A comprehensive study of sparse representation techniques for offline signature verification

E. N. Zois, D. Tsource, I. Theodorakopoulos, A. L. Kesidis and G. Economou

Abstract— In this work, a method for offline signature verification is presented that harnesses the power of sparse representation in order to deliver state-of-the-art verification performance in several signature datasets like CEDAR, MCYT-75, GPDS and UTSIG. Beyond the accuracy improvements, several major parameters associated with sparse representation; such as selected formulation, dictionary size, sparsity level and positivity priors are investigated. Besides, it is evinced that 2nd order statistics of the sparse codes is a powerful pooling function for the formation of the global signature descriptor. Also, a thorough evaluation of the effects of preprocessing is introduced by an automated algorithm in order to select the optimum thinning level. Finally, a segmentation strategy which employs a special form of spatial pyramid tailored to the problem of sparse representation is presented along with the enhancing of the produced descriptor on meaningful areas of the signature as emerged from the BRISK key-point detection mechanism. The obtained state-of-the-art results on the most challenging signature datasets provide a strong indication towards the benefits of learned features, even in WD scenarios with only a few available reference samples.

Index Terms— Off-line Signature Verification, Dictionary Learning, Sparse Coding, Spatial Pyramid, Feature Pooling, Image Preprocessing

1 INTRODUCTION

One of the most acceptable and long-standing behavioral modus to declare and verify a person’s identity or acknowledge of his/her consent in a wide variety of cases is the handwritten signature. Signatures are depicted by their trace, usually onto a sheet of paper or an electronic device. This process conveys information related not only to the imprinting of the personal details of the signatory, but also to his/her writing system and psychophysical state [1]. Ongoing research regarding the development of offline or static Automated Signature Verification Systems (ASV’s/SV’s) indicate clearly that this topic still is an active, open and important research field [2-5].

The purpose of an offline SV system is to recognize an image of a signature in question as genuine or forgery, i.e. to verify the writer’s genuine signatures and reject the forgery ones. This work addresses writer dependent signature verification (WD) by employing a unique model for each participating writer with the use of some genuine reference samples and some genuine samples from other writers as representatives of the positive and negative classes, respectively.

Conceivably, the most influential step in the design of a SV system is the feature extraction stage, which can be defined as the process that maps any given signature image into a multidimensional vector. Feature extraction methods can be divided into two major categories [2]: a) handcrafted features which aim at pre-determined characteristics of the image; examples of this branch include methods with global-local and/or grid-texture oriented features [6-8] and b) features learned directly from images, where Deep Learning (DL) [2], [9-10], Bag of Visual Words (BoW) [11-13] or Histogram of Templates (HOT) [14] are some of the most representative techniques.

Learning features from images could potentially have significant advantages, since such techniques -in general- can discover specialized spatial associations that are inherent into the signature images. Among the most powerful categories of such methods is Sparse Representation (SR) which aims to provide a parsimonious representation of a signal by means of a linear combination of only a few atoms, which are members of an overcomplete set or dictionary-lexicon. SR techniques are the subject of scientific interest for quite a long time [15-16] and have been proved to be extremely useful in computer vision and pattern recognition applications.

This work presents for the first time a method for off-line signature verification that harnesses the power of SR in order to deliver state-of-the-art verification performance with the use of few genuine reference samples. The current paper introduces several advancements in the line of work initiated with two recently published papers investigating the potential of SR [6] and Archetypal Analysis (AA) [7] on the offline SV task. Beyond the accuracy improvements and the high-end performance achieved in several popular signature datasets, this work has a significant contribution towards the following directions:

1. We innovatively address several aspects of local feature pooling for offline SV. As a result, we provide evidence of the superiority of 2nd order statistics of the sparse coefficients as a powerful pooling function for the formation of the global signature descriptor. Combined with a segmentation strategy which employs a special form of spatial pyramid (SP) tailored to the problem of SV we provide a powerful feature extraction mechanism for SV.

2. We apply a standard mechanism for enhancing the verification efficiency of the produced descriptor on meaningful areas of the signature, by means of the key-point detection mechanism of BRISK descriptors.

3. We perform a thorough evaluation of the effects of preprocessing, especially thinning and introduce an automated algorithm to select the optimum thinning level in order to derive a signature’s trace on the image plane.

4. We thoroughly investigate the impact of all the major parameters associated with SR, such as selected formulation (greedy approximation or convex relaxation), dictionary size, sparsity level etc. We also evaluate the effect to the overall performance of imposing additional priors into the corresponding optimization problems.

The paper is organized as follows: Section 2 provides a short literature review and summarizes the proposed approach. Sec-
tion 3 discusses details of the signature preprocessing steps and highlights the feature formation process while Section 4 describes the utilized databases along with the evaluation protocol. Finally, conclusions are drawn in Section 6. Appendices A, B provide elementary details regarding sparse representation and dictionary learning techniques.

2 LITERATURE REVIEW - SUMMARY OF THE METHOD

2.1 Related Work

A significant number of handcrafted feature extraction methods for offline signature verification rely on the evaluation of global and/or local signature descriptors as well as on grid-texture oriented features. With respect to the above family of feature descriptors, a diversity of feature extraction methods for offline SV has been proposed. Some examples can be found in [6-7] and their associated references. On the other hand, methods have been proposed that rely on learning features directly from the raw image data. Some efforts include the use of Restricted Boltzmann Machines (RBMs) in [17] and Convolutional Neural Networks (CNNs) [18-19]. Soleimani et al. in [9] proposed the use of Deep Neural Networks for Multitask Metric Learning by employing a distance metric between pairs of signatures in which Local Binary Patterns is used as an intermediate feature vector. Rantzsch et al. [20] presented an approach named Signature Embedding which is based on deep metric learning. Specifically they compare triplets of two genuine and one forged signature, in order for their system to learn to embed signatures into a high-dimensional space. Following, they proposed a Euclidean distance metric as a means for measuring similarity. Hafemann et al. in a series of publications, proposed methods for learning features from images. Specifically, the authors in [21] introduced a formulation for learning features from genuine signatures by a development dataset, and used them in order to train writer dependent classifiers to another set of users. In [22] the authors obtained state-of-the-art results on several GPDS datasets using CNN architecture and in [23] they demonstrated a novel formulation that leverages knowledge of skilled forgeries for feature learning.

