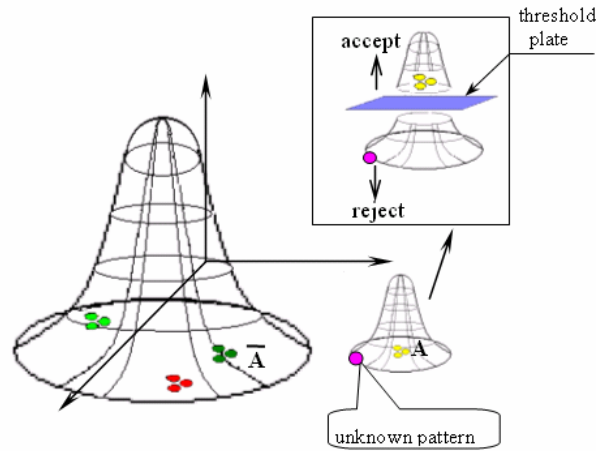


Figure 3.9 Recognition space of neuro-template with Gaussian function

Based on previous comparison and Equation (3.1) and Equation (3.2), one kind of Gaussian function is proposed as activation function of neurons in the hidden and output layer of neuro-template. All the weighted inputs fed to each neuron in the hidden and output layers are summed up, and produce outputs governed by following equation.

$$y = f(X, \xi) = \exp\left[-\frac{(W \cdot X - \theta)^2}{2\sigma^2}\right] \quad (3.5)$$

Where y is the output of each neuron with input vector X in hidden and output layers, $\xi \in \{W, \theta, \sigma\}$ is a set of parameters to be decided. The output of a single Gaussian neuron governed by Equation (3.5) with two input variables is ridge-like function as shown in Figure 3.10. Comparing it with the Figure 3.5(b), which is 3-dimensional figure of sigmoid function with two inputs, great differences between them can be easily found: the activation of the sigmoidal neuron is universal, the input space is directly divided two parts by plane formed by sigmoidal neuron. For the proposed Gaussian neuron, it is a semi-localized function, the sensitivity field of this function stretches out infinitely along the direction of d_2 , i.e. the

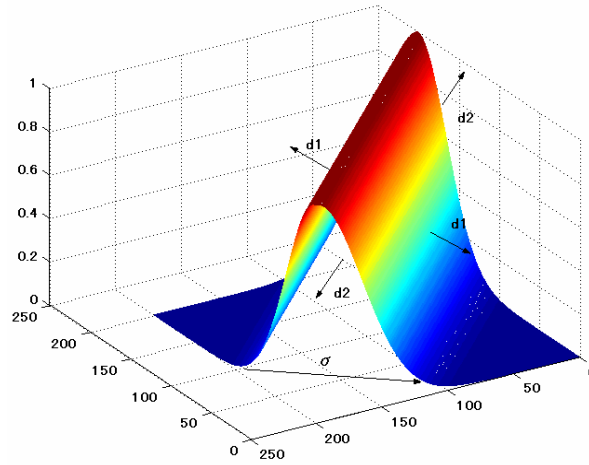


Figure 3.10 the proposed Gaussian ridge function with two input variables

direction of $\mathbf{WX} = \theta$, but the active range is localized and controlled by width parameter σ in the direction of d_2 . This ridge-like function also can be considered as be obtained by adding the two sigmoid neurons on mutually reversed directions together^[24]. Having the advantages both of sigmoid and Gaussian function, the proposed Gaussian ridge function makes the network with it as activation function hav potentials to improve rejection capabilities on premise of ensuring recognition capabilities of the system.

It should be noticed that there are great difference between the proposed Gaussian ridge function and traditional Gaussian function which is absolute localized function and usually employed in radial basis function (RBF) networks. The traditional Gaussian function is governed by Equation (3.6), a statistical transformation based on a Gaussian distribution.

$$y = \exp\left[-\frac{\|\mathbf{X} - \mathbf{C}\|^2}{2\sigma^2}\right] \quad (3.6)$$

Where X is input vector, C is the center vector and σ is the width parameter of Gaussian function.

The typical RBF network bases on Euclidean distance to portray the vector

distribution with a localized function (conventionally Gaussian function). The decision boundary constructed by RBF is therefore a concentric hyper-sphere. So it easily becomes the candidate method of improving the rejection capability of the currency recognition system in our initial stage of research. However the experiment results show that the recognition capabilities for genuine signatures decrease apparently comparing with that of the system with sigmoidal activation function. That is mainly ascribed to the practical situation of our system and the limitation of the basis of RBF network. In our system, the shortage of samples for neuro-template learning makes it difficult to predict accurately the center of data distribution of each category, which is essential parameter for RBF network construction. On the other hand, the responses of the RBF are absolutely localized and the decision boundary constructed by RBF network in the input space is the completely closed hyper-spheres in any direction. Though this hyper-spheres could effectively refuse the unknown patterns such as counterfeit signatures, the genuine signatures, as known pattern, are also greatly refused by the strict decision border of hyper-spheres. Comparing the Eq.(3.5) and Eq.(3.6), it can be found that the C in Equation (3.6) is a vector representing the center of certain class, and the θ in Equation (3.5) is a scalar, it is a bias of the proposed activation function. Thus the output of the proposed function is a ridge-like hypersurface, not a hypersphere.

There is the other kind of semi-local function, Gaussian bars function, whose expression is given in Equation (3.7).

$$y_i = \sum_j w_j \exp\left[-\frac{(x_j - \theta_{ij})^2}{2\sigma_{ij}^2}\right] \quad (3.7)$$

For a neuron with typical Gaussian function and Gaussian bars function as the activation function respectively, the corresponding 3-dimensional decision boundaries are illustrated in Figure 3.11.

From Figure 3.11(b), it can be found that there are not only a central

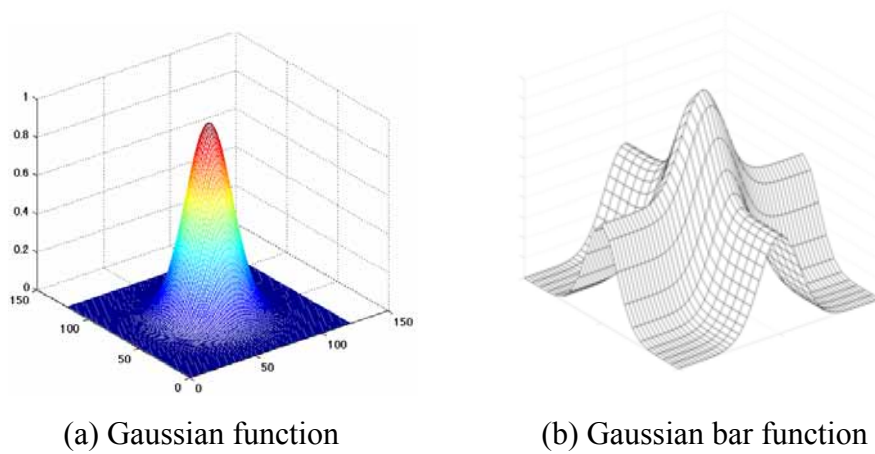


Figure 3.11 3-dimensional Gaussian function and Gaussian bar function

bump just like that of RBF, but also many other ridges stretching out to infinity for Gaussian bar functions. Though more flexible on classification can be obtained by the separating surface of Gaussian bar function than RBF. However, too much parameters have to be learned and regulated for the network with Gaussian bar functions compared with that with proposed Gaussian ridge function and it is obviously an obstacle for our system where the samples are seriously insufficient. Therefore we attempted to use the proposed Gaussian ridge activation function.

3.4.2 Customization of parameter sigma for each neuro-template

In our pilot study, width parameter σ is selected manually and kept constant once proper value is decided for each neurons in hidden layer and output layer of neuro-template. After the employment of proposed Gaussian ridge function with fixed parameter σ , the rejection capabilities of our system for counterfeit signatures are improved apparently. However the degree of improvement is not high enough, and at same time the recognition capabilities of the system for genuine signatures are decreased a little. To further the improvement of rejection capabilities of the system for

counterfeit signatures and recover the deteriorated recognition capabilities for genuine signatures, study is conducted on automatic customization of σ for each neuron in the hidden layer and the output layer of neuro-template. With proper initial value, parameter σ is optimized based on improved Back Propagation algorithm as same as that used for optimization of weights parameter, which will be detailed in next section.

3.5 Neuro-template learning

In this section, many aspects of learning in neural networks will be discussed in detail including error function, initializing parameter, optimization algorithm for weights parameters and decision criteria of learning. The error function used for neuro-template learning is the sum of difference between expected output and actual output of all of neurons in the output layer of the neuro-template, which is varied with the parameters including weights, biases, and widths in the network. The initial values of the parameters of networks are determined based on try and error considering the problem of local minimum. In our system, improved Back Propagation is used to adjust the parameter of weigh, bias and width in later study. In terms of decision criteria, the flexible two steps of criteria are employed for more reasonable classification results in our system.

3.5.1 Error function

The essential task of network learning is to model the underlying generator of the data, so that the best possible predictions for the output vector can be made when the trained network is subsequently presented with a new value for the input vector. The aim of the network training is that for the mapping of a network between inputs and outputs: $y = f(X, \xi)$, $\xi \in \Lambda$, if given a set of training samples (X_m, \hat{y}_m) , $m = 1 \sim M$, exploring a set of optimal ξ^* to minimize the error function (cost function or objective function) of the network. There are many possible choices of error functions which can

be considered to direct neural network training, depending on the particular application. For the problems of classification, whose goal is to model the posterior probabilities of class membership conditioned on the input variables. The common used error function is sum-of-squares error function which is obtained by summing over the all samples in the training set given by

$$E(\xi) = \frac{1}{2} \sum_{m=1}^M \sum_{k=1}^K (\hat{y}_m^k - y^k(\mathbf{X}_m, \xi))^2 \quad (3.8)$$

where $E(\cdot)$ is the cost function of the network. $y^k(\mathbf{X}_m, \xi)$ denotes the output of neuron k which is the function of the input vector \mathbf{X}_m and the parameter vector ξ , K is the number of outputs. \hat{y}_m^k is the target value (teacher value) for output of neuron k with the input vector is \mathbf{X}_m .

In our application, which belongs classification issue, the sum-of-square type of error function, as expressed in following equation, is also employed as cost function to guide neural network learning.

$$E = \frac{1}{2} \sum_{p=1}^P \sum_{i=0}^{n_3-1} (t_{pi} - o_{pi}^{(3)})^2 \quad (3.9)$$

Where t_{pi} is the teacher value for i th neuron in the output layer of neuro-template giving the p th learning sample. $o_{pi}^{(3)}$ is the actual output of i th neuron in the output layer of neuro-template giving the p th learning sample. P is the number of sample sets, n_3 is the number of neurons in the output layer of neuro-template, and $n_3 = 2$.

As already known, the neuro-template have two layers of adaptive parameters and activation functions, the error surface is complicated hyper-plane with many local minima which satisfies $\nabla E = 0$. ∇E denotes the gradient of E in parameter space. To avoid sticking in local minima, the initial value of parameters during network learning

should be selected carefully. On the other hand, measurement should be taken for the learning algorithm.

3.5.2 Initialization of corresponding parameters

As already described in previous section, the error function for neuro-template learning is a complex hyper-plane with many local minimum. To keep the neuro-template learning away from local minimum and explore a set of global optimal parameters, initial values of parameters should be appropriately chosen because most of optimization algorithms are, greatly or partially, initial value dependent. A set of appropriate initial values of parameters may not only lead to global optimal solution of Equation (3.9), but also excellent performance of neuro-template learning, the high convergence speed of learning for example. In our system, though improved Back Propagation (BP) algorithm which has higher possibility of escaping from local minima than traditional BP algorithm, strong sensitivity to the initial value of parameters has also been seen. The majority of initialization procedures in current use is randomly choosing the value in the predetermined range. The aim of using random values is in order to avoid problems due to symmetries in the proposed network. The initial weight values are chosen to be small so that Gaussian activation functions are not driven into the noneffective regions where is far from the centers of Gaussian functions. As mentioned in section 3.4, the distributions of the proposed ridge-like hyper-surface are determined by biases, which decide the center of the ridge-like function, the initial bias values are therefore chosen according to distributions of training samples. The width parameter of proposed Gaussian ridge function, which controls the active regions of the proposed Gaussian function, has strong influence on the performances of the system. Generally speaking the choice of initial values of parameters should be prudent and large quantities of experiment should be executed to try different range of initial values.

