Information Fusion 32 (2016) 109–119

Contents lists available at ScienceDirect

Information Fusion

journal homepage: www.elsevier.com/locate/inffus

# Score level fusion of classifiers in off-line signature verification

Mustafa Berkay Yılmaz*, Berrin Yanıkoğlu

*Faculty of Engineering and Natural Sciences, Sabancı University, İstanbul, Turkey*

ARTICLE INFO

*Article history:*
Available online 12 February 2016

*Keywords:*
Offline signature
Score-level fusion
Histogram of oriented gradients
Local binary patterns
Scale invariant feature transform

ABSTRACT

Offline signature verification is a task that benefits from matching both the global shape and local details; as such, it is particularly suitable to a fusion approach. We present a system that uses a score-level fusion of complementary classifiers that use different local features (histogram of oriented gradients, local binary patterns and scale invariant feature transform descriptors), where each classifier uses a feature-level fusion to represent local features at coarse-to-fine levels. For classifiers, two different approaches are investigated, namely global and user-dependent classifiers. User-dependent classifiers are trained separately for each user, to learn to differentiate that user's genuine signatures from other signatures; while a single global classifier is trained with difference vectors of query and reference signatures of all users in the training set, to learn the importance of different types of dissimilarities.

The fusion of all classifiers achieves a state-of-the-art performance with 6.97% equal error rate in skilled forgery tests using the public GPDS-160 signature database. The proposed system does not require skilled forgeries of the enrolling user, which is essential for real life applications.



## 1. Introduction

Signature verification is used in verifying the claimed identity of a person through his/her chosen and previously registered signature. The signature's widespread acceptance by the public and niche applications (validating paper documents and use in banking applications) makes it a desirable biometric.

Signature is considered to be a behavioral biometric that encodes the ballistic movements of the signer and as such is difficult to imitate. On the other hand, compared to physical traits such as fingerprint, iris or face, a signature typically shows higher intraclass and time variability. Furthermore, as with passwords, a user may choose a simple signature that is easy to forge.

Depending on the signature acquisition method used, automatic signature verification systems can be classified into two groups: online (dynamic) and offline (static). A static signature image is the only input to offline systems, while signature trajectory as a function of time is also available in online signatures. Main difficulties in both tasks are simple (easy to forge) signatures and variations among a user's signatures, but the dynamic information available in online signatures make the signature more unique and more difficult to forge.

Research databases define two types of forgeries: a *skilled forgery* refers to a forgery which is signed by a person who has had access to some number of genuine signatures and practiced them for some time. In contrast, a *random forgery* is typically collected from other people's real signatures, simulating the case where the impostor does not even know the name or shape of the target signature and hence uses his/her own for forgery. Random forgery detection is a much easier task compared to skilled forgery detection. In this work, as in the literature, when the term "forgery" is used without further qualifications, it may refer to either a skilled forgery or random forgery.

Systems' evaluation is often done in terms of the *Equal Error Rate (EER)* which is the point where the False Accept Rate (FAR) and False Reject Rate (FRR) are equal and occasionally in terms of the Distinguishing Error Rate (DER), which is the average of FAR and FRR.

While the use of the public signature databases has become the norm in the last years, the databases do not always have strictly specified protocols. As a result, many reported accuracies cannot be directly compared with if they use a different (sometimes random) subset of the users; or a different number of reference signatures (using more helps the system as it provides more information); or a different number of skilled forgeries.

In this work, we present a state-of-the-art offline signature verification system that uses a fusion of complementary features, classifiers and preprocessing techniques, with the aim to explore the limits in signature verification accuracy.

* Corresponding author. Tel.: +90 216 4839528; fax:+90 216 4839550.
*E-mail addresses:* berkayyilmaz@sabanciuniv.edu (M.B. Yılmaz), berrin@sabanciuniv.edu (B. Yanıkoğlu).

http://dx.doi.org/10.1016/j.inffus.2016.02.003
1566-2535/

Our main contribution is the comprehensive study and treatment of different aspects of offline signature verification, which are fused at the end to form a state-of-the-art verification system, with novel aspects including the following:

- We propose an alignment algorithm that improves overall accuracy by more than 2% on average. While alignment of test images degrades overall performance, we have found that automatic alignment of references is when used with the global classifiers.
- We improve on the use of the well-known features and approaches by novel adaptations. (i) We use coarse-to-fine grids for capturing a spectrum of global to local features when using the histogram of oriented gradients (HOG) and local binary patterns (LBP). (ii) We select the best LBP templates according to term frequencies and combine similar LBP template histogram bins to obtain a dense histogram. (iii) We use a novel scale invariant feature transform (SIFT) descriptor matching algorithm that seeks more than one global transformation in order to allow different transformations in different parts of a signature.
- We incorporate user-dependent and user-independent verification concurrently. We apply a score level fusion to combine classifiers with complementary feature types, where the weights are learnt from a separate validation set.

## 2. Literature review

Offline signature verification is a well-researched topic where many different approaches have been studied. A series of surveys covering advances in the field are available [1–10]. Here, we review some of the recent works, grouped according to focus areas.

Note that while we give some performance figures for completeness, many of the reported numbers are not directly comparable as they are obtained under different conditions (number of reference signatures, use of skilled signatures etc.). We discuss this issue in Section 6.6.

*Feature extraction*

Several different features are used in offline signature verification, especially local features such as SIFT descriptors, wavelet features and LBP, among others. Solar et al. use SIFT descriptors in conjunction with the Bayes classifier [11]. The performance is assessed using the GPDS-160 signature dataset, with a 15.3% DER. However, only a small subset of all skilled forgeries, and not the full test set, is used for testing.

Vargas et al. use complex features based on LBP to perform statistical texture analysis [12]. To extract second order statistical texture features from the image, another feature called the gray level co-occurrence matrix method is utilized. The best combination of features is reported to achieve an EER of 9.02% on the gray-level GPDS-100 database, using 10 reference signatures.

*Different base classifiers*

Ferrer et al. [13] have evaluated the effectiveness of hidden Markov models (HMMs), support vector machines (SVMs) and the Euclidean distance classifier on the publicly available GPDS-160 database. When 12 genuine signatures and 3 skilled forgeries are used in training the classifiers, the DER rates are found as 13.35%, 14.27% and 15.94% for the HMMs, SVM (radial basis function kernel) and the Euclidean distance classifier, respectively.

A comparison of probabilistic neural networks (PNN) and K-nearest neighbor (KNN) is done by Vargas et al. [14]. Genuine and skilled forgery signatures of each subject are divided into two equal parts, resulting in 12 genuine and 12 skilled forgeries in train set and the same amount in the test set. The results on the gray-level GPDS-160 database are found to be close: the best results are found to be 12.62% DER with the KNN ($k = 3$) and 12.33% DER with the PNN.

*Use of classifier combination*

There are quite a lot of studies on the effect of classifier combination in offline signature verification. In one of the earlier works, Fierrez-Aguilar et al. consider the sum rule for combining global and local image features [15]. One of the experts in this work is based on a global image analysis and a statistical distance measure, while the second one is based on local image analysis with HMMs. It is shown that local information outperforms the global analysis in all reported cases. The two proposed systems are also shown to give complementary recognition information, which is desired in fusion schemes.

Receiver operating characteristic (ROC) curves are used for classifier combination by Oliveira et al. [16]. Different fusion strategies to combine the partial decisions yielded by SVM classifiers are analyzed and the ROC curves produced by different classifiers are combined using the maximum likelihood analysis. Authors demonstrate that the combined classifier based on the writer-independent approach reduces the FRR, while keeping FAR at acceptable levels.

An ensemble of classifiers based on graphometric features is used to improve the reliability of the classification by Bertolini et al. [17]. A pool of base classifiers is first trained using only genuine signatures and random forgeries; then an ensemble is built using genetic algorithms with two different scenarios. In one, it is assumed that only genuine signatures and random forgeries are available to guide the search; while simple and simulated forgeries also are assumed to be available in the second one. Different objective functions are derived from the ROC curves, for ensemble tuning. A private database of 100 writers is utilized for evaluation, considering 5 genuine references for training and only skilled forgeries for testing. The best result is found as 11.14% DER using the area under curve optimization.

Score level combination is examined for offline signature verification by Prakash and Guru [18]. Classifiers of distance and orientation features are used individually and in combination. Distance features and orientation features individually provide 21.61% and 19.88% DER on the MCYT-75 corpus. The max fusion rule decreases the DER to 18.26%, while the average rule decreases the DER to 17.33% when the weights are fixed empirically.

