PRIS 2003

Jean-Marc Ogier and
Eric Trupin (Eds.)

Pattern Recognition in Information Systems

Proceedings of the
3rd International Workshop on
Pattern Recognition in Information Systems,
PRIS 2003
In conjunction with ICEIS 2003
Angers, France, April 2003

Sponsored by the
International Association for Pattern Recognition

An Identity Authentication System Based On Human Computer Interaction Behaviour

Hugo Gamboa[1] and Ana Fred[2]

[1] Escola Superior de Tecnologia de Setúbal,
Campo do IPS, Estefanilha, 2914-508 Setúbal, Portugal
hgamboa@est.ips.pt
[2] Instituto de Telecomunicações Instituto Superior Técnico
IST - Torre Norte, Piso 10,
Av. Rovisco Pais 1049-001 Lisboa Portugal
afred@lx.it.pt

Abstract. Human behaviour has been used in biometrics. In this paper we describe a new behavioural biometric technic based on human computer interaction. We developed a system that captures the user interaction via a pointing device, and uses this behavioural information to verify the identity of an individual. Using statistical pattern recognition technics, we developed a sequential classifier that processes user interaction, according to which the user identity is considered genuine if a predefined accuracy level is achieved, and the user is classified as an impostor otherwise.
We found that the normal user interaction with the computer entails discriminant information, useful for creating a behavioural biometric identity authentication system. The paper presents experimental results revealing that our system can achieve good performance in the collected interaction information.

1 Introduction

It is easy for a human being to find characteristics that enable the recognition of another person. Looking at ones face, or hearing a known voice, are examples of human identification means.

With the establishment of the information society, personal identification systems have gained an increased interest, either for security or personalization reasons. Traditionally, computer systems have based identification procedures on something *one has* (Keys, magnetic cards or chip cards) or something that *one knows* (personal identification numbers and passwords). Biometric authentication or identification systems use something *one is*, creating more reliable systems, more immune to authorization theft, loss or lent.

There are two types of biometric systems that enable the link between a person and his/her identity [7]. Identity verification (or authentication) occurs when a user claims who he is and the system accepts (or declines) his claim. Identity identification (sometimes called search) occurs when the system establishes a subject identity (or fails to do it) without any prior claim.

According to the type of characteristics explored, biometric technics can be divided into physiological and behavioural. A physiological trait tends to be a more stable physical characteristic, such as finger print, hand silhouette, blood vessel pattern in the hand, face or back of the eye. A behavioural characteristic is a reflection of an individual's psychology. Because of the variability over time of most behavioural characteristics, a biometric system needs to be designed to be more dynamic and accept some degree of variability. On the other hand, behavioural biometrics are associated with less intrusive systems, leading to better acceptability by the users. Two examples of behavioural biometric technics are handwritten signature verification [5] and speaker recognition via his voice print [2].

A biometric technic is normally accompanied by some metrics that tries to evaluate the technic performance[8]: False rejection rate (FRR) — rate of accesses where a legitim user is rejected by the system; False acceptance rate — rate of accesses where an impostor is accepted by the system; Equal error rate (EER) — the value at which FAR and FRR are equal.

In this paper we propose a new behavioural biometric technic based on human computer interaction via a pointing device, typically a mouse pointer. The normal interaction through this device is analyzed for extraction of behavioural information in order to link an identification claim to an individual.

We developed a prototype system, accessed via a web browser, that presents the memory game to the user: a grid of tiles is presented, each tile having associated a hidden pattern, which is shown for a brief period of time upon clicking on it; the purpose of the game is to identify the matching tiles (equal patterns). The user interaction through the web page is recorded and is used for the authentication process.

In the following section we present the authentication system, focusing on the feature extraction procedures, feature selection, statistical learning algorithms, and finally is presented the sequential classifier. Section 3 presents experimental results obtained using the collected data. Conclusions are presented in section 4.

2 The Authentication System

An experimental system was developed to verify the possibility of discriminating between users using their computer interaction information, specifically based on mouse movements performed between successive clicks, which we will call a *stroke*.

