

A New Method for the Synthesis of Signature Data With Natural Variability

Cedric Rabasse, Richard M. Guest, and Michael C. Fairhurst

Abstract—The collection of human biometric test data for system development and evaluation within any chosen modality generally requires significant time and effort if data are to be obtained in workable quantities. To overcome this problem, techniques to generate synthetic data have been developed. This paper describes a novel technique for the automatic synthesis of human handwritten-signature images, which introduces modeled variability within the generated output based on positional variation that is naturally found within genuine source data. The synthesized data were found to generate similar verification rates to those obtained using genuine data with the use of a commercial verification engine, thereby indicating the suitability of the data synthesized by using this method for a wide range of application scenarios.

Index Terms—Biometrics, data synthesis, signature.

I. INTRODUCTION

THE operational performance of all biometric systems used for the verification and identification of humans can be assessed by investigating the successful classification of genuine or nongenuine (e.g., forged) data [1]. Within the biometric research community, a constant goal is to identify new and improved sensors, performance-defining features, and classification algorithms to improve this operational system performance. A low error rate is generally a main priority and is critical, given the increasing prevalence of biometric systems deployed across ever larger populations.

The practical context of synthesizing these signatures is to enable the effective training and testing of biometric systems where test data are limited. Biometric systems are conventionally assessed by using reference data sets comprising samples captured from human participants for a biometric modality of interest. A meaningful evaluation of a system, with respect to both enrolment and usage issues, which is to be deployed in a large population, therefore requires a considerable amount of test data. Many classifiers (for example, neural networks) also generally require a large quantity of data to ensure an optimally trained and stable system prior to classification. To obtain this large-scale data from human participants requires significant time and effort (often with considerable financial implications), but subjects are also often unwilling to donate large numbers

of biometric samples due to concerns about possible security compromise, specifically if the data are to be released into the public domain. Similarly, the quality of samples may often be compromised through repetition and fatigue, particularly if a subject is asked to repetitively donate samples within a single session. To overcome a potential “data shortage” problem in practical situations, a number of studies have investigated synthesizing biometric data using either actual human samples as “seed” references or, in other approaches, data generated using typical patterns/images found within a modality but without reference to a particular individual [2]. Examples of synthesized modality studies (typically focused on image-based systems) include fingerprints [3], [4], iris [5], [6], and face [7].

Ma *et al.* [8] proposed three criteria to ensure that synthesized biometric data sets are representative of the real-life modality under study and that the artificially generated data have the “appearance” of being generated from a natural source. These criteria are flexibility, parsimony, and consistency. Flexibility implies a parameterization of the synthesis mechanism enabling variations in data representation. Parsimony aims at building a generation procedure that is as simple as the inherent complexity of data will allow. Once a generation model has been found to be flexible and simple, it is necessary to assess how well it fits the original data. The generated data must produce repeatable results within a verification system and must be representative of the original data that they aim to simulate (the consistency criterion). If this is not the case, the simulation study may result in unreliable output.

This paper describes a method of automatically generating static image-based representations of human handwritten signatures (i.e., representations based solely on visual appearance) for use in biometric-system analysis, using (two) reference signatures from an individual signer as models for the generation of synthetic data. Automatic signature verification is a commonly used form of biometric verification and identification [9] and is a modality which can offer many advantages over other forms of biometric authentication, among which is the familiarity among the general public for the everyday use of signatures to authorize transactions and verify identity. While other studies have outlined simple methods for generating synthetic signatures (such as fitting a series of Bezier curves between two sample images [10]) or have assessed the properties of signatures with reference to parameter modeling of forged data [11], [12], the approach outlined in this paper is based on dynamic time warping (DTW) techniques [13] and includes novel methods for dealing with sample signatures from the same signer but with different numbers of pen strokes. It also importantly introduces spatial variability throughout synthesized signatures based on that found within genuine

Manuscript received November 29, 2006; revised May 15, 2007. This work was supported by the EU Erasmus Exchange Program. This paper was recommended by Associate Editor V. Govindaraju.

C. Rabasse was with Laboratoire d'Informatique, de Traitement de l'Information et des Systèmes, University of Rouen, 76821 Rouen, France.

R. M. Guest and M. C. Fairhurst are with the Department of Electronics, University of Kent, CT2 7NZ Canterbury, U.K. (e-mail: r.m.guest@kent.ac.uk).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TSMCB.2008.918575

signatures, resulting in images which encapsulate the natural variability found within hand-drawn signatures rather than a simple morphological mapping between two images. Our approach uses both the static (spatial) and dynamic (temporal) information within signatures, which is stored during online capture of data to generate the static synthesized images and which is generally included in sample databases. The temporal data enable the construction order to be assessed, which enables an accurate mapping between drawing sequences. Following this mapping, all subsequent operations occur in the spatial domain.

Finally, we assess the quality of the resulting synthesized images using samples from a standard online public-domain signature database (Signature Verification Competition 2004 (SVC2004) [15]) as seed signatures and a commercial static signature verification engine as a “black-box” recognition system. In the experiment, we assess the verification performance achieved when using both genuine and forged signature images, synthesis images with variability, and, as a comparison, a second simpler method for synthesizing signature images.

II. SYNTHESIS-WITH-VARIABILITY OVERVIEW

Biometric signature samples are usually collected either as images (static representation) or as a chronological sequence of data vectors representing a user's signature execution pattern, which are sampled by using a capture device such as a graphics tablet (dynamic representation). Although, in this paper, we focus on the generation of static signature images, we utilize dynamic data to obtain the positional sequence of drawn points in order to introduce naturally occurring spatial variability within a temporally sequenced mapping phase. Specifically, we utilize three data streams from the captured dynamic data: the x and y pen coordinates and a binary pen-contact value indicating when the pen is in contact with the tablet surface. Using these three data streams, we can represent the signatures in a bidimensional space by a linear interpolation.

