Multi-script vs single-script scenarios in automatic off-line signature verification

Abhijit Das[1*], Miguel A. Ferrer[2], Umapada Pal[3], Srikanta Pal[1], Moises Diaz[2], Michael Blumenstein[1,4]

[1] School of Information and Communication Technology, Griffith University, Queensland, Australia.
[2] Instituto Universitario para el Desarrollo Tecnológico en Comunicaciones, Universidad de Las Palmas de Gran Canaria, Spain.
[3] Computer Vision and Pattern Recognition Unit, Indian Statistical Institute, Kolkata, India
[4]School of Software, University of Technology Sydney, Australia
[*]E-mail: abhijit.das@griffithuni.edu.au

Abstract. This paper introduces a novel method to build up a multi-script off-line signature database aggregating many single-script off-line databases,, along with a statistical performance analysis method for a fair comparison between single and multi-script scenarios. This analysis method is based on merging the single-script databases without increasing the number of users (signers) and selecting the users for merging, based on the probability density functions of the users' Equal Error Rates (EERs). As similar results are achieved when merging single and multi-script databases, it is concluded that multi-script signature verification is actually a generalization and interoperability problem. The research also concludes that a statistical performance analysis method that is Bhattacharyya Distance could be used for analysing multi-script vs single-script signature verification scenarios. These results have been obtained after experimenting with nine public databases with five different scripts.

1. Introduction

Most law enforcement agencies, governments of various countries, financial institutions, or forensic units use the signature as identity proof in their daily activities[1]. Usually, signature verification involves a manual procedure carried out by Forensic Handwriting Experts (FHEs), where individual characteristics of the signature are observed, such as inclination, slant, hooks, relationship between letters, and so on, using a common set of protocols and methods. This analysis is time-consuming and moreover it performance depends on many factors such as expertise of the observer, availability of data and sample quality. To mitigate these drawbacks, ASV (Automatic Signature Verification) has been proposed. Computer vision and pattern recognition frameworks have been employed to build such systems, in which the signatures are mathematically modelled, and further a quantitative similarity called the likelihood is calculated to identify the query signature. Such systems are nowadays becoming accepted for law enforcement agencies,

[1] http://legal-dictionary.thefreedictionary.com/signature

governments of various countries, financial institutions and courts. As a result, various commercial ASV systems have become available[2].

Signature verification schemes are usually focused on a single-script environment *(Here by script we mean the set of letters or characters i.e. different symbols, used for writing a particular language).* The multi-script signature scenario is also a very practical situation, which can be encountered in international security or forensic paradigms, where aggregation of a variety of script signatures coming from different geographical areas can be encountered. Performance of multi-script based ASV scenarios has hardly been studied [1] in the literature. The multi-script scenario is usually studied by merging several single-script databases to build a multi-script database and comparing their performance [2] are compared with the merged single-script databases. Furthermore, the literature implies that the performance Automatic Signature Verifier (ASV) reduces when applied to a multi-script dataset [2]. The reason behind poorer results reported in the multi-script ASV scenario in construct to the single-script database possibly could be due to the larger number of users. However, to the best of our knowledge, no studies have focused on the database merging procedure or proposing a statistical measure for a fair comparison of multi-script verses single-script scenarios.

Therefore, in order to investigate the aforementioned gap in the literature, the present work is conceived. This paper proposes a statistical measure, as well as a merging procedure for a fair comparison between single-script and multi-script databases. This study includes extracting features from the signatures by various well-known techniques, and subsequently perform an analysis of the different single-script and multi-script databases. Thereafter, a statistical measure is also employed for a fair comparison of the multi-script vs. single-script scenarios. Before explaining the proposed approach, the databases used, as well as the off-line ASV feature extraction techniques, are briefly described.

The organization of the rest of the paper is as follows. Section 2 highlights the literature of signature verification, and in Section 3 we explain single-script datasets that are merge to obtain multi-script databases. In Section 4, the proposed dataset merging procedure and various features extraction technique employed in the experiments and performance measures for comparison of multi-script verses single-script scenarios are proposed. The results are reported along with a discussion in Section 5. The overall conclusion and the future scopes of the work is explained in Section 6.

[2] http://www.biometricupdate.com/service-directory/signature-verification

2. Literature review

Automatic Signature Verification has been considered an active problem in the scientific research community since the 1980's. The majority of initiatives have produced several contributions to the field over the years [3-8]. However, some issues have not yet been fully identified due to their recent emergence [8]. One such active aspect is multi-script signature processing.

Western signatures are most commonly dealt with in regards ASV scenario in the research community. These kinds of signatures have two different parts: the text and the flourish. The text has a more regular kinematic per signer. Conversely, the variation in the flourish is unpredictable according to its lexical morphology [9]. In addition, the mean velocities in both parts are very different: the flourish average velocity is not nearly as high as the text velocity average. However, these features cannot be directly estimated from image-based signatures. Whereas, Indian, Chinese and Japanese signatures consist of mainly short and rapid strokes. This way, each pen-down is far smaller than the pen-downs of Western signatures.

A large number of approaches proposed in the literature on non-Latin signature have considered script-based text recognition and static signature recognition over the years without dependence on the number of scripts or their combination [8]. In [10], the authors evaluated three different signature scripts - Bengali, Devanagari and Roman – throughout automatic signature systems. They found that the majority of system errors were due to the misclassification of Bengali and Devanagari signatures. The same authors also used a modified gradient feature and an SVM classifier for identification and verification purposes in [11]. They concluded that the verification rates were more competitive for off-line Hindi signature scripts than for English (the achieved False Acceptance Rate (FAR) was more than twice for the latter). In a following work of [12], at first the identification of the script are considered and then verifier method is applied accordingly to the detected signature script (Hindi or English). In this work, the average error rate was significantly reduced to 4.81%, mainly due to the first stage of script identification. For Bengali and English off-line signature verification, in [13] the authors propose a combination of gradient features and chain code features as templates for signature verification. They were able to obtain similar high accuracies of 99.41%, 98.45%, and 97.75% using the respective feature extraction techniques. In the case of English and Chinese static signature scripts, in [14] the authors proposed a script identification approach on the basis of a foreground and background technique. Their contribution for multi-script verification relied on script identification before employing the verification process, which reached a accuracy 97.70% during the identification stage.

This combination opens up some relevant questions to the currently used technology. On the one hand, the accuracy of single-script systems can be progressed by the combination of scripts. However, the system achieves a competitive performance with Roman script or Bengali script-based signature when considered

separately, but similar accuracy is not obtained when these scripts are combined. This way, we wonder about the real influence of low accuracy in the multi-script ASV.

