

Article

Writer Identification Using Handwritten Cursive Texts and Single Character Words

Tobias Kutzner¹, Carlos F. Pazmiño-Zapatier², Matthias Gebhard¹, Ingrid Bönninger¹, Wolf-Dietrich Plath¹ and Carlos M. Travieso^{2,3,*}

¹ Institute of Medical Technology, Brandenburg University of Technology Cottbus, 01968 Senftenberg, Germany; Tobias.Kutzner@b-tu.de (T.K.); Matthias.Gebhard@b-tu.de (M.G.); Ingrid.Boenninger@b-tu.de (I.B.); Wolf-Dietrich.Plath@b-tu.de (W.-D.P.)

² Instituto para el Desarrollo Tecnológico y la Innovación en Comunicaciones (IDeTIC), Universidad de Las Palmas de Gran Canaria, 35017 Las Palmas de Gran Canaria, Spain; carlos.pazmino101@alu.ulpgc.es

³ Departamento de Señales y Comunicaciones, Universidad de Las Palmas de Gran Canaria, 35017 Las Palmas de Gran Canaria, Spain

* Correspondence: carlos.travieso@ulpgc.es; Tel.: +34-928-452-864

Received: 30 December 2018; Accepted: 23 March 2019; Published: 1 April 2019



Abstract: One of the biometric methods in authentication systems is the writer verification/identification using password handwriting. The main objective of this paper is to present a robust writer verification system by using cursive texts as well as block letter words. To evaluate the system, two datasets have been used. One of them is called Secure Password DB 150, which is composed of 150 users with 18 samples of single character words per user. Another dataset is public and called IAM online handwriting database, and it is composed of 220 users of cursive text samples. Each sample has been defined by a set of features, composed of 67 geometrical, statistical, and temporal features. In order to get more discriminative information, two feature reduction methods have been applied, Fisher Score and Info Gain Attribute Evaluation. Finally, the classification system has been implemented by hold-out cross validation and k-folds cross validation strategies for three different classifiers, K-NN, Naïve Bayes and Bayes Net classifiers. Besides, it has been applied for verification and identification approaches. The best results of 95.38% correct classification are achieved by using the k-nearest neighbor classifier for single character DB. A feature reduction by Info Gain Attribute Evaluation improves the results for Naïve Bayes Classifier to 98.34% for IAM online handwriting DB. It is concluded that the set of features and its reduction are a strong selection for the based-password handwritten writer identification in comparison with the state-of-the-art.

Keywords: handwriting; password identification; online strokes; writer verification; reduction feature; on-line handwriting features

1. Introduction

The new and different kinds of sensors open up the option to modern societies to analyze and collect different data in all fields. For a high percentage of these data, the software systems have to guarantee the protection of data privacy, such as bank data, examination marks in schools and universities, data in medical diagnoses, data in presidential elections, and movement data in car-assist systems. Therefore, many security and encryption methods are developed.

However, software security systems are developed by humans and therefore they can be decrypted by humans. This is one reason for the increasing interest in using biometric methods in authentication systems in recent years. There are many systems to recognize persons by gait, fingerprint, iris, and voice analysis, etc. In particular, in this proposal, a writer verification is developed using their online handwriting information.

To authenticate a system using a password, the user name and password are normally entered via keyboard.

If touch screen devices are used for authentication, the password can be entered by hand on the touch screen. In addition to the user name and password, the biometric information contained in the handwritten password can also be verified to make authentication more secure.

Therefore, the aim of this paper is the presentation of a robust writer verification system. Recent writer identification studies have focused on two main problems, the author profile and the author verification.

In Reference [1], authors propose a system that uses the topological pixel density and pixel distribution and the gradient feature Gradient Local Binary Patterns. At first, each feature is associated to one SVM classifier, then Sugeno's Fuzzy Integral with a set of SVMs is used. As test records, the three databases IAM, KHATT, and IAM + KHATT were used. The combined system reaches at least 4% better results than the individual methods.

In order to objectively compare the performance of writer identification and gender classification systems, a ICDAR2015 Competition on Signature Verification and Writer Identification for On- and Off-line Skilled Forgeries (SigWIcomp2015) using QUWI database was organized [2]. The competition received five writer identification tasks and eight gender classification tasks. The overall best results for writer identification are reported by Nuremberg method. The best results for gender classification are reached by the CVC method.

A system of classification of gender is proposed in Reference [2] based on the protocols used in three competitions, ICDAR 2013, ICDAR 2015 and ICFHR 2016. The most interesting aspect of these competitions was the use of a dataset with writing samples of the same person in Arabic and in English. The textual information that distinguishes between male and female handwriting is extracted from different combinations of Oriented Basic Image Features (oBIFs) and used to train a Super Vector Machine classifier. About the author verification, some references can be shown. In author verification systems, the author of the handwritten document is recognized. In Reference [3], authors present a textural-based method for writer identification. The General Pattern Run-Length Transformation is used for binary and gray scale images of the four public databases. The experimental results with an identification rate between 84.3% and 99.5% outperform the state-of-the-art approaches.

In Reference [4], counting methods, vote technique and supervised learning are proposed to identify if an author has written a given document or not. The counting method in combination with the vote technology delivers identification results of 70.7% but its effectiveness highly depends on the number of known documents.

Many recent studies are focused on signature verification [5–7]. One of the recent signature verification investigations [5] extracts texture features, Local Binary Patterns (LBP) and Uniform Local Binary Patterns (ULBP). As a classification algorithm, the Nearest Neighbor is used. The research work is based on a database of 6240 Bangla and Hindi signature datasets and 7800 skilled forgeries. From the results obtained, applying LBP and ULBP features were almost equal. Researchers neither could determine a significant difference using eight or 12 signatures by each writer. The K-Value of the Nearest Neighbor has an influence on the result quality, due to experimentally, the proposal reached a best accuracy of about 76%.

A previously published paper [6] proposes a two-stage score normalization method to minimize the lack of information of intra-user variability. In the first step, simple forgeries are detected, and in the second stage, more skilled forgeries. For the experiments, the MCYT and the SUSIG databases with 10 reference signatures were used. Simple forgeries were detected with an Equal Error Rate (EER) of 0.85% for the MCYT dataset and skilled forgeries with an EER of 2.13% for SUSIG dataset.

Other researchers [8,9] have attracted attention to identify the writers by their handwritten characters or numbers. Characteristic points on the image are detected by structural analysis and a SIFT-based detector in Reference [8]. For the feature selection, they use Local Binary Pattern, the Loci Descriptor, and the Index of Dissimilarity. For classification, the Support Vector Machine was chosen.

For the experiments, a database of isolated Bangla characters from 100 writers of different ages (10–65 years) was used. Each writer has written 11 samples of each character and numeral. By writing the 45 important characters, the accuracy was 86.58%. The best results of writer identification accuracy of 92.47% were reached by using the complete set of 193 Bangla characters.

In the domain of forensic application, sometimes it is required to verify writers by handwritten isolated characters or numerals.

An accuracy of 97% of writer verification by handwritten characters and numerals is reached [10] by incrementally using the construction of an Adaptive Radial Basis Function Network. The system is evaluated on characters and numerals of 15 writers.

Biometrical systems based on handwriting, which identify writers by their signature, are proposed in References [11–13].

The Competition on Signature Verification and Writer identification for On- and Off-line Skilled Forgeries (SigWIcomp2015) delivers not only interesting results in performance comparison but also in signature verification [7]. Forty systems participated in the competition, 9 for off-line Italian signature verification, 9 for off-line Bengali signature verification, 12 for on-line German signature verification, and 10 for writer identification of an English text. The best results in off-line signature verification reached a system with a histogram of oriented gradients and LBP together with SVM classification [14]. The winner of on-line signature verification was a commercial product that calculates more than 70,000 movement characteristics of each handwritten sample. The best writer identification system of the English text was a system [2] based on the combination of edge-hinge features; multi-scale features and edge-based directional features.

In Reference [15], the contribution of preprocessing methods were analyzed; the contrast normalization, the slant correction, the size normalization, and the Median filter are used as preprocessing methods.

The Maximum Likelihood trained HMM system reaches a Word Error Rate of 16.7% with samples of the databases RIMES and IAM.

Only a few studies have looked at a comparison of efficiency of different feature types like statistical, geometrical and temporal features.

Many studies have focused on statistical features like image statistics, analysis of variance [16,17], and mean and median [18]. Handwriting feature like position, velocity, acceleration, and pressure were analyzed in Reference [17]. The number of dimensions was reduced by PCA. The ANOVA test is performed between each one of the five reference signatures and the testing signature. Using a database of 130 genuine signatures and 170 forgery signatures, a False Acceptance Rate of 2% and a False Rejection Rate of 5% has been achieved.

In References [19,20], statistical and geometrical features were combined. An automatic character prototype selection is proposed in Reference [19]. By the use of HMM and graph-matching methods, the handwriting of a historical document set was recognized. Four prototype selection methods are presented, median, center, spanning, and k-center ($k = 4$) selection. All four automatic recognition methods reached better results as the manual recognition of 94.0%.

Using a combination of oriented Basic Features with the background concavity features and the support vector machine as the classification method, a precision of 95.21% was achieved in Reference [20].

Mostly geometrical features were used in References [21–23]. The authors of Reference [21] relied on local descriptors that capture the texture, shape and curvature features, and combine K-Adjacent Segments, SURF, and Contour Gradient Descriptors. The advantage of utilizing multiple features was demonstrated using three databases. The results are: With the database IAM, 98.7% accuracy; with ICDAR 2013, 99.8% (Greek dataset), 98.8% (English dataset); and with the CVL database, 99.7%.

Mainly geometrical features were used in References [24–26], too. A word spotting method using contour-based models was proposed in Reference [26]. Contour features are extracted from segmented word images to obtain a representative shape of a word class. Thereafter, the examples are used to learn a statistical model of intra-class deformations. Testing with the ICDAR'07 and ICDAR'09 databases, a recognition rate of 87.4% was reached.

The attention on geometrical and temporal features like writing velocity or acceleration was attracted by References [27–31]. It is known that signatures written by the same person differ slightly from time to time. Therefore, in References [7,28,32,33], only the part of less fluctuation is considered as the stable feature of handwriting in the signing process. The forward variance along the forward direction from the starting point and the backward variance from the end point along the backward direction are calculated. In order to extract time intervals with less variance, thresholds are used. The experiments performed on the MCYT-100 database (5.000 signatures from 100 persons) reached an Equal Error Rate of 4.49%.