Hybrid generative discriminative ensemble of classifiers is proposed by Batista et al. to design an offline signature verification system from few references, where the classifier selection process is performed dynamically [19]. To design the generative stage, multiple discrete left-to-right HMMs are trained using a different number of states and codebook sizes, allowing the system to learn signatures at different levels of perception. To design the discriminative stage, HMM likelihoods are measured for each training signature and assembled into feature vectors that are used to train a diversified pool of two-class classifiers through a specialized random subspace method. The most accurate ensembles are selected based on the K-nearest-oracles algorithm. The GPDS-160 database is used to evaluate the system and 16.81% EER is reported using 12 references per user.

An offline signature verification system using two different classifier training approaches is proposed by Hu and Chen [20]. In the first mode, each SVM is trained with feature vectors obtained from the reference signatures of the corresponding user and random forgeries, while the global Adaboost classifier is trained using genuine and random forgery signatures of signers that are excluded from the test set. Global and user-dependent classifiers are used separately. Combination of all features for writer-dependent SVMs results in 7.66% EER for 150 randomly selected signers from the gray-level GPDS-300 dataset, using 10 references. The writer-independent Adaboost using a combination of all features results in 9.94% EER for a random 100-person subset of the same dataset, again using 10 references.

Fierrez-Aguilar et al. give a comprehensive survey of different fusion strategies in the context of multimodal biometrics, but the results are also applicable to single modality combination [21]. Different approaches are categorized as global fusion-global decision, local fusion-global decision, global fusion-local decision, and local fusion-local decision. Adapted fusion and decision methods are also proposed in the same work, using both the global and local information.

In summary, among the systems that report accuracies on skilled forgery tests on the GPDS-160 database which is also used in this work, the best DER is reported to be 16.81% with 12 reference signatures [19]. There are other systems that report lower DERs; however we cannot fully compare our results with theirs because they report DERs on slightly non-standard versions of the dataset (a random subset of users) or with different testing protocols (e.g. with less skilled forgeries). Previous works that have reported results on GPDS dataset are listed in Section 6, following our proposed system described in Sections 3–5.

## 3. Preprocessing

Signature images have variations in terms of pen thickness, embellishments found in strokes, translation or relative position of strokes, rotation, scaling even within the genuine signatures of the same subject. In order to gain invariance to such natural variations, images should be normalized before they are further processed.

To compensate for large translation variations that would result from embellishments, we discard strokes that are far away from image centroid. This is done using a distance threshold which is derived from the standard deviation of the coordinates of trajectory points ( $\approx 3\sigma$).

To compensate for pen thickness variations, we find the upper and lower contours of the signature. Skeletonisation is another alternative, but it loses some details, as can be seen in Fig. 1. This

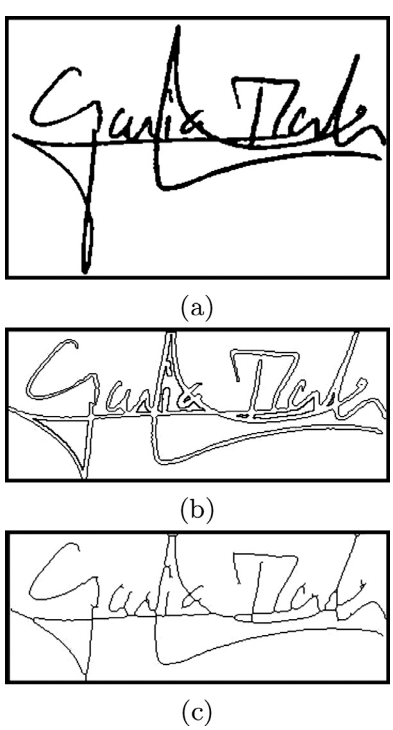


**Fig. 1.** Preprocessing (a) Original signature (b) Contour image (c) Skeleton image.

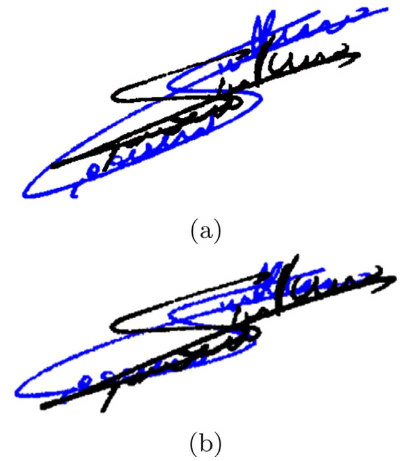


**Fig. 2.** Alignment example: (a) not aligned and (b) aligned reference and query.

step is found to be useful in feature types that use gradient information (HOG), while features that use texture information, namely LBP and SIFT, are directly extracted from the image.

For the effects of rotation, scaling and fine translation, we use the following alignment procedure. Each query signature image $Q$ of a training user is aligned to each reference signature $R^i$ of that user, with the best scaling ($\sigma$), rotation ($\theta$) and translation ($\delta$) parameters minimizing the distance between the query and reference image:

$$argmin_{\sigma,\theta,\delta}\{\|Q^i_{\sigma,\theta,\delta} - R^i\|\}, \tag{1}$$

where $Q^i_{\sigma,\theta,\delta}$ is the transformed version of $Q$.

We use the $\ell_2$-norm of the difference between LBP features. In fact, for faster alignment during testing, we apply all possible transformations to each enrolled reference $R^i$, ahead of time and apply the inverse transformation to $Q$ with parameters $1/\sigma$, $-\theta$ and $-\delta$. An example reference, query and aligned query are shown in Fig. 2.

We use a small interval to search for best transformation: $-2.5$ to $+2.5$ degrees for $\theta$, 0.8 to 1.2 for $\sigma$, $-10$ to $+10$ pixels for $\delta$. These intervals seem to be enough as there are no significant alignment differences in the used database. Larger parameter intervals naturally increase the cost of search and should be handled by more sophisticated methods such as iterative closest point (ICP) or random sample consensus (RANSAC) algorithms.

Note that preprocessing that removes individual characteristics (e.g. signer always signs with a 20 degree slope) may lead to performance degradation in biometric systems. Furthermore, in alignment, a sufficiently similar forgery can be made very similar to a genuine signature using a complex transformation such as a non-linear scaling. In that case, the system should either incorporate a cost measure reflecting the effort needed for alignment (e.g. similar to a dynamic time warping algorithm); or it should only use simple transformations (e.g. scaling, rotation) within a limited range, as done in our work.

In the proposed system, signature alignment is implemented only on the training phase of the global classifier, so as to obtain features that are better aligned with the reference signatures. It is experimentally observed that the alignment algorithm improves performance as reported in Table 3, but alignment of test signatures or alignment with user-dependent classifiers do not improve the overall performance.

## 4. Feature extraction

Feature extraction step reduces the dimension of original signature images while preserving and extracting the important

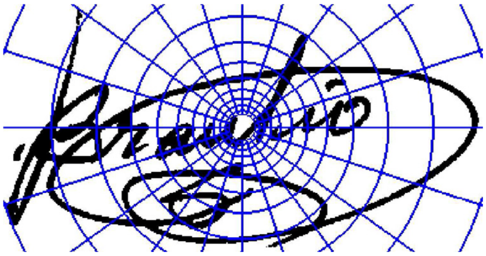


**Fig. 3.** Log-polar grids, origin taken as the image center.

**Fig. 4.** Log-polar grids, origin taken as the top-left corner.

information encoded in the image to distinguish between genuine and forgery classes. We utilize a complementary set of features that are commonly reported to be successful in the context of offline signature verification, namely HOG, LBP and SIFT features. After describing the grids where local features are extracted in Section 4.1, the features are explained in detail in Sections 4.2–4.4.

### 4.1. Grids in cartesian and polar coordinates

In order to develop a system robust to global shape variations, we extract features from local zones of the signature image. We evaluated two different alternatives:

**Cartesian grids:** First and most common choice of grids in many works is the rectangular grids in Cartesian coordinates. The grids may be overlapping to capture the signature at grid boundaries, or non-overlapping. We use overlapping grids which are found to perform better.

**Log-polar grids:** Another choice of coordinate system is the log-polar coordinate system. If the registration point is selected as the top-left point of the bounding box and the embellishments are on the right, then the left parts of the two signatures align better than the right. With this observation and at the cost of having some redundant features, we decide to use multiple registration points (center, top-left, top-right and so on) in the polar grid, to reduce the effect of registration mismatches.

