International Journal of Pattern Recognition
and Artificial Intelligence
Vol. 18, No. 7 (2004) 1339–1360

World Scientific
www.worldscientific.com

OFFLINE SIGNATURE VERIFICATION AND IDENTIFICATION USING DISTANCE STATISTICS

MEENAKSHI K. KALERA*

Department of Computer Science and Engineering,
SUNY at Buffalo, Buffalo, New York, USA
mkkalera@cedar.buffalo.edu

SARGUR SRIHARI† and AIHUA XU‡

CEDAR, SUNY at Buffalo, UB Commons,
520 Lee #202, Buffalo, New York, USA
† *srihari@cedar.buffalo.edu*
‡ *aihuaxu@cedar.buffalo.edu*

This paper describes a novel approach for signature verification and identification in an offline environment based on a quasi-multiresolution technique using GSC (Gradient, Structural and Concavity) features for feature extraction. These features when used at the word level, instead of the character level, yield promising results with accuracies as high as 78% and 93% for verification and identification, respectively. This method was successfully employed in our previous theory of individuality of handwriting developed at CEDAR — based on obtaining within and between writer statistical distance distributions. In this paper, exploring signature verification and identification as offline handwriting verification and identification tasks respectively, we depict a mapping from the handwriting domain to the signature domain.

Keywords: Offline; GSC features; Bayes classifier; k-nearest neighbor; skilled forgeries.

1. Introduction

Research on online signature verification systems is widespread while those on offline are not many. Offline signature verification is not a trivial pattern recognition problem when it comes to skilled forgeries. This is because, as opposed to the online case, offline signatures lack any form of dynamic information. Also, the performance of the two systems is not directly comparable due to this difference. In the past some authors have worked on simple forgeries while others have dealt with the verification of skilled forgeries. Our present work deals with the verification of skilled forgeries.

*Address for correspondence: Center of Excellence for Document Analysis and Recognition, UB Commons, 520 Lee #202, State University of New York at Buffalo, Amherst, NY 14228-2583, USA.

The field of offline signatures and automatic writer verification and identification based on offline handwriting has been studied for several years now. Forensic document examiners, e.g. Robertson, 1991, Russell, *et al.*, 1992, Hilton, 1993, Huber and Headrick, 1999, have cited this methodology in their textbooks. A correspondence can be drawn between this writer identification process and offline signature verification, identification due to the fact that individuality analysis of words is similar to signature verification. The theory of individuality of words has been delved into and the method of statistical distance distributions has been developed and experimented at CEDAR.

Signature verification, a behavioral biometric, is characterized by a behavioral trait that is learnt and acquired over a period of time rather than a physiological characteristic. This is true of handwriting too. Moreover, performing cursive handwriting verification or signature verification is subjective rather than being objective.[28] A signature is a unique identity of a person. No two signatures can be identical, unless one of them is a forgery or a copy of the other. Also however hard one may try, no two signatures will be exactly identical. This gives rise to intrapersonal variation, that is, the variation within a class of genuine signature samples of an individual. At the same time, we also need to consider another type of variability — interclass variability, which can be defined as the variation between classes of genuine signature samples of two different individuals. To classify two signatures as belonging to the same class or two different classes, the intraclass scatter should be as low as possible, whereas the interclass scatter should be as high as possible. Capturing this variability is the function of the feature extraction process in any verification system.

Many approaches based on neural networks,[4] Hidden Markov Models and other structural algorithms[3,27] regional correlation[21] have been discussed earlier. The paper by Lee and Pan gives a summary to feature selection[13] broadly classifying them into global, statistical and topological features. In this paper, we extract global features as well as statistical features and verify the results using statistical distance distribution. We focus on a naive approach to offline signature verification. The feature extraction process and classification process used here are different from the ones discussed previously. They involve GSC features, which take into account the global as well as the local information of a signature. The verification and identification problem are solved using distance statistics such as the Bayes classifier and the k-nearest neighbor (k–nn) classifier techniques. The uniqueness of this system is described in the following sections followed by the experiments conducted and the results obtained.

2. Steps

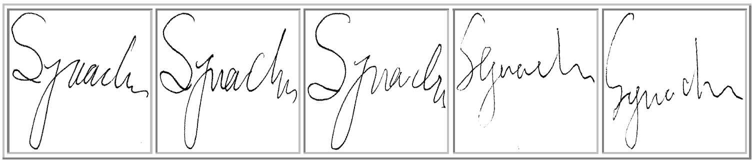
The steps performed to implement the signature verification and identification system are listed below.

Fig. 1. A sample signature of each writer from database A.

2.1. *Data acquisition*

We performed experiments on two different databases, namely databases A and B. Database A was built by us at CEDAR. This consisted of 55 writers with 24 genuine signature samples per writer. To obtain forgeries, we asked some other arbitrary people to skillfully forge the signatures of the writers in our database. In this fashion, we collected 24 forgery samples per writer from about 20 skillful forgers. The input signatures were scanned at 300 dpi in 8-bit gray scale (that is, 256 shades of gray) and stored in Portable Network Graphics (PNG) format. Writers were asked to sign in a predefined space of 2×2 inches. Figure 1 shows one sample for each of the writers from the database A. Figures 2(a) and 2(b) show 5 genuine and 5 forgery samples of a writer (writer 34) respectively.

(a)



(b)

Fig. 2. (a) Ten genuine samples of writer 34. (b) Ten forgery samples of writer 34.

We also tested our system on another database, which we termed as database B. This was a publicly available database from http://www.vision.caltech.edu/mariomu/research/data. This database comprised of two sets, sets 1 and 2, with a total of 106 writers and 3,960 samples of signatures.[12,16]

To obtain forgeries, some writers were asked to forge the signatures of other writers. There was no overlap between the writers of set 1 and those of set 2. The data for each writer was collected in three sessions that took place on different days so as to provide some variability in the signatures of a writer and at the same time to overcome the distortion due to boredom in such a repetitive task.

The database built at CEDAR (database A), was a pure offline database. The signatures were scanned and were lacking any dynamic information. On the other hand, database B was a camera captured database that had been collected for visual identification by signature tracking.[16] So the results presented here for the database A, more closely represent the truly offline signature verification, identification case. We can summarize the contents of our databases as shown in Table 1.

