Expert Systems With Applications 165 (2021) 113797

Contents lists available at ScienceDirect

Expert Systems With Applications

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

In-air signature verification system using Leap Motion

Elyoenai Guerra-Segura *, Aysse Ortega-Pérez, Carlos M. Travieso

Institute for Technological Development and Innovation in Communications (IDeTIC), University of Las Palmas de Gran Canaria, Las Palmas de Gran Canaria 35017, Spain

ARTICLE INFO

Keywords:
Leap Motion controller
In-air signatures
3D signature verification/recognition
3D signature processing
Machine learning

ABSTRACT

Signature verification is a widely explored field due to its high acceptance and its compromise between security and comfort. Recently, different techniques have appeared to improve the capture, processing, and classification of signatures. In this work, authors present a novel and robust in-air signature verification system, which applies the use of Leap Motion controller to characterize in-air strokes, due to its stability and good performance for this task, as it will be demonstrated. Therefore, a database has been built for developing the experiments, which is composed of 100 users, with 10 genuine and 10 forgery samples per user. The implemented system is tested against two tests of impostor samples, zero effort attacks and active impostors. The second type of attacks are developed by different users, who showed very good abilities with the sensor. The classification is done by a Least Square Support Vector Machine. The equal error rate was 0.25% and 1.20%, respectively. The proposed system achieves very good results in comparison with the state-of-the-art one, which suggests that in-air signature processing gives an opportunity to increase systems' security.

1. Introduction

Computer Science has notably evolved, allowing a high development of the information technologies, which results in a more electronically connected society. From this evolution, different systems are developed to perform repetitive tasks instead of the user or to facilitate remote interaction between users and administration systems. These activities show an evolution of authentication systems to ensure user's digital identity.

Biometrics, understood as the process to recognize people based on their characteristics (Anand et al., 2010), is one of the most covered fields when recognizing users. It is usually divided into two groups, according to the studied characteristics, as follows:

1. Physical features: fingerprint, iris, palm of the hand, face, etc.
2. Behavioral features: signature, walk, etc.

The ideal system does not exist, and different aspects must be studied when a new approach is designed. Physical features tend to provide high security levels due to the temporal variability of behavioral features, but they usually require more difficulties for the user to employ them. However, there are some examples of hybrid techniques, which try to combine the strength of both groups. For these reasons, it is important to know the application environment for developing the best system for the scenario.

User authentication based on signatures is a widely-studied field in biometrics since it is one of the most accepted verification techniques (Behera et al., 2017; Impedovo & Pirlo, 2008), mainly because of its usability. Signatures verification techniques are usually divided into online (Plamondon & Srihari, 2000) and offline (Vargas et al., 2009). Both approaches try to identify and characterize the features which define the signature of each user, trying to check if a signature is original or not, applying for these a wide spectrum of classifiers (Bibi et al., 2020).

Offline techniques study the scanned image of a signature, while online techniques measure the data captured while the signature is done, as velocity (Elliott & Hunt, 2006), pressure (Ammar et al., 1986), etc. In general, online techniques perform better results due to the information sources they manage, as well as the sequential information of each signature that is already known, plus all previous parameters.

As shown in Behera et al. (2017), Impedovo and Pirlo (2008), Lee et al. (1996), some signature authentication systems are not resistant to imposter attacks because they do not contemplate these kind of errors. For these reasons, authentication systems should also verify if a signature is genuine or not, to ensure their robustness and efficiency against any kind of attack. When focusing in verification signature performance,

* Corresponding author.
E-mail addresses: eguerra@idetic.eu (E. Guerra-Segura), aysse.ortega101@alu.ulpgc.es (A. Ortega-Pérez), carlos.travieso@ulpgc.es (C.M. Travieso).

https://doi.org/10.1016/j.eswa.2020.113797
Received 22 May 2019; Received in revised form 20 July 2020; Accepted 25 July 2020
Available online 11 August 2020
0957-4174/

online systems are more reliable than offline ones as it is easier to remember and reproduce the shape of a signature than its dynamic features (Guru & Prakash, 2009).

Offline and online systems usually use different kinds of features to analyze the signature (Alonso-Fernandez et al., 2009; Ferrer et al., 2005). For example, offline techniques analyze the geometry and the texture of the signature, using boundary and geometric features (Alonso-Fernandez et al., 2009; Ferrer et al., 2005).

On the other hand, online techniques study the dynamic features of signatures' strokes, which is the reason why these kinds of systems usually perform a comparison of temporal series. This temporal comparison can be performed using the Dynamic Time Warping (DTW) (Bailador et al., 2011).

Talking about online techniques and signatures capture, there are also several options. On the one hand, there is the option of capturing 2D data of the signature using pen-sensitive devices by computing the pressure, velocity, acceleration, etc. On the other hand, 3D representations of the signature have already come to reality, performing better results. The 3D representations can be developed by using different sensors, as accelerometers (Bailador et al., 2011), Microsoft Kinect (Qu, 2015), Leap Motion Sensor (Behera et al., 2017), etc.

1.1. Related work

Previous works, to the authors' knowledge, are shown in this section to present the state of the art of online signature verification and the use of several sensors to capture in-air online data for biometry purposes.

According to the purposes of the works, they could be grouped into two main areas, as follows:

- Biometrics applied to signature verification
- Biometrics applied to other fields

Signature verification systems mainly use online data. Some of these works analyze in-air information to characterize the signatures using different sensors. On the other hand, online techniques and in-air information is also applied to other fields, such as: medicine, music, videogames, etc. (Bibi et al., 2020).

Talking about feature extraction in online systems, it can be based on parameters or function methods (Okawa, 2020). While first ones use global information such as duration, second ones analyze local properties as trajectory and pressure. Generally, function-based systems provide better verification performance (Tang et al, 2017).

1.1.1. In-air information applied to signature verification

In-air strokes as a biometric technique has been proposed in a few works. Some examples of these are shown in (Bailador et al., 2011; Behera, Dash, Dogra, & Roy, 2018; Behera, Dogra, & Roy, 2018; Farella, O'Modhrain, Benini, & Riccó, 2006; Haskell et al., 2006; Liu et al., 2009; Okumura et al., 2006). These works propose both the signature and other gestures as methods of user identification.

The work presented in Bailador et al. (2011) is the first work, to the authors' knowledge, in proposing the employment of user's handwritten signature, developed in the air, as a method to verify identity. It proposes the use of an accelerometer to capture the data and DTW as one of the possible techniques to compare the temporal series. However, in this work a mobile phone is used as capture device.

In the case of Farella et al. (2006), a system based on in-air gestures that are captured by an accelerometer is presented, using KNN as classifier. Okumura et al. (2006) proposes a similar system to capture the data using a mobile phone. However, in this case, a DP-matching algorithm is applied to avoid possible fluctuations.

In Haskell et al. (2006), also an accelerometer is used. However, signature is proposed as the identification gesture. Curvature moments and Mahalanobis distance are used to classify the samples. Although signature is proposed as an identification gesture, authors detailed (Bailador et al., 2011) as the first work to do this since it is the first one, to author's knowledge, to propose an isolated final system to perform the identification.

To show the importance of the gesture, in Liu et al. (2009) authors perform a study using uWave, 'a state-of-the-art recognition system for user-created free-space manipulation, or gestures'. The results highlight the importance of gesture selection.

Recently, other works have been developed related to the in-air signature, as (Behera et al., 2017), which also proposes the use of the handwritten signature, with the difference of the sensor used to capture the data. Another example is shown in Behera, Dash et al. (2018), Behera, Dogra et al. (2018), where a system based on 3D in-air signature is presented as solution for verification and recognition.

In general, according to Bibi et al. (2020), online systems for signature verification usually manage contact devices to capture the data. On the other hand, this work proposes an approach applying a contactless device and the user of the user signature, against works that apply the use of general gestures.

1.1.2. In-air information applied to other fields

Other works focus on the use of in-air information to develop systems to solve other problems. For example, Silva et al. (2013) presents a study where the use of Leap Motion as a tool to generate digital music is evaluated. On the other hand, there are works related with medicine, as the ones shown in Haleem Buttet al. (2017), Lahanas et al. (2017), where Leap Motion is used for laparoscopy and Parkinson.

Gesture recognition is also applied to understand sign language. In Mohandes et al. (2014), the sensor is used to understand the Arabic sign language. Physical rehabilitation is another field of application for the sensor, as shown in Cohen et al. (2018), Postolache et al. (2019). In Bachmann et al. (2018), a full and current review is found. Some of the more recent studies are shown in Kumar et al. (2017), Placidi et al. (2017).

Considering the strength points of online techniques applied to in-air gestures, this work proposes the use of Leap Motion controller to characterize in-air signatures to perform a signature verification system. The selection of this sensor is based on its stability during time for this kind of data, as studied on Guerra-Segura et al. (2017). The proposal is detailed below, and the sensor is shown in Section 2.

1.2. Proposal

Motivated by recent studies and applications based on 3D models, authors developed this work to evaluate the performance of a commercial sensor to solve the problem of verification of users by their signatures. On the other hand, contactless systems are emerging based on hygiene and maintenance reasons, which increases the interest of studying this sensor. Once the temporal variability of signals for their use in this purpose has been studied in Guerra-Segura et al. (2017), following step is evaluating the performance of a verification system.

This paper presents an approach for solving the problem of detecting imitated signatures when processing the identification of a user. Particularly, the proposed solution uses in-air signatures, which are described by aerial strokes drawn with the index finger, captured by the Leap Motion sensor. Signature verification systems try to decide if a given signature has been done by the user who is accessing to the system, while signature recognition systems identify the most probable user for a given signature, without evaluating if the signature is genuine or is imitated.

Verification systems need to store users' models to compare the incoming signature. These models are created during an enrolling phase, when the user should perform its signature several times to compute the intra-user variability (see Fig. 1).

Fig. 1 shows a block diagram where system's architecture is displayed.

