Double serial adaptation mechanism for keystroke dynamics authentication based on a single password

Abir Mhenni, Estelle Cherrier, Christophe Rosenberger, Najoua Essoukri Ben Amara

Double Serial Adaptation Mechanism for Keystroke Dynamics Authentication based on a Single Password

Abir Mhenni[a,b,c,*], Estelle Cherrier[c], Christophe Rosenberger[c], Najoua Essoukri Ben Amara[b]

[a] *ENIT, University of Tunis El Manar, BP 94, Rommana 1068 Tunis, Tunisia*

[b] *Université de Sousse, Ecole Nationale d'Ingénieurs de Sousse, LATIS- Laboratory of Advanced Technology and Intelligent Systems, 4023, Sousse, Tunisie;*

[c] *Normandie Univ, UNICAEN, ENSICAEN, CNRS, GREYC, 14000 Caen, France*

Abstract

Cyber-attacks have spread all over the world to steal information such as trade secrets, intellectual property and banking data. Facing the danger of the insecurity of saved data (personal, professional, official, etc), keystroke dynamics was proposed as an interesting, non-intrusive, inexpensive, permanent and weakly constrained solution for users. Based on the typing rhythm of users, it improves logical access security. Nevertheless, it was demonstrated that such an authentication mechanism would need a larger number of samples to enroll the typing characteristics of users. Moreover, these registered characteristics generally undergo aging effects after a time span. Different solutions have been suggested to remedy these variability problems, including template adaptation. In this paper, we propose a double serial adaptation strategy that considers a single-capture-based enrollment process. When using the authentication system, the template of users and the decision/adaptation thresholds are updated. Experimental results on three public keystroke dynamics datasets show the benefits of the proposed method.

*Corresponding author
Email addresses: abirmhenni@gmail.com (Abir Mhenni), estelle.cherrier@ensicaen.fr (Estelle Cherrier), christophe.rosenberger@ensicaen.fr (Christophe Rosenberger), najoua.benamara@eniso.rnu.tn (Najoua Essoukri Ben Amara)

Keywords: Biometric authentication, Online classification, Template aging, Adaptive strategy, Keystroke dynamics, Template update, Adapted thresholds, KNN-GA

1. Introduction

Authentication systems that provide physical or logical access are currently a major concern. Physical access control verifies the identities of users based on what they are (face, fingerprint [1], etc), what they have (badge, key, etc) and/or what they know (password, pass-code, etc). Logical access control is generally based on the accuracy and conformity of an introduced password and its corresponding login, previously defined when creating the account. Nowadays, passwords are frequently hacked [2] causing various, often serious, types of damage. Keystroke dynamics is a behavioral biometric solution used to control the access of password-based applications [3, 4, 5]. It has the advantage of reinforcing the legitimacy of authenticated people. It differentiates between users' typing manners by extracting characteristics, which are generally composed of the following:

- Timing patterns: Dwell time, flight time and latencies [6] are obtained by calculating the time difference between pressure and release instants corresponding to two or three successive keys, or corresponding to the first and last keys.
- Pressure patterns: Users' pressure on keyboard keys is obtained through an additional instrument (pressure sensor [7]).

However, a major problem of keystroke dynamics characteristics is that the typing manner changes over time. In fact, the initial biometric reference template, created at the enrollment phase, tends to be less representative of a user's typing behavior as time elapses. The main cause of the keystroke template aging is the intra-class variability [8, 9, 10]. This latter may be due to:

- The acquired habit of typing the password: Through its frequent use, users' typing manners vary and cause deviations from the initial reference.
- The state of mind and activeness of users: The behavior of users is highly affected by their mood (stress, anger, happiness, beginning of the day, end of the day, etc).
- The keyboard dissimilarity: The interactions of users with the keyboard change according to its layout (AZERTY or QWERTY), its type (virtual or physical) and even according to the used device [11] (a computer, a laptop, a smart phone).

Different solutions have been proposed to mitigate performance degradation caused by such problems. We mention the presentation of a new query after a false rejection of the last one. It is a simple method, but it does not allow properly taking into consideration the variation in the characteristics of users over time [5]. Multibiometric systems verify the identity of users based on different biometric characteristics. Various modalities have been combined with keystroke dynamics like voice [12], graphical user interface [13], or face [14]. Despite their advantages, multibiometric systems can greatly complicate the configuration of system parameters and may require additional sensors. The use of "soft biometrics" traits to have additional users' characteristics facilitates differentiating several categories of users or increasing the recognition performance of existing keystroke authentication systems [15]. Commonly, these additional characteristics are not unique and not permanent to differentiate between users (*e.g.* gender, age, one or two hands, right or left-handed). The update of a biometric model throughout the use of keystroke dynamics recognition systems takes into account the variability in the typing manner over time. Each accepted query is included in the reference based on a specific mechanism to adapt the modeling of the keystroke dynamics of users.

Update methods generally depend on five parameters [16]: the reference modeling, the adaptation criteria, the adaptation mechanism, the adaptation mode, and the adaptation periodicity.

The majority of studies concerning keystroke adaptation systems only update the corresponding user references [17]. We can mention some studies concerning face adaptation that have updated both the reference and the decision threshold [18]. In the proposed approach, we are inspired by these studies. Besides, the enrollment phase for keystroke adaptation systems is a hard task to achieve. It must be efficiently done to differentiate between genuine and impostor users. As a consequence, the number of samples used in the training phase is generally high.

A couple of contributions are proposed in this paper. First, we put forward an original adaptation strategy for keystroke dynamics that requires a single sample during enrollment (called single enrollment process). This solution considerably improves the usability of keystroke dynamics, since all methods in the state of the art require many samples during the enrollment step. Second, during the operational use of the authentication system, the template of users and their personal decision/adaption thresholds are updated; whereas, in the literature, only the template of users is generally updated when keystroke dynamics modality is dealt with. This new solution, compared to other approaches in the literature, leads to better results on significant and complex datasets used by the scientific community in the field.

This paper is organized as follows: Section 2 presents the literature review on keystroke dynamics and its corresponding adaptation strategies. Section 3 describes the proposed methodology and the contributions of this paper. Section 4 details the experimental protocol and the used datasets. Section 5 shows and discusses the obtained experimental results. Section 6 presents the main conclusions of this work and some perspectives.

2. Background on keystroke dynamics

Official and corporate (e-commerce, e-banking, etc) websites as well as social network and e-mail accounts are increasingly the target of hackers' attacks. According to data collected during September 2017 by Hackmageddon [2], dif-

ferent breaches were noticed. The most significant attacks concerned personal, industrial and official data.

Keystroke dynamics is recognized as an interesting solution to enhance the security of password-based authentication systems. In this section, we recall a brief description on keystroke dynamics systems. Then we present a selection of adaptation strategies applied on this biometric modality.

2.1. Keystroke dynamics modality

Keystroke dynamics is a behavioral biometric modality. It combines the verification of syntactic password accuracy with that of conformity with the behavior of a genuine user: the typing rhythm on the keyboard. Different studies have been conducted to highlight this behavioral modality. The two main coexisting families are defined as follows:

- *Static text authentication* where the user always types the same text. This text is usually a pre-defined password. It may be common for all users (a passphrase), or it may be a user specific password. This is the most utilized category in the literature [15, 19, 20, 21].

- *Free text authentication* where the user does not always enter the same text [22, 23, 24]. There may be continuous authentication, which constantly checks the identity of the user. Challenge-based authentication should be considered in some applications. It asks the user to enter text he/she does not know in advance. The server needs to verify also whether the user typed the assigned text.

In both cases, the extracted characteristics describing the user's typing manner are practically the same. Different classifiers are involved, particularly statistical ones. They are based on calculating statistical characteristics from training samples (e.g. mean, median and standard deviation) and comparing them to those of the new introduced query using various distance metrics. Three main statistical classifiers have been used [25, 26, 27].

In addition, studies using Neural Networks (NN) have been frequently applied to keystroke dynamics [28, 29, 30]. NN have the disadvantages of requiring a huge number of labeled samples (from genuine and impostor users) in order to create a reference template. Moreover, in this case, parameters setting is rather complex. The efficiency of Support Vector Machine (SVM) classifiers has been also tested [31, 21]. They have been used in the context of either one-class or two-class classification (where impostor attacks were considered). For one-class classification, the authors proposed in [32] the Genetic Algorithm (GA)-SVM wrapper approach. They improved the SVM classification by adding the GA to perform features selection. Accordingly, the created user's model demonstrated a better performance, but the number of samples used to create the reference was large as well (equal to 50).