The overall system is divided into two main parts: (i) the *acquisition system*, that collects user interaction data and stores it in data files; (ii) the *recognition system*, that reads from the interaction files, and classifies the user as imposter or genuine. Figure 1 presents the two part system and its building blocks.

The acquisition system works over the world wide web, installed in a web server. First, it presents a web page to the user, asking for his identification (name, and a personal number). Then the system presents an *interaction acquisition page* with the memory game (that could be any html web page), depicted in figure 2. This page is monitored by an application developed for this purpose[4] (WIDAM, Web Interaction Display and Monitoring), that records all the user interaction in a file stored in the web server.

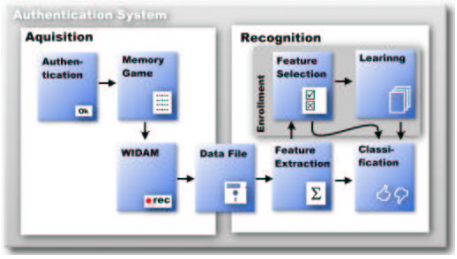

Fig. 1. System architecture.

Fig. 2. Interaction test page: the memory game.

The recognition system reads the interaction data from the stored data files, and starts a feature extraction procedure, by applying some mathematical operations (see section 2.1). The classifier (which is a sequential classifier) receives a sequence of strokes and decides to accept or reject the user as genuine.

The system has an enrolment phase, where the global set of extracted features are used in an algorithm that selects a set of "best"features for each user, using the equal error rate as performance measure (feature selection block in figure 1). The classifier is trained based on a parametric learning procedure, to estimate the probability density functions of the users' data.

2.1 Feature Extraction

The input to the recognition system is the interaction data from the users, recorded using the WIDAM application. The pointing device absolute position, x- and y- coordinates, the event type (mouse moves and mouse clicks), and the time when these events occur, are the information we use for feature extraction.

We consider a pattern in our recognition system as the set of points between two mouse clicks, that we call a *stroke*. In figure 3 we show an example of a stroke, plotting the evolution of the x-y coordinates of the mouse (input vectors) over time. Figure 4 presents the corresponding x-y representation.

Each pattern passes through several processing phases in order to generate the complete set of features. In a preprocessing phase, signals are cleaned from some irregularities via a cubic spline smoothing process. The second phase concerns the extraction of spatial and temporal information, leading to intermediate data representation vectors. A final step generates the features by exploring some statistical information from these vectors, and other general properties of the patterns.

We define six vectors in the spatial domain over the smoothed curve points: x' - horizontal coordinates; y' - vertical coordinates; s' - path distance from the origin; θ - angle of the path tangent at the point with the x-axis ; c - curvature (derivative of θ in order to space, $c = \frac{\delta\theta}{\delta s}$); Δc - derivative of curvature in order to space. The curvature is inversely proportional to the radius of the intrinsic circumference that fits the path at the point where the curvature is being calculated.

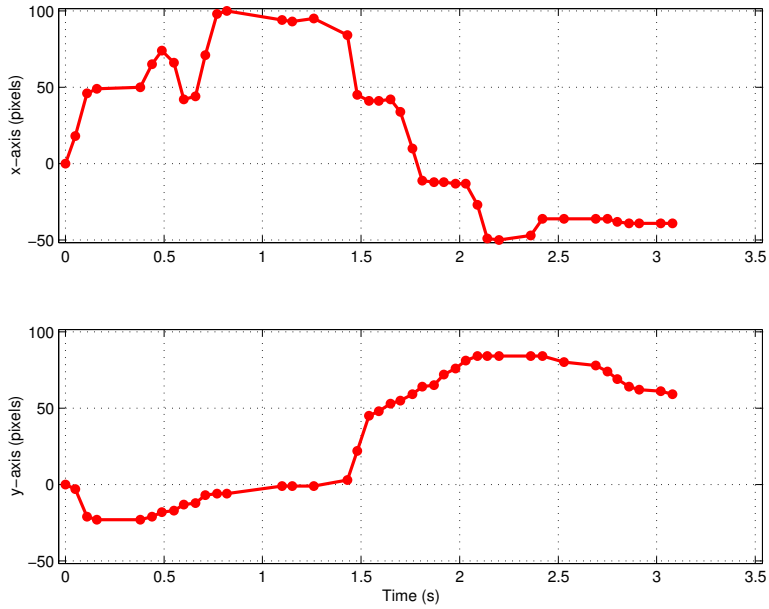

Fig. 3. Input signals generated by the mouse move events.

