

# Wavelet-based Off-line Signature Verification <sup>1</sup>

**Peter Shaohua Deng<sup>†</sup>, Hong-Yuan Mark Liao<sup>‡, 2</sup>, Chin Wen Ho<sup>†</sup>, and Hsiao-Rong Tyan<sup>\*</sup>**

<sup>†</sup> Institute of Computer Science and Information Engineering,  
National Central University, Chung-Li, Taoyuan, Taiwan

<sup>\*</sup> Institute of Computer Science and Information Engineering,  
Chung Yuan Christian University, Chung-Li, Taoyuan, Taiwan

<sup>‡</sup> Institute of Information Science,  
Academia, Sinica, Nankang, Taipei, Taiwan

In this paper, a wavelet-based off-line signature verification system is proposed. The proposed system can automatically identify useful and common features which consistently exist within different signatures of the same person and, based on these features, verify whether a signature is a forgery or not. The system starts with a closed-contour tracing algorithm. The curvature data of the traced closed contours are decomposed into multiresolutional signals using wavelet transforms. Then the zero-crossings corresponding to the curvature data are extracted as features for matching. Moreover, a statistical measurement is devised to decide systematically which closed contours and their associated frequency data of a writer are most stable and discriminating. Based on these data, the optimal threshold value which controls the accuracy of the feature extraction process is calculated. The proposed approach can be applied to both on-line and off-line signature verification systems. Experimental results show that the average success rates for English signatures and Chinese signatures are 91.71% and 93%, respectively.

---

<sup>1</sup>This work was partially supported by the National Science Council of Taiwan under grant NSC86-2745-E-001-004.

<sup>2</sup>To whom correspondence should be addressed. E-mail: liao@iis.sinica.edu.tw

## 1 Introduction

Handwritten signature verification has been extensively studied in this decade[1–16]. Its many applications include banking, credit card validation, security systems etc. In general, handwritten signature verification can be categorized into two kinds – on–line verification and off–line verification. On–line verification requires a stylus and an electronic tablet connected to a computer to grab dynamic signature information[10–12][16]. Off–line verification, on the other hand, deals with signature information which is in a static format[13–15]. Since the on–line approach can acquire more information than the off–line one, the latter is certainly more difficult to deal with. In this paper, we shall focus on the off–line signature verification problem.

The development of computer–aided handwritten signature verification systems has been ongoing for decades. In [4], Plamondon and Lorette provided a thorough survey of automatic handwritten signature verification and writer identification. Their survey covered both on–line and off–line approaches, and especially focused on preprocessing techniques, feature extraction methods, comparison processes and performance evaluation. Later, in 1993, Brault and Plamondon[6] introduced a complexity measure into handwritten curves to dynamically model signature forgery. Chen[9] studied handwritten signature verification from the viewpoint of forensic science. He categorized test handwritten signatures into different types: genuine, disguised, copying with/without observation, tracing, forgery by mental impression/transfer, and forgery by added–on strokes. Ammar *et al.*[8] proposed parametric and reference pattern based features to verify skillful simulated handwritten signatures. In [5], Parizeau and Plamondon reported a comparative study of three different matching algorithms in the context of signature verification. They concluded that no algorithm consistently out–performed others in all circumstances. Recently, off–line handwritten signature verification has been extensively explored[1–9]. However, few studies have taken advantage of curvature information[2] as well as multiresolution techniques[1][7]. In [2], Lee and Pan tried to use a stroke skeleton tracing approach to extract dynamic stroke information, but the extracted strokes were too sensitive to small intra–personal variations. Qi and Hunt[1] adopted a multiresolution approach for off–line signature verification. They only made use of lowpass data to extract some geometrical and statistical features. Highpass data, which is believed to have discriminating power, was not used at all. Wen *et al.*[7] applied a wavelet–based approach to delete trace forgery in an on–line handwritten Chinese signature system. In an off–line system, their approach can not be applied.

Since our aim is to develop an efficient off-line handwritten signature verification system, some promising techniques which can be applied to on-line systems cannot be adopted here. A major application of an off-line signature verification system is check or money-order forgery verification. In this kind of application, the system has to identify whether the signature on a check/money-order is genuine or not. The objective of this research is to identify some useful and common features which consistently exist within different signatures of the same person and, based on these features, to verify whether a signature is a forgery or not. We focus on observing the curvature change of the preprocessed signals. Since the curvature change itself is very sensitive to noise or minor variations, the multiresolution scheme must be introduced under these circumstances. It is well known that highpass data contain sharp curvature features as well as a high noise level. Usually, they possess a sensitive but unstable discriminating capability. Low-pass data, on the other hand, contain coarse curvature information but a relatively low noise level. This makes lowpass data less sensitive but enables it to possess stable discriminating power. To decompose a curvature-based signature into a multiresolution format, wavelet theory[17–18] is introduced. Wavelet theory has broad applications in image analysis[19–22]. In order to build an efficient off-line handwritten signature verification system, we propose a new representation scheme for a signature and then use wavelet transforms to decompose and analyze the transformed signals. In the first stage of the system, closed-contour tracing is applied to a preprocessed signature image. For every boundary point, the algorithm records the  $x$  and  $y$  coordinates, respectively, and the tangential angle. The three recorded pieces of one-dimensional data are transformed into another space using Mallat's[18] dyadic wavelet transform. From these multiresolution data, the desired features can be stably extracted at different resolutions. A statistical measurement has been devised to decide systematically which closed contours and their associated frequency data are most stable and discriminating. Based on these data, the optimal threshold value, which controls the accuracy of the feature extraction process, can be calculated. The features extracted from different signatures can then be compared based on a dissimilarity measure. To calculate the dissimilarity measure, the dynamic time warping[5][23][24] technique is used. One thing worth noting is that the above mentioned signature verification method can be applied to different languages, including English and Chinese. Moreover, it can be applied to either on-line or off-line verification systems. Experimental results show that the success rates for English signatures and Chinese signatures are 91.71% and 93%, respectively.

The organization of the remainder of this paper is as follows. Section 2 describes the preprocessing and feature extraction process. Section 3 introduces the dynamic time warping technique, which is necessary to calculate the dissimilarity measure. We will also explain how a specially devised statistical measurement can be used to calculate the writer-dependent optimal threshold value. Database construction and the verification algorithm are described in Section 4. Section 5 demonstrates the efficiency of our approach based on the results of a series of experiments. Conclusions and future work are described in Section 6.

## 2 Preprocessing and Feature Extraction

### 2.1 Preprocessing

To a raw signature image, some preprocessing steps have to be applied. First, we assume that each signature is written on a white piece of paper of fixed size (say a  $5\text{ cm} \times 10\text{ cm}$  rectangular space on a white sheet of paper). In our experiments, the signatures were acquired by a MicroTeck UMAC U1260 image scanner set at 600 pixels per inch. The signature is then processed with binarization and noise removal. The first row of Figure 1 shows three binarized signature images. In general, signatures signed by the same writer may vary among different trials. The circled part of the signature image illustrated in Figure 2(a) contains a gap which may not exist in another trial. In order to compensate for the variations resulting from different trials, we propose a  $3 \times 3$  morphological mask to dilate along all the contours of a signature image. Figure 2(b) shows the dilated result. It is obvious that the gap which originally existed in the signature is now filled up. This outcome indicates that a person's signature patterns can be represented using fewer reference templates by applying this technique. The second row of Figure 1 shows three more examples of dilated results.

After the morphological dilation process, all the closed contours are traced counter-clockwise from every contour's northwest corner. The third row of Figure 1 shows all the extracted closed contours. For every point on a closed contour, up to three items are recorded when the tracing process is executed. They are: (1) the  $x$ -coordinate, (2) the  $y$ -coordinate, and (3) its tangential angle. Moreover, the longer closed contours of a signature which exceed a threshold value are marked simultaneously. The bottom row of Figure 1 shows the longer closed contours extracted, respectively, from the three different signatures. In

what follows, we shall describe how features are extracted from a signature image.

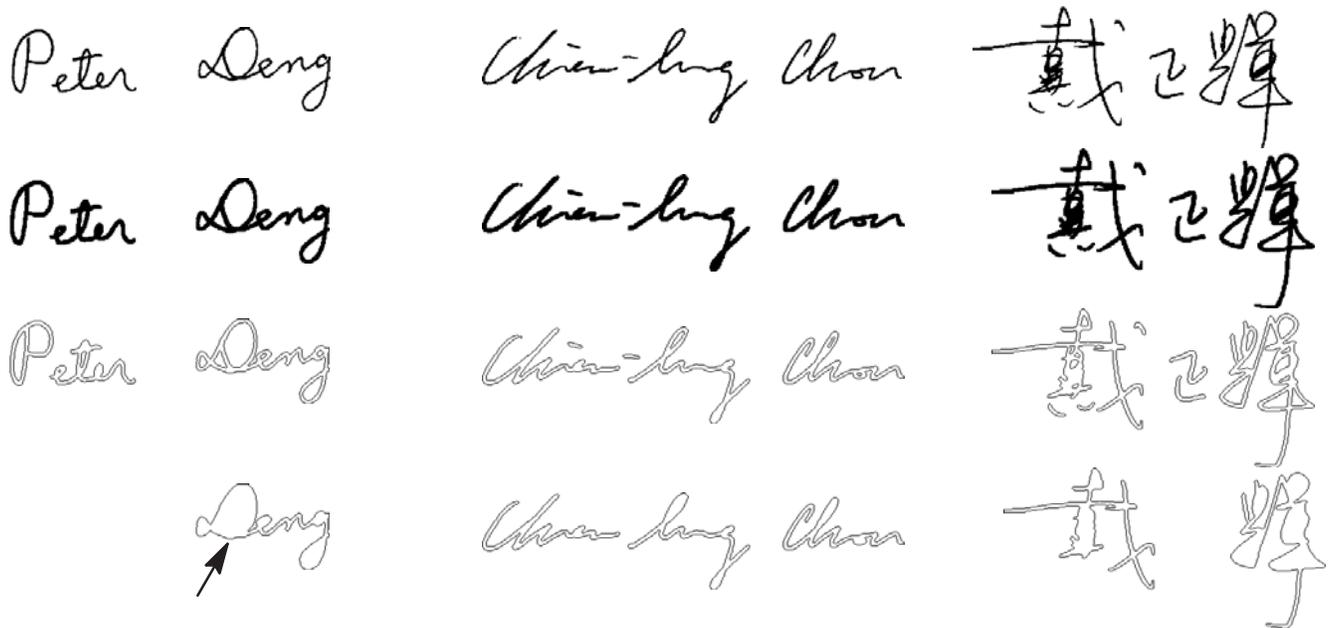


Figure 1. Illustrations demonstrating the preprocessing steps, where in each column, there are 4 images from top to bottom. They represent, respectively, the binarized, morphologically dilated, closed-contour extracted and the optimal larger closed-contour(s) of the original signature images.