Therefore, although substantial research has previously been undertaken in the area of signature verification, particularly involving single-script signatures, multi-script ASV needs further attention. Moreover, a multi-lingual country like India has many different scripts that are used for writing as well as for signing purposes, based on different locations or regions. In India, a single official transaction sometimes needs signatures using more than one script. Thus, the consideration of signatures dealing with more than one or two scripts is important mainly for multi-lingual and multi-script countries. Moreover, the development of a general multi-script signature verification system is very complicated. This is where the present research concentrates on investigating the real cause of the lower performance of multi-script ASVs, and proposes a fair method to analyse their performance.

3. Databases

To look into the cause of the lower performance in the multi-script scenario, eight different databases have been used for the experimental study of this work. Some of them contain western signatures, to follow up the single-script merging dataset case, whereas the others include different scripts such as Devanagari, Bengali, Chinese and Arabic. The datasets are described as follows:

1. The GPDS100 contains the first 100 signatures of the GPDS960 signature database [15], which was recorded in Spain in Roman script. The signers used their own pen to sign on a piece of paper. For each signer it consists of 24 genuine signatures and 30 forgeries. The 24 genuine specimens of each signer were collected in a single day writing sessions. The forgeries were produced from the static image of the genuine signature. Each forger was allowed to practice the signature for as long as they required to produce the forgery. Each forger imitated 3 signatures of 5 signers each day. The genuine signatures shown to each forger are chosen randomly from the 24 genuine ones.

2. The MCYT100 contains the first 100 signatures of the MCYT online database from Spain [16] in Roman script. This was recorded on a WACOM tablet. Each user produces 25 genuine signatures, and 25 skilled forgeries are also captured for each user. These skilled forgeries are produced by the 5 subsequent users by observing the static images of the signature to imitate and to attempt to copy.

3. The SUSIG Visual [17] contains Roman script signatures of 94 users acquired in two sessions. They were recorded in Turkey on an LCT touch device. Each signer supplied 20 samples of his/her signature

in two different sessions, supplying 10 signatures at each session. There was approximately a one-week time period in between the two signing sessions. A total of 10 skilled forgeries (5 skilled and 5 highly-skilled) were collected for each person. Skilled forgers watched an animation of the signature to be falsified and practiced as long as they required. Once they felt sufficiently skilled, they proceeded to write the forged specimen.

4. The NISDCC [18] was used for a signature competition during ICDAR 2009. It was collected by the Netherlands Forensic Institute. This corpus comprises off-line Roman script signatures of 79 users. Only 19 users of this database include forgeries. Each forger copied the genuine signature as fluently as possible, focusing on mimicking the shape of the specimen.

5. The SVC2004 [19] online database was collected in Hong Kong. It contains Chinese and Roman script signatures of 80 users. This dataset was obtained from the SVC-2004 competition held in conjunction with the First International Conference on Biometric Authentication (ICBA 2004). Each set of signers contains 20 genuine signatures and 20 skilled forgeries from five other contributors. To collect the forgeries, each contributor saw the writing order in which the signature was written. Then, after practising, they decided when to reproduce it.

6. The off-line Bengali [20] signature database was recorded in India with 100 signers using paper as the medium for capturing the writing. From each individual, 24 genuine signatures were collected. A total number of 2400 genuine signatures from 100 individuals were collected. For each contributor, all genuine specimens were collected in a single day's writing session. In addition, only skilled forged signatures were collected for this proposed work. In order to produce the forgeries, the imitators were allowed to practice their forgeries as long as they wished with static images of genuine specimens. A total number of 3000 forged signatures were collected from the writers.

7. The off-line Hindi dataset [20] was recorded in the same conditions as with the Bengali one. From each individual, 24 genuine signatures were collected. A total number of 2400 genuine signatures from 100 individuals were collected. A total number of 3000 (10 per signer) forged signatures were also collected from the writers.

8. Finally, the offline Arabic database [21] was recorded in Egypt and it contains 22 signers. A set of signature data consisting of 220 true samples and 110 forged samples was used. Every signer was asked to sign 10 times using common types of pens (fountain pen or ballpoint pen). For forgery signatures, 5 samples were collected; since it was very difficult to source professional forgers volunteers were asked to simulate the true samples of all persons. They were allowed to practice many times and correct their mistakes in the final version of the forgery samples.

Table 1: Main information of the considered datasets with genuine and the fake sample statistics

Name of the Database	No of Users	Genuine per user	Forgeries per user	Year of development	Country where database was developed
GPDS100	100	24	30	2012	Spain
MCYT100	100	25	25	2003	Spain
SUSIG Visual	94	20	10	2009	Turkey
NISDCC	100	12	6 from 19 user	2009	Netherlands
SVC2004	80 (40 Western and 40 Chinese)	20	20	2004	Hong Kong
Bengali	100	24	30	2014	India
Hindi	100	24	30	2014	India
Arabic	22	10	5	2000	Egypt

In the on-line corpuses, the off-line version was obtained by 8-connecting the on-line samples through Bresenham's lines algorithm and applying the ink deposition model [22, 24, 25]. This procedure allowed an increase in the number of off-line databases in the experiments. The off-line resolution was adjusted to 600 dpi for all the databases through a bi-cubic interpolation obtained as a weighted average of pixels in the nearest 4×4 neighbourhood and using the original resolution in which the signatures were collected. A few sample images from the above-mentioned databases are shown in Figure 1. Finally, relevant information for the datasets considered are summarised in Table 1. Apart from SVC2004 and SUSIG Visual where writers are supposed to modify their original signature, the rest of the databases included real signatures used daily by the signers.

4. Merging of Databases

This section presents our signature database merging technique. The first condition to compare the performance combination of several databases with the dataset individually is that all of them contain a very similar number of users. Let $\{N_r\}_{r=1}^{R}$ be the number of users of R databases to be merged. Then, the number of users of each individual database to be selected should be $\{L_r\}_{r=1}^{R}$, holding that $L_r \approx N_r/R$ and $\sum_{r=1}^{R} N_r/R \approx \sum_{r=1}^{R} L_r$.