Using multiple fixed registration points is motivated by the fact that there are no reference points in signatures, unlike face (e.g. eyes) or to some degree fingerprints (core point). The centroid or center of mass can be used as a lesser alternative in registering two signatures. Unfortunately, the location of both of these points may show large variations due especially to large variations in embellishment. Another alternative could be the use of Local Self-Similarities (LSS) as proposed by Shechtman and Irani [22] to extract reference points for signature matching. However, LSS tends to provide good matching results for texturally rich samples, which is not the case for binary signatures.

A sample signature divided into regions in log-polar space is shown in Fig. 3 where the origin is taken as the image center. Same signature with overlaid log-polar grids where the top-left corner is used as the origin is shown in Fig. 4.

**Hierarchical representation:** Using a small number of grids will result in features that are almost globally extracted, losing location information. In contrast, using a large number of grids will decrease the system's ability to allow for small deformations.

To eliminate the need for searching the ideal grid resolution, we use a hierarchy of grids in increasing resolution and thus extract coarse to fine features. In the top-level, the single grid corresponds to the full image, while lower levels have increasingly more grid zones. Features extracted from all levels are then concatenated at the end, to form the final feature vector.

**Feature vectors:** Once the grids are fixed, the feature vectors are obtained by the concatenation of features extracted from each grid zone. Using a fixed grid solves the problem of matching uniformly scaled signatures; however embellishments such as those at the beginning or end of a signature may significantly vary in location, orientation and size, thereby significantly changing the global shape of a signature. Our expectation is that with the use of multiple registration points, at least some of the features will capture similarities between two signatures, even in the presence of such embellishments.

### 4.2. Histogram of Oriented Gradients

We use the Histogram of Oriented Gradients (HOG) features introduced by Dalal and Triggs [23]. The HOG features represent the gradient orientation relative to the dominant orientation. They have been used before in offline signature verification by Zhang [24].

While computing the gradient orientation histogram, circular shift normalization is done within the grid zone, to allow for rotational differences of the strokes. Specifically, after finding the gradient orientation at each point, we find the dominant gradient orientation and represent it at the first bin of the histogram.

HOG features are extracted both in Cartesian and polar coordinates, separately.

### 4.3. Local binary patterns

Local binary patterns (LBP) form a powerful feature vector that is proposed to capture texture in objects [25]. LBP is used in several works and found to be suitable for offline signature verification as well [12,26].

An important drawback of the original LBP method is the sparsity of the generated histogram; for example the size of the histogram for a 3 by 3 neighborhood is 256. More importantly, many of these patterns would never be seen on a small image sample. While there are many LBP variants proposed in the literature, there are few works for LBP pattern selection. An example LBP histogram selection is used in color texture classification by Porebski et al. [27]. It consists in assigning a score to each histogram bin, measuring its efficiency in characterizing the similarity of the textures within different classes.

There are also many works in literature to offer more compact histograms instead of pattern selection. In the work by Sujatha et al. [28], a special OR operator is implemented which takes the Boolean OR function of symmetric neighbor pairs, claiming to preserve more than 90% of information content while reducing the LBP code from 8 bits to 4 bits. Another work to compactly represent exponentially growing circular neighborhoods is presented by Mäenpää and Pietikäinen [29]. Large-scale texture patterns are detected by combining exponentially growing circular neighborhoods with Gaussian low-pass filtering. Then, cellular automata are proposed as a way of compactly encoding arbitrarily large circular neighborhoods.

Because of the exponential growth of the size of the histograms, it is not feasible to directly encode farther neighborhoods with closer neighborhoods. A novel way to jointly encode multiple scales is proposed by Qi et al. [30]. When each scale is encoded into histograms individually, the correlation between different scales is ignored and a lot of discriminative information is lost.

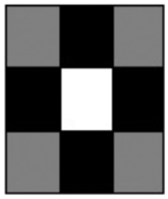


**Fig. 5.** Each 4-neighbor implicitly combines all combinations of diagonal neighbors in LBP-1 method.

The joint encoding strategy can capture the correlation between different scales and hence depict richer local structures. Reported results show about 7% accuracy improvement over baseline multi-scale LBP on texture recognition problems.

In another work in this direction, Zhang et al. offer a multi-block LBP method [31]. Inspired from Haar-like features [32], simple averaging in multiple rectangular blocks is applied to come up with 3 by 3 rectangular blocks of multiple pixels, each being treated like a single-pixel to calculate the conventional LBP code. This method is capable of taking farther neighborhoods into account, while avoiding the exponential growth in the resulting histogram.

In this work, LBP features are extracted only in Cartesian coordinates. We utilized different LBP pattern selection alternatives, as explained below.

*4.3.1. LBP-0*

We name the conventional LBP method for a 3x3 neighborhood (8-neighbors) as LBP-0. We extract LBP-0 features both globally (full image) and in finer grids in the Cartesian coordinates, in a coarse-to-fine approach. This is the baseline LBP extraction method that is used in subsequent LBP pattern selection alternatives given below.

*4.3.2. LBP-1*

LBP-0 results in a sparse feature vector since most of the patterns are never seen in a given grid zone. Also after the hierarchical grid placement, feature vector gets bigger, although the considered neighbors are just the 8 neighbors in a $3 \times 3$ neighborhood.

In the LBP-1 method, we make the system faster and concurrently improve the performance by separately considering the patterns formed by the 4-neighbors ({South, North, West, East}) and the diagonal neighbors ({North-East, North-West, South-East, South-West}), resulting in a feature vector of size $2 \times 2^4 = 32$. This circularly symmetric grouping is inspired by the work of Ojala et al. [33].

When computing the counts for the 4-neighbors' patterns, we implicitly combine the counts of all different combinations of diagonal-neighbors as *don't-care patterns*, and vice versa. This is illustrated in Fig. 5 where the gray pixels and all of their possible 16 combinations are combined into the histogram of each pattern formed by the 4-neighbors (the black pixels).

*4.3.3. LBP-2*

In this method we take all patterns in LBP-0 and select the best patterns explicitly. The selection criterion is based on the difference of the *term frequencies* ($\Delta TF$) of each pattern among the genuine and forgery samples. The aim is to select those patterns that occur more among the genuine signatures, as well as those that appear among the forgeries (e.g. patterns resulting from hesitation).

To do this, we first compute the histogram $H$ of all the LBP-0 codes over the whole image, separately for genuine signatures and skilled forgeries, using the training set (GPDS 161–300). Then we compute the $\Delta TF$ value for each LBP pattern $p$ and select the first

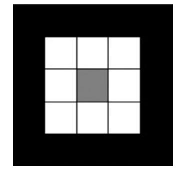
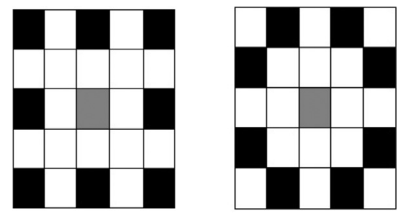


**Fig. 6.** Neighbors with Chebyshev distance of 2 are shown in black, center pixel shown in gray.



(a) (b)

**Fig. 7.** Neighbors with Chebyshev distance 2 are sampled in 2 groups, each group having 8 pixels.

32 patterns with the highest $|\Delta TF|$ values:

$$\Delta TF(p) = H_{Genuines}(p) - H_{Forgeries}(p). \tag{2}$$

**Alternative pattern selections:** To explore the effect of selecting the best patterns according to the delta term frequency criterion, we generate two other features: i) by selecting the *worst* 32 patterns with the smallest $|\Delta TF|$ values and ii) by selecting the next best 32 patterns, denoted as LBP-$2_{min}$ and LBP-$2_{n32}$, respectively.

*4.3.4. LBP-2F*

Detecting LBP patterns on a larger window than 3x3 can be useful, but the number of patterns grows exponentially with the size of the window. For instance, there are $2^{24}$ patterns to be considered in a 5x5 window. In order to take into account larger neighborhoods, we decided to consider only the borderline pixels of the considered window, in this variation. For instance in a 5x5 window, only the patterns constructed by pixels that are 2-Chebyshev distance to the center are considered, ignoring the variations in the 3x3 center, as shown in Fig. 6.

**Reducing the feature size:** Even only with the borderline pixels, there are $2^{16}$ different patterns for a 5x5 window, which is difficult to deal with in practice. In the literature, the generalized LBP operator is derived on the basis of a circularly symmetric neighbor set of a defined number of members, on a circle of radius $R$ that is designed to deal with this situation [33]. Neighbors in each of these circularly symmetric groups are completely independent and at the end all of the neighbors are covered by different groups, forming a basis. This LBP operator is previously applied to offline signature verification by Vargas et al. [12].