3.5.3 Improved Back Propagation (BP) algorithm

Back propagation algorithm, which was initially proposed in a paper by Rumelhart, Hinton and Williams (1986), has been widely utilized in learning of feed-forward neural networks. The application of the BP algorithm involves two stages: during the first stage the input is presented and propagated forward through the network to calculate the output for each neuron. Then the output of neurons in the output layer is compared with corresponding expected value and error between two values is obtained for each output neuron. In the next stage, the errors are passed backward through the network during which signal is passed to each neuron in the network and the appropriate parameter changes are made^[29].

The proposal of Back-propagation algorithm successfully addressed the problem of credit assignment, as mentioned by C. M. Bishop that if incorrect response of output neuron is resulted when an input vector is presented to network, there is no way of determining which hidden neuron should take the responsibility of generating the error, therefore it is impossible to decide which parameters should be adjusted or how much adjustment should be made.

In the BP algorithm, iterative procedure is generally involved combined with one of optimizing algorithms to adjust the parameters in a sequence of steps. In the first stage, the derivatives of the error function with respect to the parameters are calculated. The back-propagation algorithm provides a computationally effective method for evaluating these derivatives. The term back-propagation is to describe this algorithm in the sense of propagating errors backwards through the network. In the second stage, the derivatives are then used to calculate the adjustments of parameters in which the optimization method, such as gradient descent method, conjugate gradient, or quasi-Newton algorithm, is implemented. The flowchart of BP algorithm is illustrated in Figure 3.12.

Since the proposed Gaussian ridge function expressed as Equation (3.5) is employed as the activation function in hidden and output neurons of the neuro-template, the derivatives of the parameters of the network including weights, biases, are evaluated based on the BP algorithm as shown in following.

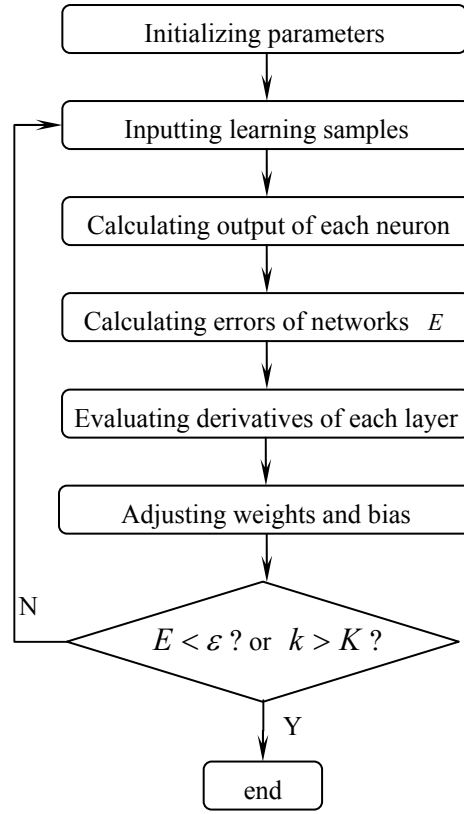


Fig. 3.12 Flowchart of Back-propagation algorithm

As can be seen from the Figure 3.4, each neuro-template is a typical feed-forward neural networks with three layers composed of an input layer, a hidden layer and an output layer. In which $o_i^{(1)}$ ($i = 0, 1, 2, \dots, n_1-1$) is the output of the i th neuron in input layer, $o_k^{(2)}$ ($k = 0, 1, 2, \dots, n_2-1$) is the output of the k th neuron in hidden layer, and $o_j^{(3)}$ ($j = 0, 1, 2, \dots, n_3-1$) is the output of the j th unit in output layer. The Equation (3.9) can be obtained get as following

$$E_p = \frac{1}{2} \sum_{j=0}^{n_3-1} (t_j - o_j^{(3)})^2 \quad (3.10)$$

$$E = \frac{1}{2} \sum_{p=1}^P \sum_{j=0}^{n_3-1} (t_j - o_j^{(3)})^2 \quad (3.11)$$

where t_j is the teacher value of the j th output neuron of the network. E_p is the sum-of-squares error of all output units for the p th set of sample. E is the sum-of-squares error of all samples. On the base of Equation (3.5), we can get

$$o_j^{(3)} = \exp \left[-\frac{\left(\sum_{k=0}^{n_2-1} w_{jk} o_k^{(2)} - \theta_j \right)^2}{2\sigma_j^2} \right] \quad (3.12)$$

$$o_k^{(2)} = \exp \left[-\frac{\left(\sum_{i=0}^{n_1-1} w_{ki} o_i^{(1)} - \theta_k \right)^2}{2\sigma_k^2} \right] \quad (3.13)$$

where w_{jk} are weights between the j th neuron in the output layer and the k th neuron in the hidden layer. θ_j and σ_j are the bias and width of the j th neuron in output layer. w_{ki} are weights between the k th neuron of hidden layer and the i th neuron of input layer. θ_k and σ_k are the bias and width of the k th neuron in hidden layer. Supposing

$$u = \frac{\left(\sum_{k=0}^{n_2-1} w_{jk} o_k^{(2)} - \theta_j \right)^2}{2\sigma_j^2} \quad (3.14)$$

The partial derivatives of E with respect to the parameters (w_{jk} , θ_j , σ_j) in the output layer are calculated as following

$$\ominus \quad \frac{\partial E_p}{\partial o_j^{(3)}} = -(t_j - o_j^{(3)}), \quad \frac{\partial o_j^{(3)}}{\partial u} = -e^{-u} = -o_j^{(3)},$$

and
$$\frac{\partial u}{\partial w_{jk}} = \frac{o_k^{(2)} \cdot \left(\sum_{k=0}^{n_2-1} w_{jk} o_k^{(2)} - \theta_j \right)}{\sigma_j^2},$$

$$\frac{\partial u}{\partial \theta_j} = -\frac{\sum_{k=0}^{n_2-1} w_{jk} o_k^{(2)} - \theta_j}{\sigma_j^2}$$

$$\frac{\partial E_p}{\partial w_{jk}} = \frac{\partial E_p}{\partial o_j^{(3)}} \cdot \frac{\partial o_j^{(3)}}{\partial u} \cdot \frac{\partial u}{\partial w_{jk}}$$

$$\frac{\partial E_p}{\partial \theta_j} = \frac{\partial E_p}{\partial o_j^{(3)}} \cdot \frac{\partial o_j^{(3)}}{\partial u} \cdot \frac{\partial u}{\partial \theta_j}$$

$$\therefore \frac{\partial E_p}{\partial w_{kj}} = -(o_j^{(3)} - t_j) \cdot o_j^{(3)} \cdot \frac{\sum_{k=0}^{n_2-1} w_{kj} o_k^{(2)} - \theta_j}{\sigma_j^2} \cdot o_k^{(2)} \quad (3.15)$$

$$\frac{\partial E_p}{\partial \theta_j} = (o_j^{(3)} - t_j) \cdot o_j^{(3)} \cdot \frac{\sum_{k=0}^{n_2-1} w_{kj} o_k^{(2)} - \theta_j}{\sigma_j^2} \quad (3.16)$$

The partial derivative of E_p with respect to the corresponding parameters(w_{ki} , θ_k) in the hidden layer are computed as following, set

$$v = \frac{(\sum_{i=0}^{n_1-1} w_{ki} o_i^{(1)} - \theta_k)^2}{2\sigma_k^2} \quad (3.17)$$

$$z = \sum_{k=0}^{n_2-1} w_{jk} o_k^{(2)} - \theta_j \quad (3.18)$$

$$\frac{\partial E_p}{\partial w_{ik}} = \frac{\partial E_p}{\partial o_j^{(3)}} \cdot \frac{\partial o_j^{(3)}}{\partial u} \cdot \frac{\partial u}{\partial z} \cdot \frac{\partial z}{\partial o_k^{(2)}} \cdot \frac{\partial o_k^{(2)}}{\partial v} \cdot \frac{\partial v}{\partial w_{ki}}$$

$$\frac{\partial E_p}{\partial \theta_k} = \frac{\partial E_p}{\partial o_j^{(3)}} \cdot \frac{\partial o_j^{(3)}}{\partial u} \cdot \frac{\partial u}{\partial z} \cdot \frac{\partial z}{\partial o_k^{(2)}} \cdot \frac{\partial o_k^{(2)}}{\partial v} \cdot \frac{\partial v}{\partial \theta_k}$$

where $\frac{\partial u}{\partial z} = \frac{\sum_{k=0}^{n_2-1} w_{kj} o_k^{(2)} - \theta_j}{\sigma_j^2}$, $\frac{\partial z}{\partial o_k^{(2)}} = w_{jk}$, and $\frac{\partial o_k^{(2)}}{\partial v} = -o_k^{(2)}$

$$\frac{\partial v}{\partial w_{ki}} = \frac{\sum_{i=0}^{n_1-1} w_{ki} o_i^{(1)} - \theta_k}{\sigma_k^2} \cdot o_i^{(1)}$$

$$\frac{\partial v}{\partial \theta_k} = -\frac{\sum_{i=0}^{n_1-1} w_{ki} o_i^{(1)} - \theta_k}{\sigma_k^2}$$

$$\therefore \frac{\partial E_p}{\partial w_{ik}} = (o_j^{(3)} - t_j) \cdot o_j^{(3)} \cdot \frac{\sum_{k=0}^{n_2-1} w_{kj} x_k}{\sigma_j^2} \cdot \left(\sum_{k=0}^{n_2-1} w_{kj} \right) \cdot o_k^{(2)} \cdot \frac{\sum_{i=0}^{n_1-1} w_{ki} o_i^{(1)} - \theta_k}{\sigma_k^2} \cdot \sum_{i=0}^{n_1-1} o_i^{(1)} \quad (3.19)$$

$$\frac{\partial E_p}{\partial \theta_k} = -(o_j^{(3)} - t_j) \cdot o_j^{(3)} \cdot \frac{\sum_{k=0}^{n_2-1} w_{kj} o_k^{(2)}}{\sigma_j^2} \cdot \left(\sum_{k=0}^{n_2-1} w_{kj} \right) \cdot o_k^{(2)} \cdot \frac{\sum_{i=0}^{n_1-1} w_{ki} o_i^{(1)} - \theta_k}{\sigma_k^2} \quad (3.20)$$

With obtained derivatives as shown above, the differences between E_p and E can be explained by Equation 3.9 and 3.10. The changes of weight parameter and bias parameter are made to get minimum of the cost function as following equation.

$$\Delta w_{ij}(k) = -\eta_\omega \nabla E|_{w_{ij}} + \alpha_\omega \Delta w_{ij}(k-1) + \beta_\omega \Delta w_{ij}(k-2) \quad (3.21)$$

$$\Delta \theta_j(k) = -\eta_\theta \nabla E|_{\theta_j} + \alpha_\theta \Delta \theta_j(k-1) + \beta_\theta \Delta \theta_j(k-2) \quad (3.22)$$

Where $\eta_w, \alpha_w, \beta_w$ are learning rate, momentum coefficient, and oscillation coefficient for weight parameter respectively, similarly $\eta_\theta, \alpha_\theta, \beta_\theta$ are that for bias parameter respectively. which will be described in the following part. $\Delta w_{ij}(k), \Delta w_{ij}(k-1), \Delta w_{ij}(k-2)$ are increase of weight parameter w_{ij} in k th, $(k-1)$ th, $(k-2)$ th iteration respectively, $\Delta \theta_j(k), \Delta \theta_j(k-1), \Delta \theta_j(k-2)$ are variation of bias parameter θ_j at k th, $(k-1)$ th, $(k-2)$ th iteration respectively. $\nabla E|_{w_{ij}}$ denotes the partial

derivative with respect to one weight parameter w_{ij} , $\nabla E|_{\theta_j}$ means that for a bias parameter θ_j . Where E is the average error caused by all learning samples and it is the sum of E_p , which is the error caused by a set of samples in which one sample corresponds to one class to learn.

