



Off-line signature verification system using features linear mapping in the candidate points

Maryam Houtinezhad¹ · Hamid Reza Ghaffari¹

Received: 3 April 2020 / Revised: 30 May 2021 / Accepted: 25 January 2022 /
Published online: 22 March 2022

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2022

Abstract

This paper proposes BoF-CP approach which is a novel scheme feature fusion for writer-dependent offline signature verification. At the heart of the new methodology lies feature extraction in surrounded candidate points. In the proposed method, at first, several features of the type of component-oriented and pixel-oriented features are extracted at the region of candidate points. Due to the different geometric structure in different signatures, several short feature vectors are created to the number of candidate points in each image. In the proposed approach, the corresponding homogeneous feature vectors are fused based on standard deviation and variance at the candidate points to create a normalized vector. We called the proposed method Bag of Features in Candidate Point (BoF-CP). Finally, the normalized feature vector enters the classification stage to verify the query samples. To evaluate the proposed method, standard datasets MCYT, GPDS, and CEDAR have been used. According to the experimental results, the proposed approach has been able to import optimal features into the classification algorithm by which the recognition rate in the separation of the genuine and forgery samples has been enhanced. According to the obtained statistical results, the proposed method has improved in several criteria such as average error rate, accuracy, sensitivity, specificity as compared to state of the art.

Keywords Off-line signature verification system (OS-VS) · Candidate point · Hand-craft feature · Bag of feature

✉ Maryam Houtinezhad
m.houtinezhad@srbiau.ac.ir

Hamid Reza Ghaffari
hghaffari@ferdowsiau.ac.ir

¹ Department of Computer Engineering, Ferdows Branch, Islamic Azad University, Ferdows, Iran

1 Introduction

“Are you really Ms. / Mr. X?” and “Who are you?” are the sentences that these days are often seen in the authentication and identification issue. In this topic, the offline signature system is one of the behavioral biometrics used to verify identity [25–28, 35, 58, 76, 82]. Biometrics is one of the most widely used tools for the verification and identification of individuals. Biometric techniques are divided into two categories [12]: behavioral and physical. In physical subsets, special properties such as gait [6], facial [9, 43, 47, 74] and iris [44, 46], and etc. are examined. On the other hand, in the behavioral subset, properties such as signature [16, 27], speech [29] and handwriting [2–4], and etc. are examined. In this research, we are focused on signature behavioral biometrics. The advantage that has made signature so popular is that signature has a long usage history of presence in different societies. Besides that, most persons are familiar with how it is used throughout life, which is why it is generic, legal, and acceptable to the majority of individuals in the community [27]. For example, the signature has been widely accepted in the bank as an indicator of the identity of the individual. Also, the tools for receiving and recognizing signatures are less expensive than other biometrics. Figure 1 shows the grouping of different aspects studied by researchers for the offline signature verification system (OSV). This classification is based on the references of [16, 27].

According to the literature, there are various terms that are discussed below:

In term online/dynamic, special tools and tablets are used in real-time. In this case, in addition to enrolling the signature samples, other dynamic features that are generated by the writers when drawing the signature such as pen pressure, pen up, pen down, pen direction, and etc. are also registered in the system. But in term offline/static, only the scanned image of the signature sample is registered, so one of the challenges of static acquisition is that it lacks the informative features that are created when drawing. Hence, static-based methods focus more on providing informative features that distinguish genuine signatures from forgery ones.

On the other hand, in the offline signature verification system, the decision-making process to classify the genuine and forgery samples are made based on the trained model. In general, there are two scenarios for the training phase in the signature verification system. (1) The writer-independent (WI) scenario in which the training is done in generic and with an analogous classification. (2) The writer-dependent (WD) scenario in which the training is done individually for each sample of signature. From another point of perspective, the process of forging signatures is divided into three groups [16, 42, 82]:

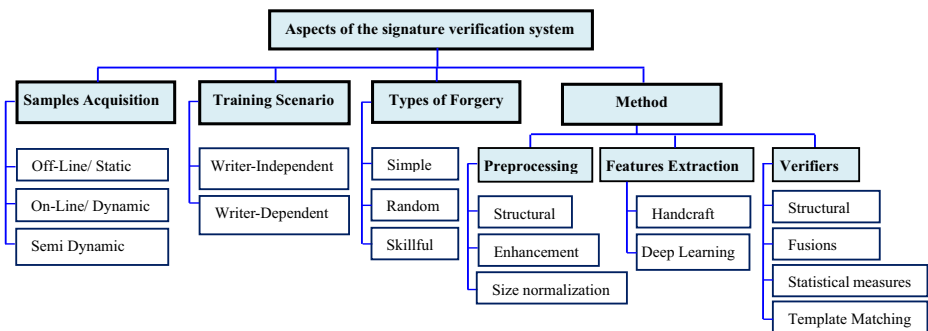


Fig. 1 Perspective of aspects of the SV

- (A) Simple forgery: In simple forgery, the forger knows the name of the user but does not know how his/her signature is drawn. In this case, the signature is less similar to the genuine sample.
- (B) Random forgery where the forger does not know the user profile or signature. So, the forger draws a random signature. In this case, forgery samples have different shapes than the genuine sample.
- (C) Skillful forgery: In this type of forgery, the forger has access to the name and signature of the user and has practiced how to draw a signature. In this case, the signature is very similar to the genuine signature, which makes it difficult to identify.

The performance of the signature verification is strongly influenced by the practice of forgers. In this case, the forger can imitate the genuine signature with a lot of practice. This issue causes increases the similarity of the class of forgery and genuine samples, which can ultimately reduce the efficiency of the signature system.

Therefore, according to the state-of-the-art, focusing on the three main phases of preprocessing, feature extraction and classification can be increased system efficiency. For example, in the preprocessing step, an operation can be performed to enhancement of image resolution [16]. Many types of research such as background removal [14], morphological operations [80], filtering [15, 20] and etc. have been done in this field. In addition, various operations such as cropping [22, 80] fixed area [71], and length equalization [79] have been performed to normalize the size of the signature box.

It is necessary to mention, in the feature extraction phase, one can focus on one of the techniques based on hand-crafted features and deep features [27]. Due to the focus of this article on hand-crafted methods, we can refer to such things as the geometric features of the signature, the direction of the baseline, graphometric and curvature, which are called component-oriented features [16]. Other commonly used features in the SV include the extraction of key-points, which can include SIFT, SURF, BRISK, and KAZE, these types of features are called pixel-oriented. All the techniques presented in the feature extraction phase are in order to better represent the features of the signature image. In addition, providing an informative feature vector can increase the recognition rate in the verify/classify phase [30].

In addition to the above-mentioned many studies in the classification phase have provided various techniques for separating the genuine and forgery signatures. Conforming to reference [16], classification methods divided into five categories: (1) template matching such as dynamic time warping (DTW) [66, 67], direct matching points [57], Euclidean [22], (2) statistical measures such as mahalanobis [72], membership [13] functions, cosine [55], (3) statistical models such as neural networks (NNs) [38], deep learning, recurrent neural networks (RNNs), deep neural networks (DNN) [1, 39, 73], deep multitask metric learning (DMML) [68], convolutional neural networks (CNNs) [25, 26, 78], hidden markov models (HMMs) [14, 19] support vector machine (SVM) [23, 81], random forest [53], (4) structural such as decision tree [56, 59], graph models [54], and (5) fusions such as ensemble of classifiers [5, 7], template level [21], and score level [67, 71, 77].

1.1 Motivation and contribution of the proposed work

Despite the widespread use of signatures in the process of the verifying Identification of persons but as mentioned, there are many challenges that must be addressed. If query signatures are created by forgery with a lot of practice, it will increase the similarity of the

genuine and forgery class. This leads to an increase in the false acceptance rate (FAR) of signatures. Of course, in some cases, the genuine signatures are also mistakenly identified as forgery samples that mean false rejection rate (FRR).

Since the FRR and FAR criteria are in conflict with each other, we intend to the trade-off between them. Deep learning methods require a large number of signature samples for the training process and they are not of interest to us. Therefore, by focusing on handcrafted feature extraction and with the aim of increasing the recognition rate, we want to obtain several levels of features for each sample of the signature. Our methodology summarizes the following steps: (1) selecting candidate points in each signature image, (2) extracting component-oriented and pixel-oriented features in the candidate point regions, (3) fusion the corresponding features in the candidate points. Thus, major contributions of the proposed method are categorized as below:

1. Introducing the approach of fusion, the bag of corresponding feature in the candidate points of each signature based on standard deviation and variance (BoF-CPS and BoF-CPV).
2. Feature extraction based on types conventional features such as of texture, direction, entropy, and shape features in the regions of candidate points.
3. Focusing on improve the rate of correct positive responses (sensitivity) and the rate of incorrect positive responses (specificity).
4. Focusing on improve average error rate.

The rest of this paper is organized as follows: section 2 is dedicated to the literature review. In section 3, the proposed methodology of fusion of the corresponding features around candidate points are presented. Experimental results are explicated in section 4. Finally, in section 5 we provide conclusions and future works.

2 Literature review

The handwritten signature is one of the popular methods of identification and verification that has attracted many researchers in the last few years. As mentioned in the previous section handwritten signature has been studied in various aspects such as pre-processing, feature extraction, and classification. In the following, some related research in the OSV has been discussed.

Faiza et al., have proposed the HPFI parallel approach to fusion GLCM and geometric features. Besides that, they have introduced the SKcPCA that selects the best features for verifying [17]. Jaine et al., have used sCNN for OSV that is based on a custom shallow convolution neural network for sample learning. In this research, two datasets CVBLSig-V1 and CVBLSig-V2 have been provided in which 137 and 467 users have registered [33]. In another study, signature samples extracted based on geometric features and separated by the artificial neural network [34]. Okawa in [49] provided a single-template matching method for verifying online signatures that is based on time-series averaging and gradient boosting. In this method several multiple DTW distances with respect to multivariate time series are calculated. In [65] the method of feature extraction presented which base on global features (such as area, width, and height) as well as local features (such as centroid, distance, angle, and slope). Then the best features are selected using a genetic algorithm. Researchers in [32], have used the

Siamese neural network (SNN) to OSV. For this purpose, they have presented a CNN sub-network which has used statistical measurements to strengthen the generated vector. Hofemann et al. in [28], have used fixed-size method in representation learning sample of different sizes in the offline signature system. In this method, the spatial pyramid pooling used to resize the sample of the signature. In another study [26], an approach based on measuring the difference vector have been presented, which is generated according to the difference between the genuine signature of the writer and other signatures of the same writer, as well as a forgery sample. In this strategy, two collections of separators of signatures are considered: collection ‘D’ for signing the writers who enrolled in the system and collection ‘E’ along with different signatures for evaluation. The set of D have used only to learn signature representation features and set of E have used to feature extraction, which is used to test and classify signature of writers. Hofemann et al., have presented a deep neural network to learn the features of an independent writer structure and used it to obtain a feature representation model to train the dependent writer structure, and to classify it [25]. Soleimani et al., have provided a method based on similarity and heterogeneity between genuine and forged signatures in each class based on distance metrics as deep multitask metric learning (DMML). In this method, the features of HOG and DRT have been used [68]. In [48], has been presented a writer-dependent structure that uses the KAZE features and the Fisher vector. In this method, the samples are evaluated based on the SVM classifier. Zoei et al., presented writer-dependent structure that using features based on hierarchical dictionary learning and sparse coding. Besides that, verification phase has done using the threshold [83]. Serdouk et al. in [61] have used the gradient local binary patterns and longest run features for the handwritten signature verification system. Also, in this method, an artificial immune recognition system has been used.

According to the research literature, a suitable feature extraction method can improve the classification performance. Figure 2 shows the classification of feature extraction methods based on [16]. In general feature extraction techniques are applied global and local to images. In order to extract the informative features from the query and reference samples image. According to the research literature such as [65, 68], it is clear that the FAR and FRR criteria are in conflict with each other. In the proposed approach, we want to improve these two criteria, along with several others such as average error rate, accuracy and sensitivity. Therefore, the basis of our proposed method in the fusion stage is the features that are extracted from around the candidate points. In this way, by considering the details of baselines of each signature, multiple short feature vectors are generated for each candidate point. Multiple feature vectors are unified by the proposed approach (BoF-CPS and BoF-CPV) afterward normalized feature vectors imported to the classification stage to separate the genuine and forgery sample.