Table 1. Description of databases A and B.

		Database B	
	Database A	Set 1	Set 2
Number of writers	55	56	50
Number of genuine samples per writer	24	25	30
Number of forgeries per writer	24	10	10

2.2. *Preprocessing*

Database B required more preprocessing than database A. Since the signatures in database B were camera-captured, we had to first convert the pen co-ordinates into the xy co-ordinate space. This step was followed by interpolation to get a continuous curve of the signature. The next step in this preprocessing stage was to make the signature rotation invariant. The technique described in the work of Munich and Perona[16] was used here.

2.2.1. *Rotation normalization*

To perform rotation normalization, the signature curve was rotated until the axis of least inertia coincided with the horizontal axis. To achieve this the following procedure was followed.

We represented the given signature curve, C as shown in Eq. (1).

$$C = \left\{ X(i) = \begin{bmatrix} u(i) \\ v(i) \end{bmatrix},\ i = 1, \ldots, N \right\}. \tag{1}$$

where

$C =$ Signature curve,
$N =$ Number of pixels in the signature,
$X(i) =$ Vector consisting of x- and y-coordinates of the ith pixel in the signature,
$u(i) =$ x-coordinate of the ith pixel in the signature curve,
$v(i) =$ y-coordinate of the ith pixel in the signature curve.

Next, we found the coordinates, $\overline{u}$ and $\overline{v}$ of the center of mass of the signature.

$$\begin{aligned} \overline{u} &= \frac{1}{N} \sum_{i=1}^{N} u(i)\,, \\ \overline{v} &= \frac{1}{N} \sum_{i=1}^{N} v(i)\,. \end{aligned} \tag{2}$$

We then calculated the second order moments $\overline{u^2}$ and $\overline{v^2}$ of the signature by Eq. (3).

$$\begin{aligned} \overline{u^2} &= \frac{1}{N} \sum_{i=1}^{N} (u(i) - \overline{u})^2\,, \\ \overline{v^2} &= \frac{1}{N} \sum_{i=1}^{N} (v(i) - \overline{v})^2\,. \end{aligned} \tag{3}$$

The orientation of the axis of least inertia was then given by the orientation of the least eigen vector of the matrix in Eq. (4).

$$I = \begin{pmatrix} \overline{u^2} & \overline{uv} \\ \overline{uv} & \overline{v^2} \end{pmatrix}. \tag{4}$$

Once this angle was obtained, all the points in the signature curve under consideration were rotated with this angle. Figure 3 shows the effect of rotation normalization on a signature sample.

(a) (b)

Fig. 3. Rotation normalization: (a) original signature, and (b) signature after rotation normalization.

2.3. *Feature extraction*

Features for offline signature verification using scanned images can be divided into three types[7,13]:

(i) *Global features* that are extracted from every pixel that lies within a rectangle circumscribing the signature. These features do not reflect any local, geometrical, or topological properties of the signature, but include transformations,[6,18] series expansions,[15] image gradient analysis[23] etc. Although global features are easily extractable and insensitive to noise, they are dependent upon the position alignment and highly sensitive to distortion and style variations.

(ii) *Statistical features* that are derived from the distribution of pixels of a signature, e.g. statistics of high gray-level pixels to identify pseudo-dynamic characteristics of signatures. This technique includes the extraction of high pressure factors with respect to vertically segmented zones (for example, upper, middle and lower zones)[2] and the ratio of signature width to short- or long-stroke height.[17] The statistical features take some topological and dynamic information into account and consequently can tolerate minor distortions and style variations.

(iii) *Geometrical and topological features* that describe the characteristic geometry and topology of a signature and thereby preserve the signatures global and local properties, e.g. local correspondence of stroke segments to trace signature.[10] Geometrical and topological features have a high tolerance to distortion and style variations, and they can also tolerate a certain degree of translation and rotation variations.

In our system, we are using a combination of three types of features, known as the Gradient, Structural and Concavity or GSC features, which measure the image characteristics at local, intermediate and large scales and hence approximate a heterogeneous multiresolution paradigm to feature extraction. The gradient features detect local features of the image and provide a great deal of information about stroke shape on a small scale. The structural features extend the gradient features to longer distances and give useful information about stroke trajectories. The concavity features are used to detect stroke relationships at long distances which can span across the image.

The GSC feature vector length for this implementation is 512 bits. This GSC vector is compact because it is binary. The GSC feature extraction algorithm works with binarized images. First, size normalization of the image is accomplished by imposing a 4×4 grid on it by partitioning the image horizontally and vertically into four equal pixel mass partitions. Then the GSC features are computed. The gradient features are the ones at the finest scale. A gradient is the angle perpendicular to the local direction of the contour boundary and is computed at every pixel. The gradient features are computed by convolving two 3×3 Sobel operators on the binary image. The 3×3 Sobel operator was chosen to approximate the gradient of a two-dimensional image, because of its computational simplicity and also early tests indicated that it performed better than other approximations. After computing the gradient, it is quantized to 12 different ranges of angles. The histogram of occurrences of angles (ranges) for each of the 16 regions in the 4×4 grid is computed and those that cross a threshold are turned "on". Here the 12-bit feature vector for each region corresponds to 12 bins of direction, thus giving $12 \times 4 \times 4 = 192$ bits of the total feature vector. The structural features work at the intermediate scale. They capture certain patterns embedded in the gradient map. These patterns are "mini-strokes" of the image. A set of 12 rules that operate on the eight nearest neighbors of the pixel is applied to each pixel. Each rule examines a particular pattern of neighboring pixels for allowed gradient changes. If any pixel falling in the 4×4 region satisfies the rule for a mini-stroke, the feature is turned "on". Thus there is a 12-bit feature vector for each region, signifying the 12 rules. These features contribute $12 \times 4 \times 4 = 192$ bits to the total feature vector. The concavity features can be broken down into three subclasses of features: 16 pixel density features, 32 large stroke features and 80 concavity features. The pixel density feature determines whether the percentage of "on" pixels in the 4×4 region exceed a threshold. The large stroke features determine whether the 4×4 region contains a horizontal or vertical run of "on" pixels greater in length than a threshold. The concavity features determine if there is a concavity pointing in the region. The total contribution of these features is $4 \times 4 \times 8 = 128$ bits. Figure 4 shows the gradient, structural and concavity maps for the numeral 5. A detailed description of the GSC feature vector can be found in the work of Favata and Srikantan.[9]

As cited earlier,[28] characters usually inhabit in words and segregation of a word into allographs is more subjective than objective. Especially for cursive handwriting, use of handwritten words for studying handwriting individuality is a natural choice. Signature verification is similar to the individuality analysis of words. The degree to which handwriting is individual has been explored in the context of handwritten words.[28] Here in our approach, we termed each signature to be a word and the word feature extraction algorithm was used. This algorithm is an extension of the GSC algorithm[9] for handwritten characters mentioned above. The algorithm works as follows.