Our method of generating images is to use a warping transform between two genuine signatures (seed signatures) from the same subject. Fig. 1 shows two example seed signatures from the same subject, which have been generated by plotting the x and y data streams from respective dynamic signature samples. It is apparent that while the general structure and shape of the two signatures show similarities, there are differences which can be exploited in the production of the synthesized signatures. Indeed, our method assesses the variability across a number of further samples with respect to the two seed signatures to enhance the subject-specific variability model.

Each intermediate warped signature, from the first seed signature $S_1 = \{S_1^1, S_1^2, \dots, S_1^m\}$ with m data vectors to the second seed signature $S_2 = \{S_2^1, S_2^2, \dots, S_2^n\}$ with n data vectors, will be the basis of a synthesized signature. S_i^t represents a two-component vector (the coordinates of a point) from the signature i at a notional time t (the t th acquired point).

Rather than simply calculate an interpolation of intermediate points, naturally occurring variability can be added to the synthesized signatures. The variability applied at a particular location within a warped signature is the location-specific observed

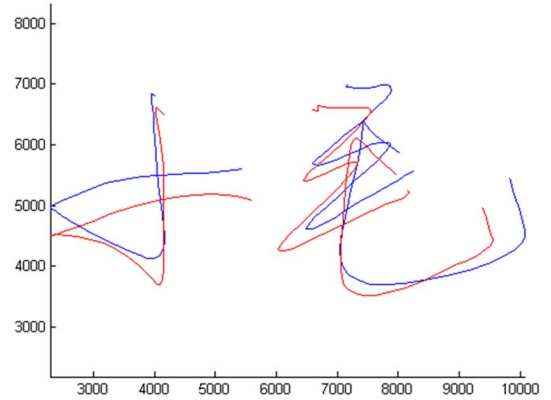


Fig. 1. Two seed signatures from the same signer.

variability across a number of a subject's signature samples. In this way, we can create a synthesized signature as a combination of two signatures onto which a randomly applied modeled variability is added. Each stage in the synthesis process is now explained in detail.

A. Normalization: A Preprocessing Stage

The initial stage of the synthesis process is to select two seed signatures from a single signer and to plot the x and y coordinates from the data stream when the pen was identified as being on the tablet surface. An individual subject's signature can vary in size (width and height) and also with respect to the starting location on the graphics tablet surface, and hence, because of this variability, we need to size- and position-normalize the two seed signatures in order to accurately calculate a mapping between signature points. This normalization transformation can be removed once the mapping between the two signatures has been generated and, hence, once the original size and position information has been reinstated.

Let S'_1 and S'_2 denote the normalized signatures.

Each signature S'_i is shifted to the origin, x' coordinates are limited to a $[0; X_{\text{lim}}]$ range, and y' coordinates are limited to a $[0; Y_{\text{lim}}]$ range. Therefore

$$x' = X_{\text{lim}} \cdot \frac{x - X_{\min}}{X_{\max} - X_{\min}},$$

$$X_{\min} = \min_x \{x\} \text{ and } X_{\max} = \max_x \{x\}$$

$$y' = Y_{\text{lim}} \cdot \frac{y - Y_{\min}}{Y_{\max} - Y_{\min}},$$

$$Y_{\min} = \min_y \{y\} \text{ and } Y_{\max} = \max_y \{y\}.$$
(1)

To simplify the implementation, we chose the linear mapping of $X_{\text{lim}} = Y_{\text{lim}}$, although a nonlinear mapping (modifying each axis independently) may be used and may result in enhanced performance where axis dimensionality is dissimilar. Hence, the signature coordinates are included within a square area.

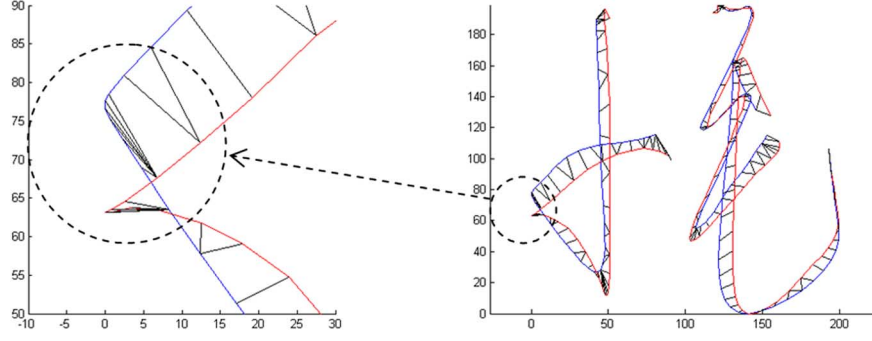


Fig. 2. Mapping with Euclidean DTW.

B. DTW

In our method for the synthesis of signature images, we identify a mapping of each point in the x and y data streams from the first seed to another point within the second seed signature. This mapping must optimize (resulting in a lower value of) a criterion measuring the global deformation made between the two signatures.

We calculate an edit distance $c(S'_1, S'_2)$ [16] between the two normalized coordinate sequences from the two seed signatures. In our case, the edit distance is the Levenshtein distance [17] measuring the overall optimal mapping between signatures as the sum of each individual optimal mapping between pairs of points within the two sequences. In this way, a large single mapping distance within the signature does not significantly affect the overall distance between the signatures.