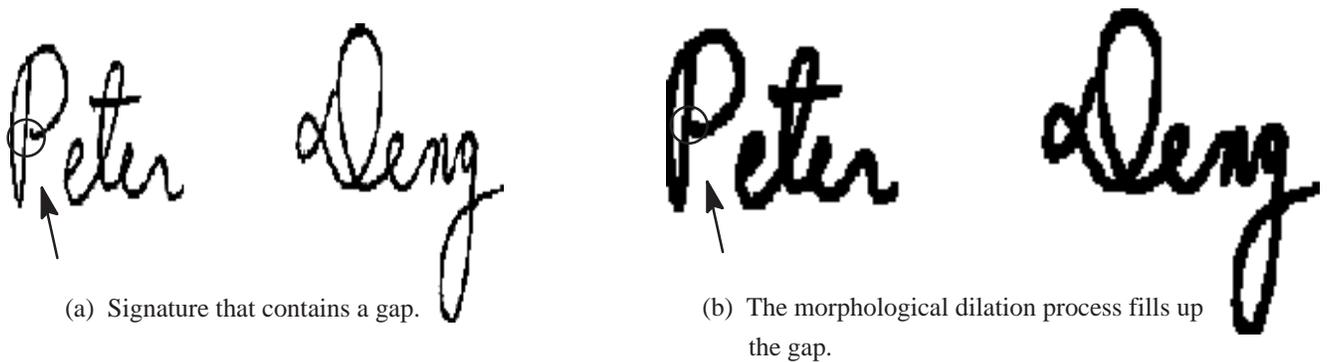


Figure 2.

## 2.2 Feature Extraction

Since we have recorded three one-dimensional signal data from a signature image, the next step is to extract features from these data and use them as the bases for signature verification. Basically, the features to be extracted should be stable and should retain the characteristics of the original pattern. At this stage, we use Mallat's[18] discrete dyadic orthogonal wavelet transform to perform feature extraction. In general,

the multiresolution wavelet transform can decompose a signal into lowpass and highpass information[17][18]. The lowpass (i.e. low frequency) information represents the main body of the original data while the highpass (i.e. high frequency) information usually represents features that contain sharper variations. Since the mother wavelet used here is the second derivative of a smoothing function, the zero-crossings of the transformed data naturally indicate sharper variation points. For the preprocessed data, we extract the following information:

1. the total number of closed contours in a signature image, and
2. the zero-crossings in the highpass data of the three pieces of one-dimensional, transformed signal data. For every zero-crossing point, three attributes associated with the zero-crossing point will be extracted. They are:
  - (1) the abscissa of the zero-crossing[25],
  - (2) the left-hand side wavelet integral between the current zero-crossing and the previous one[18],
  - (3) the corresponding amplitude with the same abscissa in the lowpass data one resolution up.

Figure 3 is a typical example showing how the second attribute is calculated. Let  $z_7$  be the zero-crossing point under consideration. Its abscissa is close to 330. The second attribute associated with  $z_7$  is the wavelet integral bounded by  $z_7$  and  $z_6$  (i.e., the area of  $a_7$ ).

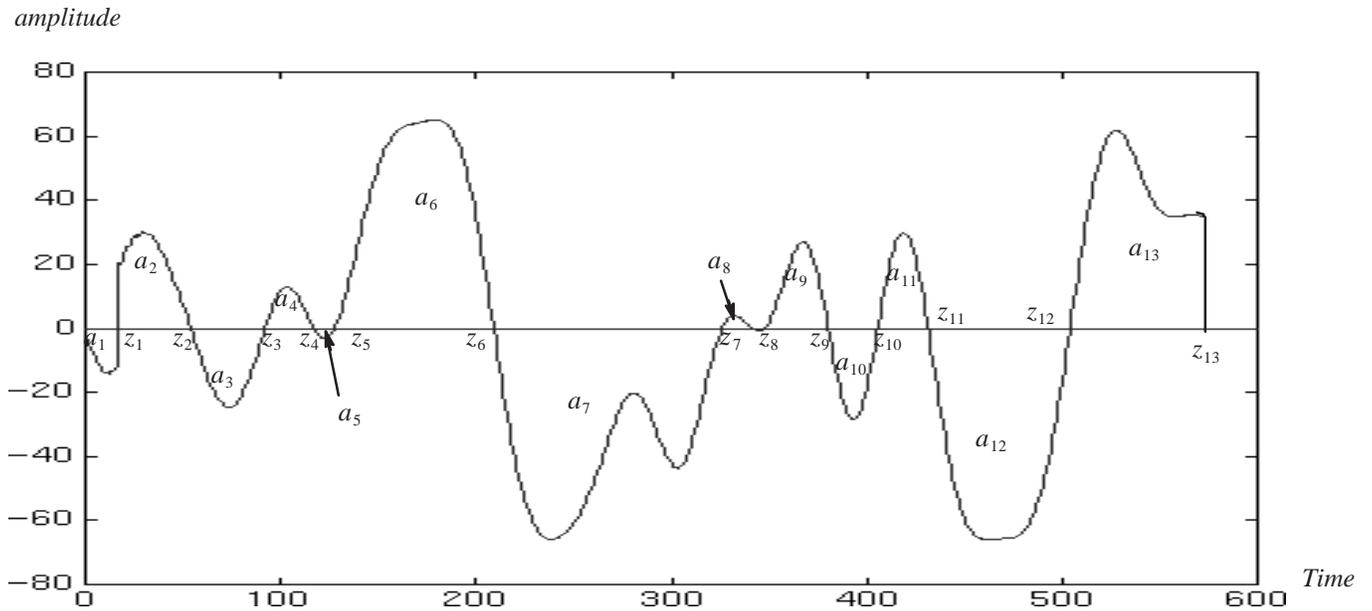


Fig. 3. Illustration showing the meaning of wavelet integrals between zero-crossings.

Mallat has proven that the original signal data can be reconstructed from the abscissae and the wavelet integrals of zero-crossings[18]. Moreover, the integral measure enables us to define a simple  $L^2$  norm on a zero-crossing representation. Basically, this measure can facilitate the signature verification task. Figures 4, 5 and 6 show, respectively, the three wavelet transformed one-dimensional signals for the closed contour indicated by an arrow in Figure 1. Figure 4 is for the  $x$ -coordinate, and Figures 5 and 6 are, respectively, for the  $y$ -coordinate and the tangential angle. From top to bottom in these figures, the signals on the left-hand side are the lowpass data with resolutions of  $2^0$  (the original signal) ,  $2^{-1}$ ,  $2^{-2}$ ,  $2^{-3}$ ,  $2^{-4}$ , and  $2^{-5}$ , respectively; and the signals on the right-hand side are the highpass data with resolutions of  $2^{-1}$ ,  $2^{-2}$ ,  $2^{-3}$ ,  $2^{-4}$ , and  $2^{-5}$ , respectively.

### 3 Optimal Dynamic Thresholding

In Section 2, we described the detailed feature extraction process. The next step is to use these features as bases to compare different signatures. Under the circumstances, a threshold value for determining rejection or acceptance of a comparison is needed. Due to different personal writing styles, it is impossible to uniquely determine a global threshold value that fits all writers. The only thing that we can do is to get training samples from every writer and to then determine his/her own threshold value. Before calculating the writer-dependent threshold value,  $T_i$ , for every writer  $i$ , we have to decide on two other writer-dependent parameters. They are the optimal number of larger closed contours,  $K_i$ , and the optimal resolution used,  $L_i$ . In what follows, we shall explain the physical meaning of  $K_i$  and  $L_i$  for writer  $i$  and the means of determining their values. We have mentioned previously that after preprocessing, every signature image becomes a number of closed contours. But even signatures of the same person may vary in the number of closed contours after the preprocessing process. Therefore, one has to determine a set of closed contours that is most stable in all signatures such that the intra-personal distance of the signatures can be minimized. Hence, finding the optimal number of closed contours,  $K_i$ , for writer  $i$  that can maximize verification performance is a critical issue. On the other hand, since our feature extraction process employs the multiresolution concept, finding the optimal resolution,  $L_i$ , for writer  $i$  is another issue to be studied.

Before describing the relationship among  $T_i$ ,  $K_i$  and  $L_i$ , a so-called *dissimilarity* degree which calculates the distance between two arbitrary signatures has to be defined. As mentioned in the previous section, the signature images for both input signature and reference signatures are represented by zero-crossings.