The authors of Reference [31] use descriptors based on spatially and temporally neighboring strokes to classify into one of the three classes: text, figure and table. They use English (700 pages) and Japanese (1520) datasets written by 40 people.

Considerable attention on time features was attracted by References [32–36]. The distribution on the handwriting feature pairs age-time up and age-time down were analyzed in Reference [35] with the goal to obtain better knowledge about the correlation between handwriting and the age. The experiment base was the BIOSECURED database with handwriting samples of 400 people. The preliminary results of the research show that changes of feature value are not affected by aging but rather by the motor or cognitive disorder of elderly people.

Previous studies [37–40] have focused on time or stroke features of handwriting. A writer recognition system for touch-screen mobile devices was proposed in Reference [37] for non-Latin languages with a large set of characters. Therefore, the authors avoid complex time-intensive algorithms like Multilayer Perceptron, Support Vector Machine or Hidden Markov Model. To recognize strokes in a character, a small number of prototype of stroke shapes is used with weighted Dynamic Time Warping and a Look-Up Table. To analyze the topographic information of pen pressure during the writing process is the central idea of the research [39].

Previous studies [40,41] have focused on pressure or used pressure as an additional feature. Only a few researchers [23,41] have used geometrical and temporal features.

Nevertheless, a robust authentication system, which identifies the writers independent of their writing style, if they write their signature or if they write isolated characters, still has not been found.

In our previous [42–48] studies, the focus was on the writer verification of handwritten single character passwords. So far no attention has been paid to establish a relationship between writer verification block letter and cursive texts.

The main objective of this paper is to find a discriminative, strong and novel set of features, which can be used for writer verification/identification using handwritten words composed of single character as well as cursive text. This approach combines statistical, geometrica, and temporal features. Furthermore, this proposal wants to analyze the efficiency of different types of features. For the classification stage, the proposal uses Bayes Net, Naïve Bayes and Nearest Neighbor classifiers. The proposed system is evaluated by public databases, the Secure Password DB 150 [42] and the IAM Online English Handwritten Text Database [49] (public) in order to show its robustness. Finally, the proposal is compared with different references from the state-of-the-art methods. The innovation of this proposal is focused on its final set of features and its reduction feature, which is applied for two different kinds of handwritten data: cursive text and isolated characters. Besides, it reaches better results than the state-of-the-art methods.

The use of Dynamic Time Warping (DTW) on signature verification can be found in the state-of-the-art methods too with good results [32,50,51]. Its use on handwritten text is new, and it is an innovation of this proposal.

An important aspect is the analysis of the signature stability of handwritings, in particular, Reference [52] shows a literature review, where the signature stability is analyzed for the verification of handwritten strokes. It is the key to develop the actual current proposal.

Most of the references deal with identification, but this proposal suggests to analyze a word composed of isolated characters, as if it were a password, and to check the grade of discrimination for using as security applications, as is the access with login and password.

The paper is organized as follows. Section 2 of this proposal describes the material: the used databases, the raw data and their pre-processing. Section 3 describes some of the 67 used features. Section 4 contains the feature reduction and classification methods. The experimental methodology can be found in Section 5. Section 6 presents the reached results of each experiment and a comparison versus the state-of-the-art methods. Finally, Section 7 offers conclusions.

2. Materials

In order to have a robust proposal, the dataset has to play a very important role. Therefore, this proposal will use two datasets for the evaluation and validation of the proposed system. They are The SECURE PASSWORD dataset [42] with 150 users, which is an own dataset; and The IAM online handwriting dataset [49] with 220 writers, which is a public dataset.

2.1. Secure Password Dataset

The database contains 2792 handwritten passwords with eight single characters of 150 users. Hereinafter, this database will be called Secure Password-DB-150. It is composed of handwritten passwords and it was collected in Spain and Germany, in particular at University of Las Palmas de Gran Canaria (ULPGC) and Brandenburg University of Technology (BTU), between 2012 and 2016. Some authors designed and built this dataset. The secure password-DB-150 considers about 18 samples of each user. The handwritten passwords are single characters, written in one or more lines (see Figure 1).



Figure 1. Sample of handwritten password in secure password-DB-150.

The passwords are written on a display of an Android smartphone, preprocessed by a Java program, and transferred to a server, where feature extraction, classification, writer identification, and verification follow (see Figure 2).

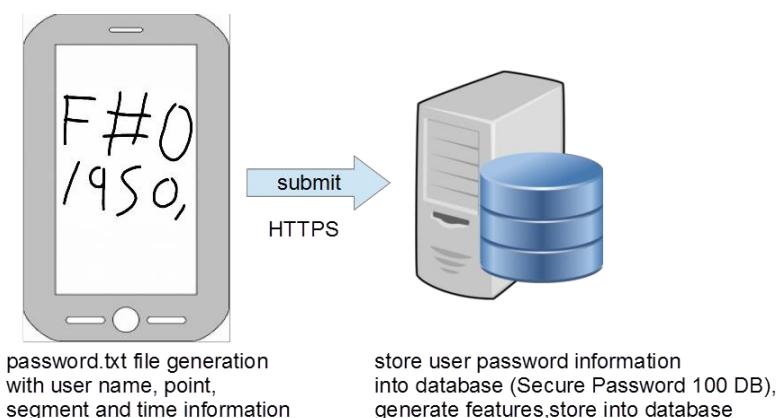


Figure 2. Structure of the writer verification process using database secure-password-DB-150.

2.2. IAM Online Handwriting Dataset

To determine the possibility of using the same features for writer verification with handwritten single character words, the experiments work with another dataset in order to validate the proposal. That used dataset was the IAM online handwriting database with cursive texts for verification.

The database contains 1760 handwritten text examples of 220 people. The text contains between 40 and 60 words on one and more than one line. The IAM online handwriting database considers eight samples of each user. (see Figure 3).



Figure 3. Sample of handwritten text of IAM online handwriting database.

2.3. Data Adaptation

In order to use the proposed system, the IAM data have to be transformed into a defined format. A XML parser program is used to transform the IAM data to our format. For the transformed texts, the same procedure is applied to the handwritten passwords of the Secure Password-DB-150.

3. Methods for Feature Extraction

This section contains information about the feature extraction. In particular, this proposal includes a set of 67 features, which are 18 geometrical, 33 statistical, and 16 temporal features. All features have a real value. The list of these parameters is included in Table 1. The name of the parameter is composed by the type of parameter, underscore and the name of parameter; for statistical features, it is “STAT_parameter_name”; for geometrical features, it is “GEO_parameter_name”; and for temporal features, it is “TEMP_parameter_name”. It will be used for the rest of the document. The most significant features will be described in the next subsections, avoiding to describe intuitive and easy parameters.

Table 1. All features used in this work with their numbers and reference.

#	Parameter	Reference	#	Parameter	Reference
1	GEO_REGRESSION_LOWER_HORANGLE	own	35	STAT_MIN_VY	own
2	GEO_REGRESSION_ON_HORANGLE	own	36	STAT_MAX_VX	own
3	GEO_CENTRAL_POINT	own	37	STAT_MAX_VY	own
4	GEO_REGRESSION_UPPER_HORANGLE	own	38	STAT_RELATION_VX_MAX	own
5	GEO_EUCLID	own	39	STAT_RELATION_NVxz	own
6	GEO_POINT_ANGLE	own	40	STAT_RELATION_NVyz	own
7	GEO_HORIZONTAL_POINT_ANGLE	own	41	STAT_VERTICAL_SKEWNESS	[53]
8	GEO_SPHI	own	42	STAT_SPREADNESS	[53]
9	GEO_HYP_ANGLE	own	43	STAT_INERTIAL_RATIO	[50]
10	GEO_REG_ANGLE	own	44	STAT_ASPECT_RATIO	[53]
11	GEO_WORD_WIDTH	own	45	STAT_HORIZONTAL_SKEWNESS	[53]
12	GEO_WORD_HEIGHT	own	46	STAT_BALANCE_HORIZONTAL_EXTENSION	[53]
13	GEO_SURFACE	own	47	STAT_BALANCE_VERTICAL_EXTENSION	[53]
14	GEO_SLANT_AMPLITUDE	[53]	48	STAT_DWH	own
15	GEO_SLANT	[53]	49	STAT_RELATION_POINTS_SPEED	own
16	GEO_ORIENTATION	[53]	50	STAT_X_NUMBER_POINTS SEGMENT	own
17	GEO_WIDTH	own	51	STAT_Y_NUMBER_POINTS SEGMENT	own
18	GEO_HEIGHT	own	52	TEMP_TIME	own
19	STAT_POINTS	own	53	TEMP_RELATIVE_WRITING_DURATION	own
20	STAT_SEGMENTS	own	54	TEMP_TIME_V_MAX	own
21	STAT_NUM_STROKES	own	55	TEMP_RELATION_TIME_GAP_ALL	own
22	STAT_STANDARD_DERIVATION_X	own	56	TEMP_TIME_MIN_X	own
23	STAT_STANDARD_DERIVATION_Y	own	57	TEMP_TIME_MIN_Y	own
24	STAT_MEAN_NUMBER_POINTS SEGMENT	own	58	TEMP_TIME_MAX_X	own
25	STAT_DTW_Y	own	59	TEMP_TIME_MAX_Y	own
26	STAT_DTW_X	own	60	TEMP_TIME_VX_MAX	own
27	STAT_DTW_T	own	61	TEMP_TIME_VY_MAX	own
28	STAT_RELATION_VX_NEGATIVE	own	62	TEMP_TIME_VX_MIN	own
29	STAT_RELATION_VX_POSITIVE	own	63	TEMP_TIME_VY_MIN	own
30	STAT_RELATION_VY_NEGATIVE	own	64	TEMP_TIME_X_POS	own

Table 1. Cont.