The work studies and deepens in the innovation of using the third

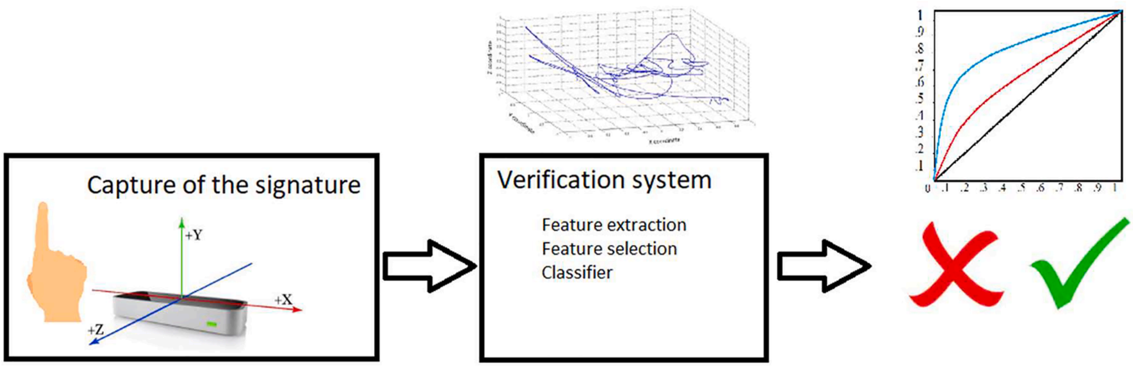


Fig. 1. System's architecture.

dimension to increase the performance of authentication systems when using a contactless device and the user signature, against other works that analyze general gestures or signature capture with contact devices. The final goal is to develop a novel and robust system to detect falsifications and to ensure the identity of the user by the performance of its in-air signature.

An in-air signature is represented in 3D in Fig. 2. As it can be observed, the 3D representation uses XYZ plane to show the coordinates of the air stroke to make easier its visualization.

The rest of the paper is organized as follows. Authors present the use of the sensor and materials in Section 2. Processing stages and algorithms are explained in Section 3, while Section 4 shows the experimental methodology developed in this work. Experiments' results, results discussion and a comparison of methods are presented in Section 5. Finally, conclusions are drawn in the last section.

2. Materials: building of the dataset

In this section, different aspects related with the database are presented. First, the chosen sensor and some of its characteristics are shown together with the protocol designed to capture the in-air signatures.

On the other hand, the signals that have been captured from the set offered by the sensor are detailed. Finally, the characteristics of the database are shown.

2.1. Leap Motion and data acquisition protocol

Leap Motion is a commercial tracking sensor which operates on the infrared spectrum and provides information about different measurements of the hand with a high accuracy. Moreover, its interaction range allows its use without moving it from the table, being this sensor a good option to develop solutions to interact with a personal computer.

There are already works that study the sensor for different purposes, mainly medicine, as explained in Bachmann et al. (2015), Behera, Dash et al. (2018), Behera, Dogra et al. (2018), Chahar et al. (2015), Haleem Buttet al. (2017), Kaji and Sugano (2017), Kamaishi and Uda (2016), Mohandes et al. (2014), Silva et al. (2013). These related works show how this sensor offers a good performance, independently of the field of study. It validates its use for this proposal.

Authors studied the temporal variability of this sensor when characterizing in-air signatures in Guerra-Segura et al. (2017). Results show its suitability for characterizing air strokes. Moreover, in Behera et al. (2017), this sensor is used for the same purpose. It has also been used for biometric authentication using handwriting, as explained in Kamaishi and Uda (2016).

Behera, Dash et al. (2018), Behera, Dogra et al. (2018) shows a performance of an online system, which uses Leap Motion for 3D air signature recognition and verification. This work focuses on improving the processing velocity by optimizing the extracted features and uses a database formed by 80 users and 20 signatures for each one. The results are higher than the ones in Behera et al. (2017), obtaining a 98% of accuracy. Although higher accuracies can be observed in the state of the art, the relationship between accuracy and velocity is very optimal.

To capture the data for this study, an algorithm developed and presented in Guerra-Segura et al. (2017) is used. The algorithm stores 21 characteristics for each recorded frame.

As explained in Guerra-Segura et al. (2017), the capture protocol consists of the following 4 steps:

- Steps 1 and 2. Hand detection and starting position. These steps are performed to detect the hand and to initialize the capture of the signature. Fig. 3 shows the starting position.

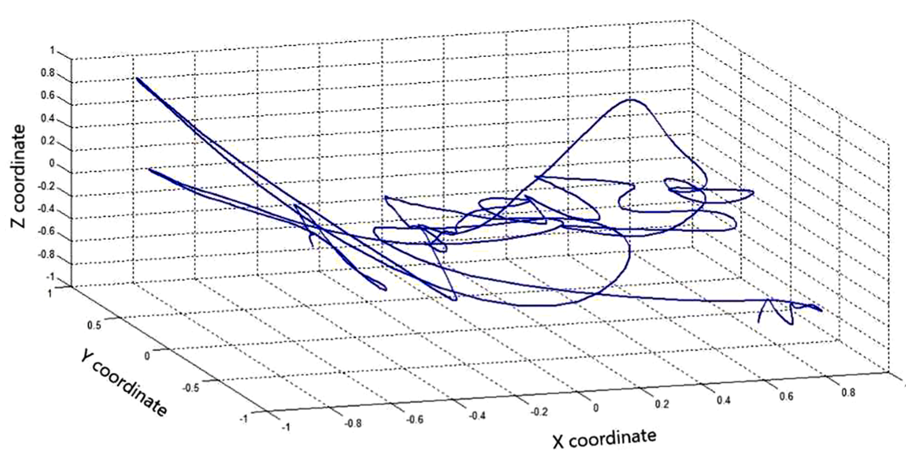


Fig. 2. 3D representation of an in-air signature.

- Steps 3 and 4. Signature performance and ending position. Once the user finishes the signature, he stretches the thumb to stop recording the data. Fig. 4 shows the signature performance.

In general, users needed an average time of 5 min to be able to perform their signature. This learning time is mainly due to the need to adapt to the 3-dimensional representation. This need for adaptation led to 4 users not being able to perform their enrollment, driving to a failure to enroll rate (FER) of 3.8% (100 successful users and 4 failures).

Based on the performance of the first sessions with users, the decision to avoid several sessions in different days for the same users was taken. This decision was made according to the difficulties shown by some users, and the fact that enrollment steps usually consist on 1 session. There were 2 sessions per user, but they were performed during the same day.

User's samples were obtained during the same day but not consecutively. Users performed a total of 20 signatures, divided in two groups of 10. Between groups, other users performed their respective signatures. With the total of 20 samples per user, an algorithm of Dynamic Time Warping was applied to select 10 samples for each user. The selection was based on the distance obtained with this algorithm. The 10 chosen samples are not the most similar, but the 10 that allow to cover the differences due to the inter-user variability.

2.2. Captured characteristics

As mentioned above, from all the possible characteristics offered by the Leap Motion, 21 are the chosen to characterize the aerial strokes. The selection of these 21 characteristics is based on cited references and the evaluation of their temporal variation for this purpose. These 21 measures are shown in Table 1.

These measures were selected according to the most used features and the way these could be emulate with the Leap Motion signals. For example, velocity and acceleration of the strokes are measured with signals 4–6 and 13–15, according to Table 1. On the other hand, pressure can be taken into consideration using the information about the position of the index finger and the palm, mainly in the *Z* axis, using signals 1–3 and 10–12. Finally, rest of signals are used to describe user behavior during the signature stroke – i.e. how to position and rotate the index finger and the hand, emulating the features related to angular moments and angles used in bibliography-.

As shown in Table 1, for each group of coordinates it is obtained the *Z* axis since it is supposed to offer a high level of personal information. Some works have studied the pressure of the manuscript signature, to the information related the *Z* axis, as done in Plamondon and Srihari (2000). With this sensor, information about the *Z* axis is as easy to obtain as information about the *X* and *Y* axes.

Main differences between the manuscript signature and the one obtained with this sensor are based on the on-line techniques. With the sensor it is possible to obtain information about the velocity, acceleration, angles, etc. while the signature is being developed. Manuscript signature offers less information although it has been more studied.

According to previous works about the use of different sensors, *Z* axis gives important information about the unconscious movement of the hand when signing. This is the main strength of these tools, although it is also a source of noise since users are not used to develop a 3-dimensional signature.

As it can be observed in Vamsikrishna et al. (2016), position and angles are some of the used characteristics when using Leap Motion to characterize the position. In addition to these features, it can be observed in Table 1 other features related to velocity, which add information about the acceleration of the signatures. These 21 features were validated by users in Guerra-Segura et al. (2017), where a signature was performed daily for 3 months to evaluate the time variation of the signals for this purpose.

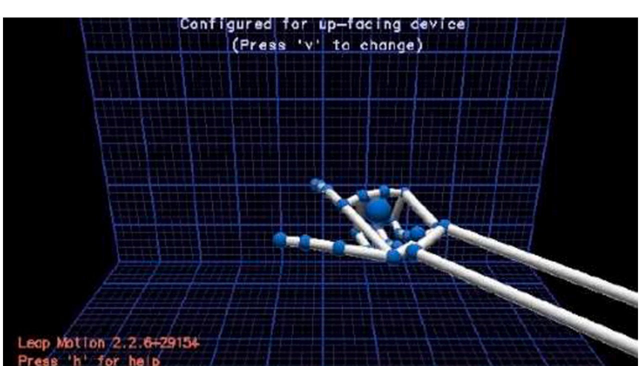


Fig. 3. Starting position.

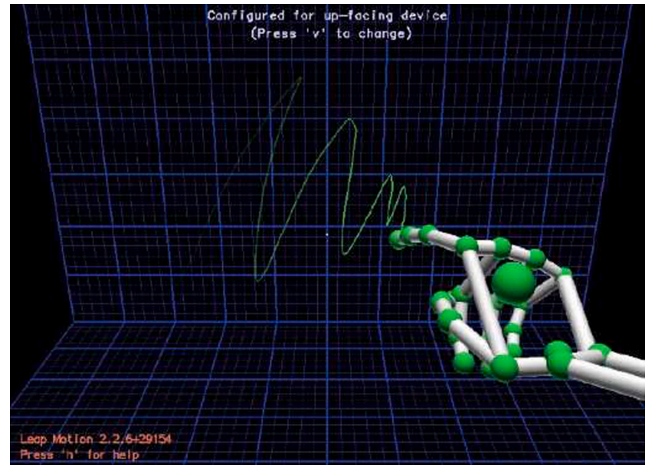


Fig. 4. Signature performance.