Learning rate η is also known as step size, it controls the moving distance of the parameter in each update and should be small enough to guarantee that the average direction of iterative search in parameter space is the approximately contrary with that of the local gradient in order to realize the steady decrease of the error function. In the case of constant η , if it is too large, the overshoot will be occurred, and that may induce the increase and instabilities of the error function, which is always reflected as a divergence of the error. On the contrary, too small learning rate may result in extremely slow convergence of the search and difficulty of striding over adjacent local minima. In our system, the learning rate is adaptively modified during the learning procedure of neuro-template: first, the average error in each iteration is calculated and compared with that of the last iteration. If the increase of average error is observed, an indicator used for reducing learning rate increases with increment of 1. Once this indicator becomes greater than a certain value which serves as punishment, the learning rate will be decreased a little. That means if the present direction of search is not expected, the step size of search is reduced correspondingly. Conversely, if average error decreases, that means the direction of search is expected one, the learning rate is increased in the similar way as that used for decreasing it.

Momentum and Oscillation terms To accelerate the convergence speed of neuro-template learning and improve the capability of escaping from local minimum, a momentum term and an oscillation term are introduced into the conventional BP algorithm, which suffers from the problem of easily getting into local minimum. In The improved BP algorithm expressed as Equation 3.23, the momentum term, which is the product of momentum coefficient and variation of the learning variable at last iteration, has the function of effectively speeding the convergence of learning, increasing the

inertia of the system and constraining oscillations of the iteration, the momentum coefficient, denoted as α , is usually chosen between 0 and 1, namely $0 < \alpha < 1$, the more adjacent to 1 α is, the greater inertia the neuro-template learning has, that leads to higher convergence speed, however, higher probability of overshoot at same time. The oscillation term, which is production of oscillation coefficient and the variation of learning variable at last but one time of iteration, serves to get the search out of local minima. Unlike the momentum coefficient, the oscillation coefficient, marked as β , is usually selected between -1 and 0 ($-1 < \beta < 0$). The greater oscillation coefficient is able to lead to great capabilities of escaping from local minimum of the search, however, too large oscillation coefficient may result in the oscillation of the search procedure and consequently slow convergence speed.

From Equation 3.23, it can be seen that the variation of learning variable in last two iterations are involved in deciding the modification of learning variable in current iteration by the momentum term and oscillation term separately. That means that in procedure of neuro-template learning, the change of variable to learn is not only directed by the partial derivative information of the error, but also influenced heuristically by the information of history modification of the variable and the degree of influence is decided by the magnitude of momentum coefficient and oscillation coefficient.

3.5.4 Customization of sigma σ

As mentioned before, the width parameter σ of neuro-template is kept constant once proper value is determined in our pilot study of the system with proposed Gaussian ridge function. Though the employment of Gaussian ridge function with fixed width parameter σ made effective improvement on the rejection capabilities of the system for skilled forgeries, however, also led to a little decrease of the recognition capabilities of the system for genuine signatures, Furthermore degree of improvement on rejection capabilities of the system is not great enough. In the further study the width parameter is

customized to each neuron during the learning procedure of neuro-template for further improvement of performance of our system.

The customization of width parameter σ is conducted as following. First the partial derivative of E_p with respect to width parameters σ in the output layer are calculated.

1) For width parameter in the output layer of the neuro-template σ_j , according to Equation 3.10, 3.12, 3.14, it can be obtained that

$$\frac{\partial E_p}{\partial o_j^{(3)}} = -(t_j - o_j^{(3)}), \quad \frac{\partial o_j^{(3)}}{\partial u} = -e^{-u} = -o_j^{(3)}, \quad \frac{\partial u}{\partial \sigma_j} = -\frac{(\sum_{k=0}^{n_2-1} w_{jk} o_k^{(2)} - \theta_j)^2}{\sigma_j^3}$$

$$\frac{\partial E_p}{\partial \sigma_j} = \frac{\partial E_p}{\partial o_j^{(3)}} \cdot \frac{\partial o_j^{(3)}}{\partial u} \cdot \frac{\partial u}{\partial \sigma_j}$$

$$\frac{\partial E_p}{\partial \sigma_j} = (o_j^{(3)} - t_j) \cdot \frac{(\sum_{k=0}^{n_2-1} w_{jk} \cdot o_k^{(2)} - \theta_j)^2}{\sigma_j^3} \quad (3.23)$$

2) The partial derivative of E_p with respect to the width parameters σ_k in the hidden layer is calculated as following.

From Equation 3.10, 3.12, 3.13, 3.14, 3.17, 3.18, we can obtain that.

$$\frac{\partial E_p}{\partial o_j^{(3)}} = -(t_j - o_j^{(3)}), \quad \frac{\partial o_j^{(3)}}{\partial u} = -e^{-u} = -o_j^{(3)}, \quad \frac{\partial u}{\partial z} = \frac{\sum_{k=0}^{n_2-1} w_{jk} o_k^{(2)} - \theta_j}{\sigma_j^2}, \quad \frac{\partial z}{\partial o_k^{(2)}} = w_{jk}, \quad \text{and}$$

$$\frac{\partial o_k^{(2)}}{\partial v} = -o_k^{(2)}, \quad \frac{\partial v}{\partial \sigma_k} = -\frac{(\sum_{i=0}^{n_1} w_{ik} x_i - \theta_k)^2}{\sigma_k^3}$$

$$\frac{\partial E_p}{\partial \sigma_k} = \frac{\partial E_p}{\partial o_j^{(3)}} \cdot \frac{\partial o_j^{(3)}}{\partial u} \cdot \frac{\partial u}{\partial z} \cdot \frac{\partial z}{\partial o_k^{(2)}} \cdot \frac{\partial o_k^{(2)}}{\partial v} \cdot \frac{\partial v}{\partial \sigma_k}$$

$$\frac{\partial E_p}{\partial \sigma_k} = -(o_j^{(3)} - t_j) \cdot o_j^{(3)} \cdot \frac{\sum_{k=0}^{n_2-1} w_{jk} o_k^{(2)}}{\sigma_j^2} \cdot w_{jk} \cdot o_k^{(2)} \cdot \frac{(\sum_{i=0}^{n_1-1} w_{ki} o_i^{(1)} - \theta_k)^2}{\sigma_k^3} \quad (3.24)$$

Second, with Equation 3.23 and 3.24, the modification of the width parameter sigma σ in the hidden layer and the output layer of the neuro-template is governed as following.

$$\Delta \sigma_i(k) = -\eta_\sigma \nabla E|_{\sigma_i} + \alpha_\sigma \Delta \sigma_i(k-1) + \beta_\sigma \Delta \sigma_i(k-2) \quad (3.25)$$

Where $\eta_\sigma, \alpha_\sigma, \beta_\sigma$ are learning rate, momentum coefficient, and oscillation coefficient for width parameter respectively, $\Delta \sigma_i(k), \Delta \sigma_i(k-1), \Delta \sigma_i(k-2)$ are variation of width parameter σ_i in k th, $(k-1)$ th, $(k-2)$ th iteration respectively, $\nabla E|_{\sigma_i}$ denotes the partial derivative with respect to one width parameter σ_i . E has the same meaning as Equation 3,21 and 3.22.

3.6 Experiments

In this section, many experiments are conducted to evaluate the effectiveness of proposed method and the customization of parameter σ for each neuro-template. In these experiments, the performances of three individual recognition systems developed in different stage of research, which are the original system based on neuro-template with sigmoid function(abbreviated System.I), the improved system based on neuro-template with proposed Gaussian ridge function in which width parameter σ is fixed once proper value is chosen (marked as System. II), and the further improved system based on neuro-template with Gaussian ridge function in which the width parameter σ has been customized for each neuro-template (denoted as System.III), are studied and compared. Furthermore other factors of system, such as database, criteria of performance measurement, convergence speed, learning parameters (learning rate, momentum coefficient and oscillation coefficient), are examined. Last, the experiments

for investigating the mutual influence of the neuro-templates are planned and corresponding results are demonstrated and explained.

3.6.1 Experiment database and learning condition

According to literature available so far, the database used in various systems vary considerably to one another. This great diversity is reflected mainly in two aspects: one is the scale of database, the other is categories of forged signatures. Two kinds of forgeries, simple forgery and skilled forgery, have been used in different systems.

Since no common criterion is available to judge various databases, considering practical factors and application situation of our system, we arrange the database as follows, 12 subjects are included in our database as target registrants, and for each target registrant, 53 signatures are collected as genuine samples from the corresponding registrant by three times on a different date, taking intrapersonal signature variance covering different period into account. First three signatures are used for enrollment as reference data and the rest is used for evaluation as test signatures. Then 50 forgeries are sampled as counterfeit signatures from five other people with 10 forgeries per person, that is to say that other target registrants may be involved in giving counterfeit signatures as forger. Each forger is shown the authentic signatures of target registrant, and then practices forging by tracing. That indicates that skilled forgeries are used in our experiment. All data are collected online and kept in a database, and then the identical data are fed to different systems offline to make sure of impartial comparison. Another item that should be noted is that signature verification, as described before, can be viewed as part of signature identification in our system, therefore performance evaluation in our experiment is only based on signature identification mode, which is considered more difficult. The conditions of neuro-template learning is shown in Table 3.1.

The other conditions of learning, such as learning rate, momentum coefficient, oscillation coefficient and two thresholds for identity decision, are different according

the systems with different activation function.

Table 3.1 Conditions of neuro-template learning

Number of input neurons	50
Number of hidden neurons	35
Number of output neurons	2
Number of patterns to learn	2
Number of samples for each pattern to learn	3
Convergence criterion	Average of square error<0.0001
Maximum iteration number	20000

3.6.2 Performance measurement and decision criteria

There are two generally accepted criteria for performance evaluation in signature verification systems. One is used to evaluate the generalization capability for genuine signatures, known as recognition capabilities and measured by the ratio of the genuine signatures being successfully recognized to the whole test genuine signatures. The other is used to measure the discrimination capability against forged signatures, known as rejection capabilities and presented by the ratio of the forgeries being correctly rejected to all test forgeries. In our experiment these two criteria are also employed to evaluate the performance of the system.

Two steps are involved in our decision-making process.

- 1) *Intra-template decision*: the test data is verified as target registrant N if and only if

$$o_1^N > th1 \quad \text{and} \quad o_1^N - o_2^N > th2 \quad (3.26)$$

Where o_1^N means the output value of neuron corresponding to objective class and o_2^N denotes that corresponding to non-objective class in the output layer of neuro-template N . $th1$ and $th2$ are two predetermined threshold values. Inequality 3.26 indicates that when matching with neuro-template N , the test signer will not be verified as target registrant N until it meets the requirements of not only similarity with

objective class (N) and but also discrepancy with non-objective class (not N), that gives more strict scrutiny for the test data especially for the counterfeit signatures.

2) *Inter-template decision*: After matching with all of the neuro-templates, if a test data is falsely verified as multiple registrants, A, C , and N for example, though this case is rarely occurred, the outputs of neurons corresponding to the objective class in the related neuro-templates, o_1^A, o_1^C and o_1^N , will be reevaluated and the test signer will be recognized as registrant N if and only if

$$o_1^N = \max(o_1^N, o_1^A, o_1^C) \quad (3.27)$$

After scrutiny of two steps, a unique identity will be assigned to the test signer.

3.6.3 Performance of the systems

1) Recognition capabilities for genuine signatures

As mentioned immediately before, recognition capabilities of the system are the essentially important criterion of evaluating the generalization performance of the individual recognition system for genuine signatures. In this section, the recognition capabilities of three systems, which are System.I, System. II and System.III, are evaluated firstly with the database described before, and obtained results are listed and compared in Table 3.2.

In Table 3.2, ‘Registrant’ means target registrant who has registered on the system and ‘Function’ denotes the activation function. the ‘Difference’ item means performance variation between the System.I and the System.III. The minus sign (-) here indicates deterioration of the performance of System.III compared with System.I and the positive one (+) denotes improvement.