Many other classifiers have been used in the literature for keystroke dynamics authentication systems, such as the Bayesian classification [26], the Hidden Markov Model [4] or the K Nearest Neighbor (KNN) classifier. For example, the authors in [33] opted for the KNN classifier to distinguish genuine samples from impostor ones, and then to create two galleries: a positive one, to save samples classified as genuine, and a negative one, to collect samples classified as an impostor. The positive reference was composed of 40 samples captured during the enrollment phase.

In the literature, most studies have required more than twenty captures to create the reference template during the enrollment phase [21], as depicted in Table 1. However, considering usability, it is not really operational to ask users to type their password 20 times.

In fact, it has been demonstrated that performances increase with the number of enrolled samples in the template. In contrast, another study [21] utilized only five samples per user. The authors considered that "5" is the maximal number of samples for usability reasons in industrial conditions. As keystroke dynamics systems suffer from intra-class variations, reference representativeness is limited in time duration. Next, subsection 2.2 presents some existing adaptation strategies to remedy this lack of representativeness [38, 39].

Table 1: Reference size in enrollment phase for some systems in literature

Works		Reference size in the enrollment phase
Çeker et al.	[34]	20-40-60-80-100-120-140-160
Çeker et al.	[35]	15
Pisani et al.	[33]	40
Yu et al.	[32]	50
Obaidat et al.	[36]	112
Killourhy et al.	[37]	200

2.2. Keystroke dynamics adaptation

Adaptation strategies generally depend on five parameters, as illustrated in Figure 1.

Reference modeling: It concerns the representation of the biometric reference in the enrollment phase. First, the user can be modeled by a single sample. This representation was used in [40] for a secret Personal Identification Number (PIN) system. It could not be considered as a keystroke dynamics since a single finger was used to type a PIN code and the approach was tested by only 10 users. Second, the reference can be composed of several samples. This is the most used representation, generally known as a reference. Third, cluster representation regroups samples in a hierarchical reference [41].

Adaptation criteria: This parameter corresponds to the criterion chosen to initiate the adaptation mechanism. Different criteria have been used for keystroke adaptation systems like the temporal errors distribution [42], the enhanced template update [43], and especially the double thresholds [44]. The latter adds a second threshold stricter than the first one [44] to select the queries for the adaptation phase.

Adaptation mode: It deals with the method to label the queries. It can be realized in a supervised (or manual) or semi-supervised way. The semi-supervised adaptation is based on a classifier decision. It is an automatic and realistic method that has been used a lot in the literature [45, 46, 47]. The

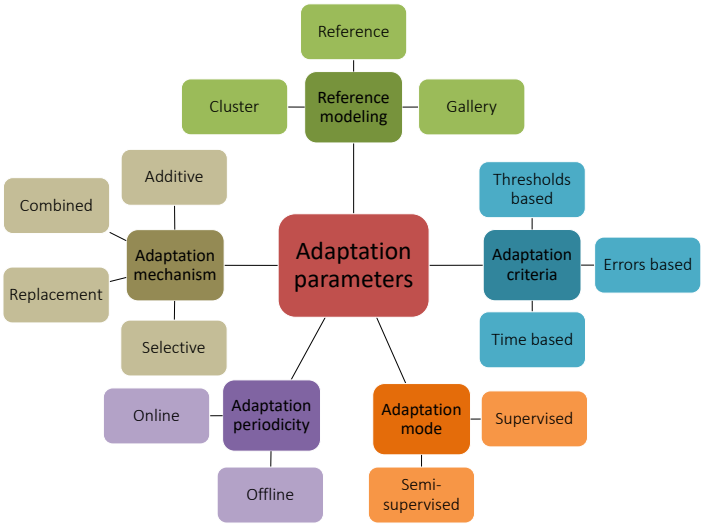


Figure 1: General adaptation parameters

supervised mode is processed by a human operator.

Adaptation periodicity: The template adaptation can be applied in an online or offline process. Indeed, the adaptation can be realized automatically after query authentication. Otherwise, the adaptation is executed after the collection of a specified number of accepted queries or after a period of time [48].

Adaptation mechanism: It details how to apply the adaptation to the reference. There are generally four main mechanisms: additive, replacement, combined and selective. For the additive mechanisms, the most known one is the growing window [49]. This method progressively adds the accepted samples to the reference. Concerning the replacement mechanisms, the sliding window mechanism [49] is the most used approach. It replaces the oldest sample among the reference by the accepted query. An Enhanced Template Update (ETU) mechanism was proposed in [33]. It consists in applying the sliding window mechanism to two references, namely positive and negative galleries containing samples classified as genuine and impostor ones, respectively. Among the combined mechanisms, we can cite the double parallel approach [17] which uses two sub-references: The first one is managed with the growing window mechanism

and the second one is updated with the sliding window. Finally, the selective mechanisms are applied to minimize the reference size. The most used ones are the clustering methods [50] and the editing ones [51]. These methods are generally used for fingerprint and face modalities.

According to [16], there are three major types of adaptive systems:
(1) Update the biometric reference of the user while using the system
(2) Adapt the system parameters depending on the user [52] or the quality of the capture [53]
(3) Adapt the decision frontier overtime [18].

For keystroke authentication, most studies have focused on the first type of adaptation. In this case, the used thresholds are generally fixed or user-dependent [54]. Thereby, the threshold varies for each user, but remains unchanged during the whole adaptation process. Mhenni et al. [47] suggested an adaptation mechanism that adapted the reference and the used thresholds, thus demonstrating competitive performances. In this paper, we propose an extension of this work that considers the requirements of industrialized applications, which is detailed in the next section.

3. Proposed method

We put forward a novel adaptive method that considers a limited number of samples used to create a user's reference while keeping a good performance. Indeed, the user introduces the password only once, when creating a new account. Thus, the reference is composed of a single sample. Afterwards, for each successful authentication, the reference is updated in a transparent way. Avoiding the enrollment phase, the growing window mechanism serves to increase the size of the reference to capture more intra-class variations. Once the size of the reference reaches 10 samples, the sliding window will be considered in order to limit the number of samples saved in the reference. Moreover, the process detailed in Figure 2, contains different contributions as follows :

- We consider a preprocessing step which intends to eliminate the noise in

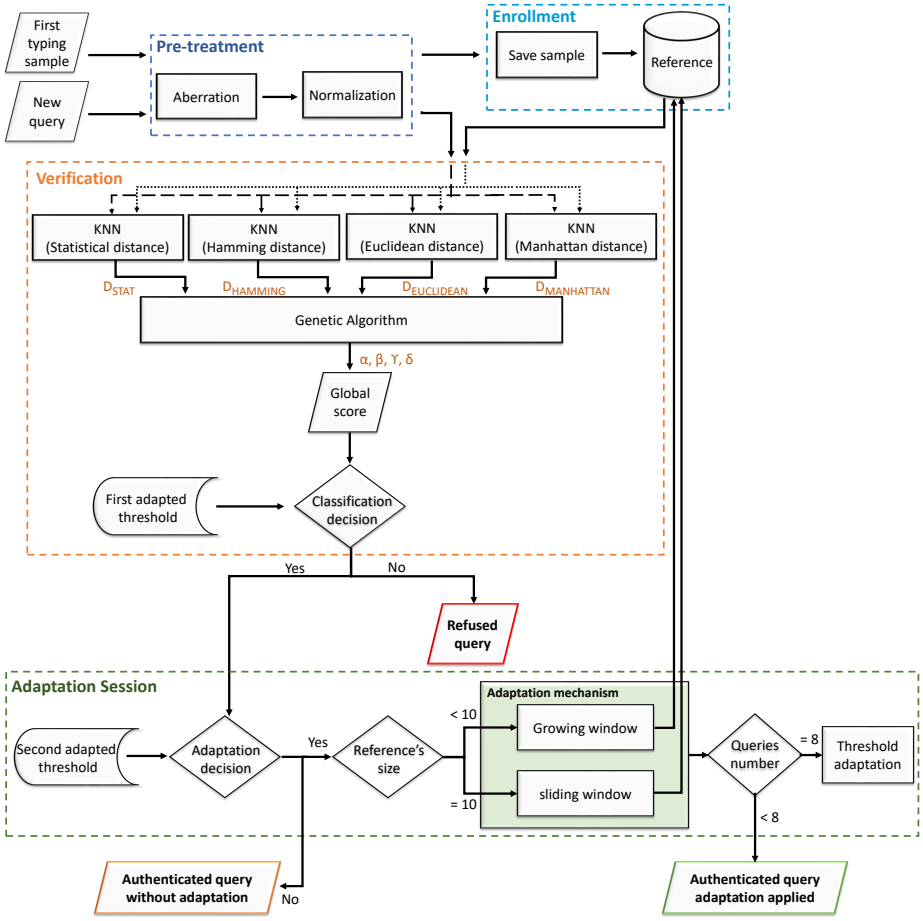


Figure 2: Adaptation process

the captured characteristics (peaks corresponding to hesitation, disturbances or workload of the computer).