Table 1
List of the characteristics and the methods to obtain them.

Measure identifier	Method	Data
1	*finger.tipPosition()*	x coordinate
2		ycoordinate
3		zcoordinate
4	*finger.tipVelocity()*	xcoordinate velocity
5		ycoordinate velocity
6		zcoordinate velocity
7	*finger.direction()*	xcoordinate of the index direction vector
8		ycoordinate of the index direction vector
9		zcoordinate of the index direction vector
10	*hand.palmPosition()*	xcoordinate
11		ycoordinate
12		zcoordinate
13	*hand.palmVelocity()*	xcoordinate velocity
14		ycoordinate velocity
15		zcoordinate velocity
16	*Hand.palmNormal()*	xcoordinate of the hand normal vector
17		ycoordinate of the hand normal vector
18		zcoordinate of the hand normal vector
19	*hand.direction().pitch ()*	Pitch angle of the hand
20	*hand.direction().yaw ()*	Yaw angle of the hand
21	*hand.direction().roll()*	Roll angle of the hand

2.3. Database characteristics

A database has been built following the previous sections. Below are presented the main characteristics of this database.

Table 2 shows a resume of the main characteristics of the database.

To improve time preprocessing for the different experiments, the database is divided into different datasets according to the experiments' goals. Differences between subsets are based on the size and the presence of imitated samples. Their characteristics are presented with the corresponding experiment in section 4.

Related to the 10 forged signatures per user, all the forgeries have been developed by the 2 most expert users. Theses 2 users have been selected by observing their performance with the sensor and their ability to adapt their aerial strokes according to the signatures to be imitated. To perform the forged signatures, forgers only observed the 2-dimensional representation of the original signatures.

3. Methods

This section explains the used concepts in this proposal, which are important to understand the proposal and to address experiments.

First, the preprocessing steps applied to the captured signatures are shown. Then, the features extraction stage is explained, showing the different characteristics applied to the signatures. Finally, a brief explanation of the applied classifier is shown.

3.1. Preprocessing

A first step is the preprocessing stage. The mission is to normalize each signature in order to make it independent of position and size; since each signature can change its size, be acquired in different coordinates, because each user always does different each signature.

This preprocessing step consists in removing the influence of size and position. It is reach applying Eq. (1).

$$v^{'} = \frac{v - \min(v)}{\max(v) - \min(v)} * (\max(v^{'}) - \min(v^{'})) + \min(v^{'}) \tag{1}$$

where v is the vector to preprocess.

This equation transforms the vectors (temporal series for each axis) to the specified range, [−1 1] in this case. Since the features are online, this step consists on normalizing the features with respect to the original capture.

In Fig. 5, 5 samples from the same user are shown in the same plot to visualize the performance of the preprocessing step. These signatures are represented according to x and y coordinates of the *finger.tipPosition ()*. Therefore, it is the projection from 3D (xyz) to 2D (xy) of these signatures.

3.2. Features extraction

After normalizing the signatures, features extraction is applied. The extraction is mainly applied to each of the 21 signals that form each signature, according to Table 1. Moreover, each of these 21 signals is divided into different numbers of segments in order not to avoid details, since characterizing very long temporal series could result in avoiding some details, which can be highlighted by dividing the data into segments (Bailador et al., 2011; Behera et al., 2017; Guerra-Segura et al., 2017; Wu, Pan, Zhang, Qi, & Li, 2009). In the experiments, as shown in next sections, signals are divided into 2, 3, 4–20 or more segments.

Segments are characterized by statistics from first to fourth order, correlation, and entropy. These features define the value of each segment, reducing the effect of details but without being a generalization. Then, the grade of discrimination of each signature can be shown. Very short segments give many details and it gives noise for the analysis intra-class. The list of features is the following:

- Mean: it shows the absolute value generated by the distribution. It is a first order statistical.
- Standard deviation: it is a second order statistical and explains the stability of the mean.
- Correlation: correlation is applied to each pair of possible signals of each segment.
- Shannon entropy: amount of information contained in a random variable (Shannon & Weaver, 1949). It is calculated as shown in equation (2).

$$H(x) = -\sum P_i \log_2(x_i) \tag{2}$$

- Kurtosis: nonlinear measure to evaluate the normality of a distribution. It is a fourth order statistical and it is a measure of the distribution's shape. Its information refers to the similarity between the distribution's shape and a Gaussian distribution. The most common definition is that the kurtosis of a distribution H is the characteristic measured by its standardized fourth central moment (Balanda & MacGillivray, 1988).

To evaluate the kurtosis of different distributions, Eq. (3) is applied:

$$k = \frac{H(x-\mu)^4}{\sigma^4} \tag{3}$$

where μ is the mean of x, σ is the standard deviation of x, and $H(t)$ represents the expected value of the quantity t. Note that the kurtosis of a normal distribution is 3, which is often used as a standard.

- Skewness: it is a measure about the asymmetry of a distribution related to its mean. It can be positive, negative, or undefined. Its value refers to the relation between left and right tails of the distribution (Mardia, 1970).

Skewness is calculated according to Eq. (4).

$$\gamma_1 = E\left[\left(\frac{X-\mu}{\sigma}\right)^3\right] = \frac{\mu^3}{\sigma^3} = \frac{E[(X-\mu)^3]}{(E[(X-\mu)^2])^{\frac{3}{2}}} = \frac{k_3}{k_2^{3/2}} \tag{4}$$

where μ is the mean, σ is the standard deviation, E is the expectation operator, μ^3 is the third central moment, and k_t are the tth cumulants.

As an example, let us suppose that we want to compute the division of the signature into 3 segments. Moreover, for each of the 3 segments, we want to use only the signatures related to the 3D finger position, i.e. signals with identifiers 1, 2 and 3, according to Table 1. In this case, we are evaluating a signature characterized by 3 of the possible signals, and we want to divide the signature into 3 segments. We obtain 9 mean values, 9 standard deviation values, 9 kurtosis values, etc. since we are calculating 3 measures (one per signal), 3 kurtosis values, etc. for each segment. Correlation is a special measure because it is applied to each possible pair of signals in each segment. In this case, we also obtain 3 measures for each segment.

Table 2
Resume of the database's characteristics.

Characteristic		Value
Number of users		100
Samples for user		10 genuine samples + 10 forgeries
Gender	Masculine	65%
———	Feminine	35%
Age	Eldest	64 years old
	Youngest	13 years old
	Average	29 years old
	Typical deviation	±12 years old

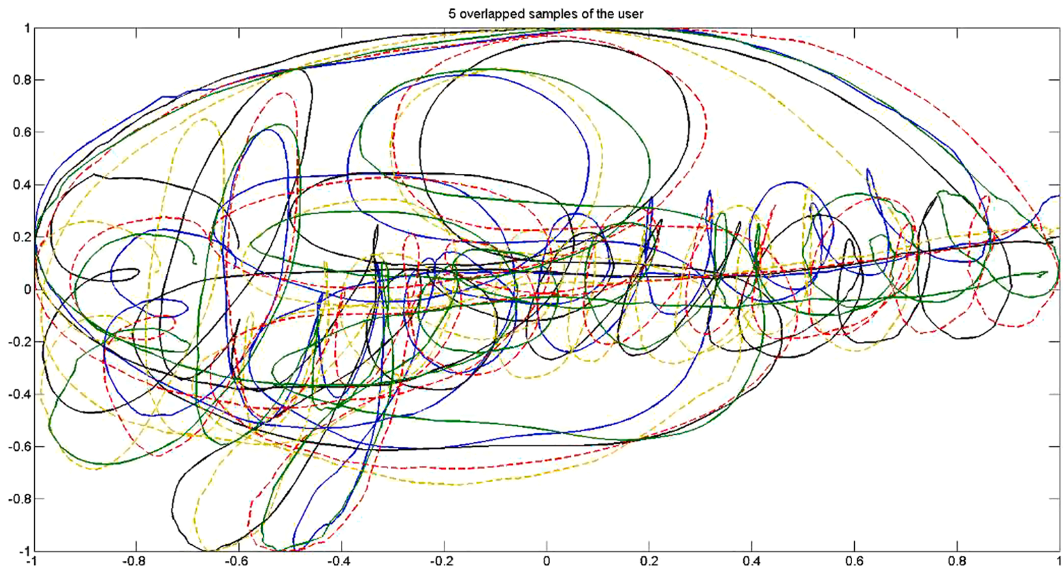


Fig. 5. Representation of 5 signatures from the same user overlapped in the same plot.

3.3. Classification

The studied classifier is the Least Squares Support Vector Machine (LS-SVM) (Suykens & Vandewalle, 1999). This classifier is chosen against SVM due to its suitability for large experiments when comparing both (Wang & Hu, 2005). The LS-SVM has demonstrated to be a good election for discriminative features and for a big size of data, and recent studies have compared the performance of generic SMV against other classifiers (Upadhyay et al., 2020; Parmar et al., 2020).

It is also used on transfer learning approaches, and recent studies have implemented theses kind of approaches using SVM (Wang et al., 2020; De Cooman et al., 2020; Bibi et al., 2020). Moreover, these kind of classifiers (SVM and LS-SVM) have been used successfully in many pattern recognition problems, as handwriting recognition (Adankon & Cheriet, 2009).

As it can be observed, there are many reasons to choose SVM or LS-SVM as the classifier for this kind of problems. Moreover, in this work, result of the implemented systems confirm the performed election.

This classifier works with a minimization model based on Lagrange functions and polynomials. It is a supervised learning algorithm and was developed as a solid and robust tool for regression and classification in complex domains (Cortes & Vapnik, 1995).