Table 3.2 Recognition capabilities of three systems for genuine signatures

Function Registrant	Sigmoid	Gaussian with fixed σ	Gaussian with optimal σ	Difference
A	96%	92%	96%	0%
B	96%	92%	94%	-2%
C	100%	100%	100%	0%
D	92%	88%	88%	-4%
E	100%	100%	100%	0%
F	96%	94%	96%	0%
G	100%	100%	100%	0%
H	96%	94%	96%	0%
I	92%	94%	92%	0%
J	90%	84%	90%	0%
K	88%	100%	100%	+12%
L	88%	86%	86%	-2%
Average	94.5%	93.7%	94.7%	+2%

From Table 3.2, it can be seen that the recognition capabilities of the original system (System.I) are very promising. In System. with the employment of Gaussian ridge function with fixed σ , small decrease of recognition capabilities with average value of -0.8% can be observed (case A, B, D, F, H, J and L) comparing with original System.I, there are also several cases(case C, E, G) in which the recognition capabilities are remained constant. The only exception cases are the case of I and K in which the recognition capability are improved instead of decrease. For System.III, it can be seen that the deteriorated recognition capability resulted by e proposed Gaussian ridge function with fixed σ was partially recovered by the optimization of σ for each neuro-template (average recognition capability is improved from 93.7% to 94.7%). So it can be concluded that the proposed Gaussian ridge function with fixed sigma may lead to a slight deterioration of recognition capabilities of our system, but the extent of deterioration of recognition capabilities is greatly reduced by the customization of width parameter σ .

2) *Rejection capabilities for counterfeit signatures*

The corresponding rejection capabilities of three systems (System.I, System. , System.III) are shown in Table 3.2.

Table 3.3 Rejection capabilities of three systems for counterfeit signatures

Function Registrant	Sigmoid	Gaussian with fixed σ	Gaussian with optimal σ	Difference
A	60%	70%	90%	+30%
B	86%	94%	94%	+8%
C	74%	82%	86%	+12%
D	56%	78%	86%	+30%
E	66%	78%	78%	+12%
F	42%	62%	72%	+30%
G	52%	64%	82%	+30%
H	92%	98%	98%	+6%
I	90%	92%	96%	+6%
J	60%	82%	82%	+22%
K	30%	82%	84%	+50%
L	46%	58%	66%	+20%
Average	63.2%	78.2%	84.5%	+21.3%

According to Table 3.3, a recognizable increase of rejection capabilities of the system for counterfeit signatures (average value of improvement is +15%) can be seen in System. comparing with system.I, that shows the effectiveness of proposed Gaussian ridge function with fixed σ on improving the rejection capabilities of our system for counterfeit signatures. Comparing System.III with System. , it can easily be seen that the improvement is greatly furthered with optimization of σ for neuro-template (the increment (+16.3%) is even greater than that from System.I to System. which is +15%).

From Table 3.2 and 3.3, it can be seen that compared with the System. , optimization of neuro-template is either able to further improves the rejection capability of the system with keeping recognition capabilities constant (case C, D, F, G, K and L) or recovered the deteriorated recognition capability while keeping rejection capability invariant (case B, J and H), or ameliorate both rejection and recognition capabilities (case A). The exception case is E in which both capabilities of system are not influenced

by optimization of neuro-template. The other exception case is I, in which the rejection capability is improved (+4%) while the recognition capability decreased a little (-2%). From these results, it can be concluded that the neuro-template based on proposed Gaussian ridge function with optimized parameter σ seems to have considerable effectiveness in improving the rejection capabilities of the system for the counterfeit signatures at the same time keeping the recognition capability for genuine signatures satisfied.

To further show the effectiveness of the proposed method, the variations of recognition and rejection ratio of the system with different threshold $th1$ are shown in Figure 3.13. The curves in Figure 3.13 are obtained by varying threshold $th1$ from 0.1 to 0.9 and at same time keeping threshold $th2$ at 0.2. This is because that in our system, the first criteria $o_1^N > th1$ is essential criteria and the second one $o_1^N - o_2^N > th2$ is auxiliary criteria. Curve 1 and 2 are variations in recognition ratio and rejection ratio of System.III with different threshold respectively, while curve 3 and 4 show the changes of recognition and rejection ratios of System.I with different threshold respectively.

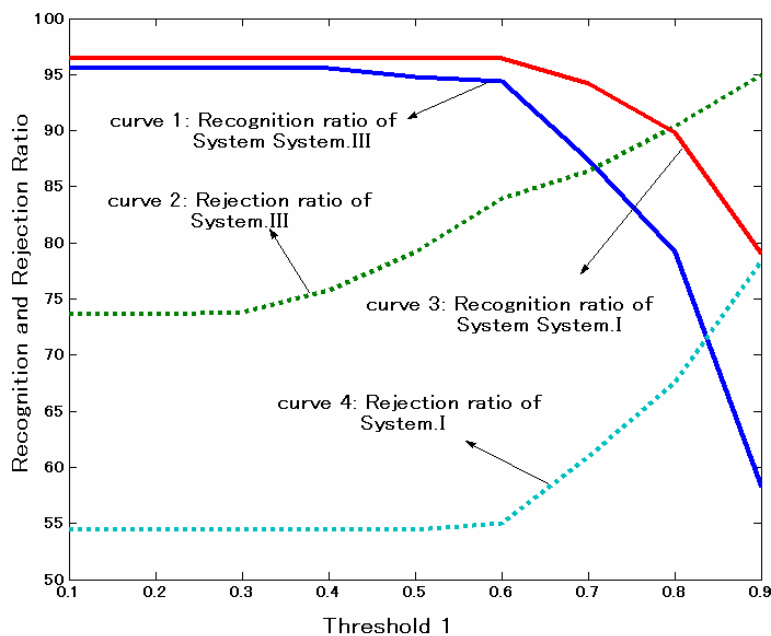


Figure 3.13 Variations of recognition ratios and rejection ratios of System.I and System.III with different threshold $th1$

From Figure 3.13, it can be seen that though the recognition capabilities of the System.III (curve 1) have a fast decrease than that of System.I (curve 3), the sharp decrease began from a certain threshold 0.6. That is to say that the decrease of recognition performance of System.III could be avoid partially by proper selection of threshold. Different from curve 1 and 3, curve 2 is always much higher than curve 4 at any threshold, and it shows that the rejection capabilities of System.III is actually improved by the proposed method. It should be noticed that the threshold $th1$ is chosen to get optimal performance of system including both recognition and rejection capabilities. Therefore different system may achieve optimal performance at different threshold $th1$. In Table 3.2 and Table 3.3, the threshold $th1$ for System.I is chosen as 0.7, while for the systems with Gaussian ridge function (both System.II and System.III), $th1$ is 0.6. Another factor to which attention should be paid is that all of results listed before are obtained at same conditions, such as initial value of learning parameters(η, α, β) and weights parameters.

3.6.4 Influence of width parameter σ

In this section, the influences of the width parameter on the performance of the system, including recognition capabilities, rejection capabilities and convergence speed, are investigated. To avoid the influences of biases, all biases of hidden neurons and output neurons are keep constant at 0. Then the same values of width parameters are selected for each neuron in hidden layer and output layer and modified manually. The recognition ratios and rejection ratios for 12 target registrants with the different widths (0.4, 0.75, 0.8, and 1.2, respectively.) are listed in Table 3.4 and Table 3.5. In both tables, the performance of original system with sigmoid function is also concluded for easy comparison. it should be noticed that previous results are obtained with same initial value of other parameters except σ .

Table 3.4 Recognition capabilities of our systems with different σ

Function Registrant	Sigmoid	Gaussian with fixed σ			
		$\sigma = 0.4$	$\sigma = 7.8$	$\sigma = 0.8$	$\sigma = 1.2$
A	96%	84%	92%	92%	88%
B	96%	82%	92%	92%	86%
C	100%	90%	100%	100%	94%
D	92%	86%	88%	90%	82%
E	100%	88%	100%	98%	92%
F	96%	82%	94%	96%	88%
G	100%	84%	100%	100%	90%
H	96%	82%	94%	96%	88%
I	92%	84%	94%	92%	86%
J	90%	76%	84%	86%	80%
K	88%	72%	100%	100%	92%
L	88%	68%	86%	86%	80%
Average	94.5%	81.5%	93.7%	94%	87.2%

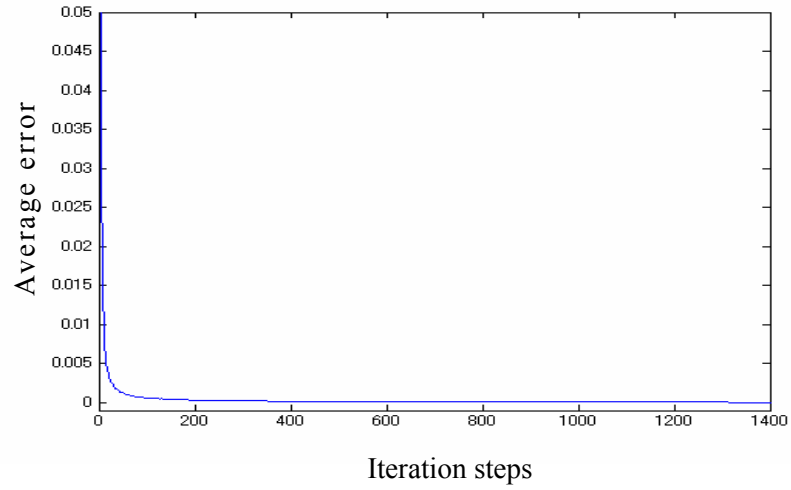
Table 3.5 Rejection ratios of our system with different σ

Function Registrant	Sigmoid	Gaussian ridge function			
		$\sigma = 0.4$	$\sigma = 0.78$	$\sigma = 0.8$	$\sigma = 1.2$
A	60%	68%	70%	72%	86%
B	86%	90%	94%	90%	94%
C	74%	78%	82%	80%	88%
D	56%	62%	78%	76%	82%
E	66%	72%	78%	78%	80%
F	42%	54%	62%	64%	62%
G	52%	60%	64%	62%	62%
H	92%	88%	98%	96%	92%
I	90%	84%	92%	94%	90%
J	60%	70%	82%	80%	84%
K	30%	68%	82%	80%	76%
L	46%	52%	58%	58%	56%
Average	63.2%	70.5%	78.2%	77.5%	79.3%

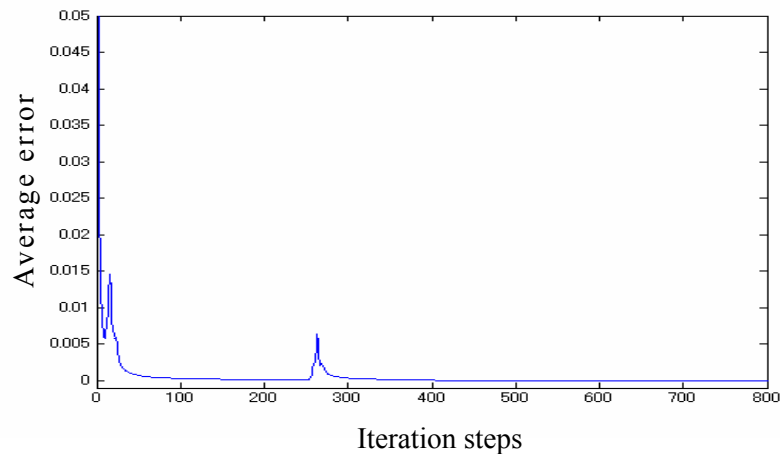
According to Table 3.4 and 3.5, it can be seen that width parameters σ have great influence on both the recognition capabilities and rejection capabilities of the system

with Gaussian ridge function. Moreover, for each registrant, variations of the recognition ratios and rejection ratios with width parameter are not monotonic and the optimal values of σ corresponding to best recognition ratio and rejection ratio respectively are different from each other, therefore the parameter σ should be selected considering both capabilities. Comparing the recognition capabilities and rejection capabilities of all target registrants, it can be seen that the value of width parameter σ corresponding to better performance (recognition and rejection) for each registrant is different from that for other one. It indicates that the width parameter σ should be customized to the neuro-template of each registrant. On the other hand, though the optimal values of σ for each registrant are different from each other, they are approximately located in same range. That provides advantages for customization of width parameter σ for each neuro-template.