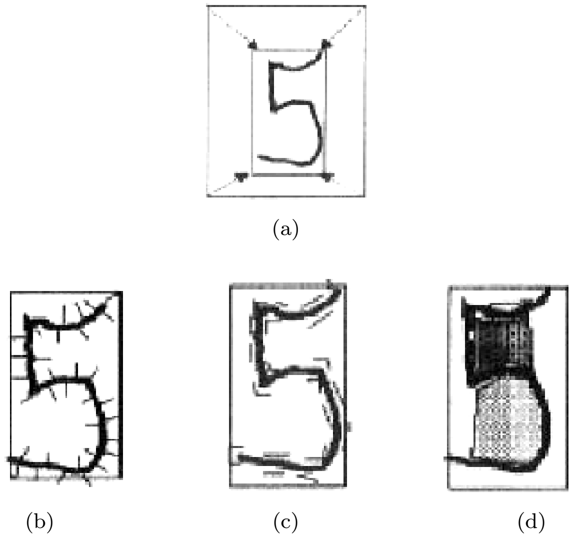


Fig. 4. GSC Features for the numeral 5: (a) Numeral 5 bounded in a bounding box, (b) gradient map, (c) structural map, and (d) concavity map.

To begin with, the signature image, S is divided into 4×8 subregions as follows.

First the image, S, is divided into four subregions along the vertical direction, such that each subregion contains the same number of black pixels; then S is divided into eight subregions along the horizontal direction in a similar way, giving 4×8 subregions in all. Then for each subregion, we use the GSC word algorithm to extract the 12-bit gradient, 12-bit structural features and 8 concavity features, thus giving a 1,024-bit feature vector. Figures 5(a) and 5(b) show a sample signature and the corresponding 1,024-bit GSC feature vector.

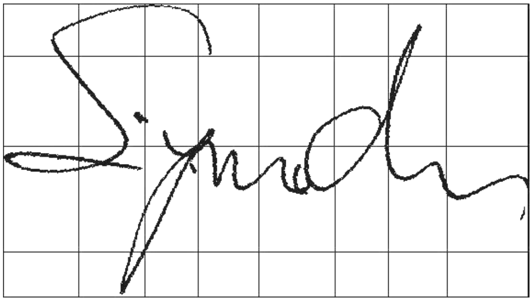
2.4. *Comparison process*

2.4.1. *Classifier design*

A methodology for handwriting verification and identification has been recently described[24] and a complete system for handwriting examination known as CEDAR-FOX system has been recently developed.[29] On the same lines, the present signature identification and verification system can be used in two modes of operation:

(i) *Verification*, where the goal is to provide a level of confidence as to whether a questioned document and a known document are from the same writer.
(ii) *Identification*, where the goal is to identify the writer of a questioned document given a repository of writing exemplars of several known writers.

Central to both identification and verification is the need for associating a quantitative measure of similarity between two samples. Such a quantitative measure brings in an assurance of repeatability and hence a degree of objectivity. Several



(a)

1011110011111011110011111111110011111111111011110111111111 00
000000000000111111111101011000111000001111001111000001000011
111001111001111011111001111111111101111111111001111101111101
000000000000111111011111111111111111011111111100111011101101
111100111111111111111111001001011011001111001111000000000100
111000001111111101111111111000111101111000111000000000000001
001101001100100111100111000011001110000011000100000011000000
000000000000000000000000000000000000001100011000000000000000
000011000000000011001110011111111100011011111100000011100100
000011111100100011111110100111000110000000000000000000000000
011000110000000000000000101001011000000001000000000000000000
100111001110000000000000001000011110001110011010000000000000
000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000
0000

(b)

Fig. 5. Feature extraction: (a) Signature sample, and (b) the corresponding 1024-dimensional GSC feature vector.

methods for comparing strings of binary feature vectors representing handwritten characters have been recently evaluated.[29] This has led to the choice of correlation measure as being the best of binary string matching measures.

In the identification model, a binary feature vector is associated with each signature sample and then the proximity of a sample to all other samples is calculated using the similarity measure mentioned below, whereas, in the verification model, a real-valued distance vector (where each component represents the distance between two signature samples) is used to describe the difference between a pair of signature samples. The weighted Euclidean distance measure (weighted by the standard deviations of features) is used to measure the distance between any two distance vectors.

2.4.2. *Classification technique*

For the verification model, we formed the same writer and different writer probability distributions to classify any unknown sample. We used the Bayes classifier to form these distributions with mean and variance measures to classify a new instance. This used the similarity metric described below.

A binary vector Z with N dimensions is defined as:

$$Z = (z_1,\, z_2\,, \ldots, z_N) \tag{5}$$

where

$$z_i \in \{0, 1\},\ \forall\, i \in \{1, 2, \ldots, N\}\,. \tag{6}$$

Let Ω be the set of all N-dimensional binary vectors, then the unit binary vector $I \in \Omega$, is a binary vector with every element equal to 1. The complement of a binary vector $Z \in \Omega$ is given by,

$$\overline{Z} = I - Z\,. \tag{7}$$

To measure the similarity between two binary images, we use the Correlation measure.[26] Let $S_{ij}(i, j \in \{0, 1\})$ be the number of occurrences of matches with i in the first pattern and j in the second pattern at the corresponding positions. Given two binary feature vectors $X \in \Omega$ and $Y \in \Omega$, each similarity measure $S(X, Y)$ above uses all or some of the four possible values, i.e. $S_{00}, S_{01}, S_{10}, S_{11}$. We define a similarity measure $S(X, Y)$ corresponding to the Correlation measure in Eq. (8).

$$S(X, Y) = \frac{s_{11}s_{00} - s_{10}s_{01}}{((s_{10} + s_{11})(s_{01} + s_{00})(s_{11} + s_{01})(s_{00} + s_{10}))^{1/2}}\,. \tag{8}$$

where

$S(X, Y) =$ correlation similarity measure.