Although SR methods have not been exploited for feature extraction in offline SV systems prior to [6] and [7], in [24] a method for writer identification based on sparse representation of handwritten structural primitives, called graphemes or fraglets is presented. Similarly, in [25] appears an online signature verification technique based on discrete cosine transform (DCT) and sparse representation. In addition, some sort of codebook formation is also proposed in [26] by means of forming codebooks using signature samples of an independent database with the k-means algorithm and in [13] by creating a codebook of first order HOGs and then coding each feature to the nearest word in the codebook. Finally, as it said "whenever using k-means to get a dictionary, if you replace it with sparse coding it’ll often work better" [27], and therefore our intuition to use SR is got stronger.

2.2 Method Overview

The proposed handwritten signature verification system utilizes a sparse representation framework in order to learn local features and construct a global signature descriptor. The two main approximations of sparse coding (greedy and convex relaxation) and their efficiency are thoroughly investigated; along with multiple pooling strategies, including a novel pooling function based on second order statistics. In addition, a widespread key-point selection algorithm is also employed in order to further emphasize salient locations across the signature.

The preprocessing stage of grayscale signature images consists of thresholding, followed by the morphological process of thinning. The Optimal Thinning Level (OTL) at each signature is defined by patches from the binary/thinned images. Patches are extracted from every pixel of the thinned signature’s trace while the percentage of the signature pixels that inhabit each patch is also taken into account. This information helps to define the patch density (PD) of a signature and plot it as a function of the number of successive thinning operations applied. In this way, the individual optimal thinning level (OTL) for each signature can be defined by utilizing the minimum value of its corresponding PD slope. Following the completion of this procedure for all reference samples, their median value defines the writer optimal thinning level (MOTL) which is used for the preprocessing of all the signature images corresponding to the writer under consideration.

For SR, the \( \{x^i\} \) patches are extracted from the grayscale signature image, at every \( j \)-position indicated by the skeleton of the signature, obtained via the thinning process. Subsequently, the gray values of the patches associated with the genuine reference samples are transformed into column vectors and used as input \( X = \{x^i\} \) to a dictionary learning algorithm which evaluates the dictionary \( D \). Following, for every other signature, the patches are encoded, by means of evaluating the sparse coefficients \( A = \{a^i\} \). The final feature of each signature image is formed by applying a pooling function \( F(A) \) on the sparse vectors, using a specially designed spatial pyramid, which segments the signature skeleton in equimass parts. Additionally, the key-points derived from the BRISK [28] algorithm pinpoint image regions of interest, whereas the sparse codes of the corresponding nearby patches are pooled together. The obtained pooled vector is concatenated with the spatial pyra-

![Fig. 1. A diagram of the proposed pipeline emphasizing the preprocessing and dictionary learning stages. The Optimal Thinning Level (OTL) at the preprocessing stage is selected from the extreme point of the patch density slope (located at the upper center of the Figure).](image-url)
mid vector in order to form the final signature descriptor. Finally, the signature verification system is formed by a binary radial basis SVM classifier, which takes the above features as inputs and tries to discriminate the genuine signatures from forgeries. In the training of the classifier, the positive class is composed from the reference genuine signatures of the writer and the negative class from random forgery signatures, randomly sampled from a few of the remaining writers without knowledge of the ownership. Figure 1 presents a motif of the proposed system with emphasis on the preprocessing and the dictionary learning stages.

3. Feature Extraction

3.1. Preprocessing

Preprocessing consists of two steps: binarization followed by thinning. The grayscale images are binarized using Otsu’s method [29] while for thinning, successive morphological operations are applied on the binary image in order to provide a gradual skeletonization of the signature. Intuitively, the outcome of the thinning operation affects the verification performance since it modifies the shape of the signature image. It has been experimentally observed [8] that the thinning level is critical for a Signature Verification (SV) system’s performance and its optimal value is not the same for all databases. This work proposes a novel method for selecting the optimal thinning level (OTL) for each signature and consequently for each writer. As mentioned in section 2.2, the OTL is defined to be the number of thinning operations that results to the steepest descend of the density function. Following the enrollment of a set of genuine reference signatures for a person, its median value of the associated OTL values (MOTL) accompanies the design of each writer’s model, i.e. 

\[ \text{MOTL}(N_{G-i}) = \text{median}(OTL(i)), \text{ where } i \in [1, ..., N_{G-i}] \]

Hence, for any input signature which claims an identity, the number of thinning operations will be determined by the MOTL value of the signing person. Figure 2 presents the corresponding plots of the patch density, the patch density derivative, expressed by its associated patch density difference and OTL-MOTL values for one writer and an indicative number of his/hers genuine samples derived from all the signature datasets namely CEDAR, MCYT-75, GPDS300 and UTSIG.

The extracted results indicate that the genuine signatures that are part of the CEDAR dataset have the majority (~95%) of their OTL values to be equal to one with few (~5%) signature samples having their OTL values to two. For the MCYT-75 dataset the majority of their OTL values are equal to two with very few samples having their OTL values to three and four. For the GPDS300 dataset, the OTL values appear to distribute between 2 and 5, with an observed mode of 4, while for the UTSIG dataset OTL is equally distributed among values 2 and 3.

Especially for the CEDAR and MCYT-75 datasets, Figure 2 indicates that they are more stable in terms of acquisition conditions comparing to datasets GPDS300 and UTSIG. Thus one could consider the SV problem addressed by CEDAR and MCYT-75 dataset to resemble a case study in which the aspect ratio and acquisition conditions do not vary significantly, similar to situations encountered in mobile banking applications.

3.2. Patch Extraction

The signature patches are extracted from the original grayscale signature image and are designated by the signature’s skeleton after the thinning operation. Specifically, the patches’ centers are located at every pixel of the signature skeleton, thus the number of image patches equals the number of pixels in the signature’s skeleton. Moreover, the patches are centered, i.e. have their average intensity been subtracted in order to have a zero mean value. The centering of each patch produces data invariant to the mean intensity and the learned structures, like edges, are anticipated to have zero mean as well. In all the conducted experiments the patch size is set to five i.e. \( n=25 \).

Our main rationale behind this selection is to keep the complexity of the local manifold of patches reasonably low. The reason is that sparse codes of data lying on smoother and more uniform manifolds tend to create more evenly distributed coefficients along the codebook’s elements. Such cases are less prone to the phenomenon where very few dictionary elements are over-represented in the resulting sparse codes and are responsible for a significant amount of the total energy, thus predominantly shaping the distance between global image descriptors formed by these codes and requiring special pooling strategies [30] to restore the discriminative power.