- We use a single sample to create a user's reference while avoiding the tedious step of typing the same password several times in the enrollment phase.

- We use a *GA-KNN verification method*: It is based on the optimized combination of multi-distance metrics for the KNN classifier, which shows a better performance. This combination is ensured by vote parameters

that are optimized by GA and updated during the use of the system.

- We propose to adapt the reference and the used thresholds over time. Hence, our method also considers the decision of the *adapted thresholds* criterion (user and time-dependent).

- We resort to a *double serial mechanism*: This progressive adaptation mechanism combines the growing-window and sliding-window mechanisms (respectively before and after reaching the number of required samples, empirically set at 10).

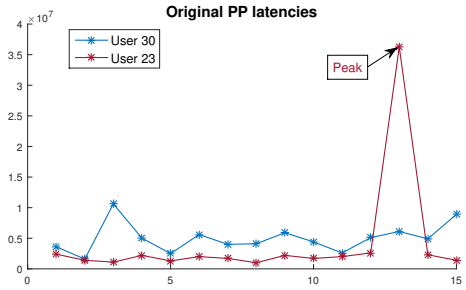
Thus, a new authentication framework is proposed in addition to the adaptation strategy. Indeed, previous works use baseline authentication method to evaluate their update system. Now, we detail our contributions in each step of the process.

3.1. Preprocessing phase

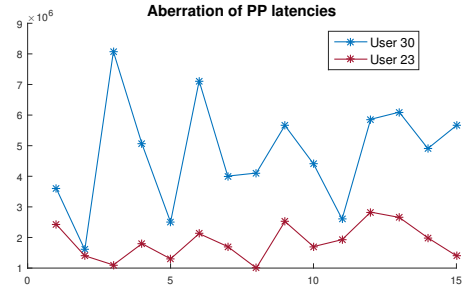
To describe the keystroke dynamics of one user, we are interested in temporal information extracted from digraph transition times:

- PP: time difference between the press events of two successive keys
- RR: latency between the release events of two successive keys
- PR: time duration between a one-key press event and its following key release event
- RP: time duration between a one-key release event and its following key press event

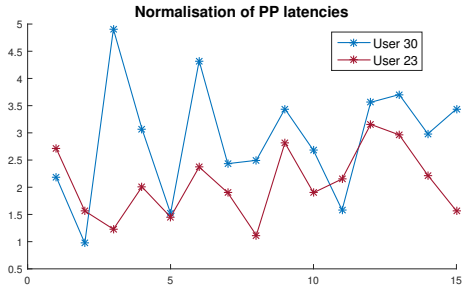
Hence, the characteristic vector C is composed of these temporal informations $C = [PP\ PR\ RR\ RP]$. These characteristics undergo preprocessing steps, as demonstrated in Figure 3. We first apply an aberration correction to the acquired characteristics aiming to detect the peaks where the user takes an abnormally longer time to type a password. In fact, these peaks do not describe



(a) Acquired characteristic vector of PP latencies



(b) Characteristic vector of PP latencies after applying aberration correction



(c) Characteristic vector of normalized PP latencies

Figure 3: Successive preprocessing steps: (b) Aberration correction and (c) Normalization, applied to (a) Characteristic vector of PP latencies.

a user's typing manner. They are generally caused by a disturbance, hesitation time, etc. For that purpose, we first define the peaks as the i^{th} characteristic value $C(i)$ three times greater than the i^{th} value of the standard deviation vec-

tor of the reference $\sigma_C(i)$. The peak is then replaced by the i^{th} value of the mean vector μ_C of the reference. This correction is applied to two peaks of the characteristic vector at most. Equation 1 summarizes this preprocessing step, in case it is applied to the characteristic vector C.

$$\begin{cases} IF\ (C(i) \geq (\ 3 \times \sigma_C(i))\ THEN \\ C(i)\ =\ \mu_C(i) \end{cases} \tag{1}$$

where:
i is the index of the i^{th} character of a vector ;
C is the characteristic vector of each keystroke dynamics acquisition presented to the preprocessing phase;
σ_C is the standard deviation vector of the reference. When the reference contains an only one sample, σ_C is a vector of fixed values (which are the standard deviation value of the one sample reference).
μ_C is the mean vector of the reference.

After that, data normalization is carried out by dividing the characteristic vector by the standard deviation σ of the reference (to ensure a standard deviation of these features to 1). Equation (2) depicts the normalization applied to the characteristic vactor.

$$C(i)\ =\ \frac{C(i)}{\sigma_C(i)} \tag{2}$$

By applying the aberration correction and normalization steps to the 4 considered characteristics, the erroneous data are almost removed. Thus, we obtain a sample composed of four characteristic vectors containing the information necessary to model the users' keystroke dynamics.

3.2. Enrollment phase

Several biometric authentication systems, essentially those based on face and fingerprint modalities [45, 55], have used a single sample in the enrollment step.

This is not the case for keystroke dynamics systems, since they are based on a behavioral modality that quickly changes over time.

According to the literature, the minimal number of samples used during the enrollment phase to create the reference is 5 samples [21]. In this paper, we use characteristics extracted from only a single sample to create reference $\mathcal{G}_j$ of user j, in the enrollment phase. Therefore, the proposed method fits the industrial and operational application conditions, for which a user introduces a password only once when creating an account.

3.3. Verification phase

This phase aims to decide whether to authorize or deny an access for a claimed user. We judge KNN to be the most appropriate classifier as it has proved to be efficient for keystroke dynamics modalities [33], hence the competitive performances. Indeed, a single sample in the reference will not be efficient for the training phase of the other classifiers like NN or SVM. Since the KNN classifier can be applied with a variety of distances, several distance metrics are tested: The Statistical [25], Hamming , Euclidean and Manhattan distances are considered to obtain four respective partial scores D_{STAT}, $D_{HAMMING}$, $D_{EUCLIDIAN}$ and $D_{MANHATTAN}$, as represented in Equation (3). Each novel query is labeled by the KNN classifier using these four distances:

- Statistical distance: It is widely used for classifying keystroke dynamics data. Based on extracting statistical values from each biometric feature (mean and standard deviation), it has the advantage of being easy to calculate and offering competitive performances.

- Hamming distance: It calculates the percentage of coordinates that differ between the query and the reference. The Hamming distance is calculated by positive integer values obtained by multiplying the characteristic vector by 10.

- Euclidean distance: It is a simple distance metric often used with a KNN classifier. It is defined as the square root of the sum of the squares of the

differences between the corresponding coordinates of the new query and the reference samples.

- Manhattan distance: It computes the sum of the differences of the corresponding components of the new query and the reference samples.

$$
\begin{aligned}
D_{STAT} &= 1 - \frac{1}{n}\sum_{i=1}^{n} e^{-\frac{|\ q_j(i)\ -\ \mu_j(i)\ |}{\sigma_j(i)}} \\
D_{HAMMING} &= (\ \#\ (\ q_j(i) \neq \mathcal{G}_j(i)\ )\ /\ n\ ) \\
D_{EUCLIDIAN} &= \sqrt{\sum_{i=1}^{n} (\ q_j(i) - \mathcal{G}_j(i)\ )^2} \\
D_{MANHATTAN} &= \sum_{i=1}^{n} |\ q_j(i) - \mathcal{G}_j(i)\ |
\end{aligned}
\tag{3}
$$

In Equation (3), q_j is the query that claims to belong to user j. Hence, it is matched against its biometric reference $\mathcal{G}_j$ whose corresponding average and standard deviation are μ_j and σ_j. We use these four metrics because we have tested different distance metrics separately, and these ones have demonstrated better performances. The global score $Score_j$ is the weighted sum of the four partial scores. For each user j, we calculate the global score according to Equation (4):

$$
\begin{aligned}
Score_j \ = \ & \alpha_s \times D_{STAT} + \beta_s \times D_{HAMMING} \\
& + \gamma_s \times D_{EUCLIDIAN} + \delta_s \times D_{MANHATTAN}
\end{aligned}
\tag{4}
$$

where $\alpha_s, \beta_s, \gamma_s, \delta_s$ are the vote parameters calculated on session s. The calculated score is compared to a previously set verification threshold. As a result, it is very critical to define the appropriate threshold. Two types of thresholds have been defined in the literature:

- Global threshold: During the use of the system, a constant and unique threshold is involved for all users.

- Individual thresholds: During the use of the system, a user-specific threshold is considered.

In [21], the authors compared these two types of thresholds for the keystroke dynamics modality and showed that the best performances were obtained with the individual threshold. Thereby, we opt for this type of threshold in our experiments. In the next section, we show how to make both thresholds (decision and adaptation) time-dependent.

3.4. Adaptation phase

This subsection presents an innovative adaptation method. It essentially updates both decision and adaption thresholds.