$s_{00} =$ the first binary vector has a 0 and the second vector too has a 0 in the corresponding positions.

$s_{11} =$ the first binary vector has a 1 and the second vector too has a 1 in the corresponding positions.

$s_{01} =$ the first binary vector has a 0 while the second vector has a 1 in the corresponding positions.

$s_{10} =$ the first binary vector has a 1 while the second vector has a 0 in the corresponding positions.

This similarity measure is not normalized. The range for this similarity measure is $[-1, 1]$. The assumption underlying this range is, $S(I, I) = S(\overline{I}, \overline{I}) = 1$ and $S(I, \overline{I}) = S(\overline{I}, I) = -1$ For the identification model we use the weighted k-nearest neighbor classification. Weighted k-nearest neighbor algorithm is a refinement to the k-nearest neighbor algorithm. In this, the contribution of each of the k-neighbors is weighted according to their distance to the query point, giving greater weight to closer neighbors. In our case, we weight the vote of each neighbor according to the inverse square of its distance from the query signatures feature vector. This distance is computed using the similarity measure mentioned above.

3. Experimental Settings

3.1. *Experimental setup for database A*

Tables 2 and 3 illustrate the experimental setup for database A for signature verification and identification respectively.

Table 2. Experimental setup for verification for database A.

	Number of Signature Pairs	
	Same Writers	Different Writers
Training Set	6600	14080
Testing Set	1540	3520

Table 3. Experimental setup for identification for database A.

	Number of Signatures
Training Set	880
Testing Set	440

3.1.1. *Verification*

To perform verification, we first formed the same writer and different writer probability distributions. The same writer and different writer probability density functions (pdfs) are shown in Fig. 6. This involved forming the same writer and different writer sets. The testing and training sets were built so as to consist of within-writer and between-writer distance vectors.

The training set consisted of two-third signatures of each writer, while the testing set consisted of the remaining one-third signatures of each writer. Thus the training and testing sets were disjoint. Since for each writer we had 24 genuine and 24 forged signature samples, we used 16 samples for training and the remaining 8 for testing for both same writers and different writers. To form the training set, $55.{}^{16}\mathrm{C}_2 = 6{,}600$ same writer signature pairs and $55.{}^{16}\mathrm{C}_{1.}{}^{16}\mathrm{C}_1 = 14{,}080$ different writer signature pairs were used. Here ${}^{n}\mathrm{C}_r = n!/(r!.(n-r)!)$. For testing, we used $55.{}^{8}\mathrm{C}_2 = 1{,}540$ same writer signature pairs and $55.{}^{8}\mathrm{C}_{1.}{}^{8}\mathrm{C}_1 = 3{,}520$ different writer signature pairs.

3.1.2. *Identification*

Two-thirds (16 signatures) of the available signatures for each writer were used for training, while the remaining one-third (8 signatures) was used for testing. Forgeries were not considered for the identification process. The training set comprised of 880 genuine signatures (55 writers with 16 samples per writer), whereas the testing set consisted of 440 genuine signatures (55 writers with 8 samples per writer).

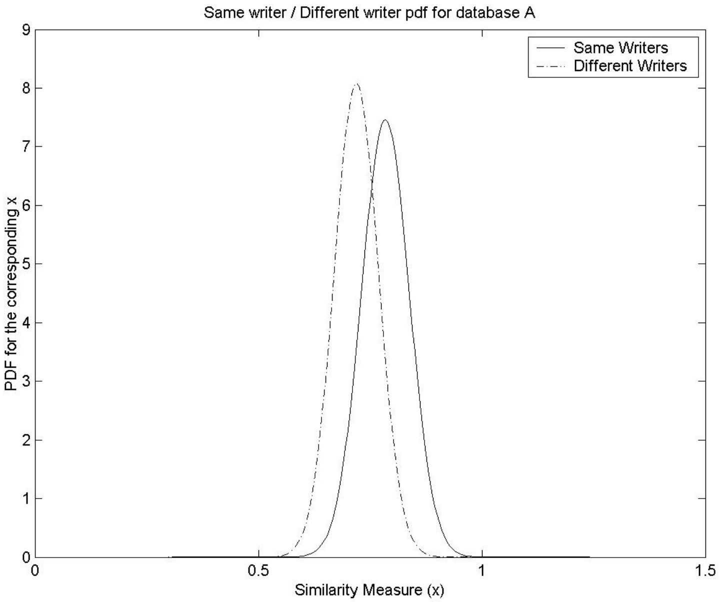


Fig. 6. PDF for database A.

3.2. *Experimental setup for database B*

Tables 4 and 5 illustrate the experimental setup for database B for verification and identification respectively.

Table 4. Experimental setup for verification for database B.

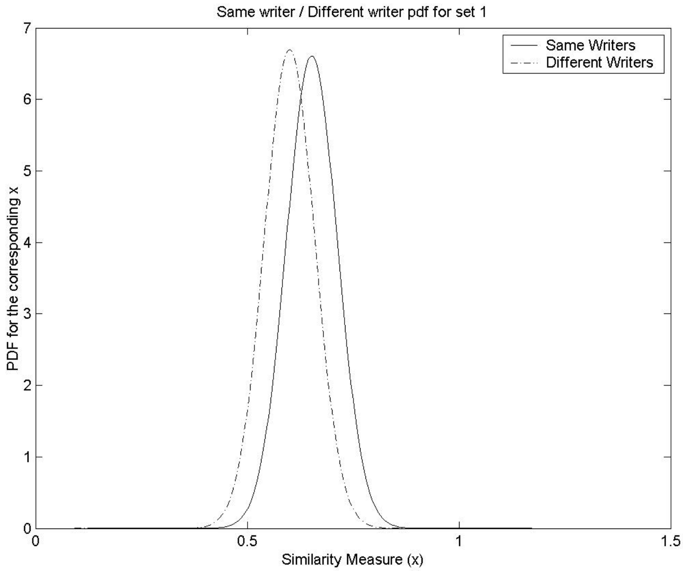
		Number of Signature Pairs Same Writers	Different Writers
Set 1	Training Set 1	5880	5040
	Testing Set 1	2520	2240
Set 2	Training Set 2	9500	6000
	Testing Set 2	2250	2000

Table 5. Experimental setup for identification for database B.