In the temporal domain we defined 9 vectors, calculated from the original acquired data points: x - the vector with the input $x_i \dots x_n$ values; y - the vector with the input $y_i \dots y_n$ values; t - the input time vector $t_i \dots t_n$; v_x - horizontal velocity; v_y - vertical velocity; v - tangential velocity; $\dot{v}$ - tangential acceleration; $\ddot{v}$ — tangential jerk; w - angular velocity.

After creating the referred vectors we pass to the phase of feature generation. Each stroke is characterized by a 63-dimensional feature vector, X, which contains relevant information for the recognition system.

Feature extraction is based on the spatial and temporal vectors. The vectors x', y', θ, c, Δc, v_x, v_y, v, $\dot{v}$, $\ddot{v}$, and w are statistically analyzed, and 5 values are computed per vector: the *minimum, maximum, mean, standard deviation*, and *range* (*maximum* - *minimum*). Two other features are computed related to the path of the stroke: the *straightness*, defined as the ratio of the Euclidean distance between the starting and ending points of the stroke, and the total path distance; the *jitter*, related to the tremor in the user movement, defined as the ratio between the original path length and the smoothed path length.

The curvature vector c is processed searching for high curvature points, that we call critical points. We search for zeros in the derivative of select the points that have absolute curvature higher than a constant $\alpha = \frac{\pi}{10} \frac{rad}{pixel^2}$; The *number of critical points* constitute an additional feature.

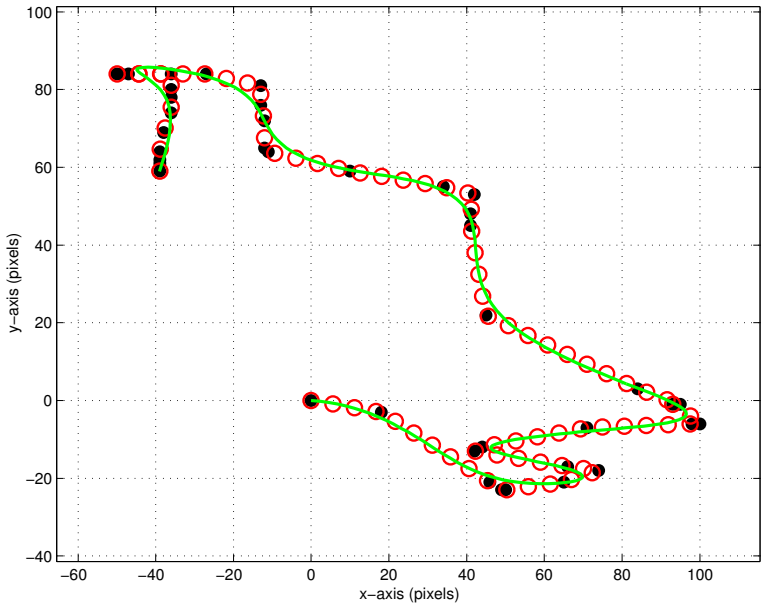

Fig. 4. The x-y representation of the signals. Black dots (•) represent the input sampled points generated by mouse move events. White dots (∘) represent linearly (equidistant) interpolated points. The line represent the smoothed spline interpolation.

We consider a pause in the user interaction when two consecutive events have a time interval greater than a constant β=0.1 sec. We compute the *time to click*, the *number of pauses*, the *paused time*, and the *paused time ratio* (the ratio between paused time and the total time of the stroke).