To avoid bias due to user selection, the L_r users of a database to be merged with the other databases are selected as follows: Let $\{EER_i\}_{i=1}^{N_r}$ be the sequence of each signer's EER for database r. Those values are sorted in ascending order, i.e. from lower to greater EER per signer as:

$$j(i) \leftarrow \mathrm{i} \mid EER_{j(i)} < EER_{j(i+1)}\,, i \in \{1, \ldots, N_r - 1\}$$

Then, we select the users to be merged by the equidistant sampling of L_r users in the j index, in other words, the selected users are those with indices $j(\lceil kN_r/L_r \rceil),\ k \in \{0, \ldots, L_r - 1\}$. This procedure

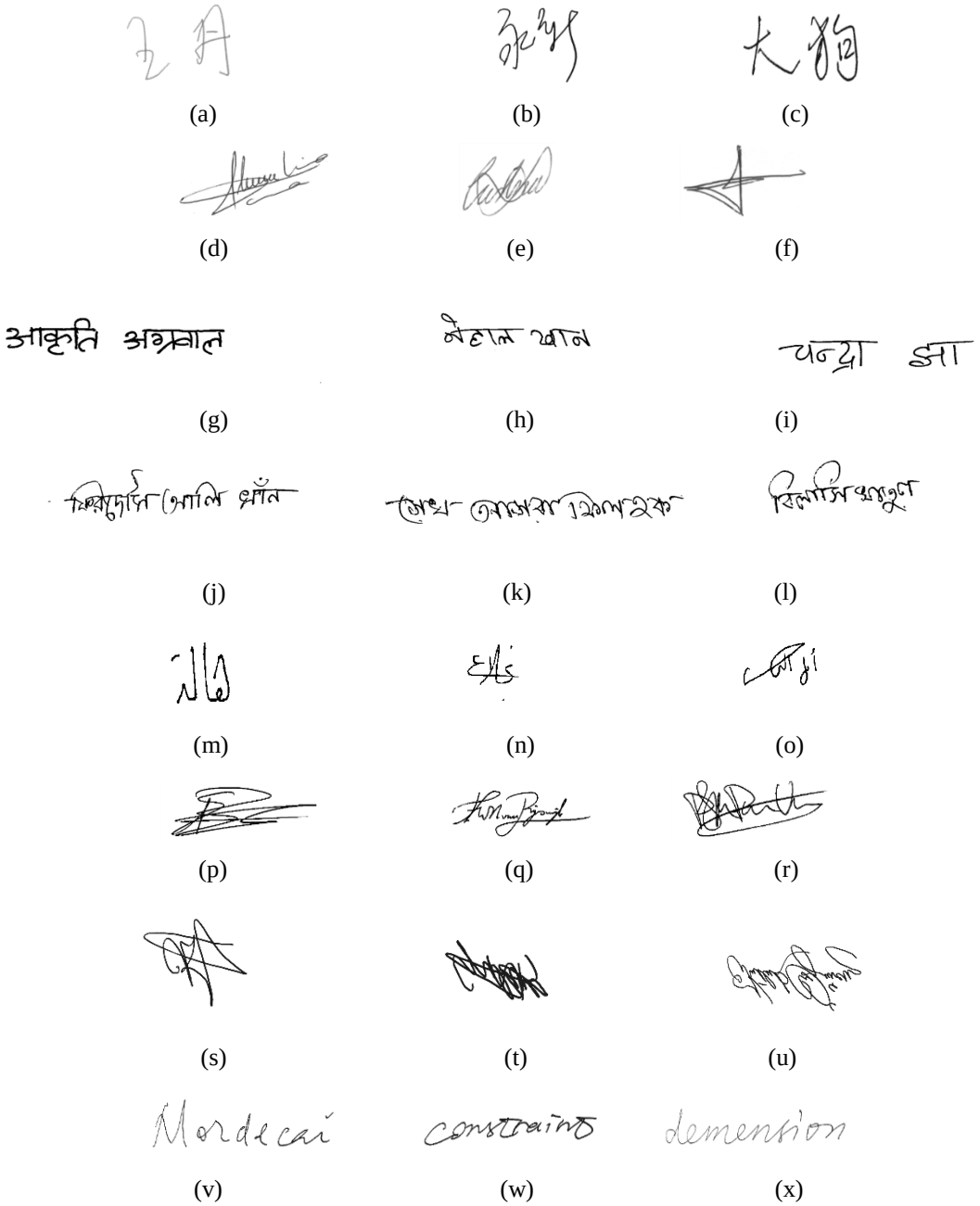

Figure 1: Sample signature images from different databases: (a-c) SVC 2004, (d-f) GPDS and MCYT, (g-i) Hindi signature, (j-l) Bengali signature, (m-o) Arab script dataset. (p-r) NFI, (s-u) SUSIG VISUAL, (v-x) SVC2004 western.

guarantees that the selected users have a similar EER distribution as the original database for a fairer merging approach and performance comparison between the merged and non-merged databases. This procedure is illustrated in Figure 2.

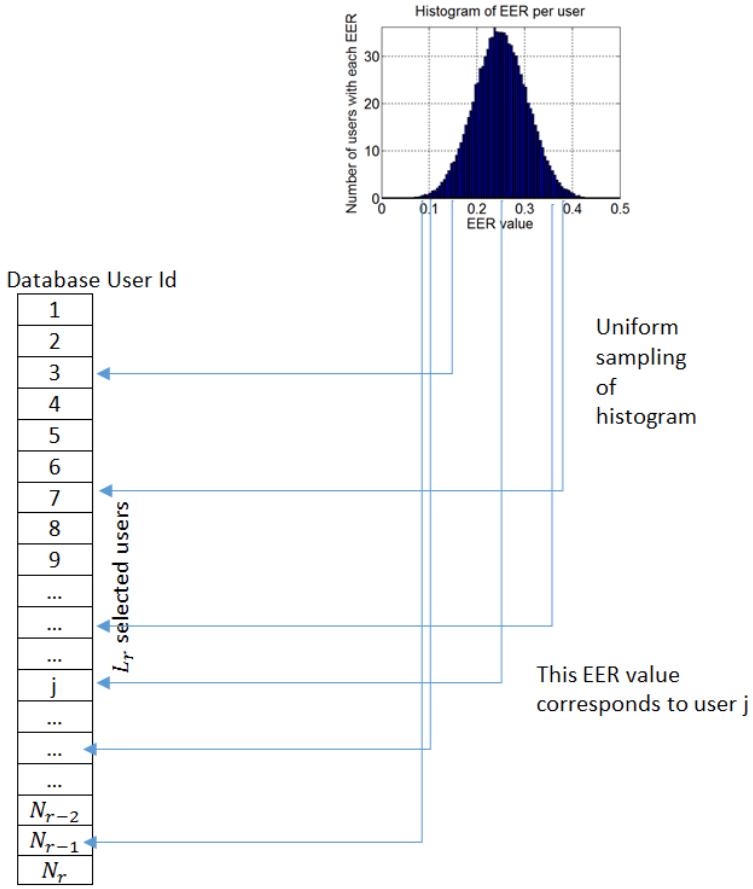

Figure 2. Similar EER distribution procedure for user selection when merging databases

5. Feature and performance analysis

Different types of feature extraction techniques that can be found in the pattern recognition literature can be classified into three categories: shape features (feature extraction based on the geometry of the object), colour-based and texture-based features. Among them, shape-based and the texture-based features are prominent for use with signatures, so they are used in the different automatic signature verifiers that can be found in the literature. The shape features aim to extract the discernible information from the signature by the ridges, blob, edges and corners present in them. Whereas, texture features give information about the spatial arrangement of colour or intensities in an image or selected region of an image locally. The texture-based features explore repetitive pattern and intensity distributions in the signature. These shape

and texture features are both applicable for multi-script signatures as well. Therefore, several geometric and texture-based feature extraction techniques are employed in this study.