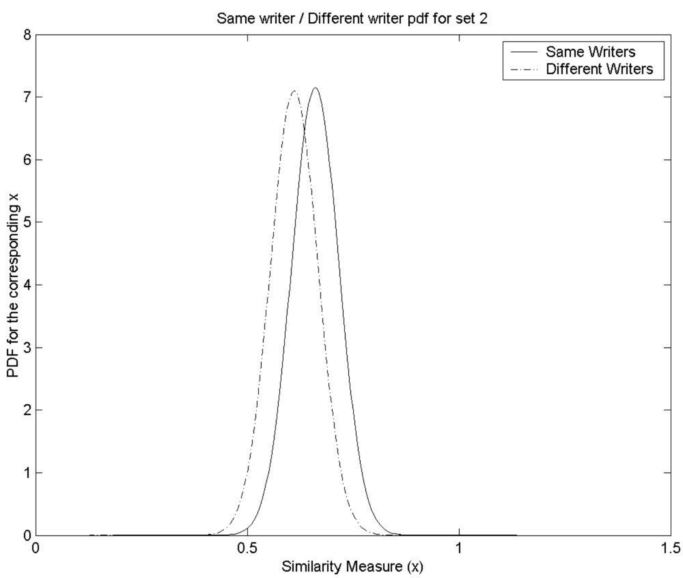
	Set 1	Set 2
Training Set	839	991
Testing Set	560	500

3.2.1. *Verification*

Similar to the setup for database A, the same writer and different writer probability distributions were formed here for each of the two sets in database B. The pdfs are plotted in Figs. 7(a) and 7(b).



(a)



(b)

Fig. 7. (a) PDF for database B — set 1. (b) PDF for database B — set 2.

For set 1, $56.{}^{15}\mathrm{C}_2 = 5{,}880$ same writer signature pairs and $56.{}^{15}\mathrm{C}_1.{}^{6}\mathrm{C}_1 = 5{,}040$ different writer signature pairs were used for training. To test for same writers and different writers, $56.{}^{10}\mathrm{C}_2 = 2{,}520$ same writer signature pairs and $56.{}^{10}\mathrm{C}_1.{}^{4}\mathrm{C}_1 = 2{,}240$ different writer signature pairs were used respectively.

For set 2, $50.{}^{20}\mathrm{C}_2 = 9{,}500$ same writer signature pairs and $50.{}^{20}\mathrm{C}_{1.}{}^{6}\mathrm{C}_1 = 6{,}000$ different writer signature pairs were used for training. To test for same writers and different writers, $50.{}^{10}\mathrm{C}_2 = 2{,}250$ same writer signature pairs and $50.{}^{10}\mathrm{C}_{1.}{}^{4}\mathrm{C}_1 = 2{,}000$ different writer signature pairs were used respectively.

3.2.2. *Identification*

Identification tests were carried out individually for sets 1 and 2. The available genuine signature samples in each set were divided into training and testing sets. Some samples were missing and so were not considered in the experiments.

Training set 1 consisted of 839 genuine signature samples (56 writers with about 15 genuine samples each), whereas testing set 1 consisted of 560 genuine signature samples (56 writers with 10 genuine samples each).

Training set 2 consisted of 991 genuine signature samples (50 writers with about 20 genuine samples each), whereas testing set 2 consisted of 500 genuine signature samples (50 writers with 10 genuine samples each).

4. Results

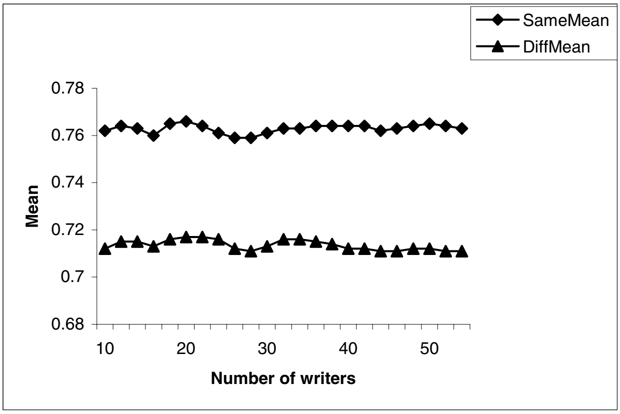
4.1. *Results for database A*

4.1.1. *Verification results*

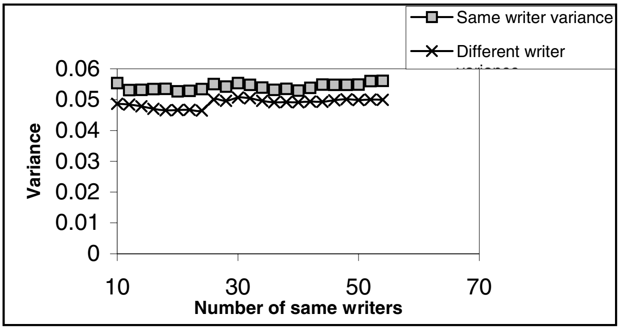
Here the probabilities of the signature distance being acceptable as genuine or a forgery are determined. The distance between the two binary feature vectors is computed using the correlation distance measure described above. The mean and variance of distance as a function of the writers is shown in Figs. 8(a) and 8(b). Here the x-axis has the number of writers n, varied from 10 to 55. For each n, there are two values in each plot corresponding to the mean and variance of $n \times {}^{16}\mathrm{C}_2$ pairs of same writer (or genuine-genuine pair) distances and $n \times {}^{16}\mathrm{C}_1 \times {}^{16}\mathrm{C}_1$ pairs of different writers (or genuine-forgery pair) distances. The average value of the mean for same writers and different writers is $\mu_{\text{same}} = 0.76$ and $\mu_{\text{diff}} = 0.72$ respectively, with corresponding variances 0.055 and 0.05. Since the variances are roughly the same, the Bayes classifier reduces to classifying as a forgery if the similarity is less than 0.74 and as genuine otherwise.