3.4.1. Thresholds adaptation

For the proposed method, we use the double threshold criterion to make the adaptation decision [44]. Two thresholds are used to make two successive decisions. The global score $Score_j$ is compared to a first threshold (decision one) to verify a user's identity. After acceptance, the same score is again compared to a second threshold to decide whether to use the query for adaptation. This adaptation criterion has been deeply used for adaptive systems concerning different modalities (face and fingerprint [44], as well as keystroke dynamics [17]).

In this work, we propose to adapt both thresholds. Some studies [18, 48] have adapted these thresholds and obtained better performances, but threshold adaptation has been mainly utilized for the face modality. These studies demonstrated that the variation in the users' characteristics over time would influence the scores obtained by the classifiers. Consequently, it is better to update the thresholds in order to cope with these variations.

It was already demonstrated in [19, 47, 56, 57] that updating the used thresholds would improve the system performance. An individual decision threshold T_j^{s+1} of session $(s+1)$ is adapted by decreasing it with a coefficient calculated by the average of the mean vector of reference μ and the standard deviation of the standard deviation vector σ, as indicated in Equation (5). The initial

thresholds are fixed to an Equal Error Rate (EER) equal to $\simeq 3\%$ which is defined later in the subsection 4.3. Indeed, this is the best performance we have obtained.

$$T_j^{s+1} = T_j^s - e^{-\frac{\mu_j}{\sigma_j}} \tag{5}$$

Thereby, the thresholds are specific to the user and to the session at the same time.

3.4.2. Template adaptation

The main contributions of the proposed template adaptation method are:
1) It is initialized with a single sample as a reference
2) A multi-distance classifier is considered with adaptive weights.
In fact, we propose a contribution in each of the five components of the template update approach, as depicted in Algorithm 1.

- **Reference modeling:** By initiating the authentication process, users are supposed to type their passwords only once for the computation of the reference template. Afterwards, they can test their identity verification. The main idea is to limit the enrollment phase to a strict minimum and to allow an adaptation of the biometric reference to fit its aging over time. Indeed, it is always mentioned that the enrollment phase annoys users [21]. Even if the proposed scheme does not capture any variability during the enrollment stage, the combination with the proposed adaptation strategy will allow users to cope with it.

- **Adaptation criterion:** Different adaptation criteria have been used in the literature to initiate the adaptation process. Based on the double threshold mechanism, we put forward our new adaptation criterion called *"adapted thresholds"*. As demonstrated in section 3.4.1, it uses individual thresholds that are decreased over time according to Equation (5). In fact, after using the password for a long period, the intra-class variation in the

user's keystroke dynamics becomes lower. This is due to the acquisition of a habit after frequent uses. Therefore, we slightly reduce the threshold during the use of the system.

- **Adaptation mode:** Adaptation is dealt with in a semi-supervised mode through the application of the "*GA-KNN verification method*". Each query is labeled with the KNN classifier and compared to a computed ground-truth based on the reference of the previous session.

  It will be accepted (*i.e.* classified as genuine) if the value of the global score $Score_j$, calculated according to Equation (4), is lower than the *"adapted threshold"*. Equation (4) permits calculating the weighted sum of the four partial scores (D_{STAT}, $D_{HAMMING}$, $D_{EUCLIDIAN}$, $D_{MANHATTAN}$) which are the nearest neighbor scores obtained by the KNN classification with four different distance metrics, defined by Equation (3). The weight parameters (α_s, β_s, γ_s, δ_s) are calculated thanks to GA. Algorithm 1 details the process.

  GA is inspired by the natural evolution process following the Darwinian model [58]. It uses a fitness function to optimize the weight parameters (α_s, β_s, γ_s, δ_s) during a number of iterations (or generations). The content of the initial population is empirically set. We opted for the values that guarantee the best performances for the first adaptation session. For our experimentations, the optimization function of GA minimizes the False Rejection Rate (FRR) and the False Acceptance Rate (FAR) by optimizing the Half-Total Error Rate (HTER). The computation of the FAR and FRR values is based on the presented queries for each adaptation session which are labeled thanks to the GA-KNN. The adopted parameters of the GA algorithm are summarized in Table 2.

  We periodically vary the classification parameters (α_s, β_s, γ_s, δ_s) of Equation (4) to ensure a better performance. Consequently, at the end of each adaptation session, the GA recalculates new global weights for all the users to optimize them. In each session, the fitness of all users is evaluated

Table 2: GA Parameters

Parameter	Value
Population size	50 (number of variables ≤ 5)
Crossover fraction	0.8
Generation	400 (100*number of variables)
Elite count	2.5 (0.05*population size)
Selection function	Stochastic uniform
Crossover function	Crossover scattered
Mutation function	Gaussian

and is usually the value of the defined optimization function.

The great advantage of GA is that it succeeds in finding optimal solutions for very complex problems, so as to take advantage of certain known properties. Furthermore, they are used in applications where a large number of parameters are involved and where it is necessary to obtain good solutions in only few iterations in real-time systems, like in the suggested approach.

- **Adaptation periodicity:** The proposed adaptation strategy operates online. Each accepted query that satisfies the adaptation criterion is included in the adaptation mechanism.

- **Adaptation mechanism:** The initial reference is composed of only a single keystroke dynamics sample. Therefore, the suggested process enriches the reference describing the user's typing manner as shown in Figure 4. At the beginning, the growing window mechanism is adopted. As a result, each request accepted by the adaptation criterion is added to the reference samples. Once the maximal size of the users' reference (set to 10 samples in our work) is reached, the sliding window mechanism will be applied. Thereby, the oldest sample in the reference will be replaced by a new query. Hence, the process is a "*double serial mechanism*".

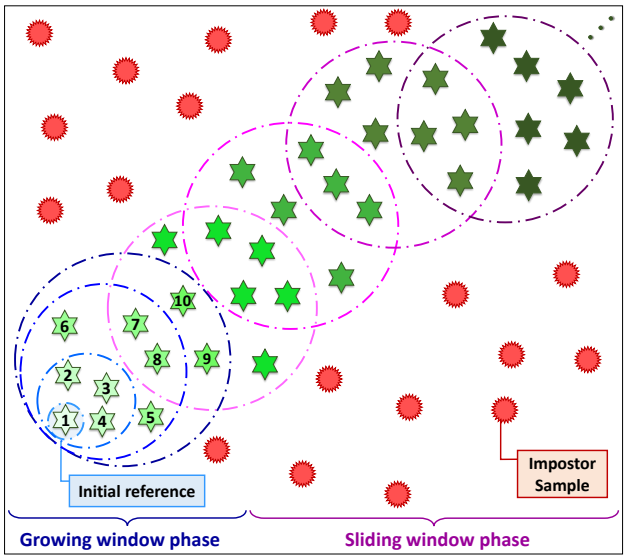


Figure 4: User's reference representation over time: The effects of the *double serial mechanism* on the reference. Each circle represents the reference samples in a specific session.

We noticed that the "double serial mechanism" allows us to obtain a satisfying model of the users' typing rhythm evolution over time. In fact, the novelty is to combine the two considered adaptation mechanisms by applying them sequentially to the same reference. At first, the growing window mechanism is useful for increasing the number of samples representing the users' reference. The purpose of this phase is to enrich the description of the users' typing manner. After that, we update the reference to take into account the intra-class variations over time. Indeed, the sliding window mechanism will start when the size of the reference reaches 10 samples in order to keep a minimal size of the reference (no waste in calculation time). Moreover, this adaptation mechanism is based on the principle that the oldest samples are the least representative of the actual keystroke dynamics of the user. As demonstrated in Figure 4, the newest samples are added while the oldest ones are deleted. In the next section, we demonstrate the efficiency of the proposed method.

Algorithm 1: Template update strategy during an adaptation session.

