Equalization of Keystroke Timing Histograms for Improved Identification Performance

Jugurta Montalvão, Carlos Augusto S. Almeida and Eduardo O. Freire

***Abstract*—The effect of parametric equalization of time interval histograms (key down-down intervals) on the performance of keystroke-based user verification algorithms is analyzed. Four algorithms are used throughout this analysis: a classic one for static (structured) texts, a second one, also proposed in literature, for both static and arbitrary (free) text, a new one for arbitrary text based verification, and an algorithm recently proposed, where keystroke timing is indirectly addressed in order to compare user dynamics. The algorithms performances are presented before and after time interval histogram equalization, and the results corroborate with the hypothesis that the nonlinear memoryless time interval transform proposed here, despite its simplicity, can be a useful and almost costless building block in keystroke-based biometric systems.**

***Index Terms*—Biometry, Keystroke Dynamics, Histogram Equalization.**

I. Introduction

In Biometric based strategies for subject identification and/or verification, static and/or dynamic biometric measures may be used as personal "passwords". Most security systems based on biometric signals demand specific data acquisition hardware. Nevertheless, there are some possible exceptions to this rule. One of them is typing biometrics, more commonly referred to as keystroke dynamics. Indeed, keystroke dynamics looks at the way a person types or pushes keys on a keyboard.

The original technology derives from the idea of identifying a sender of Morse code using a telegraphy key known as the "fist of the sender", whereby operators could identify senders transmitting a message by the rhythm, pace and syncopation of the signal taps (see [1] and references therein).

As early as 1980, researchers (e.g., Gaines et al. [2], in 1980, Umphress and Williams [3], in 1985, and Bleha et al. [4], [5], in 1988 and 1990) have been studying the use of habitual patterns in a users typing behavior for identification. The results from these works showed that, by modeling delays between strokes as a random variable, the correlation between samples from the same subject is high.

Many approaches have been proposed for the task, ranging from mean and covariance based strategies [6] to artificial neural networks [13] (See also [1] for a recent overview that compares the main published approaches).

In [15], we proposed a specific non-linear memory-less transformation for timing histogram equalization, and some experiments with static and free texts were performed with well-known algorithms. In all experiments, the proposed simple timing equalization improved performances, in terms of Equal Error Rate (EER), for all tested algorithms.

Nevertheless, recently, in August 2005, a new algorithm for biometric identification/verification through keystroke dynamics was published in [14], in which metrics are only marginally based on statistical approaches.

In this paper, we resume our former investigation by testing whether our timing equalization method is still helpful in this case.

This paper is organized as follows: first, we explain how data samples are obtained along with the experimental procedure applied during databases construction, in Section II, then we provide a statistical analysis of time intervals, from which, we propose a parametric mapping function to be applied on it, in Section III. In Sections IV and V, respectively, practical results from static- and free-text experiments are presented. Finally, we discuss our results and present some conclusions and perspectives in Section VI.

II. Data Acquisition

Besides the key code itself, time-features from keystroke data can be extracted in many ways, such as down-down, down-up , and up-down time intervals [6].

In this work, four databases were built up with down-down (DD) intervals only. In databases A and B, samples correspond to the DD intervals recorded during the typing of a single set of four fixed words in English (equally spelled in Portuguese, apart from the accent on the word "táxi"): "*chocolate, zebra, banana, taxi.*", while in database C, the DD intervals were recorded during the typing of two fixed words in Portuguese: "*computador calcula.*".

By contrast, in Database D, samples correspond to the DD intervals recorded during the typing of freely typed rows of text (like rows from an e-mail) in Portuguese.

Therefore, a total of 47 subjects were invited to take part in the experiment, 10 in Database A, 8 in Database B, 14 in Database C and 15 in Database D.

In Database A, each subject typed the set of four words 10 times, 5 times (5 samples) during a first session, and 5 more samples during a second session, about a month later. All subjects, men and women not necessarily familiar to a computer keyboard, were invited to type on the very same conventional keyboard in our laboratory (standard 101/102 keys, Brazilian layout - similar to the EUA layout) during both sessions.