2.2 Feature Selection

The total set of features generated by the feature extraction procedure is analyzed in order to select a subset of features that "best"discriminate among the several users.

For the purpose of feature selection we consider that we have a classifier system that receives a subset of features and returns the equal error rate of the system. The classifier will be explained in the subsection 2.4.

In a first approach we searched for a set of features that could be used for all users. We used a greedy search[10, 11] algorithm, typically called Sequential Forward Selection (SFS) [6] that selects the best single feature and then adds one feature at time to a the vector of previously selected features. The algorithm stops when the equal error rate does not decrease.

The algorithm is as follows:

1. Create an empty feature subset f_{subset}.
2. Initialize the best equal error rate of the previous interaction, $EER_{last} = 1$.
3. For each feature $f_i, \ \ i = 1...n_{features}$:
   (a) Create the vector with the features to test, $f_{test} = f_{subset} \cup f_i$.
   (b) Set the feature equal error rate (f_{EER_i}) equal to the result of the recognition system test, using the subset f_{test}. f_{EER_i}= TEST(f_{test}).
4. If $\min_i f_{EER_i} > EER_{last}$ exit and return f_{subset}.
5. Set $EER_{last} = \min_i f_{EER_i}$.
6. Set the best feature $f_{best} = \arg\min_i f_{EER_i}$.
7. Set $f_{subset} = f_{test} \cup f_{best}$.
8. Go to 3.

In a second approach we used the same search algorithm but using separate features vectors for each user. (feature selection tuned to the user).

2.3 Learning

The classification rule assumes a statistical model for the feature vectors. The learning phase consists of the estimation of the probability density functions, $p(X)$ (where X is the feature vector of a stroke), from each user's data. Considering that each user constitutes a recognition class, and assuming statistical independence between features, $p(X)$ factorizes into $p(X|\ user) = \prod p(x_i|\ user)$. We use as parametrical model for $p(x_i|\ user)$ the *weibull*[1] distribution ($p_{weibull}(x|a,b) = abx^{(b-1)}e^{(-ax^b)}$). The *weibull* distribution, given a data transformation, can approximately fit several distributions, such as the exponential distribution (when $b = 1$) and normal distribution (when $\mu \gg 0$ with μ equal to the mean of the distribution). In order to adjust the distribution parameters to the data we transform each feature x_i using equations 1, 2, and 3.

$$skewness = \frac{E(x_i - \mu)^3}{\sigma^3} \tag{1}$$

$$\textbf{if } skewness < 0 \Rightarrow x_i = -x_i \tag{2}$$

$$x_i = x_i - \min(x_i) \tag{3}$$

Given the data from one user and one feature, maximum likelihood estimates of the parameters a and b are obtained.

2.4 Sequential Classification

The classifier's purpose is to decide if a user is who he claims to be, based on the patterns of interaction with the computer.

We consider that the i^{th} user is denoted by the class $w_i, \ \ i = 1, \ldots, L$, and L is the number of classes. As defined before, a feature vector is associated with one stroke, the user interaction between two mouse clicks. Given a sequence of n_s consecutive strokes executed by the user, w_i, interaction information is summarized in the vector $\mathbf{X} = X^1 ... X^{n_s}$, consisting of the concatenation of the feature vectors associated with

each stroke. $X^j = x_1^j ... x_{n_{f_i}}^j$, the feature vector representing the jth stroke, has n_{f_i} elements, n_{f_i} being the number of features identified for user w_i in the feature selection phase.

Considering each stroke at a time, and assuming statistical independence between features, we can write $p(X_j|w_i) = \prod_{l=1}^{n_f} p(x_l^j|w_i)$. Considering stroke independence we can further write $p(\mathbf{X}|w_i) = \prod_{j=1}^{n_s} p(X_j|w_i)$.