Require:
$ref_{j_{(t)}}, q_j^s, N \leftarrow maxSize(ref_{j_{(t)}}), (\alpha_0, \beta_0, \gamma_0, \delta_0) = EmpiricalValues$

Ensure: $ref_{j_{(t+1)}}$

for `j = 1 : NumberOfUsers` **do**

  for `e = 1 : 8` **do**

    $D_{STAT} \leftarrow KNN_{Statistical}(ref_{j_{(t)}}, q_j^e, K = 1)$

    $D_{HAMMING} \leftarrow KNN_{Hamming}(ref_{j_{(t)}}, q_j^e, K = 1)$

    $D_{EUCLIDIAN} \leftarrow KNN_{Euclidean}(ref_{j_{(t)}}, q_j^e, K = 1)$

    $D_{MANHATTAN} \leftarrow KNN_{Manhattan}(ref_{j_{(t)}}, q_j^e, K = 1)$

    $Score_j = \alpha_s \times D_{STAT} + \beta_s \times D_{HAMMING} + \gamma_s \times D_{EUCLIDIAN} + \delta_s \times D_{MANHATTAN}$

    if ($Score_j < adaptatedThreshold$) **then**

      if ($size(ref_{j_{(t)}}) < N$) **then**

        $ref_{j_{(t+1)}} \leftarrow GrowingWindow(ref_{j_{(t)}}, q_j^e)$

      else

        $ref_{j_{(t+1)}} \leftarrow SlidingWindow(ref_{j_{(t)}}, q_j^e)$

      end if

    end if

  end

end

$\alpha_{s+1}, \beta_{s+1}, \gamma_{s+1}, \delta_{s+1} \leftarrow GeneticAlgorithm((\alpha_s, \beta_s, \gamma_s, \delta_s); (D_{STAT}, D_{HAMMING}, D_{EUCLIDIAN}, D_{MANHATTAN}))$

4. Experiments

In this section, we put forward the used datasets, the processing description and the investigated performance metrics. Moreover, we present the evolution of some parameters of the experiments like the reference size and the weight parameters over the adaptation sessions.

4.1. Datasets

For our experiments, we chose three datasets, among the most widely used in the literature, to validate the proposed approach:

- GREYC 2009 [59]: This database was developed within the GREYC Laboratory. One hundred and thirty-three users participated in the creation

of this database and typed the same password "greyc laboratory". Only 100 of them participated in five acquisition sessions during two months and provided 60 samples per user. These samples were focused on in our experiments. This database were chosen to compare our results with those of the experiments in [21].

- CMU [20]: This database includes data of 51 users. They typed the same password 400 times during eight acquisition sessions. The defined password was ".tie5Roanl". We opted for this database because it was frequently used in the literature.

- GREYC-Web [60]: For this database, 118 users were involved in its creation and typed the same password "SÉSAME". Only 45 among them participated in five sessions and provided 60 patterns. These users were the subject of our experiment.

4.2. Data stream generation

To evaluate the performance of the proposed system and to follow its evolution, we divided the adaptation process into different sessions. For each session, we introduced eight new queries to the system. These data were divided into five genuine samples and three impostor ones for each adaptation session. First, 5 genuine queries are presented to the authentication process.They were presented according to the chronological order of the database safeguard. Subsequently, the three imposter queries were randomly introduced.

The biometric data stream was then divided into 37.5% (3/8) of impostor samples and 62.5% (5/8) of genuine samples. The attack rate was higher than that generally used in keystroke dynamics studies [33, 61] (70% for genuine samples and 30% for impostor ones).

For both GREYC-2009 and GREYC-WEB databases, we had 60 samples for each user. These samples were divided into 12 sessions (5 *genuine samples/session*). As we used the first sample as initial reference, we presented in the last session only four genuine samples. The impostor attacks were randomly generated by

the samples of other users of the database. For the CMU database, 400 samples per user are available. The system operates for 80 sessions.

4.3. Performance metrics

In the experimental results, the following performance metrics were adopted to evaluate the proposed approach:

- False Non Match Rate (FNMR): It measures the ratio of legitimate users falsely rejected by the system.
- False Match Rate (FMR): It measures the ratio of impostor users accepted by the system.
- Equal Error Rate (EER): It is the error value at which the FNMR value equals the FMR value. This metric is computed when the adaptive threshold is not used.
- Area Under Curve (AUC): It is the measure of the area of the surface below the Detection Error Trade-of (DET) curve (FMR versus FNMR).
- Accuracy: It calculates the proportion of true acceptance and true rejection in all evaluated cases.

We intend to compare the obtained results with other studies in the literature based on these performance measures.

4.4. Classification parameters

In this work, we opted for a KNN classifier based on multi-distances. Thus, to set the values of the vote parameters (α_s, β_s, γ_s, δ_s) of Equation (4), we used GA. It is a widespread algorithm that provides high-quality solutions for optimization problems. Its advantage is that it can start from a collection of randomly generated data. This is quite similar to our experimentation conditions, where the initial reference contains only the first sample.

The initial values of the vote parameters are empirically set. We opted for the values that guarantee the best performances for the first adaptation session.

Table 3: Classification parameters obtained with GA during 12 sessions for GREYC 2009 database

Adaptation	**Parameters**			
sessions	α	β	γ	δ
1	0.0381	0.6295	0.3327	-0.0002
2	0.0289	0.5662	0.3143	0.0906
3	0.0183	0.6298	0.2562	0.0958
4	0.0560	0.6437	0.2854	0.0149
5	0.0404	0.6534	0.2867	0.0196
6	0.0482	0.6024	0.3172	0.0322
7	0.0506	0.6924	0.1904	0.0667
8	0.0327	0.6581	0.2582	0.0511
9	0.0616	0.6684	0.2475	0.0225
10	-0.0593	0.6681	0.3743	0.0170
11	0.0374	0.6936	0.2732	-0.0042
12	0.0468	0.6411	0.2460	0.0661

At the end of each session, after the presentation of 8 new queries, we restart the GA to update the weight parameters. These new parameters would guarantee minimal FRR and FAR rates.

This process was repeated for each adaptation session. Table 3 and Table 4 present the weight values obtained at the end of each adaptation session for GREYC-2009 and GREYC-WEB databases, respectively. We can notice that the Hamming and Euclidean distances (β_s and γ_s) have in general the highest vote values. In fact, these two distances demonstrated also better performances than the others while testing each of them separately with the KNN classifier [57].

Table 4: Classification parameters obtained with GA during 12 sessions for GREYC-WEB database

Adaptation	**Parameters**			
sessions	α	β	γ	δ
1	0.1611	0.4291	0.4201	-0.0103
2	0.1127	0.4612	0.3666	0.0595
3	0.1424	0.4427	0.3603	0.0546
4	0.1694	0.4669	0.3963	-0.0326
5	0.1325	0.4219	0.3754	0.0702
6	0.1425	0.4333	0.3368	0.0874
7	0.1369	0.3822	0.4118	0.0691
8	0.1191	0.4675	0.3841	0.0293
9	0.1453	0.3993	0.3875	0.0679
10	0.1078	0.4995	0.4102	-0.0175
11	0.1114	0.4926	0.03203	0.0757
12	0.1239	0.4366	0.3851	0.0544

4.5. Reference size

As previously mentioned, the initial reference contains only one sample of the genuine user introduced during the enrollment phase. Throughout the adaptation strategy, the reference size increases over time by adding each accepted query to the user's reference. Once the maximal size (ten samples) is reached, the reference size will remain stable.

Since the number of accepted queries is not the same for all users, the size of the reference differs from one user to another at the end of the session. We followed the reference size variation during all adaptation sessions to separate between sessions belonging to the growing window phase and those belonging to the sliding window one.

As depicted in Table 5, 6 and 7, the growing window phase operates over a

Table 5: Size of references in the beginning of each adaptation session for GREYC 2009

Reference size	1	2	3	4	5	6	7	8	9	10
Session 1	100%	-	-	-	-	-	-	-	-	-
Session 2	-	22%	43%	33%	2%	-	-	-	-	-
Session 3	-	-	-	2%	5%	18%	25%	30%	14%	6%
Session 4	-	-	-	-	-	-	1%	2%	5%	92%
Session 5-10	-	-	-	-	-	-	-	-	-	100%

Table 6: Size of references in the beginning of each adaptation session for GREYC-WEB

Reference size	1	2	3	4	5	6	7	8	9	10
Session 1	100%	-	-	-	-	-	-	-	-	-
Session 2	-	20%	51.1%	28.9%	-	-	-	-	-	-
Session 3	-	-	-	-	4.5%	22.2%	33.3%	20%	6.7%	13.3%
Session 4	-	-	-	-	-	2.2%	-	-	4.4%	93.4%
Session 5-10	-	-	-	-	-	-	-	-	-	100%

Table 7: Size of references in the beginning of each adaptation session for CMU

Reference size	1	2	3	4	5	6	7	8	9	10
Session 1	100%	-	-	-	-	-	-	-	-	-
Session 2	-	-	46%	54%	-	-	-	-	-	-
Session 3	-	-	-	-	-	-	8%	40%	38%	14%
Session 4-80	-	-	-	-	-	-	-	-	-	100%

limited number of sessions. Its duration does not exceed three sessions for some users and 5 sessions for all users. This implies that the number of false rejections is not high since the beginning. Our experimentations demonstrate that despite

the lack of samples in the initial reference, the classifier can distinguish between novel queries. Few users have a reference size equal to 5 at the end of the first session (only 2 users for the GREYC-2009 database). For the following sessions, the number of genuine accepted queries goes up quickly especially for the CMU database.