#	Parameter	Reference	#	Parameter	Reference
31	STAT_RELATION_VY_POSITIVE	own	65	TEMP_TIME_Y_POS	own
32	STAT_MEDIAN_X	own	66	TEMP_TIME_X_NEG	own
33	STAT_MEDIAN_Y	own	67	TEMP_TIME_Y_NEG	own
34	STAT_MIN_VX	own			

The extraction of features begins with parameter sampling like point coordinates, time values and touch down and touch up information for the identification of the connected components (segments). Where signatures consist of coordinates (x_i, y_i) , $i = [1, n]$ (n : number of signature points) and segments consist of coordinates (x_i, y_i) , $i = [1, np]$ (np : number of segment points). These terms will be used in all of the following formulas for a more detailed description of the features. A new connected component begins whenever the pen or finger is placed down on the display for writing and ends when the pen or finger is lifted up again. The features generated per segment are summed up and divided by the number of segments of the entire signature.

3.1. Geometrical Features

The different geometrical features are calculated and the features we have developed ourselves are described in more detail in the next paragraphs.

- REGRESSION_LOWER_HORANGLE (Table 1 #1): Sum of points under horizontal axis (Equation (1));

$$RLH = \sum_{y_i < \frac{\sum y_i}{y_n}} y_i \quad (1)$$

- REGRESSION_ON_HORANGLE (Table 1 #2): Sum of points at horizontal axis (Equation (2));

$$ROH = \sum_{y_i = \frac{\sum y_i}{y_n}} y_i \quad (2)$$

- CENTRAL_POINT (Table 1 #3): Number of the central point of the Signature in x direction (Equation (3));

$$P = \frac{\sum x_i}{n} \quad (3)$$

- REGRESSION_UPPER_HORANGLE (Table 1 #4): Sum of points above horizontal axis (Equation (4));

$$RUH = \sum_{y_i > \frac{\sum y_i}{y_n}} y_i \quad (4)$$

- EUCLID (Table 1 #5): Considers the Euclidean distance between the single points of the segment.
- POINT_ANGLE (Table 1 #6): Determines the angle between terminal points in the beginning of a segment in relation to the lower display border (Equation (5));

$$\alpha = \arctan \left(\frac{|y_{np} - y_1|}{|x_{np} - x_1|} \right) \quad (5)$$

where (x_1, y_1) is the first point of the segment and (x_n, y_n) is the last point of the segment (see Figure 4).

- HORIZONTAL_POINT_ANGLE (Table 1 #7): This angle is calculated from the horizontal corner between the terminal points of the segments of the whole signature. To calculate this angle, a point-displaced eight units to the left from the center of rectangle has to be selected in this implementation. (see Figure 5, Equation (6)).

$$a = \arctan \frac{\left| \frac{y_3 - y_1}{x_3 - x_1 - 8} \right| - \left| \frac{y_3 - y_2}{x_3 - x_2 - 8} \right|}{1 + \left| \frac{y_3 - y_1}{x_3 - x_1 - 8} \right| * \left| \frac{y_3 - y_2}{x_3 - x_2 - 8} \right|} \quad (6)$$

- SPHI (Segment length distance relation; Table 1 #8): The length of the segment (Euclidean distance of all points) divided by the distance between the starting point and endpoint of the segment, (Equation (7));

$$SPHI = \frac{\sum_{i=1}^{np-1} \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2}}{\sqrt{(x_{np} - x_1)^2 + (y_{np} - y_1)^2}} \quad (7)$$

- HYP_ANGLE (Table 1 #9): The feature HYP_ANGLE determines the angle between the first and last point of the segment concerning the hypotenuse of the segment (see Figure 6, Equation (8));

$$\varphi = \arccos \left(\frac{\vec{a}_1 \cdot \vec{a}_2}{|\vec{a}_1| \cdot |\vec{a}_2|} \right) \quad (8)$$

where $\vec{a}_1 = \begin{pmatrix} x_1 \\ y_1 \end{pmatrix}$, $\vec{a}_2 = \begin{pmatrix} x_{np} \\ y_{np} \end{pmatrix}$, and $|\vec{a}_1| = \sqrt{x_1^2 + y_1^2}$
 $|\vec{a}_2| = \sqrt{x_{np}^2 + y_{np}^2}$

- REG_ANGLE (Table 1 #10): Determines the angle in relation to the regression straight of the segment (see Figure 7, Equation (9));

$$\alpha = \arctan \left(\frac{\sum_{i=1}^{np-1} (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{np-1} (x_i - \bar{x})^2} \right) \quad (9)$$

where n is the number of points in the segment and (x_i, y_i) are the coordinates of point i.

- Width height surface of a character sequence (see Figure 8) WORD_WIDTH (Table 1 #11) WORD_HEIGHT (Table 1 #12) and SURFACE (Table 1 #13);

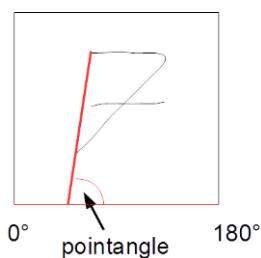


Figure 4. Example of point angle.

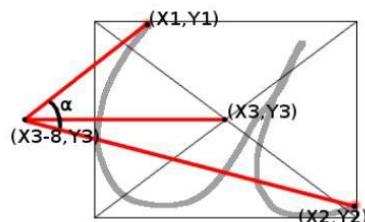


Figure 5. Example of horizontal point angle.

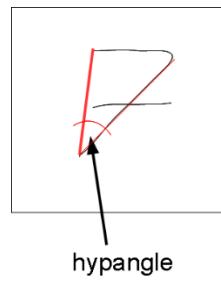


Figure 6. Representation of the parameter “HYP_ANGLE”.

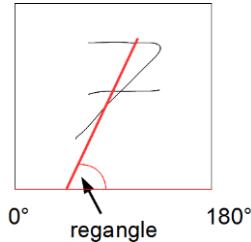


Figure 7. Representation of the parameter “REGANGLE”.

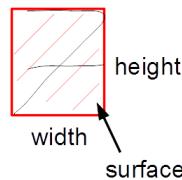


Figure 8. Width and Height of character sequence.

3.2. Statistical Features

The extracted different statistical features we have developed ourselves are described in more detail in the next paragraphs:

- POINTS (Table 1 #19): Is the number of the pixels n occupied for the signature on the display.
- SEGMENTS (Table 1 #20): consider the number of segments of the signature. A new segment begins when the display is touched by pencil, to begin the writing process, and ends when the pencil is removed from the display.
- The number of pen-ups NUM_STROKES (Table 1 #21); actually equivalent to the number of segments, hence it can be directly calculated according to the index in raw data.
- STANDARD_DERIVATION_x (Table 1 #22) or STANDARD_DERIVATION_Y (Table 1 #23): Standard derivation in x or y direction with number of all n points (see Equation (10));

$$sdY = \sqrt{\frac{1}{n-1} \sum_{i=1}^{n-1} (y_i - \bar{y})^2}; \quad sdx = \sqrt{\frac{1}{n-1} \sum_{i=1}^{n-1} (x_i - \bar{x})^2} \quad (10)$$

- MEAN_NUMBER_POINTS/SEGMENT (Table 1 #24): Average of all points per segment (see Equation (11));

$$NPS = \frac{\sum_{i=1}^{n-1} x_i}{\sum \text{no_segments}} \quad (11)$$

- DTW(x,y,t) (Table 1 #25,#26,#27): (Dynamic Time Warping Distance feature): The Dynamic Time Warping Distance is calculated by m and n, where the dimensional vectors are s[0 … m], t[0 … n]. The DTW Matrix has n rows, m columns and the first column and row are initialized by

infinity. For every pair s_i and t_j , the Euclidean distance is calculated, and the result is called cost. The minimum of $DTW_{i,j}$, $DTW_{i,j-1}$ and $DTW_{i-1,j-1}$ are added for the cost. This is done for every index in the DTW Matrix. The Result of the DTW is stored at DTW_{nm} . (see Equations (12) and (13));

$$DTW_{i,j} = f(s_i, t_j) + \min(DTW_{i-1,j}, DTW_{i,j-1}, DTW_{i-1,j-1}) \quad (12)$$

$$f(x, y) = |x - y| \quad (13)$$

For each point, the x-, y- and time stamp values are collected separately. After that, the vectors are split into two vectors. The DTW is calculated for every vector pair. (see Equation (14));

$$\begin{aligned} DTW_X &= DTW(x_1, x_2); \\ DTW_y &= DTW(y_1, y_2); \\ DTW_t &= DTW(t_1, t_2); \end{aligned} \quad (14)$$

- RELATION VX NEGATIVE (Table 1 #28) or RELATION VX POSITIVE (Table 1 #29): Total time of the positive or negative x movement in relation to the average speed (see Equation (15));

$$Vxpos = \frac{txpos}{v}, Vxneg = \frac{txneg}{v} \quad (15)$$

- RELATION VY NEGATIVE (Table 1 #30) or RELATION VY POSITIVE (Table 1 #31): Total time of the positive or negative y movement in relation to the average speed (see Equation (16));

$$V ypos = \frac{typos}{v}, Vyneg = \frac{tyneg}{v} \quad (16)$$

- MEDIAN X (Table 1 #32) or MEDIAN Y (Table 1 #33): Median of all points of the entire signature in the x or y direction (see Equation (17));

$$\tilde{x} = \begin{cases} \frac{x_{\frac{n+1}{2}}}{2} & n \text{ even} \\ \frac{1}{2}(x_{\frac{n}{2}} + x_{\frac{n}{2}+1}) & n \text{ odd} \end{cases}, \quad \tilde{y} = \begin{cases} \frac{y_{\frac{n+1}{2}}}{2} & n \text{ even} \\ \frac{1}{2}(y_{\frac{n}{2}} + y_{\frac{n}{2}+1}) & n \text{ odd} \end{cases} \quad (17)$$

- MIN VX (Table 1 #34) or MIN VY (Table 1 #35): Time when the lowest speed in x- or y-direction is reached.
- MAX VX (Table 1 #36) or MAX VY (Table 1 #37): Time when the highest speed in x- or y-direction is reached.
- RELATION VX MAX (Table 1 #38): The time of the highest horizontal speed in relation to the average speed. (see Equation (18));

$$RelVmax = \frac{Tvmax}{v} \quad (18)$$

- RELATION NVxz (Table 1 #39) or RELATION NVyz (Table 1 #40): Number of points Horizontal or Vertical depending on the average speed (see Equation (19));

$$NVxz = \frac{NSx}{v}, NVyz = \frac{NSy}{v} \quad (19)$$

- DWH (Table 1 #48): Division signature width/signature height.
- RELATION POINTS SPEED (Table 1 #49): Division number of all points of the signature/average speed.