The authors are with the Universidade Federal de Sergipe (UFS), São Cristóvão, CEP. 49100-000. E-mail:jmontalvao@ufs.br, augusto_math@yahoo.com.br, efreire@ufs.br.

Database B was built up likewise, but the interval between sessions was shorter: one weak only, all subjects were Electrical Engineering or Science Computer students, and they were free to type on different keyboards in different sessions.

Database C was prepared even more freely: subjects were provided with copies of the sampling program, and they were completely free to type samples wherever and whenever they wanted to. Some of them typed the entire set of 10 samples at once. For this reason, Database C was used for time interval analysis, along with Databases A and B, but during the verification experiments, Database C was discarded.

Finally, in Database D, each subject typed a set of 10 freely typed rows of text with about 110 strokes per row, being 5 rows (5 samples) during a first session, and 5 more rows during a second session, about a week later. Again, subjects were provided with copies of the sampling program, and they were completely free to type samples wherever and whenever they wanted to, not necessarily with the same keyboard.

III. Time Interval Analysis

Let $\mathbf{x} = [x(1)\ \ x(2)\ \ \ldots\ \ x(N)]^t$ be a column-vector of N positive DD intervals, in seconds. Here, for simplicity, we assume that DD intervals are instances of a unique continuous random variable, X, regardless the pressed key and the subject.

Figure 1 shows the intervals histogram from Databases A, B and C — 7653 DD intervals from 32 subjects —, which can be regarded as being a rough approximation of the actual probability density function (pdf) of X.

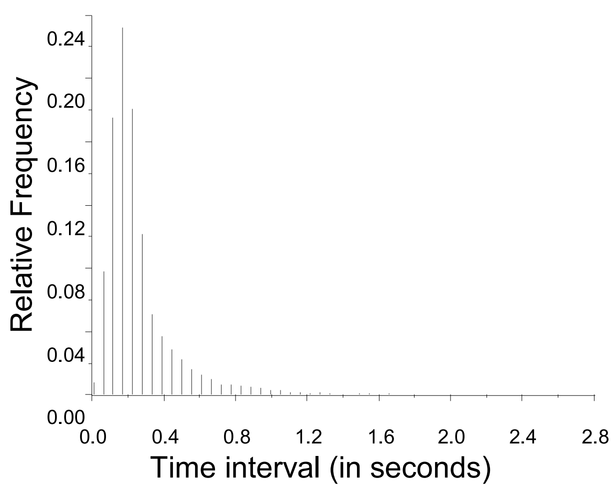

Fig. 1. Histogram of DD intervals in databases A, B and C — 7653 DD intervals from 32 subjects.

It is clear that the relative frequencies of quantized intervals (50 intervals, in Fig.1) are strongly unbalanced. Moreover, as it can be observed in Figure 2, the pdf of

$$Y = \log_e(X) \tag{1}$$

roughly follows a Normal Distribution. Consequently, X is nearly log-normal [1].

[1]An interesting remark is that known empirical data properly modeled by log-normal random variables are, for instance, blood pressures of human beings, the survival times of bacteria in given strengths of disinfectant, and even the number of words in a sentence written by George Bernard Shaw [7].

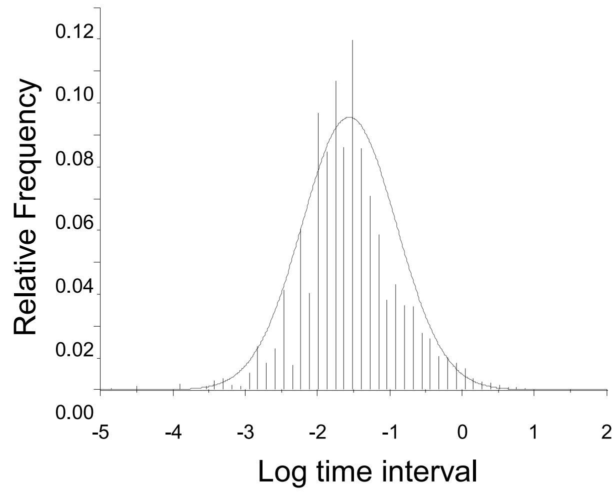

Fig. 2. Histogram of the logarithm of DD intervals.