3 Proposed scheme

In this paper, a method is presented with a focus on feature fusion to verifying the handwritten signature system. For this purpose, several features are extracted in the range of candidate points. Feature extraction approach is based on the tracing of distribution of connected pixels around the candidate points in the signature image. In the area of candidate points, conventional features are extracted (which is described in section 3.2). In addition, the pixels within the radius of the candidate points are examined in depth by recursive segmentation, and in each sub-division created, local spatial features representation is addressed. The important point in

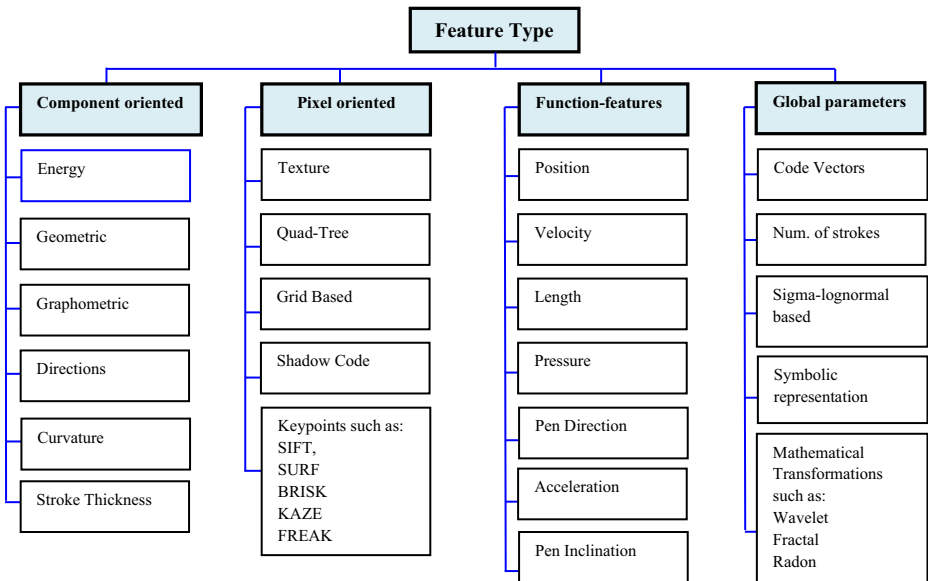


Fig. 2 Grouping types of features in the SV

the proposed approach is that the number of candidate points per image varies (because the nature of handwritten signatures varies for each writer). We propose a feature fusion approach for the corresponding feature vectors in the candidate points. Thus, the homogeneous features are fused separately for each candidate point to produce a normalized feature vector for each image. Normalization of feature vectors is performed based on the calculation of standard deviation and variance of homogeneous features of candidate points. (which is described in section 3.3). Figure 3 shows an overview of the proposed method.

In the following, each of the steps is explained:

3.1 Pre-processing phase

According to the state-of-the-art in this phase, initial steps can be taken to improve the images in the datasets. The choice of preprocessing techniques should be consistent with the feature extraction phase and lead to the extraction of the informative features in the signature samples. Hence, in the preprocessing stage, first, the gray level images are converted to binary, and after removing the noise, the signature baselines are thinned using morphological operators. The pre-processing steps are described below:

3.1.1 Convert to binary image

In the beginning, gray-level images are converted to binary images because methods of acquiring signature samples are different in various datasets and may be captured with different light intensities. In this regard, the Otsu method can be used [51]. Therefore, the thresholds are assessment in the range of [0 to 255]. And in each assessment, the amount of variance intra-class is calculated for the left and right status. This process is repeated until a threshold is found with the lowest value. Afterward considering a threshold value, the image

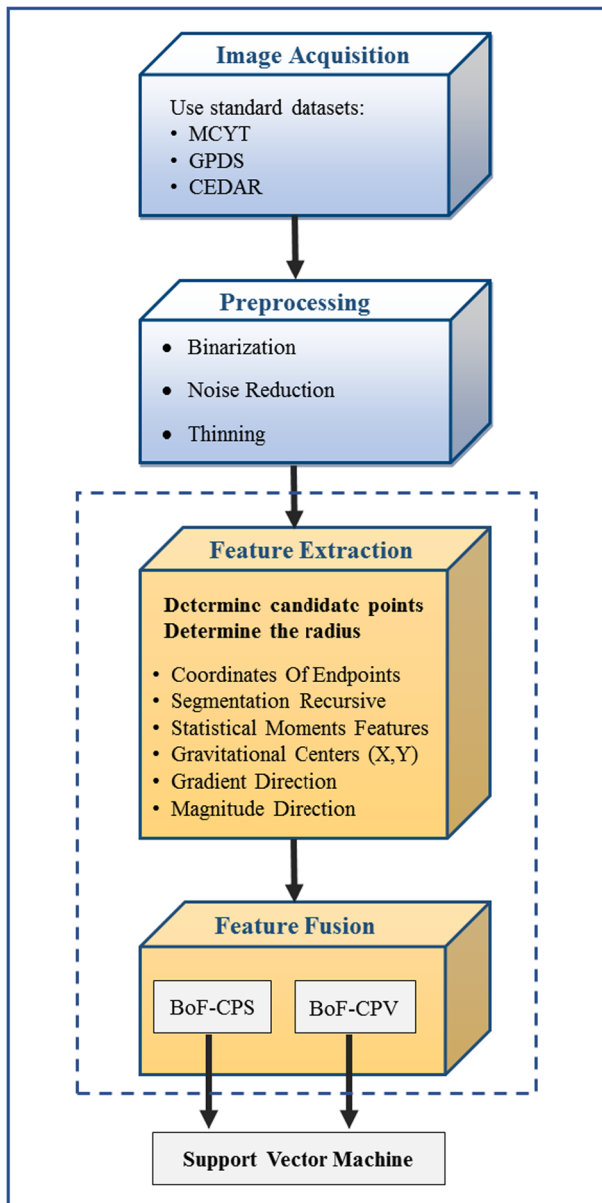


Fig. 3 Overview of the proposed methodology

histogram is divided into two background and foreground classes. The composition of the intra-class variance is calculated as follows [8, 31, 64]:

$$\sigma_w^2(t) = \omega_0(t)\sigma_0^2(t) + \omega_1(t)\sigma_1^2(t) \tag{1}$$

Weights ω_0 and ω_1 are the probabilities of classes that are separated by the t-threshold. Values of σ_1^2 and σ_2^2 are variance the above classes. Class probability $\omega_{0,1}(t)$ is calculated based on bins

of histogram.

$$\omega_0(t) = \sum_{i=0}^{t-1} p(i) \quad (2)$$

$$\omega_1(t) = \sum_{i=t}^{L-1} p(i) \quad (3)$$

The process of each class minimization is similar to maximization inter-class variance. This process is defined based on Eq. 4:

$$\begin{aligned} \sigma_b^2(t) &= \sigma^2 - \sigma_w^2(t) \\ &= \omega_0(\mu_0 - \mu_T)^2 + \omega_1(\mu_1 - \mu_T)^2 \\ &= \omega_0(t)\omega_1(t)[\mu_0(t) - \mu_1(t)]^2 \end{aligned} \quad (4)$$

That ω is the probability of class and μ indicates class. And the classes μ_0 , μ_1 and μ_T are calculated based on the following equations:

$$\mu_0(t) = \frac{\sum_{i=0}^{t-1} ip(i)}{\omega_0(t)} \quad (5)$$

$$\mu_1(t) = \frac{\sum_{i=t}^{L-1} ip(i)}{\omega_1(t)} \quad (6)$$

$$\mu_T = \frac{\sum_{i=0}^{L-1} ip(i)}{L} \quad (7)$$

3.1.2 Noise reduction

In the proposed method, after converting the signature images to binary images, the median filter is used to reduce noise. This filter represents the central pixel based on neighboring pixels as follows [77].

$$Sig_{(x,y)} = median\{sig_{bin}(s,t) \in neighboring_{pixel}\} \quad (8)$$

In the above equation sig_{bin} is the binary image, and $neighboring_{pixel}$ adjacent pixels in the neighborhood coordinates X and Y.

3.1.3 Thinning of signature lines

Due to the nature of the feature extraction approach in this research, thinning operations are necessary at this stage because the skeletal image is a compact representation of the samples of signature, the additional information of which has been reduced. In Fig. 4 how to thin baselines of one of signature sample is shown.

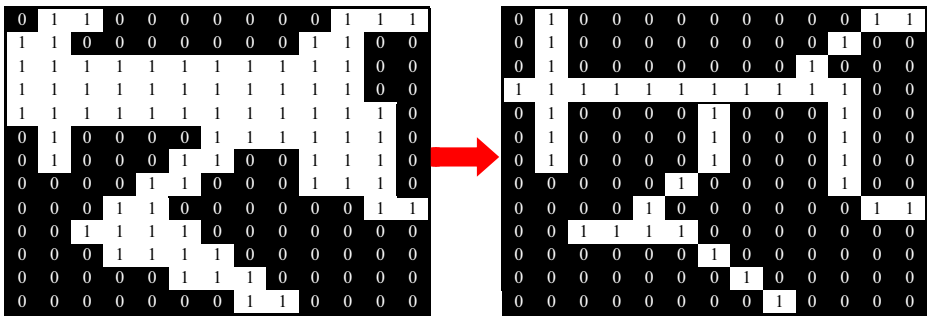


Fig. 4 Thinning the digital image pixels

To do this, we first expanded the image using morphological (dilation) operations to strengthen the weak connection of the pixels (otherwise, it will cause the signature baselines to discrete). Due to the output of the thinned image, the curves present in the pixel junctions and baselines are well represented, consequently, tracing the approximate paths of the central pixel and intersection points can be easily determined accordingly. Figure 5 shows the pre-processing operation on three samples of signature images.

3.2 Feature extraction phase

Assuming that in a signature verification system we consider the features of the genuine signature set as $G_k = \{g_{1,k}, g_{2,k}, \dots, g_{n,k}\}$ and the forgery signature set as $F_k = \{f_{1,k}, f_{2,k}, \dots, f_{n,k}\}$ while $G_k, F_k \in \mathbb{R}$ Which refers to the entire data set. Hence two feature vector N-dimensional are generated according to the genuine and forgery samples, according to which the samples will be separated. In Fig. 4 each image has background pixels with a value of zero and foreground

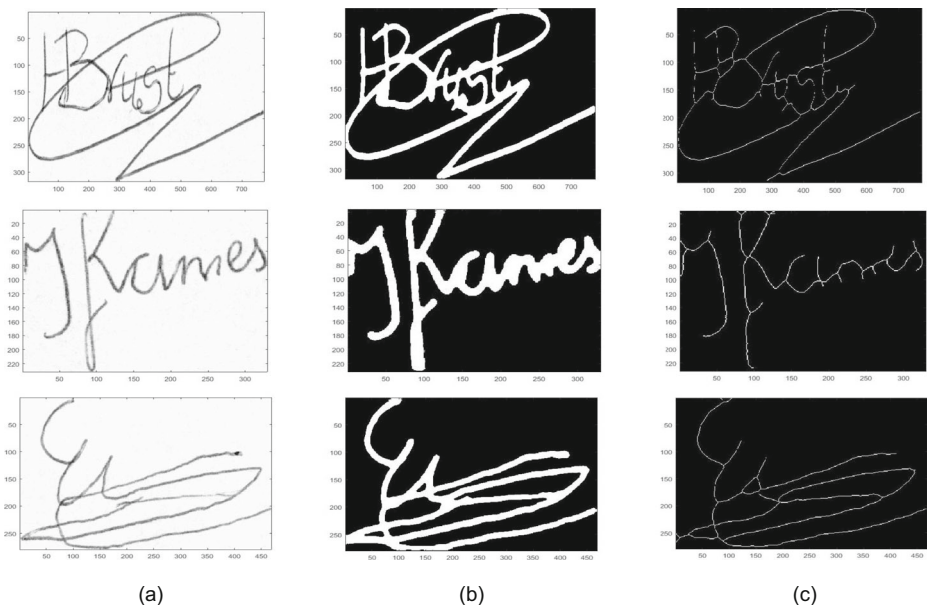


Fig. 5 a binary image, b Morphological operations to strengthen strokes, c Applying the thinning operator

pixels with value of one. Considering baseline pixels, different features of the image, such as area, signature box, and density of pixels, can be extracted.

In general, relationships between image pixels are examined based on neighborhood, proximity, connections, paths, and region. In this regard, each central pixel has 4 pixels are vertical and horizontal neighbors and 4 diagonals pixel of neighbors. The coordinates of the vertical and horizontal neighbors of the central pixel are as follows:

$$Pixel_{4(Ver,Hor)} = (i + 1, j), (i-1, j), (i, j + 1), (i, j-1) \tag{9}$$

The coordinates of the diagonal neighbors of the central pixel are as follows:

$$Pixel_{4(Diag)} = (i + 1, j + 1), (i + 1, j-1), (i-1, j + 1), (i-1, j-1) \tag{10}$$

Thus the 8-pixel neighbors of the central pixel with the coordinates (i, j) are obtained from the integration of horizontal, vertical, and diagonal neighbors. 8-neighbors of the central pixel are defined as follows:

$$Pixel_{8(Neighbor)} = (i + 1, j + 1), (i + 1, j-1), (i-1, j + 1), (i-1, j-1), (i + 1, j), (i-1, j), (i, j + 1), (i, j-1) \tag{11}$$

Figure 6 shows the arrangement of adjacent pixels relative to the center pixel.