5. Experimental results and discussion

In this section we detail the obtained results for each adaptation session. Different performances have been considered to validate the proposed approach. We tested the proposed approach while considering two scenarios: with and without adaptive thresholds.

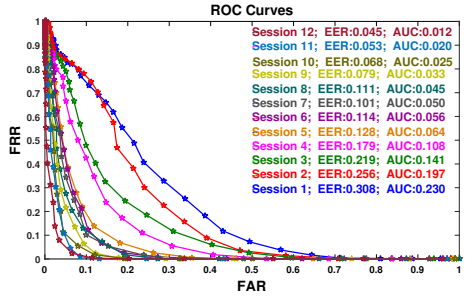
5.1. Performance without adaptive thresholds

We chose to draw the DET curve and calculate the EER as well as the AUC to show the system performance in relation to the reference without considering the adapted threshold criterion.

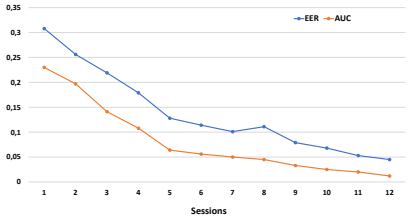
Figure 5 depicts the EER and AUC values of each adaptation session for the three considered databases. Concerning the CMU database, as the number of sessions is quite high (80 sessions), we illustrate only the performances of every ten sessions. We can conclude that the results are slightly improved in each session. The performance improvements during the sliding window phase are much clearer than those of the growing window one. These performances are expected since the reference is not entirely defined at the beginning.

The final result of the last session illustrates a statistically significant improvement. The obtained performances (EER, AUC) in the last session are much improved compared to those of the first one, as shown in Figure 5.

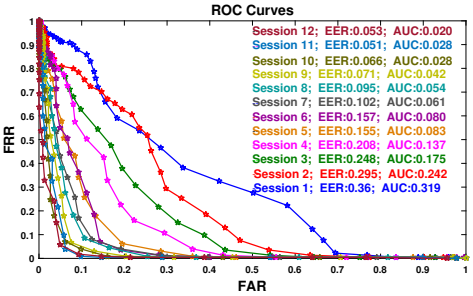
We compared the proposed method to different mechanisms of previous work. For GREYC-2009 database, the proposed method was compared to the average mechanism [21], which was applied to a reference initially composed of five samples and not exceeding 15. For that, we investigated various threshold



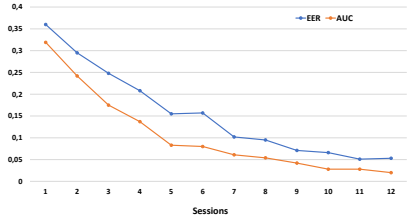
(a) GREYC 2009 database



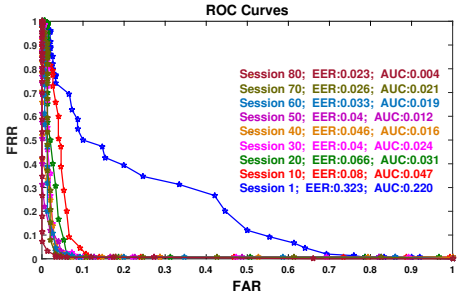
(b) GREYC 2009



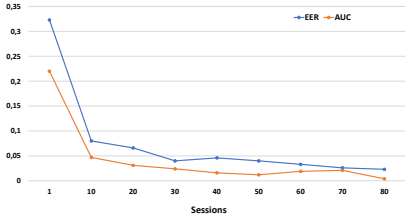
(c) GREYC-WEB database



(d) GREYC-WEB database



(e) CMU database



(f) CMU database

Figure 5: DET curves and associated performance results (EER, AUC) for all adaptation sessions.

types: global, individual. Once again, the proposed adaptation mechanism led to better performances. Table 8 shows the comparison of the obtained results on the GREYC-2009 database. In this paper, the considered performances of the average mechanism are those obtainned by [21].

Table 8: Comparison of obtained results with different thresholds for GREYC-2009 database

	Double serial mechanism		Average mechanism [21]	
Threshold	Reference size	EER %	Reference size	EER %
Global	1-10	4.5%	5-15	6.96%
Individual	1-10	6.5%	5-15	6.95%

To highlight the impact of the proposed adaptation mechanism, we implemented the proposed method with different threshold types. As depicted in Table 9, we show the performance on the three datasets with different thresholds.

Table 9: Comparison of the EER performances with different thresholds

Threshold	GREYC 2009	GREYC-WEB	CMU
Global	4.5%	5.3%	2.3%
Individual	6.5%	8.7%	5.2%

To illustrate the advantages of the proposed adaptation approach, we applied other algorithms of the literature to the GREYC-WEB database. We firstly tested the growing window mechanism with a reference containing a single sample initially. The size of the reference increases up to 43 owing to the adaptation mechanism. Secondly, we applied the sliding window mechanism based on a 10-sized reference. Thirdly, we also tested the proposed double serial mechanism while the reference was initialized to 5 samples and its maximum size was fixed to 10. Finally, the double parallel mechanism was conducted us-

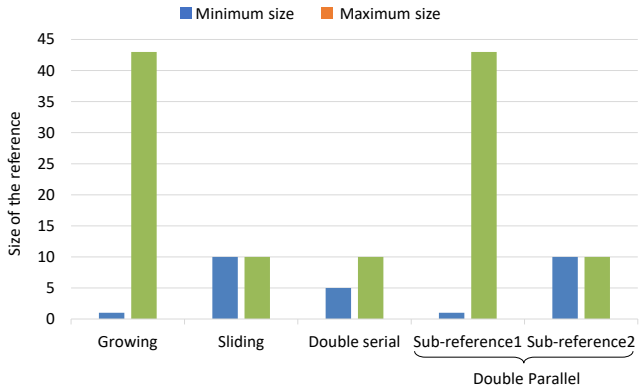


Figure 6: Minimum and maximum reference size for compared mechanisms.

ing two sub-references. One of them initially contained a single sample and it was adapted with the growing window mechanism. The other one initially comprised 10 samples and it was adapted with the sliding window mechanism. Figure 6 depicts the size variations for each adaptation mechanism.

All of these mechanisms were implemented by the GA-KNN method based on the weighted vote of 4 distance metrics. The obtained results are summarized in Figure 7. With a reference size approximately equal to the proposed approach, the double serial mechanism was the best performing among the tested mechanisms. While increasing the initial size of the reference by five samples, we obtained better performances. This was due to the larger description of the keystroke dynamics of users. However, the performance difference at the final session was not very large. Thus, we chose an approach based on a single sample in the learning phase in order to familiarize it with the industrial application environment.

5.2. Performance with adaptive thresholds

The overall results of FMR, FNMR and accuracy concerning the three considered databases are shown in Table 10. These results are calculated over all adaptation sessions while considering the whole data of the databases. The best

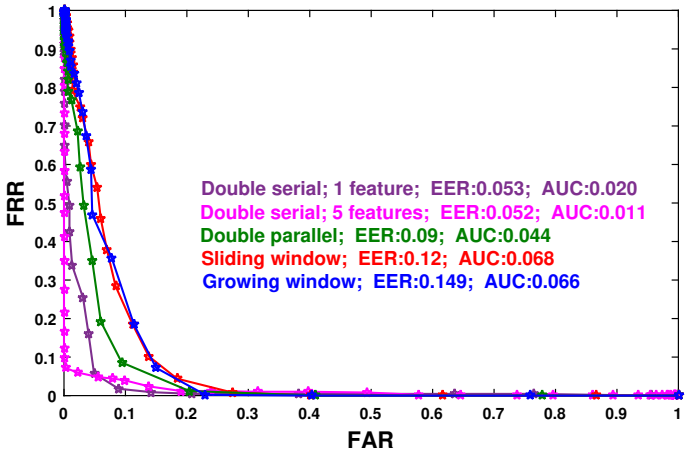


Figure 7: DET curves of last adaptation sessions and associated performances (EER, AUC) of different adaptation mechanisms applied to GREYC-WEB database

achieved results are those obtained with the GREYC-2009 database. While considering the EER and AUC performances, the CMU database presents the best obtained results.

Table 10: Overall performances for three considered databases

	GREYC-2009	GREYC-WEB	CMU
FMR	0.0833	0.1375	0.1406
FNMR	0.0463	0.0516	0.0647
Accuracy	0.828	0.810	0.794

We also compared the proposed method with some other work from the literature to analyze the impact of the number of samples used in the reference, especially in the training phase (See Table 11). For the CMU database, the best obtained result in [33] was 0.670 accuracy, although the reference was obtained by 40 samples. The results achieved with the suggested method were much better with a unique sample as an initial reference template. We obtained 0.794

accuracy.