To compute the mapping between the two seed signatures, we use the DTW technique. Given the two sequences $S'_1 = \{S_1^1, S_1^2, \dots, S_1^m\}$ and $S'_2 = \{S_2^1, S_2^2, \dots, S_2^n\}$, the standard DTW algorithm computes the elements of the edit matrix $D(i, j)$ of dimension $(m+1) \times (n+1)$ by using the following recursion formulas:

$$\begin{aligned}
 D(0, 0) &= 0 \\
 D(i, 0) &= D(i-1, 0) + d(S_1^i, (0, 0)), \quad \text{for } i = 1, \dots, m \\
 D(0, j) &= D(0, j-1) + d((0, 0), S_2^j), \quad \text{for } j = 1, \dots, n \\
 D(i, j) &= \min \left\{ \begin{aligned} &D(i-1, j-1) + 2 \cdot d(S_1^i, S_2^j) \\ &D(i-1, j) + d(S_1^i, S_2^j) \\ &D(i, j-1) + d(S_1^i, S_2^j) \end{aligned} \right\}, \\
 \text{with } d(x, y) &= \|x - y\| = \sqrt{(x - y)' \cdot (x - y)}.
 \end{aligned} \tag{2}$$

Here, $d(x, y)$ is a simple Euclidean distance commonly used in DTW methods. In the normalized signatures, this distance can be considered simply as a local measure of deformation between two points.

Finally, $D(m+1, n+1) = c(S'_1, S'_2)$. The sequence of pairs (i, j) leading to this value can be found by backtracking

and represents the mapping transform between S'_1 and S'_2 . Therefore, it also represents the mapping transform between S_1 and S_2 .

C. From Euclidean DTW to Derivative DTW

For the majority of mapping cases, the Euclidean distance provides an adequate metric to identify the nearest coordinate point from the first image to the second one. There are instances when the two images are mapped, however, where the local structure and overall shape of the signatures differ considerably. These so-called “singularities” [18] result in a single point on one signature being mapped to a larger cluster of points on the second signature. These singularities are due both to the relative positions of the signatures and to the local variations made by a user. Fig. 2 shows an example of this, where the lines between the two signature images indicate the identified mapping between two individual points. It can be seen that, in some instances, there is a mapping between a single point and many points. During the formation of the synthesized signatures using the mappings, singularities result in the creation of image artifacts which appear unnatural and do not follow a “smooth” ink path.

To avoid this problem, we do not directly use the coordinates of a point but instead use the tangent vector representing the variation at each point. The tangent vector is more specific to an individual’s signature as it reflects more clearly the evolution of the points of both sequences. Each tangent vector is also a two-component vector that can be used in the measure of the Euclidean distance.

We use simple estimation of the tangent based upon the previous and the next points. Consider a normalized signature S'_i which consists of n points. The transformation is calculated using the following:

$$S_i^{tt} \mapsto S_i^{tt+1} - S_i^{tt-1}, \quad t = 2 \dots (n-1). \tag{3}$$

For the sake of simplicity, the derivative DTW algorithm still uses the Euclidean distance as an edit distance between two tangents of two signatures: The lower the Euclidean distance, the more similar the tangents. Fig. 3 shows the improved results when using this method on the same normalized seed signatures. Again, each line between seed images represents a mapping between two points.

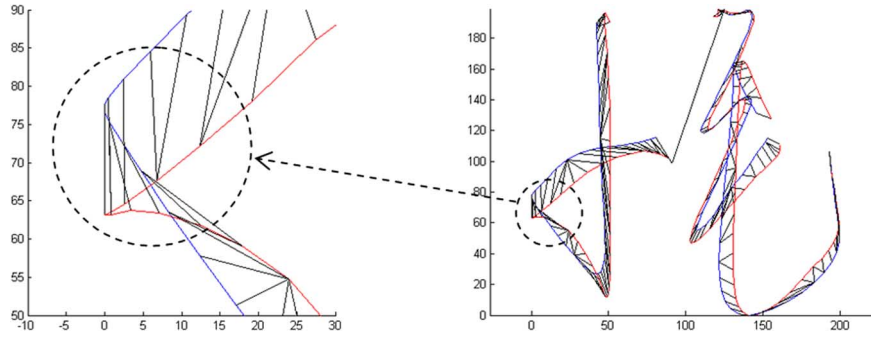


Fig. 3. Mapping with derivative DTW.

D. Synthesized Signatures

Once the mapping between the two random seed signatures has been calculated, the normalized seed signatures can be discarded. The derived mapping is now applied to the original coordinate sequences, thereby restoring the original size information.

Let k be the number of signatures to be synthesized. The l th synthesized point within the q th synthesized signature (S_q^l) can be calculated as

$$S_q^l = S_1^i + \frac{q}{k+1} \cdot (S_2^j - S_1^i) \quad (4)$$

where S_1^i and S_2^j represent the i th and j th vectors of S_1 and S_2 , respectively (this method is generally called a median string or a weighted mean of strings [19]–[21]).

E. Stroke Separations

While the warping process results in the synthesis of signatures containing a series of coordinate points, these individual points need to be connected to indicate the presence of ink on the page. Many signatures comprise several “strokes,” each with a pen removal from the writing surface separating in-between each stroke. It is therefore necessary to know whether the pen is up or down at a particular point in the synthesized signature. In determining the separate strokes within a synthesized image, a problem can occur when the two seed signatures do not contain the same number of strokes. In creating a signature, a signer may remove the pen from the tablet during a particular letter-formation execution, which may not be present in other signatures, hence creating an undesirable separation within the curvature of a letter.