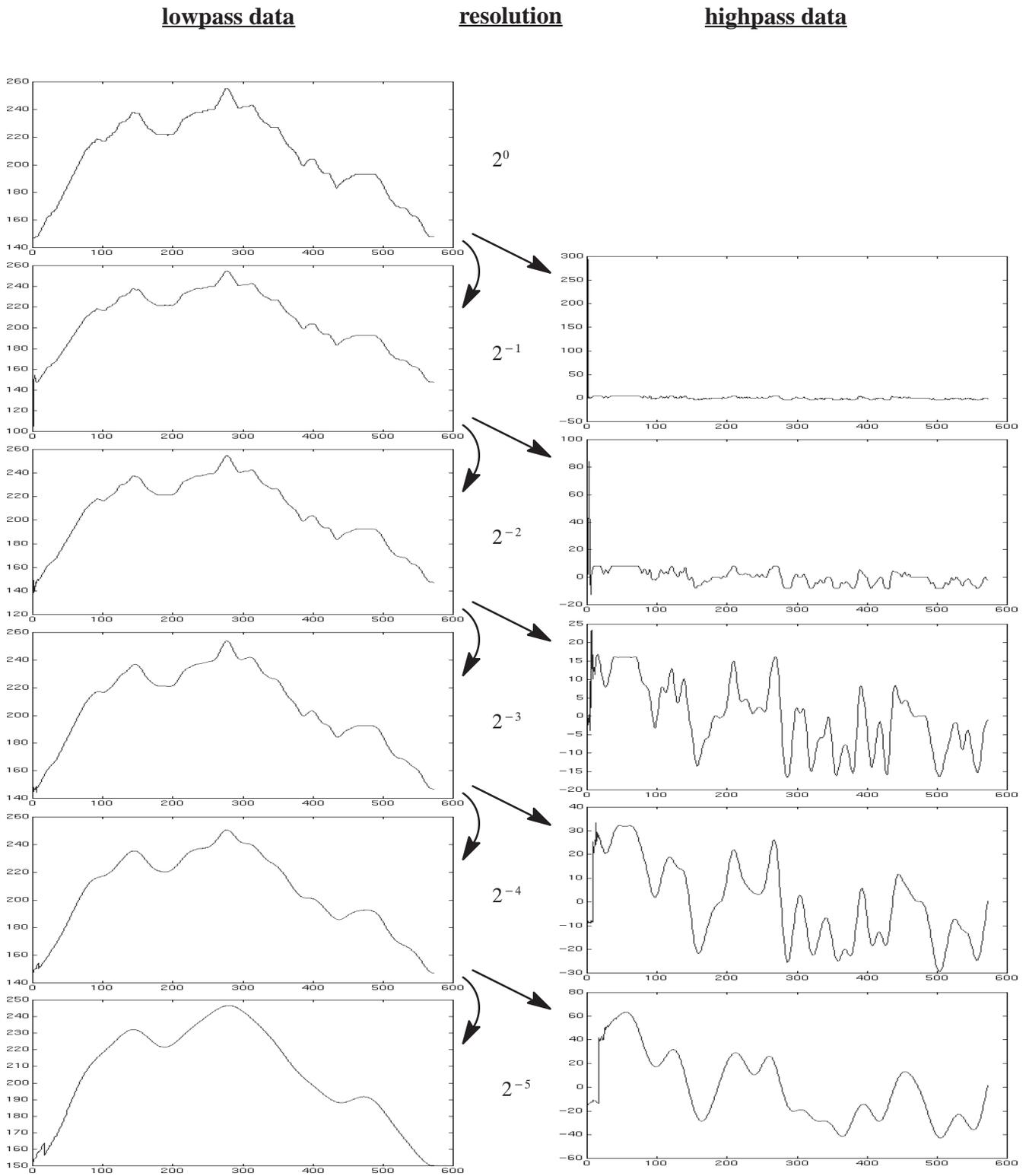


Figure 4. The wavelet transformed multiresolutional  $x$ -coordinate signal for the closed contour indicated by an arrow in Figure 1. The resolutions shown are from  $2^0$  to  $2^{-5}$ , respectively, from top to bottom. The left column represents the lowpass data, and the right column shows the highpass data.

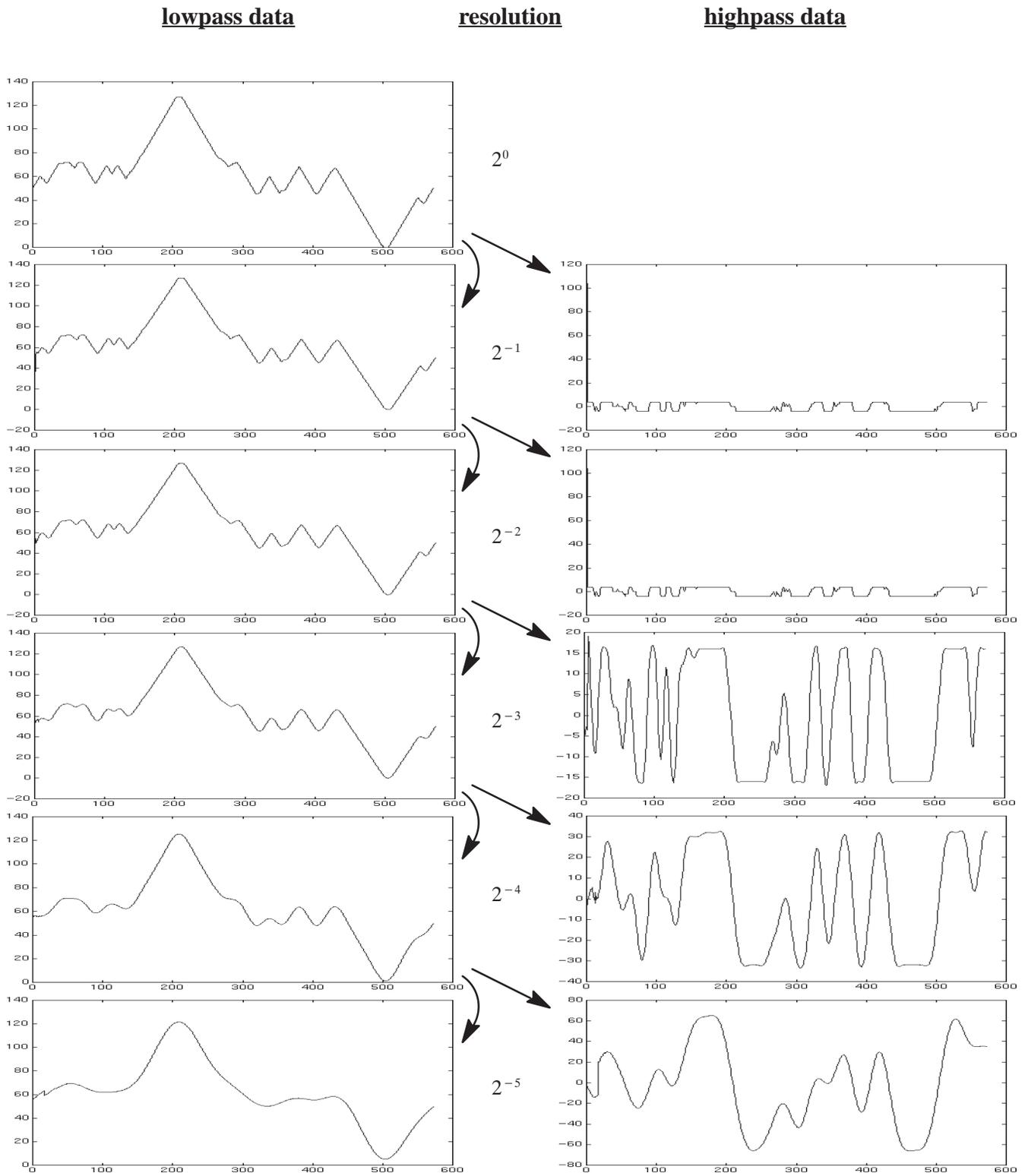


Figure 5. The wavelet transformed multiresolutional y-coordinate signal for the closed contour indicated by an arrow in Figure 1.

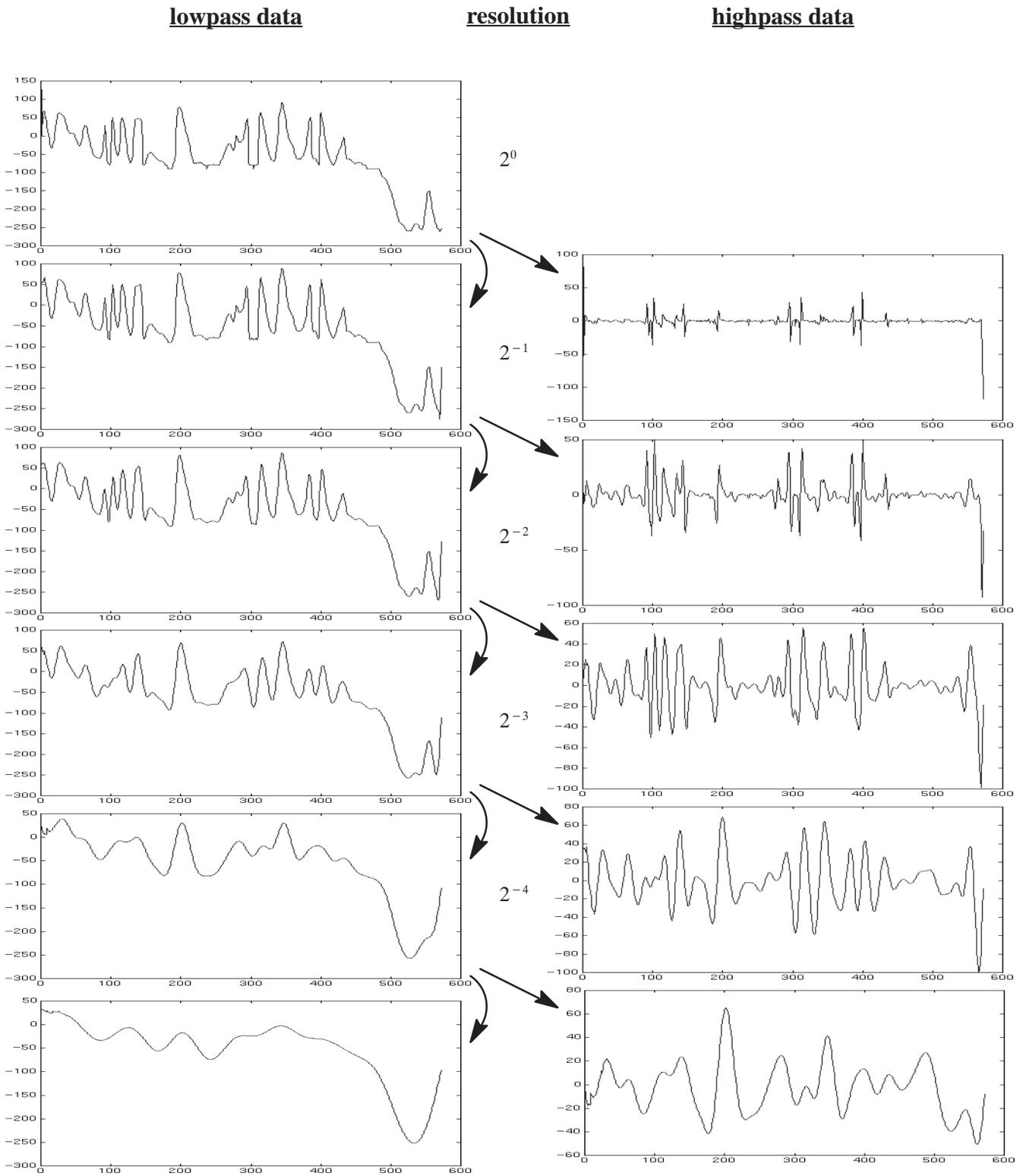


Figure 6. The wavelet transformed multiresolutional tangential angle signal for the closed contour indicated by an arrow in Figure 1.