Table 11: Performance comparison

Database	Adaptation mechanism	Reference size	Threshold	FNMR	FMR	Accuracy
CMU	Double serial mechanism	1-10	Variable	0.064	0.140	0.794
CMU	Enhanced template update [33]	40	Global	0.088	0.573	0.670
WEB-GREYC	Double serial mechanism	1-10	Variable	0.051	0.137	0.810
WEB-GREYC	Enhanced template update [33]	40	Global	0.042	0.355	0.802

5.3. Computation time

As the proposed method is processing online, we are interested in the computation time of each phase. Table 12 presents the computation time of each phase for a single user and for all considered users of the GREYC 2009 database. Concerning the computation time of a unique user, the average computation time is considered. Timing is calculated on CPU with an Intel i7 processor with a speed of 2.5 GHz and 8-Gb RAM. The adaptation phase is faster than the other steps of the process. All phases have a fast computing time except GA which operates in an offline way, so it does not affect the operating time of the proposed approach.

The proposed method had the advantage of minimizing the computation time to create the reference that was very important for the online adaptation mechanism. The experimental results showed that the obtained performance outperformed the other methods in the state of the art for the same databases and under the same test protocol conditions. Furthermore, the proposed method satisfied the suggested conditions in an industrial context. Indeed, a single sample was necessary during the enrollment step. It was a great advantage instead of asking users to type their password multiple times. Moreover, the experiment includes a novel authentication system based on the GA-KNN in addition to the adaptation strategy, thus it demonstrates competitive performances.

Table 12: Computation time in seconds involved in each phase of process for only one user and for all users (ϵ means negligible)

Phase	One user	All users	Relative time for all users
Feature extraction	0.035	4.53	17.70%
Pre-processing: aberration	0.012	0.98	3.82%
Pre-processing: normalization	0.000015	0.00162	ϵ
Enrollment	0.00009	0.0015	ϵ
Verification: Statistical	0.006	0.63	2.46%
Verification: Hamming	0.002	0.187	0.73%
Verification: Euclidean	0.0015	0.158	0.61%
Verification: Manhattan	0.0017	0.176	0.68%
Genetic algorithm	-	18.93	73.97%
Adaptation	3.2395e-05	0.0012	ϵ

6. Conclusion and perspectives

Adaptive biometric strategies provide an important solution to remedy the intrinsic intra-class variations in behavioral biometric authentication systems. As the keystroke dynamics is a biometric modality that suffers from continuous variations over time, adaptive methods are a good solution to compensate for this trouble. Most of the existing studies have used a huge number of samples to create the reference describing the users' typing rhythm in the enrollment phase.

This paper has investigated a solution that enables modeling an individual's keystroke dynamics while minimizing the used samples for the definition of the reference template. For this purpose, we proposed a single enrollment process (the password was typed only once during the enrollment step). The size of each user reference would increase while using the system, to reach a maximum size equal to 10 thanks to the *double serial mechanism*. Actually, the growing window first serves to enlarge the users' galleries so as to capture more intra-class variability. When the maximum size of the reference is attained, the sliding window will take place and allow following the temporal variation in the users'

keystroke dynamics. The proposed contribution is an interesting solution as it satisfies industrial needs (usable enrollment and good efficiency).

We also evolved a *GA-KNN verification method* to achieve better performances during the whole adaptation session. Indeed, the weights from different distances for the KNN classifier, in addition to the GA optimization, are useful to minimize recognition errors. With regards to previous work, the suggested method shows a great performance improvement. As it has been applied on several databases, it has demonstrated competitive performances in each database.

As perspectives, we are involved in a novel approach that may improve the performances of the first sessions so as to make the keystroke dynamics modality more compatible with industrialization conditions. Thus, preliminary experiments of a user specific adaptive mechanism are being conducted. Besides, it will be worth applying and comparing the proposed method to other devices like mobile phones and to other modalities like voice and touch screen interactions.

References

[1] L. Rzouga Haddada, B. Dorizzi, N. Essoukri Ben Amara, A combined watermarking approach for securing biometric data, Signal Processing: Image Communication 55 (2017) 23–31.

[2] P. Passeri, Information security timelines and statistics, hackmageddon.com.

[3] Y. Sun, H. Ceker, S. Upadhyaya, Anatomy of secondary features in keystroke dynamics - achieving more with less, in: 2017 IEEE International Conference on Identity, Security and Behavior Analysis (ISBA), 2017, pp. 1–6. doi:10.1109/ISBA.2017.7947691.

[4] R. Rodrigues, G. Yared, C. do N. Costa, J. Yabu-Uti, F. Violaro, L. Ling, Biometric access control through numerical keyboards based on keystroke dynamics, Advances in biometrics (2005) 640–646.

[5] L. C. Araújo, L. H. Sucupira, M. G. Lizarraga, L. L. Ling, J. B. T. Yabu-Uti, User authentication through typing biometrics features, IEEE transactions on signal processing 53 (2) (2005) 851–855.

[6] R. S. Gaines, W. Lisowski, S. J. Press, N. Shapiro, Authentication by keystroke timing: Some preliminary results, Tech. rep., DTIC Document (1980).

[7] H. Nonaka, M. Kurihara, Sensing pressure for authentication system using keystroke dynamics., in: International Conference on Computational Intelligence, Citeseer, 2004, pp. 19–22.

[8] C. Epp, M. Lippold, R. L. Mandryk, Identifying emotional states using keystroke dynamics, in: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '11, ACM, New York, NY, USA, 2011, pp. 715–724. doi:10.1145/1978942.1979046.

[9] A. N. H. Nahin, J. M. Alam, H. Mahmud, K. Hasan, Identifying emotion by keystroke dynamics and text pattern analysis, Behaviour & Information Technology 33 (9) (2014) 987–996.

[10] C. Gonzalez, B. Best, A. F. Healy, J. A. Kole, L. E. Bourne, A cognitive modeling account of simultaneous learning and fatigue effects, Cognitive Systems Research 12 (1) (2011) 19–32.

[11] P. Bours, J. Ellingsen, Cross keyboard keystroke dynamics, in: 2018 1st International Conference on Computer Applications & Information Security (ICCAIS), IEEE, 2018, pp. 1–6.

[12] J. R. Montalvao Filho, E. O. Freire, Multimodal biometric fusion?joint typist (keystroke) and speaker verification, in: Telecommunications symposium, 2006 international, IEEE, 2006, pp. 609–614.

[13] K. O. Bailey, J. S. Okolica, G. L. Peterson, User identification and authentication using multi-modal behavioral biometrics, Computers & Security 43 (2014) 77–89.

[14] R. Giot, B. Hemery, C. Rosenberger, Low cost and usable multimodal biometric system based on keystroke dynamics and 2d face recognition, in: Pattern Recognition (ICPR), 2010 20th International Conference on, IEEE, 2010, pp. 1128–1131.

[15] S. Z. S. Idrus, E. Cherrier, C. Rosenberger, P. Bours, Soft biometrics for keystroke dynamics: Profiling individuals while typing passwords, Computers & Security 45 (2014) 147–155.

[16] R. Giot, Contributions à la dynamique de frappe au clavier: multibiométrie, biométrie douce et mise à jour de la référence, Ph.D. thesis, Université de Caen (2012).

[17] R. Giot, C. Rosenberger, B. Dorizzi, Hybrid template update system for unimodal biometric systems, in: Biometrics: Theory, Applications and Systems (BTAS), 2012 IEEE Fifth International Conference on, IEEE, 2012, pp. 1–7.

[18] A. Drygajlo, W. Li, K. Zhu, Q-stack aging model for face verification, in: Signal Processing Conference, 2009 17th European, IEEE, 2009, pp. 65–69.

[19] A. Mhenni, E. Cherrier, C. Rosenberger, N. Essoukri Ben Amara, Adaptive biometric strategy using doddington zoo classification of users keystroke dynamics, in: 2018 14th International Wireless Communications & Mobile Computing Conference (IWCMC), 2018, pp. 488–493. doi:10.1109/IWCMC.2018.8450401.

[20] K. Killourhy, R. Maxion, Why did my detector do that?!, in: International Workshop on Recent Advances in Intrusion Detection, Springer, 2010, pp. 256–276.

[21] R. Giot, M. El-Abed, B. Hemery, C. Rosenberger, Unconstrained keystroke dynamics authentication with shared secret, Computers & Security 30 (6???7) (2011) 427 – 445. doi:http://doi.org/10.1016/j.cose.2011.03.004.

[22] K. Xi, Y. Tang, J. Hu, Correlation keystroke verification scheme for user access control in cloud computing environment, The Computer Journal (2011) bxr064.

[23] A. Messerman, T. Mustafić, S. A. Camtepe, S. Albayrak, Continuous and non-intrusive identity verification in real-time environments based on free-text keystroke dynamics, in: Biometrics (IJCB), 2011 International Joint Conference on, IEEE, 2011, pp. 1–8.