When the original signatures are captured, the graphic tablet records whether the pen is up or down by means of a simple binary flag. Therefore, we can compare the number of times that the pen is removed from the tablet surface within the two seed signatures. The number of separations assigned to an individual synthesized signature depends on whether it is “closer” in terms of order of synthesis to the first or second seed signature. Consider the q th warped signature over k synthesized signatures. If $(q/(k+1)) < 0.5$, the number and the position of separations will be mapped with those of the first signature S_1 . If $(q/(k+1)) > 0.5$, the number and the position of separations will be mapped with those of the second signature S_2 . If $(q/(k+1)) = 0.5$, the signature is assigned

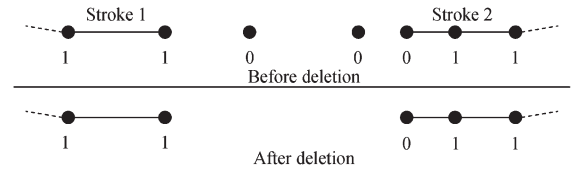


Fig. 4. Illustration of the deletion of isolated points between two strokes of a synthesized signature with their respective pen-up value.

the position and number of separations contained in S_1 or S_2 according to whichever seed that has the lower number of separations. This assumption is made on the basis that the lowest number of separations represents the most stable and realistic representation of a user's signature.

Having applied these criteria, a second problem can arise. The method described previously can generate several small sequences of synthesized “isolated points” (i.e., synthesized points that should not be connected to a stroke or are separated by a greater than normal distance from the end of a stroke).

Fig. 4 shows an example of this problem. In a certain part of a synthesized signature, two individual strokes are required to be separated. However, erroneous isolated points are generated (a pair of isolated points is shown in the top half of Fig. 4). To determine whether a point is isolated, we assess the pen-contact values of the nearest mapped point within the seed signature. The binary pen-contact value signifies, through a sequential change of 0 to 1, the beginning of a new stroke. When a sequence of isolated points is identified, only the last point with a pen-contact value of 0 is retained, as this indicates the beginning of the next stroke. Other isolated points within the synthesized signature are deleted.

A final problem to solve when adding separations is the deletion of incorrect mappings. The final point of a stroke can be mapped with the first point of the next stroke because of the use of the tangent at each point rather than the position. Therefore, after the deletion of isolated points, some endpoints of a stroke can result in incorrect mappings (Fig. 3 shows an example of this where the endpoint of the first section is mapped to the first point of the second section).

One way of removing these errors is to consider “bounding lines” identified individually at the start and endpoints of the separate strokes within the two seed signatures. Taking the two seed signatures, the two endpoints of the first stroke are identified. A bounding line is defined as being the joining line between the two points, which also extends beyond the points.

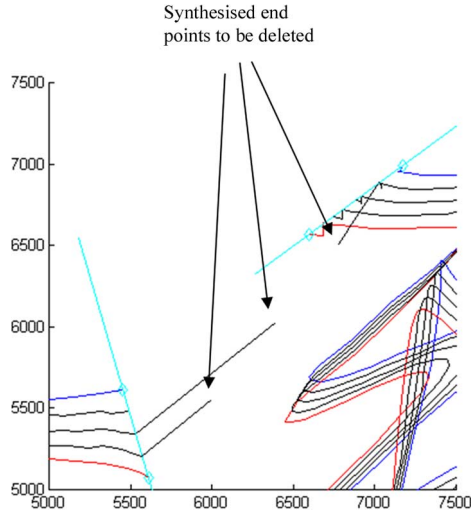


Fig. 5. Stroke bounding lines and undesired points requiring removal.

This process is repeated for each start and endpoint pair for subsequent strokes.

In the detailed example shown in Fig. 5, the left bounding line is constructed with the endpoints of the first stroke of the two signatures, and the right bounding line is constructed with the start points of the second stroke.

Any synthesized points not positioned on these bounding lines which signify the start and endpoints of a stroke within the synthesized signature (such as the three examples shown in Fig. 5) are subsequently deleted.

F. Adding Variability

Adding variability is the final stage in the synthesis process. Unlike the previous stages, this stage is based on randomization in order to introduce natural variability within synthesized signature images. A small deviation δ_q^l is added to each vector S_q^l of the q th synthesized signature. This deviation is specific both to an individual subject and also to the position within the signature. It is calculated from a topographic map—a grayscale normalized image representing the concentration of the pixel distributions at each position. Consider a black (0-filled) normalized image T . Twenty genuine signatures of a signer were normalized to fit T . Every drawn ink position from every image is set to white (255 grayscale value) in T .

The second stage is to estimate the variation of the distribution of white pixels within T using Gaussian filtering. The size and the variance of the Gaussian window were determined empirically. For the purposes of this experiment, we fixed an 11×11 pixel window with a standard deviation $\sigma = \sqrt{3}$. For the purposes of speed in execution, a relatively small window was selected, but the convolution process was applied five times to enhance smoothing. Although the use of these window parameters resulted in improved overall performance system, optimal windowing values can be determined through further experimentation.

After the convolution of T with a Gaussian filter, the resultant image (see Fig. 6 for an example) gives an approximation of the concentration at each position relative to its neighborhood.

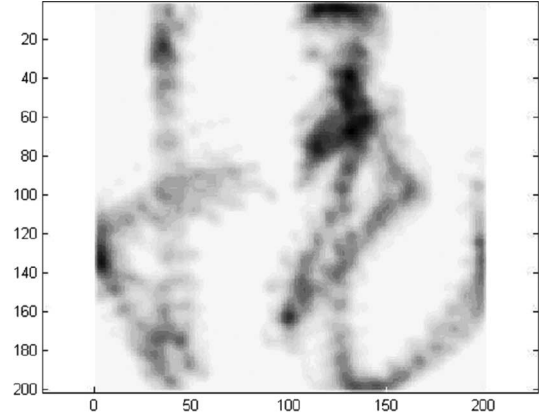


Fig. 6. Topological map from a single signer (the darker the pixel location, the higher the concentration).