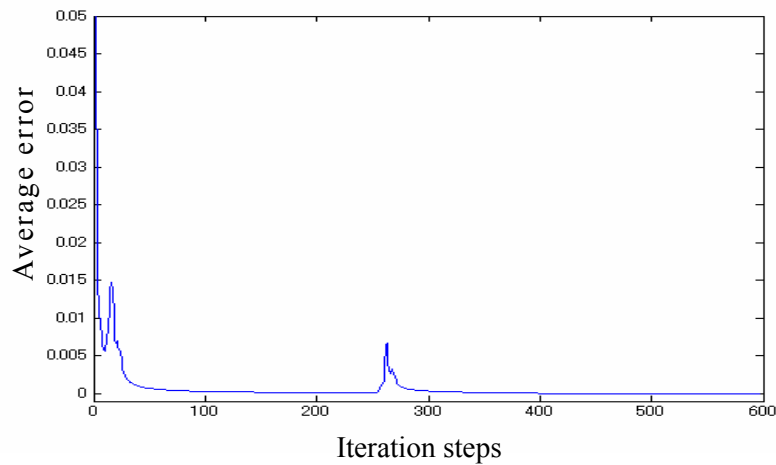
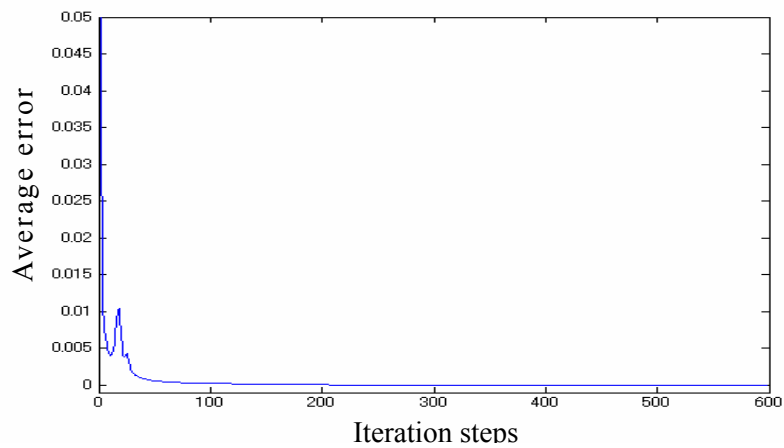
In the next part, the convergence speed of neuro-template learning with different sigma (0.4, 0.78, 0.8, 1.2) are studied and corresponding curves of variation of average error with iteration steps are shown in Figure 3.14. All of curves are obtained with same learning parameter and initial values of weights and identical conditions of neuro-template learning listed in Table 3.1.



(a) $\sigma = 0.4$



(b) $\sigma = 0.78$

(c) $\sigma = 0.8$ (d) $\sigma = 1.2$ Figure 3.14 error curve of system with different σ

According to Figure 3.14, it can be seen that the width parameter σ also affects the convergence speed of neuro-template learning. With an increase in the sigma, the convergence of learning is accelerated. This can be partially explained by the sensitive field of the proposed Gaussian ridge function, which is a ridge-like function as mentioned before, is controlled by the width parameter σ , where a larger σ means a larger sensitive field for the classifying hyper-plane formed by the neuro-template using the proposed Gaussian ridge function, leading to more data satisfying the requirement of the minimal error function in the parameter space. Therefore, it is easy to find a set of qualified parameters in the

searching procedure and high speed of convergence. It should be noticed that, however, high speed of convergence does not mean that the obtained parameter set is global optimal for the system, it is very likely to be local optimal.

3.6.5 Reciprocal Influence of neuro-templates

As mentioned in section four, recruitment of new registrants may lead to relearning of the existing templates. That indicated that neuro-templates were likely to be influenced by other templates registered later. To investigate the mutual influence of different neuro-templates, experiments are conducted on the systems with different sequence of templates. Registrant A and B are selected as target registrants because relearning of template A has been resulted in by later register of template B. However reversing register order, no relearning happened to template B when A registers after B. For convenience consideration, any other template except A and B was not included in the system. Table 3.6 shows the performance of template A under different register order and with different activation function, Table 3.7 shows that of template B.

Table 3.6 Recognition capabilities of our systems with different o

Function	Performance	A only	A, B	B, A
Sigmoid	Recognition	96%	96%	96%
	Rejection	60%	64%	60%
Gaussian ridge	Recognition	92%	92%	92%
	Rejection	70%	82%	70%

Table 3.7 Recognition capabilities of our systems with different o

Function	Performance	b only	B, A	A, B
Sigmoid	Recognition	96%	96%	96%
	Rejection	86%	86%	86%
Gaussian ridge	Recognition	92%	92%	92%
	Rejection	94%	94%	94%

In Table 3.6 and 3.7, ‘A only’ means only template A was included in system, and ‘B only’ means the same. ‘AB’ denotes the registered sequence of registrant A and B, namely registrant A enrolled in system before B, while ‘BA’ means B before A. It can be seen from Table 3.6 and 3.7, even with different activation function, the recognition performance of template A and B were not influenced by register sequence. It indicates that the recognition performance of the system is not affected by increase of templates. According to Table 3.6, no matter which activation function was employed, the rejection capability of template A was enhanced by new register of B and this improvement was especially apparent when Gaussian function was employed. From Table 3.7, the rejection performance of template B under different situation remained untouched because no relearning was caused by later register of A. According to Table 3.6 and 3.7, it can be concluded that the reciprocal influence of inter-template is favorite and helpful for rejecting un-genuine signatures and the neuro-template matching method is effective for our system.

Chapter 4

ANALYSIS ON PERFORMANCE ROBUSTNESS OF THE SYSTEM AND PROPOSAL OF UNIFORMED REGISTER CHARACTERS

4.1 Introduction

In this chapter, we deal with another aspect of our individual identification system in which dynamic pressure of signature is used exclusively as individual feature for personal identification. Different from previous chapter which concentrates on the value of performance of the system, namely the magnitude of recognition capabilities for genuine signatures and rejection capabilities for skilled forgeries, this chapter focus on

the robustness of performance of the system for different customers. From the previous study, it can be seen that though the performances of our system are very promising, the great discrepancy was seen among the recognition capabilities of the system for different registers, namely the recognition capability of the system varies greatly with different user. This fact indicates that stability of the performance of our system is insufficient. Though the performance of our system has been effectively improved by the effort made in reference [24], the performance stability of the system did not get better. To address the issue of insufficient performance stability, we studied the signatures of many people and corresponding recognition ratios, high discrepancy in the complexity of different signature characters was found and its influence on the performance of the system was analyzed. Then the uniformed characters were proposed as register characters instead of traditional personal signature aiming at removing the influence of different register characters on the performance of the system and thereby improving the robustness of performance of our system. To evaluate the feasibility and effectiveness of proposed method, four groups of characters are selected and used as uniformed register characters separately, the corresponding recognition capabilities of the systems for twenty registrants are tested and compared with that of the original signature-based system for same twenty registrants. The comparison results show that uniformity of register characters is helpful to improve the performance stability of the system for different registrants without scarifying the performance itself.

This chapter is arranged as follows. First the problem of insufficient stability in the recognition capabilities of the system for different registrants is presented and studied. In the following part, uniformed register characters is proposed to address this issue. In the last section, the comparative experiments are conducted to evaluate the recognition capabilities of original signature-based systems with three activation functions for twenty registrants and that of the systems based separately on four groups of uniformed register characters and comparison is made between them, the effectiveness of proposed method can be seen from the favorite comparison results.

4.2 Insufficient performance robustness of signature-based system

In the previous research on our system [24], though high recognition capabilities of the system has been shown for some registrants, the recognition capabilities of the system varies greatly with different users. This individual-dependent performance means insufficient robustness of performance of our system, and it is apparently bafflement to the generalization of our system. During the study on the recognition capabilities of original signature-based system for many persons, great discrepancy in the complexity of different signature characters has been observed. Though this discrepancy is helpful for differentiating each registrant from the other, it also leads to the fluctuation in performance of the system. It mainly attributes to that, for the signature-based system, the identification of signer is based on the individual feature extracted from pressure data of the signature. This extracted personal feature is the combination of feature of signer's individual signature habit, which is the real one should be focused on, and the feature of signature characters itself, such as the number of strokes and the structure of the signature characters. When a signer is recognized as a certain registrant, his (or her) signature feature has not only to be different from other registrants' signature features, but also to be in accordance with the signature feature of the registrant whom he (or she) is recognized as. Though high variance in different signature characters made it easier to differentiate each signer from other, it also leads to the different degree of easiness with which signature is identified by the system for different person, namely the performance of the system partly varies with the complexity of signature characters. According to previous analysis, it can be concluded that the difference in the characters of signatures, which are usually the names of registrants, plays an important role to the insufficient robustness of the performance of our system. To address this problem, uniformed characters is proposed as register characters in our individual recognition system instead of traditional employed personal signatures.

4.3 Proposal of uniformed register characters

To remove the influence of different complexity of signature characters on the performance robustness of the system, instead of personal signature, which is usually the user's name, employing a certain group of uniformed characters as register characters is proposed in this research. With uniformed register characters, namely same complexity of register characters, the system will focus only on learning the individual feature of writing habit to recognize each user without disturbance of the different complexity of signature characters. The scheme of this proposal is illustrated by Figure 4.1.

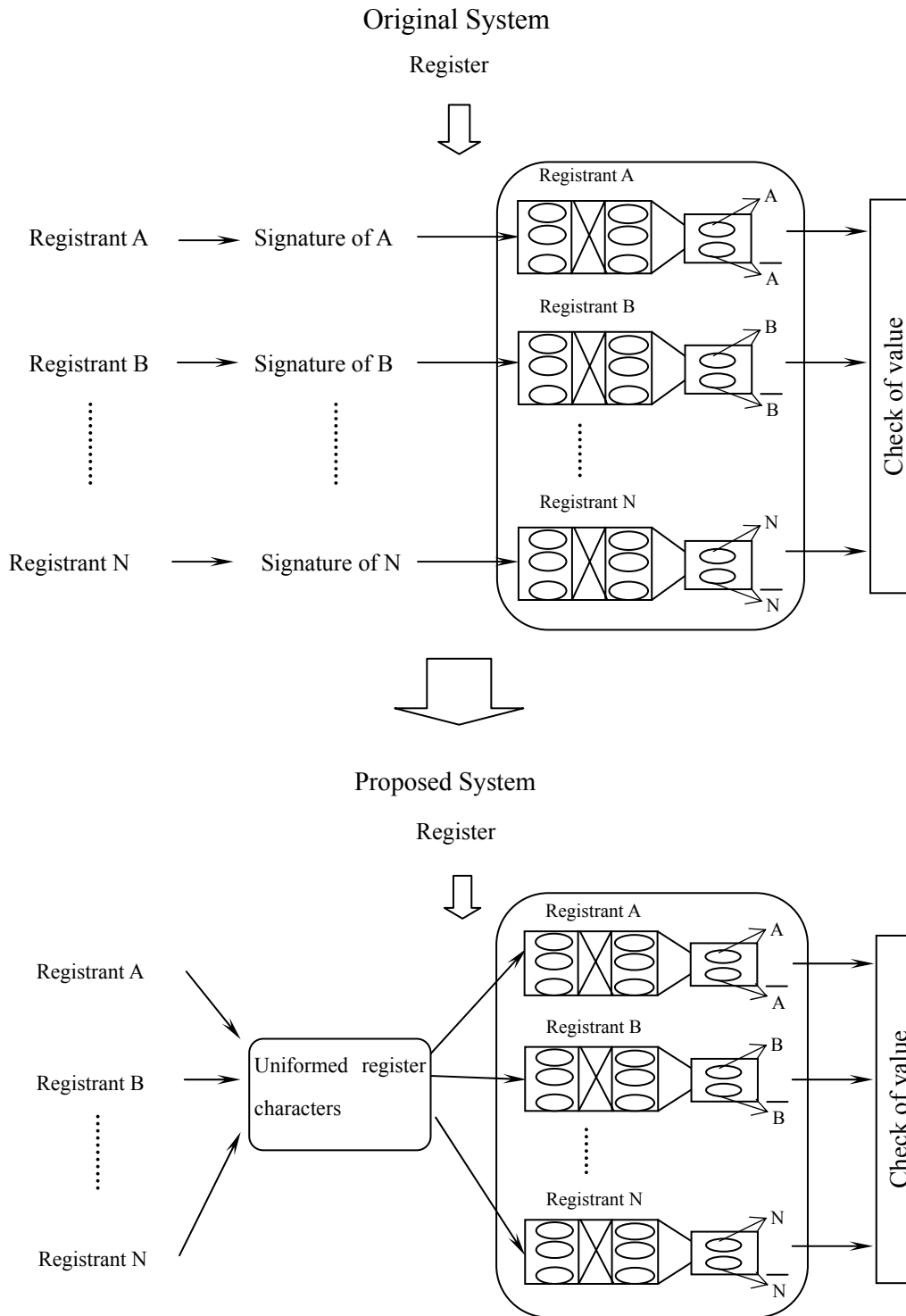


Figure 4.1 Proposal of uniformed register characters

4.4 Experiment

To examine the feasibility of our proposed strategy of uniformed register characters, two experiments are designed and twenty people are selected as experiment subjects. One experiment is evaluating the recognition capabilities of the original signature-based system for selected twenty subjects, namely each subject writes his (or her) own name three times to register on the system and then give more 30 signatures for performance evaluation. The other one is testing the recognition capabilities of the uniformed-register-characters-based systems for same twenty subjects, four groups of characters are chosen as uniformed register characters separately, they are kanji “library”, “takeda lab”, “kochi university”, and “health” respectively. For each group of uniformed characters, each subject practices the given uniformed characters several times, then, registers on the system with three times of the uniformed characters, after that, more 25 to 35 same uniformed characters are provided as test data to evaluate the recognition capability of the system.