SVM are based on the concept of decision planes, which are defined by the decision limits (Burges, 1998). A decision plane divides a set of objects formed of samples from different classes. A linear SVM classifier divides the set using a straight line. However, when it is not possible to perform a lineal separation, data must be mapped to a higher order dimension space where it is possible. This mapping is developed by a function called kernel, and the new separation lines are known as hyperplanes.

LS-SVM look for a hyperplane maximizing the separation between the hyperplane and the samples of both classes using lineal equations, while traditional SVM use the structural risk minimization principle. Moreover, while in SVM very support values are 0, in LS-SVM the support values are proportional to the errors.

About the kernel function, in this work two kernels are studied, the polynomial and the Radial Based Function (RBF). These kernel functions, which are selected according to author's experience in this kind of problems, are shown below.

3.3.1. Polynomial kernel

This kernel uses a no homogenous polynomic function of grade d, as shown in equation (5).

$$K(x,y) = \left(\sum_{i=1}^{n} x_i y_i + c\right)^d \tag{5}$$

where x and y are vectors, d is the grade of the function and c is an adjustment parameter to modify the influence of higher-order and lower-order terms of the polynomial.

3.3.2. RBF kernel

It uses an equation which values depend on the distance to a point defined as central reference, as detailed in Eq. (6).

$$K(x,y) = exp\left(-\frac{||x-y||^2}{2\sigma}\right) \tag{6}$$

where x and y are the input vectors and σ is a free parameter of adjustment.

4. Experimental methodology

To evaluate the performance of the proposed solution, values of False Rejection Rate (FRR), False Acceptance Rate (FAR), and Equal Error Rate (EER) are computed, therefore, a verification stage is applied. FRR refers to errors of rejecting a genuine user, while FAR are related with errors of accepting an impostor. The decision threshold, where the best trade-off between FAR and FRR is obtained, is the EER. To compute FRR, genuine samples for the users must be divided into training and testing samples. On the other hand, to obtain the FAR, genuine samples are used to train, and both the imitated samples and the signatures of the rest of users are used to test, depending on what kind of attack is being simulated.

The methodology is based on a typical classification architecture, where the database is separated into training and testing sets. Moreover, experiments use different subsets of the full database to optimize the processing time according to each objective, as explained below. The experiments are based on the use of a LS-SVM classifier and 2 options of kernels.

The validation of the classification experiments is driven by using a hold-out cross validation, because of the reduced number of samples. Different approaches were applied, from 50% hold-out cross validation (5 samples to train and 5 to test) to 10% hold-out cross validation (1 sample to train and 9 to test).

A total of 5 experiments are set up to compute the performance of the classifier in the studied scenario. The design of these experiments tries to develop a final system, adjusting the classifier, improving the feature selection stage, etc. Each experiment is based on the results of the previous one, and its results are tested to evaluate its performance and the need to adjust the settings obtained with the previous ones.

The experiments are briefly explained below. The used dataset for each experiment is shown.

1. Classifier adjustment: this first experiment focuses on obtaining the best values to adjust the different parameters of the SVM and the kernels (gam and sig2) to maximize the success using information related to the different planes (XY, XZ, YZ and XYZ). For the classification, features as mean, standard deviation and correlation of different signals divided in several segments are used. In this first experiment, a total of 812 complete simulations are driven, reason why a subset of the database is used because of the processing time of simulations.

Database subset: 39 users and only genuine samples.

2. Statistical features: the second experiment has the objective of evaluating the performance of additional statistical features when characterizing the in-air signatures. The additional features are Shannon entropy, skewness, and kurtosis. This experiment uses the dataset that give best results in experiment 1, when using 5 and 1 training samples. Results of adding all possible combinations of the new features are evaluated to see if these features improve the performance. In this case, a total of 126 complete simulations are performed.

Database subset: 39 users and only genuine samples.

3. Influence of the third dimension: after experiment 2, datasets with best results keep divided according to the planes involved in the feature extraction. As it can be observed, datasets could be formed by XY information or by XYZ information. At this point, it is possible to develop comparisons between dimensionality and number of training samples.

Database subset: 39 users and only genuine samples.

4. Features fusion: once results for different groups of features are obtained, this experiment tries to find a combination of these groups to improve the system performance. This experiment looks for evaluating the performance of applying features fusion to the dataset with best results. Obtained results are computed to conclude if fusion of feature improves the performance.

Database subset: 39 users and only genuine samples.

5. System performance: as previous experiments are designed to use a smaller database because of time processing, this experiment evaluates the results when increasing the size of the database, applying the configuration obtained in previous experiments. 100 complete simulations are driven to compute the mean EER for 1 and 5 training samples.

Database subset: 100 users and only genuine samples.

Final system performance: once the system has been validated with the previous subsets, final simulations are driven to compute the performance when falsifications are added to the database. For this final experiment, a total of 100 complete simulations for each number of training samples are performed.

Database subset: 100 users with genuine and imitated samples.

As it can be extracted from explanations, experiments 1–5 focus on designing and adjusting the system, while final simulations are developed to test the final performance. These finals simulations use full database and evaluate the system against real falsifications, taking into consideration both types of attacks, zero effort and active impostors.

As explained above, the experiments are set to evaluate each of the followed steps in the design and adjustment of the classification system. With each experiment, the influence of the different elements of the system, as shown in Fig. 1, can be evaluated. Following all the experiments drive authors to evaluate the final system being sure about the performance of each single element of the complete system.

5. Results and discussion

In this section, the results obtained for the 5 experiments and for the final system simulations are shown. Results of experiments 1–5 are shown briefly, since the main results are the ones obtained with the last tests.

As explained before, each signal is divided into segments, from 2 to 30 for these experiments. According to this, the nomenclature used for these experiments follows the next example: *using only the signals which refer to X and Y position of the index finger, each of them is divided into 5 segments. This dataset is called '5 segments subgroup of X and Y index finger position'.*

5.1. Experiment 1: adjustment of classifier

As it can be observed, there are many combinations for 21 signals and their division into segments. Therefore, results would include only the best performances. For this experiment signals are grouped as shown below:

a) Signals related to the XY plane.
b) Signals related to the XZ plane.
c) Signals related to the YZ plane.
d) Signals related to the XYZ plane.

In these experiments, first simulations are done with 5 training samples. These initial simulations are developed to detect the best groups for fast simulations and to adjust the classifier to obtain the highest possible EER values.

In total, 812 simulations are driven: 4 groups, 29 possible segments (dividing the signals from 2 to 30) and 7 subgroups per each group (according to Table 1).

Best results are obtained for the groups which are formed by all the possible signals. It means, for all signals related to group a), all signals related to group b), etc. Best results are shown in Table 3, according to Equal Error Rate (EER). Results are obtained using a standard polynomial kernel.

As it can be observed, the best result is obtained for dataset '5 segments subgroup of all signals related to XY plane', with an EER value of 0.5769%. This dataset is used to adjust the classifier to improve the results.

Once the best dataset is selected for a standard classifier configuration, the next simulations of this experiments try to adjust the classifier to improve the obtained EER.

5.1.1. Polynomial kernel

After driving different simulations, the optimal adjustment for the classifier, using a polynomial kernel, is achieved with the values shown in Table 4. The adjustment is driven according to the LS SVM definition shown in Brabanter et al. (2011), where for this type of kernel, gam is the regularization parameter and sig2 is the squared bandwidth.

With these parameters, the EER obtained for the selected dataset is 0.0405%, instead of the previous value of 0.5769%.

Then, simulations for other datasets with this parameter configuration are driven to check that the improvement is achieved for all the

Table 3
Best results obtained for experiment 1.

Dataset	Number of segments	EER (%)
All signals related to XY	5	0.5769.
All signals related to XY	4	0.6312.
All signals related to XZ	5	0.9430.
All signals related to XZ	4	1.0260.
All signals related to XYZ	2	1.0507.
All signals related to YZ	3	1.5380.
All signals related to YZ	4	1.5380.

Table 4
Polynomial kernel optimal configuration.

Parameter	Value
gam	300
sig2	[7;2]

cases and not only for the selected dataset.

Simulations are driven for training samples from 1 to 5 and dividing the signals into different amounts of segments. For 5 training samples, best results are obtained with datasets related to XY plane, while using 1 training sample gives better results for XYZ plane. On the other hand, the worst results are obtained for datasets related to YZ plane.

After these experiments and analyzing the results, authors conclude that XZ and YZ planes should be discarded for the following experiments, due to these planes omit information, which is relevant for the discrimination, according to the results.

5.1.2. RBF kernel

In this case, best results are achieved with the values shown in Table 5. According to Brabanter et al. (2011), for this kernel, gam is also the regularization parameter, determining the trade-off between the training error minimization and smoothness, while sig2 is also related to the squared bandwidth.

These values allow to get an ERR of 0% for the dataset selected to perform this first experiment.

According to results from the previous adjustment, next simulations are driven with dataset related to XY and XYZ planes to check that the adjustment is not only optimal for the studied dataset.

For 5 training samples, 0% of EER is obtained for datasets of all signals related to XY plane divided into 4, 5, 6, 7 and 8 segments. Although 0% of EER is obtained, it is not possible to select this system as a final system because it has been tested only for a very specific scenario and with a low level of samples, which is the reason why these results are only useful to develop more experiments according to it. On the other hand, for the dataset of all signals related to XYZ plane divided into 5 segments, the obtained EER is 0.1350%.

When evaluating the classifier with 1 training sample, the dataset of all signals related to XY plane divided into 6 segments performs an EER of 1.7094%, while the datasets of all signals related to XYZ plane divided into 4 and 5 segments show an EER of 1.1396%.

The training with 50% of samples offers enough information for a good verification, but when the number of training samples is decreased, the extra information from the third coordinate gives a more discriminative value. Therefore, authors conclude after this first experiment that when only 1 training sample is applied, 3 dimensions (XYZ plane) are more efficient due to the contribution of the third dimension, i.e. the *Z* coordinate.

5.2. Experiment 2: statistical features

As explained in section 4, this second experiment focuses on applying different statistical features to evaluate their performance. The evaluated features are usually applied in different works (Bachmann et al., 2015; Bailador et al., 2011; Bernardos, Sánchez, Portillo, Besada, & Casar, 2015; Guerra-Segura et al., 2017; Wu et al., 2009).