With this aim, it is valuable to consider the parameters which affect the dimensionality and shape of the underlying local manifolds. In [31] Peyré shows that the local manifold of patches from cartoon images (images that contain sharp variations along regular curves) can be parameterized by two variables, leading to a manifold topologically equivalent to the surface of a cylinder in 3D space. This parameterization holds as long as the signal within each patch can be approximated by two regions (black and white) separated by a linear segment. If the patch size become larger and the edges within the patches appear curved, extra degrees of freedom have to be included to the signal’s model thus leading to a more complex manifold. Similarly to cartoon images, the nature of the signal within signature patches is such that can be modeled by a handful of parameters if the patch size is small-enough, indicating a low-dimensional underlying manifold structure, but the complexity...
can be dramatically increased if the patch becomes big enough to contain curves and parts from neighboring line segments. We set patch size equal to 5, since it is a good tradeoff to the underlying signal’s complexity—since for smaller patches the local manifold obviously becomes degenerate and also to the overall computational complexity, which is dictated by the dictionary size whose over-completeness requirement points back to the patch dimensionality as the most significant parameter. It is worth noting that this scale of patches has also proven to be efficient in previous research efforts [8] on the particular problem, delivering state-of-the-art results and strengthening our decision on this selection.

3.3. SR-driven local feature extraction

Sparse Coding (SR) and consequent Dictionary Learning is a popular technique for handling computer vision problems. In this work, SR is involved in order to construct the dictionary (or Α for simplicity) that represents the characteristic properties of the signature. As already mentioned, the dictionary Α is considered to represent the characteristic properties of the signature. The patches extracted from one signature are represented as columns of the matrix Χ ∈ R^{n×k} and the dictionary for sparse representation is updated via an iterative process, in which the patches of one reference signature are used in each update and ultimately, all the reference signatures are utilized in a cascade fashion. Hence, dictionary is updated consecutively using each one of the writer’s genuine reference signatures. Thus, the system can integrate easily a new reference genuine signature of the writer and thus, it is practical for application in everyday scenarios. Following the construction of the dictionary, for every inserted signature image its patches are extracted and encoded using the dictionary and SR coding in order to obtain the sparse representation matrix Α. Depending on the case, the dictionary learning and the sparse coding are implemented with either the K-SVD/OMP or SPAMS/LARS-Lasso algorithm pairs. In addition, the impact on the system performance of some other popular optimization strategies is also explored with priors such as the positivity constraint of the coefficients, the non-negativity constraint for the dictionary atoms and the non-negative matrix factorization (NMF) method in which the matrix Χ and the vectors Α are required to be positive. For the cases in which Χ is positive, the SPAMS/LARS-Lasso algorithm is invoked but without the centering procedure of patches.

The operating parameters of the KSVD/OMP algorithms are set as follows: number of maximum iterations t_{max} = 50, number of atoms K = 60 in order to ensure the over-completeness (as a rule of thumb we use a number of atoms greater than twice the patch size dimension) and ρ = 3 for the sparsity level in order to provide a total 5% sparsity on the coefficients. For the online/LARS-Lasso case, the operating parameters are set as follows: number of atoms K = 60 (same as the KSVD/OMP), number of signals drawn at each iteration or mini-batch size n = 512, while the dictionary learning algorithm is allowed to run for a typical execution time of one minute, a time which is considered adequate for the size of the signature patches. In addition, the regularization parameter λ is set to 0.15, a value which is in proximity to the classical 1/√n normalization factor proposed by Bickel et al. [32].

3.4. Pooling Strategies

It is common for contemporary computer vision algorithms to incorporate a pooling stage, which aggregates local features over a region of interest [33-34], [11]. In this work, a number of variants are proposed as signature descriptors by globally aggregating the local patch sparse coding coefficients into a final vector, through an appropriate pooling function. Depending on the pooling function, the corresponding signature descriptor is denoted hereafter as \( f^{i^1}_t - f^{i^1}_t \), and defined as follows:

\[
(F_1): f^{i^1}_t = \{ f^{i^1}_t(j) \} = \{ \frac{1}{M} \sum_{j=1}^{M} a^{i^1}_j(j) \}, j = 1 : K
\]

\[
(F_2): f^{i^2}_t = \{ f^{i^2}_t(j) \} = \max_a^{i^2} \{ a^i(j) \}, i = 1 : M, j = 1 : K
\]

\[
(F_3): f^{i^3}_t = \{ f^{i^3}_t(j) \} = \{ \sqrt{\sum_{j=1}^{M} (a^i(j) - f^{i^1}_t(j))^2} \}, j = 1 : K
\]

\[
(F_4): f^{i^4}_t = \{ f^{i^4}_t(j) \} = \{ \sum_{i=1}^{M} a^i(j) \}
\]

\[
(F_5): f^{i^5}_t = \{ f^{i^5}_t(j) \} = \{ \sqrt{\sum_{j=1}^{M} (\sum_{i=1}^{M} a^i(j))^2} \}, j = 1 : K
\]

Average pooling (F1) is the simplest pooling function that estimates the average SR coefficients from the whole region of interest. The max pooling (F2) operation only captures the most salient representation value from the entire region of interest. Standard deviation (F3) is proposed here as an alternative pooling function that captures 2nd order statistics of the coefficients’ distribution, in an aim to investigate if this type of information can deliver better discrimination capabilities to the resulting descriptor. Normalized Sum pooling (F4) function produces vectors with intensity invariance qualities, and finally the (F5) function produces \( i^1 \) normalized vectors projected onto the unitary ball, which can be important for linear classification kernels [33].

The final feature vector’s dimensionality provided by the preceding pooling operations equals the number of dictionary atoms K. In order to encapsulate local signature information to the final feature vector, a specially designed spatial pyramid is employed for segmenting the signature images into a grid of (β × β) sub-regions. For each \( I_{β}^γ(t), t = 1 : β^Γ \) segment of this pyramid, its selected \( x_{β}^γ(t) \) patches are enabled for indexing the local \( a^i_{β}^γ(t) \) representation coefficients, which in their turn are subjected to the same pooling operations \( F(A_{β}^γ) \) that is used for the computation of the \( f^{i^1}_t - f^{i^1}_t \) global versions. As an aftermath, the dimensionality of the expanded feature \( f^{γ}_t = \{ f^{i^1}_t, f^{i^2}_t, \ldots, f^{i^5}_t \}, g = 1 : 5, \) now equals to (β × β + 1) × K. In this work we tested the system’s performance for two values of the β parameters, i.e. 2 × 2 and 3 × 3 equi-mass sub-regions for the spatial pyramid.