The use of several features and classifiers allowed a comparison of the performance of databases and parameters to form a conclusion about the performance analysis of single-script and multi-script ASV. Here, the following published ASVs have been considered for our experimental study.

1) A Hidden Markov Model (HMM) classifier with geometrical features used in [23] is employed in our work. The signature is parameterized in Cartesian and polar coordinates. Both features are combined at the score level. The Cartesian parameters consist of equidistant samples of the height and length of the signature envelope plus the number of times the vertical and horizontal line cuts the signature stroke. In polar coordinates, the parameters are equidistant samples of the envelope radius plus the stroke area in each sector. A multi-observation discrete left-to-right HMM is chosen to model each signer's features. The classification (evaluation), decoding, and training problems are solved with the Forward-Backward algorithm, the Viterbi algorithm, and the Baum-Welch algorithm.

2) A Euclidean distance between Zernike moments used in [14] is also employed in this analysis. Zernike polynomials are an orthogonal set of complex-valued polynomials: Zernike moments have mathematical properties, and make them ideal image features to be used as shape descriptors in shape classification problems. They have rotational invariant properties and could be made to be scale and translation-invariant as well. These properties are quite adequate for ASVs, therefore they are widely used in the ASV literature. Inspired from previous work, we have employed the same in our analysis.

3) Texture-based features, as Local Binary Patterns (LBP) and a Support Vector Machine (SVM) [16], are employed. In this case, the LBP operator has been used for static signature parameterization. The grey-level image is transformed into a code matrix that is divided into 4 equal vertical blocks and 3 equal horizontal blocks, which overlap by 60%. From each block, we calculate the 255-bin histograms and the features are obtained concatenating them. A Least Square Support Vector Machine (LS-SVM) with an RBF kernel has been used as the classifier.

The three verifiers are trained with the first 5 genuine signatures of each signer in the database for repeatability of the experiments. The remaining genuine signatures are used for testing the false rejection rate. The false acceptance rate for the random forgeries have been obtained with the genuine test samples

from all the remaining users, while the false acceptance rate for the skilled forgery experiments have been worked out with all the forgery samples of each signer.

The three verifiers are trained with the first 5 genuine signatures of each signer in the database. We have chosen this procedure instead of training with 5 randomly selected samples several times and providing the averaged performance for the sake of repeatability of the experiments. It is worth mentioning the fact that our procedure introduces a certain amount of bias but this is always the same, therefore the comparison between results is fair. Moreover, we are also conscious that training with the first five samples produces poorer results than the statistical procedure.

Setting aside the 5 training samples, the remaining genuine signatures are used for testing the false rejection rate. The false acceptance rate for the random forgeries have been obtained with the genuine test samples from all the remaining users, while the false acceptance rate for the skilled forgery experiments have been worked out with all the forgery samples of each signer. For the sake of repeatability, Tables 2 and 3 show the exact number of training and testing samples for each experiment.

The results are given in both (EER) and Bhattacharyya distance for both the random and skilled forgeries. The EER measures the error point when the false acceptance and false rejection are equal, obtaining the overlap of both the distributions. On the other hand, the Bhattacharyya distance is a divergence-type measure between distributions; in this case the false acceptance and false rejection score distributions called $p(x)$ and $q(x)$ respectively, are obtained as:

Table 2: FAR and FRR statistics for each dataset used in multi-script experiments.

Name of the Database	Training samples per user	Test samples for random forgery experiments per user	Test samples for skilled forgery experiments per user
GPDS100	Positive: 5 Negative: 99*5	FRR experiment: (24-5) FAR experiment: (24-5)*99	FRR experiment: 30 FAR experiment: 30*99
MCYT100	Positive: 5 Negative: 99*5	FRR experiment: (25-5) FAR experiment: (25-5)*99	FRR experiment: 30 FAR experiment: 25*99
SUSIG Visual	Positive: 5 Negative: 93*5	FRR experiment: (20-5) FAR experiment: (20-5)*93	FRR experiment: 10 FAR experiment: 10*93
E2-Comb1 Users per database GPDS: 34 MCYT: 33 SUSIG: 33 Total: 100 users	Positive: 5: Negative: 99*5	FRR experiment: if GPDS user: (24-5) if MCYT user: (25-5) if SUSIG user: (20-5) FAR experiment: (24-5)*A+(25-5)*B+ (20-5)*C	FAR experiment If GPDS user: 30 If MCYT user: 25 If SUSIG user: 10 FRR experiment: 30*A+25*B+10*C
		if GPDS user: A=33,B=33,C=33 if MCYT user: A=34,B=32,C=33 if SUSIG user: A=34,B=33,C=32	
E2-comb2 Users per database GPDS: 20 MCYT: 20 SUSIG: 20 SVC: 21 NISDC: 19 Total: 100 users	Positive: 5: Negative: 99*5	FRR experiment: if GPDS user: (24-5) if MCYT user: (25-5) if SUSIG user (20-5) if SVC user: (20-5) if NSDCC user: (12-5) FAR experiment: (24-5)*A+(25-5)*B+(20-5)*C +(20-5)*D+(12-5)*E	FAR experiment if GPDS user: 30 if MCYT user: 25 if SUSIG user: 10 if SVC user: 20 if NSDCC user: 6 FRR experiment: 30*A+25*B+10*C+ 20*D+6*E
		if GPDS user: A=19, B=20,C=20,D=21,E=19 if MCYT user: A=20, B=19,C=20,D=21,E=19 if SUSIG user: A=20, B=20,C=19,D=21,E=19 if SVC user: A=20, B=20,C=20,D=20,E=19 if NSDCC user: A=20, B=20,C=20,D=21,E=18	
E2-comb3 Users per database GPDS: 100 MCYT: 100 SUSIG: 94 SVC: 40 (western) NISDC: 19 Total: 353 users	Positive: 5: Negative: 99*5	FRR experiment: if GPDS user: (24-5) if MCYT user: (25-5) if SUSIG user (20-5) if SVC user: (20-5) if NSDCC user: (12-5) FAR experiment: (24-5)*A+(25-5)*B+(20-5)*C +(20-5)*D+(12-5)*E	FAR experiment if GPDS user: 30 if MCYT user: 25 if SUSIG user: 10 if SVC user: 20 if NSDCC user: 6 FRR experiment 30*A+25*B+10*C+ 20*D+6*E
		if GPDS user: A=99, B=100,C=94,D=40,E=19 if MCYT user: A=100, B=99,C=94,D=40,E=19 if SUSIG user: A=100, B=100,C=93,D=40,E=19 if SVC user: A=100, B=100,C=94,D=39,E=19 if NSDCC user: A=100, B=100,C=94,D=40,E=18	

Table 3: FAR and FRR statistics for each dataset used in single script experiments.