For the q th synthesized signature, we map the position of a point S_q^l to a point in T , and we get an estimation $V_{S_q^l}$ of the variance to be used. We calculate the unit vector $N_{S_q^l}$ that is orthogonal to the tangent vector at S_q^l and weight it by $V_{S_q^l}$ times a weighted coefficient c_l (allowing a degree of control in relation to the amount of variability applied) following a weighted uniform distribution

$$\delta_q^l = c_l \cdot V_{S_q^l} \cdot N_{S_q^l} \quad (5)$$

where $c_l \propto U([-1; 1])$. Then

$$S_q^l \leftarrow S_q^l + \delta_q^l.$$

Hence, we add modeled variability (equivalent to operating in an environment with uniform noise) to a basic synthesized signature according to the variability that is naturally occurring within a user's signature samples.

G. Smoothing the Variability

Adding the randomly controlled modeled variability as described previously can lead to severe variations in the position of two consecutive points. Fig. 7 shows this effect. In order to avoid the creation of totally independent variations between consecutive points, the coordinates of a point S_q^l are weighted with those of its previous and next neighbor points. The position of a point is convolved with a Gaussian kernel according to the following:

$$S_q^l \leftarrow \frac{1}{W} \cdot \sum_{i=-PT}^{PT} e^{-\frac{i^2}{\sigma^2}} \cdot S_q^{l+i} \quad W = \sum_{i=-PT}^{PT} e^{-\frac{i^2}{\sigma^2}}. \quad (6)$$

Here, PT represents the number of previous and next points in the sequence used in the smoothing process. In the case of $PT = 1$, we take the position of the immediate neighborhood points into account to weight the position of the current point. σ represents the standard deviation of the Gaussian kernel and is fixed empirically at $\sigma = 2$: The higher the value of σ , the greater the effect on the current point of its neighbors' positions.

Smoothing is applied to each stroke of a synthesized signature separately. The start and endpoints of each stroke are not affected by this transformation to avoid modification by

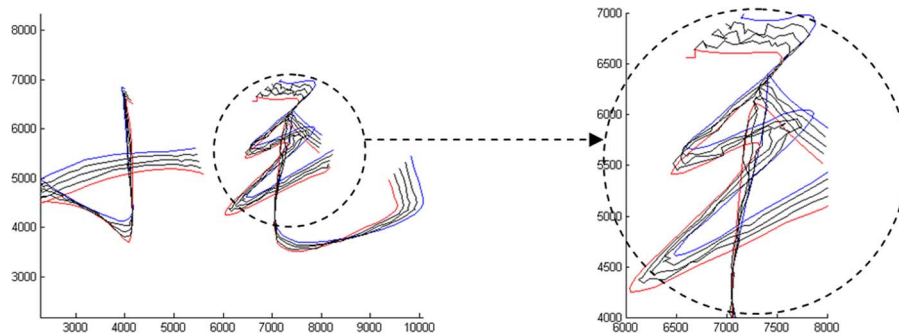


Fig. 7. Randomly controlled modeled variability added to three different synthesized signatures.

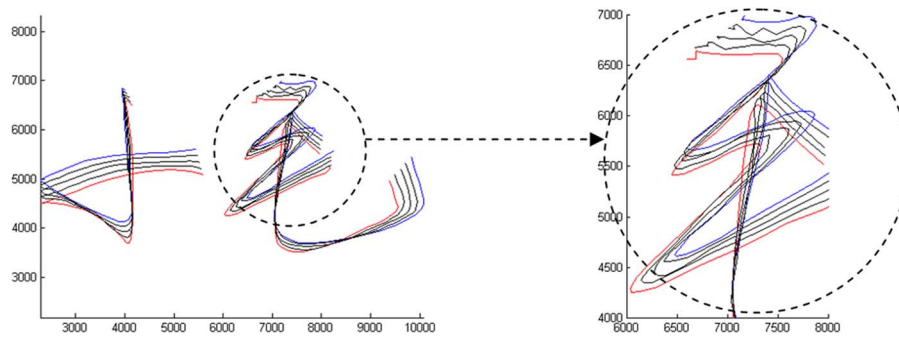


Fig. 8. Smoothing the variability of a synthesized signature.

points within other strokes. If $PT = z$, then the first and last z positions in a stroke are not filtered. Fig. 8 shows the effects of this smoothing procedure.

III. COMPARATIVE SYNTHESIS METHODS

In order to compare the performance of our devised synthesis-with-variability method, a second simpler method, based on a combination of forward and reverse nearest point morphing [14], was also implemented. Again, this method is based on two randomly selected genuine signatures that act as seed references. Following signature size normalization, each individually drawn ink coordinate point within the first signature is mapped to the nearest (measured by Euclidean distance) point within the same segment on the second signature. This process is then reversed by matching all points on the second signature to the nearest points on the first signatures (again within the same segment). This “reversed” process may not result in an identical mapping to the first mapping as more than one point on the second signature may be mapped to by the first. Fig. 9 shows this mapping process.

To generate n synthesized signatures, the distance between an individual mapping is divided into n transitional points. The first $n/2$ transitional points from each of the two separate mappings are used as the individual points within a synthesized signature. In Fig. 10, $n = 4$. The first two transitional points (shown by the dots) for each of the mappings are used to synthesize four signatures.

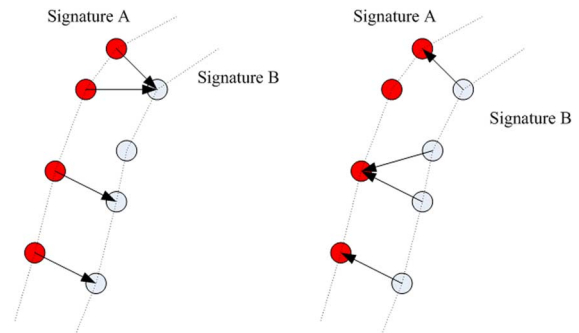


Fig. 9. Signature A-to-B and B-to-A mapping between points.