The experiment results are shown in following table and figures. In these table and figures, “Sigmoid” means the system with sigmoid as activation function of neurons in hidden layer and output layer of neuro-template, “Gaussian I” means the system with Gaussian as activation function and fix sigma is assigned to each Gaussian function, which was proposed to improve the rejection capabilities of the system for counterfeit signatures. “Gaussian II” means the system is based on customized neural template with optimal sigma for Gaussian activation function, which was proposed to further improve the rejection capabilities of our system for counterfeit signatures[12]. The recognition performance of three systems with different activation function are listed together to investigate the effect of activation function on the performance stability of the system. The recognition performance of the original signature-based systems with different activation function is shown in Table 1. Since studying the fluctuation degree in recognition capabilities of the system is major task in this chapter, the data in Table 1 are expressed in form of Figure 4.2 to give an intuitionistic picture of the distribution state of the recognition capabilities of the systems.

Table 4.1 Recognition capabilities of individual-signature-based systems with three activation functions

Function Registrant	Sigmoid	Gaussian ridge with fixed σ	Gaussian ridge with optimal σ
Registrant 1	86%	86%	86%
Registrant 2	77.4%	80%	80%
Registrant 3	63.3%	63.3%	63.3%
Registrant 4	73.3%	73.3%	73.3%
Registrant 5	83.3%	88.8%	88.8%
Registrant 6	88.9%	91.7%	91.7%
Registrant 7	73.3%	83.3%	83.3%
Registrant 8	96%	92%	96%
Registrant 9	60%	63.3%	63.3%
Registrant 10	88%	100%	100%
Registrant 11	90%	90%	90%
Registrant 12	96%	94%	96%
Registrant 13	90%	84%	90%
Registrant 14	100%	93.3%	93.3%
Registrant 15	93.9%	90.9%	90.9%
Registrant 16	100%	100%	100%
Registrant 17	70%	66.7%	70%
Registrant 18	79.4%	79.4%	79.4%
Registrant 19	92.7%	92.7%	92.7%
Registrant 20	100%	100%	100%
Average	85.08%	85.64%	86.4%

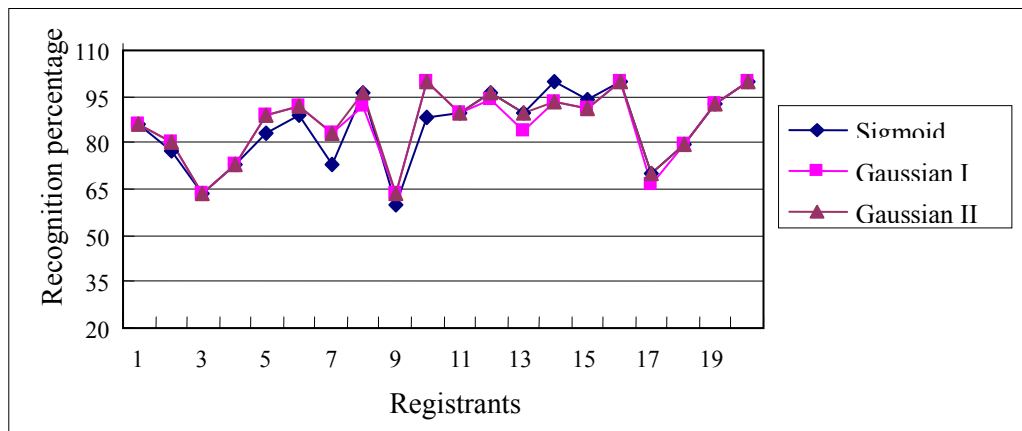


Figure 4.2 Distribution of recognition capabilities of the original signature-based systems

From the Table 1 and Figure 4.2, it can be seen that though the recognition rates of some registrants is very promising (registrant 8, 12, 14, 16, 20), the recognition rates of some registrants (registrant 3, 9, 17), however, are not satisfying. The difference between the best recognition rate and the worst one is significant(40%), and great fluctuation in recognition rates of different registrants can easily been seen from Figure 4.2. It indicates that the robustness of recognition performance in the original signature-based system is very insufficient. On the other side, though the recognition rates of some registrants varied with activation function, the variation is not considerable (especially the average of the recognition capability of the system) and no regularity can be seen (some recognition rates increased while some decreased). Further more, comparing the performance fluctuations of three systems with different activation function, it can be seen that employing which activation function has slight influence on the fluctuation of recognition capability of the system. Therefore it can be included that the employment of Gaussian function and the optimization of parameter sigma, whichever were effective on improving the rejection capabilities of the system against counterfeit signatures, have little influence on the performance stability of the system. This point will be further strengthened in the later experiment results.

The experiment results for uniformed register characters are shown directly with

Figure 4.3, 4.4, 4.5, 4.6 without the corresponding tables to avoid repeat expression and make the comparison of Figure 4.2~4.6 easier.

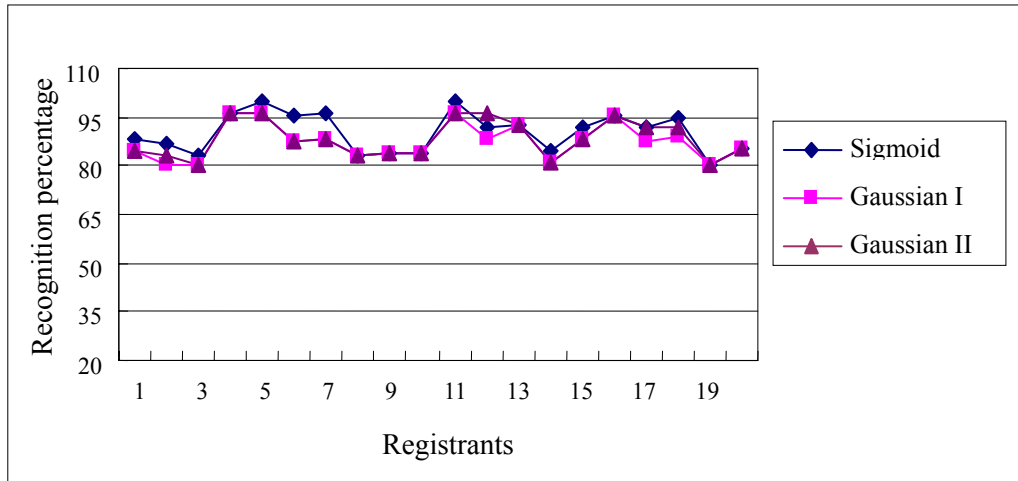


Figure 4.3 Distribution of recognition capabilities of the systems with uniformed characters “library” in Kanji

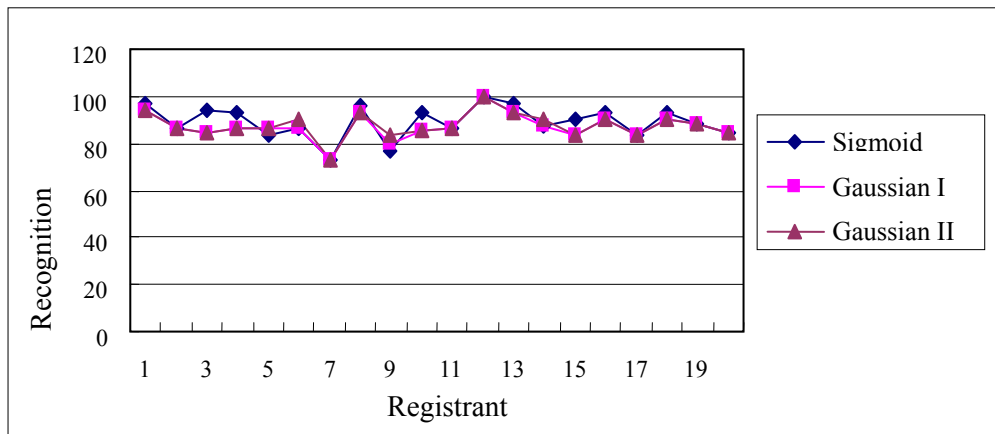


Figure 4.4 Distribution of recognition capabilities of the systems with uniformed characters “Takeda lab” in Kanji

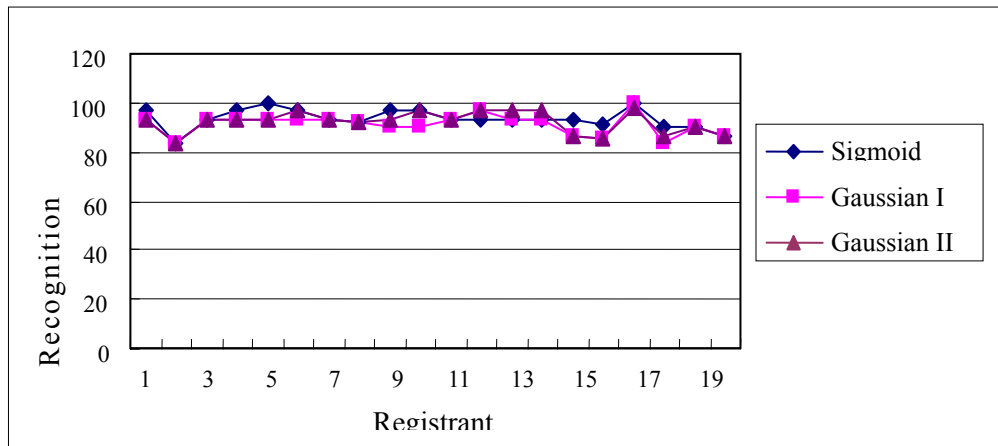


Figure 4.5 Distribution of recognition capabilities of the systems with uniformed characters “Kochi University” in Kanji