The threshold depending fraction of the falsely accepted patterns divided by the number of all impostor patterns is the False Acceptance Rate (FAR) and the fraction of the number of rejected client patterns divided by the total number of client patterns is called the False Rejection Rate (FRR). To plot the FAR versus FRR graph we need different values for the decision threshold. To obtain these values, we vary the ratio of the same writer probability to the different writer probability. Here our distance implies the similarity between the two samples under consideration.

The verification results for different values of the threshold were noted and the graphs were plotted. As seen from the graph in Fig. 9, as the threshold increases, the



(a)



(b)

Fig. 8. Same and different writer distance parameters as functions of number of writers considered: (a) mean values and (b) variances.

Table 6. Signature verification results for database A.

Threshold → Errors (%)↓	0.1	0.3	0.5	1	4	8	20
FAR	91.61	61.15	41.02	20.62	2.79	0.43	0
FRR	1.33	5.46	10.08	23.18	62.61	80.46	95.52

FAR decreases, while the FRR increases, Table 6 shows the corresponding results. The FAR for this database is 23.18% while the FRR is 20.62%. Therefore, the average error rate or Equal Error Rate (EER), that is, the point at which FAR and FRR are equal is 21.90%.

4.1.2. *Identification results*

Figure 10 shows the identification performance of the system. The system was tested for different values of k and the graph was plotted.

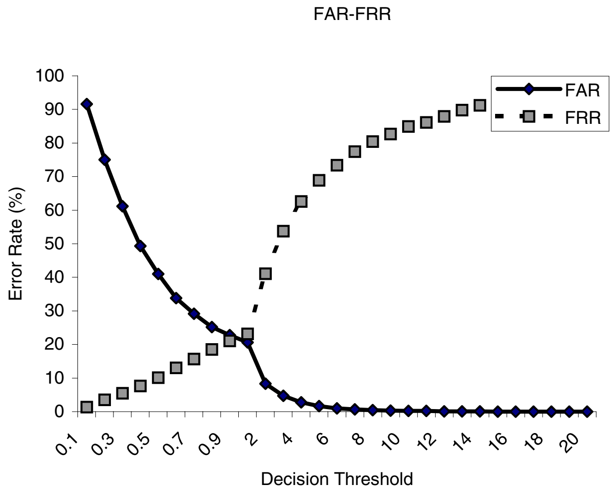


Fig. 9. FAR versus FRR for database A.

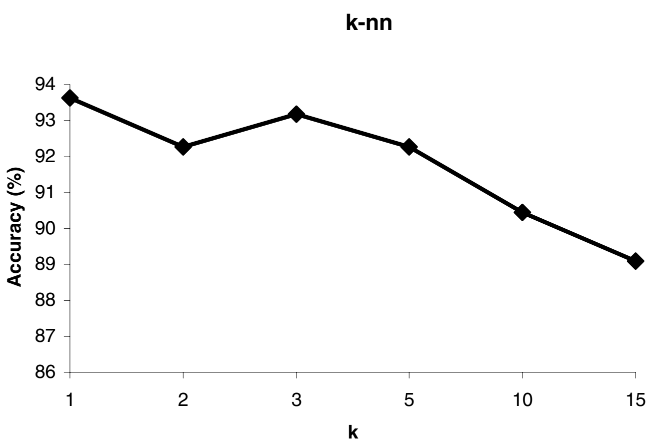


Fig. 10. Identification results for database A.

Table 7. Signature identification results for database A.

K	1	2	3	5	10	15
Accuracy (%)	93.63	92.27	93.18	92.27	90.45	89.09

As seen from the results summarized in Table 7, the optimal or the best performance of the system, giving an identification accuracy of 93.18%, is noted for $k = 3$ (neglecting the 1-nearest neighbor case).

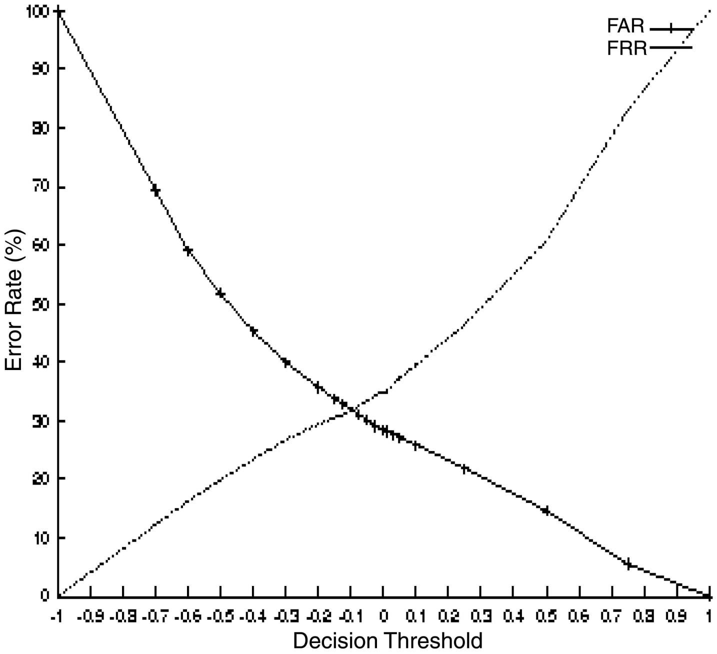


Fig. 11. FAR versus FRR for data set 1.

Table 8. Signature verification results for sets 1 and 2.

Threshold → Errors (%)↓		0.01	0.25	0.5	0.75	1
Set 1	FAR	28.17	21.9	14.64	5.55	0
	FRR	35.04	46.52	60.76	83.08	100
Set 2	FAR	30.71	23.91	16.71	7.77	0
	FRR	34.3	43.15	55.75	76.25	100

4.2. *Results for database B*

4.2.1. *Verification results*

To plot the FAR versus FRR graph we need different values for the decision threshold. To obtain these values, we vary the allowable difference between the same writer probability and the different writer probability. As expected, FAR decreases with increasing threshold, while FRR increases. Figure 11 shows the FAR versus FRR graph for set 1. Table 8 summarizes these results. For set 1, the FAR is 34.91% while the FRR is 28.33%. The average error rate of the system is 31.62%. For set 2, the FAR is 33.8% while the FRR is 30.93%. The average error rate of the system is 32.37%.