Name of the Database	Training samples per user	Test samples for random forgery experiments per user	Test samples for skilled forgery experiments per user
GPDS100	Positive: 5 Negative: 99*5	FRR experiment: (24-5) FAR experiment: (24-5)*99	FRR experiment: 30 FAR experiment: 30*99
Hindi100	Positive: 5 Negative: 99*5	FRR experiment: (24-5) FAR experiment: (24-5)*99	FRR experiment: 30 FAR experiment: 30*99
Bengali	Positive: 5 Negative: 93*5	FRR experiment: (24-5) FAR experiment: (24-5)*99	FRR experiment: 30 FAR experiment: 30*99
E2-Comb1 Users per database GPDS: 34 Hindi: 33 Bengali: 33 Total: 100 users	Positive: 5: Negative: 99*5	FRR experiment: (24-5) FAR experiment: (24-5)*99	FRR experiment: 30 FAR experiment: 30*99
E2-comb2 Users per database GPDS: 20 Hindi: 20 Bengali: 20 SVC: 20 Chinese Arabic: 20 Total: 100 users	Positive: 5: Negative: 99*5	FRR experiment: if GPDS, user: (24-5) if Hindi user: (24-5) if Bengali user (24-5) if SVC user: (20-5) if Arabic user: (10-5) FAR experiment: (24-5)*A+(24-5)*B+(24-5)*C +(20-5)*D+(10-5)*E	FAR experiment If GPDS user: 30 If Hindi user: 30 If Bengali user: 30 if SVC user: 20 if Arabic user: 5 FRR experiment 30*A+30*B+30*C+ 20*D+5*E
		if GPDS user: A=19, B=20,C=20,D=20,E=20 if Hindi user: A=20, B=19,C=20,D=20,E=20 if Bengali user: A=20, B=20,C=19,D=20,E=20 if SVC user: A=20, B=20,C=20,D=19,E=20 if Arabic user: A=20, B=20,C=20,D=20,E=19	
E2-comb3 Users per database GPDS: 100 Hindi: 100 Bengali: 100 SVC: 40 Chinese Arabic: 22 Total: 362 users	Positive: 5: Negative: 99*5	FRR experiment: if GPDS, user: (24-5) if Hindi user: (24-5) if Bengali user (24-5) if SVC user: (20-5) if Arabic user: (10-5) FAR experiment: (24-5)*A+(24-5)*B+(24-5)*C +(20-5)*D+(10-5)*E	FAR experiment If GPDS user: 30 If Hindi user: 30 If Bengali user: 30 if SVC user: 20 if Arabic user: 5 FRR experiment 30*A+30*B+30*C+ 20*D+5*E
		if GPDS user: A=99, B=100,C=100,D=40,E=22 if MCYT user: A=100, B=99,C=100,D=40,E=22 if SUSIG user: A=100, B=100,C=99,D=40,E=22 if SVC user: A=100, B=100,C=100,D=39,E=22 if NSDCC user: A=100, B=100,C=100,D=40,E=21	

$$D_B(p,q) = -\ln(BC(p,q))$$

being $BC(p,q)$ the Bhattacharyya coefficient defined as:

$$BC(p,q) = \int \sqrt{p(x)q(x)}dx$$

In statistics, the Bhattacharyya distance measures the similarity of two discrete or continuous probability distributions. It is a measure of the amount of overlap between two statistical samples or distributions.

Therefore, the relative closeness of the two samples or distributions are being considered. It is used to measure the separability of classes in a classification problem. In single-script vs. multi-script signatures, the ASV scenario is similar. In this scenario, we need to measure the overlap between the two distributions rather than the accuracy of the system which can get affected also due to increase in the database size.

6. Experimental setup, results and discussion

The following experimental setups were followed in the proposed set of experiments.

6.1. *Single-script scenario*

In the case of the single-script scenario, two experiments were conducted to study the effect of merging databases. The first experiment (E1) studied three individual databases: GPDS100, MCYT100 and SUSIG Visual. They were selected because of the similarity in their number of users. The first two databases were recorded under similar conditions while the third one is different in both methodology and geographical location.

The second experiment is divided into two protocols. The first one (E2-comb1), merges equally from the three above databases. This way, we select 34, 33 and 33 users from each database respectively by the *similar EER distribution* criterion to obtain a 100-user database. The second one (E2-comb2) merges 20 users from each of the three databases mentioned above, which are selected with a *similar EER distribution* criterion and the first 21 English signatures from the SVC2004 database as well as 19 users from the NSDCC database that includes forgeries. The two last databases were recorded under different conditions from the first three.

The next experiment (E2-comb3) compares the performance of the *Similar EER distribution* with respect to aggregate databases. This experiment sums all the aforementioned databases: GPDS100, MCYT100, SUSIG, SVC2004 and NSDCC and measures its performance.

6.2. *Multi-Script scenario*

To study the effects of the multi-script scenario in off-line automatic signature verification, two experiments have been carried out again. As above, the first experiment (E1) studies individually the GPDS100, Hindi and Bengali databases which contain 100 users in each of them. Note that the Hindi and Bengali databases were recorded under similar conditions whilst the GPDS100 was recorded differently. Again, the second experiment was performed in two steps. The first one (E2-comb1), merges 34, 33 and 33 users of the above databases selected by the *similar EER distribution* criterion. Finally, the second one (E2-comb2), merges 20 users selected from the above three databases with the first 20 users of the Arabic database and the first 20 Chinese signatures of the SVC2004 database. Again these two databases were recorded under different

conditions compared to the first three. Therefore, the comparison between single-script and multi-script results is expected to be statistically fair. In E2-comb3, to compare the performance of similar EER distributions with respect to aggregate databases, this experiment adds all the aforementioned databases and measures their performance.

6.3 *Results and discussion*

The EERs-based results are given in Table 4 and Table 5 for single and multi-script experiments, whereas the Bhattacharyya distances are given in Tables 6 and 7, respectively. A DET curve of the multi-script and singe-script signature environments for both random and skilled forgeries are shown in Figures 3 and 4.

Analysing the single script random forgery experiment results, (Table 4), it can be seen that aggregating all the databases (E2-comb3) worsens the performance significantly with respect to the single script database. It is mainly due to the increment of users. Instead, mixing the databases using the similar EER distribution procedure (E2-comb1) keeps a similar performance as compared to the single database scenario (E1-GPDS, E1-MCYT and E1-SUSIG). The small decrease in performance is due to the fact that each database contains a watermark that can help the classifier to decide on borderline signatures. This reduction is much clearer when adding 5 databases using the similar EER distribution mixing of the databases (E2-comb2). Given that the last two databases added (SVC2004 and NSDCC) were very different to the three first ones (GPDS, MCYT and SUSIG), the watermark of the database undoubtedly helps the classifier to decide.