3.5. Emphasizing into informative local keypoints

...
A human expert, who wishes to analyze a signature image in order to verify if it is genuine or forgery, focuses at certain points of the signature. In an effort to discover these signature’s points of interest we utilize the saliency-modeling mechanism implemented by the Binary Robust Invariant Scalable Keypoints (BRISK) [28]. BRISK have the advantage of dramatically lowering computational complexity and thus are suited for low power devices, such as practical portable SV systems. BRISK computation relies on an easily configurable circular sampling pattern from which it computes brightness comparisons to form a binary 512 bit descriptor string and estimates keypoint scale in continuous scale-space. In this work only the detected keypoint locations and not the keypoint descriptors (BRISK) are utilized. Hence, keypoints indicate the patches, and thereafter the corresponding Sparse Codes will be pooled together in order to obtain an additional feature vector. This vector is concatenated with the spatial pyramid vector resulting to a final feature vector $f' = (f_1', ..., f_m')$ of dimensionality is $(\beta + 2) \times K$. Figure 3, depicts a zoomed area of an example signature image where the BRISK keypoints and their nearest signature pixels are denoted as crosses and x-points, respectively.

4. Evaluation

The selected classifier is a binary radial basis SVM classifier. In the training stage of the classifier, each one of the $N_{g-ref}$ genuine samples is encoded with the OMP or the LARS-Lasso SR algorithms in order to provide the positive class $\omega^+ \in R^{N_{g-ref} \times (Dim)}$ for each one of the feature $f_1' - f_m'$ descriptors given in (1)-(5). The negative training class $\omega^- \in R^{2 \times N_{g-ref} \times (Dim)}$ is composed by taking $2 \times N_{g-ref}$ random forgery samples that is one genuine sample from $2 \times N_{g-ref}$ writers other than the examined one. Thus, a corresponding learning feature population $\{\omega^+ : \omega^-\}$ is used as an input to the classifier while a holdout cross-validation procedure returns the optimal values of the $C^{opt}$ and gamma-$\gamma^{opt}$ parameters with respect to a maximum cross validation value of the associated Area Under Curve [8]. In addition, the cross-validation procedure stores the output scores $CVS$ which are conditioned upon the positive $\omega^+$ class samples.

4.1. Datasets and Experimental Protocols

Four signature datasets are used in order to test the proposed system architecture. The first one is the popular CEDAR dataset [35]. For each one of the 55 enrolled writers, a total of 48 signature specimens (24 genuine and 24 simulated) confined in a 50 mm by 50 mm square box are provided and digitized at 300 dpi. The simulated signatures found in the CEDAR dataset are composed from a mixture of random, simple and skilled forgeries. The second signature dataset is the off-line version of the popular MCYT signature database [36-37]. A whole of 30 signature samples (15 genuine and 15 simulated) signature samples are recorded for each one of the 75 enrolled writers at a resolution of 600 dpi and for and the capture area is 127 x 97mm. Both CEDAR and MCYT-75 datasets have their samples confined within one bounding box. The third signature dataset is the GPDS300 [37-38] which contains 24 genuine signatures and 30 simulated forgeries of 300 individuals stored in an 8-bit, grey level format. A special feature of this dataset is that the acquisition of signature specimens is carried out with the aid of two different bounding boxes of size 5 x 1.8 cm and 4.5 x 2.5 cm respectively. As a result, the files of this dataset include images having two different aspect ratios; this phenomenon conveys a structural distortion highlighted during the feature extraction procedure. The fourth signature dataset is the Persian UTSIG, created by Soleimani et al. [39]. It contains specimens from 115 writers where each one has 27 genuine signatures, 3 opposite-hand signatures, and 42 skilled forgeries made by 6 forgers. As stated by the dataset creators, a property of the UTSIG dataset, compared with the other public and popular ones, is that UTSIG has more samples, more classes, and more forgers. An important characteristic of the UTSIG is that the acquisition of signature specimens is carried out with the aid of six different bounding box sizes simulating real world conditions and public services application forms. Since this work addresses writer dependent signature verification, for any signatory a specific model is created by randomly employing $N_{g-ref}$ genuine reference signature samples. The number of $N_{g-ref}$ is primarily set to 5 for addressing cases in which only a few samples are available. In order to provide comparable results with other state-of-the-art methods, we allowed the $N_{g-ref}$ parameter to assume values from 5 and/or 12 according to the specific needs of each dataset.

A number of methods exist in order to quantify the efficiency of the proposed system. The false acceptance rate FAR and its associate probability $P_{FAR}$ depicts a system’s measure of the resistance to input samples other that the genuine ones like random or skilled forgeries. On the other hand, the false rejection rate FRR and its accompanied probability $P_{FRR}$ provides a system’s measure of the failure to genuine samples. These operating system parameters are computed as a function of a sliding threshold whose extremes are located between the minimum and maximum values of the cross-validation SVM output scores $CVS$. Some examples include independent experiments and corresponding solutions like: a) the $P^*_{FAR}$ vs. $P_{FAR}$ where the upper script S denotes skilled forgery, along with their average $AVE_S$ or equal error rates: $EER^S : P_{FAR} = P^*_{FAR}$, and b) the $P^*_{FAR}$ vs. $P_{FAR}$, where the upper script R denotes random forgery, along with their corresponding average $AVE_R$ or equal error rates $EER^R : P_{FAR} = P^*_{FAR}$. Other methods provide joint solutions [40, [6-7] which initially
evaluate the $TH^{a_{EER}}$ - threshold, defined as the value which designates the EER operating point and then, taking into account this value in order to assess the $P_{FAR}^{EER}$ rate error.

Regarding the choice of metrics, several researchers suggest handling SV as either a one class pattern recognition problem or a two class pattern recognition problem. The issue arises from the fact that the negative class of the test set has representatives from both skilled and random forgery samples. The key point is that the skilled forgery class is composed of few samples compared with the random class. So, if one tries to incorporate both forgery populations there is always the danger of reporting biased results. For example, reporting an EER point of $EER^{a_{EER}}: P_{EER}$ where all types of forgery samples are employed; this is clearly biased since the forgery class has samples with two different populations of forgery. In this paper, as in [6], [7] the $EER$ is used by employing user-specific decision thresholds in order to evaluate the verification performance of the proposed system. Also the calculation of the $P_{FAR}^{a_{EER}}$ rates with the utilization of a predetermined threshold (i.e. hard decisions) is provided by using the a-priori knowledge of the cross validation procedure scores $CVS^a$. Specifically, the hard threshold value corresponds to the 50% of the average of the genuine $CVS^a$ scores for each writer. For completeness, at this specific threshold point the $P_{FAR}^{a_{EER}}$ error rate is evaluated by employing the genuine samples of the remaining writers from the test set.