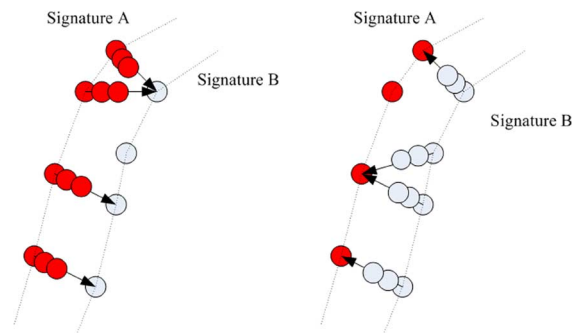


Fig. 10. Example points from the two mappings used to generate four signatures.

IV. EXAMPLE SYNTHESIS-WITH-VARIABILITY OUTPUT

Fig. 11 shows a number of examples of seed and synthesized signatures. Data used to generate the modeled variability were

collected across 20 signatures from the same signer, including the two seed signatures. It can be noted that the signature shape and stroke position is preserved alongside size and dimensional aspects. The variation within each synthesized set is obviously





















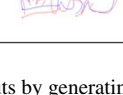
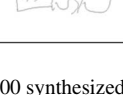
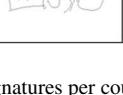
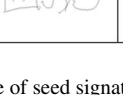
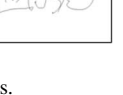
Seed signatures	20 th synthesised signature	40 th synthesised signature	60 th synthesised signature	80 th synthesised signature
				
				
				
				
				

Fig. 11. Examples of signature outputs by generating 100 synthesized signatures per couple of seed signatures.

dependent on the similarity in seed signatures with similar seeds resulting in a set of signatures which, on initial inspection, appear uniform.

V. QUALITY EXPERIMENTATION

In order to assess the quality of the synthesized images, a third-party commercial static signature engine was used as a representative system within a typical verification scenario. The engine was treated as a blind assessment “black box” in the sense that no system-specific optimization was performed (indeed, data optimization in terms of signature size, rotational correction, signature complexity, etc., for performance maximization was unknown); pairs of signatures were presented in the required image bitmap format, and the verification engine result was noted. One mode of operation of the selected verification engine is to provide a binary decision on whether a sample signature is genuine when compared against a single reference signature. The measure of confidence associated with this binary result is also returned (a default confidence threshold value, as set by the manufacturer, of over 80 out of 100 indicates a genuine signature). As the aim of the experiment is to assess the quality of signature samples rather than the performance of the verification engine, this confidence value was not modified or optimized during experimentation.

All signatures used in this trial were obtained from the publicly available and widely adopted signature database used in SVC2004 [15]. Using data set 1 from this database, this data set consists of 1600 text files containing separate signature in the form of time-stamped x , y , and binary pen-on-tablet sequences. Forty separate signers are represented in the data set. For each user, 20 files represent genuine signatures, and the 20 others represent skilled forgeries.

In our experiments, we first obtain the baseline performance of the signature engine by finding the error rates in terms of false-accept and false-reject rates (FARs and FRRs,

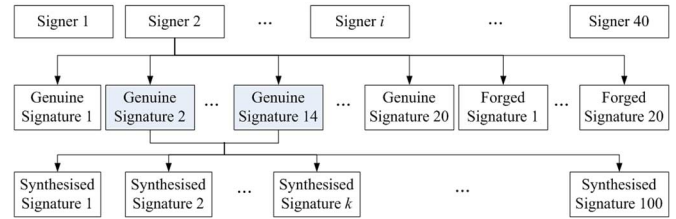


Fig. 12. Experimental signature generation.

respectively) at the 80 confidence level when matching the 18 remaining genuine signatures and 20 forged signatures for a particular subject against the two randomly selected signatures used as seeds for synthesis. Following this, we can verify a set of synthesized “genuine” signatures against genuine and forged data, thereby assessing the quality of the synthesized data set in terms of static validation against known samples.

The experiment was conducted in three steps.

A. Step 1: Signature Synthesis

Every genuine and forged signature (a total of 40 per signer) from each of the 40 individual signers was read from the SVC2004 data set 1 database and converted into a static bitmap image. Bitmap signature images were created from the pen coordinate sequences simply by connecting pen-down coordinate pairs using Bresenham’s line drawing algorithm. For each signer, two genuine signatures were randomly chosen as seed signatures, from which 100 synthesized signatures were generated by using both the synthesis-with-variability and simple synthesis methods. All signatures were stored as bitmap files, resulting in a total of $40 \times 20 = 800$ genuine signatures, $40 \times 20 = 800$ forged signatures, and $40 \times 100 = 4000$ synthesized signatures. Fig. 12 shows this experimental configuration. Note that the seed signature pair is randomly and independently chosen for each separate signer.

TABLE I
ERROR RATE RESULTS USING THE TWO SYNTHESIS METHODS

Experiment	FRR				FAR			
	Average	Min	Max	Std Dev	Average	Min	Max	Std Dev
Baseline	15.45%	12.22%	18.54%	1.39%	52.65%	47.69%	57.69%	2.31%
Synth (Variable)	14.47%	11.84%	17.19%	1.52%	52.34%	49.31%	57.50%	2.22%
Synth (Simple)	54.96%	51.60%	57.86%	1.57%	20.21%	16.58%	24.97%	2.00%

B. Step 2: Baseline System Performance Evaluation

Assessing the performance of the signature verification engine using genuine and forged signatures from the database gives a reference value against which its performance can be measured when presented with the synthesized signatures. For each signer in the database, the verification engine individually compared each of the two randomly selected seed signatures used to generate the synthetic images: first, against the 18 other genuine signature images, and second, against the 20 forged signatures for a particular signer.