On the other hand, it is well known, from digital communication and digital image processing, that a suitable nonlinear memoryless transformation applied to a random variable like X (or, equivalently, to Y) can enhance relevant aspects of this variable, as if "zooming" in on more descriptive time-scales, which also corresponds to an Information Maximization Procedure, according to [8].

Indeed, in digital voice coding, for instance, the use of μ-law and A-law companding (compressing/expanding) methods provides noise reduction, whereas in digital image processing, histogram equalization enhances images appearance.

It is also well-known that a suitable non-linear mapping of a continuously valued random variable onto a new one with flat pdf can be obtained from its distribution function (see [8] and references therein). In our case, it is preferable to handle Y instead of X, because Y is nearly normally distributed. Then, by assuming that $Y \sim \mathcal{N}(\mu_y, \sigma_y^2)$, its *cumulative distribution function* is given by:

$$G(y) = \frac{1}{\sqrt{2\pi}\sigma_y} \int_{-\infty}^{y} \exp\left(-\frac{(\xi - \mu_y)^2}{2\sigma_y^2}\right) d\xi$$

where μ_y stands for the mean of Y and σ_y^2 stands for its variance.

Unfortunately, there is no closed analytic form for $G(y)$ in this case [9] whereas, for simplicity of use, we would prefer a parametric model of it. Hopefully, an approximation can be provided by:

$$\tilde{G}(y) = \frac{1}{1 + \exp\left(-\frac{K(y-\mu_y)}{\sigma_y}\right)} \tag{2}$$

where $K = 1.7$ roughly optimizes the approximation in the sense of the minimum squared error integral: $\int_{-\infty}^{\infty}(G(y) - \tilde{G}(y))^2 dy$.

As a consequence of Equations 1 and 2, a straightforward DD interval equalization transform can be:

$$g(x) = \frac{1}{1 + \exp\left(-\frac{K(\log_e(x)-\mu_y)}{\sigma_y}\right)} \tag{3}$$

where $K = 1.7$, $\mu_y = -1.56$ and $\sigma_y = 0.65$ (estimated from databases A, B and C, altogether — 7653 intervals from 32 subjects), and x is given in seconds. It is worth noting that, though the estimated values for μ_y and σ_y are almost the same even when estimated from only one database (A, B or C), we believe (by hypothesis, for a while) that it can suffer a remarkable deviation according to the idiom in which the texts are written and, as a consequence, specific databases could be used to refine the estimation of these parameters. Moreover, such a re-estimation is clearly necessary if another kind of typing event than DD intervals (hold time, for instance) is to be considered.

Figures 3 and 4 show approximated cumulative distributions for X, estimated from databases A and B, respectively, along with the parametric model $g(x)$.

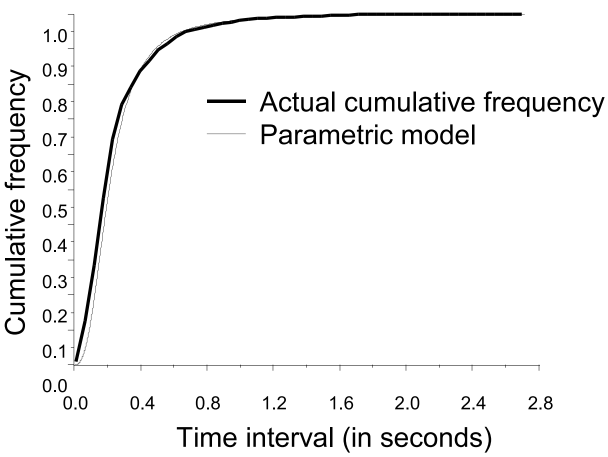

Fig. 3. Cumulative frequency of DD intervals from Database A and the parametric model $g(x)$.