Similar to this work, we sample the pixels of 2-Chebyshev distance resulting in two groups of 8 pixels, as shown in Fig. 7 for a 5x5 window. Then, we select the best patterns for each group, as explained in Section 4.3.3. Pre-selected specific paths of Chebyshev distance 2 are also used in offline signature verification by Barkoula et al. [34].

**Alternative pattern selections:** To explore the effect of selecting the best patterns according to term frequency criterion, alternative pattern selections LBP-2$F_{min}$ and LBP-2$F_{n32}$ are again considered, as in Section 4.3.3.

**Multiple Chebyshev distances:** We repeat this process for Chebyshev distances of one, two and three, where there are one, two or three neighbor groups, respectively. Building individual classifiers for each neighbor group, we obtain a total of 6 classifiers, each one being an expert on completely independent information. We then use the average of these 6 classifiers' output, to obtain the final output for the LBP-2F method. Hence the LBP-2F method implicitly includes classifier combination.

### 4.4. Scale invariant feature transform

The Scale Invariant Feature Transform (SIFT) is a popular and successful feature extraction method used in computer vision for finding the correspondence between different views of an object or detecting and recognizing objects/scenes. The SIFT descriptors are extracted around local interest points and serve as distinctive, scale and rotation invariant features [35].

While SIFT features are used for offline signature verification previously [11,36], we use both the conventional SIFT matching approach and a novel one that is found to be more suitable for offline signature verification. In the second approach, we discretize the SIFT transformations and compute a feature vector in the form of a transformation histogram showing the most voted histograms by matching points. The motivation is to allow for different transformations in different parts of the signature, to allow for non-linear deformations.

#### 4.4.1. Conventional approach

In the conventional SIFT keypoint matching, a common rigid transformation can be found by voting for the most commonly occurring rigid transformation between matching point pairs. Example matches between two signature pairs are shown in Fig. 8, where corresponding matches of the most popular transformation bin are separately shown.

In our first approach, we discretize the SIFT transformations indicated by different matching points and analyze the number of votes in the most popular transformation for deciding whether the signature is genuine or forgery.

The transformation parameters between two matching points are found as follows. Assume that $x_1$, $y_1$ and $x_2$, $y_2$ are coordinates of two SIFT descriptors to be matched. To find the transformation, we first find the normalized coordinates $xn_1 = x_1/w_1, yn_1 = y_1/h_1, xn_2 = x_2/w_2, yn_2 = y_2/h_2$ where $w_i$ and $h_i$ are the width and height of image $i$. We find the translation in two dimensions using $xd = xn_1 - xn_2$ and $yd = yn_1 - yn_2$. We then find orientations of matches using $\theta = arctan((y_1 - y_2)/(x_1 - x_2))$. We quantize $\theta$ values into 8 bins, $xd$ values into 4 bins, $yd$ values into 4 bins; in total 128 bins.

Since the number of matches will be higher for longer and more complex signatures, we have to normalize the match counts so that longer signatures are not easily matched due to the sheer number of matches. We investigate two different normalization methods. In the first normalization method called *SIFT-MP*, we simply consider the ratio of matches in the most voted transformation ($N_h$) to the total number of matches. In the second normalization method called *SIFT-MR*, we normalize $N_h$ with the average number of matches among reference pairs ($N_h^R$). Thus, SIFT-MR considers the ratio of matched points in the most voted transform, to what is observed among reference pairs.

#### 4.4.2. Handling non-linear alignments

Signatures of a person often display large non-linear variations, especially with signatures having extensive embellishments. With these signatures, finding one global transformation to align them is not sufficient. To handle such situations, we developed a novel method which we refer as *SIFT-TH*. In this method, we use the number of votes for all the transformation bins as a feature vector, resulting in a feature vector of the same size as the number of the considered transformations (in our case $8 \times 4 \times 4 = 128$).

This novel representation is intended to address signatures where two parts of a signature may undergo different transformations. For instance for the genuine signatures of a person who signs his signature without any variability in the main body but a lot of variation in the embellishing stroke, the transformation histogram may show a consistent high match in one bin (no rotation and no translation) and a smaller match in one of the other bins.

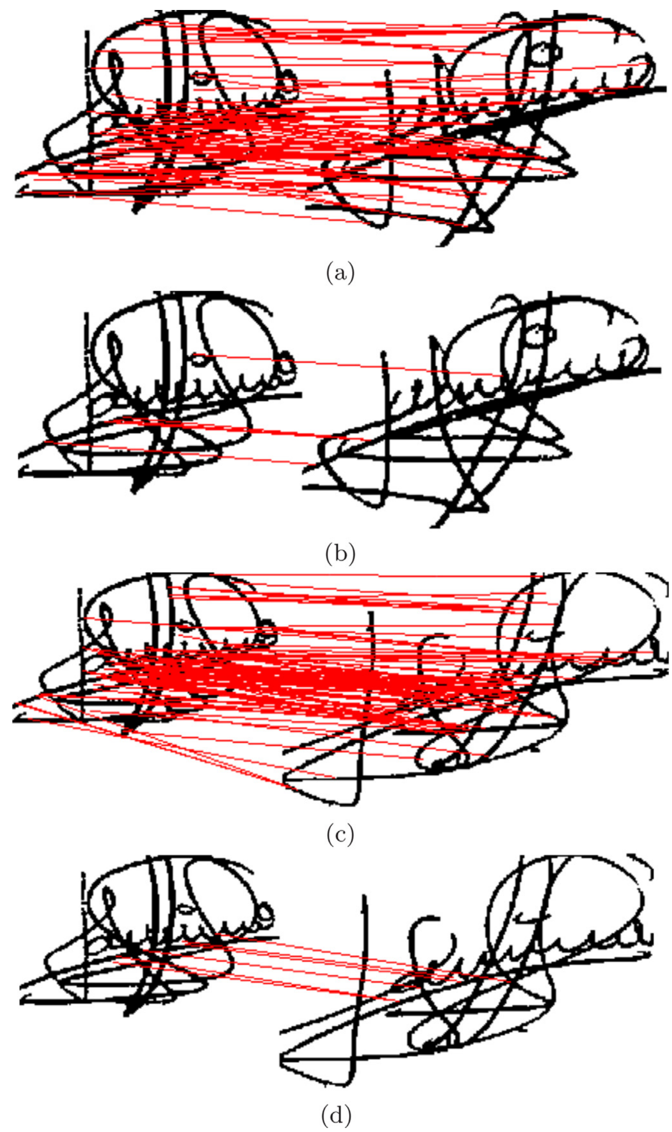



**Fig. 8.** Example SIFT matches in (a) and (c), with the corresponding most voted transformations in (b) and (d).

**Table 1**
SIFT results with different usages.

| Method | Rotation ($\theta$) bins | Translation (x,y) bins | EER |
|---|---|---|---|
| SIFT-MP | 8 | 4 | 29.12% |
| SIFT-MR | 8 | 4 | 25.84% |
| SIFT-TH | 8 | 4 | 24.09% |

Finally, we train USVM classifiers described in Section 5.2, where positive examples are collected by matching reference signature pairs and negative examples are collected by matching references to random forgeries.

For testing any of the three methods described above (SIFT-MP, SIFT-MR and SIFT-TH), we match the query to all the references of the user and use the median match score (normalized match counts or match scores) as the final SIFT score for the query.

Results given in Table 1 show the performance of three methods on the GPDS-160 dataset, using 5 genuine signatures as reference. As seen in these results, normalization based on reference set statistics (SIFT-MR) is the better normalization method, but the novel SIFT-TH approach outperforms the others.

## 5. Classification

One can use both user-based and global classifiers in offline signature verification. Because user-based classifiers are trained to discriminate just a single person against others, they are reported to be more successful [37], as long as there are enough references for each subject, to train the classifiers.

Combining user-dependent and global verification systems have been investigated before. For instance, Eskander et al. propose a hybrid offline signature verification system, consisting of writer-independent and writer-dependent classifiers that are used selectively, instead of concurrently [38].

In our system, classification is performed using SVMs, where two different approaches are investigated, namely global and user-dependent SVMs. The SVM classifier is found to be very successful in signature verification literature [13,20,26,39–42], in addition to reports of their good generalization abilities in general.

Both the user-dependent and global classifiers are trained with RBF kernels and parameters are optimized with grid search on a separate validation set (users 161–300 from the GPDS-300 dataset, who are not in the test set). The number of genuine signatures used as reference is set to 5 or 12, in-line with most previous research. For global SVMs (GSVMs), half of the users in the validation set is used for training and the other half is used for testing.