5. EXPERIMENTAL RESULTS

For the sake of sanity and in order to avoid exhausting tests on all datasets, complete results are provided in tables 1-2 which involve the popular CEDAR and MCYT-75 datasets. All the experimental protocols are repeated ten times and the results are averaged in order to provide meaningful comparisons. It can be noticed that the verification error rates for the CEDAR and MCYT datasets do not vary substantially when both $l_1$ and $l_2$ -norms dictionary learning and SR are used. This is in accordance to the material exposed to section 3.2 which states that for a given dictionary $D$, both $l_1$ and $l_2$ oriented SR solutions are equivalent, for some specific values of the design parameters $\rho$, and $\lambda$. Of course, the problem of locating the appropriate value for these design parameters is not a trivial one since it depends on the individual signature characteristics. An additional issue that arises is that tuning the system to a specific value either for $\rho$ or $\lambda$ depends mainly upon the different training modes. Moreover, we observed during the conducted experiments that the cross validation procedure, which is used for the selection of the optimal classifier parameters, is almost ineffective against skilled forgeries. A potential solution could emerge by employing WI systems with skilled forgeries which are not assigned to any specific person during the training [23]. However, this approach is out of the scope of the present work.

The experimental outcomes show that the best results for the CEDAR dataset are obtained with the use of a spatial pyramid with $\beta = 2$ while for the MCYT-75 the optimal results are obtained with the use of a spatial pyramid with $\beta = 3$. This is probably due to the fact that CEDAR signature specimens have been scanned with a resolution of 300dpi, while the MCYT-75 ones with a resolution of 600dpi, thus it is natural to expect that more pixels exist in the MCYT-75 segments. It should be noticed that, for the CEDAR dataset even when the spatial pyramid with $\beta = 2$ is used, there are a few patches in some segments, which profoundly did not provide any sort of discriminatory information, something that has not been encountered in the case of the MCYT-75.

An important result that arises from our experimental results is that the accuracy for the F3 pooling function seems to outperform all other pooling functions in almost all cases. It is also interesting that although the F2 feature performs quite well in other computer vision applications, this is not the case for signatures images. A possible explanation relies on the very nature of the signature images which are a particular class of image signals that exhibit a degenerate structure. Since all signature images essentially share a limited set of structural elements, the resulting sparse coefficients of different signatures may not differ sufficiently in a $1^\text{st}$ order statistic sense, as being the case in more complex image structures. Therefore, it is reasonable to deduce that higher-order statistics can deliver a better level of discrimination between the respective distributions of sparse coefficients, resulting into better verification performance.

### Table 1

| SET     | F(A) | SP: $\beta = 2$ (Dim=360) | SP: $\beta = 3$ (Dim=660) |
|---------|------|--------------------------|--------------------------|
|         | $P_{FAR}^{a_{EER}}$ | $P_{FAR}^{a_{EER}}$ | $P_{FAR}^{a_{EER}}$ | $P_{FAR}^{a_{EER}}$ | $P_{FAR}^{a_{EER}}$ | $P_{FAR}^{a_{EER}}$ |
| CEDAR   | F1   | 6.83                     | 7.32                     | 6.13                     | 7.18                     | 8.00                     | 0.27                     |
|         | F2   | 9.45                     | 9.83                     | 8.83                     | 8.65                     | 3.21                     | 0.76                     |
|         | F3   | 4.76                     | 4.91                     | 4.78                     | 5.63                     | 1.80                     | 0.12                     |
|         | F4   | 8.12                     | 9.94                     | 12.7                     | 20.2                     | 13.4                     | 4.82                     |
|         | F5   | 5.31                     | 5.27                     | 5.05                     | 5.11                     | 2.48                     | 0.24                     |
|         | F6   | 10.4                     | 7.32                     | 8.22                     | 7.12                     | 3.19                     | 0.14                     |
|         | F7   | 14.7                     | 14.6                     | 15.6                     | 11.5                     | 8.35                     | 1.12                     |
|         | F8   | 8.09                     | 6.75                     | 8.07                     | 6.21                     | 2.82                     | 0.07                     |
|         | F9   | 12.99                    | 16.17                    | 11.0                     | 23.7                     | 16.7                     | 6.17                     |
|         | F10  | 7.64                     | 7.76                     | 7.77                     | 7.92                     | 3.46                     | 0.13                     |

### Table 2

| SET     | F(A) | SP: $\beta = 2$ (Dim=360) | SP: $\beta = 3$ (Dim=660) |
|---------|------|--------------------------|--------------------------|
|         | $P_{FAR}^{a_{EER}}$ | $P_{FAR}^{a_{EER}}$ | $P_{FAR}^{a_{EER}}$ | $P_{FAR}^{a_{EER}}$ | $P_{FAR}^{a_{EER}}$ | $P_{FAR}^{a_{EER}}$ |
| CEDAR   | F1   | 6.99                     | 7.60                     | 2.65                     | 0.37                     | 5.59                     | 6.61                     | 3.08                     | 0.23                     |
|         | F2   | 9.42                     | 9.71                     | 3.45                     | 1.55                     | 8.52                     | 8.01                     | 3.40                     | 0.48                     |
|         | F3   | 4.61                     | 4.64                     | 1.62                     | 0.12                     | 4.88                     | 5.81                     | 2.01                     | 0.08                     |
|         | F4   | 8.19                     | 8.73                     | 3.52                     | 0.56                     | 9.17                     | 8.71                     | 3.97                     | 0.41                     |
|         | F5   | 7.05                     | 7.81                     | 3.38                     | 0.41                     | 5.49                     | 6.48                     | 3.07                     | 0.22                     |
|         | F6   | 10.46                    | 8.57                     | 3.91                     | 0.24                     | 7.82                     | 7.68                     | 3.58                     | 0.14                     |
|         | F7   | 15.8                     | 14.3                     | 11.28                    | 3.09                     | 13.9                     | 12.3                     | 8.60                     | 1.08                     |
|         | F8   | 8.35                     | 7.58                     | 3.71                     | 0.22                     | 7.59                     | 7.29                     | 3.08                     | 0.10                     |
|         | F9   | 12.2                     | 13.4                     | 3.93                     | 0.30                     | 9.62                     | 9.68                     | 3.53                     | 0.15                     |
|         | F10  | 7.89                     | 7.59                     | 3.66                     | 0.20                     | 7.07                     | 7.69                     | 3.08                     | 0.12                     |
TABLE 3