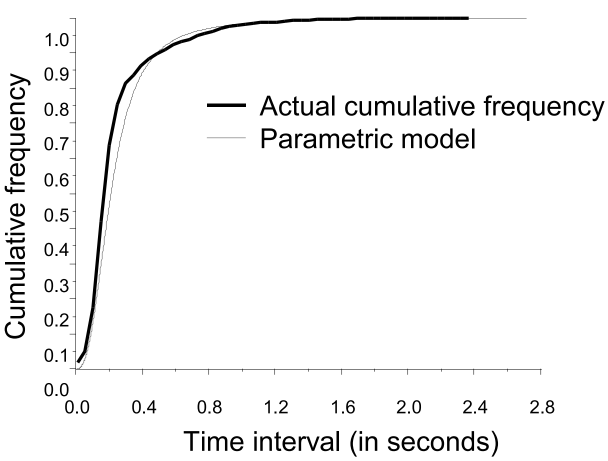

Fig. 4. Cumulative frequency of DD intervals from Database B and the parametric model $g(x)$.

Finally, we highlight that the mapping $U = \alpha g(X) + \beta$, where α and β are arbitrary constants, produces a new dependent random variable U (dependent on X) whose pdf is as close as possible to a flat density between β and $\alpha + \beta$.

IV. Evidences from Experiments with Structured Texts

In this work, we claim that the nonlinear memoryless mapping $g(x)$ of DD intervals from $\Re^+ \mapsto \Re^+$ can significantly improve the performance of any verification algorithm that does not compensate for the unbalanced pdf of X (neither explicitly nor implicitly).

In order to present some empirical evidences of this claimed improvement, we revisit a seminal work on keystroke dynamics by Bleha et al. [5], in which a quite simple comparison strategy of dynamics is applied to static texts. Furthermore, we also investigate the performance change of a more elaborated algorithm proposed by Monrose and Rubin [10], based on weighted probability measures.

Since databases A and B were built up in two sessions: 5 samples per subject during each one, we do simulate an enrollment procedure by using only samples from the first session during user prototype generation, and performing user verification exclusively with samples from the second session, from each database.

For instance, if all 5 samples from each subject, sampled during the first session, are used to produce prototypes, and each prototype is compared to each single sample from the second session, then 500 (10 prototypes $\bar{\mathbf{x}}_i \times 50$ samples $\mathbf{x}_i$) comparisons are carried out on Database A, while 320 (8 prototypes $\bar{\mathbf{x}}_i \times 40$ samples $\mathbf{x}_i$) comparisons are carried out on Database B.

A. First Experiment with Structured Text

In [5], a 30-entries enrollment procedure (username varying in length from 11 to 17 characters) and 2 trials verification (with a “so-called” shuffling procedure from the 2 trials) per subject provided a False Rejection Rate (FRR) of about 2.8 % and a False Acceptance Rate (FAR) of about 8.1 %.

Furthermore, in [5], the decision for user verification is taken according to both minimum distance classifier:

$$D_i(\mathbf{x}) = \frac{(\mathbf{x} - \bar{\mathbf{x}}_i)^t(\mathbf{x} - \bar{\mathbf{x}}_i)}{||\mathbf{x}||\ ||\bar{\mathbf{x}}_i||} < T_1 \tag{4}$$

and the normalized Bayes classifier:

$$d_i(\mathbf{x}) = \frac{(\mathbf{x} - \bar{\mathbf{x}}_i)^t\mathbf{C}_i^{-1}(\mathbf{x} - \bar{\mathbf{x}}_i)}{||\mathbf{x}||\ ||\bar{\mathbf{x}}_i||} < T_2 \tag{5}$$

where i stands for the user (subject) label, $\mathbf{x}$ is a incoming column-vector of time intervals, $\bar{\mathbf{x}}_i$ and $\mathbf{C}_i$ are, respectively, the mean vector and covariance matrix from user i (estimated from its enrollment database), and T_1 and T_2 are preset thresholds.

In our experiment, we are more restrictive, since the very same set of words is imposed to all subjects — note that subjects typing their own names or familiar strings are easier to be distinguished by their dynamics —, and no more than 5 entries are allowed during enrollment, and no shuffling procedure is applied.