In the random forgery multi-script scenario, (i.e. Table 5), we noted similar findings. To aggregate the different scripts provides a significantly worse result (E3-comb2) as opposed to blending the different scripts properly, i.e. using the EER similar distribution procedure. In this case, the mixing of different scripts barely decreases the performance (E2-comb1 and E2-comb 2) with respect to the single script database (E1-GPDS, E1-Hindi and E1-Bengali) in Geometric and Zernike-based classifiers. This fact, in comparison to the result in the single script case, is surprising since the different scripts should spread the parameters helping the classifiers to improve their performance. Instead, the performance is improved significantly with the Texture-based classifier. This result contributes to the recommendation that texture-based classifiers should be used in multi-script environments. Tables 6 and 7 confirm the results of Table 4 and 5. Usually, when the EER increases, the Bhattacharyya distance decreases. We found some exceptions in the doubtful cases, helping to make the conclusion clearer. For instance, Table 4, Geometric case in Random Forgeries. We have stated that E2-comb1 and EER-comb2 are barely better than E1-SUSIG because the EER are similar but the Bhattacharyya distances are greater for all cases.

In skilled forgeries, the results are not as clear as the above. The EER does not display a clear tendency. This fact is given due to the different skills of the forgers along with the different databases. Despite the

similarity of the EER distributions, we mixed the skilled forgeries of the GPDS dataset with those of the MCYT dataset, which are less skilful and so on. Thus, the result of blending the skilled forgeries is more affected by the skill of the individual databases than by the mixing procedure.

Looking over the DET curves, they highlight the effect of the similar EER distribution procedure. It is easy to realise that in all the curves there is one odd curve which corresponds to the E2-comb3 experiments. It is due to both the difference in performance and the mixing of different distributions that distort the DET curve. For instance, this fact is clearly seen in the two cases of the texture-based classifier for random forgeries.

Table 4. Single-script results for random and skilled forgeries in terms of EER in precent (%)

	Random Forgeries			Skilled Forgeries		
Experiment	Geometric	Zernike	Texture	Geometric	Zernike	Texture
E1- GPDS	4.72	25.07	2.05	22.50	35.16	18.80
E1-MCYT	4.21	23.06	1.78	19.98	35.51	16.07
E1-SUSIG	3.44	23.07	1.47	31.95	44.73	28.81
E2-comb1	4.10	21.27	1.02	24.84	42.19	21.23
E2-comb2	3.15	20.09	1.09	23.59	41.48	17.30
E2-comb3	7.78	30.41	2.55	38.76	54.16	34.64

Table 5. Multi-script results for random and skilled forgeries in terms of EER in precent (%)

	Random Forgeries			Skilled Forgeries		
Experiment.	Geometric	Zernike	Texture	Geometric	Zernike	Texture
E1-GPDS	4.72	25.07	2.05	22.50	35.16	18.80
E1-Hindi	3.85	17.71	1.22	16.83	20.83	12.16
E1- Bengali	3.35	15.28	1.52	20.97	21.35	12.82
E2-comb1	4.73	25.17	1.20	21.25	32.58	15.62
E2-comb2	5.43	27.04	0.94	24.54	32.55	24.24
E2-comb3	7.89	32.61	2.55	29.56	43.56	29.23

Table 6. Bhattacharyya distance between the densities of genuine and forgery scores for the single-script experiment

	Random Forgeries			Skilled Forgeries		
Experiment	Geometric	Zernike	Texture	Geometric	Zernike	Texture
E1- GPDS	1.21	0.24	2,18	0.26	0.06	0.40
E1-MCYT	1.19	0.22	2,31	0.34	0.03	0.51
E1-Susig	1.32	0.24	2,45	0.11	0.01	0.14
E2-comb1	1.38	0.25	3,20	0.22	0.01	0.30
E2-comb2	1.47	0.29	3,14	0.24	0.01	0.47
E2-comb3	0.98	0.25	1.69	0.16	0.01	0.21

Table 7. Bhattacharyya distance between the densities of genuine and forgery scores for the multi-script experiment

	Random Forgeries			Skilled Forgeries		
Experiment	Geometric	Zernike	Texture	Geometric	Zernike	Texture
E1- GPDS	1.21	0.24	2.18	0.26	0.06	0.40
E1-Hindi	1.29	0.48	2.47	0.31	0.28	0.63
E1- Bengali	1.29	0.42	2.68	0.41	0.31	0.65
E2-comb1	1.32	0.34	2.70	0.30	0.05	0.51
E2-comb2	1.31	0.37	2.86	0.19	0.06	0.23
E2-comb3	0.63	0.35	1.43	0.11	0.02	0.17