Verification error rates (%) for the CEDAR and MCYT-75 signature dataset with L2-norm: and for the following priors a) positivity constraint of the A coefficients, b) the 'non-negative' C' constraint for the dictionary atoms and c) the NMF method. \( \beta_{\text{CEDAR}} = 2 \), \( \beta_{\text{MCYT}} = 3 \).

| SET     | A positive | C dictionary constraints | NMF       |
|---------|------------|--------------------------|-----------|
| F(A)    |            |                          |           |
| \( p_{xx}^{A} \) | \( p_{xx}^{A} \) | \( EER_{xx}^{A} \) | \( P_{xx}^{A} \) | \( EER_{xx}^{A} \) | \( P_{xx}^{A} \) | \( EER_{xx}^{A} \) | \( P_{xx}^{A} \) | \( EER_{xx}^{A} \) | \( P_{xx}^{A} \) | \( EER_{xx}^{A} \) |
| CEDAR   | F1 6.91   | 6.22                      | 0.59      | 6.46               | 6.47               | 3.14               | 0.44               |
|         | F2 8.87   | 8.59                      | 3.98      | 11.79              | 9.57               | 4.43               | 1.82               |
|         | F3 6.73   | 6.13                      | 1.50      | 5.81               | 5.92               | 1.97               | 0.19               |
|         | F4 16.11  | 19.32                     | 6.23      | 18.60              | 29.40              | 20.44              | 10.00              |
|         | F5 7.79   | 7.96                      | 3.30      | 5.78               | 5.83               | 5.55               | 0.33               |
|         | F6 18.37  | 16.97                     | 1.71      | 16.59              | 4.38               | 18.60              | 4.43               |
|         | F7 6.62   | 6.69                      | 1.42      | 17.86              | 17.86              | 12.48              | 0.92               |
| UTSIG   | F1 13.65  | 13.83                     | 0.53      | 12.97              | 7.53               | 0.25               |
|         | F2 19.93  | 18.12                     | 3.06      | 18.59              | 13.11              | 0.97               |
|         | F3 13.38  | 11.38                     | 3.30      | 9.57               | 6.22               | 0.12               |
|         | F4 23.75  | 24.76                     | 3.32      | 29.31              | 21.01              | 6.63               |
|         | F5 13.27  | 12.57                     | 0.39      | 10.25              | 7.44               | 0.21               |

For further investigation we selected the spatial segmentation that corresponds to the best results, i.e. \( \beta = 2 \) for the CEDAR and \( \beta = 3 \) for the MCYT-75, and the \( \ell_1 \)-norm. Table 3 provides comparisons with exactly the same reference and training samples for the following cases: a) the positivity constraint of the A coefficients, b) the 'non-negative' C' constraint for the dictionary atoms and c) the NMF method. The results indicate that none of these cases seems to provide any indiscriminately nature of the spatial pyramid equisegment which does not take into account the two different aspect ratios of the bounding box. As mentioned earlier, the UTSIG dataset is, according to the author's opinion, a significant realization step towards the assessment of situations which resemble typical conditions and constraints that are broadly encountered in daily transactions. Table 4 also presents the results for our usual case of \( N_{g-aref} = 5 \). As expected, the provided verification results are poor regarding to the rates of the previous datasets however a closer look to \( P_{xx}^{A} \) indicates that most likely the \( N_{g-aref} = 5 \) is not adequate in this case due to the large number of bounding box sizes. Therefore, in accordance with the literature [39], [9] and for comparison purposes, we let the value of \( N_{g-aref} \) raise up to twelve. Again, Table 4 presents the corresponding results. Table 4 provides evidence that the evaluation metrics rate drops significantly when the number of reference samples increases. Following and for the sake of simplicity, Table 5 shows comparative results by means of the EER after threshold only for the case in which each input signature is thinned not by the MOTL value of the claimed writer's reference set but with an OTL number provided for each individual signature. Clearly, the results when using the MOTL value are inferior, an outcome that emphasizes the importance of the proposed preprocessing technique. Table 6 demonstrates the influence of the segmentation profile to the performance verification in case of the CEDAR dataset. Specifically, the following five segmentation scenarios are examined a) using the entire image (EI), b) applying the SP segmentation, c) using only the BRISK keypoints, d) EI in conjunction with SP and finally e) a conjunction of EI, SP and BRISK keypoints. It is evident that the proposed segmentation approach that uses EI, SP and BRISK keypoints outperforms all the other scenarios for all pooling functions (with a minor exception of F5). Moreover, F3 pooling function, and corresponding feature vector persistently provides the best results in all segmentation scenarios and achieves the overall lower error in the proposed segmentation approach (last column of Table 6).

Perhaps the most challenging task on SV is the comparison of the results emanating from several state-of-the-art systems, as there is an abundance of degrees of freedom regarding the type or number of signatures utilized during the classifier construction and evaluation. Still, Table 7 provides evidence that the proposed method achieves a lower error of verification when a few genuine samples are available which is at least comparable to the ones derived from other state-of-the-art methods. It outlines and compares the \( EER_{\text{best}, \text{threshold}} \) based results of our
proposed method, with emphasis to the F₃ pooling, with a number of state-of-the-art SV related methods for the four utilized datasets on an EER or the average error AER basis.

6. CONCLUSIONS

In this paper the potential of Sparse Representation on creating discriminative features for accurate and efficient offline signature verification is presented. We thoroughly investigated the major aspects on the selection of the appropriate SR approach, and examined the effects of the associated parameters. We demonstrated that approximate greedy techniques can deploy the full potential of SR in a SV system, in a computationally attractive manner. We described a novel pooling scheme tailored to the problem of SV, and evinced that 2nd order statistics can form more discriminative pooling functions in cases where signals exhibit degenerate structures. Finally, we proposed a carefully designed system, encompassing a novel algorithm for the automatic selection of the optimal thinning level, which is able to harness the power of SR in order to create a discriminative signature descriptor which obtains state-of-the-art results on the most challenging signature datasets.

Our future research plans include the exploitation of SR related techniques in order to construct a universal dictionary with the use of samples originating from a wider and diverse set of persons and not just from one signature in order to develop an efficient writer independent signature verifier.