Table 9. Signature identification results for sets 1 and 2.

K → Accuracy (%)↓	1	2	3	4	15
Set 1	93.33	86.61	92.14	91.43	86.25
Set 2	91.6	88.4	92.4	92	91.8

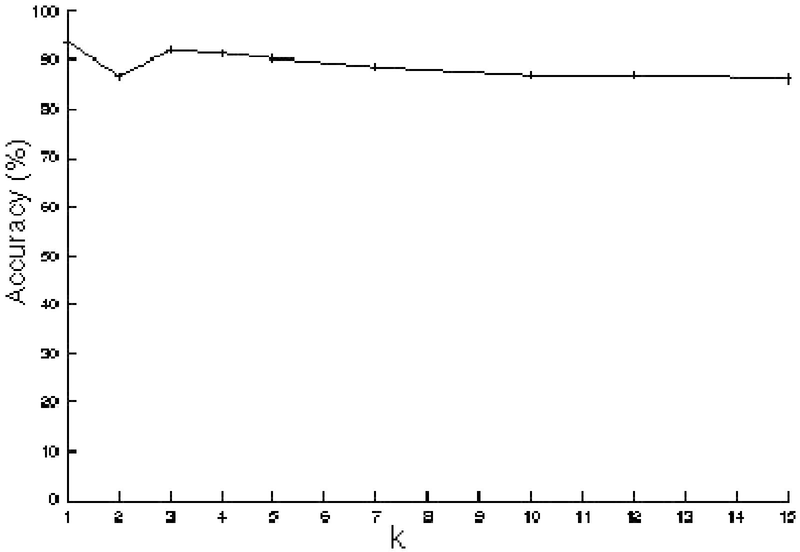


Fig. 12. Identification results for data set 1.

4.2.2. *Identification results*

As seen from the results summarized in Table 9, the optimal performance of the system for data set 1 is 93.33% noted for $k = 3$ and for data set 2, it is 91.6% at $k = 4$ (neglecting the 1-nearest neighbor case). Figure 12 displays these results graphically for set 1.

The overall performance of the signature verification and identification system is summarized in Table 10.

4.3. *Comparison with other offline methods*

It is difficult to compare the performance of different signature verification systems because different systems use different signature databases.[8] Hence, here we just list the performance achieved by some of the other systems and that achieved by our system. The novel approach to offline tracing and representation of signatures,[13] could achieve about 97% recognition rate for the test patterns derived from the training patterns. The average error rates for verification in the work of Ammar[1]

Table 10. Overall system performance.

Database	System	Evaluation Values in %		
A	Verification	FAR	23.18	
		FRR	20.62	
		EER	21.9	
		Overall Accuracy	78.1	
	Identification	Overall Accuracy	93.18	
			Set 1	Set 2
B	Verification	FAR	34.91	33.8
		FRR	28.33	30.93
		EER	31.62	32.37
		Overall Accuracy	68.38	67.63
	Identification	Overall Accuracy	93.33	91.6

and Sabourin[22] were 22.8% and 17.8% respectively. For our approach, in the case of verification the average error rate for verification on database A is 21.9% and that on database B is 31.62% for set 1 and 32.67% for set 2. The average error rate for identification for database A is 6.82% and that for set 1 is 6.67% while that for set 2 is 8.4%.

5. Summary and Conclusions

We used a combination of Gradient, Structural and Concavity (GSC) features here to extract the significant features of a signature at the local, intermediate and large scales for object recognition. These features captured the global, statistical and geometrical features of the signature. The experiments listed here demonstrate the usefulness of statistical measures like the Bayes and k-nearest neighbor classifier in the offline signature verification and identification domain respectively. The performance of this system is comparable to other offline signature verification and identification systems as indicated by the results. We achieve accuracy as high as 78.1% for verification and 93.18% for identification on a pure offline database using GSC word features. This approach could be combined with Dynamic Plane Warping algorithm to improve the performance of the system. Dynamic plane warping (DPW)[14] algorithm has been applied in the field of optical character recognition (OCR), which solves plane alignment problem. With DPW, dynamic features can be extracted from offline signatures and combined with the framework we presented in this paper.

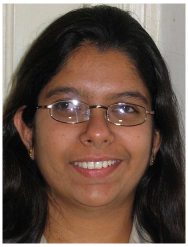
Acknowledgments

This work was supported in part by the US Department of Justice, National Institute of Justice grant 2002-LT-BX-K007.