This experiment is developed using the 7 datasets which offer the best results in the first experiment. Since this selection gives only datasets tested with 5 training samples, the 2 best results obtained for 1 training sample are also selected, given a total of 9 datasets.

The selected datasets are listed above:

- All signals related to XY plane divided into 8 segments (Dataset1)
- All signals related to XYZ plane divided into 12 segments (Dataset2)
- XYZ coordinates of the index position divided into 7 segments (Dataset3)
- XY coordinates of the index position divided into 11 segments (Dataset4)
- XYZ coordinates of the palm of the hand position divided into 22 segments (Dataset5)
- XYZ coordinates of the index velocity divided into 6 segments (Dataset6)
- XY coordinates of the palm of the hand position divided into 22 segments (Dataset7)
- All signals related to XYZ plane divided into 7 segments (Dataset8)
- All signals related to XY plane divided into 3 segments (Dataset9)

For this second experiment, simulations are driven for 5 and 1 training samples for each of the listed datasets. For nine datasets, new statistical features are added. These new features are Shannon entropy, skewness, and kurtosis. For each simulation, results before and after the addition of these features are compared to evaluate their performance.

In Figs. 6 and 7, the comparison developed for 5 and 1 training samples can be observed. The *X* axis refers to the possible combination when adding the new features. The values are the rate between the previous EER and the one obtained with the addition of the new features. The threshold refers to the previous EER, so when a value is over it, the addition has produced a worse performance, while the simulation has given a better EER when the value is under the threshold.

For the 18 studied cases, 9 for 5 training samples and 9 for 1 training sample, kurtosis gives a better performance in 1 case, while skewness improves 3 cases and Shannon entropy offers better results in 7.

In general, the results are not improved with the addition of these features, which are proposed in other works for similar purposes, as shown before. Authors conclude that these features are not relevant for the experiments related to this scenario.

5.3. Experiment 3: influence of the third dimension

The third experiment focuses on evaluating the influence of computing the third dimension, i.e. the *Z* axis. As it can be observed in previous experiments, the third dimension is important when training with 1 sample, while training with 5 samples does not need this dimension to be able to discriminate between users. In this experiment, different numbers of training samples are tested to evaluate when the third dimension starts to be important for the classification.

According to experiment 1, YZ and XZ planes are discarded because of the low information shown, while XY and XYZ show the best results. For these reasons, in this case, the datasets used for comparison are:

- All signals related to XY plane
- All signals related to XYZ plane

The following figures show the comparison of EER related to the numbers of segments the datasets are divided into for 5 and 1 training sample. Results shown in Fig. 8 are achieved with the polynomial kernel, while Fig. 9 shows the results of applying the RBF kernel.

As it can be concluded from previous figures, for the RBF kernel, when training with 5 samples, adding the third dimension provides worse results. On the other hand, training with 1 sample offers better results when applying the third dimension.

For the polynomial case, from 3 training samples the results start to be better for the datasets which are formed with the third dimension. For 1 training sample, all the divisions of the datasets show better results for the one with the signals related to the three dimensions.

Table 5
RBF kernel optimal configuration.

Parameter	Value
gam	400
sig2	150

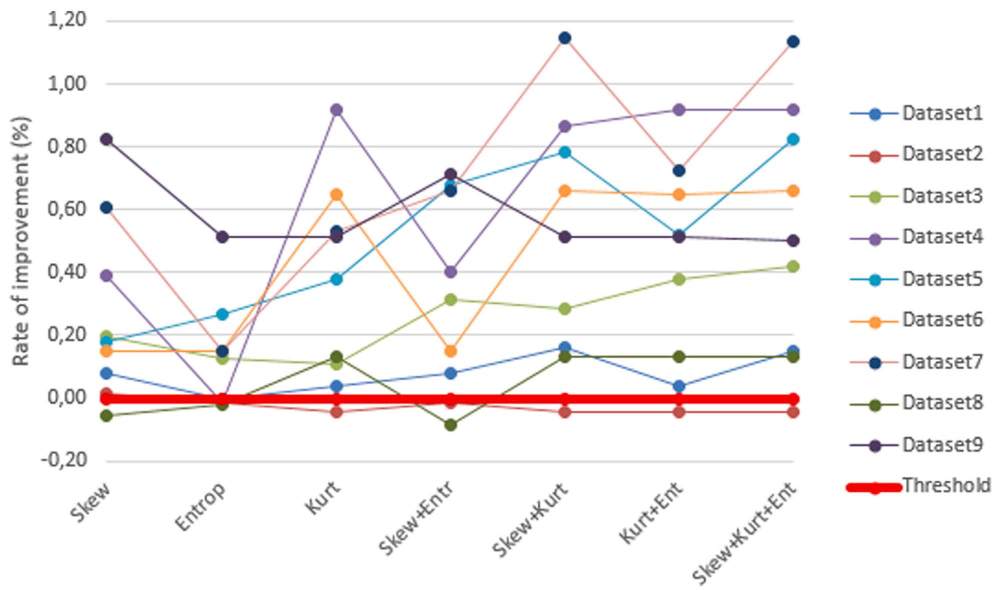


Fig. 6. Comparison developed for 5 training samples.

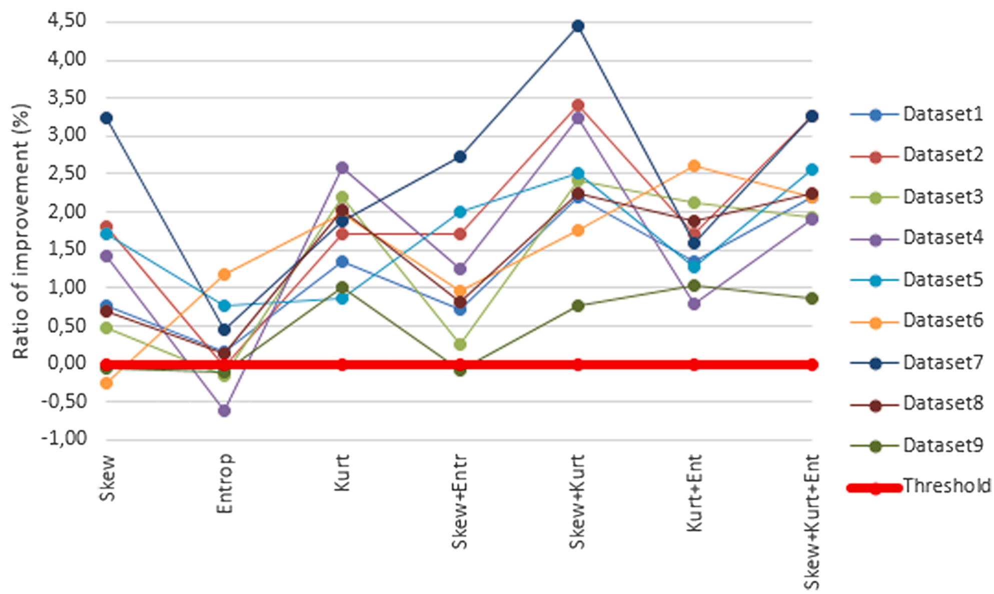


Fig. 7. Comparison developed for 1 training sample.

In the first experiment, it is possible to perceive the positive influence of the third dimension when training with a low number of samples. As explained above, the simulations driven in this third experiment allow authors to confirm the previous conclusion, since these simulations are designed to prove the influence of the third dimension.

From this experiment, authors conclude that the third dimension is important for low numbers of training samples, as it could be also concluded in first experiment.

5.4. Experiment 4: features fusion

Focusing on designing a system with best rate results/cost, i.e. the one with better EER for the minimum number of training samples, this experiment is driven with the dataset of all 3D signals. The experiment is tested for both kernels, the polynomial, and the RBF.

The objective of this experiment is to obtain a fusion of datasets which improves the EER. The tested fusions are done for the different divisions of the dataset formed with all the signals.

Although an EER of 0% is obtained in experiment 1, the used configuration it is not valid at all, since the simulations are driven with a low number of users. This experiment also tries to find different combinations of features which could be used when increasing the number of users in the system.

After having driven several experiments, the best result is achieved when merging the datasets ‘all signals divided into 5 segments’ and ‘all signals divided into 6 segments. For this combination, the EER is decreased from 1,1674% to 1,1096% for the polynomial kernel.

For the RBF kernel, the merge of ‘all signals divided into 3 segments’ and ‘all signals divided into 4 segments’ gives an improvement which reduces the EER from 1,1396% to 0,7890%.

From this experiment, authors conclude that dataset fusions offer an interesting improvement for the designed classifier, as shown before, considering the future increase of users. From here, these are the main datasets used for next experiments.

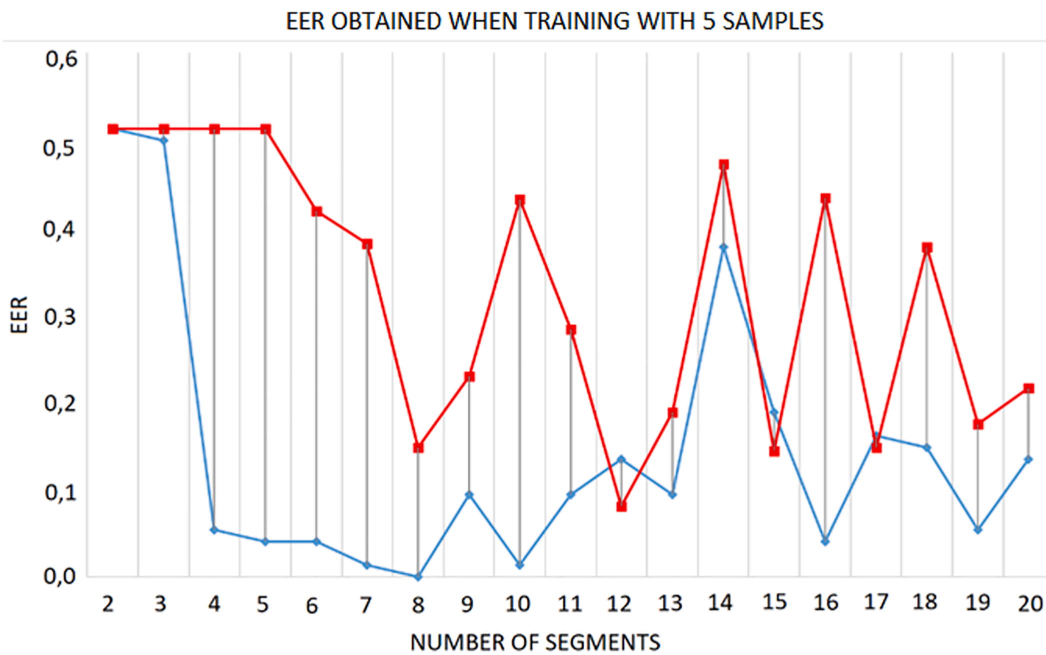


Fig. 8. Comparison developed for polynomial kernel when training with 5 and 1 samples, respectively. Red lines refer to the dataset with all XYZ signals and blues line refer to the dataset with all XY signals. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

5.5. *Experiment 5: system performance*

In this experiment, different simulations are driven with the datasets selected with the previous one. As mentioned above, it is used the total database formed by 100 users.