REFERENCES

[1] G. Pirlo, M. Díaz, M. A. Ferrer, D. Impedovo, F. Occhionero and U. Zurlo, “Early Diagnosis of Neurodegenerative Diseases by Handwritten Signature Analysis,” New Trends in Image Analysis and Processing –ICIAP 2015 Workshops. pp. 290-297.

[2] L. G. Hafemann, R. Sabourin, and L. S. Oliveira, “Offline Handwritten Signature Verification - Literature Review”, in 7th International Conference on Image Processing Theory, Tools and Applications (IPTA 2017), Montreal, Canada, 2017.

[3] D. Impedovo, G. Pirlo, and R. Plamondon, “Handwritten Signature Verification: New Advancements and Open Issues,” in 2012 International Conference on Frontiers in Handwriting Recognition, pp. 367-372.

[4] R. Plamondon, G. Pirlo, and D. Impedovo, “Online Signature Verification,” Handbook of Document Image Processing and Recognition, D. Doermann and K. Tombre, eds., pp. 917-947, London: Springer London, 2014.

[5] S. Pal, M. Blumenstein, and U. Pal, “Off-line signature verification systems: a survey,” in Proceedings of the International Conference; Workshop on Emerging Trends in Technology, Mumbai, Maharashtra, India, 2011, pp. 652-657.

[6] E. N. Zois, I. Theodorakopoulos, D. Tsourounis et al., “Parsimonious Coding and Verification of Offline Handwritten Signatures,” in 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2017, pp. 636-645.

[7] E. N. Zois, I. Theodorakopoulos, and G. Economou, “Offline Handwritten Signature Modeling and Verification Based on Archetypal Analysis,” in 2017 IEEE International Conference on Computer Vision, 2017, pp. 5515-5524.

[8] E. N. Zois, L. Alewijnse, and G. Economou, “Offline signature verification and quality characterization using poset-oriented grid features,” Pattern Recognition, vol. 54, pp. 162-177, 2016.

[9] A. Soleimani, B. N. Araab, and K. Fouladi, “Deep Multitask Metric Learning for Offline Signature Verification,” Pattern Recognition Letters, vol. 80, pp. 84-90, 2016.

[10] S. Dey, A. Dutta, J. I. Toledo et al., “SigNet: Convolutional Siamese Network for Writer Independent Offline Signature Verification,” arXiv:1707.02131, 2017.

[11] M. Okawa, “Vector of locally aggregated descriptors with KAZE features for offline signature verification,” in 2016 IEEE 5th Global Conference on Consumer Electronics, 2016.

[12] M. Okawa, “Offline Signature Verification Based on Bag-Of-Visual Words Model Using KAZE Features and Weighting Schemes,” in 2016 IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2016, pp. 184-190.

[13] A. Dutta, U. Pal, and J. Lladós, “Compact correlated features for writer independent signature verification,” in 23rd International Conference on Pattern Recognition, 2016, pp. 3422-3427.
[14] Y. Serdouk, H. Nemmour, and Y. Chibani, “Handwritten signature verification using the quad-tree histogram of templates and a Support Vector-based artificial immune classification,” Image and Vision Computing, vol. 66, pp. 26-35, 2017.

[15] B. A. Olshausen, and D. J. Field, “Emergence of Simple-Cell Receptive-Field Properties by Learning a Sparse Code for Natural Images,” Nature, vol. 381, no. 6583, pp. 607-609, 1996.

[16] Z. Zhang, X. Xu, J. Yang et al., “A Survey of Sparse Representation: Algorithms and Applications,” IEEE Access, vol. 3, pp. 490-530, 2015.

[17] B. Ribeiro, I. Gonçalves, S. Santos et al., “Deep Learning Networks for Off-Line Handwritten Signature Recognition,” 16th conference on Progress in Pattern Recognition, Image Analysis and Applications, CIARP-11, 2011, pp. 523-532.

[18] Khajalizadeh Hurieh, M. Mansouri, and M. Teshehlab, “Persian Signature Verification using Convolutional Neural Networks,” International Journal of Engineering Research and Technology (IJERT), vol. 1, no. 2, pp. 7-12, 2012.

[19] Z. Zhang, X. Liu, and Y. Cui, “Multi-phase Offline Signature Verification System Using Deep Convolutional Generative Adversarial Networks,” 9th International Symposium on Computational Intelligence and Design, 2016, pp. 103-107.

[20] H. Rantzsch, H. Yang, and C. Meinel, “Signature Embedding: Writer Independent Offline Signature Verification with Deep Metric Learning,” in International Symposium on Visual Computing, 2016, pp. 615-625.

[21] L. G. Hafemann, R. Sabourin, and L. S. Oliveira, “Writer-independent feature learning for Offline Signature Verification using Deep Convolutional Neural Networks,” in 2016 International Joint Conference on Neural Networks (IJCNN), pp. 2576-2583.

[22] L. G. Hafemann, R. Sabourin, and L. S. Oliveira, “Analyzing features learned for Offline Signature Verification using Deep CNNs,” in 2016 23rd International Conference on Pattern Recognition, 2016, pp. 2989-2994.

[23] L. G. Hafemann, R. Sabourin, and L. S. Oliveira, “Learning features for offline handwritten signature verification using deep convolutional neural networks,” Pattern Recognition, vol. 70, pp. 163-176, 2017.

[24] R. Kumar, B. Chanda, and J. D. Sharma, “A novel sparse model based forensic writer identification,” Pattern Recognition Letters, vol. 35, pp. 105-112, 2014.

[25] Y. Liu, Z. Yang, and L. Yang, “Online Signature Verification Based on DCT and Sparse Representation,” IEEE Transactions on Cybernetics, vol. 45, no. 11, pp. 2498-2511, 2015.

[26] L. Batista, E. Granger, and R. Sabourin, “Dynamic selection of generative–discriminative ensembles for off-line signature verification,” Pattern Recognition, vol. 45, no. 4, pp. 1326-1340, 2012.

[27] Ng Andrew, and Yu Kai. “ECCV-2010 Tutorial: Feature Learning for Image Classification, http://ufldl.stanford.edu/eccv10-tutorial/”

[28] S. Leutenegger, M. Chili, and R. Y. Siegwart, “BRISK: Binary Robust Invariant Scalable keypoints,” in 2011 International Conference on Computer Vision, 2011, pp. 2548-2555.