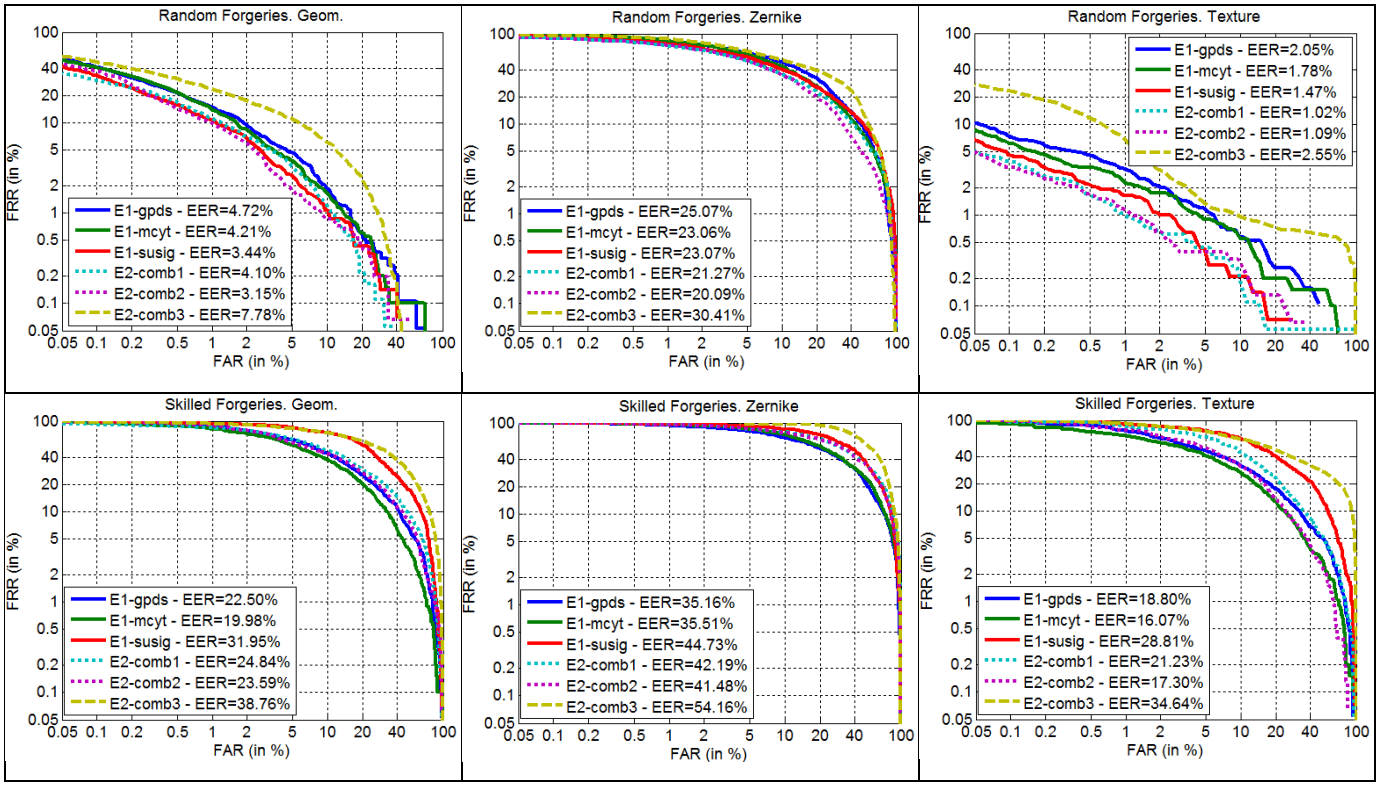

Figure 3: DET curves of the multi-script and single script signature environments for random forgeries.

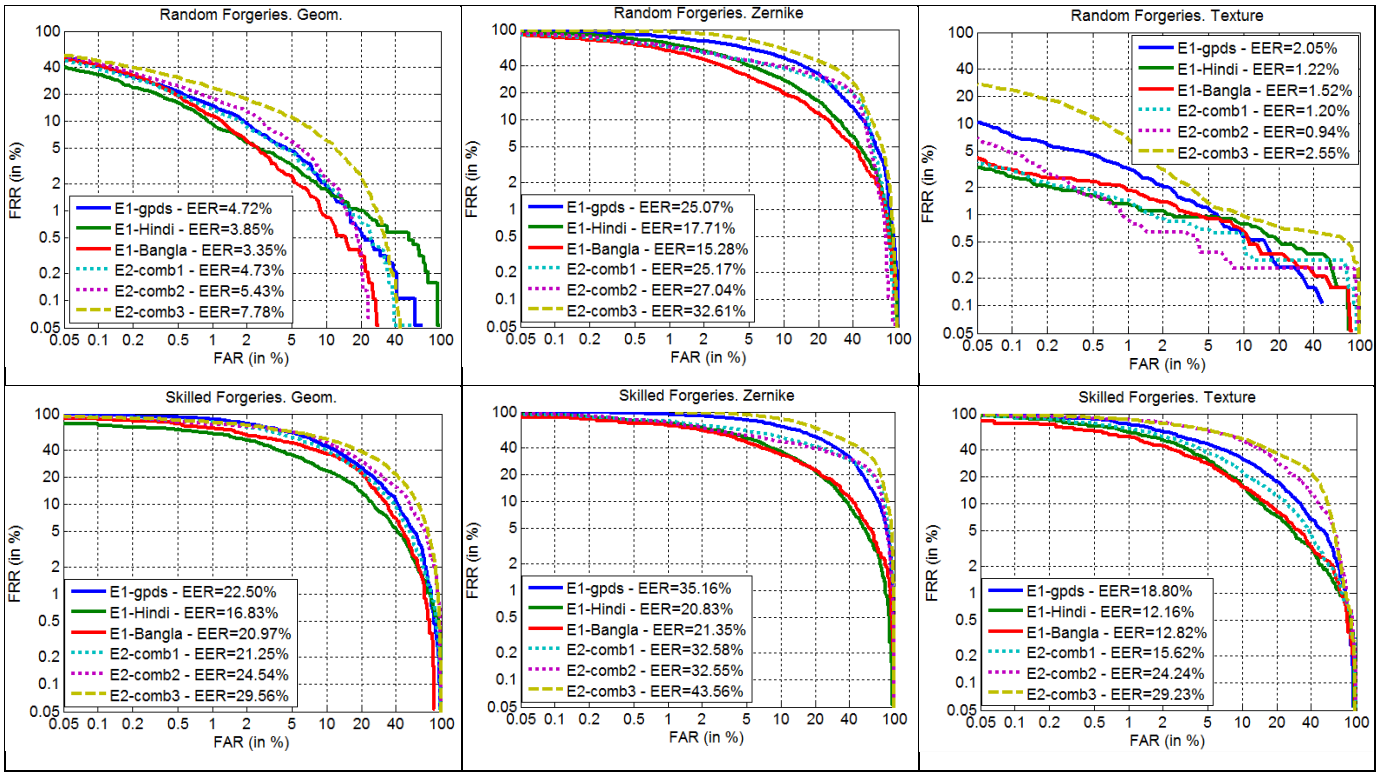

Figure 4: DET curves of the multi-script and single script signature environment for skilled forgeries.

Analysing the DET curves in detail, similar conclusions are drawn. By observing and comparing individual curves of experiments of E1 with the experiments of aggregating databases E2-com3, it seems that the multi-script database performs less accurately. Whereas, comparing the E1 experiment with the merged datasets using similar EER distribution i.e. the experiments of E2-comb1 and E2-comb2, the performance seems to be very similar or better. Therefore the problem is not merging the signatures from different scripts but merging datasets or increasing the population. This fact is also clear from the Bhattacharyya distances of the E1, E2-comb1, E2-comb2 and E2-comb3, whereby the different Bhattacharyya distances of this experimental pair is less than the EER differences that this combination poses.

It can be found in the random forgeries that merging of databases implies a reduction of the EER and an increment of the Bhattacharyya distance in both scenarios except for Zernike features in the multi-script case. So it shows that the score distributions are less overlapping and there is more distinction between them. It is supposed to be due to the features of the different databases, which are statistically different for each database reducing the confusion among signatures, and this is regardless of the single or multi-script properties of the database.

In the case of Zernike features, experiments show that these features are oriented to Hindi and Bengali scripts because of:

1) The EER obtained with Hindi and Bengali are similar between them,

2) The achieved EER with Latin, Arabic and Chinese are similar between them, and

3) The EERs with Hindi and Bengali are significantly lower than the EERs with Latin, Arabic and Chinese.

Therefore, if we properly merge just the Bengali and Hindi datasets, i.e. merging by means of the similar EER distribution procedure, the error of the merged database is lower. Similar observations are made when the Latin, Arabic and Chinese datasets are merged. Therefore we concluded that the merged databases have to display similar performance if they are fairly compared with their merged versions.