In the proposed method, we intend to specify the central pixel based on the intersections of the pixels of the baselines (section 3.2.1 explains how to determine intersections pixels). In general, these points differ according to the shape and structure of the signature baselines in different datasets. Therefore, each signature sample can have multiple numbers of intersecting points. In our proposed method, these points are introduced as candidate points. After defined the candidate points, we are specifying the neighborhood range of each point according to the windows with radius r. In this case, the features extraction in the area surrounded is applied as $2r + 1$ around the candidate points as shown in Fig. 7. (We consider the value of the radius is equal to 2; this radius produces a 5×5 window). According [38] within each radius, there are a certain number of neighbors in the range $0 \leq n_r \leq 8r$. Hence, the relative distribution of pixels scattering along predetermined signature paths at different radii be tracing in this surrounded region.

In the following, how the definition of candidate points and the extraction of features around this region are explained.

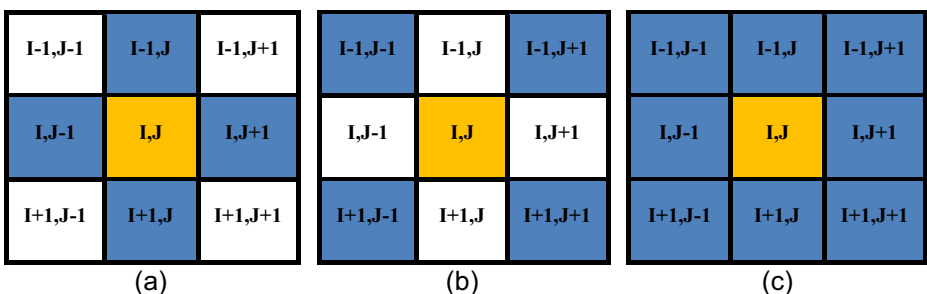


Fig. 6 a vertical and horizontal neighbor, b diagonal neighbors and c 8-pixel neighbors

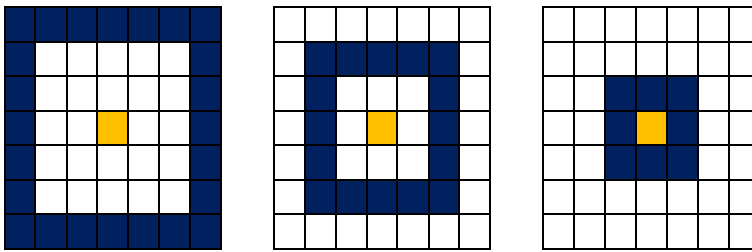


Fig. 7 Surrounded pixels with radius 3, 2 and 1 have highlighted in different colors

3.2.1 Determining candidate points

The majority of the instances in the database have lines that have crossed from each other. In the first step of the proposed method, we distinguish these points to determine the range of feature extraction. For example, in Fig. 8, the candidate points for a sample signature image are specified [8, 30].

The location of the intersection of pixels in baselines varies according to the shape and structure of the signature curve in different samples. So, the intersection points of image pixels in the horizontal, vertical and diagonal neighbor are considered as candidate points. In Fig. 9, the modes of the approximate paths of baselines of the candidate points within a radius of 2 are shown.

Given the local complexities of the signature baselines at the intersection points, it is necessary to determine the end position at the radius boundary for each sample of reference and query signatures are done.

3.2.2 Determining termination points

At this phase, the extraction of the ending points is performed to determine the final position of the signature pixel distribution. For this purpose, baselines of the signature image are tracked, and pixels with only one neighbor are specified as endpoints [41]. In this case, we consider local end pixels that lie along the boundary margin of the radius of the candidate points. Hence the last pixel at the border is introduced as the endpoint. In this case, the local endpoint has more than one neighboring pixel because the pixels within the radius have connected to the

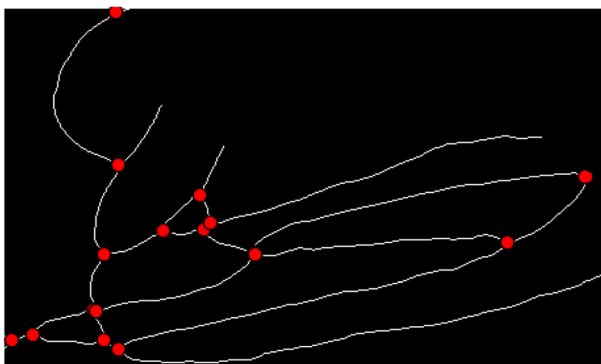


Fig. 8 Determine candidate points

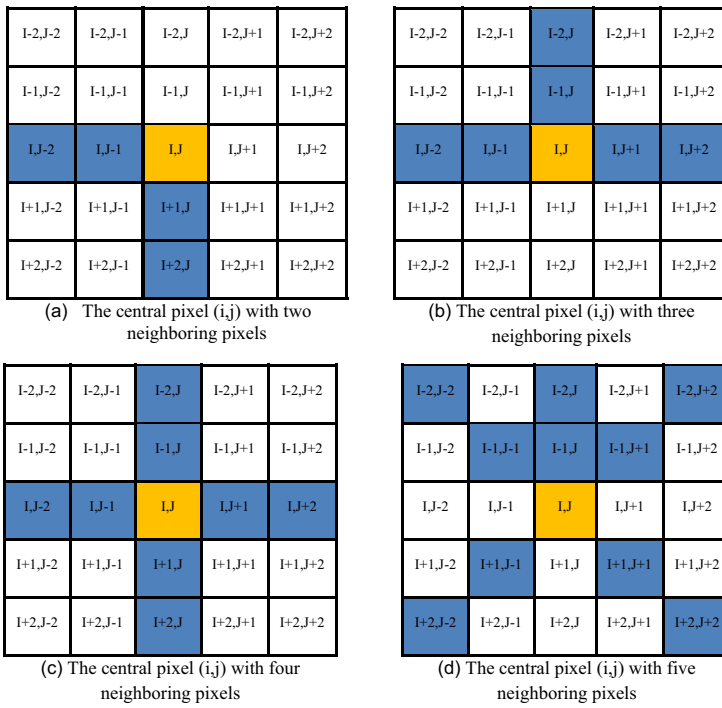


Fig. 9 Approximate paths of candidate points on signature baselines in radius 2. **a** The central pixel (i,j) with two neighboring pixels, **b** The central pixel (i,j) with three neighboring pixels, **c** The central pixel (i,j) with four neighboring pixels, **d** The central pixel (i,j) with five neighboring pixels

pixels outside the radius boundary. Figure 10 shows the local endpoints in some candidate points of sample of signature.

Finally, the positions of local endpoints are determined, and tracing of the connected pixels according to endpoints is investigated.

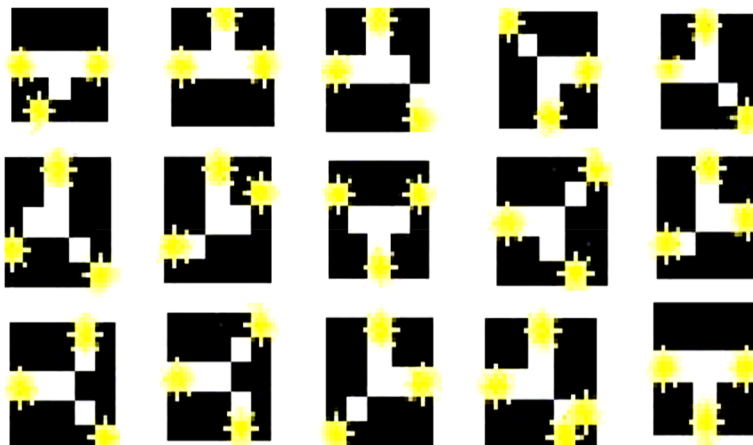


Fig. 10 Determine some local endpoints within the radius of the candidate points

3.2.3 Segmentation recursive in the radius of candidate pixels

This technique is similar to a quad-tree that divides quadratically evenly into a subregion and represents spatial features in each region. Of course, in our approach, partitioning is done by the centers of gravity at each candidate point in the signature image. Hence, due to the depth recursive partitioning locally at each point, the shape information of adjacent pixels is extracted. In descriptors that use recursive tree strategy (such as quad-tree or quin-tree), the feature extraction process applies to smaller sub-division of the image in each recursive implementation, and in each recursive implementation, spatial features are [75]. According to this strategy, the images are divided into four or five sub-division. If the N depth variable refers to the maximum tree produced and the maximum number of branches added in each iteration, is set to N_{Branch} , then the following image in i^{th} tree node at the specified depth (d) is:

$$Sub_{img}(N_{Depth}, N_{Branch}) = \sum_{d=0}^{N_D-1} (NB)^d \tag{12}$$

Where $d \in \{0, \dots, N_D - 1\}$, so the features in each observed node are computed [52]. The extracted features are redirected coarse to the fine surface at each step. In fact, by maintaining stability at higher levels, feature extraction is performed at lower levels. Given the function of the quad tree and given that the R area corresponds to the whole sample image of the signature and the P statement of the segmentation operation, if each region is $isP(R_i) = \mathbf{TRUE}$, R_i , the segmentation becomes progressively smaller. If the condition is $isP(R_i) = \mathbf{False}$, then the image is divided into four quadrants. In the next step, the condition is checked again and if the statement is false in each of the following quadrants, the steps are repeated for it. In the proposed approach, for the recursive partitioning, the centers of gravity of the candidate points are considered as roots and the properties of each candidate point are extracted as a recursive tree. Figure 11a shows how the tree splits recursively from the root node with the central coordinate of the candidate point and Fig. 11b shows the segmentation within the surrounded of candidate points.

According to Fig. 11a, the vector obtained from the candidate point decomposition as root to the lower surfaces is defined as follows:

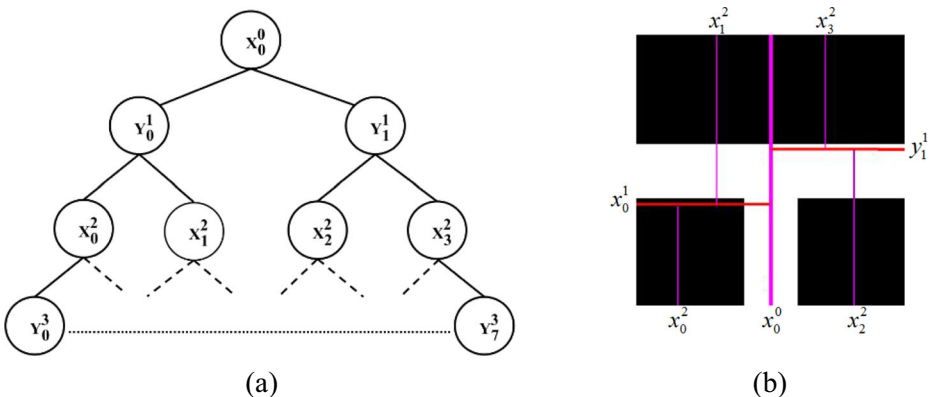


Fig. 11 a Extractive recursive feature on tree surfaces, b Segmentation in candidate pixel in the level 3

$$Sub_{regions} = (x_0^0, y_0^1, y_1^1, x_0^2, x_1^2, x_2^2, x_3^2, y_0^3, \dots, x_5^3, x_6^3, x_7^3) \quad (13)$$

Since the nature of the signature is such that it does not cover the entire signature box [62, 63], so in experiments, after several stages of evaluation, we considered the value of the segmentation level equal to 3 which leads to the division of the image into 8 sub-division. If we choose level value more than 3 in the candidate region may result in the formation of multitude null cells. Eventually, according to the value of level the numbers of features extracted in sub-division of candidate points in this methodology are equal to $2 * (2^{level} - 2)$.

According to the aforementioned above, segmentation of the candidate signature regions is done based on the gravity centers of each region. Besides that, we have considered gravity centers in the x and y coordinates separately as features. In general, the gravitational centers of the mean black pixels at the candidate points of the signature image are defined as follows:

$$CG_x = \frac{\sum_{i=1}^n x_i}{n} \quad (14)$$

$$CG_y = \frac{\sum_{i=1}^n y_i}{n} \quad (15)$$

Where n is the number of black pixels in the candidate points of the signature. x_i the value of the x coordinates for i^{th} black pixel in the candidate points. And y_i is the coordinate value of y for the i^{th} black pixel in the candidate points.