The classifier will decide to accept or reject the claimed identity based on two distributions: the genuine distribution $p(\mathbf{X}|w_i)$, and the impostor distribution $p(\mathbf{X}|\ \overline{w_i})$ that is based on a mixture of distributions (weibull distributions), one for each other user not equal to i, expressed as $p(\mathbf{X}|\overline{w_i}) = \sum_{j \neq i} p(\mathbf{X}|w_i)\frac{1}{L}$. In the previous equation we assume that the classes are equiprobable, $p(w_i) = 1/L \quad i = 1...L$. We can therefore express the posterior probability function as $p(w_i|\mathbf{X}) = \frac{p(\mathbf{X}|w_i)}{\sum_{k=1}^{L} p(\mathbf{X}|w_k)} = 1 - p(\overline{w_i}|\mathbf{X})$.

Since $p(w_i|X_j)$ represents an estimate of the probability of the classification being correct, we establish a *limit*, λ, to select one of the decisions, using the decision rule in equation 4. To present result about the classifier performance we adjust λ to operate in the equal error rate point.

$$Accept(\mathbf{X} \in w_i) = \begin{cases} true & \text{if } p(w_i|\mathbf{X}) > \lambda \\ false & \text{otherwise} \end{cases} \tag{4}$$

3 Results

We asked 25 volunteers (engineering students) to use the developed system, playing several memory games during about 10-15 minutes. This way, we created an interaction repository of approximately 5 hours of interaction, providing more than 180 strokes per user. In order to use the same number of strokes per user in the tests performed, we randomly selected 180 strokes from each user. The set of strokes was divided into two equal parts, one for the training phase and other for the testing phase. Using the training set we learnt the parametrical distribution $p(x_i|user)$ for each user and each feature. Feature selection was based in the same data set and was tuned for each user. Feature selection (see section 2.2) was based on the performance of the classifiers using sequences of 10 strokes.

When testing the system for one user, we consider an imposter as one of the other users. The test function returns the equal error rate given N sequences of strokes of length l using the classifier tuned for user i. The input sequence of strokes of a test is composed of $N/2$ strokes randomly sampled from the testing set of the user, and $N/2$ strokes randomly sampled from the testing sets of all the other users.

One of the free variables of the system is the number of strokes that the system will use in the verification task. Bootstrap [3] estimates of the system performance as a function of the sequence stroke length (from 1 to 100 strokes) was obtained using 10000 bootstrap samples from the test set. The mean duration of a stroke is approximately 1 second. The values associated with the test using 10 strokes requires approximately 10 seconds of interaction. In table 1 we present the mean results of the equal error rate for all 25 users for several stroke sequence lengths. A graphical display of these results is shown in figure 5. As shown, the mean value and the standard deviation of the

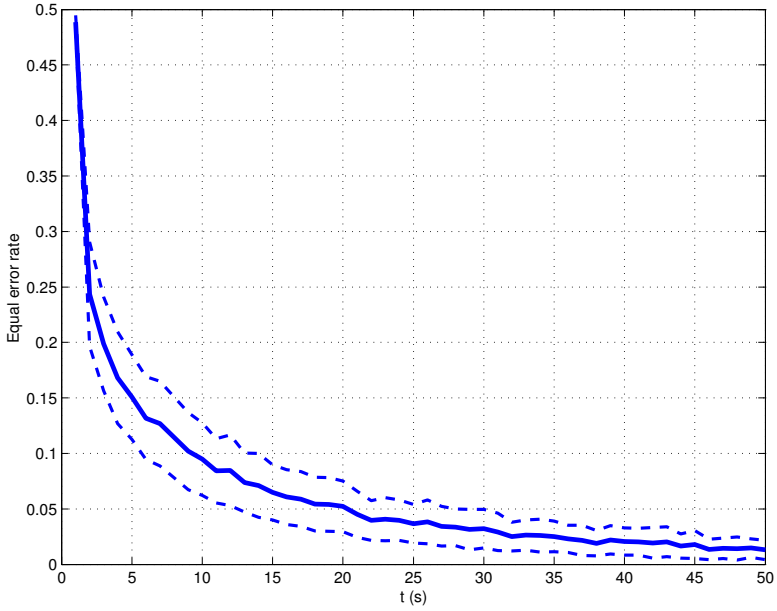

Fig. 5. Equal error rate results of the verification system. The solid line is the mean of the equal error rate of all users for different stroke length. The dashed lines are the mean plus and minus half standard deviation.