- STAT_X_NUMBER_POINTS SEGMENT (Table 1 #50) or STAT_Y_NUMBER_POINTS SEGMENT (Table 1 #51): Division number of points x or y/number of all segments.

3.3. Temporal Features

The extracted different temporal features we have developed ourselves are described in more detail in the next paragraphs:

- TIME (Table 1 #52): Whole time user needs for one secure password sample or cursive text sample.
- The Relative Duration of Writing RELATIVE_WRITING_DURATION (Table 1 #53) defined as below (see Equation (20));

$$\frac{\text{Pen} - \text{Down Duration}}{\text{Total Duration}} \quad (20)$$

It can be calculated according to the timestamps and index in raw data.

- TIME_V_MAX (Table 1 #54): Point in time of overall maximum speed.
- RELATION_TIME_GAP_ALL (Table 1 #55): Relation Time to the number of gaps.
- TIME_MIN_X (Table 1 #56) or TIME_MIN_Y (Table 1 #57): Time t , when the minimum x or y point of the signature is reached (see Equations (21) and (22));

$$t_{\min x} = \sum_{i=1}^{n-1} t_i(x_{\min}, y_i) \quad (21)$$

$$t_{\min y} = \sum_{i=1}^{n-1} t_i(x_i, y_{\min}) \quad (22)$$

- TIME_MAX_X (Table 1 #58) or TIME_MAX_Y (Table 1 #59): Time t , when the maximum x or y point of the signature are reached (see Equations (23) and (24));

$$t_{\max x} = \sum_{i=1}^{n-1} t_i(x_{\max}, y_i) \quad (23)$$

$$t_{\max y} = \sum_{i=1}^{n-1} t_i(x_i, y_{\max}) \quad (24)$$

- TIME_VX_MAX (Table 1 #60) or TIME_VY_MAX (Table 1 #61): Time when the maximum speed x or y are reached (see Equation (25));

$$t_{\max V_y} = t(v_{y\max}), t_{\max V_x} = t(v_{x\max}) \quad (25)$$

- TIME_VX_MIN (Table 1 #62) or TIME_VY_MIN (Table 1 #63): Time when the minimum speed x or y are reached (see Equation (26));

$$t_{\min V_y} = t(v_{y\min}), t_{\min V_x} = t(v_{x\min}) \quad (26)$$

- TIME_X_POS (Table 1 #64) or TIME_Y_POS (Table 1 #65): the whole time of positive x or y movements (see Equation (27));

$$txpos = \sum_{i=1}^{n-1} Tx, x >= 0, typos = \sum_{i=1}^{n-1} Ty, y >= 0 \quad (27)$$

- TIME_X_NEG (Table 1 #66) or TIME_Y_NEG (Table 1 #67): whole time of negative x or y movements (see Equation (28));

$$txneg = \sum_{i=1}^{n-1} Tx, x < 0 \quad (28)$$

Using these extracted features, we performed the following experiments.

4. Methods for Feature Reduction and Classification System

This section includes the description of feature reduction methods and classification system methods applied on this approach in order to show the comparison and robustness of different methods.

4.1. Methods of Feature Reduction

Three methods are used and described for this proposal:

- Fisher Score: The Generalized Fisher Score [10] is a joint feature selection criterion, which aims at finding a subset of features and maximizes the lower bound of the traditional Fisher Score. It also resolves redundant problems in the feature selection process. The mathematical description of the Fisher Score is shown as below in Equation (29).

$$F(X^j) = \frac{\sum_{i=1}^N n_i \cdot (\mu_i^j - \mu^j)^2}{(\sigma^j)^2} \quad (29)$$

$$\sigma^j = \sqrt{\sum_{i=1}^N n_i (\sigma_i^j)^2}$$

where,

j —the j -th feature.

i —the i -th class, which could be interpreted as the i -th subject in our test.

n —the size of the instances for a certain class

μ —the mean value for a certain class

σ —the standard deviation for a certain class

- Correlation analysis: Although there are many attributes which are correlated, this proposal removes attributes which have a correlation coefficient above ± 0.9 . For this purpose, all attributes will be compared with each other. However, before an attribute gets discarded, we compare the Fisher Score of both attributes. If the score of the actual attributing better than the other score of the compared attribute, then the attribute gets marked for removal. When all other attributes are tested and no other attribute is better than the actual, all marked attributes get removed. However, if there is one Attribute which is better than the actual attribute, then it gets removed and the marked ones get unmarked.
- Info Gain Attribute Evaluation: At second, Information Gain Attribute Evaluation (IG) [44] is used for ranking. This ranker evaluates the worth of an attribute by measuring the information gain with respect to the class. The mathematical description of Information Gain Attribute Evaluation is shown as below in Equation (30).

$$IG = H(Y) - H(Y/X) = H(X) - H(X/Y)$$

$$H(Y) = - \sum_{y \in Y} p(y) \log_2(p(y)) \quad (30)$$

$$H(Y/X) = - \sum_{x \in X} p(x) \sum_{y \in Y} p(y/x) \log_2(p(y/x))$$

In which,

$H(Y)$ is the entropy of Y .

$H(Y/X)$ is the entropy of Y after observing X .

$p(y)$ is the marginal probability density function for the random variable Y .

$p(y/x)$ is the conditional probability of y given x .

The same test procedure was performed for the extracted features of the handwritten signature from the IAM online handwriting database.

4.2. Methods of Classification Systems

- KNN: For KNN (value of $k = 1$ for all experiments), we use the distance between an instance and the centroid of a class as the decision threshold. To calculate the ROC-Curve, we determine the greatest distance over all test instances. Then we alter the threshold from 0 to this determined greatest value. When the distance of a test instance is greater than the current decision threshold, we refuse the user. For every altered threshold, we calculate the TPR and FPR.
- Naïve Bayes and Bayes Net: For every instance, we calculate the class probability. If the probability is higher than the threshold, the user is accepted. As higher class probabilities occur more often than lower probabilities, we use a higher resolution for higher probabilities.

5. Experimental Methodology

The idea is to apply the proposal on two datasets with different kinds of data in order to show its robustness and innovation. One of them is ours (Secure Password DB 150), and the approach was developed on it. The second dataset is public, and it was used to validate the proposal (IAM online handwriting database). The experimental protocol implements two strategies, K-fold cross validation (Cross 10) method and hold-out cross validation (Hold-out 66%) method for three classifiers, in order to show the robustness of the proposal. Besides, it is implemented under identification and verification stages. The verification stage is measured by receiver operating characteristic (ROC) curves and the recognition stage is measured by the success percentage of the confusion matrix. All experiments are running in an i5 core processor using WEKA tool [54]. The computational time is shown and measured by the average (AVG); time is shown in seconds (s.).

During the experimental process, the implementation of all experiments was sequentially led. The first step was to check the accuracy of each type of feature; and finally, the accuracy for all features. The second step was to apply feature-reduction methods in order to improve the accuracy of the proposal. Transversally, they were implemented for the three classifiers and all strategies and stages.

About the specific experiments, the first experiments start with the analysis of the Secure Password DB 150. At first, the classification of the handwritten passwords by using the different feature types: geometrical, statistical and temporal features separately is carried out. Afterwards, we applied all 67 features for the writer verification.

About the quality measure, the approach applies ROC curves for estimating a threshold for discriminating genuine from imposter, because the experiments are based on a verification approach. On the x-axis is the false positive rate (FPR) and on the y-axis the true positive rate (TPR). The FPR and TPR are given in percent and are calculated as shown below (see Equations (31) and (32)):

$$TPR = \frac{TP}{TP + FN} \quad (31)$$

$$FPR = \frac{FP}{TN + FP} \quad (32)$$

where the variables are defined on the following confusion matrix (see Figure 9).

The TPR told us how many of the genuine Users are correctly classified as a genuine User. The FPR told us how many imposters are classified incorrectly as genuine Users. For estimating a threshold,

the proposal uses for KNN the distance and for Naïve Bayes and Bayes Net applies the class probability. The next step is to develop a procedure to choose the optimal threshold based on the ROC-Curve.

	Genuine (OUT)	Imposter (OUT)
Genuine (IN)	True positives (TP)	False positives (FP)
Imposter (IN)	False negatives (FN)	True negatives (TN)

Figure 9. Confusion matrix in a verification stage and definition of each variable.

Since we have a multiclass problem, we need to simplify our dataset to a binary class problem. For generating the ROC-Curve, we worked with a real forgery dataset. Therefore, we have different writers for the same password. The password is randomly generated. The imposter knows the password. Thus, a real scenario where a forger tried to hack a password was implemented and tested.

For both methods with genuine users and imposters, the approach trains a classifier with 66% of genuine samples and the rest of the samples (34%) are used to test it; then, TPR is calculated. A tested sample is a true positive, when the classifier predicts the correct class label and the instance matches the acceptance rule. An imposter is a false positive, when the instance matches the acceptance rule, no matter if the predicted class label is the correct class label for the forged instance.

For another robustness test, we calculate the ROC-Curve applying a k-fold cross validation, in particular for K equal to 10. As a result, we plot the area between the maximum and minimum ROC-Curves and the average of all curves. The area is colored grey and the average is colored blue. The points on the blue line are the points we calculate for different thresholds. The horizontal red line marks the 90% TPR and the vertical red line marks the 10% FPR. In the results section, these figures can be observed.

6. Experiments and Results

The experiments evaluate the grade of discrimination for the type of features, for the set of all features and for the feature reduction of the whole feature set. The results show the grade of the robustness and its analysis.

6.1. Results for Different Feature Sets in Secure Password DB 150

The following tables show that the results first split to the feature type's geometrical, statistical, temporal, pressure and finally all features for the secure password DB 150.

- Geometrical features

Table 2 shows results for geometrical features in the secure passwords. Naïve Bayes classifier achieves the best result for the K-fold cross validation method. It seems that the geometrical features show low results in comparison with the state-of-the-art methods. The computational times are good for application.

Table 2. Results using geometrical features for Secure Password DB 150.