One of the limitations of this segmentation phase on our proposed method is that some sample of signatures may not have any intersection pixels. In these samples, we applied global recursive segmentation separately to obtain informative features. In Fig. 12 how to global equal-mass segmentation on the signature sample is shows.

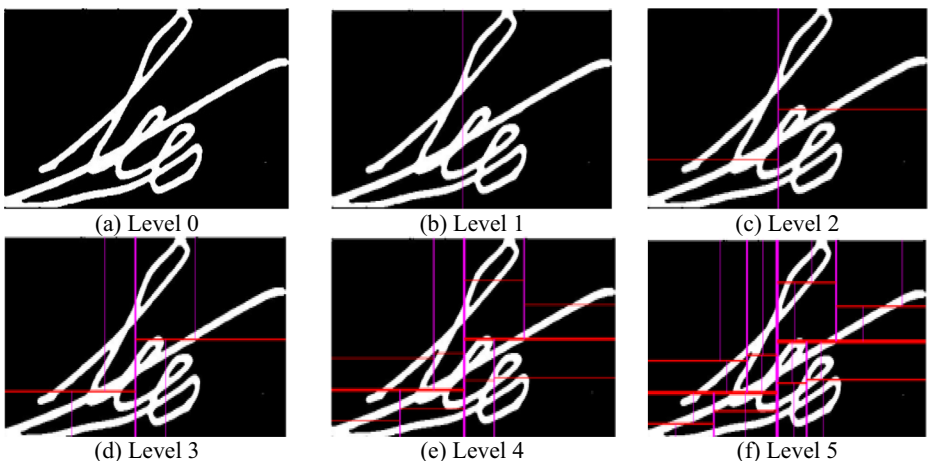


Fig. 12 Global segmentation in a sample image with different levels

3.2.4 Statistical moments features

One of the statistical methods used to describe the texture is to use statistical moments related to the intensity histogram in an image or region [45]. All candidate points of the signature image have gravitational centers according to which the complexity of adjacent pixels can be examined. So, using the statistical measure general and representative features such as the texture of the regions of each radius of the candidate points, the mean (m_1), variance (m_2), skewness (m_3) and kurtosis (m_4) can be described in candidate points region. Hence, we calculate the four moments of a probability distribution around candidate points of each instance as below [45].

$$\text{Mean or } \mu = E[x] = \sum_i x_i p(x_i) \quad (16)$$

$$\text{Variance or } \sigma^2 = E[x-\mu^2] = \sum_i (x_i-\mu)^2 p(x_i) \quad (17)$$

$$\text{Skewness or } \gamma_1 = E\left[\left(\frac{x-\mu}{\sigma}\right)^3\right] = \frac{E[x^3]-3\mu\sigma^2-\mu^3}{\sigma^3} \quad (18)$$

$$\text{Kurtosis or } k = E\left[\left(\frac{x-\mu}{\sigma}\right)^4\right] = \frac{4^{\text{th}} \text{ central moment}}{\text{variance}^2} \quad (19)$$

For more details, in the first-order moment, according to the central tendency, the shape is described. In the second-order moment, the histogram intensity contrast is calculated in the candidate point regions, and the relative smoothness of the signature baselines within the radius of the candidate points can be described. On the other hand, in the third-order moment, the skewness of the histogram in the radius of the candidate region is calculated. Besides that, this moment describes the degree of symmetry of the histogram in the candidate regions. The fourth-order moment describes the relative flatness of the histograms of the candidate points. Higher order moments distribute probability such as deviation from the mean.

3.2.5 Gradient direction and magnitude extraction

All baselines pixels with respect to the length of the adjacent pixels have horizontal, vertical, and diagonal direction. In this step, we extract the magnitude and direction of the pixels in the radius of surrounded candidate points. The direction feature extracts the slope and curvature of neighboring pixels within the radius of the candidate points in different directions. Since forgery signatures have a lot of vibration at the drawing epoch, so they have a lot of variations in the direction features of relative to the baselines of genuine signature. Figure 13 shows a perspective of the magnitude and direction feature extraction.

The gradient is used to find the direction of each pixel in the coordinates (x, y) and is defined following [40]:

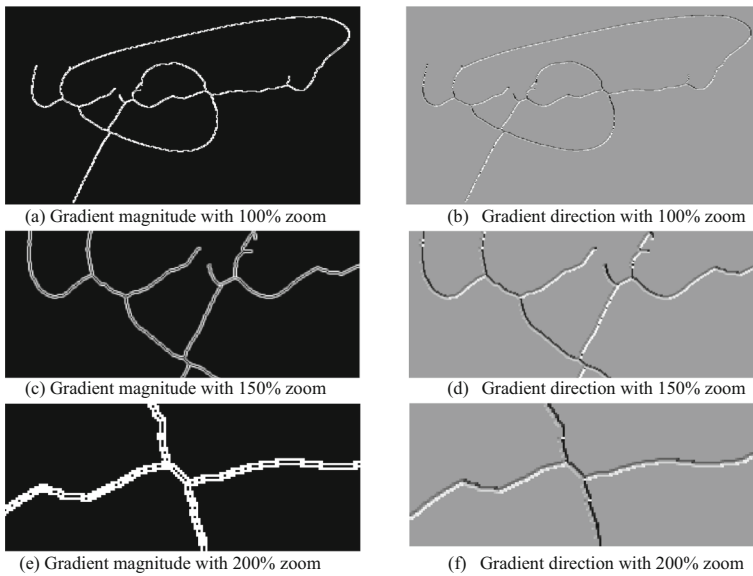


Fig. 13 Perspective of the magnitude and direction feature. **a** Gradient magnitude with 100% zoom, **b** Gradient direction with 100% zoom, **c** Gradient magnitude with 150% zoom, **d** Gradient direction with 150% zoom, **e** Gradient magnitude with 200% zoom, **f** Gradient direction with 200% zoom

$$\theta(x,y) = \tan^{-1} \begin{bmatrix} g_y \\ g_x \end{bmatrix} \tag{20}$$

The output of this feature contains the important geometrical features that have the highest variation rate of the gradient in the coordinates (x, y) [40, 69]. One of the requirements for calculating the gradient of the baseline of the signature is to calculate the partial derivatives. The partial derivative in the coordinate (x, y) is calculated as follows:

$$\begin{aligned} g_x &= \frac{\partial f(x,y)}{\partial x} = f(x+1,y) - f(x,y) \\ g_y &= \frac{\partial f(x,y)}{\partial y} = f(x,y+1) - f(x,y) \end{aligned} \tag{21}$$

Different masks such as Sobel can be used to apply gradients to images [69]. In order to remind the equation for the Sobel mask in the horizontal and vertical direction as follows:

$$\begin{aligned} Sig_x &= \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}, \quad Sig_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \\ Sig_x(i,j) &= I(i-1,j+1) + 2I(i,j+1) + I(i+1,j+1) - I(i-1,j-1) - 2I(i,j-1) - I(i+1,j-1) \\ Sig_y(i,j) &= I(i-1,j-1) + 2I(i-1,j) + I(i-1,j+1) - I(i+1,j-1) - 2I(i+1,j) - I(i+1,j+1) \end{aligned} \tag{22}$$

According to [69] we have used the mask Sobel to extract the gradient features of the signature samples.

3.2.6 Entropy extraction

The entropy feature measures the randomness of the covariance matrix elements within the radius of the candidate points. Hence, we used entropy measures to compute the randomness of the nature of the texture in candidate points, which indicates the amount of information of surrounded regions and is defined as. In Eq. (22) entropy and Eq. (23) mean entropy are defined [11, 24]:

$$\mathbf{entropy} = \sum_{i=1}^k \sum_{j=1}^k p_{ij} \log_2 p_{ij} \quad (23)$$

$$\mathbf{entropy}(z) = \sum_{i=0}^{L-1} p(z_i) \log_2 p(z_i) \quad (24)$$

We applied the entropy feature in radii 2 and 3 around the candidate points to calculate the probability density in these areas.

3.3 The proposed feature fusion phase

The proposed method focuses on feature vector fusion. Since the structure and geometric shape of the signature are different in the sample signatures of each database. This leads to the production of different numbers of candidate points. Since some of the hand-craft features have extracted within the radius of the candidate points, so the length of the feature vectors may vary in the database images. To overcome this issue, we proposed a methodology to unify the feature vector lengths by fusion the corresponding homogeneous features in the candidate points, based on the standard deviation and variance. In the proposed approach, the scatter of features around their mean is measured. For this purpose, the standard deviation and variance of the corresponding homogeneous features at the candidate points are calculated separately. We called the proposed method as BoF-CPS¹ and BoF-CPV² respectively. Figure 14 also presents the feature fusion algorithm given requirements applied in the preprocessing and feature extraction stages.

In this case, it is assumed that \mathbf{p} candidate points are found for each signature sample and \mathbf{n} features have been extracted for each point. For each candidate point, the features introduced in the section 3.2 are extracted, and thus multiple feature vectors are generated for each image according to the number of candidate points. Since features vectors must be the same length to apply to the classifier, we fusion homogeneous features at each candidate point separately. If that we have \mathbf{n} separate homogeneous features as f_1, f_2, \dots, f_n , then their standard deviation is defined by the symbol σ as follows [70]:

¹ Bag of Feature in candidate Points base on Standard deviation

² Bag of Feature in candidate Points base on Variance

Algorithm 1 Feature Fusion Approach

REQUIRE: Sig_{img}, the output of the preprocessing phase
DEFINE candidate points (CP) in each Sig_{img}
DEFINE of special radius to achieve a 5 * 5 mask on each CP
 Extracted features in section 3.2
Calculate the number of CP in each Sig_{img}
IMPOSE Feature vector for each CP
Indexing corresponding features in the FVs of each CP, **Assume that:**
 CP(1,1) = {f_{1,1}, f_{1,2}, f_{1,3},..., f_{1,n}}, CP(1,2) = {f_{2,1}, f_{2,2}, f_{2,3},..., f_{2,n}},
 CP(1,3) = {f_{3,1}, f_{3,2}, f_{3,3},..., f_{3,n}}, ..., CP(1,p) = {f_{p,1}, f_{p,2}, f_{p,3},..., f_{p,n}}
FOR i=1 to num_Sig_{img}
Calculate STD (f_{1,1}, f_{2,1}, f_{3,1},..., f_{p,1})-f₁, STD (f_{1,2}, f_{2,2}, f_{3,2},..., f_{p,2})-f₂
 STD (f_{1,3}, f_{2,3}, f_{3,3},..., f_{p,3})-f₃, ..., STD (f_{1,n}, f_{2,n}, f_{3,n},..., f_{p,n})-f_p
ASSIGN feature vector {f₁, f₂, f₃,..., f_p} as **BoF-CPS** for **Sig_{img}**
END
FOR i=1 to num_Sig_{img}
Calculate Var (f_{1,1}, f_{2,1}, f_{3,1},..., f_{p,1})-f₁, Var (f_{1,2}, f_{2,2}, f_{3,2},..., f_{p,2})-f₂
 Var (f_{1,3}, f_{2,3}, f_{3,3},..., f_{p,3})-f₃, ..., Var (f_{1,n}, f_{2,n}, f_{3,n},..., f_{p,n})-f_p
ASSIGN feature vector {f₁, f₂, f₃,..., f_p} as **BoF-CPV** for **Sig_{img}**
END
RETURN: Two separate normalized vectors **BoF-CPV** and **BoF-CPS**

Fig. 14 Algorithm of the corresponding features fusion in candidate points

$$\sigma = \sqrt{\frac{\sum_{i=1}^n (f_i - \bar{f})^2}{n}} = \sqrt{\frac{(f_1 - \bar{f})^2 + (f_2 - \bar{f})^2 + \dots + (f_n - \bar{f})^2}{n}} \tag{25}$$