The simulations are performed with variations from 1 to 5 training samples. For each dataset combination, number of training samples, and kernel, 100 simulations are driven to compute the mean EER. Each of these simulations select the training samples randomly.

In Table 6, the achieved results are shown.

In this case, to evaluate the fusion performance, the same simulation is driven for the configurations which offered best results for individual datasets with 5 and 1 training samples, using both the polynomial and the RBF kernels.

In Table 7, comparison of the performances for both kernels are shown.

These comparisons show that fusion also offers better results for the full database, for both kernels. Moreover, it can be observed that for 100 users, the third dimension adds a lot of information to the 2D, giving better performances.

These results allow to conclude that feature selection and fusion, developed for previous experiments, offer a good performance for a higher database.

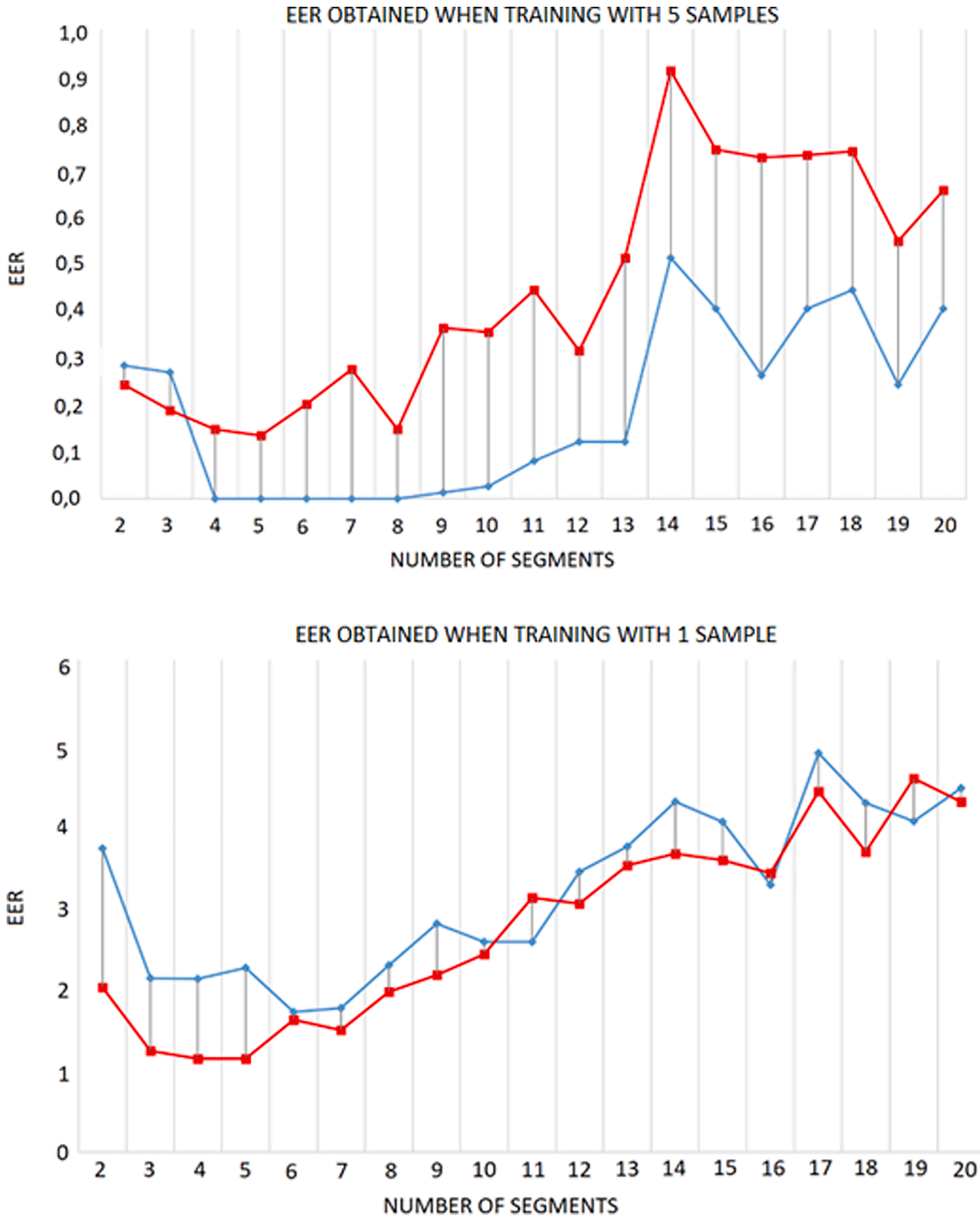


Fig. 9. Comparison developed for RBF kernel when training with 5 and 1 samples, respectively. Red lines refer to the dataset with all XYZ signals and blues line refer to the dataset with all XY signals. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Table 6
Results obtained with the database of 100 users.

Training Samples	Kernel	EER (%)	Standard deviation
5	RBF	**0,0751**	0,0897
	Polynomial	0,1207	0,0991
4	RBF	0,1318	0,1037
	Polynomial	0,1716	0,1052
3	RBF	**0,2218**	0,0996
	Polynomial	0,2897	0,1151
2	RBF	0,4209	0,1274
	Polynomial	0,5187	0,1660
1	RBF	**1,1017**	0,2290
	Polynomial	1,4185	0,3280

5.6. Final system performance

These final simulations evaluate the system for the full database with falsifications.

Fig. 10 shows the ROC curve obtained for the final system when training with 5 samples and performing 100 simulations.

FAR1 refers to False Acceptance Rate when applying random falsification, i.e. possible errors because of similarities of different users' signatures, resulting an EER of 0.25%. FAR2 is the rate when computing only spoofing attacks developed by the expert users, obtaining an EER of 1.5%. FAR and FRR build the ROC curve for the system when evaluating both kind of attacks. As it can be observed, the global system EER is 1.2%.

Table 7
Results for the RBF and polynomial kernel.

	RBF kernel		Polynomial kernel	
Training Samples	Dataset	EER (%)	Dataset	EER (%)
5	XYZ_4 segments	0.4729.	XYZ_5 segments	0.4920.
	XYZ_5 segments	0.4746.	XYZ_7 segments	0.6944.
	Fusion	0.0751.	Fusion	0.1207.
4	XYZ_4 segments	0.5957.	XYZ_5 segments	0.6371.
	XYZ_5 segments	0.5987.	XYZ_7 segments	0.8072.
	Fusion	0.1318.	Fusion	0.1716.
3	XYZ_4 segments	0.7799.	XYZ_5 segments	0.8429.
	XYZ_5 segments	0.8323.	XYZ_7 segments	1.0768.
	Fusion	0.2218.	Fusion	0.2897.
2	XYZ_4 segments	1.1836.	XYZ_5 segments	1.3355.
	XYZ_5 segments	1.3103.	XYZ_7 segments	1.5764.
	Fusion	0.4209.	Fusion	0.5187.
1	XYZ_4 segments	2.6406.	XYZ_5 segments	3.0782.
	XYZ_5 segments	2.3869.	XYZ_7 segments	3.4179.
	Fusion	1.1017.	Fusion	1.4185.

5.7. Comparison of methods

This section shows a comparison of different works referred to signature verification to evaluate the performance of in-air signature analyzed with online techniques.

Table 8 shows a comparison between references of the state-of-the-art and the proposed approach. It shows different reported EER and the composition of the used databases. There are not public datasets for these applications, and therefore, an exhaustive comparison is not possible. Table 8 can help to understand the behavior of each reference according to the method, the size of dataset and its EER. Therefore, the proposed method shows a very good answer for the applied methodology.

As it can be observed in Table 8, considering the EER, this approach offers the best result. Referring to the size of the database, Guru and Prakash (2009) performs one of the best relation between it and the obtained EER. However, as it is highlighted, the reported EER in Guru and Prakash (2009) is obtained by applying an individual threshold for each user.

On the other hand, Okawa (2020) present a low EER system. However, result related to database SVC2004 Task2 are achieved using 10 training sample. Results obtained MCYT-100 are the most like this work, since the database are formed by the same number of user and both systems use the same number of training samples. The main difference between these works lies on the capture device. While Okawa (2020) use a contact device, this study works with a contactless device.

Therefore, it could be concluded that the presented approach is a good and robust option to avoid falsifications with a common threshold, what makes easier to register new users since the computation of its threshold is not necessary.