[29] N. Otsu, “A Threshold Selection Method from Gray-Level Histograms,” IEEE Transactions on Systems, Man and Cybernetics, vol. 9, no. 1, pp. 62-66, 1979.

[30] J. C. Yunchao Zhang, Xiujie Huang1, Yongtian Wang, “A Probabilistic Analysis of Sparse Coded Feature Pooling and Its Application for Image Retrieval,” PLoS ONE., vol. 10, no. 7, pp. 1-18, 2015.

[31] G. Peyré, “Manifold models for signals and images,” Computer Vision and Image Understanding, vol. 113, no. 2, pp. 249-260, 2009.

[32] P. J. Bickel, V. a. Ritov, and A. B. Tsybakov, “Simultaneous analysis of Lasso and Dantzig selector,” Ann. Statist., vol. 37, no. 4, pp. 1705-1732, 2009.

[33] J. Feng, B. Ni, Q. Tian et al., “Geometric lp-norm feature pooling for image classification,” in CVPR, 2011, pp. 2609-2704.

[34] H. Jégou, F. Perronnin, M. Douze et al., “Aggregating Local Image Descriptors into Compact Codes,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 34, no. 9, pp. 1704-1716, 2012.

[35] M. K. Kalera, S. Srihari, and A. Xu, “Offline signature verification and identification using distance statistics,” International Journal of Pattern Recognition and Artificial Intelligence, vol. 18, no. 07, pp. 1339-1360, 2004.

[36] J. Ortega-Garcia, J. Fierrez-Aguilar, D. Simon et al., “MCYT baseline corpus: a bimodal biometric database,” IEEE Proceedings Vision, Image and Signal Processing, vol. 150, pp. 395-401, 2003.

[37] J. F. Vargas, M. A. Ferrer, C. M. Travieso et al., “Offline signature verification based on gray level information using texture features,” Pattern Recognition, vol. 44, no. 2, pp. 375-385, 2011.

[38] M. Diaz, M. A. Ferrer, G. S. Eskander et al., “Generation of Duplicated Off-Line Signature Images for Verification Systems,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 5, pp. 951-964, 2017.

[39] A. Soleimani, K. Fouladi, and B. N. Araabi, “UTSig: A Persian offline signature dataset,” IET Biometrics, vol. 6, no. 1, pp. 1-8, 2016.

[40] H. Loka, E. Zois, and G. Economou, “Long range correlation of preceded pixels relations and application to off-line signature verification,” IET Biometrics, vol. 6, no. 2, pp. 70-78, 2016.

[41] R. Kumar, J. D. Sharma, and B. Chanda, “Writer-independent off-line signature verification using surroundedness feature,” Pattern Recognition Letters, vol. 33, no. 3, pp. 301-308, 2012.

[42] S. Chen, and S. Srihari, “A New Off-line Signature Verification Method based on Graph,” in 18th International Conference on Pattern Recognition, 2006, pp. 869-872.

[43] Y. Guerbai, Y. Chibani, and B. Hadjadji, “The effective use of the one-class SVM classifier for handwritten signature verification based on writer-independent parameters,” Pattern Recognition, vol. 48, no. 1, pp. 103-113, 2015.

[44] Y. Serdouk, H. Nemmour, and Y. Chibani, “New off-line Handwritten Signature Verification method based on Artificial Immune Recognition System,” Expert Systems with Applications, vol. 51, pp. 186-194, 2016.

[45] R. K. Bharathi, and B. H. Shekar, “Off-line signature verification based on chain code histogram and Support Vector Machine,” in Advances in ICAICI 2013, pp. 2063-2068.

[46] G. Ganapathi, and R. Nadarajan, “A Fuzzy Hybrid Framework for Offline Signature Verification,” in PReMI 2013, pp. 121-127.

[47] B. H. Shekar, R. K. Bharathi, and B. Pilar, “Local Morphological Pattern Spectrum Based Approach for Off-line Signature Verification,” in PReMI 2013, pp. 335-342.

[48] A. Hamadene, and Y. Chibani, “One-Class Writer-Independent Offline Signature Verification Using Feature Dissimilarity Thresholding,” IEEE Transactions on Information Forensics and Security, vol. 11, no. 6, pp. 1226-1238, 2016.

[49] F. Alonso-Fernandez, M. C. Fairhurst, J. Fierrez et al., “Automatic Measures for Predicting Performance in Off-Line Signature,” in IEEE International Conference on Image Processing, 2007, pp. 369-372.

[50] J. Fierrez-Aguilar, N. Alonso-Hermia, G. Moreno-Marquez et al., “An Off-line Signature Verification System Based on Fusion of Local and Global Information,” Lecture Notes in Biometric Authentication, pp. 295-306, 2004.

[51] J. Wen, B. Fang, Y. Y. Tang et al., “Model-based signature verification with rotation invariant features”, Pattern Recognition, vol. 42, no. 7, pp.
[52] S. Y. Ooi, A. B. J. Teoh, Y. H. Pang et al., “Image-based handwritten signature verification using hybrid methods of discrete Radon transform, principal component analysis and probabilistic neural network”, Applied Soft Computing, vol. 40, pp. 274-282, 2016.
[53] M. B. Yılmaz, and B. Yanıkoğlu, “Score level fusion of classifiers in off-line signature verification,” Information Fusion, vol. 32, Part B, pp. 109-119, 2016.
[54] G. S. Eskander, R. Sabourin, and E. Granger, “Hybrid writer-independent writer-dependent offline signature verification system,” IET Biometrics, vol. 2, no. 4, pp. 169-181, 2013.
[55] A. Alaei, S. Pal, U. Pal et al., “An Efficient Signature Verification Method Based on an Interval Symbolic Representation and a Fuzzy Similarity Measure,” IEEE Transactions on Information Forensics and Security, vol. 12, no. 10, pp. 2360-2372, 2017.
[56] G. Pirlo, and D. Impedovo, “Cosine similarity for analysis and verification of static signatures,” IET Biometrics, vol. 2, no. 4, pp. 151-158, 2013.
[57] G. Pirlo, and D. Impedovo, “Verification of Static Signatures by Optical Flow Analysis,” IEEE Transactions on Human-Machine Systems, vol. 43, no. 5, pp. 499-505, 2013.
[58] M. Parodi, J. C. Gomez, and A. Belaid, “A Circular Grid-Based Rotation Invariant Feature Extraction Approach for Off-line Signature Verification,” in Proceedings 11th International Conference on Document Analysis and Recognition, 2011, pp. 1289-1293.