We considered (f_i-f) as the standard deviation of ith features from the mean of the feature. On the other hand, variance is a statistical concept that examines how a random variable is distributed around the mean. In general, the variance is the sum of the data distances from the center, which is defined in a statistical population as follows [70]:

$$\sigma^2 = \frac{\sum_{i=1}^n (f_i - \mu)^2}{N} \tag{26}$$

In this case we assume f_i represents i features and μ indicates the average statistical population and N data values in the statistical sample. Therefore, the calculation of variance in a corresponding feature is as follows:

$$S^2 = \frac{\sum_{i=1}^n (f_i - \bar{f})^2}{N-1} \tag{27}$$

According to the above-mentioned f indicates the average of the sample and N-1 refers to the statistical population in which the sample has been extracted. According to the dependence of the variance and the standard deviation, the standard deviation can be defined as follow:

$$\sigma = \sqrt{\sigma^2} \tag{28}$$

In our proposed method the calculation of the standard deviation in the fusion of the corresponding homogeneous features in the candidate points of an image is performed as follows:

$$\left\{ \begin{array}{l} \mathit{CandidateP}_{img1}^1 \mathit{inputFeature} = f^1_{1(n^*m)}, f^1_{2(n^*m)}, \dots, f^1_{n(n^*m)} \\ \mathit{CandidateP}_{img1}^2 \mathit{inputFeature} = f^2_{1(n^*m)}, f^2_{2(n^*m)}, \dots, f^2_{n(n^*m)} \\ \dots \\ \mathit{CandidateP}_{img1}^P \mathit{inputFeature} = f^P_{1(n^*m)}, f^P_{2(n^*m)}, \dots, f^P_{n(n^*m)} \end{array} \right.$$

$$\begin{array}{ccc} \Downarrow & \Downarrow & \Downarrow \\ \mathit{feature}_1 = \mathit{STD}[f^1_{1(n^*m)}, f^2_{1(n^*m)}, \dots, f^P_{1(n^*m)}] = f_1 \\ \mathit{feature}_2 = \mathit{STD}[f^1_{2(n^*m)}, f^2_{2(n^*m)}, \dots, f^P_{2(n^*m)}] = f_2 \\ \mathit{feature}_p = \mathit{STD}[f^1_{p(n^*m)}, f^2_{p(n^*m)}, \dots, f^P_{p(n^*m)}] = f_p \end{array}$$

$$\Rightarrow \mathit{fV}_{img1} = [\mathit{feature}_1, \mathit{feature}_2, \dots, \mathit{feature}_p]$$

In the above equation $\mathit{CandidateP}_{img1}^1$ is the first candidate point in the first sample of signature and $\mathit{CandidateP}_{img1}^2$ is the second candidate point in the first sample of signature. Similarly, $\mathit{CandidateP}_{img1}^p$ is the final candidate point in the first sample of signature. f^2_1 is the first feature of the second candidate point. f^2_n is the n^{th} feature of the second candidate point. If the standard deviation of the set of features obtained has a small value, it means that the distribution of features around their average is low, to put it simply meaning that the features are close together. On the other hand, if the standard deviation of a set of features is a large value, it indicates a high distribution of the value of the features and their distance from each other. The calculation of the variance of the corresponding homogeneous features in the

candidate points for each sample of signature is done as follows:

$$\left\{ \begin{array}{l}
 \text{Candidate}_{img1}^1 \text{ inputFeature} = f^1_{1(n^*m)}, f^1_{2(1n^*m)}, \dots, f^1_{n(n^*m)} \\
 \text{Candidate}_{img1}^2 \text{ inputFeature} = f^2_{1(n^*m)}, f^2_{2(n^*m)}, \dots, f^2_{n(n^*m)} \\
 \dots \\
 \text{Candidate}_{img1}^P \text{ inputFeature} = f^P_{1(n^*m)}, f^P_{2(n^*m)}, \dots, f^P_{n(n^*m)}
 \end{array} \right.$$

$$\begin{array}{ccc}
 \Downarrow & \Downarrow & \Downarrow \\
 \text{feature}_1 = \text{VAR}[f^1_{1(n^*m)}, f^2_{1(n^*m)}, \dots, f^P_{1(n^*m)}] = f_1 \\
 \text{feature}_2 = \text{VAR}[f^1_{1(n^*m)}, f^2_{1(n^*m)}, \dots, f^P_{1(n^*m)}] = f_2 \\
 \text{feature}_p = \text{VAR}[f^1_{1(n^*m)}, f^2_{2(n^*m)}, \dots, f^P_{n(n^*m)}] = f_p \\
 \Rightarrow fV_{img1} = [\text{feature}_1, \text{feature}_2, \dots, \text{feature}_p]
 \end{array}$$

In this way, the corresponding multidimensional features of the candidate points are transformed into a unified vector. Finally, the vector of the produced features is applied separately to the classification and the final results are compared.

4 Experimental protocols

In this method, evaluation protocols set as follows: (1) Offline signature verification system is presented based on writer-dependent (WD) approach to separate the genuine and forgery samples. (2) K-Fold cross-validation is performed on datasets to testify the performance of proposed method in evaluation criteria where k = 10, in this way, the average overall performance is presented in the results. (3) We used the classification support vector machine with RBF kernel. (4) All experiments were performed on an intel core i5 machine with a frequency of 2.50 GHz and 6 GB RAM in MATLAB environment.

4.1 Classification phase

The proposed signature verification system separates the genuine and forgery sample based on two-class Support vector machine (SVM) on feature vectors. Support vector machine is known as a desirable way to solve optimization problems. This method is a kernel-based machine learning model that has received much attention in segmentation and regression. Considering a binary classifier, the two general categories are defined [10]:

$$(x_1, y_1), (x_2, y_2), \dots (x_n, y_n) \tag{31}$$

Where is $X = \{x_i, y_i\}_{i=1}^n$, $x_i \in R^d$ and $y_i \in (+1, -1)$. The optimal separator hyper-plane is thus defined as follows:

$$y = \text{sign}[w^T \varphi(x^i) + b] \tag{32}$$

The above equation is solved by QPP terms:

$$\min_{w,b} (w) = \frac{1}{2} w w^T + c \sum_{i=1}^n \xi_i S.T : y_i [w^T \varphi(x_i) + b] \geq 1 - \xi_i \tag{33}$$

Generally, the optimal decision function is defined as follows:

$$y_i = \text{sign} \left(\sum_{i=1}^n \alpha_i y_i K \langle x_i, x_j \rangle + b \right) \tag{34}$$

α is Lagrange coefficient α for each point of learning. When the maximum margin of a hyper-plane is set, only the position of the cloud page is set to a value of $\alpha > 0$, which is known as support vector [10].

4.2 Performance evaluation

In the handwritten signature verification system, different criteria can be used, all of which are based on the confusion matrix. Table 1 is explained some of the important criteria in the experiments [8].

4.3 Dataset

We evaluated the proposed approach in this research on three standard databases MCYT database [50], GPDS [18] and CEDAR [36].

4.3.1 MCYT-75

In general, there are 75 signature samples from different users in the MCYT-75³ database. For each signature, there is a set consisting of 15 genuine samples and 15 forgery samples. In general, MCYT-75 database consists of 2250 signature images. In the Fig. 15 shown three samples of signature in the genuine and forgery groups from the MCYT-75 dataset.

4.3.2 GPDS-synthetic

In the GPDS⁴ database there are 4000 samples of signatures from different users and each sample has 24 genuine samples and 30 forgery samples. In total, GPDS database consists of 216,000 signature images that we randomly selected 100 samples for training and testing from this database. In the Fig. 16 shown three samples of signature in the genuine and forgery groups from the GPDS dataset.

³ Available on: <http://atvs.ii.uam.es/atvs/mcvt75so.html>

⁴ Available on: <http://www.gpds.ulpgc.es/download>

Table 1 Some of the important evaluation criteria used in the experiments

Symbols	Description	Mathematical Equation
TPR	Where TP is equal to true positive and FN is equal to false negative. This measure is also known as sensitivity also. This criterion identifies the class of positive samples. On the other hand, it determines to what extent the system has been successful in classifying forgery samples. Therefore, the number of genuine signatures considered to be falsely false has no effect on the sensitivity parameter calculation.	$TPR = \frac{TP}{(TP+FN)}$
FAR	False-Acceptation rate criterion which means the sample rate of forgery signatures identified as the genuine sample. Where FP equals false positive and TN equals true negative.	$FAR = \frac{FP}{(FP+TN)}$
FRR	False-rejection rate criterion, which means the rate of incorrect rejection of a sample that, is genuine.	$FRR = \frac{FN}{(TP+FN)}$
TNR	The rate of the true negative response, this criterion represents forgery samples that have been correctly identified as a forgery sample. This criterion, also known as specificity, addresses the negative class accuracy.	$TNR = \frac{TN}{(FP+TN)}$
AER	Average error rate criterion which means optimal decision threshold based on false-negative rate and false acceptance rate.	$AER = \frac{FNR+FAR}{2}$
ACC	Accuracy criterion, which means measuring system errors, based on which the overall performance of the system is evaluated.	$ACC = \frac{(TP+TN)}{(TP + TN + FP + FN)}$
F1	The F1 means the harmonic mean of the true positive rate and accuracy. This criterion is used to evaluate classification performance.	$F_1 = \frac{2TP}{(2TP+FN+FP)}$
Kappa	This criterion based on P_o and P_e . The observed probability (P_o) criterion examines the differences in evidence obtained between the observed probability and what is probability expected (P_e).	$kappa = \left(\frac{P_o - P_e}{(1 - P_e)} \right)$

4.3.3 Cedar

The CEDAR⁵ database includes 55 samples of signatures from different users. Each sample covering 24 genuine samples and 24 forgery samples. In total in this database, there are 2640 ~ different signature samples. Figure 17 shows some examples of CEDAR data sets.

4.4 Experimental results and discussion

In the following, the results of the simulation of the signature system in the fusion approach of the corresponding feature vectors in the candidate points of the samples of signature provided. According to the achieved results, the BoF-CPS approach is very useful in minimizing the dimensions of the feature. In this case, the aggregation of the corresponding homogeneous features is calculated based on the standard deviation. To achieve accurate evaluation, we compare the results of BoF-CPS and BoF-CPV approaches presented in this research with three widely used algorithms in the field of feature extraction. In this stage of evaluations, the features of the Haralick matrix, Gray Level Co-Occurrence Matrix (GLCM), and Histogram of Oriented Gradients (HOG) algorithms are manually combined.

$$Classic_{Manual\ Fusion} = \sum_{i=1}^3 \{Haralick_{1*14}, GLCM_{1*22}, HOG_{1*81}\} \quad (35)$$

⁵ Available on: <http://www.cedar.buffalo.edu/NIJ/data>

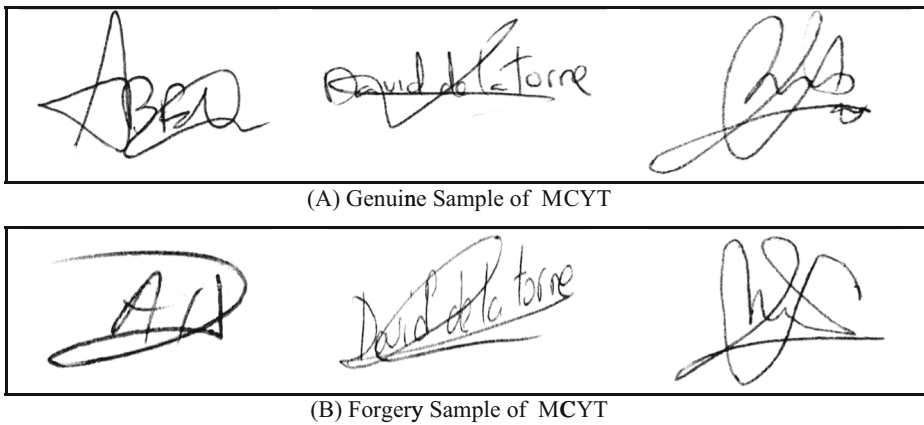


Fig. 15 Signature samples in MCYT Dataset. **A** Genuine Sample of MCYT, **B** Forgery Sample of MCYT

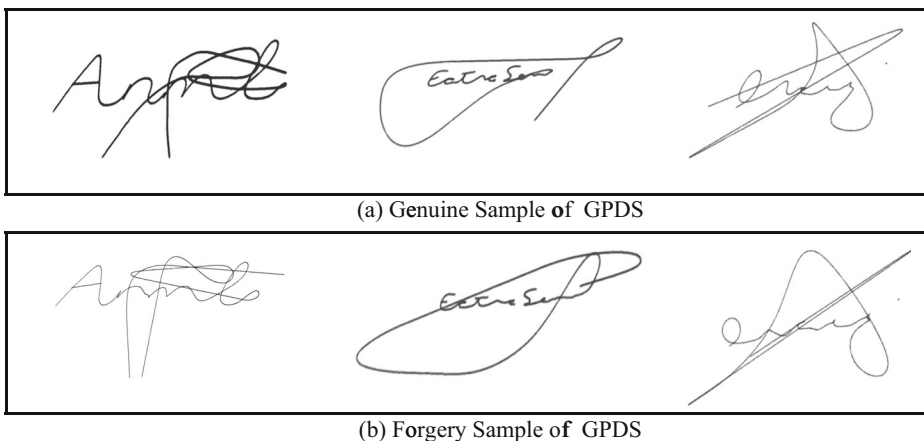


Fig. 16 Signature samples in GPDS Dataset. **a** Genuine Sample of GPDS, **b** Forgery Sample of GPDS

Afterwards, to evaluate the results, the feature vectors obtained from classical algorithms are combined with the BoF-CPS approach. In Fig. 18 the confusion matrix is shown for four methods aforementioned in the database MCYT, GPDS_{synthetic}, and CEDAR databases.