Moreover, we do not use Bayes classifier. Note that if few samples are available from enrollment, as in our experiment, it yields bad estimation of covariance matrices. Besides, since all subjects use the same string as password, no normalization is necessary. That is to say that we finally compare dynamics according to Eq. 6, for raw (non-equalized) intervals, and according to Eq. 7, for equalized ones.

$$\delta_i(\mathbf{x}) = ||\mathbf{x} - \bar{\mathbf{x}}_i||^2 < T \tag{6}$$

$$\delta_i(g(\mathbf{x})) = ||g(\mathbf{x}) - \overline{g(\mathbf{x}_i)}||^2 < Tg \tag{7}$$

Table I shows the result of such an experiment with and without equalization of DD intervals, in terms of Equal Error Rate (EER, the operational point for which FAR equals FRR).

TABLE I
STATIC TEXT BASED VERIFICATION WITH BLEHA'S ALGORITHM — 5 ENTRIES PER ENROLLMENT, 1 ENTRY PER VERIFICATION.

Database	Without equalization	With Equalization
A	EER = 32.4%	EER = 6.2%
B	EER = 32.5%	EER = 7.5%

Further, by varying the threshold value in both cases, we obtain simultaneous plots of FAR and FRR versus decision threshold, as it is shown in Figures 5 and 6, for database A, without and with equalization, respectively.

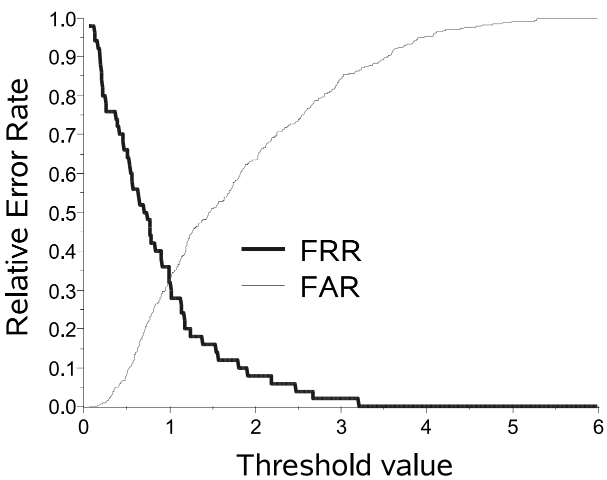

Fig. 5. FAR and FRR from database A, without equalization.

B. Second Experiment with Structured Text

In [10], three new algorithms are proposed, where each subject is associated to a profile containing a set of timing means and standard deviations of features — such as digraphs like $th, st, on, \ldots wy$, for instance. Among the three proposed algorithms, the third one, where profiles are compared through a score based on weighted probability measure (see [10] for more details), provides the best result in terms of identification rate.

Applying this third algorithm to databases A and B, with and without time interval equalization, we obtained the performances shown in Table II, in terms of EER. Here, T and Tg stand for score thresholds with and without equalization, respectively.

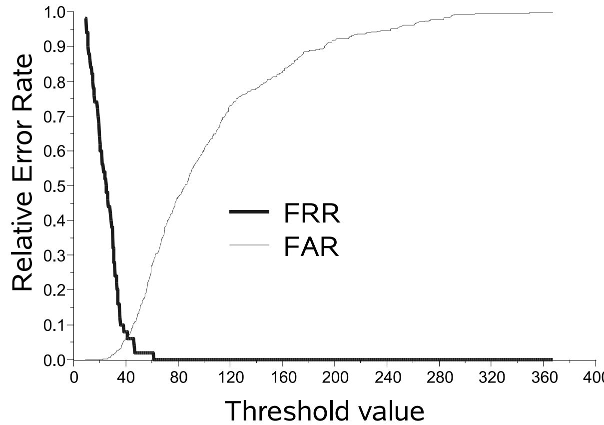

Fig. 6. FAR and FRR from database A, with equalization.

TABLE II
STATIC TEXT BASED VERIFICATION WITH MONROSE AND RUBIN'S ALGORITHM — 5 ENTRIES PER ENROLLMENT, 1 ENTRY PER VERIFICATION.

Database	Without equalization	With Equalization
A	EER = 18.0%	EER = 10.0%
B	EER = 24.6%	EER = 12.5%

V. EVIDENCES FROM EXPERIMENTS WITH FREE TEXT

Database D was also built up in two sessions, with 5 samples per subject from each one. Therefore, we again simulate an enrollment procedure by using only samples from the first session during user prototype generation, and performing user verification exclusively with samples from the second session.