However, the total number of zero-crossings for a closed contour may vary among different trials of the same person. Therefore, we have a problem, i.e., how to measure the similarity degree between two feature vectors that may differ in the number of zero-crossings. Here, we apply the dynamic time warping (DTW) technique[5][23][24] to solve the problem. DTW can nonlinearly expand or contract the time axis to match the feature points (zero-crossings) between two closed contours. The smaller the value is, the better is the match between the two closed contours. The distance measure used to compare the zero-crossings of two distinct closed contours in DTW is the aggregation of the Euclidean distances of the corresponding attributes in the zero-crossings. The formula for the dissimilarity degree,  $d$ , is defined as follows :

$$d = \sum_{k=1}^{K_i} \sum_{s \in S} d^{L_i}(k, s) , \quad (1)$$

where  $d^{L_i}(k, s)$  represents a primitive part of the total dissimilarity degree  $d$ .  $L_i$  stands for the highpass data of the best resolution that will minimize the dissimilarity. The set  $S$  may include up to three elements, i.e., the  $x$ - and  $y$ -coordinates of a point on a closed contour, respectively, and the corresponding tangential angle of the point.  $K_i$  here represents the number of larger closed contours selected especially for writer  $i$ . Therefore, one has to predetermine  $K_i$  and  $L_i$  and then decide on the value of  $d$ . In what follows, we shall explain how  $T_i$  is defined and how is it related to  $K_i$  and  $L_i$ . Basically,  $T_i$  for writer  $i$  is also a function of  $K_i$  and  $L_i$  because

$$T_i = \mu_i + \delta \cdot \sigma_i , \quad (2)$$

where

$$\mu_i \cong \frac{\sum_{j=1}^n d_{i,j}}{n} , \quad (3)$$

and

$$\sigma_i \cong \sqrt{\frac{\sum_{j=1}^n (d_{i,j} - \mu_i)^2}{n - 1}} . \quad (4)$$

$\mu_i$  and  $\sigma_i$  represent, respectively, the mean and standard deviation of the dissimilarity degree for the  $n$  reference signatures of writer  $i$ .  $d_{i,j}$  ( $j=1, \dots, n$ ) stands for the intra-distance (defined in Equation (1)) between

the reference signature  $j$  and its nearest neighbor among the other  $n-1$  reference signatures of writer  $i$ .  $\delta$  is a parameter which will decide the type I error (false rejection).  $\delta$  is set to 2.0 throughout this paper. Since  $\mu_i$  and  $\sigma_i$  are both functions of  $d_{i,j}$ , and  $d_{i,j}$  is a function of  $K_i$  and  $L_i$ , we can conclude that  $T_i$  in Equation (2) is a function of  $K_i$  and  $L_i$ . In order to determine the best combination of  $K_i$  and  $L_i$  and then determine  $T_i$  so as to minimize the corresponding type II error (false acceptance), an optimization process is needed. The objective function used to guide the optimization process is a so-called normalized threshold value  $\tilde{T}_i$ . Suppose we use  $n$  reference signatures as multiple templates for writer  $i$ ; then, we can apply the nonlinear programming technique to decide on the optimal combination of  $K_i$  and  $L_i$ . The normalized threshold value is defined as follows :

$$\tilde{T}_i = \tilde{\mu}_i + \delta \cdot \tilde{\sigma}_i \quad , \quad (5)$$

where

$$\tilde{\mu}_i \cong \frac{\sum_{j=1}^n (d_{i,j} / \mathcal{L}_{i,j})}{n} \quad , \quad (6)$$

and

$$\tilde{\sigma}_i \cong \sqrt{\frac{\sum_{j=1}^n [ (d_{i,j} - \mu_i) / \mathcal{L}_{i,j} ]^2}{n - 1}} \quad . \quad (7)$$

Equations (5) to (7) are, respectively, very similar to Equations (2) to (4). The difference between the two sets of equations is that the original threshold value  $T_i$ , mean  $\mu_i$  and standard deviation  $\sigma_i$  are now normalized into  $\tilde{T}_i$ ,  $\tilde{\mu}_i$  and  $\tilde{\sigma}_i$  by using a variable  $\mathcal{L}_{i,j}$ .  $\mathcal{L}_{i,j}$  is the total length (in pixels) of the first  $K_i$  longest closed contours in the reference signature  $j$ . Basically, the optimization process is used to minimize  $\tilde{T}_i$ , subject to  $1 \leq K_i \leq K$ , where  $K$  is a constant integer.  $L_i$  here stands for the best resolution with  $1 \leq L_i \leq L$ , where  $L$  is another constant integer indicating the maximum number of resolution levels allowed. For simplicity,  $K$  and  $L$  are globally set 7 and 9 in this paper. The best combination of  $K_i$  and  $L_i$  for writer  $i$  can be determined whenever  $\tilde{T}_i$  reaches a minimum value. The physical meaning of  $\tilde{T}_i$  can be interpreted as the normalized value of the original threshold value. Since  $\tilde{T}_i$  has been normalized, it is invariant to the number of longer

closed contours,  $K_i$ , and the best resolution,  $L_i$ . Therefore, different  $\tilde{T}_i$  values generated by different  $(K_i, L_i)$  combinations can be considered compatible on the same basis. According to the fundamental statistics theory, once  $\delta$  is determined, the type I error (false rejection) is fixed. Under these circumstances, the smaller the threshold  $T_i$  is, the smaller will the type II error (false acceptance) be. From the system design viewpoint, the normalized threshold value,  $\tilde{T}_i$ , is a guide which helps us obtain a smaller type II error.

In general, it is very difficult to find an efficient method to solve the nonlinear programming problem. The brute force method for finding the best combination of  $K_i$  and  $L_i$  is to calculate all  $\tilde{T}_i$ 's under different  $(K_i, L_i)$  combinations. Of course, the best combination will be determined whenever the minimum  $\tilde{T}_i$  value is reached. For simplicity, we adopt a simplified method to solve the problem. That is, the optimal  $K_i$  is determined first by fixing  $L_i$ . Once  $K_i$  for writer  $i$  is determined, the value of  $L_i$  is then decided by plugging in the predetermined  $K_i$ . Here, we assume that there are five resolutions that may lead to the minimum dissimilarity value, i.e.,  $L_i = 1, 2, \dots, 5$ . As to the  $K_i$  value, we set its range from 1 to 7. For each  $K_i$ , we plug five different  $L_i$  values into Equation (1) and aggregate their total dissimilarity value. Under the circumstances, the optimal  $K_i$  value is the one that corresponds to the minimum  $\tilde{T}_i$  value. When the  $K_i$  value for a specific writer  $i$  is determined, we can then go back to decide which  $L_i$  is really the "best" value by using the predetermined  $K_i$ . Since we do not know which resolution will lead to the minimum  $\tilde{T}_i$  value, we use the first nine resolutions for comparison. That is, when  $K_i$  is predetermined for writer  $i$ , we then plug  $L_i = 1, 2, \dots, 9$ , respectively, into Equation (1) to derive the optimal  $L_i$  value by choosing the specific  $L_i$  ( $L_i = 1, 2, \dots, 9$ ) that minimizes  $\tilde{T}_i$ .

For illustrative purposes, we will use an example to show the optimal  $(K_i, L_i)$  pair determination process. First of all,  $L_i$  is set to be the first five resolutions. The  $\tilde{T}_i$  value here will be determined by aggregating the results of highpass data from resolution  $2^{-1}$  to  $2^{-5}$ . Table 1 presents an example showing how the optimal number of closed contours  $K_i$  for writer  $i$  is chosen automatically. In the example, the  $K_i$  value used for experimentation can be up to 7. The three columns on the right of Table 1 represent the computed  $\tilde{T}_i$  values for writers 1, 2, and 3, respectively, under different  $K_i$ 's. These writers correspond, respectively, to the three writers who wrote the signatures shown in the first row of Figure 1. Ten reference signatures were collected from each writer for training. In Table 1, the optimal  $K_i$  equals to 1 for writer 1, 3 for writer 2, and

2 for writer 3, Their corresponding  $\tilde{T}_i$  values are in bold type. The fourth row of Figure 1 shows the corresponding optimal set of closed contours for the three writers.

Table 1. The value of  $\tilde{T}_i$  as a guide for finding the optimal number of larger closed contours,  $K_i$ .  $\tilde{T}_i$  here is the aggregation result obtained by using the first five resolutions of the highpass data of the  $x$ - and  $y$ -coordinates. For different writers, the optimal  $K_i$  value will be different.

$K_i \backslash \tilde{T}_i$	writer $i$		
	1	2	3
1	<b>0.002153</b>	0.053506	0.075504
2	0.005832	0.110717	<b>0.023803</b>
3	0.014501	<b>0.014121</b>	0.036106
4	0.011518	0.021264	0.069927
5	0.030093	0.048248	0.039321
6	0.056257	0.026086	0.190342
7	0.119720	0.008406	0.245629

Table 2.  $\tilde{T}_i$  is used as a guide for finding the optimal resolution,  $L_i$ . Here, we use the results in Table 1 to determine the optimal  $K_i$  value and then choose the specific  $L_i$  value that generates the minimum  $\tilde{T}_i$  as the optimal  $L_i$  for writer  $i$ .