As previously mentioned, in the offline signature verification system, different criteria can be used which are based on the confusion matrix. Tables 2, 3 and 4 the results of several important evaluation criteria based on tests of four⁶ types of feature vectors provide. It is worth noting that the verification time for method implies the time required to classify signature samples by classifier SVM.

According to the results obtained in the above table, the accuracy criterion in the GPDS database is in the best case for the first approach and the combination of this approach with classical algorithms, which is equal to 98%. The GPDS database has more samples than the

⁶ The two proposed fusion methods (BoF-CPS and BoF-CPV) and two comparison steps according to the combination of classical algorithms (Haralick+GLCM+HOG) without considering the proposed BoF-CPS method and once along considering the features obtained by BoF-CPS method have done.

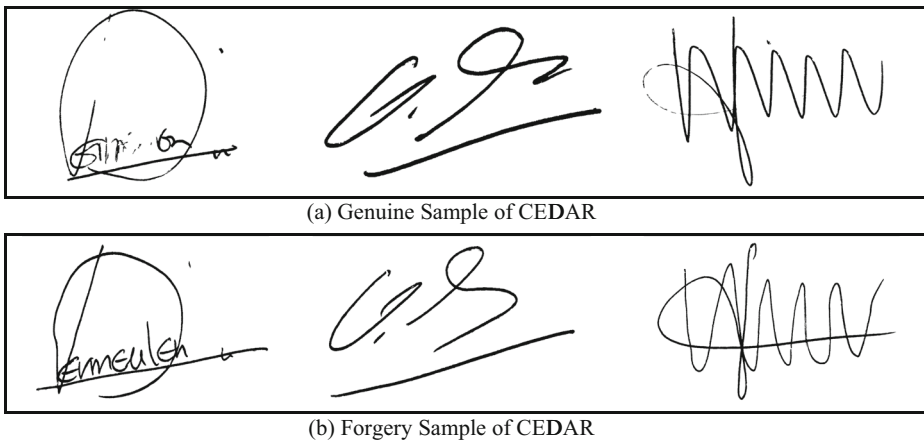


Fig. 17 Signature samples in CEDAR Dataset. **a** Genuine Sample of CEDAR, **b** Forgery Sample of CEDAR

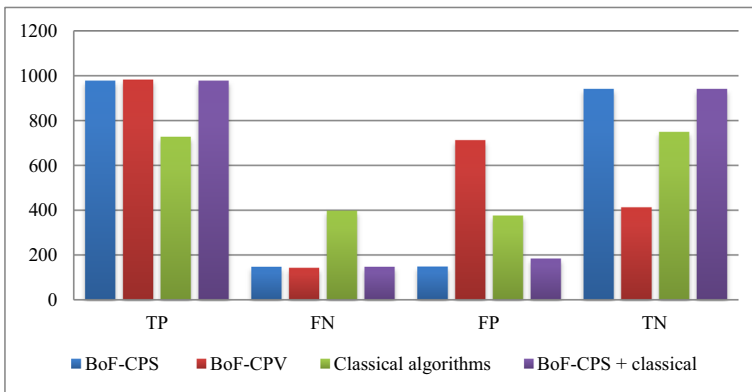
other two datasets, so the results indicate that the number of samples affects the accuracy rate. On the other hand, the highest precision for the BoF-CPS approach in the GPDS database is 98%. Then the results of this criterion in BoF-CPV and BoF-CPS + classical approach with equal value of 97% have been obtained. The results have been obtained in the criterion average error rate at best 0.01 and 0.03. If a lower value is obtained for this criterion, it indicates that the method used was more desirable.

According to the results of the FAR criterion in the MCYT dataset, it was obtained in the best case for the BoF-CPS approach, which by combining feature vectors related to classical algorithms has led to the same results. Besides, the presented BoF-CPS approach in GPDS and CEDAR datasets has achieved optimal results in the FAR criterion. Another important criterion is FRR, the values of which have the best result in the proposed BoF-CPS approach. According to the results, sensitivity and specificity criteria have achieved the best level of results in the proposed approach. Besides that, the combination of this approach with classical algorithms can provide an acceptable result. Table 4 shows the results of the evaluations in the observed probability criterion and the expected probability. In fact, these criteria examine the differences between the evidence obtained between the observed pattern and what is expected [8]. Based on these two criteria, the kappa coefficient is calculated.

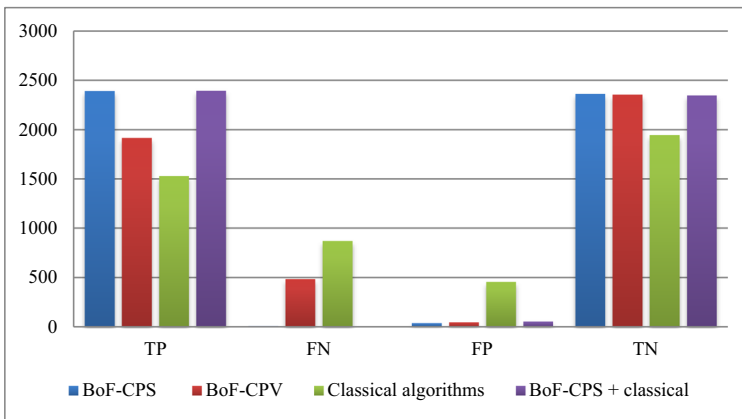
In Fig. 19 the results of FAR and FRR criteria respectively in (a) BoF-CPS approach, (b) BoF-CPV approach (c) the classical algorithms without considering the proposed method, and (d) classical algorithms+BoF-CPV features shown graphically.

Besides that, for further experiments, we evaluated the proposed method compared to a number of popular feature extraction methods in the field of biometrics. All comparisons have been performed under the uniform conditions and the results of the comparisons are shown in Table 5. In Figs. 20 and 21 the graphical representation of the accuracy and average error rate criteria for the aforementioned methods, respectively are shown.

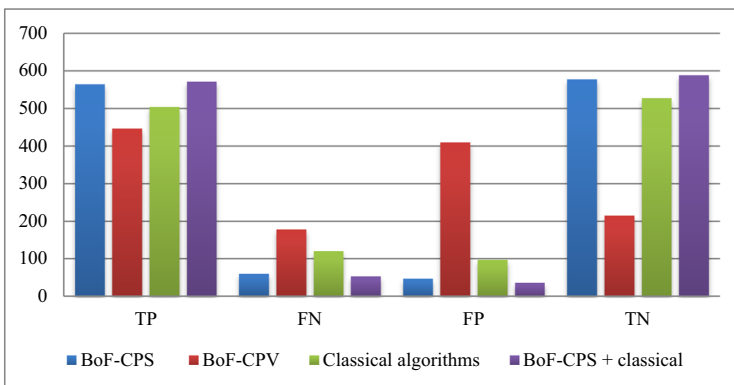
According to the background of the literature, the methods presented in offline signature verification systems have different data sets, test protocols, evaluation criteria and different validation methods. Therefore, direct comparison of the obtained results with each other does not seem appropriate [16, 27, 80]. However, in Table 6 a direct comparison of the results presented in the related articles is used.



(a) The graphical result of confusion matrix on MCYT dataset



(b) The graphical result of confusion matrix on GPDS dataset



(c) The graphical result of confusion matrix on CEDAR dataset

Fig. 18 Confusion matrix in evaluation methods. **a** The graphical result of confusion matrix on MCYT dataset, **b** The graphical result of confusion matrix on GPDS dataset, **c** The graphical result of confusion matrix on CEDAR dataset

Due to the wide range of experiments performed on feature vectors, it can be concluded that the BoF-CPV approach effectively improves classification performance; because the

Table 2 Results of criteria accuracy, precision, F_1 , average error rate and verification time on datasets

Dataset	Method	Accuracy	Precision	F_1	AER	Verification Time (s)
MCYT	BoF-CPS (Proposed)	85%	84%	58%	0.14	05.61
	BoF-CPV (Proposed)	62%	57%	69%	0.38	06.94
	Classical algorithms (VS)	65%	65%	65%	0.30	06.08
	BoF-CPS+classical (VS)	85%	84%	85%	0.14	07.60
GPDS	BoF-CPS (Proposed)	98%	98%	99%	0.03	08.71
	BoF-CPV (Proposed)	88%	97%	87%	0.11	27.70
	Classical algorithms (VS)	72%	77%	69%	0.27	23.35
	BoF-CPS+classical (VS)	98%	97%	98%	0.012	25.04
CEDAR	BoF-CPS (Proposed)	91%	92%	91%	0.08	04.74
	BoF-CPV (Proposed)	52%	52%	60%	0.47	05.50
	Classical algorithms (VS)	82%	83%	82%	0.17	05.69
	BoF-CPS+classical (VS)	92%	94%	92%	0.07	03.80

corresponding homogeneous features in the radius of the candidate points are strongly correlated with each other. On the other hand, the fusion of the corresponding homogeneous features based on the BoF-CPS approach performs better than the BoF-CPV approach. In general, according to the statistical results, the proposed method of feature extraction and fusion of feature vectors has been able to separate the samples at high speed and accuracy.

5 Conclusions and future work

Our proposed method focuses on the fusion of feature vectors. For this purpose, the corresponding homogeneous features in the candidate points are normalized based on the standard deviation and variance. Feature vector fusion is done because in our proposed approach the extracted properties are around the signature candidate points and the number of these points in each sample is different. The basis of our idea was that ideally the distinguishing features should reflect the process of how to produce signature samples, which is not the case in offline acquisition due to the use of scanners in the collection phase. For this purpose, the feature extraction phase has been done with this assumption that by examining the relationships of connected pixels that are adjacent to certain points of the signature the shape of the signature can be well described. One of the key processes is the pre-processing phase that thinning out

Table 3 Results of criteria FAR, FRR, specificity, and sensitivity on datasets

Dataset	Method	FAR	FRR	Specificity	Sensitivity
MCYT	BoF-CPS (Proposed)	0.16	0.13	83%	86%
	BoF-CPV (Proposed)	0.63	0.12	36%	87%
	Classical algorithms (VS)	0.33	0.35	66%	64%
	BoF-CPS+classical (VS)	0.16	0.13	83%	86%
GPDS	BoF-CPS (Proposed)	0.06	0.003	93%	99%
	BoF-CPV (Proposed)	0.01	0.20	98%	79%
	Classical algorithms (VS)	0.18	0.36	81%	63%
	BoF-CPS+classical (VS)	0.15	0.19	84%	80%
CEDAR	BoF-CPS (Proposed)	0.07	0.09	92%	90%
	BoF-CPV (Proposed)	0.65	0.28	34%	71%
	Classical algorithms (VS)	0.15	0.19	84%	80%
	BoF-CPS+classical (VS)	0.05	0.08	94%	91%

Table 4 Results of criteria kappa, OP, and EP

Dataset	Method	Kappa Coefficient	Observed Probability	Expected Probability
MCYT	BoF-CPS (Proposed)	0.47	0.61	0.2
	BoF-CPV (Proposed)	0.24	0.62	0.5
	Classical algorithms (VS)	0.31	0.65	0.5
	BoF-CPS+classical (VS)	0.70	0.85	0.5
GPDS	BoF-CPS (Proposed)	0.98	0.99	0.5
	BoF-CPV (Proposed)	0.77	0.88	0.5
	Classical algorithms (VS)	0.44	0.72	0.5
	BoF-CPS+classical (VS)	0.97	0.98	0.5
CEDAR	BoF-CPS (Proposed)	0.82	0.91	0.5
	BoF-CPV (Proposed)	0.05	0.52	0.5
	Classical algorithms (VS)	0.65	0.82	0.5
	BoF-CPS+classical (VS)	0.85	0.92	0.5

the pixels of signature samples. Hence, the topological skeleton of an image provides a compact representation of the signature image which according to it, baselines of the signature can be traced well. After that, the candidate points in the signature lines have been determined, which are obtained based on the intersection of the pixels in different directions. Most previous methods used a sliding window on the signature pixels to extract the features. While in the proposed method, while identifying the intersection points and introducing them as candidate