This experiment was repeated 800 times in total, using 20 different pairs of seed signatures for each signer and across all 40 signers in the database. By selecting different seeds, it is possible to establish the effect of seed choice on system performance. This resulted in $2 \text{ (seeds)} \times 38 \text{ (genuine + forged)} \times 800 = 60\,800$ comparisons being made. Using the manufacturer-recommended signature verification cutoff value of 80 to distinguish between a forged and a genuine decision, FARs and FRRs can be calculated.

C. Step 3: Evaluating the Performance of the Signature Synthesis

The performance evaluation for the synthesized signatures is similar in structure to Step 2. For each individual signer, the verification engine compares each of the 100 synthesized images with each of the 20 genuine signatures (including the two seed signatures) and then each of the 20 forged signatures for a particular signer. Again, the process was repeated 800 times using the 20 synthesized sets for each signer and across all 40 signers in the database, resulting in a total of $[(100 \text{ synthesized signatures} \times 20 \text{ genuine signatures}) + (100 \text{ synthesized signatures} \times 20 \text{ forged signatures})] \times 800 = 3\,200\,000$ comparisons. Again, the FAR and FRR were calculated by using the signature engine, with the synthesized signatures being considered as “genuine” data. Although there is a mismatch in the number of comparisons made within Steps 2 and 3 of the experiment, comparing the percentages of signatures considered as genuine (alongside the mean confidence measure across an assessment of all signatures) enables a thorough comparison of performance using all the synthesized signatures.

This process was repeated separately for the second simpler synthesis method, as described in Section III, to enable a synthesized performance comparison.

method is very similar in performance to the baseline results (conversely, the simple synthesis method produces error rate results which are very different). The remainder of this paper will therefore focus solely on the synthesis-with-variability signatures.

By statistically assessing the error rates across the experiments, it is possible to determine quantitatively whether they are representative of the original signatures, thereby establishing their validity as data for experimental studies. If the distribution of error rate results given by comparing the seed signatures with the genuine and forged signatures is similar to the distribution of error rate results in comparing the genuine and forged signatures with the synthesized signatures, it can be concluded that the synthesized signatures are representative of the genuine signatures.

To confirm the similarity of the group results, we use a two-sample Kolmogorov–Smirnov (KS) test across the error rates of the 800 experiments. This test uses the maximum absolute value observed between the two cumulative frequency curves representing the estimated cumulative distribution of both comparison results in our case. The lower this absolute value, the higher the likelihood that both distributions are identical. The KS test has the advantage of making no assumption about the distribution of data: It is nonparametric and distribution free. Therefore, it is less powerful than a parametric test but more robust to assess the validity of the null hypothesis. In our case, the null hypothesis H_0 states that, for the j th pair of seed signatures $\text{seed}_{i,j}$ from user i , the distribution of error rate results given by comparing $\text{seed}_{i,j}$ with the other 18 genuine and 20 forged signatures is identical to the distribution of error rate results given by comparing the 20 genuine and 20 forged signatures from user i , with the 100 synthesized signatures built from $\text{seed}_{i,j}$. The alternative hypothesis H_1 states the contrary: the distributions represent different populations. The test returns a p -value which represents the probability of observing a given result, or one more extreme, by chance if the null hypothesis is true. Small values of p cast doubt on the validity of the null hypothesis.

The results show that the error rates of the baseline and simple synthesis experiments are from significantly different populations (both FAR and FRR have $p < 0.001$), whereas for the variable synthesis signatures, the error rates can be said to be drawn from the same population (FRR has $p = 0.051$, whereas FAR has $p = 0.694$).

VII. CONCLUSION

We have outlined a novel method for automatically generating static synthesized signature images from two seed images. The method uses additional data collected from other genuine signatures to establish the variability at particular points within

VI. EXPERIMENTAL RESULTS

Table I shows the error rate results of the experiments, showing the average, min, max, and standard deviations over the 800 experiments. It can be seen that variable synthesis

normalized versions of an individual's signature. These data are used to add a modeled variability to individual points within the synthesized signature, which are smoothed to obtain a naturally variable signature image.

The resultant images have been shown to generate similar verification results as are obtained with genuine signatures by using a commercial signature verification engine, confirming their value as synthesized images that are suitable for a variety of experimental purposes. They have also shown to offer a significant improvement over a simpler nearest point morphing method of generating synthesized signatures.

Considering the validity of the resulting samples, it is useful to reiterate that we have met the three established criteria for the generation of synthesized biometric data: our data embrace flexibility, demonstrating naturally occurring variability within a signer's signature where the samples are generated by using a simple methodology relative to the complexity of the task; moreover, the data have been shown accurately to represent genuine data.

Alongside an optimization of the Gaussian parameters and an assessment of nonlinear size normalization, further work in this area will concentrate on extending the developed methods to the synthesis of dynamic signature data, intelligently synthesizing pen locations and timings based on the natural variability within characteristics such as pen velocities and tilt and force ranges, and on recreating representative pen traces (such as variable-ink-thickness traces according to pressure and texture as a function of velocity) within the synthesized images.