$L_i \backslash \tilde{T}_i$	writer $i$		
	1	2	3
1	0.003509	0.015381	0.029989
2	0.006410	0.014994	<b>0.016269</b>
3	0.003121	<b>0.009800</b>	0.021100
4	<b>0.002508</b>	0.028672	0.027075
5	0.022350	0.172897	0.154069
6	0.056559	0.816786	0.375193
7	1.587870	7.488531	4.583840
8	1.939688	211.736466	50.047103
9	6.034618	NA	381.579356

After the optimal  $K_i$  for writer  $i$  is determined, the next step is to decide on the optimal  $L_i$ . At this point, we substitute the optimal  $K_i$  value into Equations (5) to (7) for each writer  $i$ , respectively. For  $L_i = 1$  to 9, their corresponding  $\tilde{T}_i$  values are calculated, respectively. One thing to note is that the  $\tilde{T}_i$  value at  $L_i = 9$  of writer 2 is not available. This is because the longest detected closed contour of writer 2 is shorter than 512 pixels (  $2^9$  ). Under these circumstances, no data will be grabbed at this resolution. For an arbitrary writer  $i$ , the value of  $L_i$  that generates the minimum  $\tilde{T}_i$  value will be chosen as the optimal resolution. Once

the optimal  $(K_i, L_i)$  pair is determined, the dynamic threshold value  $T_i$  for writer  $i$  can be determined as well.

## 4 Building a Database and Verification

The dynamic threshold value determination process has been clearly described in the previous section. In this section, we shall first describe how an appropriate database can be constructed and then report the complete verification algorithm.

### 4.1 Database Construction

At the present, there is still no standard database for an off-line handwritten verification system. Before constructing an appropriate database for signature verification, one has to know the basic requirements of building such a database. Basically, the quality of a forged signature mainly depends on the following factors:

- (1) the characteristics of the reference signatures — simple or complex, stable or unstable;
- (2) the forger's level of education — professional, skilled or unskilled;
- (3) the learning time — how much time is needed by the forger to learn from the reference genuine signatures;
- (4) the forgery lead time — the length of time between the end of learning and the beginning of forging the signature;
- (5) the forging time — how much time allowed to forge the signature; and
- (6) the forgery style — whether the signature is forged by copying or tracing, forged by copying with or without observation, forged under supervision, and so forth.

Since the focus of this research is off-line signature verification, construction of a database is quite different from that of an on-line signature verification system. In what follows, we shall describe the features of our database. The database includes :

- (1) genuine signatures (type  $A$ ) — Ten genuine signatures were used as multiple templates, and another 10 genuine signatures were used as part of the test samples.
- (2) forgery signatures (type  $B$ ) — The qualities of forgery test signatures were randomized. That is, we mixed 5 different quality types (i.e.,  $B_1, B_2, B_3, B_4,$  and  $B_5$ ) of forged signatures in the data-

base for testing. Among them, 10 forged signatures were collected for each type. In order to generate forged signatures of different types, we let the forgers observe 10 genuine signatures for different periods of time. The time periods required for making the  $B_1, B_2, B_3, B_4$ , and  $B_5$  type forgery signatures were, respectively, unlimited, 3 minutes, 2 minutes, 1 minute, and forgery without learning. After observing (learning) for different periods of time, the forgers then had unlimited time to forge signatures.

Based on the above mentioned learning process, the database has randomized but smoothly distributed forgery signatures. Figure 7 shows two sets of examples, including one set of English signatures and one set of Chinese signatures. The top row of each set contains the genuine signatures of the English and Chinese set, respectively. Rows 2 to 6 in each column represent, respectively, the  $B_1, B_2, \dots, B_5$  forged signatures of the English and Chinese set.

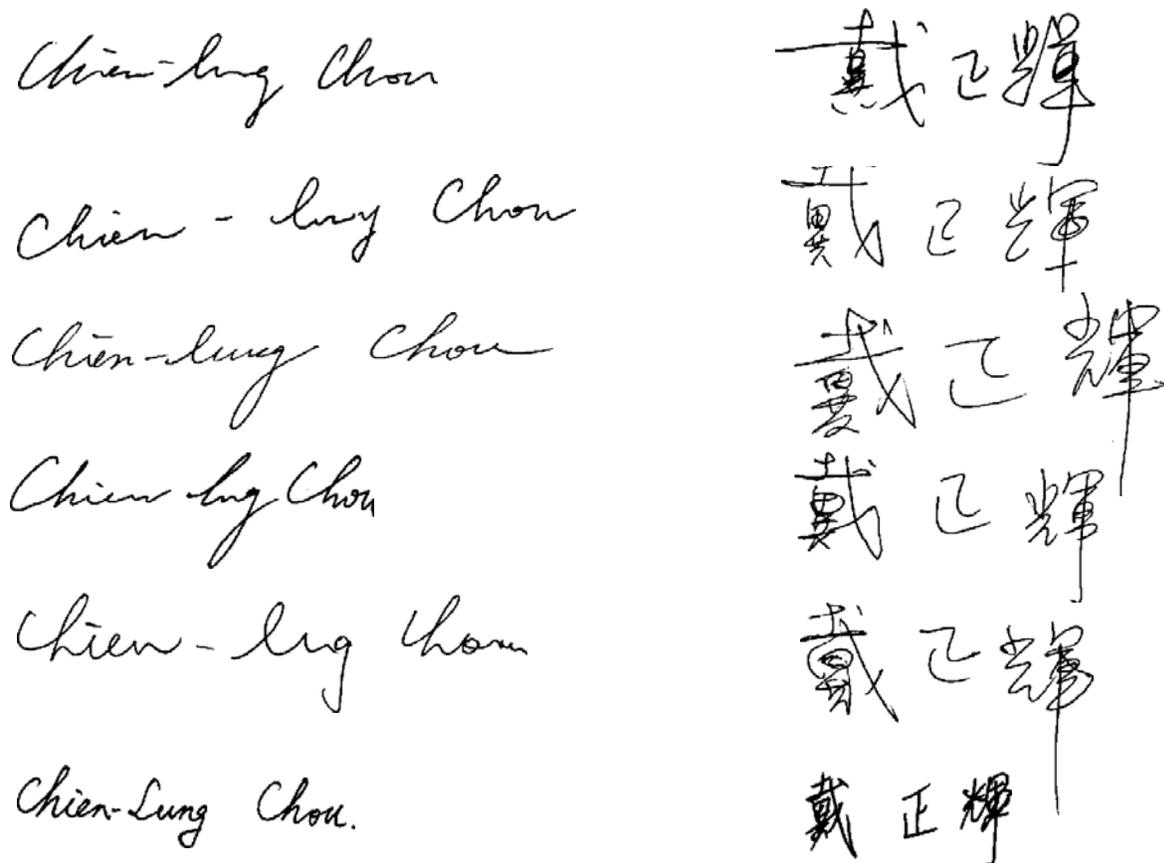


Figure 7. Examples of genuine and forged signatures. The left column contains 6 English signatures which represent the genuine signature (type A) and the forged signatures (type  $B_1$  to  $B_5$ ). The right column is a set of Chinese signatures in a similar order.

## 4.2 Verification

The verification process will be described in this section. Suppose the input signature is  $I$ , and that there exist  $n$  reference signatures,  $R_1, R_2, \dots, R_n$ , for writer  $i$ . The verification process can be divided into the following steps :

- STEP 1. Calculate the total number of closed contours in  $I$ . If this number is less than  $K_i$  ( the optimal larger closed contour number for writer  $i$  ), then reject  $I$ .
- STEP 2. Compare the boundary lengths for each closed contour pair. To speed up the verification process, we use the length of every closed contour as a feature to discard dissimilar reference signatures before using DTW for verification.
- STEP 3. Calculate the dissimilarity degree  $d_I$  between  $I$  and all the remaining signatures,  $R'_j$ , where  $1 \leq j \leq m$  and  $m \leq n$ ,  $R'_j \in \{R_1, R_2, \dots, R_n\}$ .
- STEP 4. Compare  $d_I$  with a preset dynamic threshold value  $T_i$  for writer  $i$ . If  $d_I \leq T_i$ , then *accept* the signature; otherwise, *reject* it.

## 5 Experimental Results

In this section, we shall report some experimental results. As to the database part, we collected genuine signatures from 25 English writers and 25 Chinese writers, each of whom wrote their signature 20 times. Among the 20 genuine signatures of each writer, 10 were used as templates, and the other 10 were used for testing. We also collected 5 different quality types of forged signatures from 40 senior undergraduate students during one semester, where 10 forgery signatures were collected per each forgery signature type from each writer. Therefore, there were 3,500 signatures in total in the database.

In the following, a series of experiments conducted for different purposes will be described. Before reporting the results, we shall introduce some conventions commonly adopted in statistics. These conventions will be used as bases for reporting the performance of the proposed approach. For example, the type I error represents the possibility that a genuine signature is mis-classified as a forged one. The type II error, on the contrary, stands for the possibility that a forged signature is mis-identified as a genuine one. The average error used in this work is simply the average of the type I error and the type II error. Moreover, without special explanation, the experimental results shown in the forthcoming tables were generated un-

der the following conditions: (1) the attributes used to represent a zero-crossing were the abscissa, the wavelet integral, and the corresponding amplitude with the same abscissa in the lowpass data one level up, (2) the data obtained from a closed contour were the  $x$ -coordinates and  $y$ -coordinates, (3) the algorithm used to generate the optimal number of larger closed contours,  $K_i$ , and the optimal resolution,  $L_i$ , for each writer  $i$  was based on the method described in Section 3, and (4) type II errors were determined using the five different quality types of forged signatures as test data.

In the following, the complete dissimilarity degree calculation process using a real example will be illustrated. The dissimilarity degree here includes the distance between each of the primitive parts from any two different signatures. In addition, the calculation of the optimal threshold value as well as the training and testing processes of the proposed system will also be described. Figure 8 shows 30 signature images (a mixture of genuine and forged signatures) which correspond to the signatures of writer 2 in Figure 1. The left column of Figure 8 includes 10 genuine signatures. We denote them, from top to bottom, as  $T^{01}$ ,  $T^{02}$ , ...,  $T^{10}$ , respectively. These genuine signatures were used as reference (template) signatures. The 10 signatures shown in the middle column are genuine signatures used for testing. They are denoted, respectively, as  $A^{01}$ ,  $A^{02}$ , ...,  $A^{10}$ . The right column, on the other hand, includes 10 forged signatures of the  $B_1$  type. They are denoted, respectively, as  $B_1^{01}$ ,  $B_1^{02}$ , ...,  $B_1^{10}$ . After applying the morphological dilation, closed-contour tracing, and larger closed-contour selection, the corresponding processing results were those shown in Figure 9, 10, and 11, respectively.