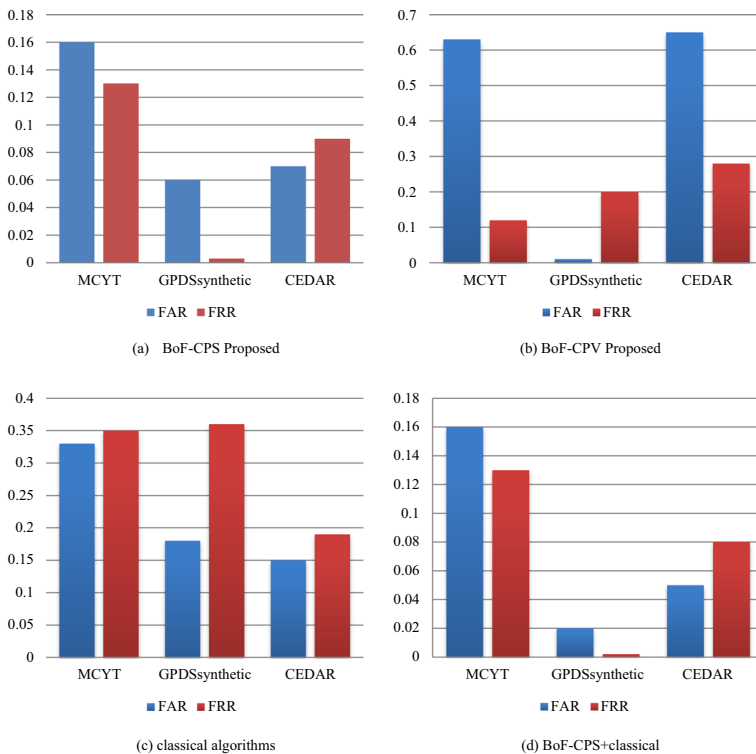


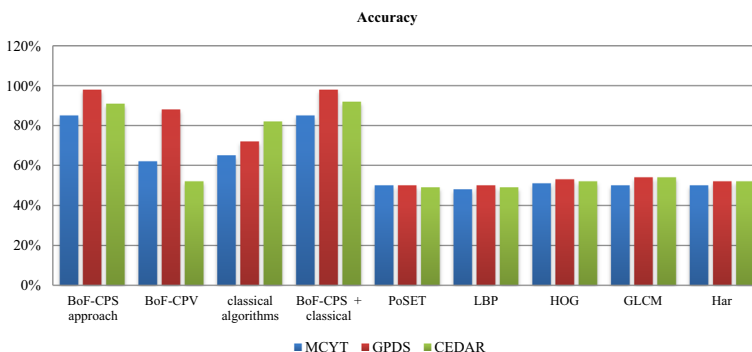
Fig. 19 Results of FAR and FRR criteria. **a** BoF-CPS Proposed, **b** BoF-CPV Proposed, **c** classical algorithms, **d** BoF-CPS + classical

Table 5 Comparison of the results of the proposed method with some feature extraction methods

Dataset	Feature Extraction method	ACC	AER	FAR	FRR
MCYT	BoF-CPS (proposed)	85%	0.14	0.16	0.13
	BoF-CPV (proposed)	62%	0.38	0.63	0.12
	Classical algorithms	65%	0.30	0.33	0.35
	BoF-CPS+classical	85%	0.14	0.16	0.13
	PoSET ¹	50%	0.5	0.50	0.49
	LBP	48%	0.51	0.42	0.61
	HOG	51%	0.4	0.24	0.72
	GLCM	50%	0.4	0.94	0.05
	Haralick	50%	0.4	0.99	0.0027
GPDS	BoF-CPS (proposed)	98%	0.03	0.06	0.003
	BoF-CPV (proposed)	88%	0.11	0.01	0.20
	Classical algorithms	72%	0.27	0.18	0.36
	BoF-CPS+classical	98%	0.01	0.15	0.19
	PoSET	50%	0.49	0.59	0.40
	LBP	50%	0.50	0.30	0.70
	HOG	53%	0.46	0.25	0.66
	GLCM	54%	0.45	0.56	0.34
	Haralick	52%	0.47	0.62	0.32
CEDAR	BoF-CPS (proposed)	91%	0.08	0.07	0.09
	BoF-CPV (proposed)	52%	0.47	0.65	0.28
	Classical algorithms	82%	0.17	0.15	0.19
	BoF-CPS+classical	92%	0.07	0.05	0.08
	PoSET	49%	0.50	0.83	0.10
	LBP	49%	0.50	0.20	0.80
	HOG	52%	0.47	0.65	0.30
	GLCM	54%	0.45	0.66	0.25
	Haralick	52%	0.47	0.78	0.16

¹ Available on: <https://github.com/ezois/Poset>

points, features are extracted in these areas. The proposed approach locally extracts the correlations and informative features of the pixels in the neighborhood of the candidate points. Therefore, considering how the arcs are distributed and the curvature of the pixels in the boundary radius of the candidate points, the potential paths of each stroke are well identified. Differences like the geometric structure of different signatures lead to the production of multi-dimensional feature vectors. In this step, for each signature sample, a number of short feature vectors are created that are around the candidate points. Hence, focusing on the fusion of

**Fig. 20** Comparison of accuracy criterion results in different methods

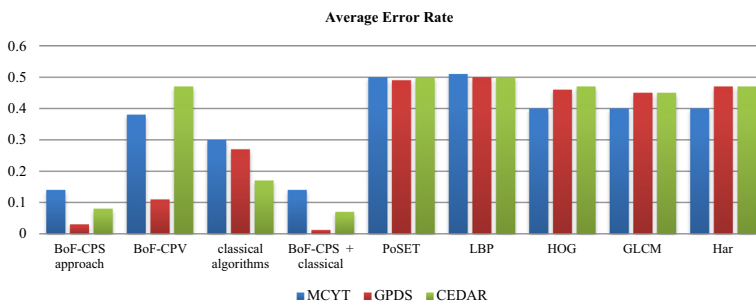


Fig. 21 Comparison of average error rate criterion results in different methods

feature vectors, we normalized the corresponding short feature vectors for each candidate point in the signature image. In this step, we implemented two approaches to fusion. First, the short feature vectors are fused by measuring the standard deviation of the corresponding homogeneous features in the short vectors. Which we call BoF-CPS. In the second approach, the corresponding homogeneous features in short vectors are fused based on the calculation of variance of homogeneous features which we call BoF-CPV. Then the feature vectors obtained from BoF-CPS and BoF-CPV approaches are applied to the classifier SVM separately and their results are evaluated. According to the experimental results, as we expected, the normalization of homogeneous features by maintaining local correlations in the radius of the candidate points has been able to provide the vector of distinguishing features. This method is very effective for equalizing the lengths of multiple feature vectors that are generated in different regions of the signature image. The proposed method has been evaluated on three standard MCYT, GPDS_{synthetic}, and CEDAR databases. According to the results, the average error rate in the MCYT database is 0.14, in the GPDS_{synthetic} database is 0.03 and in the CEDAR database is 0.08, which is a significant decrease compared to previous researches. Also, according to the results, the accuracy values for the above databases have achieved 85%,

Table 6 Comparison of the results of the proposed method with previous works

Reference#	Dataset	FAR	FRR	AER	Accuracy
[60]	GPDS	9.61	8.29	9.14	–
[60]	CEDAR	1.36	2.04	1.71	–
[61]	CEDAR	4.93	2.12	3.54	–
[61]	GPDS	13.16	11.38	12.52	–
[40]	CEADR	89.04	91.84	–	90%
[40]	GPDS	82.76	89.21	–	85%
[37]	CEDAR	11.23	12.39	–	88%
[68]	CEDAR	11.07	6.40	–	93%
[8]	MCYT	25.8	24.7	–	74%
[38]	GPDS	13.76	13.76	–	86%
[38]	CEAR	8.33	8.33	–	91%
[35]	MCYT	2.60	2.34	–	87%
[17]	GPDS	8.98	9.17	9.08	–
[17]	MCYT	5.36	1.25	3.31	–
[17]	CEDAR	1.67	2.50	2.09	–
[65]	GPDS	4.16	6.67	5.42	–
BoF-CPS approach	MCYT	0.16	0.13	0.14	85%
BoF-CPS approach	GPDS	0.06	0.003	0.03	98%
BoF-CPS approach	CEDAR	0.07	0.09	0.08	91%

98%, and 91%, respectively. In future work, the focus will be on combining the writer-dependent and the writer-independence scenarios to recognize the signature pattern. Based on the studies and results of the evaluations performed in the proposed approach, the following can be suggested for subsequent research: (1) imposing the initial restriction on the selection of candidate points, (2) determining candidate points based on pixel density in the sample that do not have intersecting pixels and (3) Frequency assessment of the occurrence of features in the corresponding candidate points in the fusion feature phase.

Declarations

Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.

References

1. Ahrabian K, Babaali B (2017) On usage of autoencoders and siamese networks for online handwritten signature verification. *Corr* (2017). arXiv:1712.02781
2. Avola A, Del Buono A, Del Nostro P, Wang R (2009) A novel online textual/graphical domain separation approach for sketch-based interfaces. In: *Book: new directions in intelligent interactive multimedia systems and services*
3. Avola A, Bernardi M, Cinque L, Foresti GL, Marini MR, Massaroni C (2017) A machine learning approach for the online separation of handwriting from freehand drawing. *Int Conf Image Anal Process*. https://doi.org/10.1007/978-3-319-68560-1_20
4. Avola A, Bernardi M, Cinque L, Foresti GL, Massaroni C (2019) Online separation of handwriting from freehand drawing using extreme learning machines. *Multimed Tools Appl*. <https://doi.org/10.1007/s11042-019-7196-1>
5. Batista L, Granger E, Sabourin R (2012) Dynamic selection of generative–discriminative ensembles for off-line signature verification. *Pattern Recogn* 45(4):1326–1340
6. Ben X, Zhang P, Lai Z, Yan R, Zhai X, Meng W (2019) A general tensor representation framework for cross-view gait recognition. *Pattern Recogn*. <https://doi.org/10.1016/j.patcog.2019.01.017>
7. Bertolini D, Oliveira LS, Justino E, Sabourin R (2010) Reducing forgeries in writer-independent off-line signature verification through ensemble of classifiers. *Pattern Recogn* 43(1):387–396
8. Chandra S, Maheshkar S (2017) Verification of static signature pattern based on random subspace, REP tree and bagging. *Multimed Tools Appl*
9. Chowdhury M et al (2016) Human detection and localization in secure access control by analysing facial features. In: *IEEE 11th conference on industrial electronics and applications (ICIEA)*. IEEE, pp 1311–1316
10. Cortes C.; Vapnik V.N. (1995) Support-vector networks. *Mach Learn* 20(3):273–297. CiteSeerX 10.1.1.15.9362. <https://doi.org/10.1007/BF00994018>
11. Cover TM, Thomas JA (2006) *Elements information theory*; Wiley: Hoboken.
12. Dargan S, Kumar M (2019) A comprehensive survey on the biometric recognition systems based on physiological and behavioral modalities. *Expert Syst Appl*. <https://doi.org/10.1016/j.eswa.2019.113114>
13. Darwish S, El-Nour A (2016) Automated offline arabic signature verification system using multiple features fusion for forensic applications. *Arab J Forensic Sci Forensic Med* 1(4):424–437
14. Diaz M, Ferrer MA, Eskander GS, Sabourin R (2017) Generation of duplicated off-line signature images for verification systems. *IEEE Trans Pattern Anal Mach Intell* 39(5):951–964
15. Diaz M, Fischer A, Ferrer MA, Plamondon R (2018) Dynamic signature verification system based on one real signature. *IEEE Trans Cybern* 48(1):228–239
16. Diaz M, Ferrer MA, Impedovo D, Malik MI, Pirlo G, Plamondon R (2019) A perspective analysis of handwritten signature technology. *ACM Comput Surv* 51(6):Article 11739 pages. <https://doi.org/10.1145/3274658>
17. Faiza EB, Attique M, Sharif M, Javed K, Nazir M, Abbasi A, Iqbal Z, Riaz N (2020) Offline signature verification system: a novel technique of fusion of GLCM and geometric features using SVM. *Multimed Tools Appl*. <https://doi.org/10.1007/s11042-020-08851-4>
18. Ferrer MA, Diaz-Cabrera M, Morales A (2013) Synthetic off-line signature image generation. In: *Biometrics (ICB), International Conference on*, pp 1–7