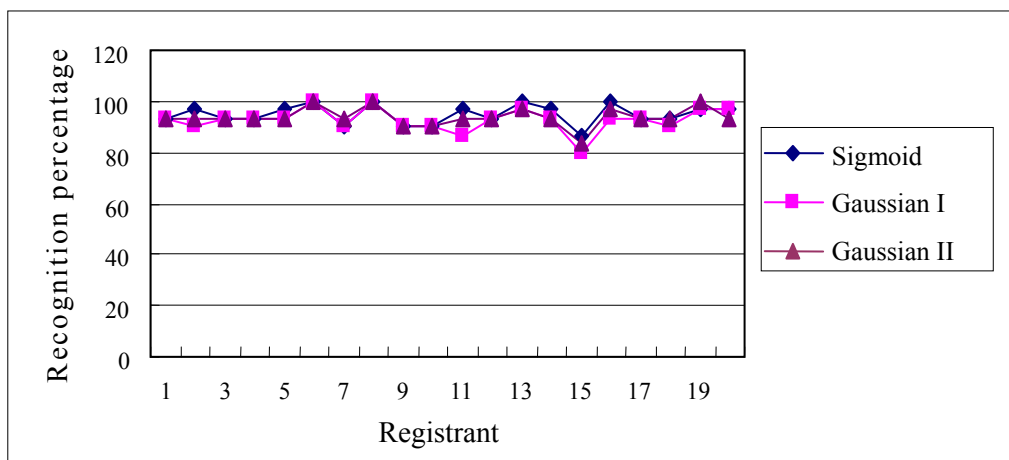


Figure 4.6 Distribution of recognition capabilities of the systems with uniformed characters “health” in Kanji

Comparing Figure 4.2 with Figure 4.3~4.6, it can be seen that no matter which activation function (sigmoid, Gaussian with fix sigma and Gaussian with optimal sigma) is employed, the high fluctuation of recognition capabilities in the original signature-based systems is effectively lowered after uniformed characters is employed

as register characters, though the extent with which the fluctuation is lowered is different for different group of uniformed characters. It shows that uniformity of the register characters seemed to be helpful for improving the robustness of recognition capabilities of the system. At same time, it indicates that the effect of uniformed register characters on the performance robustness of the system is not influenced by activation function of neuro-template. Studying Figure 4.3~4.6, on the one hand, it can be seen that, though being improved greatly, the decentralization degree of the recognition capabilities of the system are a little different for different group of uniformed register characters. It indicates that the stability of the recognition capability of the system is effected by the complexity of register characters to a certain extent. On the other hand, it seems that, the recognition capabilities of the systems do not get worse after employing uniformed register characters. It suggests that the uniformed register characters could be effective in improving the performance robustness of the system without scarifying performance itself. It should be noticed that in Figure 4.2~4.6, no matter what kind of register characters (signature or uniformed characters) and which group of uniformed characters is employed, the fluctuation state in each figure varies slightly with different activation function. Therefore, it can be concluded that the activation function almost has no influence on the performance robustness of the system.

To study the variability of the recognition capabilities of the systems mathematically, the standard deviations of recognition capabilities of the systems with different activation function and register characters for twenty registrants are calculated and shown in following figure.

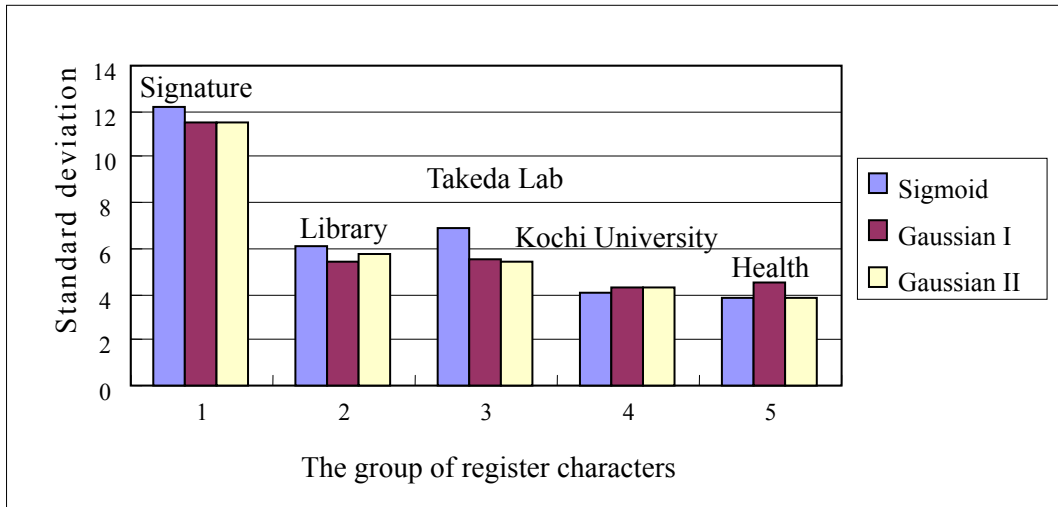


Figure 4.7 Standard deviations of recognition capabilities of the systems with different activation function and register characters

In Figure 4.7, group 1 is standard deviation of recognition capabilities of the signature-based system with different activation function. Group 2, 3, 4, 5 are that of the system with different activation function for four groups of uniformed characters respectively. The employed register characters have been marked on the corresponding bar of each group. Comparing group 1 with group 2, 3, 4 and 5 in Figure 4.7, it can be seen that no matter which activation function (sigmoid, Gaussian with fix sigma or that with optimal sigma) is employed, the standard deviation of recognition capabilities of the system is greatly decreased (from more than 11.5 to less than 7) after uniformed characters are employed as register characters. It shows that uniformity of register characters is effective on reducing the extent of dispersal of recognition capabilities of the system, namely improving the robustness of recognition performance of the system. This effectiveness is attributed to same complexity that the uniformed characters have. Comparing the group 2, 3, 4 and 5, in which different uniformed register characters are used, slight variance can be seen in the standard deviations of four groups, it demonstrates again that robustness of recognition capabilities of the system is affected by the complexity of register characters. This suggests that with properly selected uniformed register characters, the recognition performance of our system could obtain better robustness. Moreover, studying each group in Figure 4.7, it can be seen that the

employment of Gaussian function and optimization of sigma parameter have little influence on the robustness of recognition capabilities of the system.

To give better understanding on the fact that the recognition performance are not undermined by the employment of uniformed register characters, the average of recognition capabilities of the systems with different activation function and different register characters are calculated and shown in Figure 4.8.

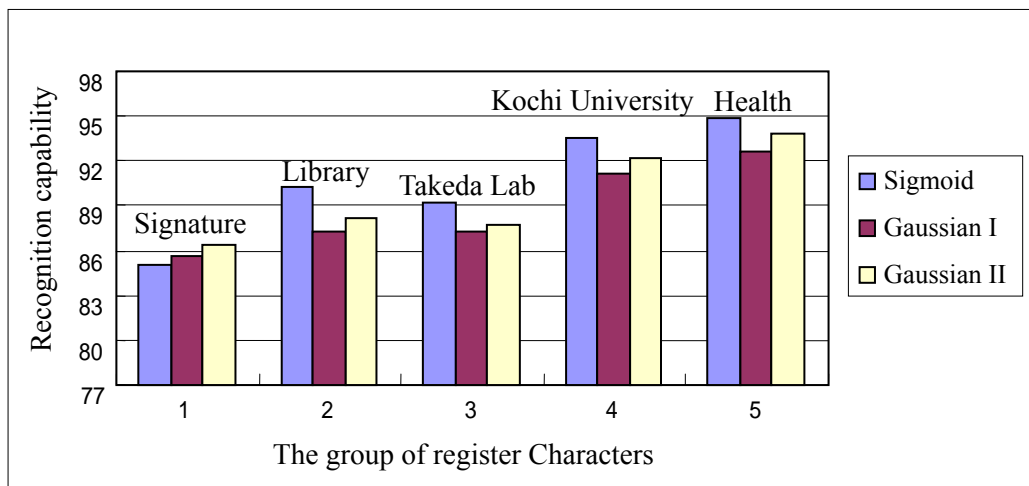


Figure 4.8 Average of recognition capabilities of the systems with different activation function and register characters

In Figure 4.8, group 1 is average of recognition capabilities of the signature-based systems with different activation function. Group 2, 3, 4, 5 are that of the systems with different activation function for four groups of uniformed characters respectively and the corresponding uniformed register characters have been marked on the bar of each group. Comparing group 1 with group 2, 3, 4 and 5 in Figure 4.8, it is easily to see that whichever activation function (sigmoid, Gaussian with fix sigma or that with optimal sigma) is used, the average of recognition capabilities of the system does not decreased with employment of uniformed register characters, contrarily it is enhanced greatly for the uniformed characters of ‘Kochi university’ and ‘health’. Combined with Figure 4.7, it can be conclude that employing uniformed register characters seems to be effective in improving the robustness of recognition capabilities of the system without scarifying the

recognition performance itself. Moreover, it suggests that with proper uniformed register characters, not only the performance stability of the system could be improved, but also better performance of the system could be obtained.

In a word, according to all the previous experiment results, it can be concluded that using uniformed characters, instead of personal signature, as register characters is effective in improving the robustness of recognition capabilities of our system, and at same time keeping the recognition performance of the system not deteriorate.

Chapter 5

CONCLUSIONS

In this research, a signature-pressure-based individual identification system aiming to internet application was introduced firstly. In this system, neuro-template, which comes from the combination of technique of neural networks and template matching method, was employed as classifier. to address the problem of poor rejection capabilities of the

original system, in which traditional sigmoid function was utilized as activation function of neuro-template, for the counterfeit signatures at same time ensuring the recognition capabilities for genuine signatures, a kind of Gaussian ridge function was proposed as activation function of each neuron in the hidden layer and the output layer of each neuro-template. Then the characteristics of proposed Gaussian ridge function, which is a kind of semi-localized function, are analyzed. This function extends to infinite along the direction of the ridge of function and closed in the orthogonal direction of the ridge. The output of the function approaches one as the distance to its center ridge approaches zero. Its sensitive field is governed by the width parameter. The characters of proposed Gaussian ridge function, which contributed to the potentiality of improving rejection capabilities of the original system for counterfeit signatures, was analyzed and superiority of neural network with proposed Gaussian ridge function in classifying the recognition space was concluded by comparing with that based on conventional sigmoid function. Moreover the difference between the proposed Gaussian ridge function and simple radial basis function and the Gaussian bars function were explained clearly. All of that showed the advantages of the system with proposed Gaussian ridge function on rejecting the counterfeit signatures owing to the distribution features of the proposed Gaussian function in multi-dimensional space.

Though the proposed Gaussian ridge function with fixed width parameter σ effectively improved the rejection capabilities of the system for counterfeit signatures, the extent of improvement was not great enough. Moreover the employment of proposed Gaussian ridge function also led to slight deterioration of recognition capabilities of the system for genuine signatures and it was unexpected. Therefore, width parameter σ was optimized for each neuro-template for further improving the rejection capabilities of the system for counterfeit signatures and recovering the deteriorated recognition capabilities of the system for genuine signatures. The experiment results suggested that the individual identification system proposed in this paper was very promising. The proposed neuro-template classifier with Gaussian ridge function was effective on improving the rejection capabilities of the system. Furthermore optimization of width parameter σ for each neuro-template with Gaussian

ridge function seemed to have high efficiency of further improving the rejection capabilities of our system for skilled forgeries and at same time ensuring the recognition performance satisfied.

In the chapter 2, the structures of the system including hardware and software were depicted firstly. By analyzing the pressure data curves of people's signature, the feasibility and reliability signature pressure to be utilized as character signal of recognizing personal identity were demonstrated. In the following section, the preprocessing of the pressure data of signatures was described in detail. It showed that the adaptive enrollment of register signature and validity check ensured the quality of sampled register signatures. In the procedure of preprocessing, moving average method was used to compress the scale of raw pressure data and remove noise data, statistical distribution analyses was employed to extracting valid information from the signature pressure data. After preprocessing, the data scale was greatly reduced from more than 1000 to 50 and personal features embedded in pressure data of signatures are also extracted.

In the section 3.3.1, neuro-template matching method is introduced, and it showed effectiveness of simplifying the registration procedure and removing the limitation on the number of registrants, which is a common problem of traditional multilayer neural network classifier, owing to the relative independence of neuro-templates. Furthermore according to the studies on the reciprocal influence of neuro-templates in the experiment section of chapter 3, the favorite influence of inter-template can be seen. It demonstrated again the effectiveness of neuro-template matching scheme in our individual identification system.

In the experiment section of chapter 3, the influence of parameter σ on the performance of our system and convergence speed of neuro-template learning was studied. The results showed that parameter σ had great influence on both recognition capabilities and rejection capabilities of the system based on proposed Gaussian ridge function, and the approximate range of optimal parameter sigma in which best performance for each registrant' neuro-template was also indicated by the results. That

provided the illumination of customization of width parameter σ for each neuro-template in next stage of study.