### 5.1. Global SVMs (GSVM)

A global (also called writer-independent or user-independent) signature verification system learns to differentiate between two types of classes: genuine and forgery. The global classifier (GSVM) in this work is a user-independent classifier that is trained to separate *difference vectors* obtained from genuine signatures of a user, from those obtained from (skilled) forgery signatures of the same user.

To obtain the difference vectors, features obtained from a query signature (genuine or forgery) are compared to the features obtained from each of the reference signatures of the claimed identity. The resulting difference vectors are then normalized so that each element of this vector represents how many standard deviations away the query feature is from the reference feature, using the standard deviation measured among the difference vectors of a given query and references.

More precisely, let $\{R^1, R^2, \ldots, R^N\}$ be the feature vectors extracted from the reference signatures of a particular user and let $Q = [q_1 \ldots q_M]$ be the feature vector extracted from a test signature, where $N$ is the number of reference signatures and $M$ is the number of features. Then, we compute $N$ difference vectors for each query, where the $i$th difference vector is computed as:

$$D^i = Q - R^i = \begin{bmatrix} (q_1 - R^i_1)/(\sigma_1 + \tau) \\ (q_2 - R^i_2)/(\sigma_2 + \tau) \\ \cdots \\ (q_M - R^i_M)/(\sigma_M + \tau) \end{bmatrix} \quad (3)$$

where $\sigma_m$ is the standard deviation of the $m$th dimension of the difference vectors between query and claimed user's reference signatures $i = 1,\ldots,N$. It is calculated as:

$$\sigma_m = \sqrt{\frac{1}{N}\sum_{i=1}^{N}((q_m - R^i_m) - \mu_m)^2} \quad (4)$$

where $\mu_m = \frac{1}{N}\sum_{i=1}^{N}(q_m - R^i_m)$ and $\tau$ is a small constant to eliminate division by zero. By the help of this normalization, the difference vector represents how many standard deviations away the query feature is from the reference feature. It is possible to evaluate a given query even with one reference of the claimed user, which is especially advantageous in real life applications. Because we have $N$ difference vectors $Q - R^i$ obtained from each reference, we have $N$ classifier scores for each query. To get a final classifier score, we calculate the average score.

Note here that the SVM is learning which differences in the feature vector may be within the normal variations of a user and which differences indicate forgeries. For instance, using global features such as size, pixel density, width-to-height ratio, the SVM learns how much variation in a particular feature matters. In the case of local features, the SVM can learn how to weight differences in the center compared to the periphery of the signature, for instance.

We devote users 161–300 from the GPDS-300 dataset for the GSVM training, such that users of train and test sets do not overlap. Actually the users devoted for training could be selected from a completely different database if appropriate image normalization is applied. In testing, references of test users are just used to calculate difference vectors; while their remaining signatures are used as queries.

### 5.2. User-dependent SVMs (USVM)

In the second approach, we train one user-dependent SVM per user, with the expectation that the user-dependent SVM can better learn to differentiate genuine signatures of a person from forgeries. Each SVM is trained with the raw feature vectors obtained from the reference signatures of the corresponding user and those obtained by random forgeries (other users' reference signatures reserved for training). Note that in this case, we do not need a separate group of users for training as opposed to GSVM, since we only use genuine signatures of others.

### 5.3. Classifier combination

In general, classifiers may differ by changing the training set, input features and parameters of the classifier. In many problems, score level combination of the classifiers using different representations is reported to perform better than feature level combination.

We combine the classifiers of the features introduced in Section 4 for user-dependent and user-independent (global) cases. Specifically, for a single query signature, there are 7 score outputs obtained: HOG-Cartesian USVM, HOG-Polar USVM, SIFT USVM, LBP-Cartesian USVM, HOG-Cartesian GSVM, HOG-Polar GSVM, LBP-Cartesian GSVM. A simple score level linear combination is used to obtain the final score where the weight set is found empirically from a validation set. We have only used well-known fusion method of averaging (with fixed and learned weight sets). However, the important contribution here is to show that we can obtain error rates that are 8–15% points lower compared to the best single feature classifier (using 5 references), either with USVMs or GSVM.

## 6. Experimental evaluation

### 6.1. Database and methodology

The GPDS-300, a publicly available subset of the GPDS-960 dataset [43] is used to train and evaluate the system performance. We perform evaluations on the GPDS-160 subset for testing in order to be compatible with most of the recent works, while the remaining 140 subjects are used for training (GSVMs).

In order to obtain results that are comparable to those reported in the literature, we train classifiers using 5 or 12 reference

**Table 2**
USVM results (EER) with different features.

| Features (Grid hierarchy) | 12 ref. | 5 ref. |
|---|---|---|
| SIFT-TH | 20.51% | 24.09% |
| HOG-Cartesian | 19.54% | 21.36% |
| HOG-Polar | 16.39% | 18.26% |
| LBP-0 (global) | 21.30% | 24.18% |
| LBP-0 | 15.32% | 17.54% |
| LBP-1 | 15.01% | 16.94% |
| LBP-2 | 15.10% | 17.43% |
| LBP-$2_{n32}$ | 15.78% | 18.16% |
| LBP-$2_{min}$ | 21.68% | 22.43% |
| LBP-2F | 11.30% | 12.77% |
| LBP-$2F_{n32}$ | 11.16% | 11.20% |
| LBP-$2F_{min}$ | 9.64% | 11.01% |
| LBP-2F (fusion) | 8.75% | 9.13% |

**Table 3**
GSVM results (EER) with different features.

| Features (Grid hierarchy) | 12 ref. | 5 ref. |
|---|---|---|
| HOG-Polar | 24.00% | 25.41% |
| HOG-Polar (aligned) | 22.35% | 23.87% |
| HOG-Cartesian | 20.83% | 26.13% |
| HOG-Cartesian (aligned) | 20.55% | 23.49% |
| LBP-2*F* (5x5, first group) | 30.25% | 30.32% |
| LBP-2*F* (5x5, first group, aligned) | 26.96% | 26.84% |

**Table 4**
Equal error rates for classifier combination.

| Method | 12 ref. | 5 ref. |
|---|---|---|
| All GSVMs, not aligned | 17.14% | 20.60% |
| All GSVMs, aligned | 18.32% | 20.88% |
| All USVMs (both HOGs, LBP-2F fusion, SIFT-TH) | 7.84% | 8.57% |
| All USVMs and all GSVMs | 7.32% | 8.30% |
| All USVMs and all GSVMs with precise weights | 6.97% | 7.98% |

**Table 5**
Effect of varying reference sets.

| Method | 5-fold EER |
|---|---|
| HOG-Cartesian GSVM, aligned | 21.38% ± 2.13% |
| HOG-Polar USVM | 16.33% ± 2.11% |
| LBP-2 USVM | 15.40% ± 2.29% |

signatures. However, we note that most real life applications involve the use of 5 or fewer reference signatures.

During testing, we use all genuine signatures of a user except those that are used as reference; thus resulting in 12 and 19 genuine tests per user for the cases of 12 and 5 reference signatures, respectively. Since we do not use any skilled forgeries of test users in training, all skilled forgeries of a user (30 of them) are used in testing.

We do not use any random forgeries during testing as they are generally much easier to detect and using them together with skilled forgeries basically amounts to averaging the performance obtained with skilled and random forgeries.

All results in Tables 2–6 are reported as EER results on skilled forgery tests, while we report DER results in Table 8 in order to compare our work with previous works that do not provide EER results.

### *6.2. Results*

The USVM results are given in Table 2, where all classifiers use a hierarchy of coarse-to-fine Cartesian grids, except for the LBP-0 global method where the features are obtained globally. Considering these results, we see that LBP features are better than HOG and SIFT-TH features. Furthermore, for farther neighborhood cases, the LBP-$2F_{min}$ performs the best; while counter-intuitive at first, this shows the discriminative power of rare LBP patterns exploited with the power of information fusion. We analyze the statistical significance of these results in Section 6.5.

**Table 6**
Comparison of score level fusion and feature level fusion results.

| Method | EER |
|---|---|
| HOG-Cartesian (1) | 21.36% |
| HOG-Polar (2) | 18.26% |
| LBP-2*F* (3x3) (3) | 17.43% |
| LBP-2*F* (5x5, first group) (4) | 13.47% |
| LBP-2*F* (5x5, second group) (5) | 13.76% |
| Feature fusion of (1) and (3) | 17.43% |
| Score fusion of (1) and (3) | 16.05% |
| Feature fusion of (2) and (3) | 18.32% |
| Score fusion of (2) and (3) | 14.42% |
| Feature fusion of (4) and (5) | 13.61% |
| Score fusion of (4) and (5), equal weights | 13.54% |

The GSVM results are given in Table 3. Considering these results, we see that GSVMs obtain higher accuracies with HOG features compared to LBP, in contrast to USVMs. However, the accuracy results are significantly lower compared to the best results obtained with USVMs. This is not very surprising as the USVMs are specifically trained for each user, while GSVMs only know about global (across all users) variations in each dimension. On the other hand, GSVM improves the overall performance slightly when used in conjunction with USVMs, as shown later.