We shall next explain how of the dissimilarity degree of two different signatures is calculated. Here, we only choose the  $x$ - and  $y$ -coordinate as the elements of  $S$  set. The two signatures chosen are  $B_1^{01}$  and its nearest neighbor,  $T^{04}$ , among the ten template signatures of writer 2. The index  $k$  here represents the closed contours ordered from left to right (Figure 11). The optimal resolution ( $L_i$ ) and the optimal number of closed contours ( $K_i$ ) for writer 2 are predetermined by using the results shown in Tables 1 and 2. Therefore, they are both 3 under these circumstances. Table 3 shows the calculated primitive parts,  $d^{L_i}(k, s)$ , of the two signatures. Based on the same calculation process, the dissimilarity process, the dissimilarity degree between  $B_1^{01}$  and all ten reference signatures of writer 2 are shown in Table 4. It is obvious that  $T^{04}$  is the closest neighbor to  $B_1^{01}$ .

Having the information shown in Table 3 and 4, we are now able to calculate the dynamic threshold value, for a specific writer  $i$ . From Equations (2) to (4), we realize that the calculation of  $T_i$  requires the

information of  $\mu_i$  and  $\sigma_i$ . Therefore we have to calculate the intra-distance among the training (reference or template) signatures. From the calculation results shown in Table 3 and 4, we can calculate the closest neighbor of every signature. The three columns on the left side of Table 5 include the set of nearest neighbors of every reference signature of writer 2. By plugging these intra-distances into Equations (2) to (4),  $\mu_i$ ,  $\sigma_i$  and  $T_i$  can be determined. They are 28.52, 3.99 and 36.50, respectively. This means that, in the testing process, if the dissimilarity value is less than 36.50, then the test data will be considered to be genuine. The middle three columns of Table 5 include a set of testing results obtained using 10 genuine test signatures. It can be seen that all the 10 test data pass the test and can be considered genuine signatures. The three columns on the right side of Table 5 are another set of results obtained by using 10 type  $B_1$  forged signatures. In this set of experiments, two test samples,  $B_1^{02}$  and  $B_1^{06}$ , were misclassified as genuine ones since their dissimilarity value were smaller than 36.50. Under these circumstances, the type II error occurred.

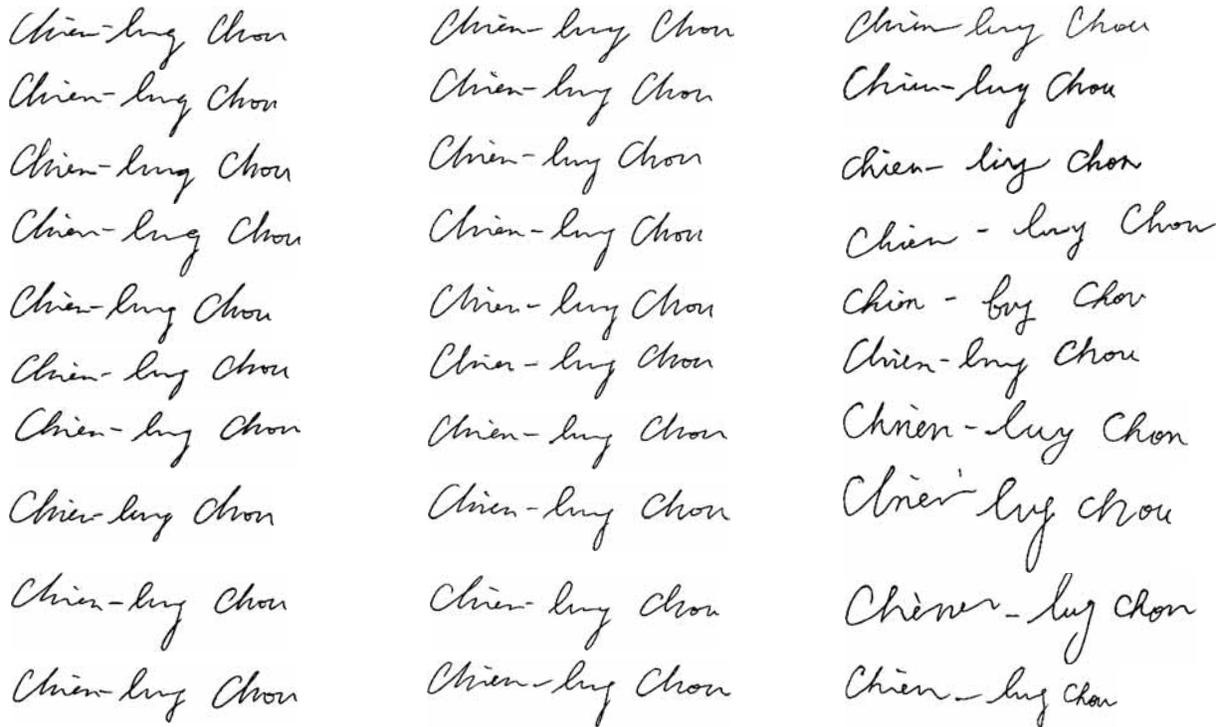


Figure 8. Some genuine and forged signatures of writer 2 shown in Figure 1. The left column includes 10 genuine signatures used as database data ( $T^{01} - T^{10}$ ). The middle column shows 10 genuine signatures for testing ( $A^{01} - A^{10}$ ). The right column shows 10 forged signatures, also used for testing ( $B_1^{01} - B_1^{10}$ ).

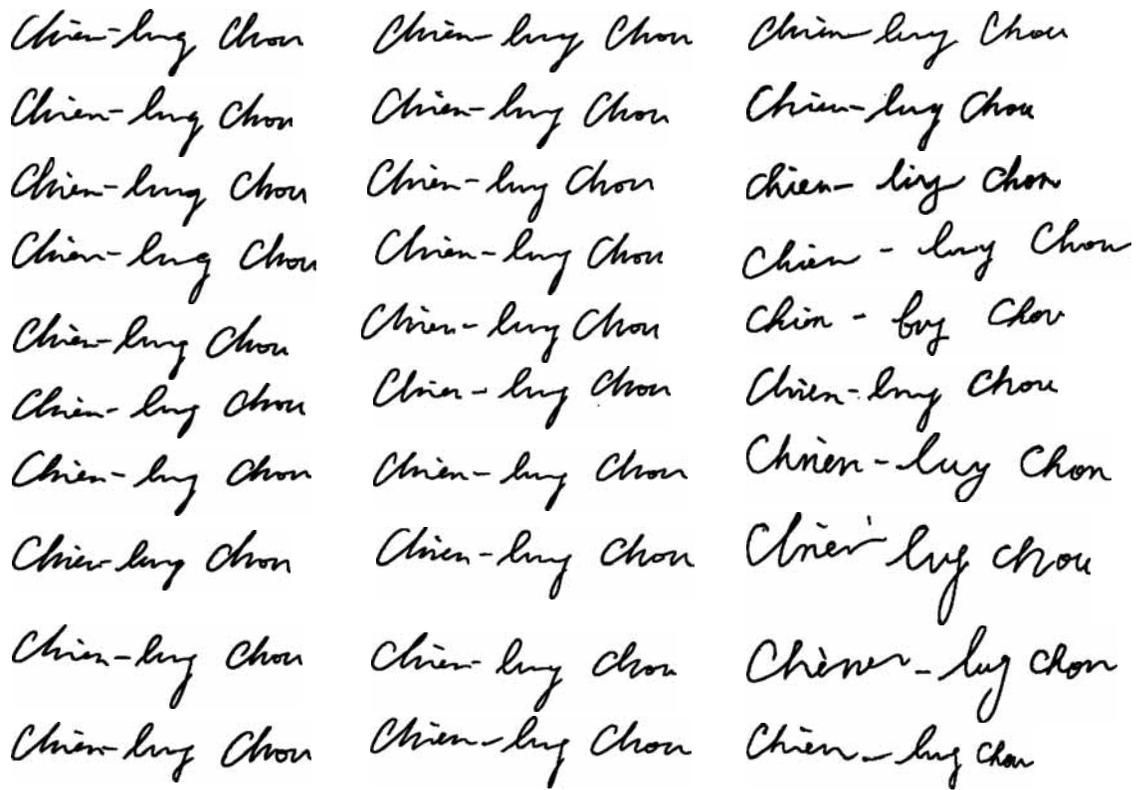


Figure 9. The morphologically dilated signature images of the signatures shown in Figure 8.

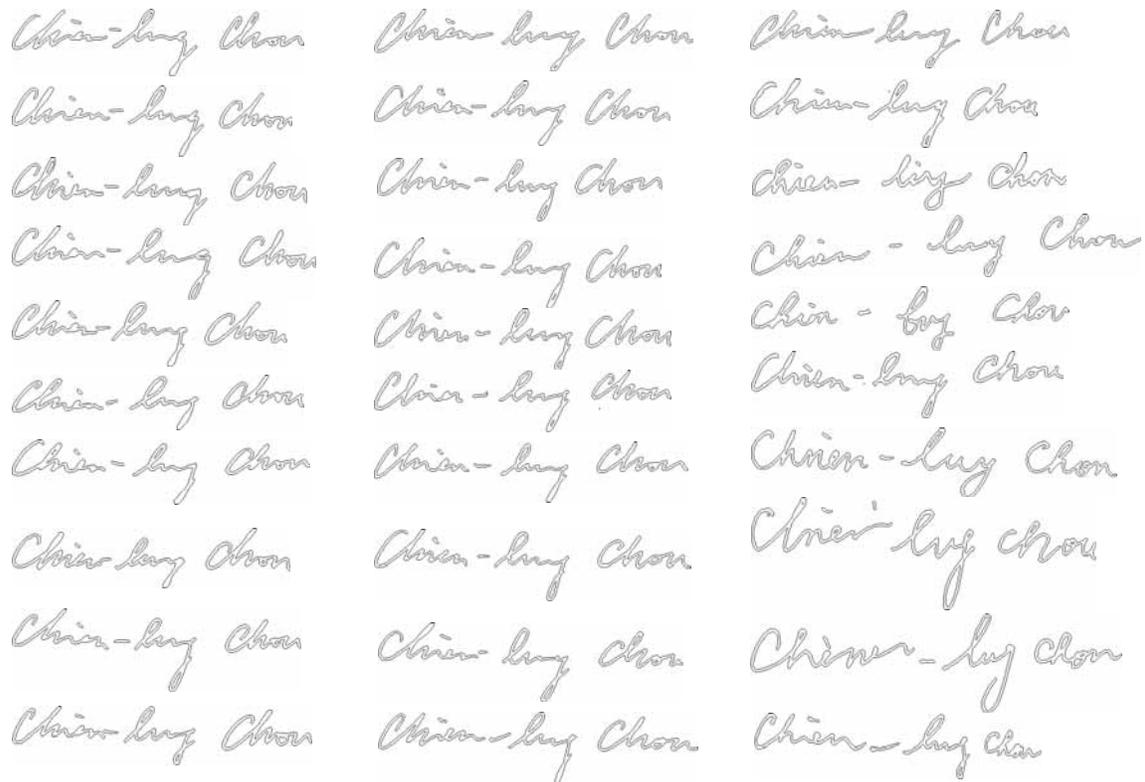


Figure 10. The closed-contour extracted signature images of the signatures shown in Figure 9.