EER progressively tends to zero as more strokes are added to the decision rule. This illustrates the refinement of the performance obtained by the sequential classifier.

Table 2 presents EER values reported in the literature for several biometric techniques [9]. Preliminary results show that the proposed user authentication system, based on behavioural information extracted from the interaction with the computer, can achieve comparable or even better performances than other biometric techniques.

4 Conclusion

We have presented a novel user verification technique based on behavioural biometrics, extracted from human-computer interaction through a pointing device. For the implementation of the proposed technique, a prototype system, working on the world wide web, was developed. This system comprises a data acquisition module, responsible for the collection of user interaction data, and a recognition module, that produces the user classification and estimates of the performance of the decision rule.

l	eer	std
1	0.489	0.01
2	0.243	0.09
5	0.151	0.07
10	0.095	0.06
20	0.052	0.04
50	0.013	0.02
100	0.005	0.001

Table 1. Mean equal error rate (eer) and respective standard deviation (std) for different stroke sequence lengths (l).

Biometric technic	Equal error rate
Retinal Scan	1:10 000 000
Iris Scan	1:131 000
Fingerprints	1:500
Hand Geometry	1:500
Signature Dynamics	1:50
Voice Dynamics	1:50
$30s$ of User Interaction	1:50
$60s$ of User Interaction	1:100
$90s$ of User Interaction	1:200

Table 2. Comparison between several biometric technics

The user authentication method applies a statistical classifier to the sequential data produced along the interaction. A large set of features were initially extracted from the collected data, using both time domain related and spatial information from the mouse movement patterns. This initial set was then reduced to a small number of features by applying a feature selection algorithm. Using a greedy search, and taking the classifier performance, measured by the EER, as objective function, feature selection was tuned for each user. A sequential classifier was then designed to decide on the authenticity of the identity claim of users, based on the selected features.

Preliminary results show that the proposed technique is a promising tool for user authentication. Furthermore, it is an inexpensive authentication technique, that uses standard human-computer interaction devices, and remotely accesses user behavioural information through the world wide web.

References

1. Robert B. Abernethy. *The New Weibull Handbook*. Robert B. Abernethy, 2000.
2. Mohamed F. BenZeghiba, Hervé Bourlard, and Johnny Mariethoz. Speaker verification based on user-customized password. Technical Report IDIAP-RR 01-13, Institut Dalle Molle d'Intelligence Artificial Perceprive, 2001.
3. Bradley Efron and Robert J. Tibshirani. *An Introduction to the Bootstrap*. Chapman & Hall, 1993.
4. Hugo Gamboa and Vasco Ferreira. WIDAM - Web Interaction Display and Monitoring. Accepted for publication on the 5^{th} International Conference on Enterprise Information Systems.
5. J. Gupta and A. McCabe. A review of dynamic handwritten signature verification. Technical report, James Cook University, Australia, 1997.
6. Anil K. Jain, Robert P. W. Duin, and Jianchang Mao. Statistical pattern recognition: A review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(1), 2000.
7. Vaclav Matyas Jr and Zdenek Riha. Biometric authentication systems. Technical report, ECOM-MONITOR, 2000.
8. Tony Mansfield and Gary Roethenbaugh. 1999 glossary of biometric terms. Technical report, Association for Biometrics, 1999.

9. Thomas Ruggles. Comparison of biometric techniques. Technical report, California Welfare Fraud Prevention System, 2002.
10. Stuart Russell and Peter Norvig. *Artificial Intelligence: a modern approach*. Prentice Hall, 1995.
11. Wojciech Siedlencki and Jack Sklansky. *Handbook of Pattern Recognition and Computer Vision*, chapter On Automatic Feature Selection. World Scientific, 1993.