Classifiers	Bayes Net	Naïve Bayes	KNN
Cross 10	66.55%	81.23%	73.25%
Hold-out 66%	57.32%	78.39%	69.34%
AVG (Time in s.)	0.9	0.03	0.01
# of features	18/67	18/67	18/67

- Statistical features

In Table 3, it is shown that statistical features deliver better results than geometrical features. K-nearest neighbor classifier delivers the best result for classification.

Table 3. Results using statistical features for Secure Password DB 150.

Classifiers	Bayes Net	Naïve Bayes	KNN
Cross 10	80.23%	83.52%	91.15%
Hold-out 66%	72.29%	80.19%	90.41%
AVG (Time in s.)	2.45	0.05	<10 ⁻²
# of features	33/67	33/67	33/67

- Temporal features

Table 4 shows that temporal features do not deliver the best results; the best classifier is the k-nearest neighbor classifier closely followed by Naïve Bayes classifier for cross validation. The time for the classification is the same as before.

Table 4. Results using temporal features for Secure Password DB 150.

Classifiers	Bayes Net	Naïve Bayes	KNN
Cross 10	58.31%	69.13%	72.24%
Hold-out 66%	36.04%	65.44%	69.02%
AVG (Time in s.)	0.96	0.03	<10 ⁻²
# of features	16/67	16/67	16/67

- Using all features

In Table 5, it is shown that using all features contributes once again to an improvement of the classification results. Although geometrical and statistical features individually achieved not so good results, by the use of all features, the classification rate grows nearly about 2% for the best result. KNN protrudes again with the best result for the cross validation. Using all features has a negative effect on the average time for the classification. It is not surprising, either, however, that with fewer features, the system needs less time for classification. However, for the practical application, this time should be reduced under the retention of the good classification results. Table 5 shows some Fisher Scores for features.

Table 5. Results using all features for Secure Password DB 150.

Classifiers	Bayes Net	Naïve Bayes	KNN
Cross 10	89.83%	89.15%	93.62%
Hold-out 66%	85.04%	85.67%	92.00%
AVG (Time in s.)	4.71	0.09	0.02
# of features	67/67	67/67	67/67

- Applying feature reduction with Fisher Score and Correlation analysis:

The first step is to apply the Fisher Scores, and later, it is applied to a quartile of 30%. Every attribute which has a lower score than the 30% quantile is removed from the dataset. After that, the authors apply the correlation matrix. If the correlation of two attributes is bigger than ± 0.9 , authors remove the attribute with the lower score. The colored rows in grey mark the final chosen attributes (see Table 6), and then, the 37 attributes are selected.

In Table 7, it is demonstrated that after the reduction of features with Fisher Score, the results for K-Nearest Neighbor classifier are a bit better. The average (AVG) time for classification is nearly the same, because of the high number of features for classification.

- Applying feature reduction with info gain attribute select

Table 8 shows the result for the use of “Info Gain Attribute Select” from Weka to reduce the number of features. The ranker selected 37 features for classification. The Attributes with bold letter were chosen by Fisher Score and “Info Gain Attribute Select”.

Table 6. Fisher Scores Secure Password DB 150 Attributes which have a greater score than the 30% quantile in Secure Password DB 150.

Index	Parameter	Index	Parameter
7.964	STAT_STANDARD_DERIVATION_Y	3.089	STAT_VERTICAL_SKewness
7.281	GEO_HEIGHT	3.058	STAT_MAX_VY
7.044	GEO_SPHI	2.788	STAT_MIN_VY
	GEO_WORD_HEIGHT		
	Correlate with STAT_STANDARD_DERIVATION_Y (0.97)	2.546	STAT_Y_NUMBER_POINTS/SEGMENT
6.975	STAT_ASPECT_RATIO		
	Correlate with STAT_STANDARD_DERIVATION_Y (-0.98)	2.381	TEMP_RELATION_TIME_GAP_ALL
6.709	STAT_INERTIAL_RATIO		
	Correlate with STAT_STANDARD_DERIVATION_Y (-0.93)	2.197	STAT_MIN_VX
5.997	STAT_STANDARD_DERIVATION_Y		
5.894	TEMP_TIME_Y_NEG	2.102	STAT_SEGMENTS
5.831	TEMP_TIME_Y_POS	2.102	Correlate with STAT_NUM_STROKES (1.0)
5.590	TEMP_TIME_X_NEG	2.099	STAT_NUM_STROKES
5.582	STAT_MEDIAN_Y	2.029	STAT_HORIZONTAL_SKewness
5.510	GEO_SURFACE	2.009	STAT_X_NUMBER_POINTS/SEGMENT
5.447	STAT_SPREADNESS	1.717	STAT_MAX_VX
5.402	GEO_WIDTH	1.579	STAT_BALANCE_HORIZONTAL_EXTENSION
5.254	GEO_CENTRAL_POINT	1.529	STAT_BALANCE_VERTICAL_EXTENSION
	Correlate with TEMP_TIME_Y_POS (0.92)		STAT_RELATION_VX_NEGATIVE
5.173	GEO_EUCLID	1.511	STAT_RELATION_VY_POSITIVE
			Correlate with STAT_RELATION_VX_NEGATIVE (0.93)
5.005	STAT_POINTS	1.458	TEMP_RELATIVE_WRITING_DURATION
4.919	STAT_MEAN_NUMBER_POINTS/SEGMENT	1.401	STAT_RELATION_NVxz
	Correlate with STAT_POINTS (0.95)		
4.801	TEMP_TIME_X_POS	1.393	STAT_STANDARD_DERIVATION_X
4.743	GEO_REGRESSION_ON_HORANGLE	1.366	STAT_DTW_X
	GEO_REGRESSION_LOWER_HORANGLE		
	Correlate with GEO_REGRESSION_ON_HORANGLE (0.99)	1.350	GEO_SLANT
4.740			
4.607	TEMP_TIME	1.328	STAT_RELATION_VX_POSITIVE
	Correlate with STAT_POINTS (0.95)		
	STAT_DTW_Y		STAT_RELATION_POINTS_SPEED
4.402	Correlate with STAT_STANDARD_DERIVATION_Y (0.92)	1.325	Correlate with STAT_RELATION_VX_NEGATIVE (0.93)
4.312	GEO_HYP_ANGLE	1.290	STAT_RELATION_VY_NEGATIVE
3.262	GEO_REGRESSION_UPPER_HORANGLE	1.280	STAT_RELATION_NVyz

Table 7. Results using Fisher Score of all features for Secure Password DB 150.

Classifiers	Bayes Net	Naïve Bayes	KNN
Cross 10	88.47%	89.68%	95.38%
Hold-out 66%	85.46%	89.36%	94.42%
AVG (Time in s.)	2.81	0.11	0.02
# of features	37/67	37/67	37/67

Table 8. Ranked Result for Info Gain Attribute Select on Secure Password DB 150.

Index	Parameter	Index	Parameter
2.118	STAT_STANDARD_DERIVATION_Y	1.363	GEO_SURFACE
2.000	STAT_ASPECT_RATIO	1.331	STAT_POINTS
1.909	GEO_WORD_WITH	1.307	STAT_MEAN_NUMBERS/SEGMENTS
1.875	STAT_INERTIAL_RATIO	1.274	GEO_ORIENTATION
1.568	GEO_SPHI	1.274	GEO_REGRESSION_UPPER_HORANGLE
1.513	TEMP_TIME_Y_NEG	1.273	GEO_EUCLID
1.500	GEO_WIDTH	1.212	TEMP_TIME

Table 8. Cont.

Index	Parameter	Index	Parameter
1.472	STAT_NUM_STROKES	1.211	STAT_VERTICAL_SKEWNESS
1.472	STAT_SEGMENTS	1.104	GEO_HYP_ANGLE
1.461	STAT_MEDIAN_Y	1.057	STAT_MAX_VY
1.459	TEMP_TIME_Y_POS	0.955	TEMP_TIME_VY_MIN
1.452	STAT_DTW_Y	0.880	TEMP_TIME_MAX_Y
1.419	GEO_REGRESSION_ON_HORANGLE	0.859	STAT_DWH
1.412	GEO_REGRESSION_LOWER_HORANGLE	0.829	STAT_MIN_VY
1.384	STAT_SPREADNESS	0.785	STAT_Y_NUMBER_POINTS/SEGMENT
1.384	TEMP_TIME_X_NEG	0.760	STAT_DTW_X
1.373	GEO_CENTRAL_POINT	0.747	TEMP_RELATION_TIME_GAP_ALL
1.366	TEMP_TIME_X_POS	0.729	STAT_RELATION_POINTS_SPEED

In Table 9, it is demonstrated that after ranking with “Info Gain Attribute Select”, the results for Naïve Bayes and K-Nearest Neighbor classifier are a little worse than before. By the feature reduction, the classification time considerably decreases to AVG of <0.01 s for KNN.

Table 9. Results using Info Gain Attribute Select of all feature features for Secure Password DB 150.

Classifiers	Bayes Net	Naïve Bayes	KNN
Cross 10	86.89%	87.86%	92.55%
Hold-out 66%	82.61%	86.09%	91.68%
AVG (Time in s.)	2.63	0.05	<10 ⁻²
# of features	37/67	37/67	37/67

- ROC Analysis

Figure 10 shows the ROC-Curve for KNN classifier. KNN can classify about 60% genuine correct and reject any impostor. About 93% TPR of all impostors were accepted.

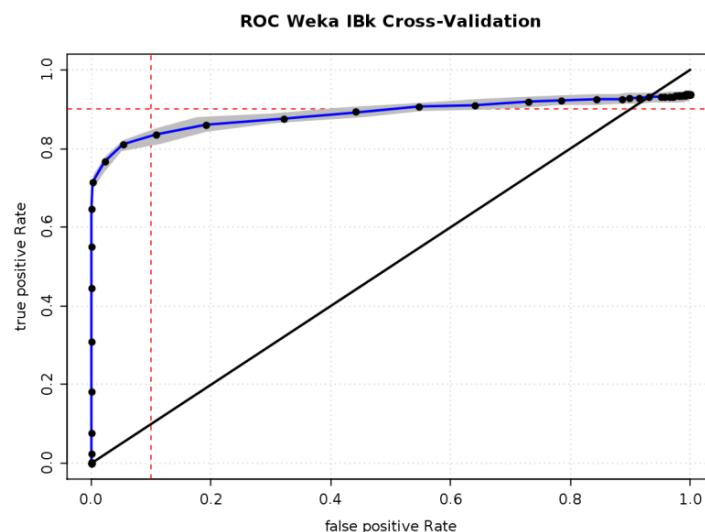
**Figure 10.** ROC-Curve for KNN classifier.