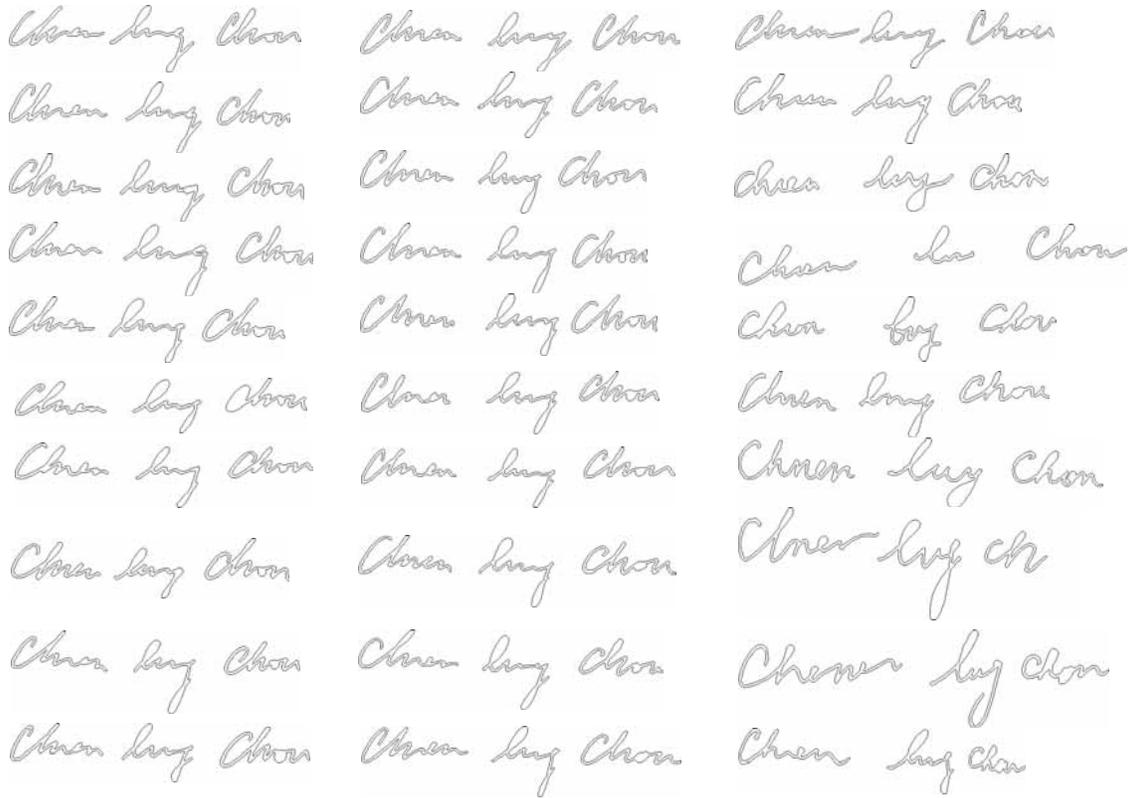


Figure 11. The optimal larger closed-contour signature images of the signatures shown in Figure 8.

Table 3. For primitive parts (  $d^{L_i}(k, s)$  ), the degree of dissimilarity between the test signature,  $B_1^{01}$ , and its nearest neighbor,  $T^{04}$ .

<div style="text-align: center;"> <i>Closed Con-</i>  <i>tour(k)</i>  <i>Signal(s)</i> \ <math>d^{L_i}(k, s)</math> </div>	1	2	3
X – axis	10.51	5.82	4.82
Y – axis	5.77	6.15	4.53

Table 4. The degree of dissimilarity between the test sample,  $B_1^{01}$ , to each of the ten reference signatures of writer 2, respectively. We can find that the distance from  $B_1^{01}$  to  $T^{04}$  is the shortest. In other words, template  $T^{04}$  is the nearest neighbor of test sample  $B_1^{01}$ .

Template	$T^{01}$	$T^{02}$	$T^{03}$	$T^{04}$	$T^{05}$	$T^{06}$	$T^{07}$	$T^{08}$	$T^{09}$	$T^{10}$
Distance	45.08	43.88	38.35	<b>37.60</b>	52.79	51.90	44.80	50.12	42.97	49.20

Table 5. Three sets of dissimilarity values obtained by applying 10 training signatures (left 3 columns), 10 genuine test signatures (middle 3 columns), and 10 forged test signatures (right 3 columns).

<i>Training Samples</i>			<i>Test Samples</i>					
<i>Reference Signature</i>	<i>Nearest Neighbor</i>	<i>Distance</i>	<i>Genuine Signature</i>	<i>Nearest Neighbor</i>	<i>Distance</i>	<i>Forged Signature</i>	<i>Nearest Neighbor</i>	<i>Distance</i>
$T^{01}$	$T^{02}$	28.06	$A^{01}$	$T^{04}$	34.31	$B_1^{01}$	$T^{04}$	37.60
$T^{02}$	$T^{01}$	27.41	$A^{02}$	$T^{01}$	25.49	$B_1^{02}$	$T^{02}$	25.00
$T^{03}$	$T^{01}$	30.74	$A^{03}$	$T^{08}$	28.70	$B_1^{03}$	$T^{04}$	36.56
$T^{04}$	$T^{09}$	35.06	$A^{04}$	$T^{09}$	30.52	$B_1^{04}$	$T^{03}$	77.38
$T^{05}$	$T^{01}$	27.73	$A^{05}$	$T^{10}$	31.38	$B_1^{05}$	$T^{04}$	37.40
$T^{06}$	$T^{07}$	25.52	$A^{06}$	$T^{07}$	32.82	$B_1^{06}$	$T^{09}$	29.49
$T^{07}$	$T^{06}$	22.09	$A^{07}$	$T^{07}$	27.53	$B_1^{07}$	$T^{03}$	69.19
$T^{08}$	$T^{03}$	34.94	$A^{08}$	$T^{09}$	32.80	$B_1^{08}$	$T^{03}$	117.36
$T^{09}$	$T^{02}$	24.30	$A^{09}$	$T^{03}$	26.63	$B_1^{09}$	$T^{04}$	74.17
$T^{10}$	$T^{05}$	29.24	$A^{10}$	$T^{03}$	36.43	$B_1^{10}$	$T^{09}$	52.75

Table 6. The results showing how resolution selection affects system performance.

<i>Resolution</i>	<i>Language Error Type Error Rate(%)</i>	<i>English Signatures</i>			<i>Chinese Signatures</i>		
		<i>Type I Error</i>	<i>Type II Error</i>	<i>Average Error</i>	<i>Type I Error</i>	<i>Type II Error</i>	<i>Average Error</i>
1. $L_i = 1$ for all writers		6.00	12.96	9.48	8.00	12.60	10.30
2. $L_i = 2$ for all writers		6.40	13.12	9.76	9.50	10.20	9.85
3. $L_i = 3$ for all writers		7.20	12.96	10.08	9.00	10.10	9.55
4. $L_i = 4$ for all writers		5.60	12.24	8.92	5.50	10.33	7.90
5. $L_i = 5$ for all writers		5.20	13.92	9.56	7.00	11.00	9.00
6. $L_i = 6$ for all writers		5.20	17.36	11.28	9.50	19.80	14.65
7. $L_i = 7$ for all writers		13.20	17.04	15.12	18.00	19.00	18.50
8. $L_i = 8$ for all writers		18.10	20.67	19.38	20.80	21.54	21.17
9. $L_i = 9$ for all writers		22.73	28.73	25.73	45.00	18.33	31.65
10. optimal dynamic $L_i$		5.60	10.98	<b>8.29</b>	6.00	7.80	<b>6.90</b>

The above example has shown clearly how the proposed system works. In what follows, we shall describe a series of experiments to show how parameters affect the system performance. First of all, we shall show how  $L_i$ , the best chosen resolution, affects the performance. Table 6 shows the type I error, the

type II error, and the average error, respectively, for English signatures as well as Chinese signatures. The first nine rows correspond, respectively, to the nine cases with  $L_i$  fixed at from 1 to 9 for all writers. The bottom row of Table 6, on the other hand, corresponds to errors determined by using the optimal  $L_i$  value dynamically for each writer  $i$ . From the data shown in Table 6, it is obvious that the average error generated by dynamic selection of  $L_i$  is smaller than that of any of the other nine cases, in which a fixed resolution for signature verification was used.

In the preprocessing stage, as we have mentioned, every point on a closed contour, its  $x$ -coordinate,  $y$ -coordinate, and corresponding tangential angle were recorded. Based on the following experiment, we shall report which items among the three recorded ones are better for signature verification. Again, the data shown in Table 7 represent, respectively, the type I error, type II error and average error, for both English signatures and Chinese signatures. The error values reported in the first row (top row) are the results obtained by simply using the  $x$ -coordinate signal. The error values reported in the second, the third, and the bottom row correspond, respectively, to the results obtained using the  $y$ -coordinate only, both the  $x$ -coordinate and  $y$ -coordinate, and all three items. From Table 7, the average error values shown in the third row, i.e., the results obtained using both the  $x$ -coordinate and  $y$ -coordinate, represent the minimum error value. When we added one more item, i.e., the tangential angle signal, the performance of the signature verification system was degraded significantly. Therefore, at the very beginning of each experiment, we did not use the tangential angle in the example.