A. First Experiment with Free-Text

Unlike the Bleha's algorithm, the algorithm proposed by Monrose and Rubin, in [10] (see subsection IV-B), is based on the comparison of typing rhythm over sets of features, such as S={th, st, on,... wy}, for instance. As a consequence, it can also be applied to free text based verification/identification tasks.

Table III shows the performance of this algorithm, in terms of EER, when applied to Database D.

TABLE III
FREE TEXT BASED VERIFICATION WITH MONROSE AND RUBIN'S ALGORITHM — 5 ENTRIES PER ENROLLMENT, 1 ENTRY PER VERIFICATION.

Database	Without equalization	With Equalization
D	EER = 28.6%	EER = 19.9%

B. Second Experiment with Free-Text: A New Straightforward Algorithm

In this subsection, we propose a new simple and quite straightforward algorithm for free text based verification. Indeed, simple strategies like that applied by Muramatsu and

Matsumoto[11], for signature verification, and by George and King [12], for speaker identification, have in common that simple 1D and 2D histograms replace stochastic matrices used in Hidden Markov Models (HMM).

Similarly, our algorithm can also be regarded as a rough simplification of a Markov chain model (not hidden) in which quantized time intervals are seen as discrete states. As a result, both transition probability matrices and prior probability vectors [9] are replaced by 2D and 1D histograms, respectively. Such a strong simplification is highly advantageous in terms of computational load, by keeping good performances, as it is reported in [11] and [12].

More precisely, both quantized-intervals histograms (1D histograms) and histograms of transition between quantized-intervals (2D histograms) are obtained as follows. Given a sample vector of N down-down intervals, $x(n)$, $1 \leq n \leq N$, each interval is mapped into one of K labels according to:

$$r(n) = Q(x(n)),$$

for linear (conventional) quantization, were

$$Q(x) = \min_k \{(x - ((k-1)\Delta x + 0.5\Delta x))^2\},\ 1 \leq k \leq K$$

where $\Delta x = 3/K$ since we assume that intervals longer than 3 seconds are to be discarded, and, for nonlinear quantization,

$$r(n) = \mathrm{round}(K \times g(x(n)) + 0.5)$$

where $\mathrm{round}(K \times g(x(n)) + 0.5)$ rounds the argument to a Natural number from 1 to K (note that $0 < g(x) < 1$ for $0 < x < \infty$).

In both cases, every entry $\mathbf{x} = [x(1)\ x(2)\ \ldots\ x(N)]^t$, $x \in \Re^+$ is mapped onto a sequence of labels $\mathbf{r} = [r(1)\ r(2)\ \ldots\ r(N)]^t$, $r \in 1, 2, \ldots, K$, where $N+1$ is the number of strokes.

As a consequence, both 1D- and 2D- histograms, namely $\mathbf{h}$ and $\mathbf{M}$, respectively, can be easily computed as in Equations 8 and 9, respectively,

$$\mathbf{h} = \frac{1}{N}\,[n_1\ n_2 \ldots, n_K]^t \tag{8}$$

where n_i stands for the number of occurrences of label i in $\mathbf{r}$, and

$$\mathbf{M} = \frac{1}{N - \Delta n}\begin{bmatrix} n_{11} & n_{12} & \ldots & n_{1K} \\ n_{21} & n_{22} & \ldots & n_{2K} \\ \cdots & \cdots & \ldots & \cdots \\ n_{K1} & n_{K2} & \ldots & n_{KK} \end{bmatrix} \tag{9}$$

where n_{ij} stands for the number of occurrences of transitions from label i to label j, separated by a gap of Δn (strokes), in $\mathbf{r}$.

When many vectors of labels are to be considered during prototype construction (enrollment), the corresponding streams of labels are just concatenated into a single longer stream, and then histograms are computed according to 8 and 9.

Finally, decisions concerning user verification are based upon the minimum distance between histograms (playing the role of likelihood or log-likelihood in conventional HMM based algorithms):

$$\sum_{n=1}^{K} (\mathbf{h}(n) - \mathbf{h}_i(n))^2 < L_h$$

and

$$\sum_{m=1}^{K}\sum_{n=1}^{K} (\mathbf{M}(m,n) - \mathbf{M}_i(m,n))^2 < L_M$$

where i stands for the user (subject) label, $\mathbf{h}_i$ and $\mathbf{M}_i$ are, respectively, 1D and 2D histograms computed from the long stream of labels corresponding to the concatenation of all streams in the enrollment database of subject (user) i, and finally L_h and L_M are preset thresholds.