Figure 11 on the next page shows the ROC-Curve for the Naïve Bayes classifier. The curve has a long linear interpolated part from origin (100% class probability) to the second threshold (99% class probability). About 60% of the impostors and about 80% of the genuine users have a class probability greater than 99%.

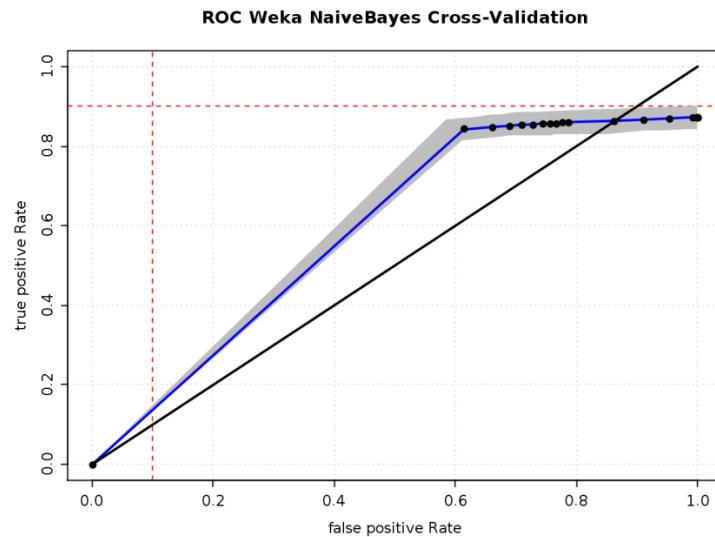


Figure 11. ROC-Curve for Naïve Bayes classifier.

Figure 12 shows the ROC-Curve for Bayes Net classifier. Authors need to decrease the step size for the threshold at a certain point to get a better resolution.

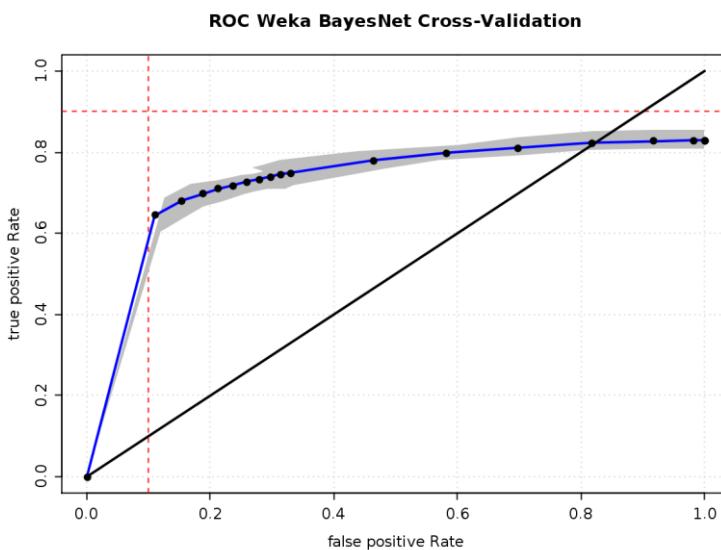


Figure 12. ROC-Curve for Bayes Net classifier.

Note that Naïve Bayes and Bayes Net never lead to low TPR, resulting in a long linearly interpolated section of the ROC-curve starting from the origin; it is quite difficult to compare the curves of them with KNN. However, at least we can say that none of the classifiers alone can classify the user and classify the genuine user and imposter with an acceptable trade-off. The results show that, possibly, it is a good idea to combine the classifiers to minimize the problem. For example: Naïve Bayes can reject nearly 40% of the impostors without rejecting too many genuine users. This would minimize the Problem and could maybe help to improve results for KNN and Bayes Net.

6.2. Results for Different Feature Sets in IAM Online Handwriting DB

Secondly, the authors show the results of the feature analysis of the handwritten passwords on a public dataset in order to validate the proposal. It is the well-known IAM online handwriting DB.

- Geometrical features

In Table 10, it is shown that the best result of 54.62% correct classification we achieved with the Naïve Bayes classifier. To sum it up: We reach only average results for the geometrical feature.

Table 10. Results using geometrical features for IAM Online Handwriting DB.

Classifiers	Bayes Net	Naïve Bayes	KNN
Cross 10	17.8%	54.62%	34.10%
Hold-out 66%	9.81%	34.53%	28.11%
AVG (Time in s.)	0.45	0.01	<10 ⁻²
# of features	18/67	18/67	18/67

- Statistical features

In Table 11 it is shown that the results of statistical features are a little bit worse than the geometrical features' best result with 35.32% correct classification reached by the Naïve Bayes classifier for cross validation. The time for classification is nearly the same as the geometrical features.

Table 11. Results using Statistical features for IAM Online Handwriting Database.

Classifiers	Bayes Net	Naïve Bayes	KNN
Cross 10	16.54%	35.32%	10.06%
Hold-out 66%	10.75%	20.38%	10.38%
AVG (Time in s.)	0.44	0.02	<10 ⁻²
# of features	33/67	33/67	33/67

- Temporal features

In Table 12 it is shown that the temporal features achieved the best result of 96.15% correctly classified for Naïve Bayes classifier with cross validation. The maximum time for classification is with 0.5 s, only a bit longer than the time for the other features.

Table 12. Results using temporal features for IAM Online Handwriting Database.

Classifiers	Bayes Net	Naïve Bayes	KNN
Cross 10	95.84%	96.15%	58.21%
Hold-out 66%	89.43%	90.94%	48.87%
AVG (Time in s.)	0.5	<10 ⁻²	<10 ⁻²
# of features	16/67	16/67	16/67

- Using all features

All features were used for getting the results of Table 13. A small increase is found in comparison with the temporal features. The best classifier is Bayes Net with 98.65% correct classification. The time for the classification lies with 1.41 s. still within the scope of state-of-the-art methods.

Table 13. Results using all features for IAM Online Handwriting Database.

Classifiers	Bayes Net	Naïve Bayes	KNN
Cross 10	98.65%	93.53%	36.03%
Hold-out 66%	94.53%	80.57%	31.70%
AVG (Time in s.)	1.41	0.05	<10 ⁻²
# of features	67/67	67/67	67/67

- Applying feature reduction with Fisher Score

The ranked results of Fisher Score Reduction for IAM online handwriting DB are shown in Table 14. The ranker selects 59 features for classification. The best score is clearly achieved with time features, followed by geometrical and statistical features.

Table 14. Ranked parameters using Fisher Score for IAM Online Handwriting DB.

Index	Parameter	Index	Parameter
344,702,213	TIME_MAX_Y	9328	MEAN_NUMBER_POINTS SEGMENT
344,309,419	TIME_V_MAX	8913	RELATION_VX_MAX
340,892,335	TIME_VX_MAX	6362	WIDTH
337,972,869	TIME_VY_MAX	5661	REG_ANGLE
329,873,423	TIME_VY_MIN	3451	X_NUMBER_POINTS SEGMENT
328,062,080	TIME_MAX_X	3047	RELATION_NVyz
328,062,080	TIME_MIN_X	2832	RELATIVE_WRITING_DURATION
327,360,471	TIME_MIN_Y	2726	RELATION_NVxz
323,060,762	TIME_VX_MIN	2645	HEIGHT
18,411	SLANT	2597	Y_NUMBER_POINTS SEGMENT
9381	TIME	2540	POINT_ANGLE
9348	POINTS		

The Fisher Score Reduction in Table 15 demonstrates that the results for Naïve Bayes and K-Nearest Neighbor are a bit better than before. By the reduction of the features, the AVG Time decreases to 1.3 s for Bayes Net classifier.

Table 15. Results using Fisher Score of all feature for IAM Online Handwriting DB.

Classifiers	Bayes Net	Naïve Bayes	KNN
Cross 10	98.65%	96.92%	64.42%
Hold-out 66%	94.53%	85.85%	59.10%
AVG (Time in s.)	1.3	0.04	$<10^{-2}$
# of features	59/67	59/67	59/67

- Applying feature reduction with info gain Attribute select

Table 16 presents the ranked result of Info Gain Attribute Select and the ranker selected 22 features for the classification.

Table 16. Raked parameters using Info Gain Attribute Select for IAM Online Handwriting DB.

Index	Parameter	Index	Parameter
7.4759	TIME_MIN_Y	7.4719	TIME_VX_MIN
7.4754	TIME_MIN_X	1.9984	SLANT
7.4754	TIME_MAX_X	1.9317	RELATION_VX_MAX
7.4744	TIME_VX_MAX	1.5304	WIDTH
7.4737	TIME_MAX_Y	1.526	TIME
7.4737	TIME_V_MAX	1.4895	POINTS
7.4732	TIME_VY_MIN	1.0784	MEAN_NUMBER_POINTS SEGMENT
7.4727	TIME_VY_MAX		

Table 17 shows that the result improved for Naïve Bayes and K-Nearest Neighbor classifier again in spite of the reduction on only 22 features. The time becomes considerably shorter with only 0.62 s for Bayes Net classifier for classification after ranking with Info gain Attribute Select.

Table 17. Results using Info Gain Attribute Select of all feature for IAM Online Handwriting DB.

Classifiers	Bayes Net	Naïve Bayes	KNN
Cross 10	98.65%	98.34%	87.44%
Hold-out 66%	94.53%	93.40%	82.26%
AVG (Time in s.)	0.62	0.01	$<10^{-2}$
# of features	22/67	22/67	22/67

This dataset does not present impostors due to its building, different texts, paragraphs and syntax; therefore, this proposal could not compute a meaningful ROC-Curve. It is easy to understand that if it randomly takes other genuine samples as impostors, this method will not lead to a meaningful curve. Most of the impostors can be easily rejected and we get ROC-Curves with the exact 0% FPR for every threshold.