19. Ferrer MA, Chanda S, Diaz M, Banerjee CK, Majumdar A, Carmona-Duarte C, Acharya P, Pal U (2018) Static and dynamic synthesis of Bengali and Devanagari signatures. *IEEE Trans Cybern* 48(10):2896–2907
20. Fischer A, Plamondon R (2017) Signature verification based on the kinematic theory of rapid human movements. *IEEE Trans Hum Mach Syst* 47(2):169–180
21. Galbally J, Diaz-Cabrera M, Ferrer MA, Gomez-Barrero M, Morales A, Fierrez J (2015) On-line signature recognition through the combination of real dynamic data and synthetically generated static data. *Pattern Recogn* 48(9):2921–2934
22. Ghandali S, Moghaddam ME (2008) A method for off-line Persian signature identification and verification using DWT and image fusion. In: *International Symposium on signal processing and information technology*. IEEE, pp 315–319
23. Guerbai Y, Chibani Y, Hadjadji B (2015) The effective use of the one-class SVM classifier for handwritten signature verification based on writer-independent parameters. *Pattern Recogn* 48(1):103–113
24. Guido RC (2018) A tutorial review on entropy-based handcrafted feature extraction for information fusion. *Inf Fusion* 41:161–175. <https://doi.org/10.1016/j.inffus.2017.09.006>
25. Hafemann LG, Sabourin R, Oliveira LS (2016) Writer-independent feature learning for offline signature verification using deep convolutional neural networks. In: *International joint conference neural networks (IJCNN'16)*. IEEE, pp 2576–2583
26. Hafemann LG, Sabourin R, Oliveira LS (2017) Learning features for offline handwritten signature verification using deep convolutional neural networks. *Pattern Recogn* 70(2017):163–176
27. Hafemann LG, Sabourin R, Oliveira LS (2017) Offline handwritten signature verification- literature review. *IEEE978-1-5386-1842-4/17/\$31.00_c 2017 IEEE*
28. Hafemann LG, Oliveira LS, Sabourin R (2018) Fixed-sized representation learning from offline handwritten signatures of different sizes. *Int J Doc Anal Recognit* 21:219–232
29. Heracleous P, Even J, Sugaya F, Hashimoto M, Yoneyama A (2018) Exploiting alternative acoustic sensors for improved noise robustness in speech communication. *Pattern Recogn Lett*. <https://doi.org/10.1016/j.patrec.2018.07.014>
30. Houtinezhad M., Ghaffary H.R. (2019) *Writer-independent signature verification based on feature extraction fusion*, Springer, Multimedia Tools and Applications.
31. Huang D-Y (2009) Optimal multi-level thresholding using a two-stage Otsu optimization approach. *Pattern Recogn Lett* 30(3):275–284. <https://doi.org/10.1016/j.patrec.2008.10.003>
32. Jagtap AB, Sawat DD, Hegadi RS et al (2020) Verification of genuine and forged offline signatures using Siamese neural network (SNN). *Multimed Tools Appl* 79:35109–35123. <https://doi.org/10.1007/s11042-020-08857-y>
33. Jain A, Kumar SS, Pratap SK (2020) Handwritten signature verification using shallow convolutional neural network. *Multimed Tools Appl*. <https://doi.org/10.1007/s11042-020-08728-6>
34. Jain A, Kumar SS, Pratap SK (2020) Signature verification using geometrical features and artificial neural network classifier. *Neural Comput & Applic*. <https://doi.org/10.1007/s00521-020-05473-7>
35. Jindal U, Dalal S (2019) A hybrid approach to authentication of signature using DTSVM. In: Rathore V, Worring M, Mishra D, Joshi A, Maheshwari S (eds) *Emerging trends in expert applications and security*. *Advances in intelligent systems and computing*, vol 841. Springer, Singapore. https://doi.org/10.1007/978-981-13-2285-3_39
36. Kalera MK, Srihari S, Xu A (2004) Offline signature verification and identification using distance statistics. *Int J Pattern Recognit Artif Intell* 18:1339–1360
37. Kumar R, Kundu L, Sharma JD (2010) A writer-independent off-line signature verification system based on signature morphology. *Conference: first international conference on intelligent interactive technologies and multimedia*. <https://doi.org/10.1145/1963564.1963610>
38. Kumar R, Sharma JD, Chanda B (2012) Writer-independent off-line signature verification using surroundedness feature. *Pattern Recogn Lett* 33:301–308
39. Lanitis A (2009) A survey of the effects of aging on biometric identity verification. *Int J Biom* 2(1):34–52
40. Larkins R., Mayo M. (2008) Adaptive feature thresholding for off line signature verification. *IEEE*.
41. Lee S, Pan JC (1992) Offline tracing and representation of signatures. *IEEE Trans Syst Man Cybern* 22(4): 755–771. <https://doi.org/10.1109/21.156588>
42. Liwicki M, Blumenstein M, Heuvel EL, Berger CEH, Stoel RD, Found B, Chen X, Malik MI (2011) SigComp11: signature verification competition for on- and offline skilled forgeries. *Proc. 11th International Conference on Document Analysis and Recognition*
43. Mamta HM (2014) Robust authentication using the unconstrained infrared face images. *Expert Syst Appl* 41(14):6494–6511
44. Marsisco MD, Petrosino A, Ricciardi S (2016) Iris recognition through machine learning techniques: a survey. *Pattern Recogn Lett* 82(2):106–115

45. Mendenhall W, Beaver RJ, Beaver BM (2019) Introduction probability and statistics 14th Edition, 1133103758 ISBN/ASIN, ISBN -10 : 1337554421, Publisher : Cengage Learning; 15th edition
46. Moi SH, Asmuni H, Hassan R, Othman RM (2014) Weighted score level fusion based on nonideal iris and face images. *Expert Syst Appl* 41(11):5390–5404
47. Muller MK, Tremer M, Bodenstern C, Wurtz RP (2013) Learning invariant face recognition from examples. *Neural Netw* 41:137–146
48. Okawa M (2018b) Synergy of foreground–background images for feature extraction: offline signature verification using fisher vector with fused KAZE features. *Pattern Recogn* 79:4 80–4 89
49. Okawa M (2020) Online signature verification using single-template matching with time-series averaging and gradient boosting. *Pattern Recogn* 102(2020). <https://doi.org/10.1016/j.patcog.2020.107227>
50. Ortega-Garcia J, Fierrez-Aguilar J, Simon D, Gonzalez J, Faundez-Zanuy M, Espinosa V, Sature A, Hernaez I, Igarza J-J, Vivaracho C et al (2003) Mcyt baseline corpus: a bimodal biometric database. *IEEE Proc Vis Image Signal Process* 150(6):395–401
51. Otsu N (1979) A threshold selection method from gray-level histograms. *IEEE Trans Syst Man Cybern* 9(1):6266. <https://doi.org/10.1109/TSMC.1979.4310076>
52. Park J, Govindaraju V, Srihari SN (2000) OCR in a hierarchical feature space. *IEEE Trans Pattern Anal Mach Intell* 22(4)
53. Parodi M, Gomez JC, Liwicki M, Alewijnse L (2013) Orthogonal function representation for online signature verification: which features should be looked at? *IET Biom* 2(4):137–150
54. Piekarczyk M, Ogiela MR (2013) Matrix-based hierarchical graph matching in off-line handwritten signatures recognition. In: 2nd IAPR Asian conference on pattern recognition. IEEE, pp 897–901
55. Pirlo G, Impedovo D (2013) Cosine similarity for analysis and verification of static signatures. *IET Biom* 2(4):151–158
56. Pirlo G, Impedovo D (2013) Verification of static signatures by optical flow analysis. *IEEE Trans Hum Mach Syst* 43(5):499–505
57. Pirlo G, Cuccovillo V, Diaz-Cabrera M, Impedovo D, Mignone P (2015) Multidomain verification of dynamic signatures using local stability analysis. *IEEE Trans Hum Mach Syst* 45(6):805–810 <https://ieeexplore.ieee.org/document/7145458>
58. Rantzsch H, Yang H, Meinel C (2016) Signature embedding: writer independent offline signature verification with deep metric learning. In: International. Symposium on visual computing. Springer, pp 616–625
59. Rivard D, Granger E, Sabourin R (2013) Multi-feature extraction and selection in writer-independent off-line signature verification. *Int J Doc Anal Recognit* 16(2013):83–103
60. Serdouk Y, Nemmour H, Chibani Y (2014). Combination of OC-LBP and longest run features for off-line signature verification. Tenth Int Conf Signal-Image Technol Internet-Based Syst. <https://doi.org/10.1109/sitis.2014.36>
61. Serdouk Y, Nemmour H, Chibani Y (2016) New off-line hand- written signature verification method based on Artificial immune recognition system. *Expert Syst Appl*. <https://doi.org/10.1016/j.eswa.2016.01.001>
62. Serdouk Y, Nemmour H, Chibani Y (2017) Handwritten signature verification using the quad-tree histogram of templates and a support vector-based artificial immune classification. *Image Vis Comput* 66:26–35. <https://doi.org/10.1016/j.imavis.2017.08.004>
63. Serdouk Y, Nemmour H, Chibani Y (2018) [New histogram-based descriptor for off-line handwritten signature verification](https://doi.org/10.1016/j.patrec.2018.01.021). *IEEE/ACS 15th Int Conf Comput Syst Appl (AICCSA)*
64. Sezgin M, Sankur B (2004) Survey over image thresholding techniques and quantitative performance evaluation. *J Electron Imaging* 13(1):146–165
65. Sharif M, Attique KM, Faisal M, Yasmin M, Fernandes SL (2018, 2018) A framework for offline signature verification system: best features selection approach. *Pattern Recogn Lett*. <https://doi.org/10.1016/j.patrec.2018.01.021>
66. Sharma A, Sundaram S (2016) An enhanced contextual DTW based system for online signature verification using vector quantization. *Pattern Recogn Lett* 84(2016):22–28
67. Sharma A, Sundaram S (2018) On the exploration of information from the DTWcost matrix for online signature verification. *IEEE Trans Cybern* 48(2):611–624
68. Soleimani A, Araabi B, Fouladi K (2016) Deep multitask metric learning for offline signature verification. *Pattern Recogn Lett* 80:84–90
69. Srikantan G, Lam SW, Srihari SN (1996) Gradient-based contour encoding for character recognition. *Pattern Reconnect* 29(7):1147–1160 0031-3203/96 \$ 15.00 + .00
70. Taboga M (2017) Lectures on probability theory and mathematical statistics. 3rd Edition Paperback, December 8, 2017, Available on: <https://www.statlect.com/fundamentals-of-statistics/variance-estimation>
71. Tolosana R, Vera-Rodriguez R, Ortega-Garcia J, Fierrez J (2015) Preprocessing and feature selection for improved sensor interoperability in online biometric signature verification. *IEEE Access* 3:478–489

72. Tolosana R, Vera-Rodriguez R, Fierrez J, Morales A, Ortega-Garcia J (2017) Benchmarking desktop and mobile handwriting across COTS devices: the e-BioSign biometric database. *PLoS One* 12(5):e0176792
73. Tolosana R, Vera-Rodriguez R, Fierrez J, Ortega-Garcia J (2018) Exploring recurrent neural networks for online handwritten signature biometrics. *IEEE Access* 2018:5128–5138
74. Ulrich L et al (2020) Analysis of RGB-D camera technologies for supporting different facial usage scenarios. *Multimed Tools Appl*:1–24
75. Vamvakas G, Gatos B, Perantonis SJ (2010) Handwritten character recognition through two-stage foreground sub-sampling. *Pattern Recogn*:2807–2816
76. Vargas JF, Ferrer MA, Travieso CM, Alonso JB (2011) Offline signature verification based on grey level information using texture features. *Pattern Recogn* 44(2):375–385
77. Wang Z, Zhang D (1999) Progressive switching median filter for the removal of impulse noise from highly corrupted images. *IEEE Trans Circuit Syst II, Analog Digit Signal Process* 46:78–80
78. Yahyatabar ME, Ghasemi J (2017) Online signature verification using double-stage feature extraction modeled by dynamic feature stability experiment. *IET Biom*:393–401
79. Yahyatabar ME, Baleghi Y, Karami MR (2013) Online signature verification: a Persian-language specific approach. In *21st Iranian conference on electrical engineering (ICEE'13)*, pp 1–6
80. Yilmaz MB, Yanikoglu B (2016) Score level fusion of classifiers in off-line signature verification. *Inf Fusion* 32:109–119
81. Zhang S, Li F (2012) Off-line handwritten Chinese signature verification based on support vector machine multiple classifiers. In: *Advances in electric and electronics*. Springer, pp 563–568
82. Zois EN, Alexandridis A, Economou G (2019) Writer independent offline signature verification based on asymmetric pixel relations and unrelated training-testing datasets. *Expert Syst Appl* 125(2019):14–32
83. Zois NE, Papagiannopoulou M, Dimitrios (2018) Hierarchical dictionary learning and sparse coding for static signature verification. *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.