For the database D, with $\Delta n = 1$ and $K = 6$, the 2D histogram based algorithms gives an EER = 41.6 % without equalization, and an EER = 12.7 % with equalization.

Further results, including thresholds for EER, are presented in Table IV.

TABLE IV

FREE TEXT BASED VERIFICATION WITH 1D AND 2D HISTOGRAMS — 5 ENTRIES PER ENROLLMENT, 1 ENTRY PER VERIFICATION.

Method	EER Without equalization	EER With Equalization
1D histogram	41.0% ($L_h = 0.0017$)	14.2% ($L_h = 0.026$)
2D histogram	41.6% ($L_M = 0.0047$)	12.7% ($L_M = 0.018$)

We empirically found that a gap of $\Delta n = 1$ provided the best performance with Database D.

C. Third Experiment with Free-Text

In [14], a new approach is proposed based on two measures:

- R_N measures: a kind of edit distance between two streams of keystrokes, where N-graphs (e.g. digraphs in R_2 and trigraphs in R_3) are sorted by their average time interval, thus forming a table of N-graps versus corresponding time intervals. Then, given two streams of keystrokes, the two corresponding tables are compared by summing up the absolute differences between position index of identical n-graps in both tables.
- A_N measures: Let G_1 and G_2 be the same N-graph occurring in two streams of keystrokes, with durations x_1 and x_2, respectively. N-graphs G_1 and G_2 are said similar if $1 < max(x_1, x_2)/min(x_1, x_2) = T$ for some threshold T greater than 1. The A_N distance between two streams of keystrokes, for a certain value of T, is then defined as: $A_N = 1-$(number of similar N-graphs)/(total number of N-graphs shared by the two streams).

It is clear that, in this new approach, only A_N measures are directly based on keystroke timing, and we believe that R_N measures are not to be affected by timing equalization, for it does not modify the positioning of N-graps in timing sorted tables.

On the other hand, we do expect a non negligible improvement of performance associated to A measures if our timing equalization method is applied.

We tested the impact of timing equalization on this approach through experiments with our own free-text database D. Note, however, that since timing equalization corresponds to a logarithm transformation, for equalized intervals, A measures are recalculated as follows: N-graphs G_1 and G_2 are said similar if

$$max(g(x_1), g(x_2)) - min(g(x_1), g(x_2)) \leq T_g$$

instead of $1 < max(x_1, x_2)/min(x_1, x_2) \leq T$, where $T_g \neq T$. For instance, in our experiments, $T = 1.25$ (according to [14]), whereas $T_g = 1.15$, empirically set, like T, but for equalized time intervals. It is important to highlight that, unlike threshold T, T_g can be less that 1. Finally, the A_N distance is: $A_N = 1 -$ (number of similar N-graphs)/(total number of N-graphs shared by the two streams).

Again, by using only 5 samples per subject, from the first session, for user prototype generation (i.e. sorted tables of N-grams), and performing user verification exclusively with samples from the second session, we got the EER shown in Table V, where measures R and A are applied separately.

Please note that we do not use the whole approach proposed in [14] for user verification. Instead, we just focus on the effect of timing equalization on the measures they use.

TABLE V

FREE TEXT BASED VERIFICATION WITH GUNETTI AND PICARDI'S ALGORITHM — 5 ENTRIES PER ENROLLMENT, 1 ENTRY (1 TEXT ROW) PER VERIFICATION. MEASURES R_2 AND A_2 ONLY.

Measure	Without equalization	With Equalization
R_2	EER = 42.9%	EER = 42.9%
A_2	EER = 16.7%	EER = 13.0%

VI. DISCUSSION AND CONCLUSIONS

In this paper, it is claimed that a single memoryless non-linear mapping of time intervals can significantly improve the performance of verification/identification algorithms based on keystroke dynamics.