Finally, in order to show the robustness of this proposal, it is presented in a comparison table with the most representative references, which uses both dataset, the Secure Password DB 150 and the IAM online dataset (see Table 18). Table 18 is split into two parts, one part for private datasets and the second part for the IAM dataset. Some references apply different protocols, but with better conditions than this proposal. The idea is to show the value of accuracies vs. the proposal. Besides, the applied methods and their results are observed in order to check the behavior of each reference versus this proposal. In both cases, this work reaches a good answer for both datasets and it indicates its robustness.

Table 18. Comparison between the actual work and the state-of-the-art.

Reference	Method	Dataset (# User)	Accuracy
[10]	Handwritten character and numerals using an Adaptive Radial Basis Function Network	Private (10)	97%
[42]	39 statistical parameters classified by Bayes Net	Private (32)	96.87%–100%
This work	Fisher Score reduction method of 67 statistical, geometrical and temporal features, classified with KNN for password (set of characters)	Private (150)	95.38%
[46]	Codebook descriptors on text line level	IAM	89.92%
[52]	Gaussian mixture models on text line level	IAM (200)	88.96%
[55]	Sparse work frame using a traditional SVM on text line level	IAM	90.28%
This work	Fisher Score reduction method of 67 statistical, geometrical and temporal features, classified with Naïve Bayes for words	IAM (220)	98.34%

The most significant comparison is the type of level. The most of the papers, which work on IAM dataset, apply the writer identification for text line level and paragraph level. Our work can be applied to handwritten passwords or/and continuous writing, being an added value of this proposal and even, reaching good accuracies for both datasets.

7. Conclusions

This work presents a feature set for writer identification and verification by online handwriting data. The innovation of the work is on the concatenation and fusion of the set of 67 features and its reduction in order to improve the accuracy for verification and identification stages.

The feature set is applied on two different types of handwritten data. The proposal of the feature set is the innovation of the proposal, which can be applied with a good answer for different handwritten data, in particular, handwritten passwords composed on isolated characters and as well as handwritten cursive texts. After feature reduction, the results for KNN also improved. In summary, it can be stated that both the feature reduction and the classification method have a strong influence on the results for both data sets. Only there is to apply the adequate feature reduction method to each dataset, due to the nature of the both writing is different. IAM Online Handwriting DB is continuous writing and Secure Password DB 150 are isolated characters (a password). The best results delivered 95.38% correct classification for handwritten single character words using the Secure Password DB 150; about 93% TPR for real impostor test; and 98.34% for cursive texts using IAM Online Handwriting DB. The proposal is considered a strong and novel proposal, after to review the state-of-the-art.

The results of this work depend on the different parameter types and data types. However, the whole feature set for both datasets presents a similar and good answer and is very significant in

comparison with the state-of-the-art due to the relation numbers of users and accuracy. In general, the feature reduction improves the accuracies of all features. In particular, the Fisher Score method reaches the best accuracy. Therefore, it is concluded that the proposal of this feature set shows a strong answer for the writer identification as can be observed in Table 18.

For future work, authors will generate an extended feature set in order to improve the actual proposal.

Author Contributions: Conceptualization, T.K., I.B. and C.M.T.; methodology, T.K., I.B. and C.M.T.; software, T.B., M.G. and W.-D.P.; validation, T.K., I.B. and C.M.T.; formal analysis and results, T.K., C.F.P.-Z., I.B. and C.M.T.; writing—original draft preparation, T.B.; writing—review and editing, C.F.P.-Z. and C.M.T.; supervision, I.B. and C.M.T.; funding acquisition, C.M.T.

Funding: This research is funded by “Ministerio de Ciencia, Innovación y Universidades” from The Spanish Government, under Grant “Estancias de profesores e investigadores senior en centros extranjeros, incluido en el Programa Salvador de Madariaga 2018” with the reference PRX18/00423. Too, this work was partially funded by the government of Spain, grant number TEC2016-77791-C4-1-R.

Conflicts of Interest: The authors declare no conflicts of interest. The founding sponsors had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, and in the decision to publish the results.

References

1. Bouadjenek, N.; Nemmour, H.; Chibani, Y. Fuzzy Integral for Combining SVM-based Handwritten Soft-biometrics Prediction. In Proceedings of the 2016 12th IAPR Workshop on Document Analysis Systems (DAS), Santorini, Greece, 11–14 April 2016; pp. 311–316. [[CrossRef](#)]
2. Gattal, A.; Djeddi, C.; Siddiqi, I.; Chibani, Y. Gender classification from offline multi-script handwriting images using oriented Basic Image Features (oBIFs). *Expert Syst. Appl.* **2018**, *99*, 155–167. [[CrossRef](#)]
3. He, S.; Schomaker, L. General Pattern Run-Length Transform for Writer Identification. In Proceedings of the 2016 12th IAPR Workshop on Document Analysis Systems (DAS), Santorini, Greece, 11–14 April 2016; pp. 60–65. [[CrossRef](#)]
4. Frery, J.; Largeron, C.; Juganaru-Mathieu, M. Author Identification by Automatic Learning. In Proceedings of the 2015 13th International Conference on Document Analysis and Recognition (ICDAR), Tunis, France, 23–26 August 2015; pp. 181–185. [[CrossRef](#)]
5. Pal, S.; Alaei, A.; Pal, U.; Blumenstein, M. Performance of an Off-line Signature Verification Method based on Texture Features on a Large Indic-script Signature Dataset. In Proceedings of the 2016 12th IAPR Workshop on Document Analysis Systems (DAS), Santorini, Greece, 11–14 April 2016; pp. 72–77. [[CrossRef](#)]
6. Fischer, A.; Diaz, M.; Plamondon, R.; Ferrer, M.A. Robust Score Normalization for DTW-Based On-Line Signature Verification. In Proceedings of the 2015 13th International Conference on Document Analysis and Recognition (ICDAR), Tunis, France, 23–26 August 2015; pp. 241–245. [[CrossRef](#)]
7. Impedovo, D.; Pirlo, G.; Plamondon, R. Handwritten signature verification: New advancements and open issues. In Proceedings of the 2012 International Conference on Frontiers in Handwriting Recognition, Bari, Italy, 18–20 September 2012; pp. 367–372. [[CrossRef](#)]
8. Adak, C.; Chaudhuri, B.B. Writer Identification from Offline Isolated Bangla Characters and Numerals. In Proceedings of the 2015 13th International Conference on Document Analysis and Recognition (ICDAR), Tunis, France, 23–26 August 2015; pp. 486–490. [[CrossRef](#)]
9. Raje, S.; Mehrotra, K.; Belhe, S. Writer Adaptation of Online Handwritten Recognition using Adaptive RBF Network. In Proceedings of the 2015 13th International Conference on Document Analysis and Recognition (ICDAR), Tunis, France, 23–26 August 2015; pp. 691–695. [[CrossRef](#)]
10. Obaidullah, S.M.; Halder, C.; Das, N.; Roy, K. A new dataset of word-level offline handwritten numeral images from four official Indic scripts and its benchmarking using image transform fusion. *Int. J. Intell. Eng. Inform.* **2016**, *4*, 1–20. [[CrossRef](#)]
11. Narwade, P.N.; Sawant, R.R.; Bonde, S.V. Offline Handwritten Signature Verification Using Cylindrical Shape Context. *3D Res.* **2018**, *9*, 48. [[CrossRef](#)]
12. Alpar, O.; Krejcar, O. Online signature verification by spectrogram analysis. *Appl. Intell.* **2018**, *48*, 1189–1199. [[CrossRef](#)]

13. Hafemann, L.G.; Oliveira, L.S.; Sabourin, R. Fixed-sized representation learning from offline handwritten signatures of different sizes. *Int. J. Doc. Anal. Recognit.* **2018**, *21*, 219–232. [[CrossRef](#)]
14. Taskiran, M.; Cam, Z.G. Offline Signature Identification via HOG Features and Artificial Neural Networks. In Proceedings of the 2017 15th International Symposium on Applied Machine Intelligence and Informatics (SAMI), Herl’any, Slovakia, 26–28 January 2017; pp. 83–86.
15. Pesch, H.; Hamdani, M.; Forster, J.; Ney, H. Analysis of Preprocessing Techniques for Latin Handwriting Recognition. In Proceedings of the 2012 13th International Conference on Frontiers in Handwriting Recognition (ICFHR), Bari, Italy, 18–20 September 2012; pp. 280–284. [[CrossRef](#)]
16. Balbed, M.A.M.; Ahmad, S.M.S.; Shakil, A. ANOVA-Based Feature Analysis and Selection in HMM-Based Offline Signature Verification System. In Proceedings of the 2009 Innovative Technologies in Intelligent Systems and Industrial Applications CITISIA, Monash, Malaysia, 25–26 July 2009; pp. 66–69. [[CrossRef](#)]
17. Ahmed, K.; El-Henawy, I.M.; Rashad, M.Z.; Nomir, O. On-line signature verification based on PCA feature reduction and statistical analysis. In Proceedings of the 2010 International Conference on Computer Engineering & Systems (ICCES), 30 November–2 December 2010; pp. 3–8. [[CrossRef](#)]
18. Indermühle, E.; Liwicki, M.; Bunke, H. Combining Alignment Results for Historical Handwritten Document Analysis. In Proceedings of the 2009 10th International Conference on Document Analysis and Recognition, Barcelona, Spain, 26–29 July 2009; pp. 1186–1190. [[CrossRef](#)]
19. Fischer, A.; Bunke, H. Character prototype selection for handwriting recognition in historical documents. In Proceedings of the 2011 19th European Signal Processing Conference, Barcelona, Spain, 29 August–2 September 2011; pp. 1435–1439.
20. Gattal, A.; Djeddi, C.; Chibani, Y.; Siddiqi, I. Isolated Handwritten Digit Recognition Using oBIFs and Background Features. In Proceedings of the 2016 12th IAPR Workshop on Document Analysis Systems (DAS), Santorini, Greece, 11–14 April 2016; pp. 305–310. [[CrossRef](#)]
21. Jain, R.; Doermann, D. Combining Local Features for Offline Writer Identification. In Proceedings of the 2014 14th International Conference on Frontiers in Handwriting Recognition, Heraklion, Greece, 1–4 September 2014; pp. 583–588. [[CrossRef](#)]
22. Simistira, F.; Katsouros, V.; Carayannis, G. Recognition of online handwritten mathematical formulas using probabilistic SVMs and stochastic context free grammars. *Pattern Recognit. Lett.* **2015**, *53*, 85–92. [[CrossRef](#)]
23. Shaw, B.; Bhattacharya, U.; Parui, S.K. Combination of Features for Efficient Recognition of Offline Handwritten Devanagari Words. In Proceedings of the 2014 14th International Conference on Frontiers in Handwriting Recognition, Heraklion, Greece, 1–4 September 2014; pp. 240–245. [[CrossRef](#)]
24. Jang, W.; Kim, S.; Kim, Y.; Lee, E.C. Automated Verification Method of Korean Word Handwriting Using Geometric Feature. In *Advances in Computer Science and Ubiquitous Computing. CUTE 2017, CSA 2017*; Park, J., Loia, V., Yi, G., Sung, Y., Eds.; Lecture Notes in Electrical Engineering; Springer: Singapore, 2018; Volume 474, pp. 1340–1345.
25. Gupta, J.D.; Samanta, S.; Chanda, B. Ensemble classifier-based off-line handwritten word recognition system in holistic approach. *IET Image Process.* **2018**, *12*, 1467–1474. [[CrossRef](#)]
26. Giotis, A.R.; Gerogiannis, D.R.; Nikou, C. Word Spotting in Handwritten Text Using Contour-based Models. In Proceedings of the 2014 14th International Conference on Frontiers in Handwriting Recognition, Heraklion, Greece, 1–4 September 2014; pp. 399–404. [[CrossRef](#)]
27. Griechisch, E.; Malik, M.I.; Liwicki, M. Online Signature Verification based on Kolmogorov-Smirnov Distribution Distance. In Proceedings of the 2014 14th International Conference on Frontiers in Handwriting Recognition, Heraklion, Greece, 1–4 September 2014; pp. 738–742. [[CrossRef](#)]
28. Wibowo, C.P.; Thumwarin, P.; Matsuura, T. On-line Signature Verification Based on Forward and Backward Variances of Signature. In Proceedings of the 2014 The 4th Joint International Conference on Information and Communication Technology, Electronic and Electrical Engineering (IJCTEE), Chiang Rai, Thailand, 5–8 March 2014; pp. 1–5.
29. Slim, M.A.; Abdelkrim, A.; Benrejeb, M. An Efficient Handwriting Velocity Modelling for Electromyographic Signals Reconstruction Using Radial Basis Function Neural Networks. In Proceedings of the 2015 7th International Conference on Modelling, Identification and Control (ICMIC), Sousse, Tunisia, 18–20 December 2015; pp. 71–76. [[CrossRef](#)]