In general, it can be concluded that online techniques and third

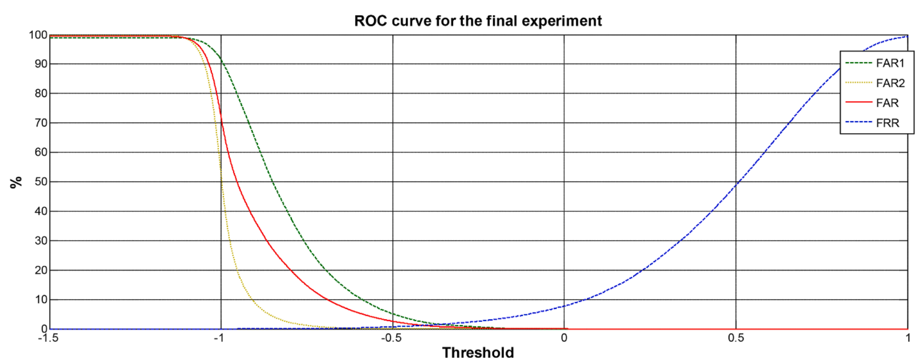


Fig. 10. ROC curve obtained for the final system.

Table 8
Comparison vs. the state-of-the-art.

Work	Database properties	Classifier	EER (%)
Behera et al. (2017)	a) 80 users * 20 samples = 1600 original samplesb) 10 impostors * 10 samples of 4 registered user signatures = 400 forged signatures	HMM DTW + K-NN	10 4.5
Guru and Prakash (2009)	a) 100 users * 25 samples = 2500 genuine samplesb) 100 users * 99 samples = 9900 imposters	Test signature	1.67(1)
Guru and Prakash (2009)	a) 330 users * 25 samples = 8250 genuine samplesb) 330 users * 229 samples = 75570 imposters	Test signature	1.65(1)
Bailador et al. (2011)	a) 96 users * 8 samples = 768 genuine signatures	Bayes	1.81 ± 0.33
Bailador et al. (2011)	a) 96 users * 8 samples = 768 genuine signaturesb) 7 impostors * 96 users * 7 samples = 4074 forged signatures	DTW	4.58 ± 0.51
Okumura et al. (2006)	a) 22 users * 5 samples = 110 genuine gestures	DP-matching	5.0
Behera, Dash et al. (2018), Behera, Dogra et al. (2018)	a) 80 users * 20 samples = 1600 original samplesb) 400 forged signatures	DTW + K-NN	8
Okawa (2020)	a) SVC2004 Task1b) SVC2004 Task2 40 users * (20 genuine + 20 skilled forgeries) = 1600 samples c) MCYT-100: 100 users * (25 genuine + 25 skilled forgeries) = 5000 samples	weighted multiple DTW	4.26(2)1.80(2) 1.28(3)
Singh and Viriri (2020)	SigComp2009: 12 genuine signers * 5 samples + 31 skilled forgeries * 5 samples * signer = 1920 samples	CNN + RNN	5.00
This proposal	a) 100 users * 10 samples = 1000 original samplesb) 2 expert users * 100 users * 5 samples = 1000 forged signatures	LS-SVM	1.20

(1) With writer dependent threshold; (2) Using 10 training samples; (3) Using 5 training samples

dimension information increases the performance of verification systems when falsification are considered, as shown in the different experiments.

6. Conclusion

A novel and robust approach has been developed for in-air signature verification, using online techniques and 3-dimensional information with a contactless device, obtaining an EER of 0,0751% and 1,1017% for 5 and 1 training samples, respectively, and an EER of 1,2% for 5 training samples when performing spoofing attacks.

The main conclusion is the importance of the third dimension, confirming results of previous works. In addition, with the results of this work, it is remarked the importance of the third dimension specially when using a low number of training samples. Moreover, the fact that the fusion of datasets formed by traditional statistical features show better results, is a secondary conclusion extracted in this work.

As shown in section 5, the third dimension has more importance when decreasing the number of training samples and increasing the number of users, since it offers behavioral information for user identification. In Figs. 8 and 9, best results are obtained by using the third dimension when decreasing the number of training samples.

Although conclusions could be compared with previous works, results and evaluation conditions give more importance to the current paper. Database is formed by 100 and skilled forgeries, while most of the rest use smaller database and only analyze random forgeries. Other difference is the number of training samples and the device used for capturing the data. For author's knowledge, there is no work with similar results using a contactless device and computing skilled forgeries.

Authors plan to evaluate offline techniques for data captured with the same sensor, to evaluate and combine the actual performance of 3-dimensional information with offline techniques. Also, different classifiers and features should also be studied in order to increase the global performance of the system.

CRediT authorship contribution statement

Elyoenai Guerra-Segura: Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Resources, Software, Validation, Writing - original draft, Writing - review & editing. **Aysse Ortega-Pérez:** Data curation, Formal analysis, Investigation, Software, Validation, Writing - original draft. **Carlos M. Travieso:** Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Software, Supervision, Validation, Visualization, Writing - original draft, Writing - review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Alonso-Fernandez, F., Fierrez, J., Martinez-Diaz, M. & Ortega-Garcia, J. (2009). Fusion of static image and dynamic information for signature verification. In 16th IEEE international conference on image processing (ICIP) (pp. 2725–2728).

Ammar, M., Yoshida, Y. & Fukumura, T. (1986). A new effective approach for automatic off-line verification of signatures by using pressure features. In Proceedings 8th international conference on pattern recognition (pp. 566–569).

Anand, P. M. R., Bajpai, G., & Bhaskar, V. (2010). 3D signature for efficient authentication in multimodal biometric security systems. *IACSIT International Journal of Engineering and Technology, 2*(2), 177–184.

Bachmann, D., Weichert, F., & Rinkenauer, G. (2015). Evaluation of the leap motion controller as a new contact-free pointing device. *Sensors, 15*, 214–233. https://doi.org/10.3390/s150100214.

Bachmann, D., Weichert, F., & Rinkenauer, G. (2018). Review of three-dimensional human-computer interaction with focus on the leap motion controller. *Sensors, 18*, 2194. https://doi.org/10.3390/s18072194.

Bailador, G., Sanchez-Avila, C., Guerra-Casanova, J., & De Santos Sierra, A. (2011). Analysis of pattern recognition techniques for in-air signature biometrics. *Pattern Recognition, 44*, 2468–2478. https://doi.org/10.1016/j.patcog.2011.04.010.

Balanda, K. P., & MacGillivray, H. L. (1988). Kurtosis: A critical review. *The American Statistician, 42*(2), 111–119. https://doi.org/10.1080/00031305.1988.10475539.

Behera, S. K., Dash, A. K., Dogra, D. P. & Roy, P. P. (2018). Air signature recognition using deep convolutional neural network-based sequential model. In 24th international conference on pattern recognition (ICPR) (pp. 3525–3530).

Behera, S. K., Dogra, D. P., & Roy, P. P. (2017). Analysis of 3D signatures recorded using leap motion sensor. *Multimed Tools and Applications*. https://doi.org/10.1007/s11042-017-5011-4.

Behera, S. K., Dogra, D. P., & Roy, P. P. (2018). Fast recognition and verification of 3D air signatures using convex hulls. *Expert Systems with Applications, 100*. https://doi.org/10.1016/j.eswa.2018.01.042.

Bernardos, A. M., Sánchez, J. M., Portillo, J. I., Besada, J. A., & Casar, J. R. (2015). A contactless identification system based on hand shape features. *Procedia Computer Science, 52*, 161–168. https://doi.org/10.1016/j.procs.2015.05.051.

Bibi, K., Naz, S., & Rehman, A. (2020). Biometric signature authentication using machine learning techniques: Current trends, challenges and opportunities. *Multimedia Tools and Applications, 79*(1), 289–340.

Brabanter, K. De, Karsmakers, P., Ojeda, F., Alzate, C., Brabanter, J. De, Pelckmans, K., … Suykens, J. A. K. (2011). LS-SVMlab Toolbox User's Guide.

Burges, C. J. C. (1998). A tutorial on support vector machines for pattern recognition. In *Data mining and knowledge discovery* (pp. 121–167). Amsterdam, The Netherlands: Kluwer Academic Publishers.

Chahar, A., Yadav, S., Nigam, I., Singh, R., & Vatsa, M. (2015). A Leap Password based verification system. In 2015 IEEE 7th international conference on biometrics theory, applications and systems, BTAS. http://doi.org/10.1109/BTAS.2015.7358745.

Cohen, M. W., Voldman, I., Regazzoni, D. & Vitali, A. (2018). Hand rehabilitation via gesture recognition using Leap Motion controller. In 2018 11th international conference on human system interaction (HSI) (pp. 404–41). IEEE.

Cortes, C., & Vapnik, V.Ñ. (1995). Support vector networks. *Machine Learning, 20*, 273–297.

De Cooman, T., Vandecasteele, K., Varon, C., Hunyadi, B., Cleeren, E., Van Paesschen, W., & Van Huffel, S. (2020). Personalizing heart rate-based seizure detection using supervised SVM transfer learning. *Frontiers in Neurology, 11*, 145.

Elliott, S. J. & Hunt, A. R. (2006). The challenge of forgeries and perception of dynamic signature verification. In Proceedings of the 6th international conference on recent advances in soft computing (pp. 455–459).

Farella, E., O'Modhrain, S., Benini, L., & Riccó, B. (2006). Gesture Signature for Ambient Intelligence Applications: A Feasibility Study. In Pervasive computing (pp. 288–304). http://doi.org/10.1007/11748625_18.

Ferrer, M., Alonso, J., & Travieso, C. (2005). Off-line geometric parameters for automatic signature verification using fixed-point arithmetic. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 993–997.

Guerra-Segura, E., Travieso, C. M., & Alonso, J. B. (2017). Study of the variability of the Leap Motion's measures for its use to characterize air strokes. *Measurement: Journal of the International Measurement Confederation, 105*. https://doi.org/10.1016/j.measurement.2017.04.016.

Guru, D. S., & Prakash, H. N. (2009). Online signature verification and recognition: An approach based on symbolic representation. *IEEE Trans. Pattern Analysis and Machine Intelligence, 31*(6), 1059–1073. https://doi.org/10.1109/TPAMI.2008.302.

Haleem Butt, A., Rovini, E., Dolciotti, C., Bongioanni, P., De Petris, G., & Cavallo, F. (2017). Leap motion evaluation for assessment of upper limbs motor skills in Parkinson's disease. *International Conference on Rehabilitation Robotics (ICORR)*, 116–121, 10.0/Linux-x86_64.