The best grid method is not conclusive: the log-polar grid is better with USVMs while the Cartesian grid is better with the GSVM.

Finally, classifier combination results are provided in Table 4. As found in many studies in different fields, we also find that score-level combination of classifiers (using a weighted sum-rule) improves overall accuracy. Specifically, using the best combination with weights that are learnt from a separate validation set, we obtain very low equal error rates: 6.97% and 7.98% EER using 12 and 5 references, respectively. Precise weights are found by more sensitive weight learning such that the discrete intervals used for weight searching are kept smaller.

While GSVMs contribute slightly to the USVM combination, they can have a significant role in applications where each user only have a few reference signatures or when re-training the system is not possible.

The ROC curve for the best combined system is shown in Fig. 9 both for 5 references and 12 references. Log-scaled values for x-axis are used for all ROC curves for easier analysis.

### *6.3. Effect of varying reference sets*

As a sensitivity analysis of the selection of reference signatures, we ran 5-fold cross validation tests, with results given in Table 5. In each fold, we selected a different subset of 12 genuine signatures as reference and used the remaining ones as genuine test samples. In all our other experiments, the first *N* genuine signatures are chosen as the reference set, where *N* is the number of references.

The mean EER values reported with varying reference sets are close to the results given with the first 5 or 12 genuine signatures used as reference. Furthermore, the relative performances of the three methods remain unchanged. On the other hand, there is a relatively high standard deviation, indicating that not only the number, but also the selection of reference signatures matters in overall accuracy.

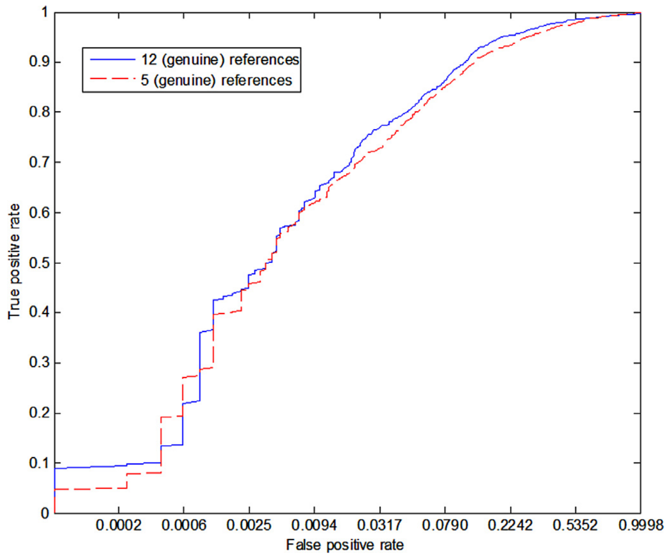



**Fig. 9.** ROC curve for the best system given in Table 4.

**Table 7**
Confidence intervals for some of the main methods.

| Method | Confidence intervals |
|---|---|
| LBP-0 | 15.63 ± 5.00% |
| LBP-1 | 15.00 ± 5.62% |
| LBP-2 | 15.31 ± 5.00% |
| LBP-2F$_{min}$ | 9.69 ± 4.38% |
| LBP-2F (fusion) | 9.07 ± 4.07% |
| All GSVMs (not aligned) | 17.50 ± 5.63% |
| All GSVMs (aligned) | 18.44 ± 5.94% |
| All USVMs | 7.81 ± 4.06% |
| All USVMs and all GSVMs (coarse weights) | 7.35 ± 3.91% |
| All USVMs and all GSVMs (precise weights) | 6.88 ± 3.75% |
| Feature fusion (2) and (3) | 17.50 ± 5.63% |
| Score fusion (2) and (3) | 14.07 ± 5.31% |

### *6.4. Comparison with feature level combination*

Another common choice for information fusion is feature level combination. We have compared feature level fusion with score level fusion, for several pairs of features and corresponding classifiers, with results given in Table 6. The results are obtained with 5 reference signatures on the GPDS-160 dataset and the weights for score level fusion are learnt from a separate set (GPDS 161–300). ROC curves are provided in Fig. 10 for detailed analysis of this comparison.

We observe that score level fusion outperforms feature level fusion. Furthermore, besides the performance advantage, score level fusion is easier to implement, requiring only a simple optimization of the weights and can be parallelized.

### *6.5. Statistical analysis*

We report 95% confidence intervals for the main methods considered in this work, in Table 7. To obtain confidence intervals, we first use a Monte Carlo simulation of the balanced repeated replicates method of Micheals and Boult [44], as described in [45]. This method samples the result of a single query from each user in one trial and computes the confidence interval of the equal error rate observed in 1000 such trials, similar to [46].

In addition, we performed statistical significance tests by considering the results of paired trials that were used to obtain the confidence intervals. These analyses indicate that LBP-2F (fusion) method is significantly better compared to LBP-0 and LBP-1 meth-

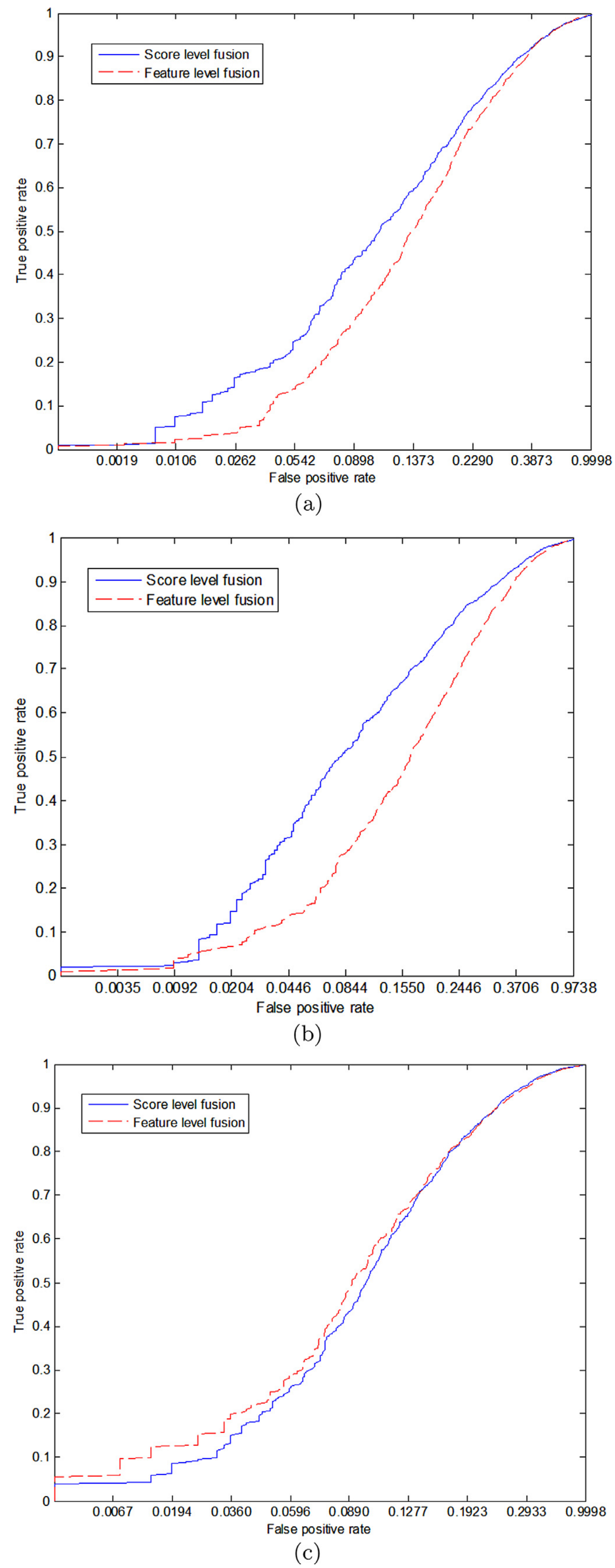



**Fig. 10.** ROC curves for score and feature level combinations: a) fusion of methods (1) and (3); b) fusion of (2) and (3); c) fusion of (4) and (5), defined in Table 6.