30. Davila, K.; Ludi, S.; Zanibbi, R. Using Off-line Features and Synthetic Data for On-line Handwritten Math Symbol Recognition. In Proceeding of the 2014 14th International Conference on Frontiers in Handwriting Recognition, Heraklion, Greece, 1–4 September 2014; pp. 323–328. [[CrossRef](#)]
31. Yamaji, Y.; Shibata, T.; Tonouchi, Y. Online Handwritten Stroke Type Determination Using Descriptors Based on Spatially and Temporally Neighboring Strokes. In Proceedings of the 2014 14th International Conference on Frontiers in Handwriting Recognition, Heraklion, Greece, 1–4 September 2014; pp. 116–121.
32. Al-Hmouz, R.; Pedrycz, W.; Daqrouq, K.; Morfeq, A.; Al-Hmouz, A. Quantifying dynamic time warping distance using probabilistic model in verification of dynamic signatures. *Soft Comput.* **2019**, *23*, 407–418. [[CrossRef](#)]
33. Parziale, A.; Fuschetto, S.G.; Marcelli, A. Exploiting stability regions for online signature verification. In *New Trends in Image Analysis and Processing*; Lecture Notes in Computer Science; Springer: Basel, Switzerland, 2013; Volume 8158, pp. 112–121.
34. Slim, M.A.; Abdelkrim, A.; Benrejeb, M. Handwriting Velocity Modeling by Sigmoid Neural Networks with Bayesian Regularization. In Proceedings of the 2014 International Conference on Electrical Sciences and Technologies in Maghreb (CISTEM), Tunis, Tunisia, 3–6 November 2014; pp. 1–7. [[CrossRef](#)]
35. Faundez-Zanuy, M.; Sesa-Nogueras, E.; Roure-Alcobe, J.; Esposito, A.; Mekyska, J.; Lopez-de-Ipina, K. A Preliminary Study on Aging Examining Online Handwriting. In Proceedings of the 2014 5th IEEE Conference on Cognitive Infocommunications (CogInfoCom), Vietri sul Mare, Italy, 5–7 November 2014; pp. 221–224. [[CrossRef](#)]
36. Nguyen, C.T.; Zhu, B.L.; Nakagawa, M. A semi-incremental recognition method for on-line handwritten English text. In Proceedings of the 2014 14th International Conference on Frontiers in Handwriting Recognition, Heraklion, Greece, 1–4 September 2014; pp. 234–239. [[CrossRef](#)]
37. Aubin, V.; Mora, M.; Santos-Peña, M. Off-line writer verification based on simple graphemes. *Pattern Recognit.* **2018**, *79*, 414–426. [[CrossRef](#)]
38. Vásquez, J.L.; Ravelo-García, A.G.; Alonso, J.B.; Dutta, M.K.; Travieso, C.M. Writer identification approach by holistic graphometric features using off-line handwritten words. *Neural Comput. Appl.* **2018**, *1*–14. [[CrossRef](#)]
39. Kalbitz, M.; Scheidat, T.; Vielhauer, C. First investigation of feasibility of contact-less non-destructive optical sensors to detect, acquire and digitally process forensic handwriting based on pressure information. In Proceedings of the 2016 4th International Conference on Biometrics and Forensics (IWBF), Limassol, Cyprus, 3–4 March 2016; pp. 1–6. [[CrossRef](#)]
40. Okawa, M.; Yoshida, K. Off-line writer verification using shape and pen pressure information. In Proceedings of the 2012 International Conference on Frontiers in Handwriting Recognition, Bari, Italy, 18–20 September 2012; pp. 625–630. [[CrossRef](#)]
41. Bhateja, A.K.; Chaudhury, S.; Saxena, P.K. A Robust Online Signature based Cryptosystem. In Proceedings of the 2014 14th International Conference on Frontiers in Handwriting Recognition, Heraklion, Greece, 1–4 September 2014; pp. 79–84. [[CrossRef](#)]
42. Kutzner, T.; Dietze, M.; Bonninger, I.; Travieso, C.M.; Dutta, M.K.; Singh, A. Online Handwriting Verification with Safe Password and Increasing Number of Features. In Proceedings of the 2016 3rd International Conference on Signal Processing and Integrated Networks (SPIN), Noida, India, 11–12 February 2016; pp. 656–661. [[CrossRef](#)]
43. Aubin, V.I.; Doorn, J.H. *Exploring New Handwriting Parameters for Writer Identification*. Encyclopedia of Information Science and Technology, 4th ed.; IGI-Global: Hershey, Pennsylvania, 2019. [[CrossRef](#)]
44. Lee, P.X.; Ding, J.J.; Wang, T.C.; Lee, Y.-C. Automatic Writer Verification Algorithm for Chinese Characters Semi-Global Features and Adaptive Classifier. In Proceedings of the 2018 IEEE International Conference on Multimedia & Expo Workshops (ICMEW), San Diego, CA, USA, 23–27 July 2018; pp. 1–6. [[CrossRef](#)]
45. Kutzner, T.; Bönninger, I.; Travieso, C.M. Nutzer-Authentifizierung mittels handschriftlichen Passworten auf Mobil-Geräten. In 12. *Wissenschaftstage der Hochschule Lausitz (FH)*; University of Applied Sciences: Senftenberg, Germany, 2012.
46. Venugopal, V.; Sundaram, S. An improved online writer identification framework using codebook descriptors. *Pattern Recognit.* **2018**, *78*, 318–330. [[CrossRef](#)]
47. Kutzner, T.; Ye, F.; Bönninger, I.; Travieso, C.M.; Dutta, M.K. User Verification using Safe Handwritten Passwords on Smartphones. In Proceedings of the 2016 9th International Conference on Contemporary Computing (IC3), Noida, India, 11–13 August 2016.

48. Kutzner, T.; Travieso, C.M.; Bonninger, I.; Alonso, J.B.; Vasquez, J.L. Writer identification on mobile device based on handwritten. In Proceedings of the 2013 47th International Carnahan Conference on Security Technology (ICCST), Medellin, Colombia, 8–11 October 2013; pp. 1–5. [[CrossRef](#)]
49. Liwicki, M.; Bunke, H. IAM-OnDB—An on-line English sentence database acquired from handwritten text on a whiteboard. In Proceedings of the 8th International Conference on Document Analysis and Recognition (ICDAR’05), Seoul, South Korea, 31 August–1 September 2005; pp. 956–961. [[CrossRef](#)]
50. Hafemann, L.G.; Sabourin, R.; Oliveira, L.S. Offline handwritten signature verification—Literature review. In Proceedings of the 2017 Seventh International Conference on Image Processing Theory, Tools and Applications (IPTA), Montreal, QC, Canada, 28 November–1 December 2017; pp. 1–8. [[CrossRef](#)]
51. Sharma, A.; Sundaram, S. On the exploration of information from the DTW cost matrix for online signature verification. *IEEE Trans. Cybern.* **2017**, *48*, 611–624. [[CrossRef](#)] [[PubMed](#)]
52. Chapran, J. Biometric writer identification: Feature analysis and classification. *Int. J. Pattern Recognit. Artif. Intell.* **2006**, *20*, 483–503. [[CrossRef](#)]
53. Schlapbach, A.; Liwicki, M.; Bunke, H. A writer identification system for on-line whiteboard data. *Pattern Recognit.* **2008**, *41*, 2381–2397. [[CrossRef](#)]
54. Frank, E.; Hall, M.A.; Witten, I.H. *The WEKA Workbench. Online Appendix for “Data Mining: Practical Machine Learning Tools and Techniques”*, 4th ed.; Morgan Kaufmann: Auckland, New Zealand, 2016.
55. Venugopal, V.; Sundaram, S. Online Writer Identification With Sparse Coding-Based Descriptors. *IEEE Trans. Inf. Forensics Secur.* **2018**, *13*, 2538–2552. [[CrossRef](#)]



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).