Haskell, R. E., Hanna, D. M. & Van Sickle, K. (2006). 3D signature biometrics using curvature moments. In International conference on artificial intelligence (pp. 718–721).

Impedovo, D., & Pirlo, G. (2008). Automatic signature verification – the state of the art. *IEEE Transactions on Systems, Man, and Cybernetics, 38*(5), 609–635.

Kaji, H. & Sugano, M. (2017). A noncontact tremor measurement system using leap motion. In Proceedings of the 6th international conference on informatics, environment, energy and applications (pp. 76–79). http://doi.org/10.1145/3070617.3070628.

Kamaishi, S. & Uda, R. (2016). Biometric authentication by handwriting using Leap Motion. In Proceedings of the 10th international conference on ubiquitous information management and communication – IMCOM '16. http://doi.org/10.1145/2857546.2857583.

Kumar, P., Saini, R., Roy, P. P., & Dogra, D. P. (2017). 3D text segmentation and recognition using leap motion. *Multimedia Tools and Applications, 76*(15), 16491–16510.

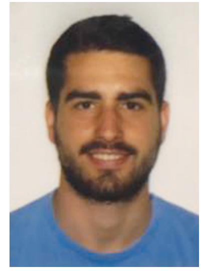
Lahanas, V., Loukas, C., Georgiou, K., Lababidi, H., & Al-Jaroudi, D. (2017). Virtual reality-based assessment of basic laparoscopic skills using the Leap Motion controller. *Surgical Endoscopy*. https://doi.org/10.1007/s00464-017-5503-3.

Lee, L. L., Berger, T., & Aviczer, E. (1996). Reliable on-line human signature verification systems. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 18*(6), 643–647. https://doi.org/10.1109/34.506415.

Liu, J., Zhong, L., Wickramasuriya, J. & Vasudevan, V. (2009). User evaluation of lightweight user authentication with a single tri-axis accelerometer. In 11th international conference on human-computer interaction with mobile devices and services. http://doi.org/10.1145/1613858.1613878.

Adankon, M. M., & Cheriet, M. (2009). Model selection for the LS-SVM. Application to handwriting recognition. *Pattern Recognition, 42*(12), 3264–3270.
Mardia, K. V. (1970). Measures of multivariate skewness and kurtosis with applications. *Biometrika, 57*(3), 519–530.
Mohandes, M., Aliyu, S., & Deriche, M. (2014). Arabic sign language recognition using the Leap Motion controller. In 23rd International symposium on industrial electronics (ISIE) (pp. 960–965).
Okawa, M. (2020). Online signature verification using single-template matching with time-series averaging and gradient boosting. *Pattern Recognition, 102*.
Okumura, F., Kubota, A., Hatori, Y., Matsuo, K., Hashimoto, M., & Koike, A. (2006). A study on biometric authentication based on arm sweep action with acceleration sensor. In International symposium on intelligent signal processing and communications, ISPACS'06. (pp. 219–222). http://doi.org/10.1109/ISPACS.2006.364871.
Parmar, M., Puranik, N., Joshi, D., Malpani, S., & Thakare, B. (2020). State of art survey signature verification techniques 2019. *Asian Journal for Convergence In Technology (AJCT), 5*(3), 91–96.
Placidi, G., Cinque, L., Petracca, A., Polsinelli, M. & Spezialetti, M. (2017). A virtual glove system for the hand rehabilitation based on two orthogonal LEAP motion controllers. In 6th International conference on pattern recognition applications and methods (pp. 184–192).
Plamondon, R., & Srihari, S. N. (2000). On-line and off-line handwriting recognition: A comprehensive survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 22*(1), 63–84. https://doi.org/10.1109/34.824821.
Postolache, G., Carry, F., Lourenço, F., Ferreira, D., Oliveira, R., Girão, P. S., & Postolache, O. (2019). *Serious games based on Kinect and leap motion controller for upper limbs physical rehabilitation. Modern sensing technologies*. Cham: Springer.
Qu, C. (2015). Online Kinect handwritten digit recognition based on dynamic time warping and support vector machine. *Journal of Information and Computational Science, 12*(1), 413–422. http://doi.org/10.12733/jics20105578.
Shannon, C., & Weaver, W. (1949). *The mathematical theory of communication*. Urbana, IL, USA: University of Illinois Press.
Silva, E. S., De Abreu, J. A. O., De Almeida, J. H. P., Teichrieb, V., & Ramalho, G. L. (2013). A preliminary evaluation of the leap motion sensor as controller of new digital musical instruments. *Compmus. Ime. Usp. Br, 12*.
Singh, A. & Viriri, S. (2020). Online signature verification using deep descriptors. In 2020 Conference on information communications technology and society (ICTAS) (pp. 1-6). IEEE.
Suykens, J., & Vandewalle, J. (1999). Least squares support vector machine classifiers. *Neural Processing Letters, 9*, 293–300.
Tang, L., Kang, W., & Fang, Y. (2017). Information divergence-based matching strategy for online signature verification. *IEEE Transactions on Information Forensics and Security, 13*(4), 861–873.
Upadhyay, A., Nadar, S., & Jadhav, R. (2020). Comparative study of SVM & KNN for signature verification. *Journal of Statistics and Management Systems, 23*(2), 191–198.
Vamsikrishna, K. M., Dogra, D. P., & Desarkar, M. S. (2016). Computer-vision-assisted palm rehabilitation with supervised learning. *IEEE Transactions on Biomedical Engineering, 63*(5), 991–1001.
Vargas, J. F., Ferrer, M. A., Travieso, C. M. & Alonso, J. B. (2009). Offline signature verification based on pseudo-cepstral coefficients. In 10th International conference on document analysis and recognition (pp. 126–130). http://doi.org/10.1109/ICDAR.2009.68.
Wang, G., Zhang, G., Choi, K. S., Lam, K. M., & Lu, J. (2020). Output based transfer learning with least squares support vector machine and its application in bladder cancer prognosis. *Neurocomputing, 387*, 279–292.
Wang, H. & Hu, D. (2005). Comparison of SVM and LS-SVM for regression. In 2005 International conference on neural networks and brain (pp. 279–283). Beijing.
Wu, J., Pan, G., Zhang, D., Qi, G., & Li, S. (2009). Gesture Recognition with a 3-D Accelerometer. *Ubiquitous Intelligence and Computing, 25–38*. https://doi.org/10.1007/978-3-642-02830-4_4.

Elyoenai Guerra-Segura received the M.Sc. degree in 2014 in Telecommunication Engineering at University of Las Palmas de Gran Canaria (ULPGC), Spain, obtaining the award for the best academic records of the promotion. He has already 2 postgraduate certifications in TICs for environment and health and in training for Secondary school professor, from the ULPGC. He is beginning his research career and have published some papers in internationals journals related to biometrics and image and signal processing, and a national patent. He got a scholarship for training in technological development at Instituto de Astrofísica de Canarias (IAC) and have done different tasks of technological research at hospitals. Currently, he is studying to obtain the Ph.D. degree and working as developer and analysist in the development of public administration projects.



Aysse Ortega Pérez received the M.Sc. degree in 2017 in Telecommunication Engineering at University of Las Palmas de Gran Canaria (ULPGC), Spain. During his studies, he has studied mainly signal and image processing, reason why he is also starting his research career, focusing on signature identification and verification, and in biometrics applied to the in-air writing. Currently, he is working as web developer for clients of the public administration.

Carlos M. Travieso-González received the M.Sc. degree in 1997 in Telecommunication Engineering at Polytechnic University of Catalonia (UPC), Spain; and Ph.D. degree in 2002 at University of Las Palmas de Gran Canaria (ULPGC-Spain). He is Full Professor and Head of Signals and Communications Department at ULPGC. From 2001, he is teaching in ULPGC subjects on signal processing and learning theory. His research lines are biometrics, biomedical signals and images, data mining, classification system, signal and image processing, machine learning, and environmental intelligence. He has researched in more than 48 International and Spanish Research Projects, some of them as head researcher. He is co-author of 4 books, co-editor of 24 Proceedings Book, Guest Editor for 8 JCR-ISI international journals and up to 25 book chapters. He has over 450 papers published in international journals and conferences (74 of them indexed on JCR – ISI - Web of Science). He has published 7 patents on Spanish Patent and Trademark Office. He has been supervisor on 8 PhD Thesis (10 more are under supervision), and 140 Master Thesis. He is founder of The IEEE IWOBI conference series and President of its Steering Committee, of The InnoEducaTIC conference series; and of The APPIS conference series. He is reviewer of project proposals for European Union (H2020), Medical Research Council (MRC – UK), Spanish Government (ANECA), DAAD (Germany), National Agency of Research (ANR- France), Argentinian Government and Colombian Institutions. He has been reviewer in different indexed international journals (<65) and conferences (<180) since 2001. He is member of IASTED Technical Committee on Image Processing from 2007 and member of IASTED Technical Committee on Artificial Intelligence and Expert Systems from 2011. He will be APPIS 2020 General Chair and Springer-IWOBI 2020, and was IEEE-IWOBI 2019, APPIS 2019 General Chair, IEEE-IWOBI 2018 General Chair, APPIS 2018 General Chair, InnoEducaTIC 2017 General Chair, IEEE-IWOBI 2017 General Chair, IEEE-IWOBI 2015 General Chair, InnoEducaTIC 2014 General Chair, IEEE-IWOBI 2014 General Chair, IEEE-INES 2013 General Chair, NoLISP 2011 General Chair, JRBP 2012 General Chair and IEEE-ICCST 2005 Co-Chair. He is Associate Editor on Computational Intelligence and Neuroscience journal (Hindawi – Q2 JCR-ISI) and Entropy (MPDI – Q2 JCR-ISI). He was Vice-Dean from 2004 to 2010 in Higher Technical School of Telecommunication Engineers in ULPGC; and Vice-Dean of Graduate and Postgraduate Studies from March 2013 to November 2017. He got the awards in Catedra Telefonica 2018, Catedra Telefonica 2019 and Catedra Telefonica 2020 on Knowledge Transfer Modality.