**Table 8**
Summary of recent results on GPDS dataset.

| Reference | Method | GPDS Set | Training | Testing | DER |
|---|---|---|---|---|---|
| Vargas et. al. [49] | Wavelets | GPDS-100 gray level | 5 gen. + rand. forg. | 19 gen. and 24 skl. forg. | 14.22% |
| Vargas et. al. [12] | Texture features | GPDS-100 gray level | 10 gen. + rand. forg. | 14 gen. and 24 skl. forg. | 9.02% |
| Hu and Chen [20] | Writer indep., Adaboost | $GPDS_{rand} - 100$ gray level | 10 gen. + rand. forg. | 14 gen. + 30 skl. forg. | 9.94% |
| Hu and Chen [20] | Writer dependent, SVM | $GPDS_{rand} - 150$ gray level | 10 gen. + rand. forg. | 14 gen. + 30 skl. forg. | 7.66% |
| Parodi et. al. [50] | Graphomt. features | $GPDS_{rand} - 130$ | 13 gen. + rand. forg. | 11 gen., 24 skl. forg., and simple forg. (not detailed) | 4.21% |
| Bharathi and Shekar [42] | Chain code histogram | GPDS-100 | 12 gen. + rand. forg. | 12 gen. + 30 skl. forg. | 11.4% |
| Nguyen et. al. [40] | SVM | GPDS-160 | 12 gen. + rand. forg. | 12 gen. + 30 skl. forg. | 20.07% |
| Nguyen et. al. [51] | Global features | GPDS-160 | 12 gen. + rand. forg. | 12 gen. + 30 skl. forg. | 17.25% |
| Batista et. al. [19] | EoCs | GPDS-160 | 12 gen. + rand. forg. | 12 gen. + 30 skl. forg. | 16.81% |
| Eskander et. al. [38] | Writer indep. classifier | GPDS-160 | 12 gen. + rand. forg. | 12 gen. + 30 skl. forg. | 26.73% |
| Eskander et. al. [38] | Writer dependent classifier | GPDS-160 | 12 gen. + rand. forg. | 12 gen. + 30 skl. forg. | 22.71% |
| Proposed | SVM | GPDS-160 | 5 gen. + rand. forg. | 19 gen. + 30 skl. forg. | 7.98% |
| Proposed | SVM | GPDS-160 | 12 gen. + rand. forg. | 12 gen. + 30 skl. forg. | 6.97% |

ods; and the USVMs are significantly better compared to GSVMs ($p \leq .05$, two-tailed).

When we consider 90% confidence intervals, the LBP-2F$_{min}$ results are also found to be significantly better compared to LBP-0 and LBP-1 results ($p \leq .10$, two-tailed).

### 6.6. Comparison with state-of-the-art results

For evaluation of our results, we give recent results on the GPDS database in Table 8. We note that most of these results are not directly comparable, as performance depend heavily on several factors: the used database (GPDS-100, GPDS-160, or some random subset of the full database denoted with subscript *Rand*); the number of reference signatures used in training (more references normally help with verification); the use of skilled forgeries in training (some systems use none, while others may use a varying number); and the image type (binary or gray-level). Nonetheless, we give results from the literature for context.

This comparison set includes systems that use skilled forgeries in training [13,14,47], but we do not use any skilled forgeries in training as the use of skilled forgeries is not really suitable for real life applications. Also some systems use smaller subsets of the GPDS database [12,20,42,47–49], while we use the whole GPDS-160 database. Finally, some systems utilize the gray-level version of the database and benefit from the richer information present in the gray-level image [12,14,20,47,49], but our work is done on the binary version of the database.

In summary, our experimental evaluation protocol considers the binary GPDS-160 with 12 genuine signatures as reference set and without skilled forgeries in training. The best previous result with this configuration is reported as 16.81% DER [19]. Our fusion results that are directly comparable to this system are 6.97% and 7.98% EERs with 12 and 5 genuine references, respectively.

## 7. Summary and conclusions

We present a state-of-the-art automatic offline signature verification system based on HOG and LBP features extracted from local grid zones. For either of the representations, features obtained from grid zones are concatenated in a coarse-to-fine hierarchy to form the final feature vector. Two different types of SVM classifiers are trained to perform verification, namely global and user-dependent SVMs. We also evaluate the fusion of classifiers and show that fusion improves the overall verification performance and that score level fusion outperforms feature level fusion.

Using the definitions in [21], our system can be defined as an adapted classification, global fusion and global decision system. It is experimentally shown that when enough training data (at least 5 genuine signatures and many random forgeries as the reference set) is available, user-based classifiers are much more successful, as previously observed in literature, but user-independent classifiers complement them to improve performance.

Obtained results are in par or better compared to those reported in the literature for the GPDS database without using any skilled forgeries in training.

## 8. Future work

While state-of-the-art in offline signature verification achieves around 10–15% EER in various databases, the performance of these systems would be expected to be significantly worse with signatures collected in real life scenarios. In the future, systems research needs to concentrate on increasing the robustness of systems towards larger variations encountered in real life (e.g. signatures signed in smaller spaces, or in a hurry, or on documents with interfering lines).

Another issue is to allow the system work well with less number of references, such as three as is the case in many banking operations or even with one reference. Importance of user-based score normalization becomes significant with such extreme cases. Developing a simpler and better score normalization method is a part of our future work.

Measuring the complexity level of a signature can also help with many issues such as user-based score normalization or enforcing the strength of the signature.

## References

[1] R. Plamondon, G. Lorette, Automatic signature verification and writer identification – the state of the art, Pattern Recognit. 22 (2) (1989) 107–131.
[2] R. Sabourin, R. Plamondon, G. Lorette, Off-line identification with handwritten signature images: survey and perspectives, Struct. Doc. Image Anal. (1992) 219–234.
[3] F. Leclerc, R. Plamondon, Automatic signature verification: the state of the art, 1989–1993, Int. J. Pattern Recognit. Artif. Intell. 8 (3) (1994) 643–660.
[4] R. Plamondon, S.N. Srihari, On-line and off-line handwriting recognition: a comprehensive survey, IEEE Trans. Pattern Anal. Mach. Intell. 22 (1) (2000) 63–84.
[5] J. Coetzer, Off-line signature verification, University of Stellenbosch, South Africa, 2005 Ph.D. thesis.
[6] D. Impedovo, G. Pirlo, Automatic signature verification: the state of the art, Trans. Syst. Man Cybern. Part C 38 (5) (2008) 609–635.
[7] M.S. Arya, V.S. Inamdar, A preliminary study on various off-line hand written signature verification approaches, Int. J. Comput. Appl. 1 (9) (2010) 55–60.Published By Foundation of Computer Science.
[8] S. Pal, M. Blumenstein, U. Pal, Off-line signature verification systems: a survey, in: Proceedings of the International Conference & Workshop on Emerging Trends in Technology, in: ICWET '11, ACM, New York, NY, USA, 2011, pp. 652–657.
[9] V. Bhosale, A. Karwankar, Automatic static signature verification systems: a review, Int. J. Comput. Eng. Res. 3 (2) (2013) 8–12.
[10] D. Impedovo, G. Pirlo, M. Russo, Recent advances in offline signature identification, in: Frontiers in Handwriting Recognition (ICFHR), 2014 14th International Conference on, IEEE, 2014, pp. 639–642.