References

1. M. Ammar, Progress in verification of skillfully simulated handwritten signatures, *Int. J. Pattern Recognition and Artificial Intelligence* **5**(1–2) (1991) 337–351.
2. M. Ammar, Y. Yoshido and T. Fukumura, A new effective approach for offline verification of signatures by using pressure features, in *Proc. 8th Int. Conf. Pattern Recognition*, 1986, pp. 566–569.
3. M. Ammar, Y. Yoshido and T. Fukumura, Structural description and classification of signature images, *Patt. Recogn.* **23**(7) (1990) 697–710.
4. R. Bajaj and S. Chaudhury, Signature verification using multiple neural classifiers, *Patt. Recogn.* **30**(1) (1997) 1–7.
5. B. Boser, I. Guyon and V. Vapnik, A training algorithm for optimal margin classifiers, in *Proc. Fifth Ann. Workshop on Computational Learning Theory*, 1992, pp. 144–152.
6. F. Cheng, W. Hsu and M. Chen, Recognition of handwritten chinese characters by modified Hough transform techniques, *IEEE Trans. Patt. Anal. Mach. Intell.* **11** (1989) 429–439.
7. B. Fang, C. H. Leung, Y. Y. Tang, K. W. Tse, P. C. K. Kwok and Y. K. Wong, Offline signature verification by the tracking of feature and stroke positions, *Patt. Recogn.* **36**(1) (2003) 91–101.
8. B. Fang, C. H. Leung, Y. Y. Tang, P. C. K. Kwok, K. W. Tse and Y. K. Wong, Offline signature verification with generated training samples, *IEE Proc. Vis. Imag. Sign. Process.* **149**(2) (2002) 85–90.
9. J. T. Favata and G. Srikantan, A multiple feature/resolution approach to hand printed word and character recognition, *Int. J. Imag. Syst. Technol.* **7** (1996) 304–311.
10. J. K. Guo, D. Doermann and A. Rosenfeld, Local correspondence for detecting random forgeries, in *Proc. Int. Conf. Document Analysis and Recognition*, 1997, pp. 319–323.
11. T. Joachims, Text categorization with support vector machines: Learning with many relevant features, in *Proc. Eur. Conf. Machine Learning*, 1998, pp. 137–142.
12. M. K. Kalera, B. Zhang and S. N. Srihari, Offline signature verification and identification using distance statistics, in *Proc. Int. Graphonomics Soc. Conf.*, 2003, Scottsdale, AZ.
13. S. Lee and J. C. Pan, Offline tracing and representation of signatures, *IEEE Trans. Syst. Man Cybern.* **22** (1992) 755–771.
14. E. Leven and R. Pieraccini, Dynamic planar warping for optical character recognition, in *Proc. ICASSP*, 1992, pp. III149–152.
15. C. C. Lin and R. Chellappa, Classification of partial 2-D shapes using Fourier descriptors, *IEEE Trans. Patt. Anal. Mach. Intell.* **9** (1997) 696–690.
16. M. E. Munich and P. Perona, Visual identification by signature tracking, *IEEE Trans. Patt. Anal. Mach. Intell.* **25**(2) (2003).
17. R. N. Nagel and A. Rosenfeld, Computer detection of freehand forgeries, *IEEE Trans. Comput.* **C-26** (1997) 895–905.
18. W. F. Nemcek and W. C. Lin, Experimental investigation of automatic signature verification, *IEEE Trans. Syst. Man Cybern.* **SMC-4** (1974) 121–126.
19. R. Plamondon and S. N. Srihari, Online and offline handwriting recognition: a comprehensive survey, *IEEE Trans. Patt. Anal. Mach. Intell.* **22**(1) (2000) 63–84.
20. R. Plamondon and G. Lorette, Automatic signature verification and writer identification — the state of art, *Patt. Recogn.* **22**(2) (1989) 710–717.
21. R. Plamondon and M. Parizeau, A comparative analysis of regional correlation, dynamic time warping and skeletal tree matching for signature verification, *IEEE Trans. Patt. Anal. Mach. Intell.* **12**(7) (1990) 710–717.
22. R. Sabourin and G. Genest, An extended-shadow-code-based approach for offline

signature verification: Part I. Evaluation of the bar mask definition, in *Proc. 12th Int. Conf. Pattern Recognition*, October 1994, Jerusalem, Israel, Vol. 2, pp. 450–453.
23. R. Sabourin and R. Plamondon, Preprocessing of handwritten signatures from image gradient analysis, in *Proc. 8th Int. Conf. Pattern Recognition*, 1986, pp. 576–579.
24. S. N. Srihari, S.-H. Cha, H. Arora and S. Lee, Individuality of handwriting, *J. Foren. Sci.* **47**(4) (2002) 856–872.
25. G. Srikantan, S. Lam and S. Srihari, Gradient-based contour encoding for character recognition, *Patt. Recogn.* **7** (1996) 1147–1160.
26. J. D. Tubbs, A note on binary template matching, *Patt. Recogn.* **22**(4) (1989) 359–365.
27. Q. Wu, S. Lee and Z. Jou, On-line signature verification based on split-and-merge matching mechanism, *Patt. Recogn. Lett.* **18** (1997) 665–673.
28. B. Zhang and S. N. Srihari, Analysis of handwriting individuality using word features, in *Proc. Seventh Int. Conf. Document Analysis and Recognition*, Edinburgh, Scotland (IEEE Computer Society Press, Los Alamitos, CA, 2003).
29. B. Zhang and S. N. Srihari, Binary vector dissimilarities for handwriting identification, in *Proc. SPIE, Document Recognition and Retrieval X*, January 20–24, 2003, Santa Clara, CA, USA.



Meenakshi Kalera received her bachelor's in computer engineering from Thadomal Shahani Engineering College, Mumbai University, India, in 2001 and is currently working towards her master's in computer science at University at Buffalo, USA. She has been working as a Research Assistant at the Center of Excellence for Document Analysis and Recognition (CEDAR) since 2002.

Her current research interests include offline verification of signatures, handwriting individuality, data mining and various other biometric methods. She is a member of the International Graphonomics Society.

Aihua Xu is currently working as a software engineer in Hyland Software Inc., USA. He obtained his B.S. and M.S. in computer science from Fudan University in 1998 and 2001, respectively. He also received his M.S. in computer science from SUNY at Buffalo in 2004.

His scientific interests include pattern recognition and data mining.

Sargur Srihari is a SUNY Distinguished Professor at the University of Buffalo, the State University of New York (SUNY), in the Department of Computer Science and Engineering where he teaches courses in machine learning and pattern recognition.

He is the founding director of the Center of Excellence for Document Analysis and Recognition (CEDAR). The handwritten address interpretation system developed at CEDAR led to systems now being deployed by the postal services of the United States, Australia and the United Kingdom. More recently he has been involved in the development of information processing models for human individuality.

He is a Fellow of the Institution of Electrical and Electronics Engineers (IEEE) and a Fellow of the International Association of Pattern Recognition (IAPR). He has been General Chairman of three IEEE-sponsored international conferences in Handwriting Recognition and Document Analysis and Recognition. He is a member of the Board of Scientific Counselors of the National Library of Medicine, and a member of the Technical Advisory Boards of two pattern recognition and natural language processing software companies.

He received a B.E. in electrical communication engineering from the Indian Institute of Science, Bangalore, India, in 1970 and an M.S. and Ph.D. in computer and information science from the Ohio State University, Columbus in 1972 and 1976, respectively. He received a Distinguished Alumnus Award by the Ohio State University College of Engineering in 1999. He received the best paper award at the International Conference on Document Analysis and Recognition in Edinburgh, Scotland in 2003.