[24] P. Pinto, B. Patrão, H. Santos, Free typed text using keystroke dynamics for continuous authentication, in: IFIP International Conference on Communications and Multimedia Security, Springer, 2014, pp. 33–45.

[25] S. Hocquet, J.-Y. Ramel, H. Cardot, User classification for keystroke dynamics authentication, in: International Conference on Biometrics, Springer, 2007, pp. 531–539.

[26] S. Bleha, C. Slivinsky, B. Hussien, Computer-access security systems using keystroke dynamics, IEEE Transactions on Pattern Analysis and Machine Intelligence 12 (12) (1990) 1217–1222. doi:10.1109/34.62613.

[27] K. Revett, S. T. De Magalhães, H. M. Santos, Enhancing login security through the use of keystroke input dynamics, in: International Conference on Biometrics, Springer, 2006, pp. 661–667.

[28] A. A. Ahmed, I. Traore, Biometric recognition based on free-text keystroke dynamics, IEEE Transactions on Cybernetics 44 (4) (2014) 458–472. doi:10.1109/TCYB.2013.2257745.

[29] P. Kobojek, K. Saeed, Application of recurrent neural networks for user verification based on keystroke dynamics, Journal of Telecommunications and Information Technology 3 (3) (2016) 80–90.

[30] H. Çeker, S. Upadhyaya, Sensitivity analysis in keystroke dynamics using convolutional neural networks, in: Information Forensics and Security (WIFS), 2017 IEEE Workshop on, IEEE, 2017, pp. 1–6.

[31] Y. Sang, H. Shen, P. Fan, Novel impostors detection in keystroke dynamics by support vector machine, in: Parallel and distributed computing: applications and technologies, Springer, 2004, pp. 666–669.

[32] E. Yu, S. Cho, Keystroke dynamics identity verification???its problems and practical solutions, Computers & Security 23 (5) (2004) 428 – 440. doi:http://doi.org/10.1016/j.cose.2004.02.004.

[33] P. H. Pisani, R. Giot, A. C. De Carvalho, A. C. Lorena, Enhanced template update: Application to keystroke dynamics, Computers & Security 60 (2016) 134–153.

[34] H. Ceker, S. Upadhyaya, Transfer learning in long-text keystroke dynamics, in: 2017 IEEE International Conference on Identity, Security and Behavior Analysis (ISBA), 2017, pp. 1–6. doi:10.1109/ISBA.2017.7947710.

[35] H. Ceker, S. Upadhyaya, Adaptive techniques for intra-user variability in keystroke dynamics, in: 2016 IEEE 8th International Conference on Biometrics Theory, Applications and Systems (BTAS), 2016, pp. 1–6. doi:10.1109/BTAS.2016.7791156.

[36] M. S. Obaidat, B. Sadoun, Verification of computer users using keystroke dynamics, IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics) 27 (2) (1997) 261–269.

[37] K. S. Killourhy, R. Maxion, et al., Comparing anomaly-detection algorithms for keystroke dynamics, in: Dependable Systems & Networks, 2009. DSN'09. IEEE/IFIP International Conference on, IEEE, 2009, pp. 125–134.

[38] A. Rattani, B. Freni, G. L. Marcialis, F. Roli, Template update methods in adaptive biometric systems: A critical review, in: International Conference on Biometrics, Springer, 2009, pp. 847–856.

[39] N. Poh, A. Rattani, F. Roli, Critical analysis of adaptive biometric systems, IET biometrics 1 (4) (2012) 179–187.

[40] N. Grabham, N. White, Use of a novel keypad biometric for enhanced user identity verification, in: Instrumentation and Measurement Technology Conference Proceedings, 2008. IMTC 2008. IEEE, IEEE, 2008, pp. 12–16.

[41] A. Lumini, L. Nanni, A clustering method for automatic biometric template selection, Pattern Recognition 39 (3) (2006) 495–497.

[42] A. Serwadda, Z. Wang, P. Koch, S. Govindarajan, R. Pokala, A. Goodkind, D.-G. Brizan, A. Rosenberg, V. V. Phoha, K. Balagani, Scan-based evaluation of continuous keystroke authentication systems, IT Professional 15 (4) (2013) 20–23.

[43] P. H. Pisani, A. C. Lorena, A. C. de Carvalho, Adaptive approaches for keystroke dynamics, in: Neural Networks (IJCNN), 2015 International Joint Conference on, IEEE, 2015, pp. 1–8.

[44] A. Rattani, Adaptive biometric system based on template update procedures, Dept. of Elect. and Comp. Eng., University of Cagliari, PhD Thesis.

[45] C. Ryu, H. Kim, A. K. Jain, Template adaptation based fingerprint verification, in: Pattern Recognition, 2006. ICPR 2006. 18th International Conference on, Vol. 4, IEEE, 2006, pp. 582–585.

[46] N. Poh, J. Kittler, A. Rattani, Handling session mismatch by semi-supervised-based co-training scheme, in: Adaptive Biometric Systems, Springer, 2015, pp. 35–49.

[47] A. Mhenni, C. Rosenberger, E. Cherrier, N. Essoukri Ben Amara, Keystroke template update with adapted thresholds, in: Advanced Technologies for Signal and Image Processing (ATSIP), 2016 2nd International Conference on, IEEE, 2016, pp. 483–488.

[48] A. Rattani, G. L. Marcialis, F. Roli, Self adaptive systems: An experimental analysis of the performance over time, in: Computational Intelligence

in Biometrics and Identity Management (CIBIM), 2011 IEEE Workshop on, IEEE, 2011, pp. 36–43.

[49] P. Kang, S.-s. Hwang, S. Cho, Continual retraining of keystroke dynamics based authenticator, Advances in biometrics (2007) 1203–1211.

[50] U. Uludag, A. Ross, A. Jain, Biometric template selection and update: a case study in fingerprints, Pattern Recognition 37 (7) (2004) 1533–1542.

[51] B. Freni, G. Marcialis, F. Roli, Template selection by editing algorithms: A case study in face recognition, Structural, Syntactic, and Statistical Pattern Recognition (2008) 745–754.

[52] S. Hocquet, J.-Y. Ramel, H. Carbot, Estimation of user specific parameters in one-class problems, in: Pattern Recognition, 2006. ICPR 2006. 18th International Conference on, Vol. 4, IEEE, 2006, pp. 449–452.

[53] N. Poh, J. Kittler, S. Marcel, D. Matrouf, J.-F. Bonastre, Model and score adaptation for biometric systems: Coping with device interoperability and changing acquisition conditions, in: Pattern Recognition (ICPR), 2010 20th International Conference on, IEEE, 2010, pp. 1229–1232.

[54] M. M. Seeger, P. Bours, How to comprehensively describe a biometric update mechanisms for keystroke dynamics, in: Security and Communication Networks (IWSCN), 2011 Third International Workshop on, IEEE, 2011, pp. 59–65.

[55] A. Rattani, D. Kisku, A. Lagorio, M. Tistarelli, Facial template synthesis based on sift features, in: Automatic Identification Advanced Technologies, 2007 IEEE Workshop on, IEEE, 2007, pp. 69–73.

[56] A. Mhenni, E. Cherrier, C. Rosenberger, N. Essoukri Ben Amara, User dependent template update for keystroke dynamics recognition, in: 2018 International Conference on Cyberworlds (CW), IEEE, 2018, pp. 324–330.

[57] A. Mhenni, E. Cherrier, C. Rosenberger, N. Essoukri Ben Amara, Towards a secured authentication based on an online double serial adaptive mechanism of users' keystroke dynamics, in: International Conference on Digital Society and eGovernments (ICDS), 2018, pp. 73–80.

[58] K. Deb, An introduction to genetic algorithms, Sadhana 24 (4) (1999) 293–315. doi:10.1007/BF02823145.
URL https://doi.org/10.1007/BF02823145

[59] R. Giot, M. El-Abed, C. Rosenberger, Greyc keystroke: a benchmark for keystroke dynamics biometric systems, in: Biometrics: Theory, Applications, and Systems, 2009. BTAS'09. IEEE 3rd International Conference on, IEEE, 2009, pp. 1–6.

[60] R. Giot, M. El-Abed, R. Christophe, Web-based benchmark for keystroke dynamics biometric systems: A statistical analysis, in: Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP), 2012 Eighth International Conference on, IEEE, 2012, pp. 11–15.

[61] R. Giot, C. Rosenberger, B. Dorizzi, Hybrid template update system for unimodal biometric systems, in: 2012 IEEE Fifth International Conference on Biometrics: Theory, Applications and Systems (BTAS), 2012, pp. 1–7. doi:10.1109/BTAS.2012.6374539.