[11] J. Ruiz-Del-Solar, C. Devia, P. Loncomilla, F. Concha, Offline signature verification using local interest points and descriptors, in: CIARP '08: Proceedings of the 13th Iberoamerican congress on Pattern Recognition, Springer-Verlag, Berlin, Heidelberg, 2008, pp. 22–29.
[12] J.F. Vargas, M.A. Ferrer, C.M. Travieso, J.B. Alonso, Off-line signature verification based on grey level information using texture features, Pattern Recognit. 44 (2011) 375–385.
[13] M.A. Ferrer, J.B. Alonso, C.M. Travieso, Offline geometric parameters for automatic signature verification using fixed-point arithmetic, IEEE Trans. Pattern Anal. Mach. Intell. 27 (6) (2005) 993–997.
[14] F.J. Vargas, M.A. Ferrer, C.M. Travieso, J.B. Alonso, Off-line signature verification based on high pressure polar distribution, in: Proceedings of the 11th International Conference on Frontiers in Handwriting Recognition, ICFHR 2008, 2008, pp. 373–378.
[15] J. Fierrez-Aguilar, N. Alonso-Hermira, G. Moreno-Marquez, J. Ortega-Garcia, An off-line signature verification system based on fusion of local and global information, in: D. Maltoni, A. Jain (Eds.), Biometric Authentication, Lecture Notes in Computer Science, vol. 3087, Springer Berlin Heidelberg, 2004, pp. 295–306.
[16] L.S. Oliveira, E. Justino, R. Sabourin, F. Bortolozzi, Combining classifiers in the ROC-space for off-line signature verification, J. Univers. Comput. Sci. 14 (2) (2008) 237–251.
[17] D. Bertolini, L.S. Oliveira, E. Justino, R. Sabourin, Reducing forgeries in writer-independent off-line signature verification through ensemble of classifiers, Pattern Recognit. 43 (2010) 387–396.
[18] H.N. Prakash, D.S. Guru, Offline signature verification: an approach based on score level fusion, Int. J. Comput. Appl. 1 (18) (2010) 52–58.
[19] L. Batista, E. Granger, R. Sabourin, Dynamic selection of generative-discriminative ensembles for off-line signature verification, Pattern Recognit. 45 (4) (2012) 1326–1340.
[20] J. Hu, Y. Chen, Offline signature verification using real adaboost classifier combination of pseudo-dynamic features, in: Document Analysis and Recognition (ICDAR), 2013 12th International Conference on, 2013, pp. 1345–1349.
[21] J. Fierrez-Aguilar, D. Garcia-Romero, J. Ortega-Garcia, J. Gonzalez-Rodriguez, Adapted user-dependent multimodal biometric authentication exploiting general information, Pattern Recognit. Lett. 26 (2005) 2628–2639.
[22] E. Shechtman, M. Irani, Matching local self-similarities across images and videos, in: Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on, IEEE, 2007, pp. 1–8.
[23] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1 - Volume 01, in: CVPR '05, IEEE Computer Society, Washington, DC, USA, 2005, pp. 886–893.
[24] B. Zhang, Off-line signature verification and identification by pyramid histogram of oriented gradients, Int. J. Intell. Comput. Cybern. 3 (2010) 611–630.
[25] T. Ojala, M. Pietikainen, D. Harwood, Performance evaluation of texture measures with classification based on Kullback discrimination of distributions, 1994, pp. A:582–585.
[26] R. Wajid, A. Bin Mansoor, Classifier performance evaluation for offline signature verification using local binary patterns, in: Visual Information Processing (EUVIP), 2013 4th European Workshop on, 2013, pp. 250–254.
[27] A. Porebski, N. Vandenbroucke, D. Hamad, LBP histogram selection for supervised color texture classification, in: Image Processing (ICIP), 2013 20th IEEE International Conference on, 2013, pp. 3239–3243.
[28] B. Sujatha, V.V. Kumar, P. Harini, A new logical compact LBP co-occurrence matrix for texture analysis, Int. J. Sci. Eng. Res. 3 (2) (2012) 1–5.
[29] T. Mäenpää, M. Pietikäinen, Multi-scale binary patterns for texture analysis, in: J. Bigun, T. Gustavsson (Eds.), Image Analysis, Lecture Notes in Computer Science, vol. 2749, Springer Berlin Heidelberg, 2003, pp. 885–892.
[30] X. Qi, Y. Qiao, C.-G. Li, J. Guo, Multi-scale joint encoding of local binary patterns for texture and material classification, in: British Machine Vision Conference (BMVC), 2013, pp. 1–11.
[31] L. Zhang, R. Chu, S. Xiang, S. Liao, S. Li, Face detection based on multi-block LBP representation, in: S.-W. Lee, S. Li (Eds.), Advances in Biometrics, Lecture Notes in Computer Science, vol. 4642, Springer Berlin Heidelberg, 2007, pp. 11–18.
[32] P. Viola, M. Jones, Rapid object detection using a boosted cascade of simple features, in: Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on, vol. 1, 2001, pp. I–511–I–518.
[33] T. Ojala, M. Pietikäinen, T. Mäenpää, Multiresolution gray-scale and rotation invariant texture classification with local binary patterns, IEEE Trans. Pattern Anal. Mach. Intell. 24 (7) (2002) 971–987.
[34] K. Barkoula, E.N. Zois, E. Zervas, G. Economou, Off-line signature verification based on ordered grid features: An evaluation., in: M.I. Malik, M. Liwicki, L. Alewijnse, M. Blumenstein, C. Berger, R. Stoel, B. Found (Eds.), AFHA, CEUR Workshop Proceedings, vol. 1022, CEUR-WS.org, 2013, pp. 36–40.
[35] D.G. Lowe, Distinctive image features from scale-invariant keypoints, Int. J. Comput. Vis. 60 (2) (2004) 91–110.
[36] K.E. Mwangi, Offline Handwritten Signature Verification Using SIFT Features, Makerere University, Uganda, 2008 Master's thesis.
[37] M.B. Yılmaz, B. Yanıkoğlu, Ç. Tırkaz, A.A. Kholmatov, Offline signature verification using classifier combination of HOG and LBP features, in: Proc. of the 2011 Intl. Joint Conf. on Biometrics (IJCB 2011), in: IJCB '11, 2011, pp. 1–7.Washington, DC, USA
[38] G.S. Eskander, R. Sabourin, E. Granger, Hybrid writer-independent writer-dependent offline signature verification system, IET Biometrics 2 (12) (2013) 169–181.
[39] E.J.R. Justino, F. Bortolozzi, R. Sabourin, A comparison of SVM and HMM classifiers in the off-line signature verification, Pattern Recognit. Lett. 26 (9) (2005) 1377–1385.
[40] V. Nguyen, M. Blumenstein, V. Muthukkumarasamy, G. Leedham, Off-line signature verification using enhanced modified direction features in conjunction with neural classifiers and support vector machines, in: ICDAR '07: Proceedings of the Ninth International Conference on Document Analysis and Recognition, IEEE Computer Society, Washington, DC, USA, 2007, pp. 734–738.
[41] Y. Guerbai, Y. Chibani, N. Abbas, One-class versus bi-class SVM classifier for off-line signature verification, in: Multimedia Computing and Systems (ICMCS), 2012 International Conference on, 2012, pp. 206–210.
[42] R. Bharathi, B. Shekar, Off-line signature verification based on chain code histogram and support vector machine, in: Advances in Computing, Communications and Informatics (ICACCI), 2013 International Conference on, 2013, pp. 2063–2068.
[43] F. Vargas, M.A. Ferrer, C.M. Travieso, J.B. Alonso, Off-line handwritten signature GPDS-960 corpus, in: IAPR 9'th International Conference on Document Analysis and Recognition, 2007, pp. 764–768.
[44] R. Micheals, T. Boult, Efficient evaluation of classification and recognition systems, in: Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on, vol. 1, 2001, pp. I–50–I–57vol.1.
[45] M.E. Schuckers, A. Hawley, K. Livingstone, N. Mramba, A comparison of statistical methods for evaluating matching performance of a biometric identification device: a preliminary report, in: A.K. Jain, N.K. Ratha (Eds.), Biometric Technology for Human Identification, Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, vol. 5404, 2004, pp. 144–155.
[46] C. Shen, Z. Cai, X. Guan, Y. Du, R. Maxion, User authentication through mouse dynamics, Inf. Forensics Secur. IEEE Trans. 8 (1) (2013) 16–30.
[47] J.F.V. Bonilla, M.A.F. Ballester, C.M.T. Gonzalez, J.B.A. Hernandez, Offline signature verification based on pseudo-cepstral coefficients, in: Proceedings of the 2009 10th International Conference on Document Analysis and Recognition, in: ICDAR '09, IEEE Computer Society, Washington, DC, USA, 2009, pp. 126–130.
[48] R. Larkins, M. Mayo, Adaptive feature thresholding for off-line signature verification, in: IVCNZ '08: Proceedings of the 23rd International Conference In Image and Vision Computing New Zealand, 2008, pp. 1–6.
[49] J. Vargas, C. Travieso, J. Alonso, M.A. Ferrer, Off-line signature verification based on gray level information using wavelet transform and texture features, in: Frontiers in Handwriting Recognition (ICFHR), 2010 International Conference on, 2010, pp. 587–592.
[50] M. Parodi, J.C. Gomez, A. Belaid, A circular grid-based rotation invariant feature extraction approach for off-line signature verification., in: Document Analysis and Recognition (ICDAR), 2011 International Conference on, IEEE, 2011, pp. 1289–1293.
[51] V. Nguyen, M. Blumenstein, G. Leedham, Global features for the off-line signature verification problem, in: Proceedings of the 2009 10th International Conference on Document Analysis and Recognition, in: ICDAR '09, IEEE Computer Society, Washington, DC, USA, 2009, pp. 1300–1304.