This claim is based on the hypothesis that the very unbalanced pdf of the random variable that models such intervals reduces the performance of most naif algorithms (naif in the sense that they do not incorporate any kind of explicit or implicit intervals distribution equalization). It was briefly illustrated through practical results from simple static and free text experiments, where only down-down time intervals are considered.

Concerning the experiments with the Gunetti and Picardi's algorithm, the results we got, with and without timing equalization, point to a stronger importance of A measures, whereas, according to the results presented by themselves, R measures are to be more important in user verification tasks.

We believe that our results diverge from theirs because our samples (free-texts units) are very smaller then theirs. That is to say that each sample, in our small database, is just a row of freely-typed text — about 110 characters —, whereas their individual samples are texts 700 to 900 characters long.

Then, it raises a hypothesis to be tested in the future: that measure A plays a more important role than measure R for small samples.

Nevertheless, we presented some evidences of our initial claim. Although the databases used here are quite small, performance improvements in terms of EER, with those databases, seem to fairly illustrate the point raised here, by comparing algorithm performances with and without timing equalization.

Furthermore, in spite of the smallness of the databases, to allow further comparisons between the results reported here and performances of new algorithms, with or without intervals equalization, our databases are available to download at www.ufs.br/biochaves[2].

ACKNOWLEDGMENTS

This work was partially granted by both the *Fundação de Amparo à Pesquisa de Sergipe* (FAP-SE) and the *Conselho Nacional de Desenvolvimento Científico e Tecnológico* (CNPq). We also thank a lot all students and fellows whose keystrokes dynamics were used as samples in this work.

REFERENCES

[1] A. Peacock, X. Ke, M. Wilkerson, Typing patterns: A key to user identification, IEEE Security and Privacy 2 (5) (2004) 40–47.
[2] R. Gaines, W. Lisowski, S. Press, N. Shapiro, Authentication by keystroke timing: some prelimary results, Tech. rep., Rand Corporation (1980).
[3] D. Umphress, G. Williams, Identity verification through keyboard characteristics, in: International Journal Man - Machine Studies, Vol. 23, 1985, pp. 263–273.
[4] S. Bleha, Recognition systems based on keystroke dynamics, Ph.D. thesis, Univ. Missouri - Columbia (1988).
[5] S. Bleha, C. Slivinsky, B. Hussien, Computer-access security systems using keystroke dynamics, IEEE Trans. on Pattern Analysis and Machine Intelligence 12 (12) (1990) 1217–1222.
[6] L. C. F. Araújo, L. H. R. S. Jr., M. G. Lizárraga, L. L. Ling, J. B. T. Yabu-Uti, User authentication through typing biometrics features., IEEE Trans. on Signal Processing 53 (2) (2005) 851–855.
[7] E. Keeping, Introduction to Statistical Inference, Dover Publication, Inc., New York, 1999.
[8] A. J. Bell, T. J. Sejnowsky, An information maximisation approach to blind separation and blind deconvolution, Neural Computation 7 (6) (1995) 1129–1159.
[9] R. Duda, P. Hart, Pattern Classification and Scene Analysis, Wiley-Interscience, New York, 1973.
[10] F. Monrose, A. D. Rubin, Authentication via keystroke dynamics, in: ACM Conference on Computer and Communications Security, 1997, pp. 48–56.
[11] D. Muramatsu, T. Matsumoto, An HMM on-line signature verifier incorporating signature trajectories, in: ICDAR 2003, 2003, pp. 438–442.
[12] M. George, R. King, A robust speaker verification biometric, in: IEEE Security Technology, 1995, 1995, pp. 41–46.
[13] M. Obaidat, B. Sadoun, Verification of computer users using keystroke dynamics, IEEE Trans. on Systems, Man, and Cybernetics 27 (2) (1997) 261–269.
[14] D. Gunetti, C. Picardi, Keystroke analysis of free text, ACM Trans. Inf. Syst. Secur.8 (3), (2005) 312–347.
[15] J. Montalvão, E. O. Freire, On the Equalization of Keystroke Timing Histograms, Pattern Recognition Letters (to appear).

[2]Website's mirror at http://www.infonet.com.br/biochaves/br/download.htm