REFERENCES

- [1] A. K. Jain, A. Ross, and S. Prabhakar, "An introduction to biometric recognition," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 14, no. 1, pp. 4–20, Jan. 2004.
- [2] N. M. Orlans, D. J. Buettner, and J. Marques, "A survey of synthetic biometrics: Capabilities and benefits," in *Proc. IC-AI*, 2004, vol. I, pp. 499–505.
- [3] R. Cappelli, "Synthetic fingerprint generation," in *Handbook of Fingerprint Recognition*, D. Maltoni, D. Maio, A. K. Jain, and S. Prabhakar, Eds. New York: Springer-Verlag, 2003, pp. 203–232.
- [4] D. R. Cappelli and D. Maio, "Synthetic fingerprint-database generation," in *Proc. 16th ICPR*, 2002, vol. 3, pp. 744–747.
- [5] J. Cui, Y. Wang, J. Huang, T. Tan, and Z. Sun, "An iris image synthesis method based on PCA and super-resolution," in *Proc. 17th ICPR*, 2004, vol. 4, pp. 471–474.
- [6] A. E. Lefohn, R. T. Caruso, E. Reinhard, B. Budge, and P. Shirley, "An ocularist's approach to human iris synthesis," *IEEE Comput. Graph. Appl.*, vol. 23, no. 6, pp. 70–75, Nov./Dec. 2003.
- [7] B. Chalmoud, "Modeling and inverse problems in image analysis," in *Applied Mathematical Sciences*, vol. 155. New York: Springer-Verlag, 2003.
- [8] Y. Ma, M. Schuckers, and B. Cukic, "Guidelines for appropriate use of simulated data for bio-authentication research," in *Proc. 4th IEEE Workshop Autom. Identification Adv. Technol.*, 2005, pp. 251–256.
- [9] A. Jain, F. Griess, and S. Connell, "On-line signature verification," *Pattern Recognit.*, vol. 35, no. 12, pp. 2963–2972, 2002.
- [10] S. Yanushkevich, A. Stoica, S. Srihari, V. Shmerko, and M. Gavrilova, "Simulation of biometric information: The new generation of biometric systems," in *Proc. Int. Workshop BT*, 2004, pp. 87–98.
- [11] M. Ammar, "Progress in verification of skillfully simulated handwritten signatures," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 5, pp. 337–351, 1993.
- [12] J. Brault and R. Plamondon, "A complexity measure of handwritten curves: Modeling of dynamic signature forgery," *IEEE Trans. Syst., Man, Cybern.*, vol. 23, no. 2, pp. 400–413, Mar./Apr. 1993.
- [13] C. S. Myers and L. R. Rabiner, "A comparative study of several dynamic time-warping algorithms for connected word recognition," *Bell Syst. Tech. J.*, vol. 60, no. 7, pp. 1389–1409, Sep. 1981.
- [14] G. Wolberg, *Digital Image Warping*. Los Alamitos, CA: IEEE Comput. Soc. Press, 1990.
- [15] D.-Y. Yeung, H. Chang, Y. Xiong, S. George, R. Kashi, T. Matsumoto, and G. Rigoll, "SVC2004: First international signature verification competition," in *Biometric Authentication*, vol. 3072/2004. Berlin, Germany: Springer-Verlag, 2004, pp. 16–22.
- [16] R. A. Wagner and M. J. Fisher, "The string to string correction problem," *J. Assoc. Comput. Mach.*, vol. 21, no. 1, pp. 168–173, 1974.
- [17] V. I. Levenshtein, "Binary codes capable of correcting deletions, insertions and reversals," *Sov. Phys. Dokl.*, vol. 10, no. 8, pp. 707–710, 1966.
- [18] E. J. Keogh and M. J. Pazzani, "Derivative dynamic time warping," in *Proc. 1st SIAM Int. Conf. and Data Mining*, 2001.
- [19] X. Jiang, H. Bunke, K. Abegglen, and A. Kandel, "On the weighted mean of a pair of strings," *Pattern Anal. Appl.*, vol. 5, no. 1, pp. 23–30, 2002.
- [20] X. Jiang, H. Bunke, K. Abegglen, and A. Kandel, "Curve morphing by weighted mean of strings," in *Proc. 16th ICPR*, 2002, vol. 4, pp. 192–195.
- [21] X. Jiang, H. Bunke, and J. Csirik, "Median strings: A review," in *Data Mining in Time Series Databases*, M. Last, A. Kandel, and H. Bunke, Eds. Singapore: World Scientific, 2004, pp. 173–192.

Cedric Rabasse, photograph and biography not available at the time of publication.



Richard M. Guest received the B.Eng. degree in computer science from the University of York, York, U.K., in 1995, and the Ph.D. degree in electronic engineering from the University of Kent, Canterbury, U.K., in 2000.

He has been a member of the faculty at the University of Kent since 2000, where he is currently with the Department of Electronics. He has published widely in the academic literature and served on numerous publication, conference, standards development and research council review committees in this

field. His research interests include hand-drawn data analysis for biometric signature verification and forensic applications and neuropsychological testing; statistical classification systems, and biometric integration.



Michael C. Fairhurst is currently the Head of the Department of Electronics, University of Kent, Canterbury, U.K. His current projects include works on multimodal biometrics, on verification engines for biometrics, on assessing the vulnerability of biometric identification techniques, and on the analysis of handwriting, both for identification purposes and the improvement of the effectiveness of automated processing for forensic applications. Biometric processing also underpins work which is investigating document encryption that is linked to

biometric data. In related work, he is further developing work that he pioneered at Kent, which has established novel techniques for the assessment and monitoring of neurological conditions through the analysis of patients' writing and drawing abilities. He has served on numerous conference, workshop, and government committees and on the editorial boards of several international journals. He has also published some 350 papers in the scientific literature. His research interests focus on computational architectures and algorithms for image analysis and classification and on applications including handwritten-text reading and document processing, medical image analysis, and, specifically, security and biometrics.

Prof. Fairhurst is an elected Fellow, in recognition of his contributions to the field, of the International Association for Pattern Recognition.