In the last study, in order to decrease the high discrepancy in the recognition capabilities of the original signature-based system for different registrants and improve performance robustness of the system, uniformed characters were proposed as register characters in this paper. The experiment results showed that the uniformity of register characters seemed to be effective to lower the fluctuation of the recognition capabilities of the system for different registrants and improve the performance robustness of our system without sacrificing the recognition performance itself. Furthermore, the discrepancy in the robustness of recognition capabilities of the systems with different uniformed register characters shown that the performance stability of system was influenced by the complexity of uniformed characters, it suggested that the uniformed characters should be carefully selected to get better robustness of the recognition performance of the system and better recognition performance. Last, it seemed that the robustness of our system was almost not affected by the employment of Gaussian function with fix sigma and optimization of parameter sigma for Gaussian function, which were proposed to improve the rejection capabilities of the system for counterfeit signatures in previous study.

REFERENCES

- [1] R. Plamondon, G. Lorette: Automatic Signature verification and Writer Identification -- the State of Art; *Pattern Recognition*, Pergamon Press plc, Vol. 22, pp. 107-131(1989)
- [2] R. Plamondon, F. leclerc: Automatic Signature Verification : the State of Art - 1989-1993; *International Journal of Pattern Recognition and Artificial Intelligence*, Vol. 8, pp. 643--660 (1994)
- [3] C. Vielhauer, R. Steinmetz, and A. Mayerhofer: Transitivity Based Enrollment Strategy for Signature Verification Systems; *Proceedings of the Sixth International Conference on Documents Analysis and Recognition*, pp. 1263-1266 (2001)
- [4] J.P. Drouhard, R. Sabourin and M. Godbout: Evaluation of a Training Method and of Various Rejection Criteria for a Neural Network Classifier Used for Off-line Signature Verification; *1994 IEEE World Congress on Computational Intelligence*, pp. 4294-4299 (1994).
- [5] H. Cardot, M. Revenu: A Static Signature Verification System Based on a Cooperating Neural Networks Architecture; *International Journal of Pattern Recognition and Artificial Intelligence*, Vol. 8, No. 3 pp. 679-692 (1994)

- [6] B. Herbst, H. Coetzer: On an Off-line Signature Verification System; *Proceedings of the 9th Annual South African Workshop on Pattern Recognition*, pp. 39-43 (1994)
- [7] F.R. Rioja, M.N. Miyatake, Hector Perez M. and Karina Toscano M.: Dynamics Features Extraction for On-line Signature Verificaiton; *Proceedings of the 14th International Conference on Electronics, Communication and Computers*, pp. 156-161 (2004)
- [8] K. Tanabe, M.Yoshihara, H. Kameya and S. Mori: Automatic Signature Verification Based on the Dynamic Feature of Pressure; *Proceedings of the Sixth International Conference on the Documents Analysis and Recognition*, pp. 1045-1049 (2001)
- [9] H. D. Crane, J. S. Ostrem: Automatic Signature Verification Using a Three-axis Force-sensitive Pen; *IEEE Trans on Sys-tem, Man, and Cybernetics*, Vol. SMC-13, pp. 329-338 (1983)
- [10] M. Kikuchi, N. Akamatsu: Development of Speedy and High Sensitive Pen System for Writing Pressure and Writer Identification; *Proceedings of the Sixth International Conference on the Documents Analysis and Recognition*, pp. 1040-1044 (2001)
- [11] W. Nelson, E. Kishon: Use of Dynamic Feature for Signature Verification; *Proceedings of 1999 International Conference on Systems, Man, and Cybernetics*, pp.201-205 (1991)
- [12] C. Schmidt, K.-F. Kraiss: Establishment of Personal Templates for Automatic Signature Verification; *Proceedings of 1997 International Conference on Documents Analysis and Recognition*, pp. 263-267 (1997)
- [13] C.C. Han, P.C. Chang: An On-line Signature Verification System Using Multi-template Matching Approaches; *IEEE International Carnahan Conference on Security Technology*, pp. 477-480 (1999)

- [14] T. Wessels and C.W. Omlin: A Hybrid System for Signature Verification; *International Joint Conference on Neural Networks*, Vol. 5, pp. 509-514 (2000)
- [15] C.N. Liu: Reference Design Procedure for Signature Verification; *IBM tech. Disclos. Bull.21*, pp. 426-429 (1978)
- [16] Takeda, F. and Omatu, S.: High Speed Paper Currency Recognition by Neural Networks; *IEEE Transaction on Neural Networks*, Vol. 6, No. 1, pp. 73-77 (1995)
- [17] Takeda, F. and Omatu, S.: A Neuro-system Technology for Bank Note Recognition; *Proceedings of the Japan/USA Symposium on Flexible Automation*, Vol. 2, pp. 1511-1516 (1998)
- [18] Takeda, F. and Omatu S.: Neural Network Systems Technique and Applications in Paper Currency Recognition Neural Networks Systems; *Technique and Applications*, Vol. 5, Chapter 4, pp. 133-160 (1998)
- [19] Takeda, F. Omatu, S. and Nishikegee, Y.: A Neuro-recognition Technology for Paper Currency using Optimized Masks by GA and its Hardwar; *Proceedings of the International Conference on Information System Analysis and Synthesis*, pp. 147-152 (1996)
- [20] Takeda, F., Nishikage, T. and Matsumoto, Y.: Characteristic Extraction of Paper Currency Using Symmetrical Masks Optimized by GA and Neuro-recognition of Multi-national Paper Currency; *Proceedings of IEEE World Congress on Computational Intelligence*, Vol. 1, pp. 634-639 (1998)
- [21] Stephen, V.Crowder: A Simple Method for Studying Run-Length Distribution of Exponentially Weighted Moving Average Charts; *Technometrics*, Vol. 29, No. 4, pp. 401-407 (1987)
- [22] Fnakubo, M.: Pattern Recognition, *Japan, Kyouritsu Press*, pp. 15-25 (1991)
- [23] Hetch-Nielsen, R., *Neurocomputing*, Addison-Wesley, Reading, MA, 1990

- [24] Lina Mi, Fumiaki Takeda: A Writing Pressure Individual Recognition System Using Customized Neuro-template, *Proceeding of International Conference of SICE 2005*, pp. 606-610, Okayama, Japan, Aug. 8, 2005
- [25] C. M. Bishop, *Neural Network for Pattern Recognition*, Oxford University Press, New York, U.S.A.(1995)
- [26] Baiqing Sun, Fumiaki Takeda: Application of a Neural Network with Gaussian Function on Currency Recognition System, *Proceedings of SICE annual conference 2005*, Japan, pp601-605 (2005)
- [27] Pierre Baldi: Gradient Descent Learning Algorithm Overview: A General Dynamical Systems Perspective, *IEEE Trans. on Neural networks*, Vol. 6, NO. 1, pp182-195 (1995).
- [28] Waymaere N., et al.: On the Initialization and Optimization of Multilayer Perceptrons, *IEEE Trans. on Neural Networks*, Vol.5, No.5, pp.738-751 (1994)
- [29] Terrence L. Fine: *Feedforward Neural Network Methodology*, Springer Press, New York, 1999
- [30] Gregory A. Baxes: *Digital Image Processing Principles and Application*, John Wiley & Sons, Inc. 1994
- [31] Cornelius T. Leondes: *Image Processing and Pattern recognition*, Academic Press, San Diego, 1998
- [32] Andrew R. Webb: *Statistical Pattern Recognition (Second Edition)*, John Wiley & Sons, LTI, 2002
- [33] Zhi-Quan Luo: On the Convergence of the LMS Algorithm with Adaptive Learning Rate for Linear Feedforward Networks, *Neural Computation*, Vol. 3, NO. 2, pp.226-245 (1991)

- [34] E. J. Hartman and J. D. Keeler: Predicting the Future. Advantages of Semilocal Units, *Neural Computation*, Vol. 3, NO. 4, pp566-578 (1991)
- [35] Park J., et al.: Approximations and RBF Networks. *Neural Computation*, Vol. 5, pp305-316 (1993)
- [36] Park J., et al.: Universal Approximations Using RBF Networks. *Neural Computation*, Vol. 3, pp246-257 (1991)
- [37] Mirosław K., Włodzisław D.: A Survey of Factors Influencing MLP Error Surface, *Control and Cybernetics*
- [38] Park J., et al.: Approximation and Radial Basis function Networks. *Neural Computation*, Vol. 5, pp305-316 (1993)
- [39] Bors A. G., et al.: Median Radial Basis Function neural network, *IEEE Trans. on Neural Networks*, Vol.7, No.6, pp.1351-1365 (1996)
- [40] Elanayar S., et al.: Radial Basis Function Neural Network for Approximation and Estimation of Nonlinear Stochastic Dynamic Systems. *IEEE Trans. on Neural Networks*, Vol.4, No.5, pp.594-604 (1994)
- [41] Barron A. R.: Universal Approximation Bounds for Superpositions of a Sigmoidal Function, *IEEE Trans. on Information Theory*, Vol.39, No.3, pp.930-945 (1993)
- [43] Sanger T. D.: Optimal Unsupervised Learning in a Single-layer Feedforward Neural Network, *Neural Networks*, Vol.1, pp.459-473 (1989)
- [44] Bianchini, M., et al.: On the problem of local minima in recurrent neural networks, *IEEE Trans. on Neural Networks*, Vol.5, No.2, pp.738-751 (1994)
- [45] Corwin E.M., et al.: An Iterative Method for Training multiplayer networks with threshold functions, *IEEE Trans. on Neural Networks*, Vol.5, No.3, pp.507-508 (1994)

ACKNOWLEDGEMENTS

I would like to express my greatest appreciations to my supervisor Prof. Fumiaki Takeda for his careful guidance, continuous encouragements and serious supervising in the past three years. I profit significantly from his profound special knowledge and abundant practical experiences.

I also would like to express my sincere appreciations to leadership and colleagues of Shenyang University of Technology who gave many supports to me in last three years.

I will appreciate research assistants Nishikage Toshihiro, and Shirashi Yuhki, doctor Baiqing Sun for their beneficial discussions and helps on my research. I also appreciate Satoh Hironobu and Saeki Yoshihiro for his many kind helps.

I would like to thank secretaries Yoshida Naka and Yamanaka Miyuki for their helps and warm heart. I also appreciate all members of Takeda Laboratory. Just because of them I spend three unforgettable and happy years.

Especially, I will appreciate to my family sincerely for their encouragements and support in my three years of study.

I will appreciate Kochi University of Technology for providing me with excellent research environment to complete my doctor research smoothly.

PUBLICATIONS

Journal Papers

1. Lina Mi, Fumiaki Takeda, “Analysis on the Robustness of the Pressure-based Individual Identification System Based on Neural Network”, *International Journal of Innovative Computing, Information and Control (IJICIC)* for the special issue, “Innovative Computing Methods of Management Engineering”, Vol.3, No.1, February 2007.

2. Lina Mi, Fumiaki Takeda, Signature-pressure-based Individual Identification System and Improvement on the Rejection Capability of the System Using Customized Neuro-templates with Gaussian Function, submitted to KES Journal, is under review.

International Conference Papers

1. Lina Mi, Fumiaki Takeda, “Improvement on the individual recognition system with writing pressure based on RBF”, Proceedings of *International Conference of CEIS'2005*, Miami, USA, May, 2005.

2. Lina Mi, Fumiaki Takeda, “A Writing Pressure Individual Recognition System Using Customized Neuro-template”, Proceeding of *International Conference of SICE 2005*, pp. 606-610, Okayama, Japan, Aug. 8, 2005

3. Lina Mi, Fumiaki Takeda, “Research on Individual Recognition System with Writing Pressure Based on Customized Neuro-template with Gaussian Function”, Proceeding of *International Conference of KES 2005*, pp. 263-269, Melbourne, Australia, Sep. 2005.

Domestic Conference Papers

1. Lina Mi, Fumiaki Takeda, “Research on the Application of Individual Recognition System with Writing Pressure Based on RBF”, *Proceeding of the Fifth Conference of SICE on System Simulation*, pp. 1035-1036, Tsukuba, Japan, Dec. 2004.