Table 7. System performance evaluated using different signal combinations.

<i>Language</i> <i>Error</i> <i>Type</i> <i>Error</i> <i>Rate(%)</i>	<i>English Signature</i>			<i>Chinese Signature</i>		
	<i>Type I</i> <i>Error</i>	<i>Type II</i> <i>Error</i>	<i>Average</i> <i>Error</i>	<i>Type I</i> <i>Error</i>	<i>Type II</i> <i>Error</i>	<i>Average</i> <i>Error</i>
<i>x</i> -coordinate	6.00	15.30	10.68	8.00	20.96	14.48
<i>y</i> -coordinate	6.00	15.12	10.56	6.00	21.44	13.72
<i>x</i> -coord. and <i>y</i> -coord.	5.60	10.98	<b>8.29</b>	6.00	7.80	<b>6.90</b>
<i>x</i> -, <i>y</i> -coord. and tan. angle	4.80	12.48	8.64	7.20	18.08	12.64

In the feature extraction process, three attributes were extracted from each zero-crossing point. They are: the abscissa of the zero-crossing, its left-hand side wavelet integral, and the corresponding am-

plitude with the same abscissa in the lowpass data one resolution up. Basically, the abscissa attribute for a zero-crossing was proposed by Logan[25]. Later, in 1991, Mallat[18] proposed use of the abscissa and the left-hand side wavelet integral as attributes. In this study, we added another attribute for a zero-crossing, i.e., the amplitude with the same abscissa in the lowpass data one resolution up. In Table 8, we show in the first row to the third row, the verification results obtained by applying Logan’s method[25], Mallat’s method[18], and our method, respectively. Mallat[18] claimed that his method is more stable than Logan’s method, but they are very close in terms of the verification rate. As to our method, by simply adding one attribute, the verification results improve greatly for both English and Chinese signatures.

Table 8. System performance evaluated using different combinations of attributes.

Method	Language		English Signature			Chinese Signature		
	Error Type	Rate(%)	Type I	Type II	Average	Type I	Type II	Average
			Error	Error	Error	Error	Error	Error
1. Logan’s method			6.40	22.88	14.64	9.00	24.50	16.75
2. Mallat’s method			6.00	21.52	13.76	17.00	21.70	19.35
3. Our method			5.60	10.98	<b>8.29</b>	6.00	7.80	<b>6.90</b>

Table 9. System performance for different types of forgery.

Language	Test Sample	Sample Type	Error Rate(%)	Genuine Sample					Forgery Sample					
				A	B <sub>1</sub>	B <sub>2</sub>	B <sub>3</sub>	B <sub>4</sub>	B <sub>5</sub>					
English				5.60	21.20	13.60	10.48	9.60	0.00					
Chinese				6.00	13.50	9.50	9.00	7.00	0.00					

In the last set of experiments, we studied the robustness of the proposed system as using different types of test samples. These test samples included a  $A, B_1, B_2, B_3, B_4,$  and  $B_5$  types of signatures as mentioned in Section 4.1. Table 9 shows a set of error rates for the robustness test of the proposed system. It can be seen in Table 9 that the average error of  $B_5$  (a simple forgery which was written knowing only the spelling of the name) for both English and Chinese signatures is zero. This outcome indicates that when a forger blindly

signs (without learning) a signature, the proposed system can detect the forgery correctly every time. On the other hand, if a forger has a longer period of time to learn a genuine signature, then the proposed system has a higher chance of making a wrong judgement. This conclusion is reflected in the results of the  $B_1$  type test. For the English signature, the proposed system mis-identified about 21.20% of forged signatures as genuine ones. As for the Chinese signature, the mis-identification rate was about 13.50%.

## 6. Conclusions and Future Work

In this paper, we have proposed a novel approach to off-line handwritten signature verification. The method includes application of a closed-contour tracing technique to transform a signature image into one-dimensional signals to extract not only static, but also dynamic curvature information, and use of a wavelet-based feature extractor to extract complete and stable features from multiresolutional signals. An optimization-based dynamic thresholding algorithm has been proposed to detect not only simple, but also skillful forgeries. Following are some further works that can be done in the future:

- (1) Combine the proposed features with traditional statistical or geometrical features, for example, the area, geometrical center, and center of gravity of the close contours, and the width, height, and histogram of the signature image. Basically, the combination of these features can enhance the performance of the verification system.
- (2) Apply the method proposed in this work to solve the on-line handwritten signature identification and verification problem.
- (3) Devise an effective wavelet-based approach to detection of forgeries written by tracing a genuine signature. This is one of the most difficult types of forged signatures to detect.

## REFERENCES

- [1] Y. Qi and B. R. Hunt, "A multiresolution Approach to Computer Verification of Handwritten Signature," *IEEE Trans. Image Processing*, vol. 4, pp. 870-874, June 1995.
- [2] S. Lee and J. C. Pan, "Off-line tracing and representation of Signatures," *IEEE Trans. Syst., Man and Cybern.*, vol. 22, no. 4, pp. 755-771, July/August 1992.

- [3] M. Ammar, "Progress in Verification of Skillful Simulated Handwritten Signatures," *Character & Handwriting Recognition*, World Scientific Inc., pp. 337–351, 1990.
- [4] R. Plamondon and G. Lorette, "Automatic Signature Verification and Writer Identification – The State of the Art," *Pattern Recognition*, vol. 4, no. 2, pp. 107–131, 1989.
- [5] M. Parizeau and R. Plamondon, "A Comparative Analysis of Regional Correlation, Dynamic Time Warping, and Skeletal Tree Matching for Signature Verification," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 12, no. 7., pp. 710–717, July 1990.
- [6] J. Brault and R. Plamondon, "A Complexity Measure of Handwritten Curves: Modeling of Dynamic Signature Forgery," *IEEE Trans. Syst., Man and Cybern.*, vol. 23, no. 2, pp. 400–413, March/April 1993.
- [7] C. Wen, B. Jeng, P. Ting and H. Yau, "Handwritten Chinese Signature Verification using Tremor Features," in *Proc. 5th Optical Character Recognition and Document Analysis*, 1996.
- [8] M. Ammar, Y. Yoshida and T. Fukumura, "Structural description and Classification of Signature Images," *Pattern recognition*, vol. 23, no. 7., pp. 697–710, 1990.
- [9] H. S. Chen "Handwriting Verification," *Journal of Police Science*, vol. 26, no. 5, pp.17–29, March 1996.
- [10] H. Chang, J. Wang and H. Suen, "Dynamic Handwritten Chinese Signature Verification," in *Proc. 2nd Int. Conf. on Docu. Analysis and recognition*, pp. 258–261, Oct. 1993.
- [11] M. J. Paulik and N. Mohankrishnan, "A 1–D Sequence Decomposition Based, Autoregressive Hidden Markov Model for Dynamic Signature Identification and Verification," in *Proc. 36th Midwest Symposium on Circuit and Sys.*, pp. 138–141, Aug. 1993.
- [12] K. Dhar and A. Kunz, "Digital Technique to Analyze Handwritten Signatures," in *Proc. IEEE 1988 Inter. Carnahan Conf. on Security Tech.*, pp. 9–13, 1988.
- [13] I. Pottier and G. Burel, "Identification and Authentication of Handwritten Signatures with a Connectionist Approach," in *Proc. 1994 IEEE Conf. on Neural Networks*, pp. 2948–2951, July 1994.

- [14] A. M. Darwish and G. A. Auda , “A New Composite Feature Vector for Arabic Handwritten Signature Recognition, ” in *Proc. 1994 IEEE Int. Conf. on ASSP*, pp. 613–616, April 1994.
- [15] R. Sabourin and J. Drouhard, “Off–line Signature Verification Using Directional PDF and Neural Networks, ” in *Proc. 11th IAPR Int. Conf. on Pattern Recognition*,, pp. 321–325, Sept. 1992.
- [16] T. Wakahara, H. Murase and K. Odaka, “Online Handwritten Recognition, ” *Proceedings of the IEEE*, vol. 80, no. 7, pp. 1181–1194, 1992.
- [17] S. G. Mallat, “A Theory for Multiresolution Signal Decomposition: The Wavelet representation” *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 11, no. 7., pp. 674–693, July 1989.
- [18] S. G. Mallat, “Zero–Crossings of a Wavelet Transform, ” *IEEE Trans. Information Theory*, vol. 37, no. 4, pp. 1019–1033, July 1991.
- [19] J. W. Hsieh, H. Y. Mark Liao, M. T. Ko, and K. C. Fan, “Wavelet–based Shape from Shading, ” *Graphical Models and Image Processing*, vol. 57, pp.343–362, July 1995.
- [20] J. W. Hsieh, H. Y. Mark Liao, K. C. Fan, M. T. Ko, and Y. P. Huang, “A New Edge–based Technique for Image Registration, ” *Computer Vision and Image Understanding*, Vol.67, No.2, pp.112–130, August 1997.
- [21] J. W. Hsieh, M. T. Ko, H. Y. Mark Liao, and K. C. Fan, “A New Wavelet–based Edge Detector via Constrained Optimization, ” *Image and Vision Computing*, Vol.15, pp.511–527, 1997.
- [22] C. J. Sze, H. Y. Mark Liao, H. L. Hung, K. C. Fan, and J. W. Hsieh, “Multiscale Edge Detection on Range Images via Normal Changes”, to appear in *IEEE Trans. on Circuits and Systems II, Special issue on Multirate Systems, Filter Banks, Wavelets, and Applications*.
- [23] S. Furui, “Digital Speech Processing, Synthesis, and Recognition, ” *Marcel Dekker Inc.*, pp. 244–251, 1989.
- [24] H. F. Silverman and D. P. Morgan, “the Application of Dynamic Programming to Connected Speech recognition, ” *IEEE ASSP Magazine*, vol. 7, Iss. 3, pp. 6–25, July 1990.

[25] B. Logan, "Information in the Zero-Crossings of Bandpass Signals," *Bell Syst. Tech. J.*, Vol. 56, P. 510, 1977.