7. Conclusions

In this paper, we have designed and implemented a novel method to merge individual databases according to the statistical similarities in the user-performance distribution. We also proposed a statistical performance analysis method namely the Bhattacharyya distance for a fair comparison between single and multi-script scenarios. We conclude that the performance of the merged databases is slightly better than or quite similar to the individual ones if the individual databases i) have a similar performance, ii) are merged keeping the same number of users, and iii) if the users are selected on the basis of similar EER. This analysis advocates

that the lower performance of the multi-script signatures that was reported in the literature in contrast to the single script scenario is not due to the presence of multi-script signatures. Rather it is due to the increase in the number of classes while merging the databases. Furthermore, this finding is independent of the single-script or multi-script properties of the merged databases. Therefore, the multi-script automatic signature verifier can be seen as an interoperability problem from the system point of view or a generalization problem from the pattern recognition perspective. This research opens the door for applying generalization and interoperable techniques to the multi-script signature problem.

Acknowledgments

This study was funded by the Spanish government's MIMECO TEC2012-38630-C04-02 research project, partially funded with European Union, FEDER funds.

References

[1] Pal S., Blumenstein M., Pal U.: 'Non- English and Non-Latin Signature Verification Systems: A Survey', Proceedings of the 1st International Workshop on Automated Forensic Handwriting Analysis, pp. 1-5, Beijing, China, September 2011.

[2] Pal S., Pal U., Blumenstein M.: 'Multi-script Off-line Signature Verification: A Two Stage Approach', Proceedings of the 2nd International Workshop on Automated Forensic Handwriting Analysis, pp. 31-35, Washington D.C., USA, September 2013.

[3] Plamondon, R. , Lorette, G. ,: Automatic signature verification and writer identi- fication—the state of the art, Pattern Recognition. No. 22 pp 107–131, 1989.

[4] Leclerc, F., Plamondon, R., :Automatic signature verification: the state of the art, 1989–1993, Int. J. Pattern Recognit. Artif. Intell. Vol. 8 , pp. 643–660, 1993.

[5] Fairhurst, M., Signature verification revisited: promoting practical exploitation of biometric technology, Electronics. Communication. Eng. J. vol. 9, 273–280, 1997.

[6] Plamondon, R., Srihari, S.N., : On-line and off-line handwriting recognition: a comprehensive survey, IEEE. Trans. Pattern Anal. Mach. Intell. Vol 22, pp. 63–84, 2000.

[7] Impedovo, D. , Pirlo, G. , :Automatic signature verification: the state of the art, IEEE Trans. Syst. Man Cybern. Part C: Appl. Rev. vol. 38 no. 5, pp. 609–635, 2008.

[8] Diaz-Cabrera, M., Morales, A., Ferrer, M. A., : Emerging Issues for Static Handwritten Signature Biometric, (Eds.), Advances in Digital Handwritten Signature Processing. A Human Artefact for e-Society, Word Scientific, pp. 111-122, 2014.

[9] Diaz-Cabrera, M., Ferrer, M. A., Morales, A., "Modeling the Lexical Morphology of Western Handwritten Signatures", PLoS ONE 10(4): e0123254, April 2015.

[10] Pal, S. , Alireza, A., Pal U. , and Blumenstein, M. , : Multi-script off-line signature identification", 12th Int. Conf. on Hybrid Intelligent Systems, Pune, India, pp. 236–240 2012.

[11] Pal S., Pal U., Blumenstein M,:' Hindi and English Off-line Signature Identification and Verification", Proceedings of International Conference on Advances in Computing. Advances in Intelligent Systems and Computing Ed, vol 174, pp 905-910. 2012.

[12] Pal S., Pal U., Blumenstein M,:'A Two-Stage Approach for English and Hindi Off-line Signature Verification', International Workshop on Emerging Aspects in Handwritten Signature Processing, Naples, Italy, September 9-10, 2013.

[13] Pal, S.; Alireza, A.; Pal, U.; Blumenstein, M., "Off-line Signature Identification Using Background and Foreground Information," in Digital Image Computing Techniques and Applications (DICTA), 2011 International Conference on , pp.672-677, 6-8 Dec. 2011.

[14] Pal S., Pal U., Blumenstein M., 'Off-line English and Chinese signature identification using foreground and background features, Proceedings of the 2012 International Joint Conference on Neural Networks, pp.1,7, Brisbane, Australia, June 2012.

[15] Ferrer M., Vargas F., Morales A., Ordoñez A.: 'Robustness of Off-line Signature Verification Based on Grey Level Features', in IEEE Transactions on Information Forensics and Security, vol. 7, no. 3, pp. 966-977, June 2012.

[16] Ortega-Garcia J., Fierrez-Aguilar J., Simon D., Gonzalez J., Faundez M., Espinosa V., Satue A., Hernaez I., Igarza J., Vivaracho C., Escudero D. Moro Q.: 'MCYT baseline corpus: A bimodal biometric database', in IEE Proceedings Vision, Image and Signal Processing, Special Issue on Biometrics on the Internet, vol. 150, no. 6, pp. 395-401, December 2003.

[17] Kholmatov A., Yanikoglu B., "SUSIG: an on-line signature database, associated protocols and benchmark results", in Pattern Analysis and Applications, vol. 12, pp. 227-236, 2009.

[18] Blankers V., Heuvel C., Franke K, Vuurpijl L.: 'Signature verification competition', in 10th International Conference on Document Analysis and Recognition, pp. 1403-1407, Barcelona, Spain, 2009.

[19] Yeung D., Chang H., Xiong Y., George S., Kashi R.: 'SVC2004: First international signature verification competition', in Lecture Notes in Computer Science: Biometric Authentication, Ed. by D. Zhang and A. Jain, vol. 3072, pp. 16-22, Springer, Berlin Heidelberg, 2004.

[20] Pal S.: "Multi-Script Off-line Signature verification", PhD. Dissertation, Griffith University, October 2014.

[21] Ismail M., Gad S.: ‘Off-line Arabic signature recognition and verification’, in Pattern Recognition, vol. 33, pp. 1727-1740, 2000.

[22] Ferrer M., Diaz M., Morales M., "Static Signature Synthesis: A Neuromotor Inspired Approach for Biometrics," in proceedings of IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 37, no.3, pp.667-680, March 2015.

[23] Ferrer M., Alonso J., Travieso C.: ‘Offline geometric parameters for automatic signature verification using fixed-point arithmetic’, in IEEE Transactions on pattern analysis and machine intelligence, vol. 27, no. 6, pp 993-997, June 2005.

[24] Ferrer, M. A., Diaz-Cabrera, M., Morales, A., Galbally, J., Gomez-Barrero, M., "Realistic Synthetic Off-Line Signature Generation Based on Synthetic On-Line Data", 47th IEEE International Carnahan Conference on Security Technology, Medellin, pp. 116-121, 8-11 October 2013.

[25] Ferrer, M. A., Diaz-Cabrera, M., Morales, A., "Synthetic Off-Line Signature Image Generation", 6th IAPR International Conference on Biometrics, Madrid, pp. 1 – 7, 4-7 June 2013.